கணிப்பொறி ஒருங்கும்
பேசிக் மொழியும்

தமிழ்ப் பல்கலைக் கழகம்
தஞ்சாவூர்.

கணிப்பொறி ஒருங்கும் பேஸிக் மொழியும்

COMPUTER SYSTEMS AND PROGRAMMING IN BASIC

ஆசிரியர்

கா. செ. செல்லமுத்து, பி.இ (ஆனர்சு), எம்.இ.,
இணைப் பேராசிரியர், கணிப்பொறியியல்
தமிழ்ப் பல்கலைக் கழகம்
தஞ்சாவூர்

தமிழ்ப் பல்கலைக் கழகம்
தஞ்சாவூர்

இந்நூலின் பின் அட்டையில் உள்ள எழுத்தமைப்பு
தமிழ்ப் பல்கலைக் கழகக்
கணிப்பு மையத்தால்
வடிவாக்கப்பட்டது.

இந்நூலின் எப்பகுதியும் ஆசிரியரின் இசைவின்றி எவரும் அச்சிடவோ வேறு வகையில் வெளியிடவோ கூடாது.

தமிழ்ப் பல்கலைக் கழக வெளியீடு: **26**

தி. ஆ. 2016 புரட்டாசி	:	செப்தம்பர் 1985
ஆசிரியர்	:	**கா. செ. செல்லமுத்து**
நூல்	:	**கணிப்பொறி ஒருங்கும் பேசிக் மொழியும்**
பதிப்பு	:	**முதற்பதிப்பு**
விலை நூலகப் பதிப்பு	:	உரு. 50.00
மாணவர் பதிப்பு	:	
அச்சு	:	தமிழ்ப் பல்கலைக் கழக மறுதோன்றி அச்சகம் தஞ்சாவூர்.

அணிந்துரை

வ. அய். சுப்பிரமணியம்,
துணைவேந்தர்
தமிழ்ப் பல்கலைக் கழகம், தஞ்சாவூர்.

கணிப்பொறி பற்றிய விளக்க நூல் இன்று வெளியாகின்றது. இதற்குமுன் சில நூல்களும் பல கட்டுரைகளும் வெளியாகியுள்ளன. ஆனால் கணிப்பொறியைப் பயன்படுத்தி வகுப்பு நடத்தும் ஆய்வாளர் ஒருவர் செய்த முதல் நூல் இது. அளவிற்பெரிய சிக்கல் களுக்குத் தீர்வுகாண மனித நினைவாற்றலால் இயலாது. எந்திரங்களின் குறிப்பாகக் கணிப்பொறியின் துணை மிகவும் தேவை. சந்திர மண்டலத்திற்கு மனிதனை அனுப்பியதற்கும், விண்வெளிப் பயணம் முதலியவற்றிற்கும் கணிப்பொறியின் துணை சிறிய அளவினதன்று.

மொழிச் சிக்கலைத் தீர்த்தற்கும் இன்று கணிப்பொறியை நம்பியிருக்கும் நிலை ஏற்பட்டுள்ளது. எந்திர மொழிபெயர்ப்பில் பல நாடுகள் கணிப்பொறி வழி முயன்று வருகின்றன. இதில் வெற்றியும் பெற்றுள்ளன. தோல்வியும் அனுபவித்துள்ளன. அவற் றிற்குக் கணிப்பொறி காரணமன்று. இயற்கை மொழியும் அதன் பொருள் நிலையும் பல அடுக்குகளால் ஆனவை. தவளைப் பாய்த்தாக நடைபோடுபவை. தொடர் நடையிட்டுக் காரண காரியத் துடன் ஒவ்வொரு நடையாக இயங்குவது கணிப்பொறி. மொழி யாகிய உள்ளீட்டை இவ்வாறு திருத்தி வகை செய்யமுடியுமாயின் இன்று எந்திர மொழிபெயர்ப்பு வெற்றி பெறும்.

ஏனைய மொழித் துறைகளில் கணிப்பொறியின் உதவி அளவிடற்கரியது. திருக்குறளின் யாப்பமைதியை, சொல்லடைவு, பொருளடைவு, எழுத்துவகை முதலியவற்றைக் கணிப்பொறி வழியாக்கி, திரு கா. செ. செல்லமுத்து அடுத்து வெளியிட விருக்கிறார்.

உரோமன் எழுத்தில் தமிழ் எழுத்தை மாற்றுவதும், எதிர் மாற்றுவதும், அவற்றை மின்மத் தட்டச்சு எந்திரத்தில் இணைத்து அச்சேற்ற உருவாக்குவதும் இப்போது இயலும். இதன்வழி அச்சேற்றுவது வேகம் பெறுகிறது. அது மட்டுமன்றி, தமிழ் எழுத்தில் எழுதினால் அதனைக் கன்னடம், மலையாளம், வங்காளம் தேவநாகரி எழுத்தில் சில மணித்துளிகளில் பெயர்த்திட முடியும். சிறிய கணிப்பொறியில் இந்த எழுத்துமாற்று முறையை உள்ளடக்கினால், நம் நாட்டின் எப்பகுதிக்குச் சென்றாலும், அங்குள்ள எழுத்து முறையை அது தமிழாக்கித் தரும்.

இந்தியாவில் எழுத்து மாற்றம் துயரம் தரும் சிக்கலாக இனி மேல் இராது. பல எழுத்துமுறைகளை உடைய இந்தியாவிற்கு இது பெரும் உதவியாக அமையும். அறிவியலால் கணிப்பொறியாளர் மொழி/எழுத்துச் சிக்கல்களுக்குத் தீர்வு கண்டிட முடியும். திரு கா. செ. செல்லமுத்து இந்தத் துறையில் உழைத்துக் கணிசமாக முன்னேறியுள்ளார். அதைப்போன்று எந்திர மொழி பெயர்ப்பில் வெற்றிகண்டால் மொழிகள் மக்களுக்குப் பெரும் பளுவாக அமையாமல் செய்தி வருதலுக்கு உரிய கருவியாக மாறி மகிழ்விக்கும். அந்த நாள் நெடுந்தொலைவில் இல்லை. உலக அரங்கிற்குக் கணிப்பொறி வழங்கும் பெரும் பரிசாக அது அமையும்.

'தேவையே எல்லா வகையான புதுமையாக்கங்களுக்கும் தாய்' தமிழில் நூல் வேண்டும் என்ற தேவை எழுந்தபோது விரைவில் சீரான இந்த நூல் உருவாகியுள்ளது. மேலும் செப்பமுற அறிஞர் ஆதரவும் ஆக்கக் கருத்துகளும் துணை செய்யும்.

இங்ஙனம்,

வ. அய். சுப்பிரமணியம்

முகவுரை

காலந்தோறும் மாறிவரும் நமது சமுதாய, பொருளாதார அமைப்பில் அறிவியலின் பயனால் படைக்கப்பட்ட கருவிகள் பற்பல. அவற்றில் கணிப்பொறியும் ஒன்றாகும். கல்வி, மருத்துவம், பொறியியல், விண்வெளிப்பயணம், பாதுகாப்புப் படை, தொழில், ஆராய்ச்சி முதலான பல துறைகளின் முன்னேற்றத்திற்குக் கணிப்பொறி ஓர் இன்றியமையாச் சாதன மாக விளங்கி வருகிறது. மேற்கண்ட பல துறைகளில் கையாளப் படும் அளவற்ற செய்திக்கூறுகளைக் குறிப்பிட்ட காலக்கெடுவில் பகுப்பாய்வு செய்து விடை காணவும் தொகுக்கவும் பகுத்து வகைப்படுத்தவும் கணிப்பொறியால் மட்டுமே இயலும் என்று உறுதியாகக் கூறலாம். முன்னேறியுள்ள மேலை நாடுகளில் கணிப்பொறியியல் துறையில் பல வியத்தகு சாதனைகள் செய்யப்பட்டுள்ளன. நமது பாரதத்திலும் படிப்படியாக இந்தத் துறையில் முன்னேற்றங்கள் நிகழ்ந்து வருவதை நாம் காண்கிறோம்.

கணிப்பொறி — ஓர் இன்றியமையாக் கருவி

கணிப்பொறியைத் திட்டமிட்டுப் பயன்படுத்தினால் சிறந்ததொரு சமுதாய மறுமலர்ச்சியை உருவாக்கமுடியும். இருபதாம் நூற்றாண்டைச் சேர்ந்த நாம் ஒவ்வொருவரும் கணிப்பொறியைப் பற்றிய அடிப்படைச் செய்தியையாவது தெரிந்திருத்தல் அவசியமாகும். ஏனெனில், கணிப்பொறிகள் நமது வாழ்க்கையுடன் பின்னிப் பிணைந்த ஓர் அங்கமாகத் திகழப்போகும் நாள் வெகு தொலைவில் இல்லை. தற்காலத்தில் செலவில் மிகவும் குறைந்ததும் செயலாற்றல் திறன் மிக்கதுமான மிகச்சிறிய கணிப்பொறிகள் [Micro Computers] கிடைக்கக்கூடிய வசதிகள் பெருகியுள்ளன. மற்றும் மனைகளில் உணவுப் பட்டியல் தயாரித்தல், ஒலி—ஒளி விளையாட்டு [Video Game], முதலான பணிகளுக்குப் பயன்படும் மனைக் கணிப்பொறிகள் [Home Computers] எளிதில் கிடைக்கும் நிலை ஏற்பட்டுள்ளது. 1950-ஆம் ஆண்டில் உருவாக்கப்பட்ட கணிப்பொறிகள், அமைப்பில் பெரிய அளவிலான கருவியங்களைக் [Hardware] கொண்டு அமைந்திருந்தன. குறைந்த நினைவகம், வழியமைப்பில் சிரமம், வேகக்குறைவு, குறிப்பிட்ட சில பணிகளை மட்டுமே செய்யக்

கூடிய வசதி முதலியன இதன் குறைபாடுகளாகக் கருதப்பட்டது. ஆனால் தற்காலத்தில் உருவாக்கப்படும் மிகச்சிறிய கணிப்பொறிகள் உருவ அமைப்பில் மிகவும் சிறியவை. இடம்விட்டு இடம் எடுத்துச் சென்று பயன்படுத்த உகந்தவையாகும். ஒரு விநாடியில் பல்லாயிரம் பணிகளைப் பாங்குடன் செய்துமுடிக்கும் மிகைவேக வல்லமை பொருந்தியவை. வேகம் மட்டுமன்றித் தேவைக்கு ஏற்ற நினைவகத்தையும் பற்பல துறைகளின் பயன்பாட்டிற்கு ஏற்ற கணிமங்களையும் [Software] தன்னகத்தே பெற்றுள்ளன. நுண்ணியக்கிகளைக் [Micro-Processors] கொண்டுஉருவாக்கப் படும் இவைபோன்ற கணிப்பொறிகள் குறைந்த பொருட்செலவில் கிடைப்பதால் தேவையுள்ள ஒவ்வொருவரும் வாங்கிப் பயன் பெறும் வாய்ப்பு ஏற்படுகிறது. வருங்காலத்தில் கல்வித்துறையில் கணிப்பொறி புகுத்தப்படும் பொழுது, கல்வி கற்றலிலும் கற்பித்தலிலும் குறிப்பிடத்தக்க மறுமலர்ச்சி ஏற்படுமென உறுதியாக நம்பலாம்.

அழகு தமிழில் அறிவியல் நுட்பம்

கணிப்பொறியைப் பற்றிய பயிற்சி வகுப்பைத் தாய்மொழியில் நடத்தவேண்டுமென்ற முதல் முயற்சிக்குத் துணைசெய்யும் வகையில் இந்நூல் எழுதப்பட்டுள்ளது. தமிழ் மொழியில் அறிவியற் கருத்துக்களை, குறிப்பாகக் கணிப்பொறியியலைப் பற்றிக் கூறுமுகத்தான் ஏராளமான கலைச்சொற்களைப் புதிதாக உருவாக்க வேண்டியுள்ளது. எனவே, முதன் முதலில் படிக்கும் பொழுது அவை புதிதாகத் தோன்றலாம். புதிய கலைச்சொற்கள் பழக்கத்திற்கு வரும்வரை அருகில் அடைப்புக் குறிக்குள் வழங்கப்பட்டுள்ள ஆங்கிலச் சொற்கள் கை கொடுத்து உதவும் ஆதலால் படிப்போர்க்கு இந்த இடையூறு நீங்கப்பெறும். ஆனால் பழக்கத்தில் அவை யாவும் நமது மனதில், வழக்கில் நீங்கா இடம் பெறும் என நம்புகிறேன்.

''கணிப்பொறி ஒருங்கும் பேஸிக் மொழியும்'' (Computer systems and programming in BASIC) என்ற இந்த நூல் கணிப்பொறி பற்றியும் பேஸிக் மொழி பற்றியும் தமிழ்ப் பல்கலைக் கழகக் கணிப்பு மையத்தால் நடத்தப்பெற்ற நான்கு முழுநேரப் பயிற்சி வகுப்புகளுக்குப் பாடமாயமைந்தது. கணிப்பொறியியலில் உருவாக்கப்பட்டுள்ள கலைச் சொற்கள், தாய்மொழியில் இப் பாடத்தைப் பயில்வோர்க்குச் சற்றுக் கடினமாக இருக்குமோ

என்ற ஐயப்பாடு முதலில் இருந்தது. ஆனால், பயிற்சி வகுப்பில் பயின்ற மாணவர்கள் வெகு எளிதில் பாடக்குறிப்புகளைப் புரிந்துகொண்டு, கலைச் சொற்களைப் பயன்படுத்தியமை அந்த ஐயப்பாடு தேவையில்லை என்பதை அறிவுறுத்தியது.

''புத்தம் புதிய கலைகள் - பஞ்ச
பூதச் செயல்களின் நுட்பங்கள் கூறும் ;
மேன்மைக் கலைகள் தமிழினில் இல்லை''

என்ற நிலை இல்லாதொழிய வேண்டும்; பொறியியலின் நுட்பங்களையும் மருத்துவத்தின் மேன்மைகளையும் தாய்மொழியில் பயிற்றுவித்தலே சாலச்சிறந்த பணியென்ற கருத்து கல்வியாளர்கள் அனைவரும் முழுமனதுடன் ஏற்றுக்கொண்ட ஓர் உண்மை ஆகும்.

''சொல்லவும் கூடுவதில்லை - அவை
சொல்லுந் திறமை தமிழ் மொழிக்கில்லை'' என்று

தங்களால் இயலாத் தன்மையை மொழிமீது பழியுரைத்துக் கூறுவோர் பேதையர்களாவர்; செயற்பாட்டில் சோர்வும் மொழியுணர்வுமற்ற அறிவிலிகளே இங்ஙனம் உரைப்பர் எனச் சாடுகின்றார் மகாகவி பாரதியார். பாரதியின் கூற்று தீர்க்கமான ஒன்று என்பதில் சிறிதும் ஐயமில்லை.

பழம்பெருமை பேசுதல் தவிர்த்தல் வேண்டும். பாரெங்கும் காணும் கலை, தமிழில் வேண்டும். அதற்குப் பங்காற்ற ஒவ்வொரு தமிழனும் சூளுரை மேற்கொள்ள வேண்டும்.

''கருத்துலகப் பூம்பொழிலின் விரிவில் கோடி
கலைமலர்தல் காணீரோ; அவைய னைத்தும்
திருத்தமிழிற் கண்டீரோ, அணுயு கத்தைச்
செந்தமிழிற் பார்த்தீரோ; மொழிவ ளர்க்க
மருத்துவரின் பொறிவலரின் அறிவி யல்நூல்
வல்லுநரின் துணைதேட வழிசெய் தீரோ!''

என ஆழமாகவும் அழகாகவும் வினாவினைத் தொடுத்துத் தமிழ் வளர்ப்போரைத் தட்டியெழுப்புகிறார் கவிஞர் குலோத்துங்கன் அவர்கள். செய்தவை ஒன்றுமில்லை; செய்ய இருப்பவை மலையென நம்முன் நிறைந்துள்ளன. அதற்குரிய பெருந்திட்டங்களும் ஆக்கங்களும் உருப்பெறுதல் இன்றியமையாததாகும். அது தமிழராய ஒவ்வொருவரின் கடமையுமாகும்.

வளமிக்க தாய்மொழியாம் தமிழ்மொழியில் வாழ்வியற் கோட்பாடுகளும் பொறியியலும் தொழில் நுட்பம் நிறை கணிப்பொறியியல் முதலிய பல துறைகளும் பயிற்றுவிக்கப்படல் வேண்டும். அதற்கான முயற்சிகள் பல மேற்கொள்ளப்படல் இன்றைய உடனடித் தேவையாகும். இதில் தமிழரான ஒவ்வொரு வருக்கும் கணிசமான பங்குண்டு. கல்லூரிகளிலும் பல்கலைக் கழகங்களிலும் இப்பாடங்களைத் தாய்மொழிவழி பயிற்றுவிக்கும் முதற்கட்டமாகத் திறம்படப் பல நூல்கள் தாய்மொழியில் இயற்றுவித்தல் மொழிக்குத் தகைமை சேர்க்கும் தக்கதொரு பணியாகும். இவ்வகையில் தோன்றிய ஆர்வத்தில் எழுதப் பட்டதே கணிப்பொறி பற்றிய இச்சிறிய நூலாகும்.

நூல் அமைப்பு

இந்த நூல் இரண்டு பிரிவுகளாகப் பிரிக்கப்பட்டு, ஒன்பது பாகங்களை உள்ளடக்கியுள்ளது. முதற்பிரிவில் கணிப்பொறி பற்றியும் [Computer System], இரண்டாம் பிரிவில் பேஸிக் மொழி பற்றியும் விவரிக்கப்பட்டுள்ளன. BASIC என்பது - Beginners All purpose Symbolic Instruction Code என்பதன் சுருக்கமாகும்.

முதல் இரண்டு பாகங்களில் கணிப்பொறி என்றால் என்ன? அதன் தோற்றம், வரலாறு, அதன் முன்னேற்றம், தற்காலத்தில் கணிப்பொறியின் வடிவமைப்பு முதலான செய்திகள் கூறப்பட்டு உள்ளன. கணிப்பொறியின் இடுவரல் [Input], விடுவரல் [Output] ஒருங்குகள், வழியமைப்பு முறைமை [Programming Methodology] முதலியவை பற்றிப் பாகம் 3, 4-இல் விவரிக்கப் பட்டுள்ளன. பாகம் 5-இல் கருவிய உறுப்புக்களான செயலகம், நினைவகம், மின்காந்த நாடாப்பொறி, மின்காந்தத் தட்டுப் பொறி முதலியவை பற்றிய விளக்கங்களும் பாகம் 6-இல் கணிப் பொறியின் பல்வேறு பயன்கள் பற்றியும் கூறப்பட்டுள்ளன. ஒவ்வொரு பாகத்திலும் உரைநடையில் கூறப்படும் கருத்துக்களை விளக்குவதற்கேற்ற வகையில் படங்கள் தரப்பட்டுள்ளன.

பேஸிக் மொழியின் அமைப்பு, அதில் பயன்படுத்தப்படும் மாறிகள், மாறிலிகள் [Variables and Constants] மற்றும் சூத்திரங்களும் சமன்பாடுகளும் [Formula and Equations] பற்றிய விளக்கமான செய்திகள் பாகம் 7-இல் விளக்கப்பட்டு

உள்ளன. பேஸிக் மொழியில் பயன்படுத்தப்படும் பலவகை ஆணைகளைப் [Statements] பற்றியும், பேஸிக் மொழியில் எவ்வாறு வழியமைப்புச் [Program] செய்வது? அதை எங்ஙனம் செயற்படுத்துவது? ஆகியவிவரங்கள் பாகம் 8, 9-இல் கூறப்பட்டு உள்ளன.

கணிதத்தில் சில குறிப்பிட்ட கணக்குகளுக்குத் தீர்வு காண்பதற்கெனப் பேஸிக் மொழியில் எழுதப்பட்ட வழியமைப்புகளும் சம்பளப்பட்டியல் தயாரிப்பதற்கென உருவாக்கப்பட்ட மாதிரி வழியமைப்பும் பின்னிணைப்பு—1-இல் தரப்பட்டுள்ளன. இவ் வழியமைப்புகள் அனைத்தும் AL 2016 என்ற நான்காம் காலக கட்டக் கணிப்பொறியில் செயற்படுத்திச் சோதித்தறியப் பட்டவைகளாகும். பின்னிணைப்பு—2-இல் கலைச் சொற்பட்டியல் தொகுத்துக் கொடுக்கப்பட்டுள்ளது.

நன்றியுரை

அறிவியல் தமிழ் வளரவேண்டும் என்ற எனது ஆர்வப் பசிக்கு அமைந்த உணவே இச்சிறிய நூல் எழுதும் பணியாகும். இந்நூலினை உருவாக்கும் பணியில் ஈடுபட்டிருந்த எனக்கு உற்ற உதவிகள் அனைத்தும் செய்ய உறுதுணையாய் நின்றவர் தமிழ்ப் பல்கலைக் கழகத் துணைவேந்தர் **மாண்புமிகு முதுமுனைவர் வ. அய். சுப்பிரமணியம்** அவர்களாவர். நற்கருத்துக்களை வழங்கும் சான்றோராய், உற்ற நல் ஆசானாய், உயர் அலுவலராய், நல் நண்பராய் விளங்கும் துணை வேந்தரவர்கள் இந்நூல் உருவாகப் பல்வகையாலும் உதவியதுடன் இந்தத் திட்டத்தைத் தொடங்கி நிறைவேற்ற பெருந்தூண்டுகோலாய் அமைந்தார்கள் என்றும் கூறவேண்டும். அறிவியல் தமிழ் வளர ஆக்கப்பணிகள் பலவற்றைத் திட்டமிட்டு நிறைவேற்றிவரும் அவர்களுக்கு எனது மனமார்ந்த நன்றியைத் தெரிவித்துக் கொள்கிறேன்.

இந்நூலில் பயன்படுத்தப்பட்டுள்ள கலைச்சொற்கள் பல வற்றை உருவாக்க தமது நிறைந்த பணிகளுக்கிடையிலும் நேரத்தைச் செலவிட்டு, பயன்மிக்க கருத்துக்களைக் கூறியும் இதில் பயன்படுத்தப்பட்டுள்ள சிலபடைப்படங்களை வழங்கியும் உதவிய

அண்ணா பல்கலைக் கழகத் துணைவேந்தர் **மாண்புமிகு முதுமுனைவர் வா. செ. குழந்தைசாமி** அவர்களுக்கு எனது நன்றியைத் தெரிவித்துக் கொள்கிறேன்.

இந்நூலினை எழுதும் பணியின் தொடக்கத்திலும் அதன் பிறகு இப்பணியின் பல்வேறு நிலைகளிலும் தக்க கருத்துக் களையும் விளக்கங்களையும் தந்து உறுதுணையாய் விளங்கிய எமது பேராசிரியர். **முனைவர் பி. ஏ. வெங்கடாசலம்** அவர்களுக்கு எனது நன்றி உரித்தாகுக.

இந்நூலினை அச்சிட்டு வெளிக் கொணரவும் பதிப்பு சார்ந்த பணிகள் பலவற்றைச் செவ்வனே நிறைவேற்றவும் பெரிதும் உதவியாய் விளங்கிய பதிப்புத்துறைத் துணை இயக்குநர் **கவிஞர்கோ கோவை. இளஞ்சேரன்** அவர்களுக்கு எனது நன்றியைத் தெரிவித்துக் கொள்கிறேன்.

இந்நூல் எழுதும் பணியில் ஈடுபட்டிருந்த எனக்குப் பல்வகையாலும் உதவிபுரிந்த **திரு. ச. பாஸ்கரன்** மற்றும் கணிப்பு மையப் பணியாளர்களுக்கு எனது நன்றி உரித்தாகுக.

இந்நூல் வெளிவர உறுதுணை புரிந்த தமிழ்ப் பல்கலைக் கழக நிருவாகத்தாருக்கும் தமிழ்ச் சான்றோர்களுக்கும், அச்சக ஊழியர்களுக்கும், இதனைத் தட்டச்சு செய்தும், ஒப்புநோக்கித் திருத்தம் செய்தும் உதவிய பணியாளர் அனைவருக்கும் எனது மனமார்ந்த நன்றி உரித்தாகுக.

தமிழ் மொழியில் கணிப்பொறி பற்றியும் பேஸிக் வழி யமைப்பு மொழி பற்றியும் அடிப்படைச் செய்திகளும் பிற விளக்கமான செய்திகளும் எளிமையாகப் புரிந்துகொள்ளும் வகை யில் இந்நூல் அமைந்திருக்கிறது. இந்நூல் பற்றிய ஆக்கபூர்வ மான கருத்துக்களை வரவேற்கிறேன்.

தஞ்சாவூர்—1
செப்டம்பர்—1985

கா. செ. செல்லமுத்து

உள்ளடக்கம் [CONTENTS]

1.	முன்னுரை	3
2.	கணிப்பொறியின் வரலாறு	11
2.1	கணிப்பொறியின் தோற்றம்	
2.2	கணிப்பொறியின் வளர்ச்சி	19
2.3	கணிப்பொறியின் வடிவமைப்பில் முன்னேற்றம்	23
2.4	கணிப்பொறியின் காலக்கட்டப் பிரிவு [GENERATION FEATURE OF A COMPUTER]	25
2.4.1	முதலாம் காலக்கட்டக் கணிப்பொறி [FIRST GENERATION COMPUTER]	28
2.4.2	இரண்டாம் காலக்கட்டக் கணிப்பொறி [SECOND GENERATION COMPUTER]	31
2.4.3	மூன்றாம் காலக்கட்டக் கணிப்பொறி [THIRD GENERATION COMPUTER]	
2.4.4	நான்காம் காலக்கட்டக் கணிப்பொறி [FOURTH GENERATION COMPUTER]	33
2.5	எண்ணிலக்க மற்றும் ஒப்புமைக் கணிப்பொறி [DIGITAL & ANALOG COMPUTER]	35
2.5.1	ஒப்புமைக் கணிப்பொறி [ANALOG COMPUTER]	
2.5.2	எண்ணிலக்கக் கணிப்பொறி [DIGITAL COMPUTER]	
2.5.3	கலப்பினக் கணிப்பொறி [HYBRID COMPUTER]	36
2.6.	கணிப்பொறியின் இயக்கத்திறனில் ஓர் ஒப்பு நோக்கு	37
3.	கணிப்பொறியின் அமைப்பும் செயற்பாடும்	41
3.1	கணிப்பொறியின் அமைப்பும் அங்கங்களும்	42
3.1.1	கணிப்பொறியின் அங்கங்கள்	43
3.2	இடுவரலகம் [INPUT UNIT]	44
3.2.1	அட்டைத் துளைப்பொறி [CARD PUNCHING MACHINE]	48
3.2.2	அட்டைப் படிப்பொறி [CARD READER]	51
3.2.2.1	துடைப்பான் முறை அட்டைப் படிப்பொறி [BRUSH TYPE CARD READER]	52

3.2.3	துளைத்தாள் நாடா [PUNCHED PAPER TAPE]	52
3.2.4	காகித நாடாப் படிப்பொறி [PAPER TAPE READER]	54
3.2.5	காந்தமைக்குறியீடு முறை [MAGNETIC INK CHARACTER RECOGNITION TECHNIQUE]	55
3.3	விடுவரலகம் [OUTPUT UNIT]	56
3.3.1	அச்சுப்பொறி [PRINTER]	57
3.3.1.1	இறுக்கநிலை அச்சுப்பொறி [IMPACT PRINTER]	58
3.3.1.2	சங்கிலி அச்சுப்பொறி [CHAIN PRINTER]	
3.3.1.3	சக்கர அச்சுப்பொறி [WHEEL PRINTER]	60
3.3.1.4	சட்ட அச்சுப்பொறி [BAR PRINTER]	61
3.3.1.5	எழுத்தச்சுப்பொறி [DOT MATRIX OR WIRE MATRIX PRINTER]	62
3.3.1.6	உருளை அச்சுப்பொறி [DRUM PRINTER]	63
3.3.1.7	இறுக்கமிலா அச்சுப்பொறி [NON IMPACT PRINTER]	64
3.4	ஆளகம் [CONTROL UNIT]	65
3.5	கணக்ககம் [ARITHMETIC UNIT]	
3.6	கணிப்பொறியின் இயக்கமுறை [OPERATIONAL PROCEDURE]	
3.7	நினைவகம் [MEMORY]	66
4	கணிமம்	71
4.1	முன்னுரை	
4.2	செய்வழியமைப்பு முறைமை [PROGRAMMING METHODOLOGY]	74
4.3	கணிப்பொறி மொழிகள் [COMPUTER LANGUAGES]	77
4.4	மொழி மாற்றிகள் [TRANSLATORS AND COMPILERS]	78
4.5	உயர்நிலை மொழிகள் [HIGHER LEVEL LANGUAGES]	81
4.5.1	பேஸிக் மொழி [BASIC LANGUAGE]	82
4.5.2	கோபால் மொழி [COBOL LANGUAGE]	84
4.5.3	போர்ட்ரான் மொழி [FORTRAN LANGUAGE]	86

4.6	வழியமைப்பு முறை நிருணயம் [PROGRAM CONSTRUCTION]	87
4.6.1	செய்வழி வரையறுப்பு [ALGORITHM DEFINITION	88
4.6.2	வழியமைப்புக் கட்டப்படம் [PROGRAMMING FLOW CHART]	89
4.6.3	முறையமை வரையறுப்புகள் [LOGICAL DEFINITIONS]	90
4.6.4	பதிவு செய்தலும் மொழிஆய்வு செய்தலும்	93
4.6.5	சரிபார்த்தலும் மொழிமாற்றம் செய்தலும் [VERIFICATION & TRANSLATION / COMPILATION]	
4.6.6	சோதித்தலும் தவறுகள் திருத்தலும்-	
4.6.7	செயற்பாடு [EXECUTION]	94
5	கருவியம் [HARDWARE]	97
5.1	முன்னுரை	
5.2	கருவியத்தின் அங்கங்கள்	
5.2.1	செயலகம் [PROCESSOR]	98
5.2.1.1	உம்மிணைக் கதவு [AND GATE]	99
5.2.1.2	அல்லதிணைக் கதவு [OR GATE]	100
5.2.1.3	எதிர்மறைக் கதவு [NOT GATE]	
5.2.2	இயக்கு நிலையகம் [CONSOLE]	101
5.2.3	நினைவகம் [MEMORY]	102
5.2.3.1	மிகைவேக நினைவகம் [FAST ACCESS MEMORY]	103
5.2.3.2	முதன்மை நினைவகம் [PRIMARY MEMORY]	105
5.2.3.2.1	காந்த இரும்பு வளைய நினைவகம் [MAGNETIC CORE MEMORY]	106
5.2.3.2.2	காந்த உருளை நினைவகம் [MAGNETIC DRUM MEMORY]	
5.2.3.2.3	ஒருங்கமை மின்சுற்று நினைவகம் [INTEGRATED CIRCUIT MEMORY]	107
5.2.3.2.4	மின்காந்தக் குமிழி நினைவகம் [MAGNETIC BUBBLE MEMORY]	108

5.2.4	ஆளகம் [CONTROL UNIT	109
5.2.5	இடுவரல், விடுவரல் ஒருங்குகள் [INPUT–OUTPUT SYSTEMS]	
5.2.5.1	மின்காந்த நாடாப் பொறி [MAGNETIC TAPE UNIT]	
5.2.5.2	மின்காந்த நாடாவில் செய்தியமைப்பு முறை [DATA ORGANIZATION IN MAGNETIC TAPE]	111
5.2.5.3	மின்காந்த நாடாவின் அமைப்பு [STRUCTURE OF A MAGNETIC TAPE]	112
5.2.5.4	மின்காந்த நாடாவில் செய்தி அடர்வு முறை [DATA RECORDING DENSITY IN MAGNETIC TAPE]	115
5.2.5.5	மின்காந்த நாடாவில் செய்திகள் பதிவு செய்யப்படும் முறை [DATA RECORDING METHOD IN A MAGNETIC TAPE]	116
5.2.5.6	மின்காந்தத் தட்டு [MAGNETIC DISC]	117
5.2.5.7	மின்காந்தத் தட்டின் அமைப்பு	
5.2.5.8	மாற்று மின்காந்தத் தட்டடுக்கு [CHANGEABLE MAGNETIC DISC PACK]	118
5.2.5.9	மின்காந்தத் தட்டின் தனி இயல்புகள்	120
5.2.5.10	மின்காந்த மென் வளைத்தட்டு [FLOPPY DISC]	
5.2.5.11	காட்சித் திரை [DISPLAY SCREEN]	122
5.2.5.11.1	காட்சித் திரையின் அமைப்பு	
5.2.5.11.2	காட்சித் திரையின் சிறப்பியல்புப் பண்புகள்	123
6.	கணிப்பொறியின் பயன்கள்	127
6.1	முன்னுரை	
6.2	கல்விப் பணியில் கணிப்பொறி	128
6.3	மருத்துவத் துறையில் கணிப்பொறி	131
6.4	வேளாண்மைத் துறையில் கணிப்பொறி	133
6.5	தொழில் துறையில் கணிப்பொறி	134
6.6	தொலை தொடர்புத் துறையில் கணிப்பொறி	138
6.7	போக்குவரத்துத் துறையில் கணிப்பொறி	139
6.8	பொறியியல் துறையில் கணிப்பொறி	140
6.9	விண்வெளிப் பயணத்தில் கணிப்பொறியின் பங்கு	141

6.10	பாதுகாப்புப் படையில் கணிப்பொறி	143
6.11	காவல்துறைச் செயல்பாட்டில் கணிப்பொறியின் பங்கு	144
6.11.1	கணிப்பொறியில் கைரேகைப் பகுப்பாய்வு [THUMB IMPRESSION ANALYSIS]	145
6.12	வங்கிகள் மற்றும் ஆயுட்பாதுகாப்பு நிறுவனங்களில் கணிப்பொறி	146
7.	பேஸிக் மொழி [BASIC LANGUAGE]	151
7.1	முன்னுரை	
7.2	பேஸிக் வழியமைப்பு மொழி [BASIC PROGRAMMING LANGUAGE]	154
7.3	பேஸிக் மொழியின் தனிச் சிறப்பியல்புகள் [SPECIAL FEATURES OF BASIC LANGUAGE]	156
7.4	பேஸிக் மொழி மாற்றிகள் [BASIC LANGUAGE INTERPRETER / COMPILER]	158
7.5	மாறிலிகள் [CONSTANTS]	161
7.6	மாறிகள் [VARIABLES]	163
7.6.1	மாறிகளின் பெயர் [VARIABLE NAME]	
7.6.2	தனி மாறிகள் [SIMPLE VARIABLES]	164
7.6.3	நினைவக நிருணய மாறிகள் [SUBSCRIPTED VARIABLES]	165
7.7	குறிகள் [CHARACTERS]	
7.8	இயக்கிகள் [OPERATORS]	166
7.9	கணக்கியல் இயக்கிகள் [ARITHMETIC OPERATORS]	
7.10	முறையமை இயக்கிகள் [LOGICAL & BOOLEAN OPERATORS]	167
7.11	தொடர்பமை இயக்கிகள் [RELATIONAL OPERATORS]	
7.12	கூறியக்கிகள் [FUNCTIONAL OPERATORS]	168
7.13	சூத்திரங்களும் சமன்பாடுகளும் [FORMULA AND EQUATIONS]	

7.14	நினைவக வரிசை [ARRAYS]	170
7.15	பேஸிக் மொழிக் கட்டளைகள் [BASIC LANGUAGE COMMANDS]	172
8.	பொது வழியமைப்பு ஆணைகள் [GENERAL PROGRAMMING STATEMENTS]	185
8.1	தன்மை ஆணைகள் [TYPE STATEMENTS]	186
8.2	கட்டுப்பாட்டு ஆணைகள் [CONTROL STATEMENTS]	188
8.3	செய்வளை ஆணைகள் [LOOPS]	192
8.4	பிறவகை ஆணைகள் [OTHER STATEMENTS]	195
9.	துணைநினைவகக் கோவை ஆணைகளும் கூறுகளும் [DISK BASED INPUT/OUTPUT STATEMENTS AND FUNCTIONS]	203
9.1	இடுவரல்/விடுவரல் ஆணைகள் [INPUT/OUTPUT STATEMENTS]	204
9.1.1	இடுவரல் ஆணைகள் [INPUT STATEMENTS]	
9.1.2	விடுவரல் ஆணைகள் [OUTPUT STATEMENTS]	207
9.1.3	எண்படிவ ஆணைகள் [NUMERIC FORMAT STATEMENTS]	210
9.1.4	குறிக்கோவைப் படிவ ஆணைகள் [STRING FORMAT STATEMENTS]	211
9.2	கோவை ஆணைகள் [DISC BASED FILE STATEMENTS]	212
9.2.1	BBASIC	213
9.2.2	ABASIC	219
9.3	வழியமைப்புக் கூறுகள் [PROGRAMMING FUNCTIONS]	223
9.3.1	கணக்கியற் கூறுகள் [ARITHMETIC FUNCTIONS]	
9.3.2	குறிக்கோவைக் கூறுகள் [STRING FUNCTIONS]	227
9.3.3	தனிக் கூறுகள் [SPECIAL FUNCTIONS]	234
9.4	கட்டுப்பாட்டுக் கட்டளைகள் [CONTROL COMMANDS]	236

9.5	திருத்துமுறைக் கட்டளைகள் [EDIT COMMANDS]	237
9.6	ABASIC மற்றும் BBASIC இல் வழியமைப்புச் செயற்படும் முறை	239
9.6.1	ABASIC மொழியில் செயற்பாடு	
9.6.2	BBASIC மொழியில் செயற்பாடு	240

பின்னிணைப்பு

பின்னிணைப்பு—1 [APPENDIX—A] மாதிரி வழியமைப்புகள் [SAMPLE PROGRAMS]	243
பின்னிணைப்பு—2 [APPENDIX—B] கலைச்சொற்பட்டியல் [LIST OF TECHNICAL TERMS]	271
பின்னிணைப்பு—3 [APPENDIX—C] நூற்பட்டியல் [BIBLIOGRAPHY]	285

பாகம் – 1

1. முன்னுரை

பண்டைய மனிதர்கள் காடுகளில் விலங்குகளை வேட்டையாடியும் குகைகளில் உறைந்தும் வாழ்ந்து வந்தனர். அக்காலத்தில் எண்ணங்களையும் செயல்களையும் உணர்ச்சிகளையும் வெளிப்படுத்தும் ஒரு தலையாய கருவி, மொழி என்பதை அவர்கள் உணர்ந்திலர். தங்களது எண்ணங்களையும் செய்கைகளையும் வெளிப்படுத்த மொழிவழி வெளிப்பாடு இன்மையால் ஊமையர் போன்று கை மூலம் செய்கை ஒன்றையே வெளிப்பாட்டுக் கருவியாகப் பயன்படுத்தினர். எனினும் இச்செய்கைகளின் மூலம் நாகரிகம் முன்னேறாத நிலையில் வாழ்ந்த ஆதி மனிதர்கள்கூட கணிதத்தின் தத்துவத்தை நன்கு உணர்ந்திருந்தனர் என்பதற்குப் பல சான்றுகள் காணக் கிடைக்கின்றன. அதன் பின்னர் மொழி தோன்றியது. மொழியறிவு வளர்ச்சியின் காரணமாக மனிதகுல வாழ்வின் முன்னேற்றத்தில் புதியதோர் எழுச்சி உண்டாயிற்று என்பது உண்மையாகும். அறிவியல் மேம்பாடு அடையவும், அதன் பலனாய் வாழ்வின் வசதிகள் பல பெருகுவதற்கும், வாழ்க்கை நாகரிகமடைவதற்கும் மொழி அறிவே ஒரு பயனுறு சாதனமாயமைந்தது என்பதில் ஐயமில்லை.

இன்று மனிதன் அறிவியற் சிகரத்தின் உச்சியை அடைந்துள்ளான் என்றுதான் கூறவேண்டும். மிகைவேக வானூர்தி படைத்த மனிதன், சந்திர மண்டலத்தியல் கண்டு தெளியவும் தலைப்பட்டான். வானியல் ஆராய்ச்சியில் பல புதிய கண்டுபிடிப்புகள் நாள்தோறும் நிகழ்ந்து வருவது நாம் கண்

கூடாகக் காண்கின்ற நிகழ்ச்சியாகும். அறிவியல் மேம்பாட்டால் காலமும் இடமும் மிகவும் சுருங்கிவிட்டன. பணியை விரைவில் செய்து முடிக்கவும் குறிப்பிட்ட நேரத்தில் ஓர் இடம் விட்டு மற்றோர் இடம் சென்று சேரவும், பல போக்குவரத்து கருவிகள் தேவைப்பட்டன. இதன் பொருட்டு இரு சக்கர வண்டி, புகைவண்டி, மிகை வேகப் பேருந்து, வானூர்தி போன்ற ஊர்திகள் உருவாக்கப்பட்டன. எனினும் காலந்தோறும் மாறிவரும் வாழ்வின் தேவைகளுக்கேற்ப ஒவ்வொரு பயன்பாட்டிற்கும் ஏற்ற வகையில் மேலும் பல புதிய கருவிகள் உருவாக்கப்பட வேண்டியிருந்தன.

மனிதன் தனது எண்ணங்களை வெளிப்படுத்த, மொழி எங்ஙனம் சிறந்ததொரு வெளிப்பாட்டுக் கருவியாக [Media] அமைகிறதோ அதேபோன்று ஒரு செய்தியை அல்லது ஒரு சமன்பாட்டை அல்லது ஒரு கணக்கை நிரந்தரமாகப் பதிவு செய்து வைத்துக் கொள்வதற்குச் சில வெளிப்பாட்டுக் கருவிகள் [Media] தேவைப்பட்டன. இதன் அடிப்படையிலேயே ஆவணங்கள், நூல்கள் மூலம் செய்திகளும், விவரங்களும் நிரந்தர வடிவம் பெறலாயின குறிப்பாகக் கூறின் பண்டைய மனிதர்கள் கற்களையும், காய்களையும், கை, கால் விரல்களையும் கணக்கிடும் கருவிகளாகப் பயன்படுத்தினர் என்பதற்குச் சான்றுகள் பல உள்ளன.

மனிதன் தான் அறிந்த செய்திகளையும் கணக்குகளையும் மனத்தில் நிலைநிறுத்தவும் பின்னர் அவற்றை வெளிக்கொணர்ந்து பயன்படுத்தவும் மிகவும் சிரமமாய் இருந்தது. எடுத்துக்காட்டாக நாம் ஒரு கணக்கைச் செய்யும்பொழுது, எண்களைப்பெருக்கவோ, கூட்டவோ வேண்டியுள்ளது. அவ்வாறு செய்யும்பொழுது அதன் தொடர்ந்த செயல் குறித்து, சில எண்களைத் தற்காலிகமாக மனத்தில் நிலை நிறுத்த வேண்டியுள்ளது. ஆனால் பணிகள் அதிகமாகப் பெருகும்பொழுதும் கணக்கின் செயற்பாட்டுத் தன்மை அதிகமாகும் பொழுதும் செய்திகள் அனைத்தையும் நினைவிற் கொண்டு, பின்னர் எடுத்தாள்வது மிகவும் கடினமான பணியாகும். இக்காலத்தில்தான் செய்திகளை எழுதிச் சேகரிக்கும் அரிய கருவியான காகிதம் உருவாக்கப்பட்டது. எனினும் பல்லாயிரக் கணக்காண செய்திக் குறிப்புகளைத் தேவைக்கேற்ப, வேண்டிய பொழுது வேண்டியவாறு தெரிவு செய்து பயன் படுத்துவது கடினமான பணியாயிருந்தது. இதற்குக் காரணம்

மலையளவு பெருகியுள்ள செய்திகளில் வேண்டியவற்றைக் குறிப்பிட்ட நேரத்தில் தெரிவு செய்து பயன்படுத்துவதிலுள்ள மனித மூளையின் இயலாமை மற்றும் இச்செயலைச் செய்வதில் வேக மின்மையாகும்.

இக்காலக் கட்டத்தில்தான் வாழ்வின் நலம் பேண உருவாக்கப்பட்ட பற்பல எந்திரங்களில் சில கருவிகள் மட்டுமே மனிதனின் அறிவு வளர்ச்சிக்கு ஏற்ற,உற்ற உரமாக நின்று ஊக்கம் புரிந்தன. உலகியற் செய்திகளையும் பற்பல சமூக, இலக்கிய, அறிவியல் குறிப்புகளையும் ஏதாவது ஒரு வடிவில் ஆவணம் செய்து பாதுகாத்தல் அவசியமான பணியாகும். கல்வி உலகில் கடமை பல புரிவதற்கும், பிற்கால சந்ததியினரின் வாழ்க்கை வளம் பெறுவதற்கும் இது ஓர் இன்றியமையாப் பணியாகும். இவ்வாறு செய்திகளும், விபரங்களும் ஆவணப்படுத்தப்பட்டால் மட்டும் போதாது. உரிய முறையில் குறிப்பிட்ட நேரத்தில் தேவைக்கேற்ப, செய்திகளைத் தெரிவு செய்து தருதல் அதனினும் அவசியமிக்கதொரு பணியாகும். விண்வெளிப் பயணம், அணு ஆய்வுத்துறை, பொது அறிவியல், ஆராய்ச்சி முதலானவை பெருகி வளர்ந்துள்ள இக்காலத்தில் செய்திகளின் பெருக்கம், மாமலையையும் மிஞ்சும். இந்நிலையில் ஒரு வினாடியில் பத்தாயிரத்தில் ஒரு பங்கு (1/10,000) நேரத்தில் பல்லாயிரம் பணிகள் பாங்குடன் செய்து முடிக்கப் பெறுவதைப் பார்க்கிறோம். மனிதனின் செயல் வேகம் இதற்கு ஈடு கொடுக்க இயலாது என்பது உறுதி. எனவே இதுபோன்ற பணிகளில் கணிப்பொறி இன்றியமையாச் சாதனமாய் விளங்கும் என்பது உறுதி.

மனிதன் ஒரு குறிப்பிட்ட பணியைத் திரும்பத் திரும்பச் செய்யும் பொழுது சோர்வடைகிறான். இதனால் தவறுகள் நேரவும்,குறிப்பிட்ட பணி குறிப்பிட்ட காலத்தில் செய்துமுடிக்கப் பெறாமற் போகவும் வாய்ப்புண்டு. மேலும் பல பணிகளைக் கணிப்பொறி செய்யும் வேகத்தில், மனிதனால் செய்து முடிக்க இயலாது. பல்லாயிரக்கணக்கான செய்தித் தொகுப்புகளை ஒரே சமயத்தில் மனத்தில் நிறுத்தி, கணித்து விடை காணல் கனவிலும் கருதமுடியா ஒன்றாகும். எனவே செயற்பாட்டில் வேகமும், துல்லியமும் மிகுந்த ஒரு கருவியான கணிப்பொறிக்கு ஒரு செயலைச் செய்யும் வழிமுறைகளை முறைப்படி பயிற்றுவிப்பதன் மூலம் பன்மடங்கு திறன்பட பணி செய்ய முடியும் என்ற உண்மை கண்டறியப்பட்டது.அறிவியலின் ஆழ்ந்த நுணுக்கத்தின் துணைகொண்டு மனிதன் கணிப்பொறியை உருவாக்கினான்.

மனிதனைவிடப் பல்லாயிரம் மடங்கு வேலைப்பளுவை ஏற்றுச் செயற்படும் ஆற்றல் வாய்க்கப் பெற்றிருந்த போதிலும், கணிப்பொறி ஒரு மனிதனின் அடிமையேயாகும். ஏனெனில் அது அவனால் உருவாக்கப்பட்ட ஒரு கருவி. மேலும் மனிதனால் உரியமுறையில் கட்டளையிடப்படும் பொழுது மட்டுமே பணி புரியக் கூடியதும், தானாகச் சிந்திக்கத் திறனற்றதுமான ஓர் அடிமையாகும். இவ்வகையில் மனிதனுக்கும், கணிப்பொறிக்கும் பல வேறுபாடுகள் உள்ளன. எனினும் சக்திமிக்க, பாதுகாப்புக் கருவிகளைப் படைக்கவும் வான்வெளியில் வியக்கத்தக்க பல விந்தைமிகு சாதனைகளைப் புரியவும் கணிப்பொறியின்றேல் முடியாமற் போயிருக்கும் என்பது முற்றிலும் உண்மை.

நமது வாழ்வில் அன்றாடம் நிகழும் பல நிகழ்ச்சிகளின் செயற்பாடுகள் அனைத்தும் ஏதோ ஒருவகையில் கணக்கியற் கோட்பாடுகளை அடிப்படையாகக் கொண்டே செய்து முடிக்கப் பெறுகின்றன. எனவேதான் நமது மூதாதையர்கள் கூட, கணித அறிவு, வாழ்க்கையின் அன்றாடச் செயல்களுக்கு உற்ற துணை என்பதை நன்கு உணர்ந்திருந்தனர். இதற்குப் பழம் இலக்கியங்கள் சான்று பல பகர்கின்றன. இதன் முக்கியத்துவத்தை உணர்ந்தே கணக்கியலின் கோட்பாடுகளை அடிப்படையாகக் கொண்ட பல கணக்கீட்டுக் கருவிகள் அக்காலத்தில் உருவாக்கப்பட்டன. அவை அவ்வக் காலத்தில் நிலவிய அறிவியல் மேம்பாட்டுக்கேற்ப, தொழில்நுட்பம் பெற்றிலங்கின. பற்சக்கரங்களையும், சட்டங்களையும் கொண்டு சில கருவிகள் உருவாக்கப்பட்டன. காய்கள், சங்கிலிகள், தட்டுப் பலகைகள் முதலியவை இணைந்த நிலையில் சில அடிப்படைக் கணிப்பியல் கருவிகள் கண்டறியப்பட்டன. இவையாவும் மனிதனின் துணையால் கூட்டல், கழித்தல் ஆகிய எளிய கணக்குகளை மட்டும் செய்யும் தன்மையுடன் இருந்தன. ஆனால் தற்காலத்தில் உருவாக்கப்படும் திறன்மிகு கணிப்பொறிகள், இலட்சக்கணக்கான ஆணைகளை ஒரு நொடியில் ஏற்றுச் செயற்படும் ஆற்றல் வாய்ந்தவைகளாகும். எனவேதான் இன்றைய அறிவியல் யுகத்திலே, கணிப்பொறி மனித வாழ்வின் ஓர் அங்கமாகப் பின்னிப் பிணைந்துள்ளது என்று கூறலாம்.

கணிப்பொறி நமது வாழ்க்கையின் ஒவ்வொரு நிலையிலும் ஒழுங்கு நிறைந்த ஒரு செயற்பாட்டு மனப்பாங்கை உருவாக்கும் முக்கிய காரணியாக விளங்குகிறது. நாம் இன்று பெற்றுள்ள வாழ்வின் வசதிகள் யாவும் அறிவியலினால் பெற்ற ஆக்கங்கள்

ஏனில் அஃது மிகையாகாது. அறிவியலின் எண்ணற்ற கண்டு பிடிப்புகளில் குறிப்பிடத்தக்க கருவி கணிப்பொறியாகும் என்பதில் சிறிதும் ஐயமில்லை. முன்பே கூறியபடி கணிப்பொறி ஒரு மனிதனின் அடிமையென்றால் அதைக்கொண்டு வியக்கத்தகு முறையில் பணிகள் பலவற்றைச் செய்யும் வழி யாது? உடலியங்க உயிர் தேவையாதல் போல கணிப்பொறியை இயக்கவும், பயன்படுத்தவும் வழியமைப்புகளின் தொகுப்பான கணிமம் [Software] அவசியமாகிறது. உயிரற்ற கருவியாகிய கணிப் பொறிக்கு ஏற்ற முறையில் தக்காங்கு கட்டளையிடின் அது நமது பணிகள் அனைத்தையும் சிறப்பாகச் செய்யும் வலிமை பெற்றது. தொழிற்புரட்சியின் பயனாய் மனிதனின் உடல் உழைப்பை ஏற்றுப் பணிகளைச் செய்யப் பல கருவிகள் உருவாக்கப்பட்டன. அது போன்றே உள உழைப்பின் நுட்பத்தைத் தானேற்றுத் திறம்படச் செயற்படும் தனிப்பாங்கு பெற்ற கருவியாகக் கணிப்பொறி உருவாக்கப்பட்டது. எனவே கணிப்பொறியின் தோற்றம்,வடிவமைப்பில் விரிவாக்கம். மற்றும் அதன் பயன்பாடு போன்றவற்றின் பின்னணியில் பல கருத்துக்களை நாம் அறிதல் மிகவும் அவசியமாகும். கல்வி, தொழில், மருத்துவம், போக்குவரத்து முதலான எண்ணற்ற துறைகளில், இணையற்ற பணிசெய்யும் கணிப்பொறியின் பயன்கள் பலப்பல. அவற்றின் தோற்றம், வரலாறு, அமைப்பு, பயன்கள் ஆகியவை பற்றிக் கீழ்க்கண்ட பாகங்களில் விரிவாகக் காணலாம்.

பாகம் – 2

2. கணிப்பொறியின் வரலாறு

2.1. கணிப்பொறியின் தோற்றம்

கணிப்பொறியின் தோற்றம் பல ஆயிரம் ஆண்டுகளுக்கு முற்பட்டதாகும். இது இன்றோ, நேற்றோ துவங்கப்பட்ட பணியின் பயனன்று. பழங்காலத்திலிருந்தே கணிப்பொறி ஏதோ ஒரு வடிவத்தில் கணக்கியற் கோட்பாடுகளுக்குப் பயன்பட்டு வந்துள்ளமை மனித வாழ்வில் கணிதத்தின் இன்றியமை யாமையை நன்கு புலப்படுத்தும். இத்துறையில் அவ்வக்கால கட்டங்களில் பல்வேறு அறிஞர்களால் செயற்படுத்தப்பட்டுள்ள பல பணிகள் இதற்குச் சான்று பகர்கின்றன. கணிப்பொறியின் வரலாறும், தோற்றமும்பற்றிக் காணும் முன்பு கணிப்பொறி என்றால் என்ன? என்ற வினாவுக்கு விடையளித்தல் நலமாகும். கணிப்பொறி என்பது ''தானாகவே சிந்திக்கத் திறனற்றது. ஆனால் மனிதர்களால் உரிய முறையில் கட்டளையிடப்படும் பொழுது மிகப்பெரிய பணியைக்கூட, சில நிமிடங்களில் துல்லிய மாகச் செய்துமுடிக்கும் திறன் வாய்க்கப்பெற்றது. மேலும் ஒரே பணியைப் பல்லாயிரக்கணக்கான தடவைகள் அலுப்பின்றி, திரும்பத் திரும்பச் செய்யக்கூடிய ஆற்றல் வாய்ந்த ஓர் அடிமை'' என்று வரையறுத்துக் கூறலாம்.

வேகமாகச் செயற்படுதலிலும்ஏராளமான செய்திகளைத் தொகுத்து வகைப்படுத்தி விடை காண்பதிலும் மனிதனைக் காட்டிலும் பன்மடங்கு ஆற்றல் நிறைந்தது கணிப்பொறியாகும். ஆனால் மனிதன் ஒரு குறிப்பிட்ட பணியைச் செய்ய அதிக நேரத்தை எடுப்பதுடன் அதே சமயத்தில் ஒரே பணியை திரும்பத்

திரும்பச் செய்யும்பொழுது சோர்வும் அடைகிறான். எனவே செய்யும் பணியில் விருப்பமின்மையும் சோர்வும் பணியைப் பிழையின்றிச் செப்பமுறச் செய்யத் தடையாக நிற்கின்றன. எனவே காலத்தையும், செய்திக் குவியலையும் கடந்த கணிப் பொறி செய்யும் பணியில் சொந்த விருப்பமோ, வெறுப்போ கொள்வதில்லை. பணியைத் திட்பமுறவும், சிறப்பாகவும், விரைவிலும் செய்துமுடிக்கும் ஆற்றல் வாய்ந்த கருவி, கணிப் பொறி என்பதில் ஐயமில்லை.

நமது வாழ்க்கையில் கணிதம் பெரும் பங்கு வகிக்கிறது. ஆதிகால மனிடர்கள் குச்சிகளையும், கற்களையும் பயன் படுத்தித் தாங்கள் வளர்த்து வந்த ஆடு, மாடு போன்ற மிருகங்களைக் கணக்கிட்டதாகக் கூறுவர். காடுகளில் வேட்டை யாடிய விலங்குகளைக் குறிப்பிடுவதற்குக் கற்களையும் கைவிரல் களையும் பயன்படுத்தியதாக நாம் அறிவோம். சிறுகற்கள், காய்ந்த காய்கள் முதலியவற்றைக் கொண்டு எண்களைக் குறிப்பிடுதலும் சிறு கணக்குகளைச் செய்தலும் அக்காலத்தில் நடைமுறையில் இருந்தமை தெரியவருகிறது. நமது முன்னோர் கள் கணிதச் செயல்களைச் செய்ய வியக்கத்தக்க பல முறைகளை பயன்படுத்தியுள்ளமை அவ்வக் காலங்களில் உருவாக்கிப் பயன் படுத்தப்பட்ட பல கணக்கியற் கருவிகள் ஆராய்ச்சியாளர்களால் கண்டறியப்பட்டுள்ளமையிலிருந்து நன்கு புலனாகிறது. எடுத்துக் காட்டாக, பாபிலோனியர்கள் பெருக்குவதற்கும், கணக்கிடு வதற்கும் எண்களைக் கையாள்வதற்குக் களிமண்ணாலான பலகை ஒன்றை, கி. மு. 1700–இல் பயன்படுத்தியுள்ளனர். அவர்கள் வகுத்த நேரக் கணக்குப்படி மணி, நிமிடம், நொடி போன்றவற்றைக் கணக்கிட 60 என்ற எண்ணை அடிப்படையாக [Base of 60] கொண்டு செயற்பட்டுள்ளனர். இந்த அடிப்படை எண்ணையே நவீன அறிவியல் யுகமான இன்றும்கூட காலத்தை அளக்கப் பயன்படும் அலகாக [unit] நாம் பயன்படுத்தி வருகிறோம். மேலும் அவர்கள் சிறு கணக்குகளுக்கும், சமன்பாடு களுக்கும் தீர்வு காண்பதற்கான படிப்படியான நியதிகளைக் [Algorithmic approach] கண்டறிந்தனர். இதன் மூலமாக அவர்களால் கிரகணங்களின் காலங்களை மிகவும் துல்லியமாகக் கண்டறிய முடிந்தது.

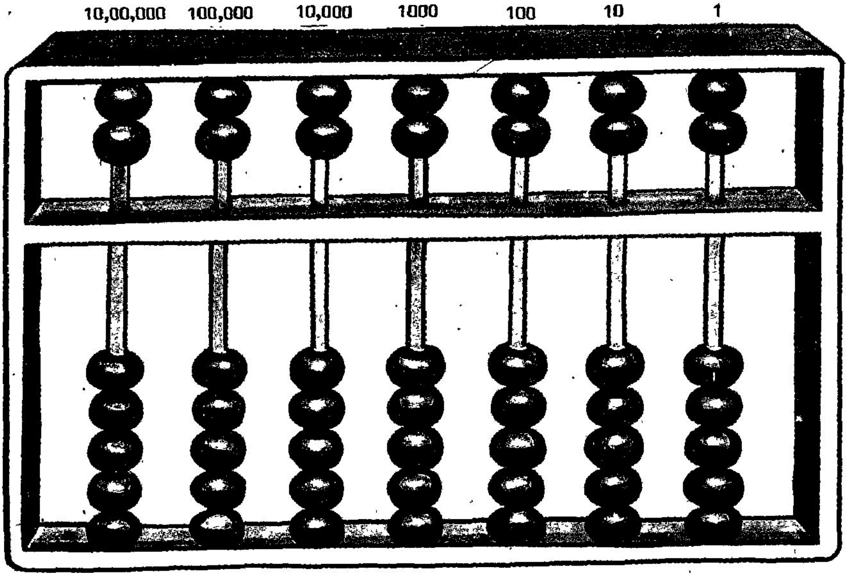
கணக்கிடுவதில் உள்ள கடினம் மற்றும் மனித மனத்தின் எல்லை போன்றவற்றை உணர்ந்த மனிதன் கணக்கிடப் பலவித

எண்ணற்ற கருவிகளை உருவாக்கினான். கணக்கிடுதலும், அதற்கேற்ற கருவிகளை உருவாக்குதலும் அவசியம் என்பதை நமது முன்னோர் நன்கு உணர்ந்திருந்தனர் என்பதற்கு மற்றொரு சான்றாக கி. மு. *1900 -1600* -இல் தென் இங்கிலாந்தில் கண்டு பிடிக்கப்பட்டுள்ள 'Stone henge' என்ற கற்கருவியைக் கூறலாம். இவ்வமைப்பு **கணிதவியலார்க்கு** இன்னும் ஒரு சவாலாக இருப்ப தாகக் கூறப்படுகிறது. இக்கருவி வானாராய்ச்சிக் கருவியாகவும், கணிப்பொறியாகவும் பயன்படுத்தப்பட்டதாக *1965* -இல் ஜெரால்டு ஹாவ்கின்ஸ் [Gerald S.Hawkins] என்ற அறிஞர் கண்டறிந்தார். இக்கருவி சேர்க்கை முறையில் [Combinations] பலவித அமைப்புகளைக் கொண்டு வியக்கத்தகு முறையில் கல்லி னால் உருவாக்கப்பட்டுள்ளது. கல்லினால் ஆன இவ்வமைப்பு வானவெளியில் சூரியன், சந்திரன் ஆகியவற்றின் நிலைகளுக் கேற்ப வடிவமைக்கப்பட்டிருந்தது. இதன் உதவியுடன் காலநிலை, பருவகாலங்கள், கிரகணம் முதலியவற்றைத் துல்லியமாகக் கண்டறிந்தனர். இதில் பொதிந்துள்ள தொழில் நுட்பம் [Sophistication] ஒரு நவீன கணிப்பொறியை உருவாக்கத் தேவை யான தொழில் நுட்பத்தை ஒத்திருந்தது. இதே போன்ற அமைப்பு அமெரிக்க நாட்டின் பல பகுதிகளிலும் காணப்படுவதிலிருந்து, நமது முன்னோர்களின் கணித அறிவின் பாங்கு நன்கு புலப்படும்.

கூட்டல், கழித்தல் முதலிய சிறிய கணக்கீடுகள் செய்வதற்கு கை, கால் விரல்களையும், காய்களையும் நமது முன்னோர்கள் பயன்படுத்தினர் இதனை அடிப்படையாகக் கொண்டுகி.மு.*2000* ஆண்டுகளுக்கு முன்பு பயன்படுத்தப் பட்ட மணிச் சட்டம் [ABACUS] என்ற கருவி உருவாக்கப்பட்டது இதுவே தற்காலக் கணிப்பொறி உருவாவதற்கு ஒரு முன்னோடியாக அமைந்தது எனக்கூறலாம். இவ்வமைப்பு மரச் சட்டம், கம்பி காய்ந்த காய்கள் அல்லது மணி முதலியவற்றைக் கொண்டு உருவாக்கப்பட்டது.

மணிகள் துளையிடப்பட்டு, கம்பிகளில் கோக்கப்பட்டு மரச்சட்டங்களில் பொருத்தப்பட்டிருந்தன. (படம் எண் *1*ஐப் பார்க்க). சட்டத்தில் பொருத்தப்பட்டுள்ள கம்பிகளின் எண்ணிக் கையைப் பொருத்து, பல வரிசைகள் ஒரே சட்டத்தில் அமைக்கப்பட்டன. கம்பிகளின் குறுக்கே ஒரு குறுக்குச்சட்டம் [Cross bar] பொருத்தப்பட்டிருந்தது. ஒவ்வொரு கம்பியிலும் குறுக்குச் சட்டத்திற்கு மேலே இரண்டு மணிகளும் கீழே ஐந்து

மணிகளுமாக மொத்தம் ஏழு மணிகள் கோக்கப்பட்டிருந்தன. வலமிருந்து இடமாகக் கம்பி வரிசைகளின் நிலை மதிப்பு ஒன்று, பத்து, நூறு, ஆயிரம் என்ற வகையில் அமைக்கப்பட்டிருந்தது. அதாவது வலக்கோடியிலுள்ள மணியின் நிலை மதிப்பு ஒன்றாக வும், அடுத்த வரிசை மணி பத்து நிலை மதிப்பையும், அதற்கடுத்த வரிசையிலுள்ள மணிகள் முந்திய வரிசை மணிகளின் நிலை மதிப்பைப் போல் 10 மடங்கு நிலை மதிப்பையும் பெற்றிருந்தன. மேலும் கணக்கிடுவதற்கேற்பக் குறுக்குச் சட்டத்திற்கு மேலுள்ள மணிகளுக்கு 5 என்றும் அதற்குக் கீழே உள்ள மணிகளுக்கு 1 எனவும் எண் மதிப்பு கொடுக்கப்பட்டிருந்தது. மணிச் சட்டத்தின் அமைப்பு, படம் எண் 1, இல் விளக்கிக் காட்டப்பட்டுள்ளது.

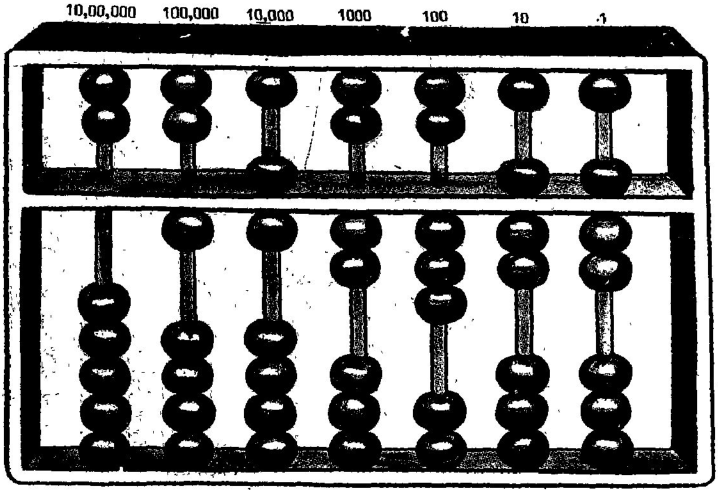


படம் 1

மணிச்சட்டம் (ABACUS)

மணிச்சட்டத்திலுள்ள மணிகளைக் குறிப்பிட்ட விதத்தில் நகர்த்தி எண்களைக் குறிப்பிடலாம். படம் எண் 1-இல் காட்டப் பட்டுள்ள மணிச்சட்டத்தைக் கொண்டு 99,99,999 வரை உள்ள எண்களைக் குறிப்பிட முடியும். மேலே விவரிக்கப்பட்டுள்ள

நிலைமதிப்பையும் (1,10,100,1000 முதலியன), மணிகளுக்குக் கொடுக்கப்பட்ட எண் மதிப்பையும் கொண்டு, எண்களைக் குறிப்பிடுகிறோம். **குறுக்குச் சட்டத்திற்கு அருகில் உள்ள மணிகள் மட்டுமே எண்களைக் குறிக்கின்றன. படம் எண் 2 -இல் 1,62,377 என்ற எண் குறிக்கப்பட்டுள்ளது.**



படம் 2

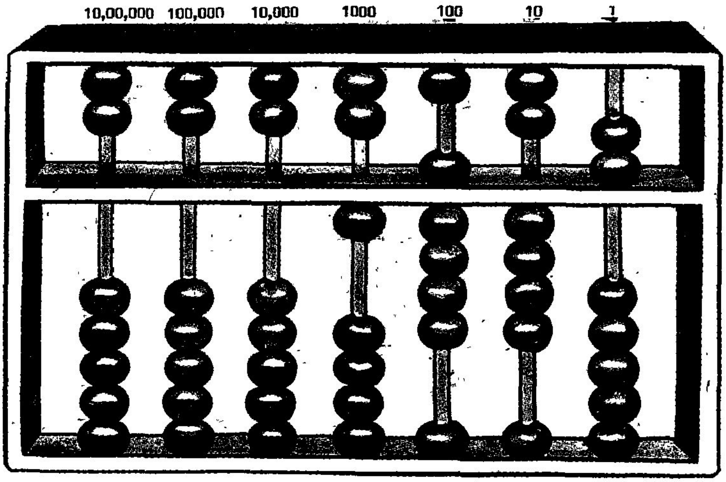
மணிச்சட்டம்

(1, 62, 377 என்ற எண் குறிக்கப்பட்டுள்ளது)

மணிச்சட்டத்தைக் கொண்டு இரண்டு எண்களை எங்ஙனம் கூட்டுவது என்பது பற்றிச் சிறிது பார்ப்போம். முதல் எண்ணை மேற்குறிப்பிட்ட முறையில் உரிய மணிகளை நகர்த்திக் குறிப்பிட வேண்டும். பின்னர் இந்நிலையை மாற்றாமல் இரண்டாம் எண்ணையும் குறிப்பிடவேண்டும். பின்னர் 10 அல்லது அதற்கு மேல் மதிப்புள்ள கம்பிகளில் குறுக்குச் சட்டத்தின் மேலுள்ள மணிகள் நகர்த்தப்பட்டு அதற்கு இடப்பக்கம் உள்ள கம்பியின் மதிப்புடன் 1 -ஐக் கூட்ட வேண்டும். இந்த வகையில் மணிச் சட்டத்தைக் கொண்டு எண்களைக் கூட்டுதலும் கழித்தலும் செய்ய முடிகிறது. மணிச்சட்டத்தைக் கொண்டு 1,235 மற்றும்

715 என்ற எண்களைக் கூட்டும் முறை, படம் எண் 3 -இல் விளக்கிக் காட்டப்பட்டுள்ளது.

மேற்குறிப்பிட்ட மணிச்சட்டத்தைக் கொண்டு கூட்டல், கழித்தல் முதலிய பணிகளை மட்டுமே செய்ய முடிந்தது. ஆனால் இதைக் கொண்டு பெருக்கல், வகுத்தல்களைச் செய்ய இயல வில்லை.



படம்—3

மணிச்சட்டம் மூலம் கூட்டல் கணக்கு (1235+715=1950)

1617-ஆம் ஆண்டு ஜான் நேப்பியர் [John Napier] என்ற ஸ்காட்லாந்தைச் சேர்ந்த கணக்கியல் அறிஞர் ஒரு கருவியை உருவாக்கினார். இதற்கு நேப்பியர் கட்டங்கள் [Napier Rods] என்று பெயர். கணிதத்தில் பல கணக்குகளைச் செய்யப் பயன் படும் சுருக்கெண் குறியீட்டு முறைக்குத் [Logarithm] தந்தையாக அமைந்தவரும் இவரே ஆவர். நேப்பியர் கட்டங்களைக்கொண்டு பெருக்கல், வகுத்தல், வர்க்கமூலம் [Square Root] காணல் ஆகிய பணிகளைச் செய்ய இயலும்.

நேப்பியர் கட்டங்கள், மரம் அல்லது எலும்புச் சட்டங் களைக் கொண்டு உருவாக்கப்பட்ட அமைப்பை உடையதாகும். இதில் ஒவ்வொரு எண் நிலைக்கு [Digit] ஒவ்வொன்றாக 9 சட்டங்கள் உள்ளன. ஒவ்வொரு சட்டமும் ஒன்பது சம சதுரங்

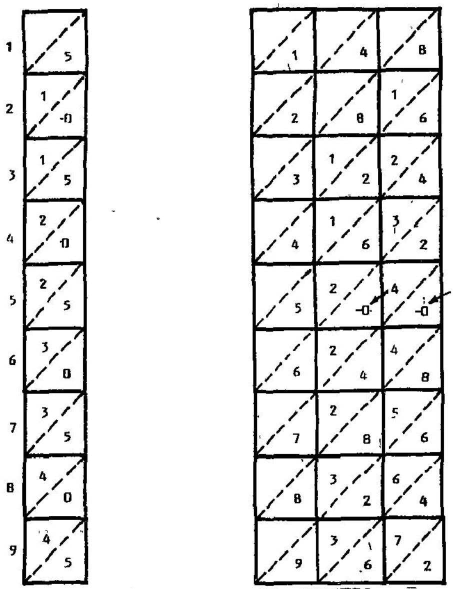
களாகப் பிரிக்கப்பட்டுள்ளது. இந்தச் சட்டங்கள் ஒவ்வொன்றை யும் ஒரு பெருக்கல் அட்டவணைக்கு ஒப்பிடலாம். இந்த ஒன்பது நெடுங் கட்டங்களில் எண்கள் கீழ்க்கண்டவாறு குறிக்கப் பட்டிருக்கும். மேல் வரிசையில் 1-இலிருந்து 9 வரையும் மற்றும் 0-வும் [Rows] பக்கவாட்டில் 1-இலிருந்து 9 வரையும் (முதல் குத்து வரிசை) எண்கள் குறிக்கப்பட்டிருக்கும். நீள் வரிசையும் குத்து வரிசையும் [Column] சந்திக்கும் இடத்தில் மேற்குறிப்பிட்ட எண்களைப் பெருக்கி வந்த எண்களும் குறிக்கப்பட்டிருக்கும். நேப்பியர் கட்டங்களைக் கொண்டு கூட்டல், கழித்தல் பணிகளை மணிச்சட்டத்தைக் காட்டிலும் வேகமாகச் செய்ய முடிந்தது. இதன் அமைப்பு, படம் எண் 4-இல் காட்டப்பட்டுள்ளது.

1	2	3	4	5	6	7	8	9	0
2	4	6	8	1/0	1/2	1/4	1/6	1/8	0
3	6	9	1/2	1/5	1/8	2/1	2/4	2/7	0
4	8	1/2	1/6	2/0	2/4	2/8	3/2	3/6	0
5	1/0	1/5	2/0	2/5	3/0	3/5	4/0	4/5	0
6	1/2	1/8	2/4	3/0	3/6	4/2	4/8	5/4	0
7	1/4	2/1	2/8	3/5	4/2	4/9	5/6	6/3	0
8	1/6	2/4	3/2	4/0	4/8	5/6	6/4	7/2	0
9	1/8	2/7	3/6	4/5	5/4	6/3	7/2	8/1	0

படம் 4
நேப்பியர் கட்டம்

நேப்பியர் கட்டங்களைப் பயன்படுத்தி 5 மற்றும் 148 என்ற இரண்டு எண்களை எங்ஙனம் பெருக்குவது என்பதைக் கீழ்க் கண்டவாறு விவரிக்கலாம். இந்தக் கணக்கைச் செய்ய, படம் எண் 4-இல் குறிப்பிட்டுள்ள அமைப்பின் வழியாக 5,1,4,8 என்ற வரிசைச் சட்டங்களைத் தெரிவு செய்து கொள்ளலாம். படம்

எண் 5-இல் இந்த அமைப்பு காட்டப்பட்டுள்ளது. முதல் சட்டத்தில் ஐந்தாவது வரிசையை எடுத்துக்கொண்டு அதற்கு நேராக 1, 4, 8 சட்டங்களில் குறிப்பிடப்பட்டுள்ள எண்களைக் கவனிக்கவும். இதில் சமநிலையில் 5, 0, 0 என்ற எண்களும் மேல் நிலையில் 2, 4 (20, 40) என்ற எண்களும் உள்ளன. விட்ட வாக்கில் [Diagonal] எண்களைக் கூட்டும் பொழுது நமக்கு 740 விடையாகக் கிடைக்கிறது. இவ்வாறு கூட்டும் பொழுது மிகுதி [Carry] இருக்குமாயின், உரியவகையில் அதை எண்ணுடன் சேர்த்துப் பெருக்கல் தொகையைக் கண்டறிய வேண்டும். 5 × 148 என்ற பெருக்கல் கணக்கைச் செய்யும் விதம் படம் எண் 5-இல் விளக்கப்பட்டுள்ளது.



படம் 5

நேப்பியர் கட்டம் மூலம் பெருக்கல் கணக்கு (5 × 148 = 740)

நேப்பியரின் கட்டங்களை அடிப்படையாகக் கொண்டு 1633-ஆம் ஆண்டு வில்லியம் ஆட்ரெட் [William Oughtred] என்ற ஆங்கில அறிஞர் ஒரு கருவியை உருவாக்கினார். இது சுருக்கெண் குறியீட்டு [Logarithm] முறையைக் கொண்டு கணக்கிடும் நியதியை உள்ளடக்கியது. இக்கருவி கணக்குச் சட்டம்

[Slide Rule] என்று அழைக்கப்படுகிறது. இக்கருவியில் எண்களைப் பெருக்குதலும் வகுத்தலும், சட்டங்களின் நீளவாக்கில் குறியீடு [Calibration] செய்யப்பட்டுள்ள அலகுகளைக் கூட்டியும் கழித்தும் செய்யப்பட்டன. இரண்டு நிலையான சட்டங்களும் அதன் இடையில் எளிதாக நகர்த்தும் முறையில் அமைந்த மற்றொரு சட்டமும் இக்கருவியில் பொருத்தப்பட்டிருந்தது. யானைத் தந்தத்தைக் கொண்டு சட்டங்கள் உருவாக்கப்பட்டதால் இதனை நகர்த்திக் கணக்கிடுவது மிகவும் எளிதாக இருந்தது. இச்சட்டங்களில் கணிதக் கூறுகளும் [Mathematical Functions] பிற எண்களும் கணக்கீட்டு முறையில் குறிக்கப்பட்டு இருந்தன. இக்கருவியானது பெரும்பாலான அறிவியல், மற்றும் பொறியியல் வல்லுநர்களால் கணக்குகளைத் தீர்வு காண்பதற்குச் சிறிது காலம் முன்புவரை பயன்படுத்தப்பட்டு வந்தது. ஆனால் விலையில் குறைந்த மின்னணுக் கணிப்பான்கள் [Electronic Calculators] தோன்றிய பிறகு, இவற்றின் பயன்பாடு வழக்கத்திலிருந்து மறைந்து வருகிறது.

2.2. கணிப்பொறியின் வளர்ச்சி

பின்னர், கி. பி. 1642-இல் பிளெய்ஸ் பாஸ்கல் [Blaise Pascal] என்ற பிரான்சு நாட்டுக் கணித மேதை, ஒரு பொறியை உருவாக்கினார். இது தற்காலத்தில் பயன்படும் பெரிய கணிப்பான்களின் [Calculators] உருவாக்கத்திற்கு அடிப்படைக் காரணியாய் அமைந்தது. மேலும் இக்கருவியே தற்காலக் கணிப்பொறி உருவாவதற்கு அடிகோலியது எனலாம். இவர் உருவாக்கிய கருவி சுற்றக்கூடிய பற்சக்கரங்களையும், சட்டங்களையும் கொண்ட அமைப்பாக இருந்தது. 10-ஐ அடிப்படையாகக் கொண்ட ஒவ்வொரு எண்ணடுக்கிற்கும் [Power of 10] ஒவ்வொரு சக்கரம் வீதம் இக்கருவியில் பத்துச் சக்கரங்கள் இருந்தன. ஒவ்வொரு சக்கரத்திலும் 0-விலிருந்து 9-வரை எண்கள் குறிக்கப்பட்டிருந்தன. இக்கருவியில் குறிப்பிட்ட நிலையில் சக்கரங்களைச் சுழற்றி எண்களைக் கூட்டுதலும் கழித்தலும் செய்ய முடிந்தது. ஆனால் நேரடியாக இக்கருவியைக் கொண்டு பெருக்கல், வகுத்தல் முதலிய செயல்களைச் செய்ய இயலவில்லை. அவருக்குப் பிறகு வந்த காட்ப்ரெயிட் வில் ஹெல்ம் லிப்னீட்ஸ் [Gottfried Wilhelm Leibnitz] என்பவர் கணிதத்தில் மிகவும் ஆர்வம் உடையவராய் இருந்தார். இவர் கணிதம், கடற்பயணம் முதலியவற்றில் பயன்படுத்துவதற்கென ஒரு கருவியை

உருவாக்கினார். இது பாஸ்கலின் பொறியை அடிப்படையாகக் கொண்டு அமைந்தது. இதன் அமைப்பு, பற்சக்கரங்களையும் உருளைகளையும் கொண்டதாக இருந்தது. இது கழித்தல், கூட்டல் மட்டுமன்றி பெருக்கல், வகுத்தல் ஆகிய செயல்களையும் செய்யும் தன்மை உடையதாய் விளங்கியது.

மேற்குறிப்பிட்ட லிப்னீட்ஸுக்குப் பிறகு, கணிதத் துறையிலே குறிப்பிடத்தக்க பணிகளைச் செய்த பெருமை கி. பி. 1820–ஐச் சேர்ந்த சார்லஸ் பாப்பேஜ் [Charles Babbage] என்ற ஆங்கிலக் கணிதவியல் அறிஞருக்கே உரியதாகும். இவரைத் தற்காலக் கணிப்பொறியின் தந்தை என்று கூறலாம். இவரது காலத்தில் காலக் கணிப்புக்காகப் பயன்படுத்தி வந்த காலக் கணிப்பு அட்டவணையில் தவறு நிறைந்திருந்தது. எனவே கணிதத்தில் பல சீர்திருத்தங்கள் செய்யவும், அவற்றைக் காலக் கணிப்பு, கடற்பயணம், அஞ்சல் துறைகளில் பயன்படுத்தவும் இவர் கருதினார். இதற்காக அவர் பல கண்டுபிடிப்புகளைச் செய்தார். கணிதத் திறமையைக் கொண்டும், கருவிகளைக் கொண்டும் தொழிலகங்களில் பொருள் உற்பத்தியைப் பெருக்க முடியும் என்பதை உறுதியாக நம்பினார். இதன் அடிப்படையில் ‘நடைமுறை ஆய்வுத்’ துறையில் [Operational Research] பல ஆய்வுப் பணிகளை மேற்கொண்டார். இந்த ஆய்வின் வழி, பல புதிய கண்டுபிடிப்புகளை நிகழ்த்தினார். எல்லாவற்றிலும் இவர் உருவாக்கிய ‘டிஃபரன்ஸ் எஞ்சின்’ [Difference Engine] மிகவும் பெயர் வாய்ந்தது. இந்தக் கருவி, குறியீடுகளைக் கொண்டு கணக்கிடப்பட்ட எண்களின் வேறுபாட்டால் வேலை செய்யும் திறன் பெற்றிருந்தது. எனவே இதற்கு இப்பெயர் இடப்பட்டது. இக்கருவியின் திருத்தப்பட்ட அமைப்பை உருவாக்குவதற்கு ஆங்கில அரசு இவருக்குப் போதிய பணவுதவி அளித்த போதிலும், உடனடியாக இவரால் கருவியை வடிவமைக்க முடியவில்லை. ஏனெனில் இவரது வடிவமைப்பிற்கு ஏற்ப இக்கருவியைச் செய்துதர அக்காலத்தில் நுட்பம் வாய்ந்த தொழிற்கூடங்கள் இல்லை. எனவே இதற்கென, வடிவமைப்பைத் திருத்துவதிலும் ஆயிரக்கணக்கான வரைபடங்களை உருவாக்குவதிலும் பாப்பேஜ் குறிப்பிடத்தக்க அளவு தமது நேரத்தைச் செலவு செய்தார்.

இத்திட்டத்திற்கென அரசு கொடுத்த மானியத் தொகையும் போதவில்லை. இக்கருவியை உருவாக்கும் பணியில் ஈடுபட்டு

இருந்த இவருடைய உதவியாளர் கூட இக்கருவிக்கெனப் பெற்ற பொருட்களைக் களவாடிச் சென்று விட்டார். இன்றைய காலத்தில் நாம் பெற்றிருக்கும் அறிவியல் வசதிகள் அக்காலத்தில் கிடைத்திருக்குமேயாயின், திறமை மிக்கதொரு கணிப்பொறியைப் பாப்பேஜால் உருவாக்க முடிந்திருக்கும் என்பது உறுதி.

போதிய வசதியின்மையால் சிரமப்பட்டுக் கொண்டிருந்த பாப்பேஜ், 'பகுப்பாய்வுக் கருவி' [Analytical Engine] ஒன்றை உருவாக்க எண்ணினார். இது வழியமைப்பு [Program], கணக்ககம் [Arithmetic unit], நினைவகம் [Memory] முதலியன கொண்ட அமைப்பாய் இருந்தது. இது தற்காலத்தில் உருவாக்கப்படும் நவீன கணிப்பொறிகளின் அடிப்படைத் தத்துவத்தைக் கொண்டிருந்தது, பாப்பேஜின் உயர்ந்த அறிவுத் திறனுக்குச் சான்றாகும். இது நினைவிருத்த வழியமைப்பு முறையில் [Stored Program Concept] செயற்பட்டது. இந்தப் பொறி வெவ்வேறு கணிதக் கூறுகளையும் [Mathematical Functions], சமன்பாடுகளையும் [Equations] வழியமைப்பு செய்து, தீர்வு காண்பதற்கு ஏற்ப அமைந்திருந்தது.

'டிஃபரன்ஸ் எஞ்சினின்' வடிவமைப்பு, குறிப்பிட்ட காலத்தில் உருவாக்கப்படாததால் கி. பி. 1842—ஆம் ஆண்டு அரசு தனது மானிய உதவியை நிறுத்திக் கொண்டது. வடிவமைப்பை பற்றிய கருத்து முற்றுப் பெற்றிருந்தபோதிலும், இதனை வடிவமைக்கத் தேவையான தொழில்நுட்பம் இன்மையால் இவரால் இக்கருவியைச் செய்து முடிக்க இயலவில்லை. அவருடைய காலத்தில் நிலவிய சமூக, அரசியல் நிலைக்கேற்ப புதிய கருத்துகளும், அறிவியல் உண்மைகளும் சமயவாதிகளாலும், மக்களாலும் ஏற்றுக்கொள்ளப்படவில்லை. மாறாக இவரை இழித்தும், பழித்தும் பேசிய சிலர் இவரது கண்டுபிடிப்பான கருவிகளை 'பாப்பேஜின் மடமைச் சின்னம்' என்று எள்ளி நகையாடி ஏளனம் செய்தனர். அதுமட்டுமன்றி, இவரது காலத்தில் வாழ்ந்த சில அறிஞர்களும் இவருடன் பகைமை கொண்டிருந்தனர். ஏனெனில் அவர்கள் உருவாக்கியிருந்த வான்வெளிப்பட்டியல் [Astronomical Table], கணிப்புப் பட்டியல் [Mathematical Table] ஆகியவற்றிலிருந்த தவறுகளை இவர் எடுத்துக் கூறினார்.

வான்வெளிப்பயணம், அறிவியல் ஆய்வு, இரசாயனம் முதலிய துறைகளில் கணித உண்மைகளையும், கணிதச் சமன்

பாடுகளையும் கொண்டு சிறந்த ஆராய்ச்சி செய்ய முடியும் என்று பாப்பேஜ் எடுத்துக் கூறினார். பாப்பேஜின் பகுப்பாய்வுக் கருவியைப் [Analytical Engine] பயன்படுத்திய பெருமை லேடி லவ்லேஸ் [Lady Lovelace] என்ற பெண்மணியையே சாரும். இவர் ஆங்கிலக் கவிஞர் லார்டு பைரனின் [Lord Byron] மகள் ஆவார். நல்ல கணித அறிவு வாய்க்கப் பெற்றிருந்த இவர், பகுப்பாய்வுக்கருவியில் பல கணக்குகளைச் செய்து அதன் நிறைகளையும், குறைகளையும் எடுத்துக் கூறிப் பெருமை பெற்றார். இந்த விதத்தில் உலகின் முதல் வழியமைப்பாளர் [Programmer] இவரே ஆவார்.

தற்காலத்தில் வழங்கும் கணிப்பொறிகளைப் போலவே வழியமைப்புகளை இடுவரல் [Input] செய்தல், பிரிவு ஆணை முறை [Branch Instructions], பதிவு செய்தல் [Storing] முதலிய வசதிகளைப் பெற்று, பாப்பேஜின் பகுப்பாய்வுக் கருவி அமைந்திருந்தது. எனவே இது காலக் கணிப்புப் பட்டியல் செய்தல், கடற்பயணம், அஞ்சல், வான்வெளிப் பட்டியல் தயாரித்தல் முதலியவற்றில் பயன்படுத்தப்பட்டது.

தற்காலத்தில் செய்திக் கூறுகளையும் [Data], வழியமைப்புகளையும் [Programs] கணிப்பொறிக்கு இடுவரல் செய்யப் பயன்படுத்தப்பட்டு வரும் துளையிடப்பட்ட அட்டை [Punched Cards] முதன் முதலில் கி.பி. 1882-ஆம் ஆண்டு கண்டறியப்பட்டது. இதை முதலில் கண்டறிந்த பெருமை ஹெர்மன் ஹாலரித் [Herman Hollerith] என்ற அமெரிக்காவைச் சார்ந்த மக்கள் தொகைக் கணக்கெடுப்பு நிபுணரையே சாரும். எனினும் கி.பி. 1880-இலேயே நெசவுப் பொறியில் வண்ண அமைப்புகளை [Colour Pattern] உருவாக்கும் பொருட்டு, பிரான்சு நாட்டைச் சார்ந்த ஜோசப் மேரி ஜாக்வார்டு [Joseph Marie Jacquard] என்ற அறிஞர் இந்த அட்டைகளைப் பயன்படுத்தி உள்ளதாகத் தெரிகிறது எனினும் முதன் முதலில் கணிப்பொறியில் விவர ஆய்வனை [Data Processing] செய்ய இந்த அட்டை முறையைப் பயன்படுத்தியர் ஹெர்மன் ஹாலரித் என்பவரே ஆவர். 1880ஆம் ஆண்டு அமெரிக்க மக்கள் தொகைக் கணக்கெடுப்பின் முடிவு தெரிய எட்டு ஆண்டுகள் ஆயின. இதனை விரைவில் செய்யக் கருதியே, துளையிடப்பட்ட அட்டைகள் புள்ளி விவரங்களைப் பதிவு செய்யப் பயன்படுத்தப்பட்டன. இதன் மூலம் இந்த மக்கள் தொகைக் கணக்கெடுப்பு மூன்று ஆண்டுகளில் திறம்படச் செய்து முடிக்கப்பட்டது.

இம்முறையின் வெற்றியை உணர்ந்த பிற நாடுகளான உருசியா, ஆஸ்திரியா, கனடா முதலியவை கி.பி.1890-லிருந்து மக்கள் தொகைக் கணக்கெடுப்பிற்கு இம்முறையைப் பின்பற்ற லாயின. இதன் பிறகு காப்பீட்டுக் கழகங்களும் [Insurance Corporations] பிற நிறுவனங்களும் தங்களது தொழில்துறையின் சிக்கல்களுக்குத் தீர்வு காண, இம்முறையைப் பின்பற்றலாயின. இதன் பின்னர் ஹாலரித் [Hollerith], 'டேபுலேட்டிங் மெஷின் கம்பெனி' [Tabulating Machine Company] என்ற ஒரு நிறுவனத்தை அமைத்தார். இதன்மூலம் அட்டைகளில்உள்ள செய்திகளைப் பயன்படுத்த கணக்கீட்டுக் கருவிகள் [Calculating Machines] உருவாக்கப்பட்டன. பின்னர் 1924-ஆம் ஆண்டில் இக்கருவிகளை உருவாக்குவதற்கென ஐ.பி.எம் [I.B.M - International Business Machines] என்ற புகழ் வாய்ந்த தொழில் நிலையம் உருவாக்கப் பட்டது. இதுவே பிற்காலத்தில் ஏராளமான திறன்மிகு கணிப் பொறிகளை உருவாக்கி உலகப் புகழ்பெற்றது.

கி.பி.1930-க்கும் 1940-க்கும் இடைப்பட்ட காலத்தில் கணிப்பொறியை உருவாக்குவதற்கான பல துரிதமுயற்சிகள் மேற் கொள்ளப்பட்டன. ஏனெனில் இக்காலம் இரண்டாம் உலகப் போர் நிகழும் தறுவாயில் இருந்தமையால் அரசாங்கமும் பாதுகாப்பு நிறுவனங்களும் கணிப்பொறியின் உருவாக்கத் திட்டத்திற்குப் பெரிதும் ஊக்கமளித்தன. எனவே இக்காலத்தில் கணிப்பொறி உருவாவதற்கான பலவித முயற்சிகள் ஒரே சமயத் தில் மேற்கொள்ளப்பட்டன. அறிவியல் கணிப்பு [Scientific Computation] செய்வதற்கும் ஏவுகணைகளைப் பற்றிய கணிப்புப் பட்டியலை உருவாக்குவதற்கும் கணிப் பொறியின் உதவி பெரிதும் தேவைப்பட்டது.

2.3. கணிப்பொறியின் வடிவமைப்பில் முன்னேற்றம்

பாப்பேஜுக்குப் பிறகு மசாஜுசெட்ஸ் பல்கலைக் கழகத் தில் [Massachusetts Institute of Technology] 1925-ஆம் ஆண்டு வேறுபாட்டுக் கருவி [Differential Analyser] என்ற எந்திர விசையால் இயங்கக்கூடிய கணிப்பொறியை டாக்டர் வன்னிவர் புஷ் என்ற அறிஞர் வடிவமைத்தார். இதன் பிறகு அமெரிக்காவில் பெல் தொலைபேசி நிறுவனத்தில் [Bell Telephone Laboratory] மேற்கொள்ளப்பட்ட முயற்சி காரணமாக, பெல் இயங்குதடைக் கணிப்பொறிகள் [Bell Relay Computers] உருவாக்கப்பட்டன.

இவை முந்தைய கணிப்பொறிகளைவிடத் திறன் மிக வாய்க்கப் பெற்றிருந்தன. இயக்கவேகத்தில் சுற்றும் சக்கரங்களைக்கொண்ட [Rotating Wheels] கணக்கிடு கருவிகளைவிட, இக்கருவி பன் மடங்கு வேகத்திறன் கொண்டிருந்தது. கி.பி. 1930-ஆம் ஆண்டில் அயோவா பல்கலைக் கழகத்தின் [Iowa University] இயற்பியல் துறையில் ஜான் வின்சன்ட் அட்டனாசோஃப் [John Vincent Atanasoff] என்ற அறிஞர் மின்னணு முறையிலான கணிப்பானை [Electronic Calculator] உருவாக்கும் முயற்சிக்கு வித்திட்டார். இதுவே கி.பி. 1939-இல் ஒரு திறம் வாய்ந்த அட்டனாசோஃப் பெர்ரி [Atanasoff Berry] என்ற கணிப்பொறியாய் அமைந்தது. பல சமன்பாடுகளுக்குத் தீர்வு காணும் திறன் பெற்றிருந்தமையும் இருநிலை எண்ணை [Binary] அடிப்படை எண்ணாகக் கணக்கீடு களுக்குப் பயன்படுத்தியமையும் இதன் சிறப்பியல்புகளாகும். 1973 -ஆம் ஆண்டில் இது பற்றிய ஒரு வழக்கு மின்னியா பொலிஸ் [Minneapolis] நீதி மன்றத்தில் கொண்டு வந்த பிறகே, இதன் பெருமை உலகிற்குத் தெரிந்தது.

1944-ஆம் ஆண்டு மார்க்-1 [Mark-I] என்ற தானியங்கிக் கணிப்பொறி உருவாக்கப்பட்டது. ஹார்வார்டு பல்கலைக் கழகக் [Harvard University] கணிதப் பேராசிரியரான டாக்டர் ஹோவார்ட் ஐக்கன் [Howard Aiken] என்பவர் இக்கணிப் பொறியை உருவாக்கினார். இது முற்றிலும் இயங்கு தடைகளைக் [Relays] கொண்டு வடிவமைத்து உருவாக்கப்பட்டது. இப்பணி யைச் செய்வதற்கு இன்று உலகில் தலைசிறந்து விளங்கும் ஐ.பி.எம். [I.B.M -International Business Machines] நிறுவன மும் அமெரிக்கக் கடற்படையும் உதவி புரிந்தன. இக்கருவி சுமார் 100 ஆண்டுகளுக்கு முன்பாக பாப்பேஜ் கண்டறிந்த தத்து வத்தை அடிப்படையாகக்கொண்டு இயங்கியது. இந்தக் கணிப் பொறி, துளைத்தாள் நாடா [Punched Paper Tape] மூலம் செய்முறை ஆணைகளைப் [Operating Instructions] பெற்றுச் செயற்படும் தன்மையதாய் இருந்தது. 20 மனிதர்கள் செய்யக் கூடிய பணியைச்செய்யும் திறனை இக்கருவி பெற்றிருந்தது. பிறகு மின்னணுக்குமிழ்களைக் கொண்ட [Electronic Tubes] மார்க்-3 [Mark-III] என்ற கணிப்பொறி உருவாக்கப்பட்டது. மார்க் வகைக் கணிப்பொறிகள் வெளிப்புறக் கம்பியிணைப்பு முறை யினால் [External Plug in/out wire system] செயற்பட்டன. எனவே இதனைத்திறம்பட இயக்குவதிலும் வழியமைப்புகளையும்

[Programs] செய்திக் கூறுகளையும் [Data] இடுவரல் [Input] செய்வதிலும் மிகவும் துன்பம் இருந்தது.

2.4. கணிப்பொறியின் காலக்கட்டப் பிரிவு (GENERATION FEATURE OF A COMPUTER)

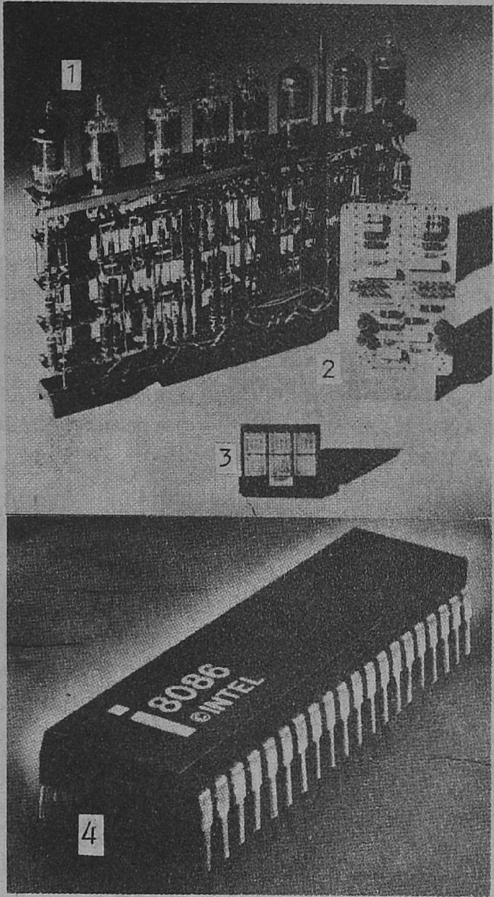
அதன் பிறகு அபர்டீனில் [Aberdeen] உள்ள ஓர் ஆய்வு மையம், பென்சில்வேனியா பல்கலைக் கழகத்தில் [Pennsylvania University] உள்ள மின்னணுவியல் பள்ளியில் பெரியதொரு கணிப்பொறியைஉருவாக்கும் முயற்சிக்குப்பொருளுதவி செய்தது. இதன் பயனாக 1945-இல் முற்றிலும் மின்னணு [Electronic] முறையில் செயற்படும் 'எனியாக்' [ENIAC-Electronic Numerical Integrator And Calculator] என்ற கணிப்பொறி உருவாக்கப்பட்டது. இதனை உருவாக்கிய பேரறிஞர்கள் டாக்டர் ஜான்மாச்லே[John Mauchly], பிரீஸ்பர் எக்கார்ட் [Presper Eckert] ஆகியவர்கள் ஆவர். இக்கணிப்பொறியில் முந்தைய கணிப்பொறிகளில் பயன்படுத்தப்பட்ட இயங்குதடைகளுக்குப் [Relay] பதிலாக மின்னணுக் குமிழ்கள் பயன்படுத்தப்பட்டன. இதன் காரணமாக இவற்றின் வேகம் முந்தைய கணிபொறிகளை விட ஆயிரம் மடங்கு அதிகமாய் இருந்தது. தற்கால நவீனக் கணிப்பொறிகள் உருவாவதற்கு அடிகோலியது எனியாக் என்று குறிப்பிடலாம்.

இந்த எனியாக் கணிப்பொறியில் 18,000 மின்னணுக் குமிழ்களும் 70,000 மின்தடைகளும் [Resistors], 10,000 மின்தாங்கிகளும் [Capacitors] இருந்தன. இது 5,00,000 மின் இணைப்புகளையும் [Electric Connections] 6,000 விசை களையும் [Switches] 5,000 மின் முனைகளையும் [Electric Terminals] கொண்டிருந்தது. இதை இயக்குவதற்குச் சுமார் 150 கிலோவாட் [Kilo Watt] மின்சக்தி தேவைப்பட்டது. சுமார் 30டன் எடை இருந்த இக்கணிப்பொறி 15,000 சதுர அடிகள் உள்ள தளத்தையும் அடைத்திருந்தது. இக்கருவியானது ஒரு நொடிக்கு 5,000 கூட்டல், கழித்தல்களையும் சுமார் 300 பெருக் கல்களையும் செய்யும் செயலாற்றல் பெற்றிருந்தது. இதிலிருந்து வெளிப்பட்ட வெப்பம் காரணமாக 7 நிமிடங்களுக்கு 1 மின்னணுக்குமிழ் வெடித்ததாகக் கூறப்படுகிறது. இதனுடைய செயற்காலத்தில் சுமார் 80,000 மணி நேரங்கள் பல பணிகளுக் காக இக்கருவி பயன்படுத்தப்பட்டது. இதில் சில குறைபாடுகள் இருந்தன. செய்திக் கூறுகளை உட்செலுத்தவும், விடையைப்

[Output] பெறவும் துளையிடப்பட்ட அட்டைகள் பயன்படுத்தப் பட்டன. எனினும் வழியமைப்புக்களைச் செயற்படுத்த, கருவியின் பாகங்களை முறையாக இணைக்க வேண்டியிருந்தது. இது மிகவும் கடினமான பணியாகும். மேலும் இதில் குறைந்த அளவு விவரங்களையே பதிவு செய்து பயன்படுத்த முடிந்தது. இக்கருவியில் இணைப்புகளைக் கொண்டு வழியமைப்பு செய் வதற்கும் பின்னர் அதை மாற்றுவதற்கும் சில நிமிடங்களில் செய்து முடிக்கக்கூடிய பணிகளுக்குக் கூட சில நாள்கள் ஆயின. இது இக்கருவியின் பயன்பாட்டைப் பெரிதும் பாதித்தது.

'எனியாக்' கணிப்பொறியில் இருந்த குறைகளை நீக்கவும் வழியமைப்பு ஆணைகளைச் [Programming Instructions] செய்திக் கூறுகளைப் போன்றே பதிவு செய்து வைத்துக்கொள்ள வும் ஒரு முயற்சி மேற்கொள்ளப்பட்டது. ஜான் வான் நியூமன் [John Von Neumann] என்ற அறிஞர் இம்முறையைக் கண்டறிந் தார். எனவே இவ்வாறு உருவாக்கப்பட்ட கணிப்பொறிக்கு 'நியூமன் மெஷின்' [Neumann Machine] என்று பெயரிடப் பட்டது. இத்திட்டத்திற்குப் பர்க்ஸ், கோல்ட்ஸ்தீன் [Gold Stine] முதலான அறிஞர்கள் உதவி புரிந்தனர். *1952*-ஆம் ஆண்டு பிரின்ஸ்டன் [Princeton] பல்கலைக் கழகத்தில் வான் நியூமனின் மேற்பார்வையின் கீழ் 'நினைவிருத்த வழிமுறை'க் [SPC-Stored Program Concept] கணிப்பொறி உருவாக்கப் பட்டது. இதைத் தொடர்ந்து இல்லியாக் [ILLIAC], மானியாக்-1 [MANIAC-1], ஜானியாக் [JANIAC] ஆகிய கணிப்பொறிகள் உருவாக்கப்பட்டன. மானியாக் என்ற கணிப்பொறியின் உதவியைக் கொண்டு அமெரிக்க அரசு உலகப் போரில் நீர்வாயுக் குண்டைத் தயாரித்தது.

எனியாக் கணிப்பொறியை உருவாக்கிய டாக்டர் மாச்லே, எக்கார்ட் ஆகிய அறிஞர்கள் 'யுனிவேக்' [UNIVAC] என்ற கணிப் பொறித் திட்டத்தைச் செயற்படுத்தினர். இம்முறையில் உருவாக்கப்பட்ட கணிப்பொறிகள் மின்குமிழ்களுக்குப் பதிலாக இருங்கடத்திகளைப் [Diode] பயன்படுத்தியதால் பன்மடங்கு அதிக வேகம் பெற்றிருந்தன. இதில் செய்திகளைப் பதிவு செய்யவும் பல கணிப்புகளைச் செய்யவும் புற ஒருங்குகளைப் [Peripheral Units] பயன்படுத்தவும் வசதி செய்யப்பட்டிருந்தது. இவை யாவும் முதலாம் காலக்கட்டத்தைச் சார்ந்த [First generation] கணிப்பொறிகளாகும். அதன்பின்னர் சுமார்



படம்— 6

நான்கு காலக்கட்டங்கள்

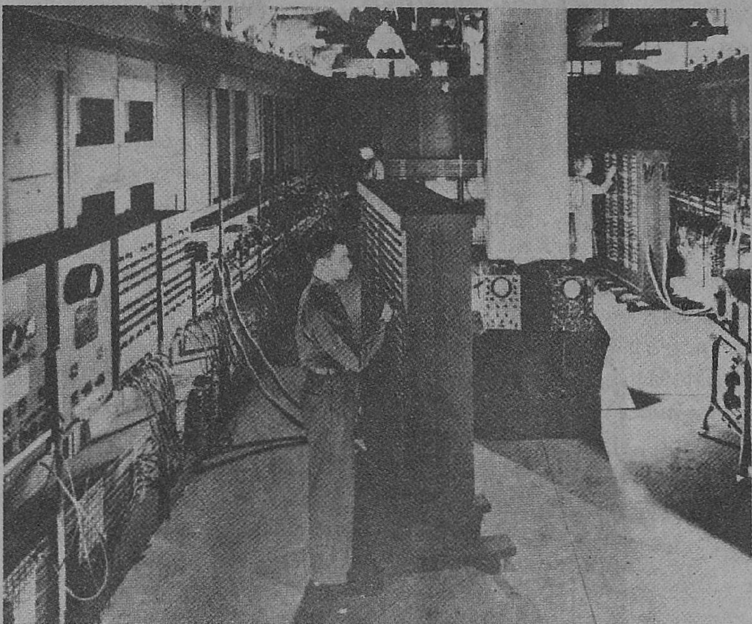
[Four Generations]

1 மின்னணுக்குமிழ்கள் [Electronic Tubes]
2 ட்ரான்ஸிஸ்டர் [Transistor]
3 ஒருங்கமை மின்சுற்றுகள் [Integrated Circuits]
4 நுண்ணியக்கிகள் [Micro-processors]

1960–65–இல் நிறைவிலா மின் கடத்திகளைக் கொண்டு [Semiconductor Transistors] உருவாக்கப்பட்ட இரண்டாம்காலக் கட்டக் கணிப்பொறிகளும் [Second Generation], *1965–70*–இல் மூன்றாம் காலக்கட்டத்தைச் சார்ந்த கணிப்பொறிகளும் [Micro Miniaturisation-VLSI] உருவாக்கப்பட்டன. *1970*க்குப் பிறகு வடிவமைக்கப்பட்டு உருவாக்கப்பட்ட கணிப்பொறிகள் அனைத்தும் நான்காம் காலக்கட்டத்தைச் சார்ந்தனவாகக் கருதப்படுகின்றன. மேற்குறிப்பிட்டவாறே கணிப்பொறியின் உருவாக்கத்திலும் விரிவாக்கத்திலும் பல்வேறுவிதமான முன்னேற்றங்கள் உண்டாயின. கணிப்பொறித்துறையில் மறு மலர்ச்சி பல்வேறு காலக்கட்டங்களில் நிகழ்ந்துள்ளமை கீழ்க் கண்டவாறு விளக்கப்பட்டுள்ளது.

2.4.1. முதலாம் காலக்கட்டக் கணிப்பொறி (*FIRST GENERATION COMPUTER*)

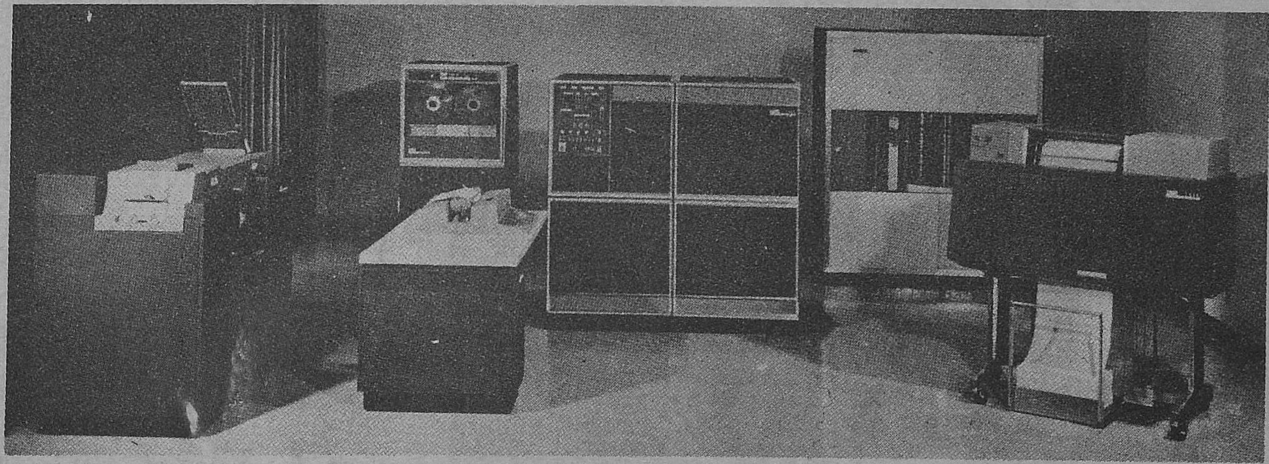
முற்றிலும் மின்னணு முறையில்இயங்கிய எனியாக் [ENIAC] என்ற கணிப்பொறி *1950*-ஆம் ஆண்டின் முற்பகுதியில் நிறுவப் பட்டது. முதலாம் காலக்கட்டத்தைச் சார்ந்த எனியாக் [ENIAC] கணிப்பொறியின் அமைப்பு, படம் எண் 6-(அ) -இல் காட்டப் பட்டுள்ளது. இது தொழில் முறையில் (ரீதியில்) பயன்படுத்தத் தகுதி வாய்ந்ததாகக் கருதப்பட்டது. இதற்கு முன்பு அமைக்கப் பட்ட ஐ.பி.எம் [IBM] நிறுவனத்தில் கணிப்பொறி சார்ந்த பல கருவிகள் உருவாக்கப்பட்டன. அவை யாவும் கணிப்பொறியின் புற ஒருங்குகளான [Peripherals] அட்டை துளையிடும் கருவி, பட்டியிலிடும் கருவி போன்றவையேயாகும். ஐ. பி. எம். நிறுவனம் முதன் முதலில் உருவாக்கிய பெரிய கணிப்பொறி IBM 701 என்பதாகும். எனினும் மின்னணுக்குமிழ்களையும் மின் இணைப்புகளையும் கொண்டு உருவாக்கப்பட்ட எனியாக் கணிப் பொறி முதலாம் காலக்கட்டத்தைச் சார்ந்ததாகக் கருதப் பட்டது. இதில் பல குறைபாடுகள் காணப்பட்டன. குறைந்த வேகத்தையுடைய இடுவரல் [Input], விடுவரல்களையும்[Output] குறைந்த நினைவகத்தையும் பெற்று இருந்தன. கணிப் பொறி இயக்கத்திற்குத் தேவையான செயலாக்கக் கணிமம் [System Software] போதிய வசதிகளைப் பெற்றிருக்கவில்லை. எடுத்துக்காட்டாக IBM 705, யுனிவேக் [UNIVAC] முதலிய கணிப் பொறிகள் இவ்வகையைச் சார்ந்தவையாகும்.



படம்—6(அ)

எனியாக்.முதலாம் காலக்கட்டக் கணிப்பொறி

(ENIAC–Electronic Numerical Integrator And Calculator)



படம்—7

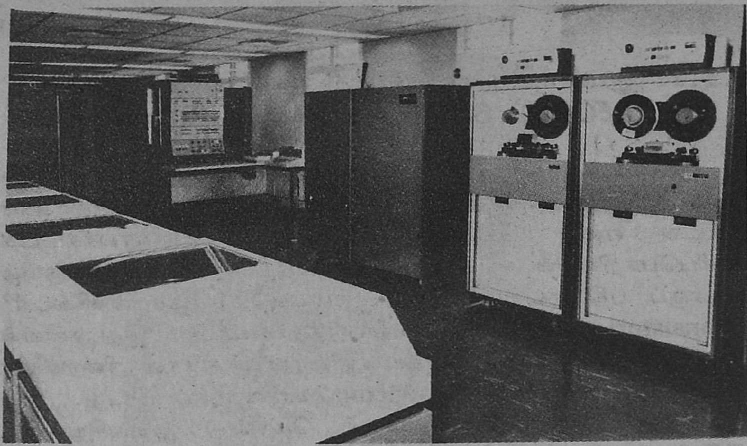
IBM-1401-இரண்டாம் காலக்கட்டக் கணிப்பொறி

2.4.2. இரண்டாம் காலக்கட்டக் கணிப்பொறி (SECOND GENERATION COMPUTER)

இரண்டாம் காலக்கட்டக் கணிப்பொறிகள் சுமார் 1960—1965-ஆம் ஆண்டுகளுக்கு இடைப்பட்ட காலத்தில் உருவாக்கப் பட்டன எனக் கொள்ளலாம். முதலாம் காலக்கட்டக் கணிப்பொறிகளில் பயன்படுத்தப்பட்ட மின்னணுக்குமிழ் களுக்குப் பதிலாக முக்கடத்தி எனப்படும் 'ட்ரான்ஸிஸ்டர்' [Transistor] என்ற உறுப்பு பயன்படுத்தப்பட்டது. இது அளவில் மிகச்சிறியதும் இயக்கச் சக்தியில் விரைவானதுமான நிறைவிலா மின்கடத்தியால் [Semi Conductors] உருவாக்கப்பட்டது. இது குறைந்த அளவு மின்சக்தியைக் கொண்டு இயங்கியதால், இதிலிருந்து வெப்பம் வெளிப்பட வில்லை. இக்கணிப்பொறிகள் தொழிலகங்களில் விவர ஆய்வனை செய்வதற்கும், அறிவியற் கணிப்புகளைச் [Scientific Computation] செய்வதற்கும், பயன் படுத்தப்பட்டன, இரண்டாம் காலக்கட்டத்தைச் சார்ந்த கணிப் பொறிகள் அதிக அளவு நினைவகத்தையும் [Memory], செயலாக்கக் கணிமங்களையும் [SystemSoftware], பலவிதமான வழியமைப்பு மொழிகளையும் [Programming Languages] பெற்றிருந்தன. IBM-1620, IBM-1400, IBM-7094 முதலிய கணிப்பொறி களை இவ்வகைக்கு எடுத்துக்காட்டாகக் கூறலாம். இரண்டாம் காலக்கட்டக் கணிப்பொறியான IBM-1401 படம் எண் 7-இல் காட்டப்பட்டுள்ளது.

2.4.3. மூன்றாம் காலக்கட்டக் கணிப்பொறி (THIRD GENERATION COMPUTER)

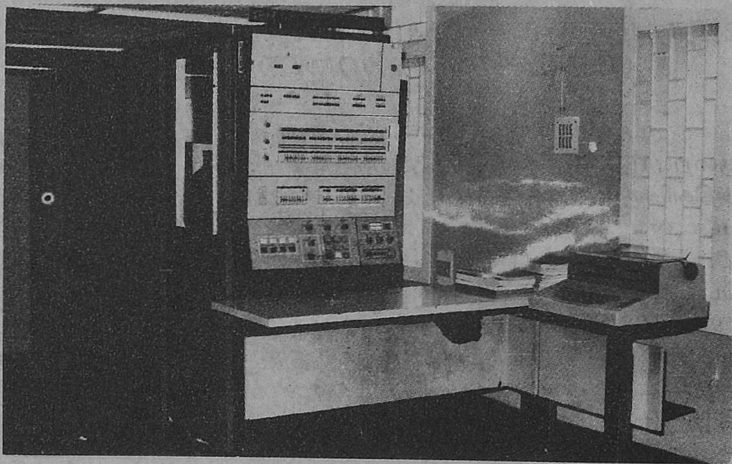
முதல் இரண்டு காலக்கட்டக் கணிப்பொறிகளில் இருந்த குறைபாடுகள் நீக்கப்பெற்று, கணிப்பொறியின் செயல் திறனை அதிகரிக்கும் முயற்சிகள் பல இக்காலக்கட்டத்தில் மேற்கொள்ளப் பட்டன. மூன்றாம் காலக்கட்டக் கணிப்பொறிகள் ஒருங்கமை மின்சுற்றுகளைக் [Integrated Circuits] கொண்டு உருவாக்கப் பட்டன. மூன்றாம் காலக்கட்டக் கணிப்பொறிகள் கி.பி. 1965-1970-ஆம் ஆண்டுக்குள் வடிவமைக்கப்பட்டு உருவாக்கப் பட்டன இது முந்தைய கணிப்பொறிகளைக் காட்டிலும் பல நூறுமடங்கு அதிகமான செயல்திறன் பெற்றிருந்தது.



படம்—8 [அ , ஆ]

IBM-360_மூன்றாம் காலக்கட்டக் கணிப்பொறி

படஉதவி: இராமானுசம்
கணிப்பொறிமையம்
அண்ணா பல்கலைக் கழகம்
நன்றி :



இரண்டாம் காலக்கட்டக் கணிப்பொறியில் பயன்படுத்தப் பட்ட ட்ரான்சிஸ்டருக்குப் பதிலாக அதைவிடப் பன்மடங்கு செயல்திறன் வாய்ந்த மிகச் சிறிய அளவிலான ஒருங்கமை மின்சுற்றுகள் இதில் பயன்படுத்தப்பட்டன. இதன் காரணமாக கணிப்பொறியின் அளவும் விலையும் குறிப்பிடத் தக்க அளவில் குறைந்தன. மேலும் இது நிறைவிலா மின் கடத்தியாலான [Semi Conductors] நினைவகத்தையும் வசதியைப் பகிர்ந்தி யங்கும் [Resource Sharing] திறனையும் பெற்றுள்ளது. மூன்றாம் காலக்கட்டத்தைச் சார்ந்த கணிப்பொறிகள், ஒருமித்த இயக்க வசதி [Concurrent Processing], பன்வழியமைப்புச் செயற்பாடு [Multi Programming], பன்மைய இயக்கம் [Multi Processing] ஆகிய பல வசதிகளைப் பெற்றுவேகத்திலும் பயன்பாட்டு முறையிலும் சிறந்து விளங்குகின்றன. எடுத்துக் காட்டாக IBM-360, IBM-370, CDC-7600, PDP-11 ஆகிய கணிப் பொறிகள் இந்தக் காலக்கட்ட வகையைச் சார்ந்தனவாகும். மூன்றாம் காலக்கட்டத்தைச் சார்ந்த IBM-360, என்ற கணிப் பொறியின் அமைப்பு, படம் 8–இல் காட்டப்பட்டுள்ளது.

2.4.4. நான்காம் காலக்கட்டக் கணிப்பொறி (*FOURTH GENERATION COMPUTER*)

கி. பி. 1970–75–இற்கு இடைப்பட்ட காலத்தில் மின்னணுத் துறையின் வளர்ச்சி குறிப்பிடத்தக்க விதத்தில் முன்னேறி யிருந்தது. திறன் வாய்ந்த பல மின் நுண் கருவிகள் [Electronic Components] இக்காலத்தில் உருவாக்கப்பட்டன.

மூன்றாம் காலக்கட்டக் கணிப்பொறியில் பயன்படுத்தப்பட்ட ஒருங்கமை மின்சுற்றுகளுக்குப் பதிலாக, ஆயிரக்கணக்கான மின்சுற்றுகளை உள்ளடக்கிய ஒருங்கமை மின் சுற்றுகள் [Large Scale Integrated Circuits-LSI] என்ற மின் இணைப்பிகள் பயன் படுத்தப்பட்டன. இது அளவில் மிக மிகச் சிறியதாகவும் முந்தைய கணிப்பொறியின் வேகத்தினைவிட நூற்றுக்கணக்கான அளவு அதிவேகத் திறன் உடையதாகவும் இருந்தது. இதன் காரணமாக ஆயிரக்கணக்கான மின் சுற்றுகளை உள்ளடக்கியதும் கணிப் பொறியின் இயக்கத் திறனுக்குத் தேவையான மின்சுற்றுகளைக் கொண்டதுமான நுண்ணியக்கிகள் [Micro processors] தோன்ற லாயின. எடுத்துக்காட்டாக, INTEL-8080, Z-80 போன்ற நுண்ணியக்கிகளைக் கூறலாம். தற்காலத்தில் INTEL-8080,Z-80

மற்றும் பல நுண்ணியக்கிகளைக்கொண்டு உருவாக்கப்படும் மிகச் சிறிய மற்றும் சிறிய [Micro, Mini Computers] கணிப்பொறிகள் நான்காம் காலக்கட்டத்தைச் சார்ந்தவைகளாகும். Z-80 என்ற நுண்ணியக்கியைக் கொண்டு உருவாக்கப்பட்ட AL-2016 என்ற கணிப்பொறியின் அமைப்பு, படம் எண் 9—இல் காட்டப்பட்டு உள்ளது. இவை முந்தைய காலக்கட்டக் கணிப்பொறிகளைக் காட்டிலும் அளவில் பன்மடங்கு சிறியனவாகவும் செயல்திறனில் பலநூறு மடங்கு வேகமுடையனவாகவும் பன்மடங்கு நினைவகத் திறன் வாய்ந்தனவாகவும் மிகக் குறைந்தவிலையில் கிடைக்கக் கூடியனவாகவும் உள்ளன.

அறிவியல், விண்வெளி ஆய்வு, தொலை தொடர்பு, ஆய்வுக் கூடங்கள் ஆகியவற்றில் செய்திகள் நாள்தோறும் மலையெனப் பெருகுகின்றன. இதன் பொருட்டே செய்திகளை விவர ஆய்வனை செய்வதற்கும் மனிதனால் தீர்வு காணமுடியாத சில கணக்குகளைத் தீர்வு காண்பதற்கும் கணிப்பொறியின் சேவை மிகவும் தேவையாய் உள்ளது. இதன் பயன்பாட்டிற்கென விலையிற் குறைந்த, சிறந்த கணிப்பொறிகளை உருவாக்குவதற்கு உலகெங்கிலும் உள்ள பல நிறுவனங்கள் முயன்று வருகின்றன. ஐ.பி.எம். நிறுவனமானது உலகெங்கிலும் பல்லாயிரக் கணக்கான கணிப்பொறிகளை நிறுவியுள்ளமை, தொழிற்சாலை களில் கணிப்பொறியின் தேவையைத்தெற்றெனப் புலப்படுத்தும். PDP-11 முதலிய சிறு கணிப்பொறிகளும் [Mini Computers] நுண்ணியக்கிகளைக் கொண்டு உருவாக்கப்படும் மிகச்சிறிய கணிப் பொறிகளும், [Micro Computers] வீடுகளில் பயன்படுத்துவதற் கென உருவாக்கப்படும் மனைக் கணிப்பொறிகளும் [Home Computers] தற்காலத்தில் குறைந்த செலவில் எளிதில் கிடைக்கக் கூடியனவாய் உள்ளன. ஒரு புள்ளி விவரப்படி அமெரிக்க நாட்டில் 1950—இலிருந்து 20வருட காலத்திற்குள் சுமார் 90,000 பெரிய கணிப்பொறிகள் நிறுவப்பட்டுள்ளமை இதன் முக்கியத்துவத்தை நன்கு எடுத்துக்காட்டும்.

பல நூறு மடங்குகள் வேகத்திறனும் செயற்பாட்டு மேன்மையும் பெற்று, குறைந்த விலைக்குக் கிடைக்கக்கூடிய ஐந்தாம் காலக்கட்டக் கணிப்பொறியை வடிவமைத்து உருவாக் குவதில் மேலை நாடுகளின் பல்கலைக் கழகங்கள், ஆராய்ச்சி நிறு வனங்கள் மற்றும் பல கணிப்பொறி நிறுவனங்கள் மிகத்தீவிரமாக ஈடுபட்டுள்ளன. 1984-ஆம் ஆண்டு முற்பகுதியில் கிடைத்த செய்தி

படம்--9 AL 2016 கணிப்பொறி
(கணிப்பு மையம் : தமிழ்ப் பல்கலைக் கழகம்)

களின்படி அமெரிக்காவிலுள்ள கார்னீகி மெலான் பல்கலைக்கழகத்தில் [Carnegie Mellon University] 1985–ஆம் ஆண்டுக்குள் அங்குப் பயிலும் ஒவ்வொரு மாணாக்கரும் ஒரு கணிப்பொறியைச் சொந்தமாகப் பயன்படுத்தும் நிலைமை ஏற்படுமெனில் கல்வித் துறையில் கணிப்பொறியின் இன்றியமையாமை நன்குபுலனாகும்.

2.5. எண்ணிலக்க மற்றும் ஒப்புமைக் கணிப்பொறி (DIGITAL AND ANALOG COMPUTER)

செய்திக்கூறுகளை இடுவரல் செய்யும் நிலையையும் அதன் செயலாக்க வடிவமைப்பு [Operational Design] முறையையும் பொறுத்து, கணிப்பொறிகளை இரண்டு பிரிவுகளாகப் பிரிக்கலாம். ஒன்று மற்றொன்றிலிருந்து சற்று வேறுபட்டுள்ளது. இக்கணிப்பொறிகள் ஒவ்வொன்றும் ஒரு குறிப்பிட்ட பயன்பாட்டிற்கென உதவுகின்றன. இவற்றைக் கீழ்க்கண்டவாறு ஆராய்ந்தறியலாம்.

2.5.1. ஒப்புமைக் கணிப்பொறி (ANALOG COMPUTER)

இந்த வகைக் கணிப்பொறிகள் செய்திக் கூறுகளை இடுவரலாகப் பெறும் விதத்தில் மற்ற கணிப்பொறிகளிலிருந்து வேறுபடுகின்றன. இவை எண்களுக்குப் பதிலாகத் தொடர்ச்சியான அளவை [Analog Signal] முறையில் இடுவரலைப் பெற்றுச்செயற்படும் தன்மை வாய்ந்தனவாகும். மின்சார அளவைகளான வோல்ட் [Volt], ஆம்பியர் ஆகியவற்றை வோல்ட் மீட்டர் [Volt meter], ஆம்பியர் மீட்டர்களில் [Ammeter] படித்தறிதலும் காற்றழுத்தத்தைக் காற்றழுத்த மானியில் படித்தறிந்து கணித்தலும் இவ்வகையைச் சார்ந்தனவாகும். இது எண்ணிலக்கக் கணிப்பொறியைக் காட்டிலும் வேகத்திலும் துல்லியத்திலும் [Precision] குறைந்ததாகும்.

2.5.2. எண்ணிலக்கக் கணிப்பொறி (DIGITAL COMPUTER)

இதன் இயக்கம் எண்களை அடிப்படையாகக் கொண்டு அமைந்தது. இடுவரல் மூலம் கணிப்பொறிக்கு இடப்படும் ஆணைகள் மற்றும் செய்திக்கூறுகள் முதலியவை கணிப்

பொறியின் நினைவகத்தில் எண்களாகப் பதிவு செய்யப்படு கின்றன. எண் வடிவிலுள்ள இந்த விவரங்கள் படித்துணரப்பட்டு அதற்கேற்ப மையச் செயலகம் செயற்படும்பொழுது, நமது கணக்குக்கேற்ற விடை கிடைக்கிறது.

இயக்க ஆளுகையில் [Process Control] செய்திக் கூறு களைச் சோதித்து அறிதலும் பலவிதக் கண்காணிப்புகளைச் செய்தலும் அதற்கேற்ப இடுவரல் செய்திகளின் திறனை மாற்றி யமைத்தலும் எண்ணிலக்கக் கணிப்பொறியில் செய்யப்படு கின்றன. அறிவியல் கணிப்பில் [Scientific Computation] குறைந்த அளவு இடுவரல், விடுவரல் செய்திகள் இருந்தபோதிலும், இதில் ஆயிரக்கணக்கான கணிக்கும் செயல்கள் செய்யப்படு கின்றன. கணிப்பொறியைத் தொழிலகங்களில் விவர ஆய்வனை [Data Processing] செய்யப் பயன்படுத்தும்பொழுது, பல்லாயிரக் கணக்கான பதிப்புகள் [Records] உள்ள செய்திகளைப் படித் தறிதலும் அவற்றை ஆய்வு செய்து விடையளித்தலும் நிகழ் கின்றன. எனவே மேற்குறிப்பிட்ட பணிகளைச் செய்வதற்கு எண்ணிலக்கக் கணிப்பொறிகள் [Digital Computers] பெரிதும் பயன்படுகின்றன.

2.5.3. கலப்பினக் கணிப்பொறி (HYBRID COMPUTER)

சில வகைப் பணிகளைக் கணிப்பொறி மூலம் செய்யும் பொழுது, செய்திக்கூறுகள் எண்களாக அமையலாம், அல்லது தொடர்ச்சியான அளவையாக [Analog Signal] அமையலாம். இந்நிலையில் எண்ணிலக்க அல்லது ஒப்புமைக் கணிப்பொறிகள் உதவுகின்றன. ஆனால், கலப்பினக் கணிப்பொறிகள், மேற்குறிப் பிட்ட இரண்டு முறைகளிலும் இடுவரலைப் பெற்றுச் செயல்படு கின்றன. தொழிற்கூடங்களில் இயக்க ஆளுகையில் [Process Control] கணிப்பொறி ஈடுபடுத்தப்படும்பொழுது குறிப்பிட்ட சில பணிகளில் இடுவரல் அளவையாக [Analog] இடப்படுகிறது. எடுத்துக்காட்டாக இரசாயனப் பொருட்கள் அல்லது மருந்து தயாரிக்கும் தொழிற்கூடங்கள் முதலியவற்றில் இம்முறையிலேயே கணிப்பொறிக்கு இடுவரல் செய்யப்படுகிறது. வெப்பம், காற்ற ழுத்தம், மூலப்பொருட்களின் சரிவிகிதப் பங்கு [Percentage of Ingredients] முதலியவற்றை அளவு கருவியில் [Meter] குறித்துத் தொடர்ச்சியாகக் கணிப்பொறிக்கு இடுவரல் செய்யப்படுகிறது.

அங்ஙனம் இடுவரல் செய்யப்பட்ட விவரங்கள் கணிக்கப்பட்டு, விடைகள் இறுதியில் அளவு கருவியில் [Meter] தொடர்ச்சியாகவோ, எண் பட்டியலாகவோ அச்சுக் கருவியின் மூலம் பெறப்படுகின்றன. இக்கணிப்பொறிகள் எண்ணிலக்கக் கணிப்பொறியைக் காட்டிலும் குறைந்த அளவே புழக்கத்தில் உள்ளன.

2.6. கணிப்பொறியின் இயக்கத்திறனில் ஓர் ஒப்புநோக்கு

மேலே விவரித்தபடி ஒவ்வொரு காலக்கட்டத்திலும் கணிப்பொறியின் செயல்திறன், அமைப்பு, இயக்கமுறை முதலிய பல பிரிவுகளிலும் ஒரு வரையறைக்குட்பட்டே அதன் வளர்ச்சி நிகழ்ந்து வந்துள்ளது. கடந்த 35 ஆண்டுக் காலத்தில் மிகவும் குறிப்பிடத்தக்க மறுமலர்ச்சி, கணிப்பொறியின் விரிவாக்கத்தில் [Developments] ஏற்பட்டுள்ளது. முதலாம் காலக்கட்டக் கணிப்பொறி குறைந்த செயல்திறன் பெற்றிருந்தது மட்டுமன்றி, பலநூறு, சதுர அடி பரப்புள்ள இடத்தை அடைக்கும் உருவ அமைப்பையும் பெற்றிருந்தது. ஆனால், தற்காலத்தில் மிகக் குறைந்த பொருள் செலவில் கிடைக்கக்கூடிய நான்காம் காலக் கட்டக் கணிப்பொறிகள் சிறிய உருவ அமைப்பைப் பெற்றுள்ளன. மேலும், இவை பலகோடி ஆணைகளை ஒரு நொடியில் ஏற்றுச் செயற்படும் மிகை வேகத்திறனும் [Fast Speed] வாய்க்கப் பெற்றுள்ளன. இவற்றிற்கிடையேயுள்ள வேறுபாடு மலைக்கும் மடுவுக்கும் உள்ள வேறுபாட்டை யொத்ததெனில் அது மிகையன்று. நவீன அறிவியல் யுகத்தில் திறன்பாடுமிக்க, பல கருவிகள் உருவாக்கப்படுகின்றன. மிகைவேகத் தொழில் வல்லமையால்[Technological breakthrough] சக்தி வாய்ந்த பல நுண்ணியக்கிகள் [Micro Processors] வடிவமைக்கப்பட்டுள்ளன. இவை மிகக் குறைந்த விலையில் எளிதில் கிடைக்கின்றன. இதன் காரணமாக உலகெங்கிலும் நூற்றுக்கணக்கான கணிப்பொறி நிறுவனங்கள், இந்த நுண்ணியக்கிகளைக்கொண்டு மிகச் சிறிய மற்றும் சிறிய [Micro and Mini Computers] கணிப்பொறிகளை உருவாக்குவதில் முனைந்துள்ளன, எனவே கணிப்பொறியானது, கல்வி, தொழில், விண்வெளி ஆய்வு, சமூக வாழ்க்கை, ஆராய்ச்சி ஆகிய பல துறைகளில் மிகவும் குறிப்பிடத்தக்க தாக்குதலையும் அதன் விளைவாகப் பயன்மிகு ஒரு திருப்பத்தையும் உண்டாக்கும் என்பதில் சிறிதும் ஐயமில்லை.

கணிப்பொறியின் அமைப்புமுறை, அது செயற்படும் விதம், அதன் பல்வேறு பாகங்கள் பற்றியும் பலதுறைகளில் கணிப்பொறியின் பயன்பாடு, வழியமைப்புமுறை, பேஸிக் மொழி முதலியவைபற்றியும் பின்வரும் பாகங்களில் விரிவாகக் காணலாம்.

பாகம் – 3

3. கணிப்பொறியின் அமைப்பும் செயற்பாடும்

நான்கு காலக்கட்டக் கணிப்பொறிகளிலும் ஒவ்வொரு காலக் கட்டத்திற்கும் ஏற்ப கணிப்பொறியின் அமைப்பு, செயலாற்றல், அதன்வேகம் முதலியவற்றில் வியத்தகு முன்னேற்றம் கண்டுள்ளமை பற்றி முன்பே விவரித்தோம். இதற்குக்காரணம் மின்னணுவியல்புரட்சி[Electronic Revolution] எனில் அஃது மிகையன்று., அளவில் மிகவும் பெரியதும்,குறைந்த செயலாற்றல் கொண்டதுமான எனியாக் கணிப்பொறியையும் தற்காலத்தில் உருவாக்கப்படும் மிகச்சிறிய அள்விலானதும் எனியாக் கணிப்பொறியைவிட பல்லாயிரம் மடங்கு செயல்திறன் மிக்கதுமான IBM-PCபோன்ற கணிப்பொறிகளையும் ஒப்பு நோக்கின் இத்துறையின் முன்னேற்றம் தெற்றெனப் புலப்படும்.

அவ்வக் காலக்கட்டங்களில் உலகம் முழுவதும் மின்னணு வியல் மற்றும் கணிப்பொறியின் வடிவாக்கம் முதலியவற்றில் நிகழ்ந்த பல உயராய்வுகளின் பலனாக மேன்மேலும் திறன்மிக்க பற்பல பயன்பாடுகளுக்கேற்றபல்லாயிரக்கணக்கானகணிப்பொறி கள் வடிவமைக்கப்பட்டு உருவாக்கப்படுகின்றன. இவ்வாறு உருவாக்கப்படும் கணிப்பொறிகள் உருவ அமைப்பு, நினைவக அளவு, செயற்பாட்டுத்திறன், விலை இவற்றினடிப்படையில் ஒன்றுக்கொன்று வேறுபட்டிருக்கின்றன. எனினும் அவை செயற்படத் தேவையான அமைப்பில் [Minimum Configuration] ஒன்றையொன்று ஒத்திருந்தன. கணிப்பொறியின் அமைப்பையும் அதன் பல உறுப்புகளையும் கீழ்க்கண்டவாறு ஆராய்ந்தறியலாம்.

3.1. கணிப்பொறியின் அமைப்பும் அங்கங்களும்

கணிப்பொறியின் வேகத்திறன், நினைவக அளவு, பயன்பாடு, விலை ஆகியவற்றின் அடிப்படையில் பொதுவாகக் கணிப்பொறிகளை நான்கு வகைகளாகப் பிரிக்கலாம்.
அவைகளாவன :

1. உயர் நுட்பநிலைக் கணிப்பொறி (Super Computer)
2. இடைநிலைக் கணிப்பொறி (Medium Range Computer)
3. சிறிய கணிப்பொறி (Mini Computer)
4. மிகச்சிறிய கணிப்பொறி (Micro Computer)

உயர்நுட்பநிலைக் கணிப்பொறிகள் [Super Computers] தமது பெயருக்கேற்ப மிக உயர்ந்த செயல்திறன் வாய்க்கப் பெற்றவை. அதிக அளவு நினைவகத்தையும் மிகை வேகத்திறன் கொண்ட புற ஒருங்குகளையும்[Peripherals] இக்கணிப்பொறிகள் பெற்றுள்ளன.

எடுத்துக்காட்டாகக் கூறின் Cray-1 என்ற உயர் நுட்ப நிலைக் கணிப்பொறி ஒரு வினாடிக்கு 1000 இலட்சம் ஆணைகளை ஏற்றுச் செயற்படும் மிகைவேக ஆற்றல் படைத்தது. ஐ.பி.எம். நிறுவனம் உற்பத்தி செய்யும் Star, Cyber-7600 ஆகிய உயர்நுட்பநிலைக் கணிப்பொறிகளை எடுத்துக்காட்டாகக் கூறலாம். இடைநிலைக் கணிப்பொறிகள் [Medium Range Computers], உயர் நுட்பநிலைக் கணிப்பொறிக்கு அடுத்த நிலையைச் சார்ந்தவையாகும். இந்த வகைக் கணிப்பொறிகள் வர்த்தக ரீதியில் அதிகளவு பயன்படுத்தப்படுகின்றன. இவற்றிற்கு மாத வாடகையாக ரூ. 100,000 வரை வசூலிக்கப்படுகிறது. இவற்றில் பல பயன்பாடுகளுக்குத் தேவையான வழியமைப்புகள் உள்ளன இடைநிலைக் கணிப்பொறிக்கு எடுத்துக்காட்டாக IBM-4341 மற்றும் S/370 முதலிய கணிப்பொறிகளைக் கூறலாம். விலை, நினைவக அளவு, செயல்திறன் ஆகியவற்றின் அடிப்படையில் சிறிய கணிப்பொறிகளும் [Mini Computer] மிகச் சிறிய கணிப்பொறிகளும்[Micro Computer] வகைப்படுத்தப்படுகின்றன. வங்கி, வாணிபம், கல்வி, போக்குவரத்து முதலான துறைகளில்

பயன்படுத்தப்படும் கணிப்பொறிகள் சில குறிப்பிட்ட பணிகளைத் திறம்படச் செய்யும் திறன் பெற்றிருத்தல் வேண்டும். சிறிய கணிப்பொறிக்கு PDP-11 ஐ எடுத்துக்காட்டாகக் கூறலாம்.

பல்வேறு துறைகளில் கணிப்பொறியைக்கொண்டு கணிப்புகள் செய்யவும்,விவர ஆய்வனை [Data Processing] செய்வதற்கும் கணிப்பொறி பயன்படுத்தப்படும்பொழுது, அதற்கென சில குறிப்பிட்ட வசதிகள் வாய்க்கப் பெற்றிருத்தல் அவசியமாகும். கணிப்பொறியின் வகைகளான உயர்நுட்பநிலைக் கணிப்பொறி, இடைநிலைக் கணிப்பொறி, சிறிய மற்றும் மிகச்சிறிய கணிப்பொறிகளுக்கேற்ப அதனுடைய கருவிய அமைப்பு அமைந்திருக்கும். எனினும் சில இன்றியமையா அங்கங்கள் மேற்கண்ட எல்லா வகைக் கணிப்பொறிகளிலும் அமைந்துள்ளன. கணிப்பொறியின் பல்வேறு அங்கங்கள், அவற்றின் செயற்பாடு முதலியவற்றைக் கீழ்க்கண்டவாறு ஆராய்ந்தறியலாம்.

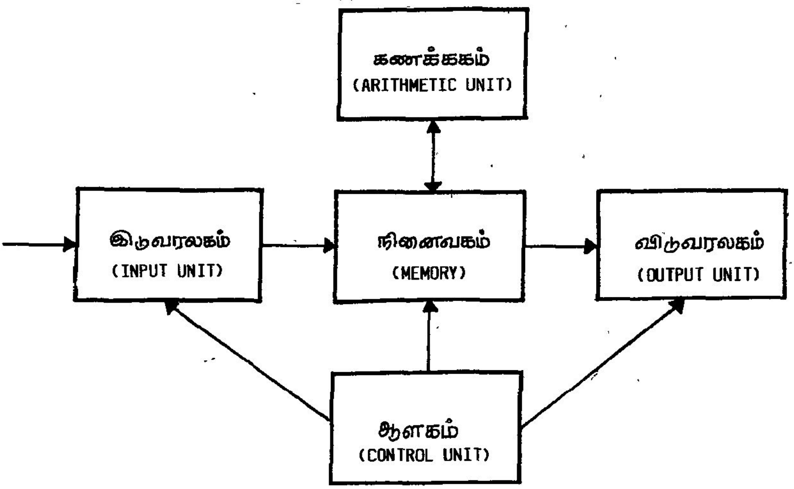
3.1.1. கணிப்பொறியின் அங்கங்கள்

கணிப்பொறியின் அங்கங்களைக் கீழ்க்கண்டவாறு பிரிக்கலாம்.

1. இடுவரலகம் [INPUT UNIT]
2. விடுவரலகம் [OUTPUT UNIT]
3. நினைவகம் [MEMORY]
4. கணக்ககம் (ARITHMETIC UNIT)
5. ஆளகம் [CONTROL UNIT]

கணிப்பொறியைக் கொண்டு ஒரு பணியைச் செய்வதற்குச் செய்திக் கூறுகளும் வழியமைப்புகளும் இடுவரல்கம் மூலமாகக் கணிப்பொறிக்கு இடுவரல்செய்யப்படுகின்றனஎடுத்துக்காட்டாகக் கூறின், துளையிடப்பட்ட 80 குத்துவரிசைகளை உடைய அட்டை [Punched Cards], துளைத்தாள் நாடா, தொலை அச்சுத்தட்டுப் பலகை [Tele Type Key Board], மின்காந்த நாடா [Magnetic Tape]ஆகியன இடுவரல் ஒருங்குகளாகும் [Input system], அச்சுப் பொறி [Printer], காட்சித் திரை [Display Screen], மின்காந்த நாடா [Magnetic Tape], மின்காந்தத்தட்டு [Magnetic Disk]

ஆகியன விடுவரல் ஒருங்குகளுக்கு [Output System] எடுத்துக் காட்டுகளாகும். படம் எண் 10-இல் கட்டப்படம் மூலம் கணிப் பொறியின் அமைப்பு விளக்கப்பட்டுள்ளது. கணிப்பொறியின் பல அங்கங்கள் பற்றி விரிவான விளக்கத்தைக் கீழ்க்கண்டவாறு காணலாம்.



படம்—10

கணிப்பொறியின் அமைப்பு

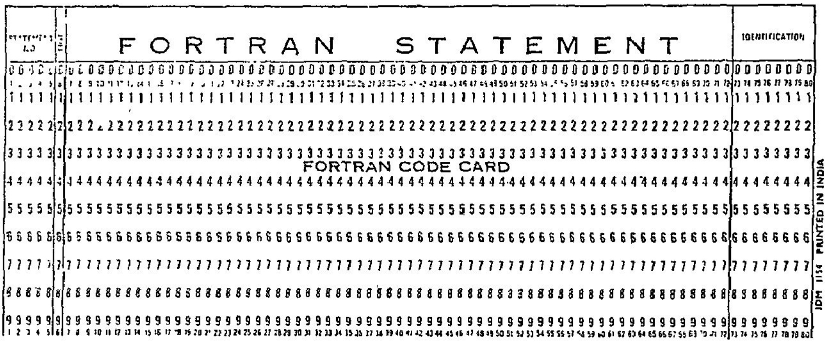
3.2. இடுவரலகம் (*INPUT UNIT*)

குறிப்பிட்ட ஒரு பணியைக் கணிப்பொறி மூலம் செய்வதற்குத் தேவையான செய்திக்கூறுகளையும் [Data] அதற்கான வழியமைப்புகளையும் அதனைச் செயற்படுத்தத் தேவையான கட்டளைகளையும் [Commands] கணிப்பொறிக்கு அறியத் தருதல் வேண்டும். எவ்வாறு செய்திக் கூறுகள் கணிப்பொறிக்கு ஊட்டப்படுகின்றன? என்பவற்றைப் பொறுத்து, அவற்றுக்கான இடுவரல் கருவிகள் உருவாக்கப்பட்டுள்ளன. எடுத்துக்காட்டாகக் கூறின், துளையிடப்பட்ட அட்டை, துளைத்தாள் நாடா [Punched Paper Tape] ஆகியவற்றின் மூலம் செய்திக்கூறுகள் அட்டைத் துளைப்பொறியின் உதவிகொண்டு துளைவடிவில்

பதிவு செய்யப்பட்டு இடுவரல் செய்யப்படுகின்றன. அட்டைப் படிப்பொறி [Card Reader], துளைத்தாள் நாடாப் படிப்பொறி [Punched Paper Tape Reader] ஆகிய கருவிகள் முன்பே பதிவு செய்யப்பட்ட விவரங்களைக் கணிப்பொறிக்குப் படித்து உணர்த்துகின்றன. அட்டைகளின் மூலம் கணிப்பொறி செய்திக் கூறுகளை அறிந்துணரும் வண்ணம் எங்ஙனம் துளையிடுவது என்பது பற்றிச் சிறிது ஆராய்வோம்.

அண்மைக் காலம் வரை கணிப்பொறிக்குச் செய்திகளை இடுவரல் செய்யச் சிறந்ததொரு சாதனமாகத் துளையிடப்பட்ட அட்டை [Punched Cards] பரவலாகப் பயன்படுத்தப்பட்டு வந்தது; பயன்படுத்தப்பட்டும் வருகிறது. ஹெர்மன் ஹாலரித் [Herman Hollerith] என்ற அறிஞரால் அமெரிக்க நாட்டின் மக்கள்தொகைக் கணக்கெடுப்பிற்குச் செய்திதிரட்டும் கருவியாகப் பயன்படுத்தபட்ட துளையிடப்பட்ட அட்டை, கணிப்பொறியின் இடுவரல் சாதனமாகப் பெரும்பயன் அளிக்கிறது. செவ்வக வடிவமுள்ள இவ்வட்டை 18•7 செ. மீ. நீளமும் 8.2 செ. மீ. அகலமும் உடையது. இந்த அட்டையில் நீளவாக்கில் 12 நீள வரிசைகளும் [Rows] அகலவாக்கில் 80 நேர் குத்துவரிசைகளும் [Columns] உள்ளன. இந்த அட்டையின் மூலைகள் [Corners] கூர்மையாகவோ, வளைவாகவோ இருக்கும். அட்டையின் ஒரு மூலைப்பகுதி [Corner] வெட்டப்பட்டிருக்கும். இதன் உதவியைக் கொண்டு விவர ஆய்வனையில் பயன்படுத்தும்பொழுது அட்டைகளைவரிசையாக அடுக்குவதற்கும் தவறிய அட்டைகளைக் கண்டறிவதற்கும் முடிகின்றன. அட்டையின் மேல்பரப்பில் நிறுவனம் பற்றிய பெயர் குறிப்பிடப்பட்டிருக்கும். இல்லாவிடில் அதனுடைய குத்துவரிசைகளிலும் நீளவரிசைகளிலும் 0-விலிருந்து 9—வரையிலுள்ள எண்களை அடையாளம் காண்பதற்கேற்ற வகையில் எண்கள் குறிக்கப்பட்டிருக்கும். அட்டையின் நீள வரிசைகளுக்கு (0,1,2,3,.... 9) என்று எண்கள் கொடுக்கப் பட்டிருக்கும். அட்டையின் கடைசி வரிசையில் 9 என்ற எண் குறிக்கப்பட்டுள்ளது. 0— வரிசைக்கு மேல் வரிசை 11—ஆம் வரிசை என்றும், அதற்கும் மேல் உள்ள வரிசை 12—ஆம் வரிசை என்றும் குறிப்பிடப்பட்டிருக்கும். அட்டையின் மேல் வரிசைப் பகுதி துளையிடப்படும் செய்திகளை அறிந்துணர்வதற்காக

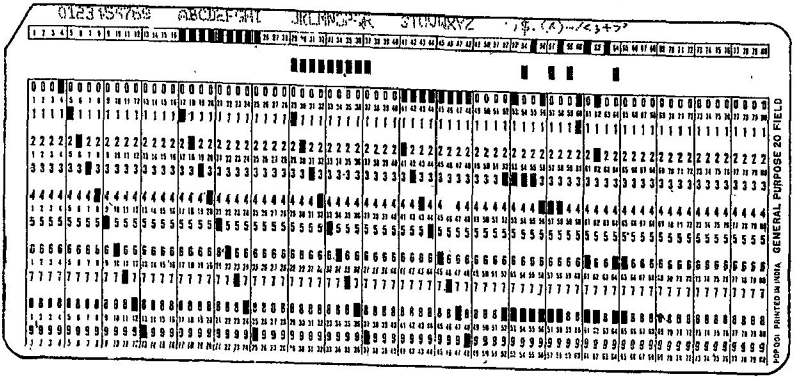
அச்சிடப் பயன்படுகிறது. படம் எண் 11—இல் துளை செய்யப் படாத அட்டையின் அமைப்பு காட்டப்பட்டுள்ளது.



படம்—11

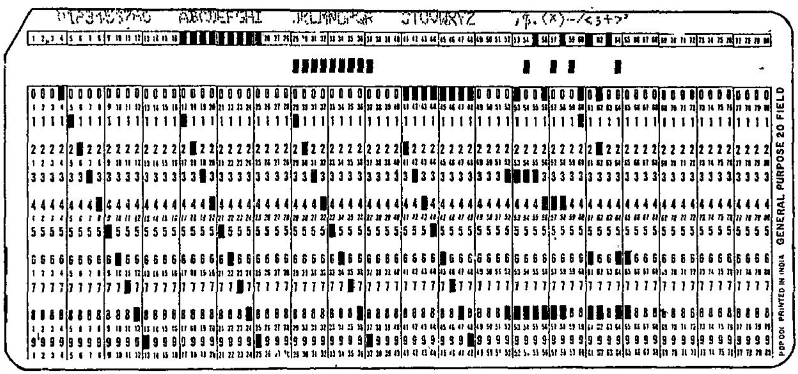
துளையிடப்படா அட்டை (Unpunched Card)

அட்டையில் விவரங்கள் துளைசெய்யப்படும் பொழுது எண்களைக் குறிப்பிடுவதற்கு ஒரு துளையும், எழுத்துகளைக் குறிப்பிடுவதற்கு ஒரே துளை நிலையில் இரண்டு அல்லது மூன்று தடவையும் துளைகள் செய்யப்படும். படம் எண் 12—இல் [A, B, C..... Z] மற்றும் (0,1,2,3,.....9) முதலியவற்றைத் துளை செய்யப்பட்ட நிலையில் அட்டை காட்டப்பட்டுள்ளது. இவை தவிர, சில தனிக் குறிகளையும் [Special Symbols] கணிப்பொறி அட்டையில் துளை செய்ய முடியும். இந்தத் தனிக்குறிகள் யாவும் அட்டையின் ஒரே துளை நிலையில் ஒன்றுக்கும் மேற்பட்ட தடவைகள் துளை செய்யப்பட்டு, பதிவு செய்யப்படுகின்றன. = , $. (*) — / < ; + > ' ஆகியன தனிக்குறிகள் ஆகும். தனிக்குறிகள் துளையிடப்பட்ட நிலையில் உள்ள அட்டை, படம் 13—இல் காட்டப்பட்டுள்ளன.



படம்—12

துளை செய்யப்பட்ட அட்டை (Punched Card)



படம்–13

தனிக்குறிகள் துளையிடப்பட்ட அட்டை

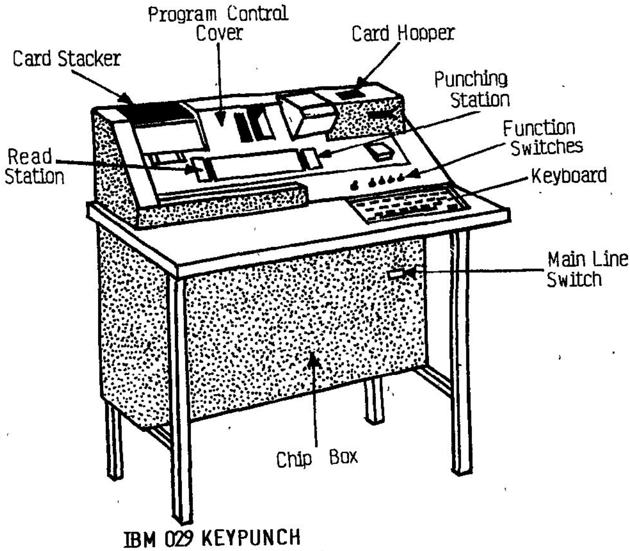
3.2.1. அட்டைத் துளைப்பொறி (CARD PUNCHING MACHINE)

விவர ஆய்வணையில் கணிப்பொறியின் பங்கு மிகவும் பெரியது. தொழிலகங்களில் பல்வேறு துறைகளில் பொருள் நிலையாளுகை [Inventory Control], சம்பளப்பட்டியல் தயாரித்தல் [Payroll Processing], விற்பனையியல் [Sales Management] முதலியவற்றின் செய்திகளைச் சேகரித்து, குறிப்பிட்ட தேவைக்கேற்ற படிவத்தில் இதற்கென உள்ள அட்டைகளில் துளை செய்யப்படும். இவ்வாறு துளைகள் மூலம் பதிவு செய்யப்பட்ட செய்திகளை, இடுவரல் கருவியான அட்டைப் படிப்பொறி [Card Reader] படித்தறிந்து கணிப் பொறியின் துணை நினைவகத்தில் [Auxiliary Storage] பதிவு செய்கிறது.

இவ்வாறு செய்திகளைத் துளையிட்டு அட்டைக்கொத்து களின் [Card Deck] மூலம் பதிவு செய்யும் கருவிக்கு அட்டைத் துளைப்பொறி [Card Punching Machine] என்று பெயர். அட்டைத் துளைப்பொறி எங்ஙனம் வேலை செய்கிறது என்பதைச் சுருக்கமாகக் கீழ்க்கண்டவாறு ஆராயலாம்.

கணிப்பொறியில் விவரங்களையும் வழியமைப்புகளையும் இடுவரல் செய்யப் பயன்படுத்தப்படும் முறைப்படிவ [Standard Format] அட்டையானது, 12 நீள் வரிசைகளையும் [Rows], 80 குத்து வரிசைகளையும் [Columns] பெற்றுள்ளது. இதுபற்றி விரிவாக முன்பே கூறப்பட்டுள்ளது. அட்டைத் துளைப்பொறியின் அமைப்பு, படம் எண் 14—இல் காட்டப்பட்டிருக்கிறது. இந்தப் பொறியில் ஹோப்பர் [Hopper] என்ற அட்டைகளை அடுக்கும் பகுதியும் தட்டச்சுப் பொறியிலுள்ளது போன்ற பித்தான்கள் கொண்ட தட்டுப் பலகையும் [Key Board] துளையிடப்பட்ட அட்டைகளை முறைப்படி அடுக்கும் பகுதியும் அட்டையில் குத்து வரிசை எண்ணை அறிவிக்கும் பகுதியும் உள்ளன. அட்டைத் துளைப் பொறியை இயக்குபவர் [Card Punch Operator] துளை யிடப்படா அட்டைக் கொத்தை [Card Deck] ஹோப்பர் பகுதியில் அடுக்க வேண்டும். பின் FEED என்ற பித்தானை அழுத்த ஓர் அட்டை தானாகக் கீழே தள்ளப்படும். பின்னர் REG பித்தானை அழுத்தும்பொழுது, அட்டை சரியான நிலையில் விவரங்களைத் துளை செய்வதற்கு ஏற்றவாறு வைக்கப்படுகிறது.

இயக்குபவர் ஒவ்வொரு பித்தானையும் குறிப்பிட்ட விவரத்திற்குத் தகுந்தபடி அழுத்தும்பொழுது, அட்டை அதற்கேற்ற முறையில் தகுந்த குத்து வரிசை, நீள் வரிசையில் துளையிடப்பட்டு ஒவ்வொரு குறிநிலையாக [Character Position] நகர்ந்து செல்லும். பின் REL என்ற பித்தானை அழுத்தும் பொழுது, அட்டை உள்ள நிலையிலிருந்து விடுபட்டு அட்டை அடுக்கும் பகுதிக்கு வந்து சேரும். இதில் பல விசைகள் [Switches] உள்ளன. அவற்றைப் பயன்படுத்தி அட்டையைத் தானாகத் [Automatic] துளை செய்யும் மேடைக்குக் கொண்டு வந்து, துளை செய்யவும் மறுபடி அடுக்கவும் இயலும். ஒன்றன்பின் ஒன்றாக அட்டைகள் துளை செய்யப்பட்டு வரிசை முறையில் அடுக்கப் பெறுகின்றன. அட்டையின் 12-ஆவது நீள் வரிசையில் விவரங்கள் நாம் துளை செய்தபடி அச்சிடப்பட்டிருக்கும்.



படம்—14

அட்டைத் துளைப் பொறி [CARD PUNCHING MACHINE]

நாம் துளை செய்யும் பொழுது குத்துவரிசையில் நிலை எண்ணைக் [Column Number] காட்டும் உருளையைப் பார்த்து, அட்டையின் எந்தப் பகுதி துளை செய்யப்படுகிறது என்பதை

அறியலாம். இவ்வாறு ஏராளமான விவரங்களும் வழியமைப்பு ஆணைகளும் [Program Instructions] இந்தப் பொறிமூலம் அட்டையில் துளை செய்யப்படுகின்றன. துளை செய்யும்பொழுது ஏதாவது தவறு நேரின், அந்த அட்டையை ஒதுக்கிவிட்டு வேறு அட்டையில் மீண்டும் அதே விவரத்தைத் துளை செய்யவேண்டும்.

உதாரணமாக, சம்பளப்பட்டியல் தயாரித்தலில் [Payroll Preparation] ஒவ்வொரு வகையிலும் தொழிலாளியின் வருமான அளவு சரியாகத் துளை செய்யப்பட்டிருக்கிறதா என்று சரி பார்ப்பதற்கு, சரிபார்ப்பி [Verifier] பயன்படுத்தப்படுகிறது. மேலும், இது அகர முதலி வரிசையில் தொழிலாளர்களின் பெயர் கொண்ட அட்டையை அடுக்கவும் பதவி அல்லது துறை வாரியாகப் பிரிக்கவும் வரிசைப்படுத்தவும் பயன்படுத்தப்படு கிறது. சுமார் 500,000—க்கும் மேலான அட்டைத் துளைப் பொறிகள் உலகெங்கிலும் புழக்கத்தில் உள்ளதாகக் கருதப்படு கிறது. உதாரணமாக ஐ.பி.எம்.நிறுவனத்தாரின் IBM-029, IBM-129 ஆகியவற்றைக் கூறலாம். இவற்றின் செயலாக்க முறையில், சில குறிப்பிடத்தக்க வேறுபாடுகள் உள்ளன.

கணிப்பொறியின் விரிவாக்கத்தில் மேற்கொள்ளப்பட்ட ஆராய்ச்சிகளின் பயனாக இடுவரல் ஒருங்குகளின் வேகமும் செயல்திறனும் பன்மடங்கு அதிகரிக்கப்பட்டுள்ளன. மேலும் செய்திகளைப் பதிவுசெய்தலிலும் அதனை ஒழுங்குபடுத்துதலிலும் [Data Storage and Organization] பெரிதும் முன்னேற்றம் ஏற்பட்டுள்ளது. அட்டைகளில் விவரங்களைத் துளை செய் வதற்குப் பதிலாக மின்காந்தத் தட்டு, மின்காந்த நாடா, மென்வளைத் தட்டு [Floppy Disk] ஆகியவற்றில் விவரங்களைப் பதிவு செய்யலாம். காட்சிப் பொறி [Display Unit], விவரம் பதியும் கருவி [Data Entry Terminal] முதலிய பல கருவிகள் செய்திக் கூறுகளைப் பதிவு செய்வதற்கெனத் தோன்றியுள்ளன. இவைதட்டச்சுப்பொறியைப்போன்றேதட்டுப்பலகையையும்[Key Board], தொலைக்காட்சிப் பெட்டியைப் போன்ற திரையையும் பெற்றிருக்கின்றன. இதன் மூலமாகச் செய்திகளைப் பதிவு செய்யும் பொழுது ஏற்படும் தவறுகளை உடனுக்குடன் சரிசெய்ய முடிகிறது. மேலும், அட்டையில் செய்திகளைத் துளைசெய்து பதிவிடும் முறையைவிட இது விரைவானதும் குறைந்த பொருட் செலவே ஆகக்கூடியதுமாகும். ஆனால், அட்டையில் பதிவு செய்யப்படும் செய்திக் கூறுகளும் ஒரு கணக்கைச் செய்யத்

தேவையான ஆணைகளும் [Instructions] அட்டைப் படிப்பொறி மூலம் [Card Reader] கணிப்பொறிக்கு இடுவரல் செய்யப்படுகின்றன.

3.2.2. அட்டைப் படிப்பொறி (CARD READER)

அட்டைத் துளைப் பொறியைப் பற்றியும் அஃது எங்ஙனம் செய்திக் கூறுகளையும் வழியமைப்புகளையும் துளை செய்கிறது என்பதைப் பற்றியும் முன்பு பார்த்தோம். இதில் பயன்படுத்தப் படும் அட்டை, முற்றிலும் மின்கடத்தாப்பொருளால் [Insulator] உருவாக்கப்பட்டது. இதில் துளை செய்யப்பட்டுள்ள செய்திகளைக் கணிப்பொறி மூலம் ஆய்வு செய்வதற்கு, அதனுடைய நினைவகத்திற்குச் செய்திகளைக் கொண்டு வருதல் அவசியம். 80 குத்து வரிசைகளையும் 12 நீள் வரிசைகளையும் உடைய இந்த அட்டையில் 960 குறி நிலைகளில் செய்திகளைத் துளை செய்ய முடியும். ஒவ்வொரு குறியும் துளை அல்லது துளையற்ற நிலையைக் கொண்டு குறிப்பிடப்படுகிறது. அட்டைகளில் உள்ள செய்திகள் அட்டைப் படிப்பொறி மூலம் படித்துணரப்பட்டு, கணிப்பொறியின் நினைவகத்தில் பதிவு செய்யப்படுகின்றன. எடுத்துக்காட்டாக ஓர் உலோகத் தகட்டின் மீது துளைத்தாள் அட்டை வைக்கப்பட்டு, ஒரு துடைப்பானால் படித்தறிவதாக வைத்துக் கொள்வோம். அட்டையில் துளையுள்ள இடங்களில் துடைப்பானுக்கும் உலோகத் தகட்டிற்கும் இடையில் மின் தொடர்பு ஏற்படுகிறது. மற்ற இடங்களில் மின் தொடர்பு ஏற்படுவதில்லை. இதுவே அட்டைப் படிப்பொறி மூலம் செய்திகளைப் படித்தறியும் முறையின் அடிப்படைத் தத்துவமாகும்.

அட்டைப் படிப்பொறியைக்கொண்டு சுமாராக நிமிடத்திற்கு 600— இலிருந்து 1200 அட்டைகள் வரை படித்தறிய முடியும். அட்டைப் படிப்பொறியைக் கீழ்க்கண்டவாறு விவரிக்கலாம். இடுவரலகத்தில் அட்டைகளை அடுக்கும் ஹோப்பர் [Hopper] என்ற பகுதியும் அட்டைகளைப் படித்தறியும் பகுதியான படிப்பகத்தில் உலோகத்தால் ஆகிய ஓர் உருளையும் மின் தொடர்ச்சியை உண்டாக்கும் இணைப்பிகளும் உள்ளன. மேற் குறிப்பிடப்பட்ட முறையில் செய்திகளைப் படித்தறிந்த பின்னர் அட்டைகளைச் சேகரிக்கும் நிலையமும் உள்ளது. அட்டைப் படிப்பொறிகளை, அட்டையைப் படித்தறியும் முறையைப் பொறுத்து இரண்டு பிரிவுகளாகப் பிரிக்கலாம்.

3.2.2.1. துடைப்பான் முறை அட்டைப் படிப்பொறி (BRUSH TYPE CARD READER)

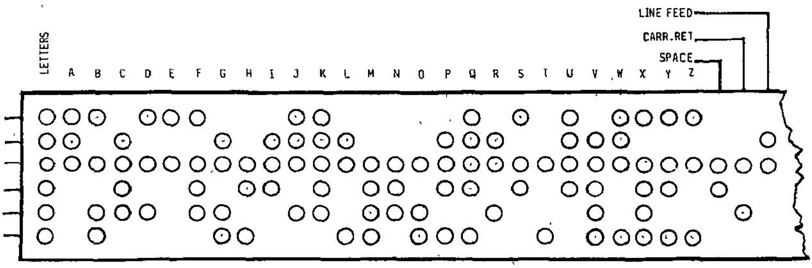
அட்டையானது துளையிடப்படா நிலையில் ஒரு சிறந்த மின் கடத்தாப் பொருளாக [Insulator] விளங்குகிறது என்பதன் அடிப்படையில் இக்கருவி செயற்படுகிறது. முன்பு குறிப்பிடப் பட்டது போல் அட்டைகளையடுக்கும் பகுதி, அட்டைகளைப் படித்தறியும் பகுதி, படித்தறியப்பட்ட அட்டைகளை அடுக்கும் பகுதி ஆகியவை இதில் அமைந்துள்ளன. இதில் உலோகத்தால் ஆகிய ஓர் உருளை, குறிப்பிட்ட நிலையான வேகத்தில் சுழன்று கொண்டிருக்கிறது. இதன் மேற்பகுதியில் நூற்றுக்கணக்கான, மெல்லிய உலோகத்தால் ஆகிய துடைப்பான்கள் [Brushes] பொருத்தப்பட்டுள்ளன. இடுவரல் செய்யப்படும் துளையிடப் பட்ட அட்டைக் கொத்திலிருந்து [Card Deck] அட்டைகள் ஒவ்வொன்றாக உள்ளனுப்பப்படுகின்றன. பின்னர் இந்த அட்டைகள் உருளையின் மீது சுற்றப்படுகின்றன. அதாவது அட்டை, உருளைக்கும் துடைப்பான்களுக்கும் இடையில் சுழல் கிறது. எனவே துளைகள் வரும் இடங்களில் மட்டும் துடைப்பான் கள் உருளையுடன் தொடர்புகொண்டு, மின் தொடர்ச்சியை உண்டாக்குகின்றன. அவ்வாறு மின்தொடர்ச்சி ஏற்படும் பொழுது, அட்டையில் துளையுள்ள இடங்களையறிந்து அதற் கேற்ப என்ன எழுத்து என்று இக்கருவி கணிப்பொறிக்கு அறிவிக் கிறது. இம்முறை வேகக் குறைவானது. இம்முறையில் ஒரு நிமிடத்திற்குச் சுமார் 600 அட்டைகளை மட்டுமே படித்தறிய முடியும்.

3.2.3. துளைத்தாள் நாடா (PUNCHED PAPER TAPE)

முதலாம் காலக்கட்டத்தில் (1925—1945) உருவாக்கப் பட்ட கணிப்பொறிகள் யாவும் பெரும்பாலும் ஏராளமான சங்கிலிகள், சட்டங்கள், விசைகளைக் [Switches] கொண்டிருந் தன. அவைகளைப் பயன்படுத்தும்பொழுது எண்ணிக்கைகளில் அதிகமான விசைகளைத்தகுந்தமுறையில் மாற்றி இயக்கிக் கணக் கிடுவதற்காக, கணிப்பொறிக்கு ஆணைகளை இடவேண்டி யிருந்தது. ஏனெனில், அப்போதைய கணிப்பொறிகள் யாவும், பெரும்பாலும் மின்எந்திர [Electro Mechanical] அமைப்பையே பெற்றிருந்தன. அதன் பின் உருவாக்கப்பட்ட கணிப்பொறிகள்,

சிறியஅளவிலான நினைவகமுடையனவாகவும், ஓரளவுமின்னணு முறையில் [Electronically] இயங்கக்கூடியனவாகவும் இருந்தன. இவை யாவும் வழியமைப்புகளையும் ஆணைகளையும் பல குறிகளாகத் துளையிடப்பட்ட நிலையில் துளைத்தாள் நாடா மூலம் இடுவரலாகப்பெற்று இயங்கின. எடுத்துக்காட்டாகக் கூறின், 1944—ஆம் ஆண்டு வடிவமைத்து உருவாக்கப்பட்ட மார்க்—1 என்ற கணிப்பொறி, இவை போன்ற துளைத்தாள் நாடா மூலம் ஆணைகளை ஏற்று இயங்கும் தன்மை வாய்ந்ததாக இருந்தது. இந்தக் கணிப்பொறியில் துளைத்தாள நாடா, நிமிடத்திற்கு 200 தடவைகள் தத்தித்தத்தி விவரங்களை இடுவரல் செய்தது.

கணிப்பொறியில் ஆணைகளை இடுவரல் செய்யத் துளை செய்யப்பட்ட நாடாவைப் பயன்படுத்தியதற்கு வேறொரு காரணமும் உண்டு. ஏனெனில், இந்த நாடாவைக் கணிப்பொறியில் பயன்படுத்துவதற்கு முன்பே, இது தொலை தொடர்பு, தந்தித்துறை ஆகியவற்றில் செய்திகளைத் துளை செய்து அனுப்புவதற்குப் பயன்படுத்தப்பட்டது. இது பற்றிய நல்ல பலனறிவு அறிஞர்களிடையே அப்பொழுது நிலவியிருந்தது. மேலும் அன்று முதல் இன்றுவரை, துளைத்தாள் நாடா செலவு குறைந்த ஒரு பதிவுச்சாதனமாகப் பயன்பட்டு வருகிறது. இந்தத் துளைத்தாள்களில் பலவிதங்கள் இருக்கின்றன. பிளாஸ்டிக்காலும் எண்ணெய் மூலம் பூசப்பட்ட தாளாலும் செய்யப்பட்ட நாடாக்களும் பயன்படுத்தப்படுகின்றன. இதன் அகலம் அரை அங்குலத்தில் இருந்து, மூன்று அங்குலம் வரை இருக்கும்.



படம்—15 துளைத்தாள் நாடா [PUNCHED PAPER TAPE]

படம் எண் 15–இல் குறிப்பிட்ட துளைக் கோவைகள் சேர்ந்த தொகுதியாக, செய்தி துளை செய்யப்பட்ட நிலையில் நாடா விவரிக்கப்பட்டுள்ளது. விவரங்களின் குறிகள், துளைக கோவையாக இந்த நாடாவில பதிக்கப்படுகின்றன. அதன் நடுவே செல்லும் சிறு துளைகளோடு கூடிய தொடர்ச்சியான தாரை, நாடாவை அதற்குரிய சாதனத்தில் இருத்தத் துணைபுரிகிறது. செய்தியானது இந்த நாடாவில் ஒவ்வொரு முறையும ஒரு வரியாகத் துளை செய்யப்படுகிறது நீளவாக்கில் ஐந்து அல்லது ஏழு [5 or 7 Tracks] தாரைகள் செல்லுகின்றன ஐந்து தாரை நாடாவில் எண், எழுத்து, தனிகுறிகளைக் குறிகக, துளைச் சேர்க்கை முறை பயன்படுகிறது. அஃதாவது குறிப்பிட்ட பகுதியில் எல்லாத்தாரைகளிலும் துளையிடப்பட்டிருக்குமெனில் அதன் பின் உள்ள துளைகளின் பதிப, எழுத்து எனக்கொள்ளவேண்டும்

இந்த நாடாவைத் துளை செய்யும் கருவி, நமது அலுவலகங்களில் பயன்படுத்தப்படும் தட்டச்சுப் பொறி போன்றதாகும் மின்தட்டச்சுப் பொறியில்கூட, சில மாற்றங்கள் செய்து துளை செய்யும் பகுதியை இணைப்பதன் மூலம், இந்தத் துளைத்தாள் நாடாவில் ஆணைகளைத் துளையிட முடியும். ஒவவொரு குறிக்கும் ஏற்ற சின்னத்தை, நாம் பித்தானை அழுத்தித் துளைசெய்யும்பொழுது, தாள் நாடாவானது ஒவ்வொரு முறையும் அடுத்த வரிக்கு நகர்ந்து செல்கிறது. சில சமயங்களில் நாம் துளை செய்யும் விவரங்களைத் தனித்தாளில் அச்சிட்டு அளிக்கும். விவரங்கள் சரியாகத் துளை செய்யபபட்டு உள்ளனவா எனச் சோதித்தறிவதற்கு இதைபபயன்படுத்தலாம்.

இவ்வாறு துளை செய்யப்பட்ட நாடா, துளைத்தாள் படிப்பொறி [Paper Tape Reader] மூலம் படித்து, கணிப் பொறிக்கு இடுவரல் செய்யப்படுகிறது. இது எங்ஙனம் நடைபெறுகிறது என்பதைக் கீழ்க்கண்டவாறு காணலாம்.

3.2.4. காகித நாடாப் படிப்பொறி (PAPER TAPE READER)

காகித நாடாப் படிப்பொறியில் நாடாவில் உள்ள குறிகளைப் படிக்கத் தனிப்பட்ட பொறி உள்ளது. துளைகளின் வழியாக ஒளியைச் செலுத்தி, புகை மின் [Photo-Electric] முறையைப்

பயன்படுத்தி, குறிகளைப் படிக்கிறது இப்பொறி. ஒரே சமயத்தில் நாடாப் படிப்பொறியில்குறிப்பிட்ட கூட்டுச்செய்திக் கூறுகளைத் தான் [Block Data] செலுத்த முடியும். ஒவ்வொரு அட்டையின் மூலம் ஒரே சமயத்தில் குறிப்பிட்ட கூட்டுச் செய்திக் கூறுகளை அனுப்ப முடிகிறது. ஆனால், காகிதநாடாவில் தொடர்ச்சியாகக் குறிகள் துளைக்கப்படுவதால் குறிப்பிட்ட அளவு குறிகளைப் பிரித்து, எப்படி உள் அனுப்புவது என்பதுதான் இடையூறாக உள்ளது. இந்த விதத்தில்தான் அட்டையும் காகித நாடாவும் வேறுபடுகின்றன. ஆனால், காகிதநாடாவில் குறிக்கோவையைப் [String] பிரிக்க, தனித்தன்மை கொண்ட துளைகள் போடப் படுகின்றன.

துளையிடப்பட்ட அட்டைக்கும் துளையிடப்பட்ட காகித நாடாவுக்கும் உள்ள வேறுபாடுகள் .

1 படிக்கும் வேகம் இரண்டுக்குமே ஒன்றுதான்.

2. காகித நாடாவினால் செலவு குறைவு.

3. அட்டைகளை எப்படி வேண்டுமானாலும் மாற்றிக் கொள்ள முடியும். துளைகளிடுவதில் தவறு நேர்ந்தால் உடனே வேறு அட்டையைப் பயன்படுத்தலாம்.ஆனால் நாடாவை அப்படி மாற்ற இயலாது.

4. அட்டைகளில், சொல்நீளத்தை [Word Length] நினைத்த வாறு மாற்ற இயலாது. நாடாவில் மாற்ற முடியும்.

5. அட்டைகளில் சில இடமாற்றம் செய்யப்பட்டாலோ, காணாமல் போனாலோ தொடர்ச்சி இருக்காது நாடா வில் தொடர்ச்சி மாற வாய்ப்பில்லை.

3.2.5. காந்தமைக் குறியீடு முறை (*MAGNETIC INK CHARACTER RECOGNITION TECHNIQUE*)

இம்முறையில் துளையிடப்பட்ட அட்டைகளோ நாடாக் களோ நீக்கப்படுகின்றன. இம்முறையில் துளையிடப்பட

வேண்டிய அவசியம் இல்லை. காகிதத்தில் எழுதிய செய்திகளைக் கணிப்பொறி நேராகப் படித்தறியும். ஆனால் அவ்வாறு எழுதப்படும் முறையில், குறிகள் தனித்தன்மை கொண்டனவாக இருக்கவேண்டும். இம் முறையில் குறிகளை அளவு மாறாமல் எழுதுதல், எழுதும்போது மையின் அடர்த்தி சீர்மையாக இருத்தல் முதலியவை மிகவும் அவசியமாகும். மற்றொரு முறையான அமெரிக்கன் முறையில், மையின் மின்காந்தத் தன்மையைக் கொண்டு படிமுனை [Read Head] குறியின் மேலாகச் செல்லுகையில் காந்தம் மின்னோட்டத்தை ஏற்படுத்துகிறது. இவ்வாறு உண்டாகும் மின்னோட்டத்தின் தன்மையைப் பொறுத்து, குறிகளை அறிந்துணர முடிகிறது. மேற்குறிப்பிட்ட வகை இடுவரல்கள் அல்லாமல் கணிப்பொறிக்குச் செய்திக்கூறுகளையும் ஆணைகளையும் இடுவரல் செய்ய, தொலை மின் தட்டச்செழுத்துப் பொறி [Tele Typewriter], காட்சித்திரை [Display Screen] முதலியவைகள் தற்காலத்தில் பெருமளவு பயன்படுத்தப்படுகின்றன.

3.3. விடுவரலகம் (OUTPUT UNIT)

கணிப்பொறியில் ஆணைகளையும் விவரங்களையும் வழியமைப்புகளையும் இடுவரல் செய்வதற்குப் பலதரப்பட்ட இடுவரலகங்கள் உள்ளன. அங்ஙனமே கணிப்பொறி ஆணைகளையும் விவரங்களையும் ஏற்று அதற்கேற்றபடி செயற்படும் பொழுது நமது வழியமைப்பிலோ, ஆணைகளிலோ பிழைகள் இருக்கலாம். அவ்வாறு ஏதேனும் பிழையிருக்குமேயானால், கணிப்பொறி அதை இயக்குபவருக்கு உடனே அறிவிக்கும். பிழைகளில்லாவிடில், வழிமுறைக்குத் தகுந்த வகையில் அது செயற்பட்டு, அதனால் பெறப்படும் விடையை நமக்கு விடுவரலகங்கள் மூலமாக அளிக்கிறது.

கணிப்பொறியானது விடையை அல்லது அறிவிப்பை, துளை செய்யப்பட்ட அட்டை, தாள், அச்சுப்பொறி, காட்சித்திரை [Display Screen] முதலிய பலவகையான விடுவரலகங்கள் மூலம் நமக்கு அளிக்கிறது. தற்காலத்தில் **பயன்படுத்தப்படும்** பெரும்பாலான கணிப்பொறிகள், நேர் தொடர்பு வசதியைக் [Interactive Facility] கொண்டவையாக உள்ளன. எனவே பிழைச்

செய்திகளைப் [Error Messages] பெறவும், அதற்கேற்ற வகையில் கணிப்பொறியை நாம் ஆணையிடவும் இது மிகவும் வசதியாக உள்ளது. நமது கணக்குக்கேற்ற விடையைத் திரையில் [Screen] பார்த்தறியலாம். நமக்குத் திறப்படிவமாகத் [Hard Copy] தேவைப்பட்டால், அச்சுப்பொறியிலுள்ள வெண்தாளில் அச்சிடுமாறு ஆணையிடலாம். அச்சுப்பொறியின் பல வகைகளையும், அவை எங்ஙனம் வேலை செய்கின்றன என்பதையும் கீழ்க்கண்டவாறு விவரிக்கலாம்.

3.3.1. அச்சுப்பொறி (PRINTER)

கணிப்பொறியானது அதன் மைய இயக்ககம் [CPU] செயற்படும் பொழுது, அதனால் தெரிவிக்கப்படும் செய்திகளை அல்லது விடையை அச்சுப் பொறியில் அச்சடிக்கிறது. இந்த அச்சுப்பொறியை அதன் செயலாக்கத் திறனுக்கும் அமைப்பிற்கும் ஏற்றாற்போல் பலவகையாகப் பிரிக்கலாம். எழுத்து அச்சுப் பொறி [Dot Matrix Printer] ஒவ்வொரு எழுத்தாக அச்சிடும் தன்மை வாய்ந்தது. இது தட்டெழுத்துப் பொறி போன்றதாகும். இது ஒரு நொடியில் பத்துக் குறிகளை அச்சிடக்கூடிய வேகம் பெற்றது. இது பெரும்பாலும் நேரியக்க முறை [On Line] அல்லது நேரப்பகிர்வுச் செயற்பாட்டு முறைக்ளில் [Time Sharing] குறைந்த அளவு உள்ள விவரங்களை அச்சிடுவதற்குப் பயன்படுகிறது. இந்த வகை அச்சுப் பொறியானது, கணிப்பொறியின் வேகத்திற்கு ஈடுகொடுக்க இயலாததாகவுள்ளது. மேலும் அச்சுப்பொறியானது எண், எழுத்து, தனிக்குறிகள் முதலியவற்றை அச்சடிக்கும் திறன் வாய்க்கப் பெற்றிருக்க வேண்டும்.

பெரும்பாலான வெளியீடுகள் தாள்களில் அச்சாகிய நிலையிலேயே நமக்குப் பெரிதும் பயன்படுகின்றன. எடுத்துக் காட்டாக நிதி நிலை அறிக்கை, பேச்சுக் குறிப்புகள், அஞ்சல் தொடர்புகள், தொழிலகங்களில் சம்பளப் பட்டியல்தயாரித்தல், கொடுக்கல் வாங்கல், விற்பனை முதலியவற்றின் விவரப் பட்டியல்கள் தாள்களில் இருப்பதால் கோப்புகளில் சேகரித்து

வைக்கவும் பிறருக்கு அஞ்சல் செய்யவும் செய்திகளைப் பயனுக்கேற்ற வகையில் புயன்படுத்தவும் மிகவும் எளிதாகின்றன.

கணிப்பொறி மூலம் ஆய்வு செய்து கணிக்கப்பட்ட விடை பகுத்துணரப்பட்ட செய்தி ஆகியன தாள்களில் அச்சாகிய நிலையில் அவற்றின் பயன்பாடுகள் அதிகமாகின்றன. எனவே, கணிப்பொறியின் விடுவரலகங்களில் அச்சுப்பொறி மிகவும் முக்கியத்துவம் வாய்ந்ததாகக் கருதப்படுகிறது.

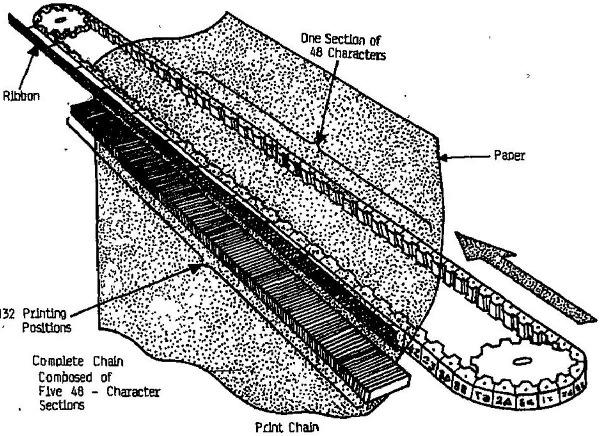
3.3.1.1. இறுக்கநிலை அச்சுப்பொறி (IMPACT PRINTER)

இந்தவகை அச்சுப்பொறியில் அச்சுச்சுத்தி, கரிநாடா, தாள், குறிகள் அமையப்பெற்ற உருளை அல்லது சங்கிலி ஆகிய இவ்வகை பாகங்கள் அமையப்பெற்றிருக்கின்றன. அச்சுச் சுத்தியானது தாளையும் கரிநாடாவையும் தெரிவு செய்யப்பட்ட எழுத்தின் மீது அழுத்துகிறது அப்பொழுது தாளானது தொடர்ந்து நகர்ந்து செல்வதால் தொடர்ச்சியான வரிகள் அச்சாகின்றன. தற்காலத்தில் சுமார் ஒரு நிமிடத்திற்கு 10 வரிகளிலிருந்து 20,000 வரிகள் (ஒவ்வொரு வரியும் 132 குறிகள்) வரை அச்சு செய்யும் திறன் வாய்ந்த அச்சுப் பொறிகள் உள்ளன. இவையாவும் வரி அச்சுப் பொறிகள் [Line Printers] என அழைக்கப்படுகின்றன. பெரும்பாலும் இறுக்க நிலை அச்சுப் பொறிகளே தற்காலத்தில் புழக்கத்தில் உள்ளன.

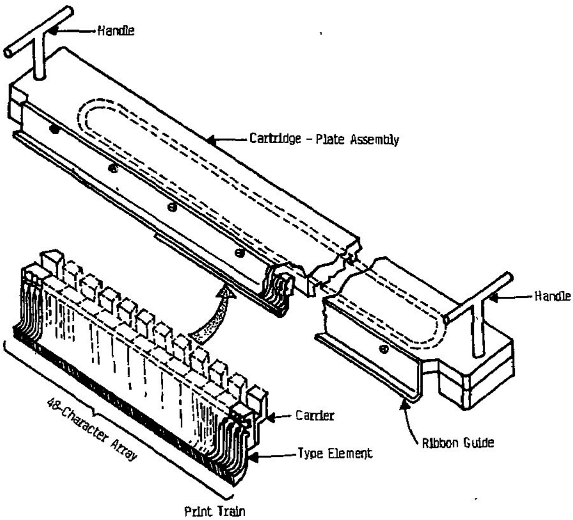
இறுக்கநிலை அச்சுப் பொறிகளில் பலவகைகள் உள்ளன. அவற்றில் சங்கிலி அச்சுப் பொறி [Chain Printer], சக்கர அச்சுப் பொறி [Wheel Printer], சட்ட அச்சுப்பொறி [Bar Printer], உருளை அச்சுப்பொறி [Drum Printer] எழுத்தச்சுப்பொறி [Dot Matrix Printer] ஆகியவற்றின் அமைப்பும் அவை இயங்கும் முறையும் மற்றும் இறுக்கமிலா அச்சுப்பொறியின் இயங்கும் தன்மையும் கீழே சுருக்கமாகக் கொடுக்கப்பட்டுள்ளன.

3.3.1.2. சங்கிலி அச்சுப்பொறி (CHAIN PRINTER)

மின் மற்றும் பொறியமைப்பு [Electro Mechanical] முறையில் இயங்கும் இந்தச் சங்கிலி அச்சுப் பொறி, தொடர்ச்சியானசங்கிலி யைப் பெற்றுள்ளது. படம் எண் 16 (அ) மற்றும் 16 (ஆ)—இல்



படம்—16 (அ) சங்கிலி அச்சுப் பொறி [CHAI



படம்—16 (ஆ)

சங்கிலி அச்சுப் பொறியில் அச்சுப் பலகையமைப்பு

இதன் அமைப்பு விவரிக்கப்பட்டுள்ளது. இந்தச் சங்கிலி 5 பகுதிகளாகப் பிரிக்கப்பட்டு, ஒவ்வொரு பகுதியும் 48 குறிகளை உடையதாகஅமைக்கப்பட்டுள்ளது. எண்,எழுத்து மற்றும் சிறப்புக் குறிகள் இதில் அடங்கும். சங்கிலியானது தொடர்ச்சியான தாளுக்குப் பின்னால் மிகுந்த வேகத்துடன் சுற்றிக்கொண்டிருக்கும். வெண்தாளின் முன்பாகக் கரிநாடாவும், காந்தத் தூண்டுதல் செய்யப்பட்ட 132 சுத்திகளும் அமைக்கப்பட்டிருக்கின்றன.

அச்சடிக்கப்பட வேண்டிய குறிகள் அச்சாகும் நிலைக்கு வரும் பொழுது, சுத்திகள் முன் தூண்டப்பட்டு, சங்கிலியிலுள்ள எழுத்து மற்றும் கரி நாடா மீது வெண் தாளை அழுத்துகின்றன. உடனே எழுத்து அச்சாகிறது. இவ்வாறு பல எழுத்துக்கள் ஒரே சமயத்தில் அச்சாகின்றன.

நமது தேவைக்கேற்ப எழுத்துக்களின் அமைப்பு முறையை மாற்றியமைக்க முடியும். ஏனெனில் எழுத்துக்களைத் தட்டச்சு செய்யும் சங்கிலிகள் சுலபமாக மாற்றக் கூடிய வசதி வாய்ந்தவையாகும்.

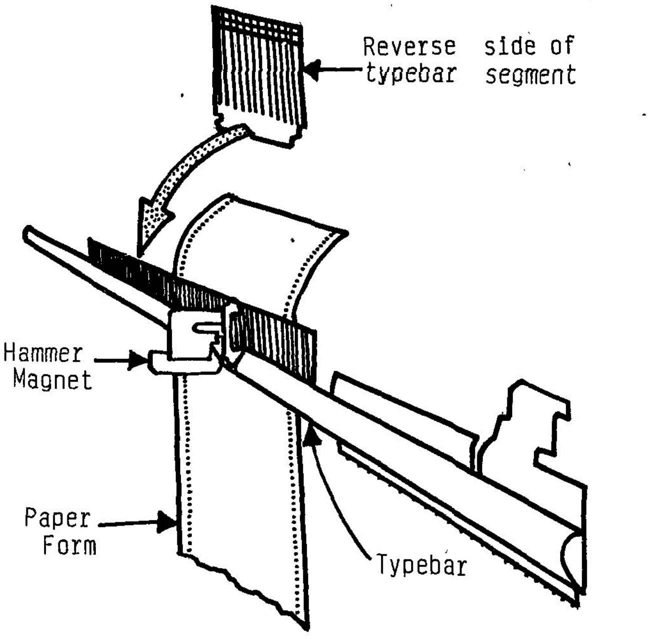
3.3.1.3. சக்கர அச்சுப்பொறி (WHEEL PRINTER)

சக்கர அச்சுப்பொறி இறுக்க நிலை அச்சுப்பொறி வகையைச் சார்ந்ததாகும். இதிலுள்ள உலோகம் அல்லது பிளாஸ்டிக்கால் ஆன அச்சுச் சக்கரம், 88 வகையான எண், எழுத்து மற்றும் சிறப்புக் குறிகளைப் பெற்றுள்ளது. இக்குறிகள் சக்கரத்தின் வெளிப்பகுதியில் சரியான நிலையில் அமைக்கப் பெற்று விளங்குகின்றன.

தேவைப்படும் எழுத்தை அச்சிடுவதற்குத் தகுந்தவாறு ஒவ்வொரு சக்கரத்தையும் தனித்தனியாகச் சுழற்ற முடியும். ஒவ்வொரு சக்கரத்திலும் அந்த எழுத்து அச்சிடும் நிலைக்கு வந்தவுடன் கரிநாடாவுடன் சக்கரங்கள் காகிதத்தில் அழுத்தப் படுகின்றன. இவ்வாறு அழுத்தப்படும் பொழுது தாளில் ஒரு வரி அச்சிடப்படுகிறது.

3.3.1.4. சட்ட அச்சுப்பொறி (*BAR PRINTER*)

சட்ட அச்சுப் பொறியின் இயக்க அமைப்பு, படம் எண் 17–இல் குறிப்பிடப்பட்டுள்ளது. இதனுடைய இயக்கம் சங்கிலி

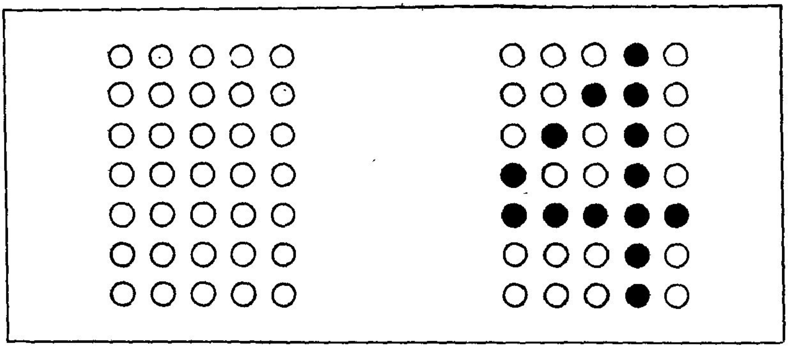


படம்–17 சட்ட அச்சுப் பொறி [BAR PRINTER]

அச்சுப் பொறியைப் போன்றதேயாகும். ஒவ்வொரு சட்டத்திலும் தேவையான எல்லா எழுத்து அச்சுகளும் இருக்கும். ஒரு வரியை அச்சு செய்யும்பொழுது, வேண்டிய எழுத்து வருமாறு ஒவ்வொரு சட்டமும் நகர்ந்து செல்லும். அச்சிடவேண்டிய வரி அவ்வாறு தயாரானதும், சுத்திகளால் சட்டங்கள் தாக்கப்படுகின்றன. தாக்கப்பட்ட சட்டங்களிலுள்ள எழுத்துகள் கரிநாடாவின் [Carbon Tape] *துணையால் காகிதத்தில் வரிசையாக அச்சிடப் படுகின்றன. இந்த வகை அச்சுப் பொறிகள் நிமிடத்திற்குச் சுமார் 200-இலிருந்து 600 வரிகள்வரை அச்சிடும்திறன் வாய்ந்தவை ஆகும்.*

3.3.1.5. எழுத்தச்சுப்பொறி (*DOT MATRIX OR WIRE MATRIX PRINTER*)

இந்த வகை அச்சுப் பொறிகளில் 5 × 7 என்ற புள்ளி அமைப்பில் பல இணைப்பிகளால் தேர்வு செய்யப்படும் புள்ளிகளைக் கொண்டு குறிகள் குறிக்கப்படுகின்றன். படம் எண் 18 (அ) வில் எவ்வாறு 5 × 7 என்ற புள்ளி அமைப்பில் நான்கு (4) குறிக்கப் படுகிறது என்பதைக் காணலாம். பல இணைப்பிகளால் தெரிவு செய்யப்படும் குறியை மை நாடா மீது அழுத்தும்பொழுது, பல விதமான எழுத்துகள் வெண்தாளில் அச்சாகின்றன. எனவே இந்த வகையில் அச்சாகிய எழுத்துகள் அமைப்பில் தொடர்ச்

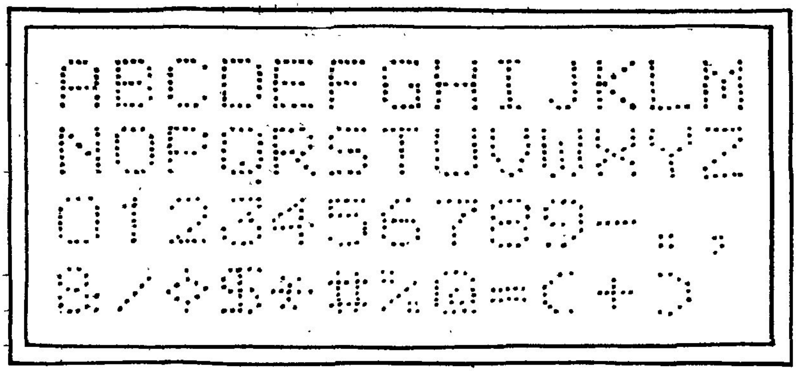


படம்—18 (அ)

எழுத்தச்சுப் பொறியில் அச்சுப் புள்ளிகளின் அமைப்பு

[Dot Patterns in a Dot Matrix Printer]

சியான கோடுகளைக் கொண்டமையாமல் பல புள்ளிகளின் சேர்க்கை முறையில் அமைந்திருக்கும். படம் எண் 18 (ஆ) வில் எழுத்தச்சுப்பொறியில் பயன்படுத்தப்படும் குறித்தொகுப்புகள் புள்ளி அமைப்பு முறையில் காட்டப்பட்டுள்ளன.

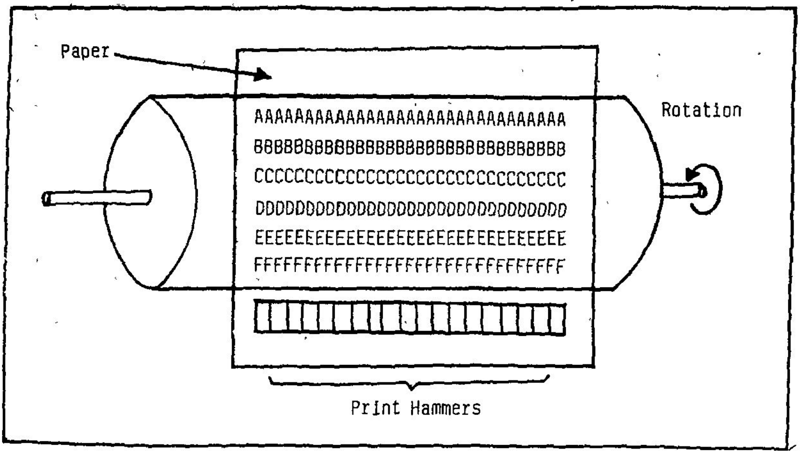


படம்18— (ஆ)
எழுத்தச்சுப் பொறியில் குறித்தொகுப்பு
[Character set in a Dot Matrix Printer]

Matrix என்று கூறப்படும் புள்ளி அமைப்புடைய பகுதியானது வலப்புறத்திலிருந்து இடப்புறத்திற்கு வெண்தாளின் முன்பாக நகர்ந்து செல்கிறது. அவ்வாறு செல்லும்பொழுது எழுத்துகள் ஒரு வரியில் தொடர்ச்சியாக அச்சாகின்றன. இம்முறையில் அச்சிடப்படும் எழுத்துகள் மிகவும் தெளிவாக இருக்கும் எனினும், இவ்வகை அச்சுப்பொறிகள் வேகம் குறைந்தவையாகும்.

3.3.1.6. உருளை அச்சுப்பொறி (DRUM PRINTER)

உருளை அச்சுப் பொறியின் அமைப்பு, படம் எண் 19-இல் விவரிக்கப்பட்டுள்ளது. இதில் உருளை வடிவமுள்ள பாகமும் அதன் மேற்பரப்பில் பல வரிசைகளில் எழுத்துகளும் அமைக்கப் பட்டுள்ளன. உருளை சுற்றும்பொழுது சரியான எழுத்து அச்சாகும் நிலைக்கு வருகிறது. அதே சமயம் அச்சுச் சுத்தியானது இயக்கப்படுகிறது.ஒரே சமயத்தில் ஒரு வரி முழுவதும் 'A' ஆகவும் பின் 'B' ஆகவும் அச்சாகிறது. ஒரு வரி அச்சாவதற்கு உருளை ஒரு முழுச் சுற்றை முடிக்கிறது.எனவே இந்த வகை அச்சுப்பொறிகளில் அச்சிடும் வேகம், உருளை சுற்றும் வேகத்தைப் பொறுத்து அமையும்.



படம்–19 உருளை அச்சுப்பொறி [Drum Printer]

3.3.1.7. இறுக்கமிலா அச்சுப்பொறி (NON IMPACT PRINTER)

கணிப்பொறி ஒரு வினாடியில் ஆயிரக்கணக்கான ஆணைகளைப் புரிந்துகொண்டு செயற்படும் திறன் வாய்ந்தது என முன்பே பார்த்தோம். ஆனால் இதன் வேகத்துடன் அச்சுப் பொறியின் வேகத்தை ஒப்பிடும்பொழுது, அச்சுப்பொறி மிகவும் வேகக் குறைவானது என்பது நன்கு புலனாகும். ஆனால் இறுக்கமிலா அச்சுப்பொறி,மின் மற்றும் ஒளிப்பட இயக்க ரீதியில் இயங்குவதால் மற்றவகை அச்சுப்பொறிகளைக் காட்டிலும் வேகம் மிக்கதாகும். இறுக்க நிலை அச்சுப்பொறியில் எந்திரப் பகுதிகளின் [Mechanical] செயற்பாட்டு வேகத்தைப் பொறுத்துச் செய்திகளின் அச்சிடும் வேகம் அமைந்திருந்தது. ஆனால் இது போன்ற எந்திரப் பகுதிகள் நீக்கப்பட்டமையால் இறுக்கமிலா அச்சுப் பொறியின் வேகம் பன்னூறு மடங்கு அதிகரித்தது. ஒரு நிமிடத்திற்கு 20,000 வரிகளை அச்சாக்கும் லேசர் [Laser] அச்சுப் பொறிகளையும் ஒளிப்பட ரீதியில் செயற்படும் கணிப் பொறி நுண்பட இடுவரல் மற்றும் விடுவரல் [Computer Input/ Output Micro film] அச்சுப் பொறிகளையும் எடுத்துக்காட்டாகக் கூறலாம்.

3.4. ஆளகம் (CONTROL UNIT)

இடுவரல் மூலமாகக் கணிப்பொறிக்கு ஊட்டப்படும் செய்திக்கூறுகளையும் ஆணைகளையும் அறிந்துணர்ந்து அவை களை நினைவகத்திற்குக் கொண்டு செல்லவும் தவறு இருப்பின் பிழைச்செய்தியைத் தெரிவிக்கவும் ஆளகம் ஆணைபிறப்பிக்கிறது. இவ்வாறு நினைவகத்திற்குக் கொண்டு செல்லப்பட்ட செய்தி களைத் துணை நினைவகங்களில் பதிவு செய்யவும் கணக்கீடு செய்யவும் கணக்கிடப்பட்ட விடைகளை விடுவரல் மூலமாக நமக்கு அளிக்கவும் தேவையான கண்காணிப்பு வேலைகளையும் ஆளகம் மேற்கொள்கிறது. மொத்தத்தில் ஆளகம் கணிப்பொறி யின் செயற்பாட்டினைக் கட்டுப்படுத்தும் ஒரு முக்கிய உறுப்பு என்று கூறின் அது மிகையாகாது.

3.5. கணக்ககம் (ARITHMETIC UNIT)

இடுவரல் மூலம் பெறப்பட்ட, நினைவகத்திலுள்ள செய்தி யைப் பகுத்து விடை காணவும் பெருக்கல், கழித்தல், கூட்டல் முதலிய செயல்களைச் செய்யவும் கணக்ககம் துணை புரிகிறது. எடுத்துக்காட்டாகக் கூறின், சம்பளப் பட்டியல் தயாரித்தலில் ஊதிய விகிதம், செலவினங்கள், சேமிப்பு நிதி, பிற படிகள் முதலானவற்றை அறிந்துணர்ந்து கழிக்கவும், கூட்டவும் செய்து, மாத ஊதியத்தைக் கணக்கிடுவதில் கணக்ககம் பெரும் பங்கு வகிக்கிறது. மேலும் அறிவியல், ஆராய்ச்சி முதலியவற்றில் வரும் ஏராளமான கணக்குகளைக் கணிப்பொறிமூலம் செய்யும்பொழுது கணக்ககம் அதிகஅளவு கணிப்புச்செயல் [Computation] புரிகிறது.

3.6. கணிப்பொறியின் இயக்கமுறை (OPERATIONAL PROCEDURE)

இனி, கணிப்பொறி எப்படி இயங்குகிறது என்ற விவரத் தைச் சுருக்கமாகப் பார்ப்போம். கணிப்பொறியுடன் மனிதன் தொடர்பு கொள்வதற்குப் பல தனிப்பட்ட மொழிகள் உள்ளன. நாம் பேசும் மொழிக்குப் பல இலக்கண வரையறைகள் இருத்தல் போலவே, ஒவ்வொரு கணிப்பொறி மொழியும் பல இலக்கண வரையறைகளுக்குட்பட்டு இயங்குகிறது. நாம் கணிப்பொறியைக் கொண்டு ஒரு கணக்குக்குத் தீர்வு காணவேண்டுமாயின், ஏதாவது நமக்குத் தெரிந்த ஒரு கணிப்பொறிமொழியில் அதை வழிமுறைப்

படுத்த வேண்டும் [Program-Coding]. பின் அதை முன்பு குறிப்பிட்ட ஏதாவதொரு வகை இடுவரல் மூலம் அதற்கு ஊட்ட வேண்டும்.

ஆளகத்தின் உத்தரவின் பேரில் இந்த வழிமுறைக் கட்டளைகள் கணிப்பொறியின் நினைவகத்தில் பதிவு செய்யப்படுகின்றன. கணிப்பொறியானது 0 அல்லது 1 என்ற எண்களால் ஆகிய பொறிமொழியை மட்டுமே படித்தறிந்து அதற்கேற்பச் செயற்படும் திறன் வாய்ந்தது. எனவே, நாம் கொடுக்கும் கட்டளைகளைப் பொறிமொழிக்கு மாற்றத் தேவையான மொழிமாற்றிகள் [Translators–Interpreter–Compiler] நினைவகத்தில் முன்பே நிலையாகப் பதிவு செய்யப்பட்டிருக்கும். ஆளகம் நாம் கொடுக்கும் கட்டளைகளை ஏற்று, நினைவகத்திலிருந்து மையச் செயலகத்திற்கு வழியமைப்பைக் கொணர்ந்து மொழி மாற்றி மூலம் பொறிமொழிக்கு மாற்றம் செய்கிறது. இவ்வாறு மொழி மாற்றம் செய்யப்பட்ட பொறிமொழி வடிவத்திலுள்ள கணக்கை மையச் செயலகம் அறிந்துணர்கிறது. பின்னர் கொடுக்கப்பட்ட கட்டளைப்படி கணித்து, விடையை நமது விருப்பத்தின்படி மின்காந்த நாடாவிலோ, அச்சுப்பொறியின் வெண்தாளிலோ, காட்சிப் பொறியின் திரையிலோ கொடுக்கின்றது. நாம் விரும்பினால் இந்த விடையைத் துணை நினைவகமாகிய மின்காந்தத் தட்டிலோ, மின்காந்த நாடாவிலோ பதிவு செய்து வைத்துக்கொண்டு, திரும்பத்திரும்ப விடை தேவைப்படும் பொழுது புதிய செய்திக் கூறுகளைக் கொடுத்து, அதற்கான விடையைப் பெறலாம்.

3.7. நினைவகம் (MEMORY)

ஒரு செய்தியை அல்லது காட்சியை நினைவிற் கொண்டு, அதன்படி செயற்பட மனிதனுக்கு நினைவு அவசியம். அதே போன்று ஆணைகளையும் வழியமைப்புகளையும் செய்திக்கூறுகளையும் இடுவரல் [Input] மூலமாக நாம் கணிப்பொறிக்கு ஊட்டும்பொழுது, அவை ஆளகத்தின் ஆணையின் பேரில் நினைவகத்தில் பதிவு செய்யப்படுகின்றன. இவ்வாறு ஊட்டப்படும் செய்திகளை அல்லது ஆணைகளைக் கணிப்பொறி ஏற்று, விரைவாகவும் துல்லியமாகவும் செயற்படுகிறது. இது மனிதன் மற்றவர் கட்டளையிடும்பொழுது அதை நினைவு கூர்ந்து, பின் செயலாற்றும் தன்மையை ஒத்ததாகும். ஆனால் அமைப்பு முறையிலும்

செயலாக்கத் திறனிலும் கணிப்பொறியின் நினைவகத்திற்கும் மனிதன் மூளைக்கும் நிறைய வேறுபாடுகள் உண்டு.

கணிப்பொறியின் நினைவகத்தை அதன் செயலாக்க ரீதியில் இரு பெரும் பிரிவுகளாகப் பிரிக்கலாம்.

1. முதன்மை நினைவகம்
2. துணை நினைவகம்.

கணிப்பொறியின் செயற்பாட்டின்பொழுது முதன்மை நினைவகம் தன்னகத்தே செய்திக்கூறுகளையும் வழியமைப்பு ஆணைகளையும் [Program Instructions] கொண்டுள்ளது. இதன் மூலமாகக் கூட்டல், கழித்தல் முதலான கணிப்புப் பணிகளையும் மற்றும் இதர வேலைகளையும் மையச் செயலகம் வேண்டியாங்கு செய்து முடிக்கத் துணை செய்கிறது. ஆனால் துணை நினைவகம் செய்திகளைத் தன்னகத்தே நிலையாக இருத்தி வைத்துக்கொண்டு, மையச் செயலகம் தேவை என்று கட்டளையிடும் பொழுது மட்டுமே செய்திகளையும் ஆணைகளையும் அனுப்பி வைக்கிறது. இரும்பு வளைய நினைவகம் [Core Memory], காந்த உருளை நினைவகம் [Magnetic Drum Memory], நிறைவிலா மின் கடத்தி நினைவகம் [RAM, MOS Memories] ஆகியவை முதன்மை நினைவகத்திற்கு எடுத்துக்காட்டுகளாகும். இவைகளின் அமைப்பு முறை, செயலாக்கம் முதலியவற்றை, பாகம் எண் 5—இல் விரிவாகக் காணலாம். மின்காந்த நாடா, மின்காந்தத் தட்டு, மென்வளைத்தட்டு [Floppy Disk] ஆகியவை துணை நினைவகங்களுக்கு எடுத்துக்காட்டுகளாகும். இவைபற்றி, பாகம் எண் 5—இல் கூறப்படும்.

ஒரு பெரிய விடுதியில் ஒரே மாதிரியான பல அறைகள் உள்ளன. ஒவ்வொரு அறையும் ஒவ்வோர் எண்ணால் குறிப்பிடப் பட்டுள்ளது. அதைப்போன்றே கணிப்பொறியின் நினைவகம் பல அகங்களைக் (பதிவு அறைகளைக்) கொண்டது. ஒவ்வொரு அகத்துக்கும் ஒரு தனி எண் அல்லது சுட்டு எண் அதாவது முகவரி [Address] கொடுக்கப்பட்டுள்ளது. நாம் இடுவரல் மூலம் செலுத்தும் செய்திக்கூறு, ஆணை ஆகியவை ஒரு குறிப்பிட்ட அகத்தில் நினைவிலிருத்தப்படும். இதுவரை நாம் கணிப்பொறியின் சில உறுப்புகளைப்பற்றி அறிந்தோம். மின்னணுவியல் கருவிகளையும் அவற்றை ஆட்டி வைக்கும் ஆணைகளின் திறமையையும் பொறுத்து, கணிப்பொறியின் ஆற்றல் அமைகிறது.

பாகம் – 4

4. கணிமம்

4.1. முன்னுரை

கணிப்பொறிக்கு மனிதனைப் போன்று உணர்ச்சியோ, தானாகச் சிந்திக்கும் திறனோ கிடையாது. இவ்விதமான ஒரு கருவியைக்கொண்டு உலகமே வியக்கும் வண்ணம், பலதரப்பட்ட பணிகளை மிகவிரைவாகவும் திறம்படவும் செய்து முடிப்பதற்குத் துணை செய்வது வழியமைப்பாகும். செய்ய வேண்டிய பணியை முறைப்படித் தொடங்கி, செயற்படுத்தி முடிக்கக் கணிப்பொறிக்கு உதவும் முறையே வழியமைப்பாகும். இன்றைய நவீன உலகில் சிற்றங்காடிகளில் பொருள் விற்பனை, மனைப்பாங்குக் கணக்கு முதலிய பயன்பாட்டிலிருந்து, வானவெளியில் விண்கலத்தைச் செலுத்தி விந்தைமிகு பல ஆய்வுப் பணிகள் செய்வதுவரை கணிப்பொறி வாழ்வின் ஒவ்வொரு நிலையிலும் உற்றதொரு பயன்பாட்டுக் கருவியாகத் திகழ்கிறது என்பதில் ஐயமில்லை. பல பயன்பாடுகளுக்கும் ஏற்ப, சிறந்த கணிமங்களை வடிவமைக்காவிடில் மிகத்திறன் வாய்ந்த கருவியும் வாய்க்கப் பெற்றிருந்தாலும் பயனின்றிப் போகும். எனவே கணிப்பொறியில் ஒவ்வொரு குறிப்பிட்ட செயலைச் செய்வதற்கும் தனிப்பட்ட வழியமைப்புகள் [Special Programs] தேவைப்படுகின்றன. ஒவ்வொரு வழியமைப்பும் பலவழியமைப்பு ஆணைகளை [Program Instructions] உள்ளடக்கியுள்ளது. இவ்வாறு பல வழியமைப்புகள் சேர்ந்த தொகுதியே கணிமம் [Software] என்று பெயர் பெறும். மனித

உடல் இயங்குவதற்கு எவ்வாறு உயிர் அவசியமோ, அவ்வாறே கணிப்பொறி இயங்குவதற்குக் கணிமம் மிகவும் அவசியமாகிறது.

தொழிற்கூடங்களில் பொருள் உற்பத்திக்குப் பற்பல மூலப் பொருட்கள் தேவைப்படுகின்றன. பொருட்களின் உற்பத்தித் திட்டமிடலுக்கேற்ப மூலப்பொருள்தேவை, மற்றும் நாள்தோறும் தேவைப்படும் மூலப்பொருட்களின் அளவு, குறைந்த பட்ச மூலப் பொருள் கையிருப்பு, மற்றும் மூலப்பொருளை வாங்குவதற்கான வழங்காணை அனுப்புதல் குறித்த அறிவுரை வழங்குதல் மற்றும் நிருவாகம், விற்பனை, பொருள் வழங்கு துறைகளுக்கேற்ற அறிக்கைகளைத் தயார் செய்தல் முதலான பல பணிகளைச் செவ்வனே சூழ்ந்துணர்தல் கணிப்பொறி வழிபொருள் நிலையா ளுகையில் [Inventory Control] கவனிக்கப்பட வேண்டிய முக்கிய மான பணிகளாகும். எனவே, கணிப்பொறி மூலம் பொருள் நிலையாளுகையைச் செயற்படுத்த வேண்டுமாயின், அதற்கான வழியமைப்பை உருவாக்கும்போது கடைப்பிடிக்கப்பட வேண்டிய வழிமுறைகளைக் கீழ்க்கண்டவாறு விவரித்துக் கூறலாம்.

எ.கா:

1. உற்பத்தி செய்யப்படும் பொருளின் தரம், அதன் எண்ணிக்கை [Quantity], உற்பத்தித் திட்டமிடல் பற்றிய விவரங்கள், உற்பத்திக் காலம், நாள்தோறும் தேவைப்படும் மூலப்பொருளின் அளவு முதலான செய்தி களைப் படித்து அறி.

2. பொருளைப் பல உட்பிரிவுகளாகப் பிரித்து [Sub-Assembly], அவற்றிற்குக் குறிப்பிட்ட எண்களை [Assembly Number] நிருணயம் செய்.

3. ஒவ்வோர் உட்பிரிவிலும் உள்ள மூலப்பொருள்களைக் [Component] கண்டறிந்து, அவற்றிற்கு எண் [Stock No.] நிருணயம் செய்.

4. குறிப்பிட்ட காலத்திற்குப் பொருள் உற்பத்தியைச் சூழ்ந்துணர்ந்து [Production Planning] அதற்குத்தேவை யான மூலப்பொருள்களின் அளவைத் தீர்மானம் செய்

5. இடைநிலைக் காலத்திற்குத் [Interim Period] தேவையான மூலப் பொருட்களையும் கணக்கிட்டுக்கொள்.

6. தொழிற் கூடத்திற்கு நாள்தோறும் ஒரு குறிப்பிட்ட பொருளின் உற்பத்திக்காக அனுப்பப்பெறும் மூலப் பொருட்களின் அளவைக் குறித்துக் கொள்.

7. அவ்வாறு அனுப்பப்பட்ட மூலப்பொருட்களின் எண்ணிக்கையை இனம் வாரியாக மொத்த இருப்பிலிருந்து கழி.

8. ஒவ்வொரு மூலப் பொருளிலும் குறைந்த அளவு இருப்பு [Minimum Inventory] இருக்கிறதா என்று சோதித்தறி. அவ்வாறு இல்லாவிடில் முன்பே சூழ்ந்து [Planned] உணர்ந்தபடி அம்மூலப் பொருட்களை வாங்குவதற்கு வழங்காணை [Purchase Order] அனுப்புவதற்கான அறிக்கையைத் [Inventory Report] தயார் செய்.

9. நாள்தோறும் செலவிடப்படும் மூலப் பொருட்களின் எண்ணிக்கை, குறைந்த பட்ச மூலப்பொருள் கையிருப்பு [Minimum Stock], புதிதாக மூலப்பொருள் வாங்கிய விவரங்கள் ஆகியவற்றைக்கொண்டு அறிக்கை தயார் செய்.

மேற்கண்ட பொருள் நிலையாளுகைப் பணியைச் செய்வதற்கும் தொழிற்கூட நிருவாகத்தில் தேவைப்படும் பலவகையான நிருவாக அறிக்கைகளைத் தயார் செய்யவும் [Management Report] கணிப்பொறி பெரிதும் உதவுகிறது. மேலும் குறிப்பிட்ட மூலப் பொருளின் கையிருப்பு, குறைந்தபட்ச அளவிற்கு வரும் பொழுது உரியநேரத்தில் அதை வாங்கும்பொருட்டு வழங்காணை அனுப்புமாறு, பொருள்வாங்கு துறைக்கு அறிவுறுத்துவதற்கான அறிக்கையைத் தயார் செய்யவும் இயலுகிறது.

ஆனால், இப்பணியைத் தவறில்லாமல் விரைவில் செய்வதற்குத் தேவையான வழியமைப்பையும் அதைச் செய்யும் வழிமுறைகளையும் [Procedures] நாம் கணிப்பொறிக்குத் திறம்பட முறைப்படுத்திக் கூறுதல் அவசியமாகும். வழியமைப்பைச் செய்யும் பொழுது கீழ்க்கண்ட வழிமுறைகளைப் பின்பற்றல் வேண்டும்.

1. மொத்தப் பொருள் உற்பத்தி, அதற்குத் தேவைப்படும் மூலப்பொருட்கள், கால அளவு. இடைநிலைக் காலத்திற்குத் தேவைப்படும் மூலப்பொருட்கள், குறைந்தபட்ச மூலப்பொருள் கையிருப்பு போன்ற செய்திகளை இடுவரலகம் [Input Unit] மூலமாக இடுவரல் செய்ய வேண்டும். உற்பத்தித் திட்டமிடல் [Production Planning] பற்றிய விவரங்கள் மற்றும் முறைப்படுத்தப்பட்ட அட்டைகளில் பதிவு செய்யப்படும் செய்திகள், [Ex: CardexRecord] பொருள் வழங்கு துறையில் நாள் தோறும் கையாளப்படும் செய்திகள், ஆகியவற்றிலிருந்து மேற்கண்ட செய்திக் கூறுகள் பெறப்படுகின்றன.

2. இவ்வாறு இடுவரல் செய்யப்பட்ட செய்திக் கூறுகளும் வழியமைப்பு செயற்பட்டபின் கணிக்கப்பட்ட விடைகளும் நினைவகத்தில் பதிவு [Storage] செய்யப்படுகின்றன.

3. விவரங்களை நமது கணக்குக்கேற்பக் கூட்டவோ, கழிக்கவோ, பெருக்கவோ தேவையான கணக்கீட்டுச் செயல்களைச் [Arithmetic Operation] செய்ய வேண்டியுள்ளது.

4. இவ்வாறு தொகுத்து வகைப்படுத்தப்பட்ட [Analysed] செய்திகளிலிருந்து விடை, விடுவரல் ஒருங்கு [Output System] மூலமாக நமக்குக் கிடைக்கிறது.

4.2. செய்வழியமைப்பு முறைமை (PROGRAMMING METHODOLOGY)

கணிப்பொறி மூலம் ஒரு கணக்குக்கு நாம் விடைகாண வேண்டுமென்றால், குறிப்பிட்ட வழியமைப்பு முறைமையைக் [Programming Methodology] கவனத்தில் கொள்ளவேண்டும். முதலில் கணக்கை நன்கு ஆய்வு செய்தல் அவசியம். நமது கணக்கு அறிவியல் தொடர்பானது எனில், அதைச் சமன்பாடுகளாகக் குறிப்பிடலாம். நன்கு படித்துணர்ந்து ஆய்வு செய்தபின் கணக்கைச் சமன்பாடுகளாகவோ, கணிதப் பாங்கிலோ [Mathematization] குறிப்பிடும்பொழுது கணக்கியற் கோட்பாடுகளுக்கேற்பக்குறிகளைக் கொண்டுமுறைப்படுத்தல் [Formulation]

அவசியமாகும் பின்னர் இவற்றை ஆய்வு செய்து தவறு ஏதேனும் இருப்பின், திருத்தல் வேண்டும். தவறு ஒன்றும் இல்லாவிடில், மேற்குறிப்பிட்ட விவரங்களைப் படம் மூலம் குறிப்பிடத் தேவை யான கட்டப்படம் [Flow chart] வரைதல் அவசியம். கட்டப் படமானது ஒரு கணக்குக்கு முறைப்படி எவ்வாறு தீர்வு காண வேண்டும் என்பதைப் படம் மூலம் சுலபமாக விளக்கும் ஒரு வழியாகும். இப்படம், கணக்கைப் பற்றிய எல்லா விவரங்களை யும், கணிப்பொறி மூலம் தீர்வுகாண முற்படும் பொழுது ஏற்படும் இடைநிலை மாற்றங்களையும் கணக்குக்கான இறுதித் தீர்வைப் பெறும் வழியையும் நன்கு விளக்கிக் காட்டும். இவ்வாறு படிப் படியாகக் கணக்குக்குத் தீர்வு காணும் பொழுது, அடுத்தடுத்து என்ன செய்யவேண்டும் என்பதை உறுதி செய்து கொள்வதற்கு ஏற்ற கருவி, வழியமைப்புக் கட்டப்படம் ஆகும் [Programming Flow Chart]. பின்னர் இந்த விவரங்களை வழியமைப்பு ஆணை களாக மாற்ற வேண்டும். செய்வழியமைப்பு முறைமையைப் பற்றியும், ஒரு கணக்கைக் கணிப்பொறி மூலம் தீர்வு காண்பதற் கான படிப்படியான வழிமுறைகளைப் பற்றியும் பின்னர் விரிவாக ஆராய்ந்தறியலாம். வழியமைப்பு ஆணை ஒவ்வொன்றையும் இரண்டு பகுதிகளாகப் பிரிக்கலாம்.

1. இயக்கி [Operator]

2. இயமி [Operand]

எடுத்துக்காட்டாக A = B + C என்ற சமன்பாட்டை வழி யமைப்பு ஆணையாகக் குறிப்பிடும்பொழுது B, C இயமிகளென்றும் [Operands] '+' இயக்கியென்றும் அழைக்கப் பெறுகின்றன. அதாவது **இயமிகளின்** மீது குறிப்பிடப்பட்ட + என்ற செயல் புரியப்பட்டு, விடை கணிக்கப்படுகிறது. இந்த இயக்கம் [Operation] அல்லது செயலின் தன்மை '+' என்ற இயக்கியால் பெறப்படுகிறது.

மேற்கண்ட செய்திகளை விளக்க, ஒரு சிறிய கணக்கைத் தீர்வு செய்வதற்கான வழியமைப்பை எங்ஙனம் வடிவமைப்பது என்பதைக் கவனிப்போம்.

கணக்கு: மூன்று வெவ்வேறு எண்கள் நினைவக எண் [Memory address] 1000, 2000, 3000 என்ற இடங்களில் பதிவு செய்யப் பட்டு இருப்பதாக வைத்துக் கொள்வோம். முதல் எண்ணையும்

2—வது எண்ணையும் கூட்டி அதிலிருந்து மூன்றாவது எண்ணைக் கழித்து,விடைபெற வேண்டியிருப்பதாக வைத்துக்கொள்வோம். இதற்கான செயல் முறையைக் கீழ்க்கண்டவாறு வரையறுத்துக் கூறலாம்.

1. ஆரம்பநிலைப்படுத்துக [INITIALISE]

கணிக்கும் பணியைச் செய்வதற்குக் கணக்ககத்தில், சேர்ப்பகம் [Accumulator] என்ற பதிவறை [Register] உள்ளது. இந்தச் சேர்ப்பகத்தின் துணையுடன் கூட்டல், கழித்தல் முதலிய கணக்கியற் செயல்களைச் செய்ய முடிகிறது. கணிப்புச் செயல் நடைபெறுவதற்கு முன், சேர்ப்பகம் ஆரம்ப நிலைப்படுத்தப்படுதல் [Clear the Accumulator] வேண்டும். இல்லையெனில், சேர்ப்பகத்திலுள்ள முந்தைய விடை, கணக்கின் தீர்வைப் பாதிக்கும்.

2. நினைவக எண் 1000—இல் உள்ள செய்திக் கூறுகளைச் சேர்ப்பகத்திற்கு மாற்று.

3. நினைவக எண் 2000—இல் உள்ள செய்தியைச் சேர்ப்பகத்திலுள்ள எண்ணுடன் கூட்டு. விடையைச் சேர்ப்பகத்திலேயே இருத்தி வை.

4. நினைவக எண் 3000-இல் உள்ள செய்தியைச் சேர்ப்பகத்திலுள்ள எண்ணிலிருந்து கழி. விடையை நினைவக எண் 1000—இற்கு மாற்று. இப்போது நமக்குத் தேவையான விடை நினைவக எண் 1000—இல் கிடைக்கிறது.

1. Initialise the Accumulator, Read the data from the memory location 1000 and store it in the Accumulator, then read from the memory location 2000 and add this Number with the contents of the Accumulator, and store the result in the Accumulator. Subtract the Number from the location 3000 from the contents of the Accumulator. Store the result in the Memory location 1000.

மேலே குறிப்பிடப்பட்ட வழியமைப்பு முறைமையை யொட்டி உருவாக்கப்பட்ட வழியமைப்பு, கணிப்பொறியின் மூலமாக ஆய்வு செய்யப்பட்டு, தவறு இருப்பின் பிழைச் செய்தி [Error

Message] வெளிப்படுத்தப்படுகிறது. எனவே வழியமைப்பாளர் தவறைத் திருத்தி மீண்டும் மொழி மாற்றம் செய்கிறார். இல்லாவிடில் நமக்குத் தேவையான விடை கணித்துக் கொடுக்கப்படுகிறது.

4.3. கணிப்பொறி மொழிகள் (COMPUTER LANGUAGES)

மொழி ஒன்று இல்லையேல் மனித அறிவு வளர்ச்சி, நாகரிக முன்னேற்றம், அறிவியல் வளர்ச்சி முதலியவை கிடையா. எனவே மொழிஒன்றே மனிதனின் பண்பாட்டு மேன்மையையும் அறிவியல் முன்னேற்றத்தையும் எடுத்துக்காட்டும் உற்ற சாதனமாகும். உலகில் பல நூற்றுக்கணக்கான இயற்கை மொழிகள் [Natural Languages] பேசப்பட்டு வருகின்றன. ஒவ்வொரு மொழியும் தனித்தன்மை வாய்ந்ததாகவும், அதற்கு என, குறிப்பிட்ட இலக்கண இலக்கிய வரையறைகளையும் பெற்று விளங்குகின்றது. இலக்கணம் இல்லையேல் ஒரு மொழியின் அமைப்பே இல்லை எனக் கூறலாம். இதைப் போன்றே கணிப்பொறிகளுக்கும் பல வகை மொழிகள் உண்டு, அதன் மூலமாகத்தான் நாம் கணிப்பொறிக்குத் தேவையான செய்திக் கூறுகளையும் வழியமைப்புகளையும் இடுவரல் செய்யமுடிகிறது. இந்த வழியமைப்பு மொழிகளும் [Programming Languages] இயற்கை மொழியைப் போன்றே, சில குறிப்பிட்ட இலக்கண வரையறைகளுக்கு உட்பட்டுச் செயல்படுகின்றன.

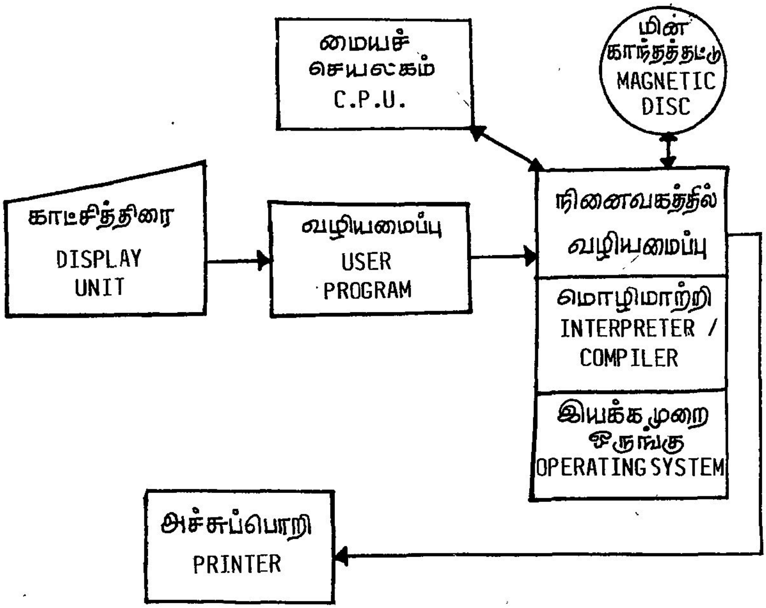
கணிப்பொறித் துறையில் ஏற்பட்டுள்ள மறுமலர்ச்சி காரணமாக, தற்காலத்தில் மிகவும் திறன் வாய்ந்த அதிநுட்பம் பொருந்திய [Sophisticated] கணிப்பொறிகள் உருவாக்கப்படுகின்றன. எனவே, கணிப்பொறிகளின் நுட்பம் வாய்ந்த செயற்பாட்டிற்கும் தற்காலத்தில் பல்துறைகளின் பயன்பாட்டிற்கும் ஏற்பப் பல நூற்றுக்கணக்கான கணிப்பொறி மொழிகள் உலகு எங்கிலும் புழக்கத்தில் உள்ளன. எனினும் கோபால், பேஸிக், போர்ட்ரான், ஆல்கால் ஆகிய கணிப்பொறி மொழிகளே உலகில் பெரும்பாலும் பயன்படுத்தப்பட்டு வருகின்றன. இந்த மொழிகளைப்பற்றி விரிவாகப் பின்னர் காணலாம்.

4.4. மொழி மாற்றிகள் (TRANSLATORS OR COMPILERS)

கணிப்பொறியில் பலவகையான வழியமைப்பு மொழிகள் பயன்படுத்தப்பட்ட போதிலும், இறுதியில் அது படித்துணரக் கூடிய மொழி, பொறி மொழி [Machine Language] ஒன்றே யாகும். கணிப்பொறியானது 0 அல்லது 1 என்ற எண்ணால் ஆகிய பொறி மொழியை மட்டுமே படித்துணர்ந்து அதற்கேற்பச் செயற்படும் திறன் வாய்ந்தது. எடுத்துக்காட்டாகக் கூறின், நமக்குத் தெரியாத ஒரு மொழியில் ஒருவர் சொற்பொழிவு ஆற்றுகிறார் என்று வைத்துக்கொள்வோம். அதை நாம் புரிந்து கொள்ளவேண்டுமெனில், அவர் பேசும் மொழியும் நமக்குத் தெரிந்த மொழியும் கற்றுத் தேர்ச்சிபெற்ற ஒருவர் அதை மொழி மாற்றம் செய்து கூறினால் மட்டுமே நம்மால் புரிந்துகொள்ள இயலும். ஆகவே, நமக்குத் தெரியாத ஒரு மொழியில் செய்திகளை ஒருவர் பேசும் பொழுது, அதைப் புரிந்துகொள்வதற்குக் குறைந்த அளவு இருமொழிகளும் கற்று அறிந்த ஒரு மொழிபெயர்ப்பாளர் [Translator] தேவை. இதைப்போன்றே, வழியமைப்பாளர்கள் தங்களுக்குப் புரியக் கூடிய கணிப்பொறி மொழியில் வழியமைப்பினைச் செய்வர். ஆனால், கணிப்பொறியானது இதைப் புரிந்துகொண்டு செயற்பட வேண்டுமெனில் இதைப் பொறி மொழியாக மாற்ற, திறன் வாய்ந்த ஒரு மொழிபெயர்ப்புச் சாதனம் அவசியமாகிறது. எனவே, வழியமைப்பாளர் உருவாக்கும் வழியமைப்பு, ஒரு குறிப்பிட்ட கணிப்பொறி மொழியிலிருக்கும்பொழுது, அதைக் கணிப்பொறி புரிந்துகொண்டு செயற்படுதலுக்கேற்ப மொழி மாற்றம் செய்யும் சாதனமே 'மொழி மாற்றி' என வரையறுத்துக் கூறலாம்.

எடுத்துக்காட்டாகக் கூறின் போர்ட்ரான் [Fortran], பேஸிக் [Basic], கோபால் [Cobol], பாஸ்கல் [Pascal] ஆகிய பல உயர்நிலை மொழிகள் வழியமைப்பாளர்கள் வழியமைப்புகளை உருவாக்க ஏற்றவையாகும். இந்த உயர்நிலை மொழிகள் [Higher Level Language] எளிதில் படித்தறிந்து கொள்ளக்கூடிய ஆங்கிலச் சொற்களையும் பதினடிப்படையில் [Base of 10] உள்ள எண்களையும் கொண்டு அமைந்துள்ளன. இம்மொழி களைக் கணிப்பொறியில் பயன்படுத்திக் கணக்குகளுக்கு விடை காண வேண்டுமெனில், முதலில வழியமைப்பைப் பொறி மொழிக்கு மாற்றத் தகுந்த மொழி மாற்றி [Compiler] தேவைப்

படுகிறது. உயர்நிலை மொழியிலிருந்து பொறிமொழிக்கு வழி அமைப்பு எவ்வாறு மாற்றம் செய்யப்படுகிறது என்பது படம் எண் 20—இல் விளக்கிக் காட்டப்பட்டுள்ளது.



படம்—20 மொழிமாற்றி செயற்படும் விதம்

கணிப்பொறியில் வழியமைப்பாளர்கள் பயன்படுத்தும் மொழியைப் பொதுவாக உயர்நிலை மொழி [Higher Level Language], இடைநிலை மொழி [Assembly Language], பொறி மொழி [Machine Language], என மூன்று பிரிவுகளாகப் பிரிக்கலாம். போர்ட்ரான், கோபால், பேஸிக் ஆகியவை உயர் நிலை மொழிகளுக்கு எடுத்துக்காட்டுகளாகும். முன்பு குறிப் பிட்டதுபோல் கணிப்பொறி மொழிகள் இயற்கை மொழிகளைப் போன்று, இலக்கண வரையறைகளுக்குட்பட்டவை. வழியமைப் பாளர்கள் செய்யும் வழியமைப்புகளில் இருவிதமான பிழைகள் [Errors] ஏற்படுவதற்கு வாய்ப்புண்டு. ஒன்று வெளிப்பிழை

[Syntax Error]; மற்றொன்று உட்பிழை [Semantic Error] என்று அழைக்கப்படுகிறது. குறிப்பிட்ட மொழிக்கென அமைந்துள்ள இலக்கண முறையிலிருந்து தவறி வழியமைப்புச் செய்யின், அதன் செயற்பாட்டின் பொழுது பிழைச் செய்தி [Error Message] மூலம் வெளிப்பிழையும் [Syntax Error], அதன் தன்மையும் நமக்குக் கணிப்பொறியால் உணர்த்தப்படும். ஆனால் உட்பிழையைக் [Semantic Error] கணிப்பொறியால் தானாகவே கண்டறிதல் இயலாததாகும். இந்த உட்பிழை, வெளிப்பிழைகளைக் கீழ்க்கண்ட எடுத்துக்காட்டு மூலமாக விளக்கிக் கூறலாம்.

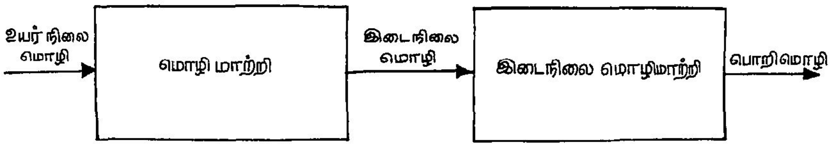
1. ''இராமன் பள்ளிக்கூடம் சென்றான்''.

   இவ்வாக்கியம் இலக்கண வரையறைப்படி சரியாக அமைந்திருக்கிறது. மேலும் உட்பிழையோ வெளிப்பிழையோ இதில் கிடையாது. ஆனால்,

   ''பள்ளிக்கூடம் இராமனுக்குச் சென்றது'' இந்த வாக்கியம் இலக்கணப்படி சரியாக அமைந்திருந்த போதிலும், உட்பொருளில் தவறு உள்ளது எனவே மேற் குறிப்பிடப்பட்ட வாக்கியத்தில்உட்பிழை அதாவது பொருட்பிழை இருப்பதாகக் கொள்ளலாம். இது போன்ற பிழைகளை மொழிமாற்றியால் கண்டுபிடித்தல் இயலாது. ஏனெனில், மொழிமாற்றிகள் வழியமைப்புகளில் நிகழும் வெளிப்பிழையைக் கண்டறிவதற்கேற்ப வடிவமைக்கப்பட்டுள்ளன.

வழியமைப்பைச் செயற்படுத்தும்பொழுது, வழியமைப்பாளர்கள் கொடுக்கும் உயர்நிலை மொழியில் அமைந்த வழியமைப்பை மொழிமாற்றி நேரடியாகப் பொறி மொழிக்கு மாற்றம் செய்வது இல்லை. அதற்கு மாறாக, இச்செயல் இரண்டு நிலைகளில் நடைபெறுகிறது. முதலில் உயர்நிலை மொழியிலுள்ள வழியமைப்பு 'கம்பைலர்' என்ற மொழிமாற்றியால் இடைநிலை மொழிக்கு மொழிமாற்றம் செய்யப்படுகிறது. பின்னர் அவ்வாறு மொழிமாற்றம் செய்யப்பட்ட வழியமைப்பு, இடைநிலை மொழிமாற்றி [Assembler] மூலம் இறுதியாகப் பொறி மொழிக்கு மாற்றம் செய்யப்படுகிறது. இம்

மொழி மாற்றம் பற்றிய விளக்கம் கட்டப் படம் எண் 21—இல் குறிப்பிடப்பட்டுள்ளது.



படம்—21 மொழி மாற்றிகள் [Translators/Compilers]

உயர்நிலை மொழியில் வழியமைப்பு செய்தல் சுலபமான பணியாகும். ஆனால் பொறிமொழியிலோ, இடைநிலை மொழியிலோ வழியமைப்பு செய்தல் சற்றுக் கடினமான பணி ஆகும். ஆனால் இடைநிலை மொழி மற்றும் பொறி-மொழியில் செய்யப்படும் வழியமைப்புகள், கணிப்பொறியில் விரைவாகவும் குறைந்த நினைவகத் தேவையுடனும் செயற்படும் தன்மை உடையன. எனவே சில குறிப்பிட்ட பயன்பாடுகளுக்குத் தேவையான வழியமைப்புகள் மட்டுமே, இடைநிலை மொழியில் செய்யப் பெறுகின்றன.

4.5. உயர்நிலை மொழிகள் (HIGHER LEVEL LANGUAGES)

வழியமைப்புகளை உருவாக்கப் பயன்படும் கணிப்பொறி மொழிகளை உயர்நிலை மொழி, இடைநிலை மொழி, பொறி மொழி என மூன்று பிரிவுகளாகப் பிரிக்கலாமென்பதை முன்பே கண்டோம். உயர்நிலை மொழிகள் இயற்கை மொழிகளை

ஒத்த இலக்கண அமைப்பைப் பெற்றுள்ளமையால், இவற்றைக் கற்றுக்கொண்டு பயன்பாட்டின் தன்மைக்கேற்ப வழியமைப்பு களை உருவாக்குதல் சாத்தியமாகிறது. கணிப்பொறியை விவர ஆய்வனை மற்றும் அறிவியல் முதலிய துறைகளில் பயன்படுத்தும் பொழுது, இடைநிலை மொழிகளைவிட உயர்நிலை மொழி மிகவும் பயனுள்ளதாக அமைகிறது.

உயர்நிலை மொழிகளில் முக்கியமான மொழிகளான பேஸிக் [Basic], கோபால் [Cobol], போர்ட்ரான் [Fortran] ஆகிய மொழிகளைப் பற்றி இங்குச் சிறிது காண்போம்.

4.5.1. பேஸிக் மொழி (*BASIC LANGUAGE*)

பேஸிக் [BASIC-Beginners All Purpose Symbolic Instruction Code] மொழி, சுமார் 1964—ஆம் ஆண்டு அமெரிக்காவிலுள்ள டார்ட் மௌத் [Dartmouth] கல்லூரியில் உருவாக்கப் பட்டது. அதன்பின்னர் பலவிதமான மாற்றங்களும் முன்னேற்றங் களும் இம்மொழியில் செய்யப்பட்டுள்ளன. இம்மொழி, முதன் முதலில் கணிப்பொறி மொழியைக் கற்பவர்க்கு ஏற்ற சிறந்த கருவியாகும். இது கற்பதற்கு மிகவும் எளிது. இதில் எழுதப் பெறும் வழியமைப்புகள் புரிந்து கொள்ளக்கூடிய இயற்கை மொழியைப் போன்றே அமையும். மேலும் இம்மொழி அறிவியல், பொறியியல், தொழில் முதலான துறைகளில் கணக்குகளைத் தீர்ப்பதற்கேற்ற வழியமைப்பினைச் செய்யப் பெரிதும் ஏற்றது. இம்மொழியில் நேரடித் தொடர்பு வசதி முறை [Interactive Facility], நேரப் பகிர்வுச் செயல்பாட்டு முறை [Time Sharing Facility] முதலிய பல வசதிகள் உள்ளன.

இம்மொழி, செய்திக்கூறுக் கோவைகளைக் [Data files] கையாளும் வசதிகள் வாய்க்கப் பெற்றுள்ளமையால் தொழிற் கூடங்களில் சம்பளப் பட்டியல் தயாரித்தல், நிதி நிலையாளுகை, பொருள் நிலையாளுகை, உற்பத்தித் திட்டமிடல் முதலான பணிகளுக்கு வழியமைப்புகளை உருவாக்க இயலும்.

பொதுவாக பேஸிக் மொழியின் அமைப்பைக்கீழ்க்கண்ட வாறு பிரித்துக் கூறலாம்.

அ. செயலில்லா ஆணைகள்: [Non Executable Statements] இந்த ஆணை வழியமைப்பு செயற்படும்பொழுது, கணக்கின் தீர்வில் எந்தவித மாற்றத்தையும் ஏற்படுத்துவதில்லை. எனவே வழியமைப்பில் இந்தஆணைசெயற்படுத்தப்படுவதில்லை.

எ. கா :

```
100 REM BASIC IS A HIGHER LEVEL LANGUAGE
```

ஆ. செயலாணைகள் [Executable Statements]

ஒரு வழியமைப்பு செயற்படும் பொழுது செயலாணைகள் ஒரு கணக்கைத் தீர்வு காணும் காரணிகளாக அமைகின்றன. செயலாணைகளைக் கீழ்க்கண்டவாறு பிரிக்கலாம்.

1. தன்மை ஆணைகள் [Type Statements]
2. கட்டுப்பாட்டாணைகள் [Control Statements]
3. செய்வளை ஆணைகள் [Loop Statements]
4. பிறவகை ஆணைகள் [Other Statements]

பேஸிக் மொழியில் அமைந்த ஒரு வழியமைப்பு எடுத்துக் காட்டாகக் கீழே தரப்பட்டுள்ளது.

எ. கா :

```
10 REM TO FIND THE CLASS AVERAGE MARKS

20 LET TOTAL=0

30 FOR I=1 TO 25

40 INPUT "TYPE THE INDIVIDUAL'S MARK:";MARK

50 LET TOTAL=TOTAL+MARK

60 NEXT I

70 PRINT "CLASS AVERAGE MARK";"=";TOTAL/25

80 STOP

90 END
```

இ. இடுவரல்—விடுவரல் மற்றும் கோவை ஆணைகளைக் கீழ்க்கண்டவாறு பிரிக்கலாம்.

1. இடுவரல், விடுவரல் ஆணைகள் [Input/Output Statements]

2. கோவை ஆணைகள் [Disk Based I/O Statements]

3. கூறுகள் [Functions]

4. கட்டுப்பாட்டாணைகள் [Control Statements]

பேஸிக் மொழியின் மேற்கண்ட பிரிவுகளையும் அதில் வழங்கப் பெறும் ஆணைகளையும் பற்றி, பாகம் எண் 8,9—இல் விளக்க மாகக் காணலாம்.

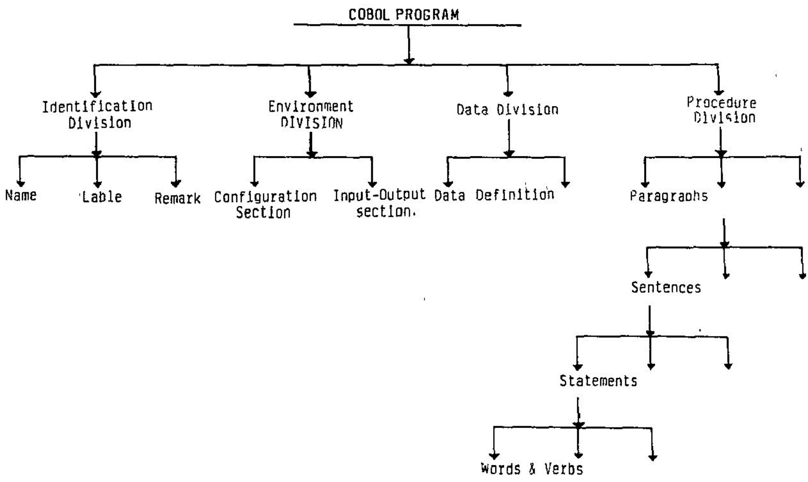
4.5.2. கோபால் மொழி (COBOL LANGUAGE)

கோபால் என்பது [COBOL-Common Business Oriented Language] என்பதின் சுருக்கமேயாகும். இம்மொழி 1959—ஆம் ஆண்டு பல அறிஞர்களின் ஆய்வின் பலனாக உருவாக்கப்பட்டு, கணிப்பொறியில் பயன்பாட்டிற்கு வந்தது. உலகம் முழுவதும் தொழில் துறையில் பெருமளவு பயன்படுத்தப்படும் ஓர் உயர்நிலை மொழி கோபால் ஆகும். இந்த மொழியில் எழுதப்பெறும் வழி யமைப்புகள் யாவும் நன்கு படித்தறியக் கூடிய விதத்தில் அமைந் திருக்கும். ஆனால், இதில் எழுதப்பெறும் வழியமைப்புகளைச் செயற்படுத்த அதிக அளவு நினைவகம் தேவைப்படுகிறது. நாள் தோறும் நிகழ்ந்துவரும் ஆராய்ச்சிகளின் பலனாக இம்மொழியில் பற்பல முன்னேற்றங்களும் [Improvements], புதிய வழியமைப்பு ஆணைகளும் சேர்க்கப்பட்டுள்ளன. எனவே தற்காலக் கணிப் பொறிகளில் பயன்படுத்த, திறமைவாய்ந்த ஒரு வணிகத்துறை மொழியாக [Commercial Language] இது விளங்குகிறது. பொது வாக இந்த மொழியில் எழுதப்பெறும் வழியமைப்பைக் கீழ்க் கண்டவாறு நான்கு பெரும் பிரிவுகளாகப் பிரிக்கலாம்.

1. அறிமுகப் பிரிவு [Identification Division]
2. சூழ்நிலைப் பிரிவு [Environment Division]
3. செய்திக் கூறுப்பிரிவு [Data Division]
4. வழிவகைப் பிரிவு [Procedure Division]

அறிமுகப் பிரிவில் வழியமைப்பு எழுதப்பட்ட தேதி, அதன் பெயர், வழியமைப்பாளர் மற்றும் நிறுவனத்தின் பெயர் மற்றும் சில தனிப்பெயர்கள் குறிப்பிடப்படுகின்றன. வழியமைப்பை மொழிமாற்றம் செய்யவும் பின்னர் செயற்படுத்தவும் பயன்படுத்தப்பட்ட கணிப்பொறி பற்றிய விவரங்களும், வழியமைப்பில் பயன்படும் கோவைகள் [Files] பற்றிய குறிப்புகளும் சூழ்நிலைப் பகுதியில் விளக்கப்படுகின்றன. செய்திக்கூறுப் பிரிவில் செய்திக் கூறுப் பதிப்புகளின் [Data records] அமைப்புகள், அவற்றில் பதிவு ஆகியுள்ள செய்திக்கூறுகள், அவற்றின் தன்மை, கோவைகளில் செய்திக்கூறுகள் பதிவுசெய்யப்பட்டுள்ளவிதம் முதலிய செய்திகள் முறைப்படி தரப்பட்டுள்ளன. வழிவகைப்பிரிவில் ஒரு கணக்கைத் தீர்வு காண்பதற்குத் தேவையான செய்வழி விளக்கப்பட்டுள்ளது.

கோபால் மொழியில் அமைந்துள்ள வழியமைப்பைக் கீழ்க் கண்டவாறு படம் 22 மூலம் விளக்கலாம்.



படம்—22 கோபால் வழியமைப்பின் படிநிலையமைப்பு

தற்காலத்தில் தொழிற்கூடங்களில் கணிப்பொறி மூலம் சம்பளப்பட்டியல் தயாரித்தல் [Payroll Preparation], நிதி நிலைக் கணக்கு [Financial Accounting], பொருள் நிலையாளுகை, உற்பத்தித் திட்டமிடல் முதலிய பல துறைப் பணிகளில் வழியமைப்பு செய்வதற்கு இம்மொழி பெரும்பாலும் பயன்படுத்தப் படுகிறது. இம்மொழியில் சில குறிப்பிட்ட அறிவியற் செயற்பாடுகளைச் [Scientific Operations] செய்ய இயலாது. ஏனெனில் வணிகத் துறைக்கென்றே இம்மொழி உருவாக்கப்பட்டுள்ளது.

4.5.3. போர்ட்ரான் மொழி (FORTRAN LANGUAGE)

போர்ட்ரான் [Fortran] என்பது [Formula translation] என்பதன் சுருக்கமாகும். இந்த உயர்நிலை மொழி ஐ. பி. எம். [IBM] நிறுவனத்தினரால் 1954-ஆம் ஆண்டு உருவாக்கப்பட்டு வெளியிடப்பட்டது. அதன் பின்னர் பல தட்வைகள் பயன் பாட்டின் தேவைக்கேற்பத் திருத்தப்பட்டும் மாற்றங்கள் பல செய்யப்பட்டும் உள்ளது. உலக அரங்கிலுள்ள கணிப்பொறி மொழியின் தரம் உயர்த்தும் நிறுவனமான ANSI — American National Standard Institution இவ்வாறு போர்ட்ரான் மொழியில் செய்த பல சீர்திருத்தங்கள் பயனாக ஆன்சி போர்ட்ரான்—66 [ANSI FORTRAN-66], ஆன்சி போர்ட்ரான்—77 [ANSI FORTRAN-77] ஆகிய உருக்கள் [Versions] தோன்றியுள்ளன. இந்த மொழி அறிவியல் ஆராய்ச்சிக் கணக்குகளைக் கணிப்பொறி மூலம் தீர்வு காண்பதற்கென்றே உருவாக்கப்பட்ட மொழி ஆகும். சமன்பாடுகளைத் [Equations] தீர்வு காண்பதற்கும் சூத்திரங்களைக் [formulae] கணக்கிடுவதற்கும் இம்மொழி பெருமளவு பயன்படுத்தப்படுகிறது. இன்றும் பெரும்பாலான பல்கலைக் கழகங்கள், கல்லூரிகளில் மாணாக்கர்களுக்குக் கணிப்பொறி மொழி பயிற்றுவித்தல், ஆராய்ச்சி முதலிய பணிகளுக்குப் பெரிதும் பயன்படுத்தப்படும் ஒரு மொழியாக இது விளங்குகிறது.

இம்மொழி அறிவியற் கணிப்பிற்கென உருவாக்கப்பட்ட மொழியாதலால் தொழிற்கூடங்களில் விவர ஆய்வனை செய்வதற்குப் பேஸிக் அல்லது கோபால் மொழியைப் போன்று உகந்ததன்று. மேலும் இதில் வழியமைப்புச் செய்தல் பேஸிக், மற்றும் கோபால் மொழியில் வழியமைப்புச் செய்தலைவிடச் சற்றுக் கடினமாகும்.

போர்ட்ரானிலுள்ள வழியமைப்பு ஆணைகளைக் கீழ்க் கண்டவாறு பிரிக்கலாம்.

1. இடுவரல், விடுவரல் ஆணைகள் [Input & Output Instructions]
2. தன்மை ஆணைகள் [Type Statements]
3. எண் கணக்காணைகள் [Arithmetic Statements]
4. படிவ ஆணைகள் [Format Statements]
5. வழிமாற்றி ஆணைகள் [Branch Instructions]
6. செய்வளை ஆணைகள் [Do Statements]
7. துணை வழியமைப்புகள் [Sub-Programs]
8. வழியமைப்புக் கூறுகள் [Functions]

மேற் குறிப்பிட்ட ஆணைகளைப் போர்ட்ரானில் வழியமைப்புச் செய்யும்பொழுது முறைப்படி பயன்படுத்துதல் அவசியமாகும்.

4.6. வழியமைப்பு முறை நிருணயம் (PROGRAM CONSTRUCTION)

இதுவரை கணிப்பொறி மொழிகளைப் பற்றிக் கூறினோம். இப்பொழுது எப்படி வழியமைப்பு முறையை நிருணயிப்பது என்பது பற்றி அறிவோம். வழியமைப்பாளர் இரண்டு முக்கியமான கருத்துகளைக் கவனத்திலிருத்த வேண்டும். ஒன்று வழியமைப்பு முறையின் செயல்திறன், எல்லைகள் கட்டுப்பாடுகள் மற்றும் பயன்பாட்டின் தன்மை [Type of application], இரண்டாவது பயன்படுத்தும் கணிப்பொறியின் கருவிய அமைப்பு அதிலுள்ள வசதிகள், கட்டுப்பாடுகள், வழியமைப்பிற்கான கணிம வசதிகள் [Software facility] முதலியன ஆகும்.

வழியமைப்பு முறையில் கவனிக்கவேண்டிய படிகள்

1. முறையமை வரையறுப்புகள் [Logical Definitions]

2. அவைகளைப் பதிவு செய்தல் [Recording]

3. வழியமைப்பு ஆணைகளை மொழி ஆய்வு செய்தல் [Encoding the Instructions]

4. சரிபார்த்தல் [Verification]

5. மொழிமாற்றம் செய்தல் [Compilation]

6. செயற்படுத்துதலும், நடைமுறையாக்கலும் [Execution & Implementation]

4.6.1. செய்வழி வரையறுப்பு (*ALGORITHM DEFINITION*)

கணிப்பொறி மூலம் ஒரு கணக்குக்கு [Problem] விடை காண முற்படும் பொழுது, நாம் பலவிதமான நடைமுறைகளைப் பின்பற்ற வேண்டியுள்ளது. வழியமைப்பாளர் கணக்கை நன்கு ஆய்ந்து படித்துணர்தல், அக்கணக்குக்கேற்ப முறையமை வரையறுப்புகளை மனத்திற்குள்ளேயே உருவாக்கல், முதலியவை ஒரு கணக்குக்குத் தீர்வு காணத் தேவையான முன்னேற்பாடுகளாகும்.

வழியமைப்புச் செய்வதற்குமுன் [Program Coding] கணக்குக்குத் தீர்வுகாணப் படிப்படியாகவும் தொடர்ச்சியாகவும் செய்யக்கூடிய வழிமுறைகளை வகுத்துக்கொள்வதே செய்வழி என்று வரையறுத்துக்கூறலாம். அவ்வாறு செய்வழியை வகுத்துக் கொண்டபின், அதன் துணை கொண்டு ஏதாவதொரு கணிப்பொறி மொழியில் ஆணைகளை முறையாகத் தொகுத்து வழியமைப்பை உருவாக்கலாம். மேலும், இந்தச் செய்வழியானது வழியமைப்பை உருவாக்கிய பிறகு அதைப்பற்றிய தகவல்களைப் பெறவும் பயன்படுகிறது. செய்வழியை எங்ஙனம் விளக்குவது என்பதைக் கீழ்க்கண்ட எடுத்துக்காட்டில் காணலாம்.

வினா: ஒரு வகுப்பில் நான்கு பாடங்களைப் பயிலும் 20 மாணவர்களின் தேர்வு மதிப்பெண்கள் தரப்பட்டுள்ளன. கணிப்பொறியில் இதைக் கணக்கிட்டு, அவர்களுடைய சராசரி மதிப்பெண்களைக் கொண்டு அவர்களில் தேறிய அல்லது தவறிய மாணவர்களை எங்ஙனம் கண்டறியலாம்?

மேற்கண்ட வினாவுக்கு வழியமைப்பு செய்யுமுன், கீழ்க் கண்டவாறு செய்வழியை அமைக்கவேண்டும்.

1. மாறிலிகளை ஆரம்ப நிலைப்படுத்துக [Initialise the Variables].

2. துளையட்டையிலுள்ள முதல் மாணவனின் கணிதம், இயற்பியல், வேதியியல், ஆங்கிலம் ஆகியவற்றின் மதிப் பெண்களைப் படித்தறிக.

3. எண்பானுடன் [Counter] ஒன்றைக் கூட்டுக.

4. சராசரியைக் கணக்கிடுக.

5. சராசரி **மதிப்பெண்ணை 45 உடன்** ஒப்பு நோக்கி, சராசரி மதிப்பெண் 45-இற்குக் குறைந்த மதிப்பெண் பெற்றவர் தேர்வில் தவறினாரென்றும், 45 மதிப்பெண்கள் அல்லது அதற்கு மேல் பெற்ற மாணவர் தேறினாரென்றும் அறிவிக்க.

6. எண்பானின் மதிப்பை 20—உடன் ஒப்புநோக்குக. எண் பானின் மதிப்பு 20—ஐ விட அதிகமென்றால் பணியை நிறுத்திவிடுக. இல்லையெனில் படி 2—இலிருந்து பணி யைத் தொடர்க.

மேற்கண்ட செய்வழியைப் படிப்படியாகக் கணிப்பொறி செய்யும் பொழுது, நமது கணக்குக்கேற்ற விடையைப் பெற முடிகிறது. எந்த ஒரு வழியமைப்பாளரும் வழியமைப்பை வடிவமைக்கும் பணியைத் [Program Design] தொடருமுன், அந்தக் கணக்குபற்றிய தெளிவான பொருளைப் புரிந்துகொண்டு, அதற்கேற்றவகையில் செய்வழியை அமைத்தல் இன்றியமையாத தாகும்.

4.6.2. வழியமைப்புக் கட்டப் படம் (PROGRAMMING FLOW CHART)

வழியமைப்பு செய்வதற்கு முன்பாக, நாம் செய்வழியை உருவாக்கிக் கொள்கிறோம். படம் மூலம் ஒரு செய்தியை விவரித்தல் ஆயிரம் வார்த்தைகளைக் கொண்டு விவரித்தலுக்குச் சமமாகும். மேலும் படம் மூலம் நமது எண்ணங்களையோ, மற்ற

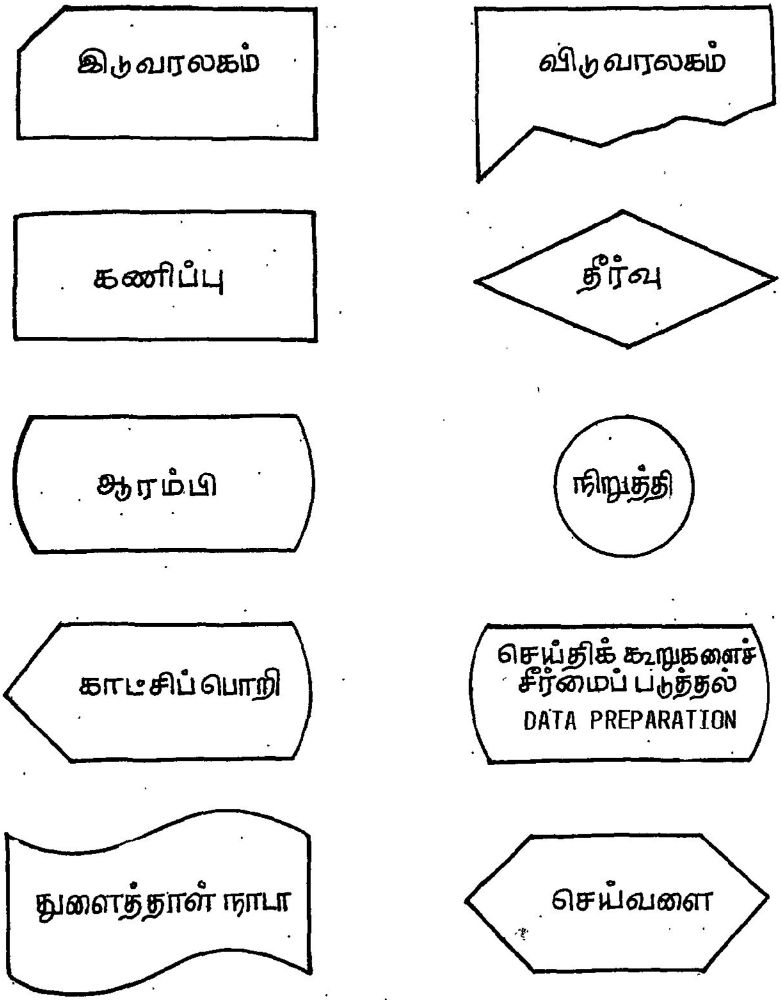
விவரங்களையோ குறிப்பிடுதல் மிகவும் எளிதாகும். படம் மூலம் குறிப்பிடும் பொழுது புரிந்து கொள்ளுதல் மிகவும் சுலபமாகிறது.

ஒவ்வொருதுறையிலும்பலஅறிஞர்கள் தமதுஎண்ணங்களைத் தெளிவாக வெளிப்படுத்துவதற்கும், சில அவசியமான நுட்பம் நிறைந்த அறிவியல் கண்டுபிடிப்புகளைப் புரிந்துகொள்வதற்கும் கணிதவியலார் வான்நியூமன் [Von Neumann] காலத்திலிருந்தே வரைபடங்களைப் பயன்படுத்தியுள்ளனர். ஒரு கணக்கின் அல்லது சிக்கலின் தர்க்க ரீதியான செய்வழியமைப்பையும் அதனிடையே உள்ள தொடர்புகளையும் வரை படம் மூலம் குறிப்பிடும் பொழுது சுலபமாகவும் மிகத்தெளிவாகவும் புரிந்து கொள்வதற்குப் பெரிதும் வாய்ப்பு ஏற்படுகிறது. கணிப்பொறி யின் செய்வழியைக் கட்டப் படம் மூலம் தெரிவிப்பதற்குப் பல நியதிகள் இருக்கின்றன. அவற்றைக் கீழ்க்கண்டவாறு குறிப்பிடலாம்.

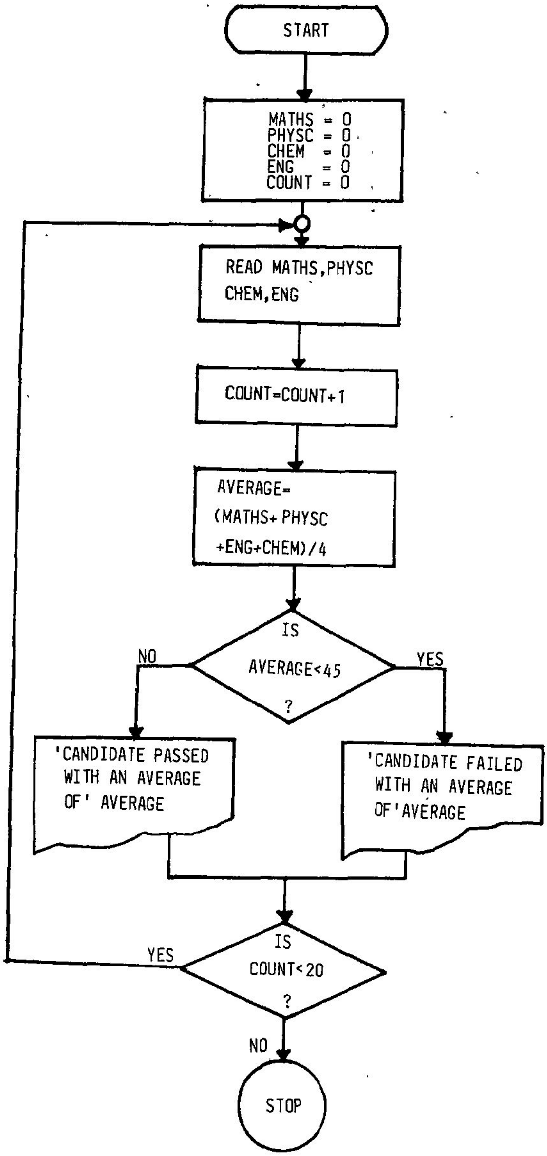
கணிக்கும் செயல் செவ்வக வடிவத்தாலும் தீர்வு காணல் வைர அமைப்பையுடைய [Diamond Shape] படங்களாலும் இடுவரல், விடுவரல் ஆகிய செயல்கள் அதைச் சார்ந்த இடுவரல், விடுவரல் ஆகிய அகங்களையொத்தஅமைப்பினாலும் குறிப்பிடப் படுகின்றன. படம் எண் 23—இல் இவைபற்றிய வினைக்கட்ட வரைவுகளும் படம்எண் 24—இல் முன்பு விவரிக்கப்பட்ட கணக்குக் கேற்ற செய்வழியைக் குறிப்பிடுவதற்கான வரைபடமும் விவரிக்கப்பட்டுள்ளன.

4.6.3. முறையமை வரையறுப்புகள் (LOGICAL DEFINITIONS)

வழியமைப்பாளர் திறம்பட வழியமைப்பதற்கான பல புதிய உத்திகளைக் கண்டறிந்து பயன்படுத்துதல் இன்றியமையாத ஒன்றாகும். ''ஒரு கணக்கைச் செய்வதற்குச் சரியான முறையில் தொடங்கி, தெளிவானவகையில் செய்துமுடிக்கும் வழிமுறையை வரையறுத்துக் கூறுதலே முறையமை வரையறுப்பாகும்''. முறையமைவரையறுப்பு சரியாக அமையாவிடில்,கணக்குக்கேற்ற சரியான விடை கிடைக்காததுடன் பொருளும், நேரமும் வீணாகும். வழியமைப்பாளர் தமது அனுபவத்தால் ஒரு கணக்கினை நன்கு புரிந்து கொண்டு, அதற்கேற்ற திறமையான வழியில் முறையமை வரையறுத்தலை அமைத்தல் இன்றி யமையாப் பணியாகும்.



படம்—23 வினைக் கட்டவரைவு [Block Diagram]



படம்—24

வழியமைப்புச் செய்வழி வரைவு [Programming Flow Chart]

4.6.4. பதிவு செய்தலும் மொழிஆய்வு செய்தலும்

முறையமை வரையறுப்புகள் மூலம் நிருணயம் செய்யப் பட்ட செயல்களை வரிசைப்படுத்துதல் வேண்டும் [Sequencing]. பிறகு இதைச் செய்வழி வரைவு மூலம் [Flow Chart] குறிப்பிட வேண்டும். இதில் தவறு ஏதேனும் இருப்பின், சரியான முறையில் சோதனை செய்து திருத்திக் கொள்ளலாம். இவ்வாறு வழிமுறை நிருணயம் செய்யப்பட்ட கணக்கின் விவரங்கள் குறிப்பிட்ட வழியமைப்பு மொழியில் வழியமைப்பாக வழியமைப் பாளர்களால் எழுதப்படுகிறது [Program Coding]

4.6.5. சரிபார்த்தலும் மொழிமாற்றம் செய்தலும் *(VERIFICATION AND TRANSLATION COMPILATION)*

கணிப்பொறியில் வழியமைப்பை இடுவரல் செய்வதற்கு முன்பே வழியமைப்பில் பிழையில்லாமல் சரிபார்த்தல் அவசியம். ஏனெனில்,இவை போன்ற பிழைகள் கணிப்பொறியின் நேரத்தை வீணாக்கிவிடும். இவ்வாறு தயார் செய்யப்பட்ட வழியமைப்புக் கணிப்பொறி அட்டை [Computer Punched Cards], அல்லது காட்சித்திரை [Display Unit] ஆகிய இடுவரல் அகங்கள் மூலம் கணிப்பொறிக்கு இடுவரல் செய்யப்படுகிறது. கணிப்பொறியின் நினைவகத்தில் நாம் எந்த மொழியில் வழியமைப்பைச் செய்து உள்ளோமோ அதற்கேற்ற மொழிமாற்றி முன்பே பதிவு செய்யப் பட்டுள்ளது. நமது வழியமைப்பு மொழிமாற்றிக்கு ஒரு செய்திக் கூறாக அமைகிறது. இவ்வாறு வழியமைப்பு ஆணைகளை மொழிபெயர்க்கும் பொழுது அதில் தவறு ஏதேனும் இருக்கு மாயின், பிழைச் செய்தி மூலம் கணிப்பொறி, வழியமைப் பாளருக்கு அறிவிக்கிறது. இதற்குப் பிழையுணர்தல் [Diagnostic] என்று பெயர் தவறு ஏதும் இல்லாவிடில், வழியமைப்பானது பொறி மொழிக்கு மொழிமாற்றம் செய்யப்படுகிறது.

4.6.6. சோதித்தலும் தவறுகள் திருத்தலும்

குறிப்பிட்ட கணக்குக்கு மாதிரிச் செய்திக் கூறுகளைக் [Sample Data] கொண்டு விடைகளைப் பெற்று, அதை நாமாகவே சோதித்து முடிவு [Results] சரியாக உள்ளதா என்று அறியவேண்டும். வழியமைப்பைச் செயற்படுத்தும் பொழுது

வெளிப்பிழை அல்லது இலக்கணப்பிழை, உட்பிழை என்ற இருவகைப் பிழைகள் நிகழலாமென முன்பே பார்த்தோம். வெளிப்பிழையைக் கணிப்பொறி கண்டறிந்து பிழைச் செய்தி மூலம் தெரிவிக்குமாதலால் அதை எளிதில் திருத்த இயலும். எனவே மாதிரிச் செய்தி கூறுகளைக் கொண்டு பெறும் முடிவு சரியாக இல்லாவிடில், வழியமைப்பில் உட்பிழை இருப்பதாக முடிவு செய்யலாம். உடனே வழியமைப்பை ஆராய்ந்து தவறைத் திருத்தி மீண்டும் மொழிமாற்றம் செய்யவேண்டும். இவ்வாறு மொழிமாற்றி எடுத்துக்காட்டிய தவறுகளைத் திருத்தும் முறைக்குப் பிழை திருத்தல் [Debugging] என்று பெயர்.

4.6.7. செயற்பாடு (EXECUTION)

மேற்கண்ட முறைப்படி, பிழைகள் திருத்தப்பட்ட வழியமைப்புத் துணை நினைவகமாகிய மின்காந்த நாடாவிலோ [Magnetic Tape], மின்காந்தத் தட்டிலோ [Magnetic Disk] பதிவு செய்து வைக்கப்படுகிறது. பின்னர் நமக்கு விடை தேவைப்படும் பொழுது செய்திக் கூறுகளைக் கணிப்பொறிக்கு ஊட்டி, விடையைப் பெறலாம். நமது தேவைக்கேற்ப விடை வெண்தாளில் அச்சுப் பொறி மூலம் [Printer] திறப்படிவமாகவோ [Hard Copy] காட்சித் திரையிலோ [Display Screen] கிடைக்கிறது.

கணிப்பொறி ஒருங்கும்
பேஸிக் மொழியும்

பாகம் – 5

5. கருவியம் (HARDWARE)

5.1. முன்னுரை

இதற்கு முன்புள்ள பகுதிகளில் கணிமம் என்றால் என்ன? என்பதையும் அது கணிப்பொறி செயல்படுவதற்கு எங்ஙனம் மிக்க அவசியமாகிறது என்பது பற்றியும் கண்டறிந்தோம். அதே போன்று, நமது கணக்குக்கு வழியமைப்பு மூலம் கணிப்பொறியில் விடை காண்பதற்குக் கருவியம் [Hardware] மிகவும் அவசியமாகிறது. கருவியம் என்பது மின் நுண் கருவிகளைக் [Electronic Components & Circuits] கொண்டு உருவாக்கப்பட்ட மையச் செயலகம், இடுவரல், விடுவரல் ஒருங்குகள், நினைவகம், ஆளகம் ஆகியவற்றின் தொகுப்பேயாகும். புதிய ஆராய்ச்சிகளின் பலனாக முந்தைய க்ணிப்பொறிகளைக் காட்டிலும் தற்காலக் கணிப்பொறிகள் அளவில் மிகவும் சிறியனவாகவும் ஆனால் அதே சமயம் பன்மடங்கு செயல் வேகமும் பயன்பாட்டுத் திறனும் பெற்றனவாகவும் விளங்குகின்றன. நவீன ஆராய்ச்சிகளின் பலனாக, நாளுக்கு நாள் கருவியத்தின் விலை குறைந்து கொண்டு வருவதைக் காண்கிறோம். கருவியத்தின் பல்வேறு அங்கங்களைப் பற்றியும் அவற்றின் செயற்பாட்டைப் பற்றியும் இந்தப் பகுதியில் ஆராய்ந்தறியலாம்.

5.2. கருவியத்தின் அங்கங்கள்

கருவியத்தின் முக்கிய அங்கங்களாவன

1. செயலகம் [Processor]
2. இயக்கு நிலையகம் [Console]

3. நினைவகம் [Memory]

4. ஆளகம் [Control Unit]

5. இடுவரல் விடுவரல்—அகங்கள் [Input Output Systems]

கீழ்க்கண்ட கருவியப் பாகங்களை இடுவரல் விடுவரல் அகங்களுக்கு எடுத்துக்காட்டுகளாகக் கூறலாம்.

அ. அட்டைத் துளைப்பொறி [Card Punching Machine]

ஆ. அட்டைப் படிப்பொறி [Card Reader]

இ. காந்த நாடாப்பொறி [Magnetic Tape Unit]

ஈ. காந்தத் தட்டுப்பொறி [Magnetic Disc Unit]

உ, காட்சிப்பொறி [Display Unit]

ஆளகம்,அட்டைத்துளைப் பொறி, அட்டைப் படிப்பொறி முதலியவற்றைப் பற்றிச் சுருக்கமாக முன்பே கவனித்தோம். நினைவகம், செயலகம், இடுவரல், விடுவரல் ஒருங்குகளான மின்காந்த நாடா, மின்காந்தத் தட்டு, காட்சித்திரை ஆகியவை பற்றி இங்குச் சிறிது விளக்கமாக ஆராய்வோம்.

5.2.1. செயலகம் (PROCESSOR)

மையச் செயலகத்தின் இதயமாக விளங்குவது செயலகமாகும் [Processor] இந்தச் செயலகமே பொறி மொழி வடிவிலுள்ள ஆணைகளையும் [Commands] வழியமைப்புகளையும் செய்திக் கூறுகளையும் அறிந்துணர்ந்து அதற்கேற்பச் செயற்படுகிறது. சேர்ப்பகத்தையொத்த [Accumulator] பல பதிவகங்கள் [Registers] செயலகத்தில் உண்டு. இப்பதிவகங்கள் எண்களைக் கூட்டுதல், பெருக்குதல், கழித்தல், வகுத்தல் ஆகிய கணக்கியல் செயல்களைச் [Arithmetic Operations] செய்யவும் இடைநிலை விடையைச் சேமிக்கவும் செயற்பாடு முடிந்தபின் விடையைப் பகுப்பாய்வு [Analysis] செய்யவும் பயன்படுகின்றன. இந்தச் செயலகத்தால் மேற்குறிப்பிட்ட நான்கு செயற்பாடுகளைக் கொண்டு எல்லாவகைக் கணக்குகளையும் தீர்வு செய்யமுடியும். குறிப்பாகச் செயலகத்தில் உள்ள கணக்ககம் [Arithmetic Unit]

இப்பணியை மேற்கொள்கிறது. கருவிய அமைப்பு ரீதியில் இந்தக் கணக்ககம் பல கதவுகளைக் [Gates] கொண்டுள்ளது.

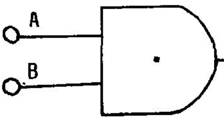
அவைகளாவன :

1. உம்மிணைக் கதவு (AND GATE)
2. அல்லதிணைக் கதவு (OR GATE)
3. எதிர்மறைக் கதவு (NOT GATE)

மேற்குறிப்பிட்ட கதவுகள் மூலம் மாறிகளில் [Variables] '1' அல்லது '0' என்ற மதிப்பைக் கொண்டு கதவுகளின் தன்மைக் கேற்ப விடைகள் கணித்துக் கொடுக்கப்படுகின்றன. கணக்ககத்தில் பல்லாயிரக்கணக்கான மின்னிணைப்புகள் வடிவமைக்கப்பட்ட [Design] முறையில் இணைப்புச் செய்யப்பட்டிருக்கும். இதன் மூலமாகவே மையச் செயலகத்தில் கணக்கிடும் செயல் நடைபெறுகிறது. மேற்குறிப்பிடப்பட்ட கதவுகளின் கட்டப்படமும் அவற்றின் செயற்பாட்டு விளக்க அட்டவணையும் கீழே கொடுக்கப் பட்டுள்ளன.

5.2.1.1. உம்மிணைக் கதவு (*AND GATE*)

படம்–25 படம்–26

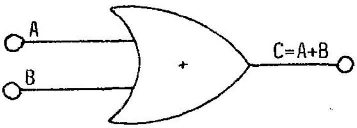


இடுவரல் (INPUT)		விடுவரல் (OUTPUT)	
A	B	A.B	C
1	1	1.1	1
1	0	1.0	0
0	1	0.1	0
0	0	0.0	0

படம்–25 உம்மிணைக் கதவு [And Gate]
படம்–26 உம்மிணைக் கதவு செயற்பாட்டு அட்டவணை [AND Gate Boolean Table]

5.2.1.2 அல்லதிணைக் கதவு (OR GATE)

படம்—27



படம்—28

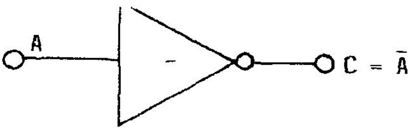
இடுவரல் (INPUT)		விடுவரல் (OUTPUT)	
A	B	A+B	C
1	1.	1+1	1
1	0	1+0	1
0	1	0+1	1
0	0	0+0	0

படம்—27 அல்ல திணைக் கதவு [OR Gate]

படம்—28 அல்லதிணைக் கதவு செயற்பாட்டு அட்டவணை [OR Gate Boolean Table]

5.2.1.3 எதிர்மறைக் கதவு (NOT GATE)

படம்—29



படம்—30

இடுவரல் INPUT	விடுவரல் OUTPUT
A	C
1	0
0	1

படம்—29 எதிர் மறைக் கதவு [NOT Gate]

படம்—30 எதிர்மறைக் கதவு செயற்பாட்டு அட்டவணை [NOT Gate Boolean Table]

5.2.2. இயக்கு நிலையகம் (CONSOLE)

கணிப்பொறியின் புற ஒருங்குகள், நினைவகம், கணக்ககம் முதலியவை செயற்படும்பொழுது அவற்றிற்கிடையே மிகுந்த வேகத்தில் செய்திப் பரிமாற்றங்கள் நிகழ்கின்றன. செய்திக்கூறு களையும் வழியமைப்புகளையும் பதிவு செய்தல் வழியமைப்பு களைச் செயற்படுத்துதல், கணித்தல், விடையைப் பதிவுசெய்தல், அல்லது அச்சுப்பொறியில் அச்சடித்தல் முதலான பல பணிகள் முறைப்படியாகவும் ஒழுங்குடனும் செயற்படுத்தப்படுகின்றன. இச்செயற்பாடுகள் அனைத்தையும் முறைபிசகாது கணிப்பொறி செய்து முடித்தற்கு மையச் செயலகம் உதவுகிறது. கருவியத்தின் உள்நிலைச் செயற்பாடுகள் மட்டுமே ஆளகத்தால் கட்டுப்படுத்தப் படுகின்றன. ஆனால், கணிப்பொறி வெளியுலகத்துடன் தொடர்பு கொள்ளவேண்டுமாயின் அதற்கென அதன் கருவியத்தில் வசதி கள் அமையப்பெற்றிருத்தல் மிகவும் அவசியமாகும். கணிப் பொறியில் அமைந்துள்ள இந்தக் கருவியப் பகுதியே இயக்கு நிலையகம் என வழங்கப்படும்.

கணிப்பொறியின் இடுவரல், விடுவரல் அகங்கள், மையச் செயலகம் [CPU] முதலியவை செயற்படும்பொழுது, கீழ்க் கண்டவாறு பலசெயற்பாடுகள் நிகழலாம்.

1. வழியமைப்பாளர் கொடுத்த வழியமைப்பில் பிழைகள் இருக்கலாம். இதன் காரணமாகப் பிழைச் செய்தி வெளிப்படுத்தப்படலாம்.

2. கணிப்பொறியில் ஏதேனும் செயற்பாடு தொடர்பான [Operational Problem] குறைகள் நிகழின் அவை குறித்த செய்திகள் தெரிவிக்கப்படலாம்.

3. கணிப்பொறியின் செயற்பாட்டுக்குத் தேவையான செய் திக்கூறுகள் அல்லது வழியமைப்புகள் அடங்கிய மின் காந்தத் தட்டு அல்லது பிற இடுவரல் ஒருங்குகளைக் கணிப்பொறியில் இணைக்குமாறு கோரப்படலாம்.

எ.கா: Mount or Dismount Magnetic Tapes

4. சில குறிப்பிட்ட செயல்களைக் கணிப்பொறி இயக்குபவர் [Computer Operator] செயற்படுத்தவும் கணிப்பொறியின் இயக்கத்தைக் கட்டுப்படுத்தவும் இயக்கு நிலையகம் பயன்படுகிறது.

இயக்கு நிலையகமானது முதலாம், இரண்டாம், மூன்றாம் காலக்கட்டக் கணிப்பொறிகளில் பெரும்பாலும் தொலை அச்சுத் தட்டுப் பலகை [Tele Type Key Board] ஆகவும் தற்காலத்திய கணிப்பொறிகளில் பெரும்பாலும் காட்சித் திரையாகவும் [Display Unit] விளங்குகின்றன. மேற்குறிப்பிடப்பட்ட செய்திகள் [Messages], தொலை அச்சுத்தட்டுப் பலகையில் உள்ள வெண்தாளில் அச்சு செய்யப்படுகிறது. அதே போன்று காட்சித் திரையில் இது வெளிப்படுத்தப்படுகிறது [Displayed]. இதை உணர்ந்து கணிப்பொறி இயக்குபவர் செயல்படுவார்.

இயக்கு நிலையகம் மூலம் கணிப்பொறியில் செயற்படும் வழியமைப்புகளையும் இதர பல செயல்களையும் திறம்படக் கண்காணிக்கவும் கட்டுப்படுத்தவும் இயலும். இதன் மூலமாகக் குறிப்பிட்ட இடுவரல் விடுவரல் ஒருங்குகளை இணைப்பறவு [Disconnect] செய்யவும் தேவையானபோது இணைத்துக் கொள்ளவும் சில குறிப்பிட்ட வழியமைப்புகளை முன்னுரிமை அடிப்படையில் செயற்படுத்தவும் இயக்கு நிலையகம் பயன்படுகிறது. இயக்கு நிலையகம் காட்சித் திரையாக இருக்குமெனில் இதன் மூலமாகச் சில வழியமைப்புகளை இடுவரல் செய்யவோ, செயற்படுத்தவோ முடியும்.

5.2.3. நினைவகம் (*Memory*)

கணிப்பொறியின் அமைப்பைப் பொறுத்து அவற்றைக் கீழ்க் கண்டவாறு பல வகைகளாகப் பிரிக்கலாம் என்பதை முன்பே பார்த்தோம்.

[i] உயர்நுட்ப நிலைக் கணிப்பொறி [Super Computer]

[ii] பெரிய கணிப்பொறி [Medium Range Computer]

[iii] சிறிய கணிப்பொறி [Mini Computer]

[iv] மிகச்சிறிய கணிப்பொறி [Micro Computer]

கணிப்பொறியின் வேகத்திறன், விலை, நினைவக அளவு, செயற்பாட்டுத்தன்மை ஆகியவற்றின் அடிப்படையிலேயே அவை மேற்கண்டவாறு பிரிக்கப்படுகின்றன. மற்ற அங்கங்களைக் காட்டிலும் கணிப்பொறியின் செயற்பாட்டிற்கு மிக அவசியமாக விளங்குவது நினைவகமாகும். நினைவகத்தைக் கீழ்க்கண்டவாறு மூன்று பிரிவுகளாகப் பிரிக்கலாம்.

அ. மிகைவேக நினைவகம் [Fast Access Memory]

ஆ. முதன்மை நினைவகம் [Primary Memory]

இ. துணை நினைவகம் [Auxiliary Memory]

முதன்மை நினைவகம் ஒரு கணிப்பொறியின் முக்கியமான திறமை பொருந்திய [Powerful resource] பகுதியாகக் கருதப் படும். இதன் மூலம் செய்திக்கூறுகளையும் வழியமைப்புகளையும் செயற்படுத்த இயலும் கணித்தல், பகுப்பாய்வு செய்தல், தொகுத்து வகைப்படுத்தல் முதலிய பல பணிகளை மையச் செயலகம் செய்யும்பொழுது, செய்தி தாங்கியாக முதன்மை நினைவகம் செயற்படுகிறது துணை நினைவகங்கள் செய்திகளை நிலையாகப் பதிவு செய்து வைத்துக்கொள்ள உதவுகின்றன. ஒவ்வொரு கணிப்பொறியின் முதன்மை நினைவகத்தின் அளவிற்கும் ஓர் எல்லை உண்டு. குறிப்பிட்ட அளவிற்கு மேல் முதன்மை நினைவகத்தை அதிகரித்தல் இயலாததாகும். ஆனால், துணை நினைவக அளவைத் தேவைக்கேற்ப அதிகரிக்க இயலும்.

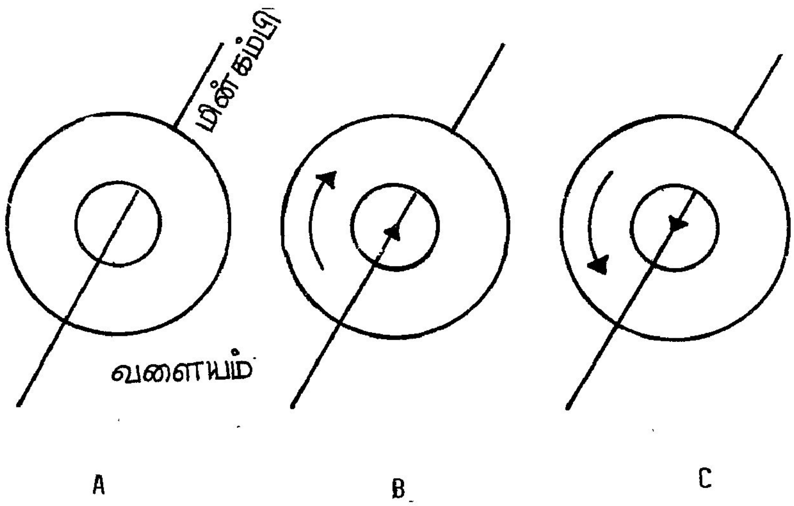
5.2.3.1. மிகைவேக நினைவகம் (FAST ACCESS MEMORY)

மையச் செயலகத்திலுள்ள [CPU] கணக்ககம் எண்களைக் கூட்டுதல், கழித்தல், முதலிய பணிகளைச் செய்யும் பொழுது மிகைவேக நினைவகம் [Fast Access Memory] பெரிதும் துணை புரிகிறது. மையச் செயலகத்தின் செயற்பாட்டு வேகத்திற்கு ஏற்ற வகையில் செயற்படுவதற்கேற்ப மற்ற நினைவகங்களைவிட இது சில குறிப்பிட்ட சிறப்பியல்புகளைப் பெற்றுள்ளது. அதாவது செய்திக்கூறு, ஆணை முதலியவற்றை அதிவிரைவு அணுக்க [Fast Access] முறையில் படிக்கவோ, கணித்த இடைநிலை விடையைப் பதிவு செய்யவோ இது பயன்படுகிறது. இந்த அதி வேக நினைவகத்தை இரண்டு வகைகளாகப் பிரிக்கலாம்.

1. மின்னணுவியல் நிலைமாற்றிகள்
[Electronic Flipflops]

2. காந்த இரும்பு வளைய நினைவகம்
[Ferrite core Memory]

மின்னணுவியல் நிலைமாற்றி, காந்த இரும்பு வளைய நினைவகத்தைவிட அதிக வேகமுடையது. மின்னணுவியல் நிலைமாற்றிகள், மின்னழுத்தத்தில் உயர்வு(1)அல்லது தாழ்வு(0) என்ற அடிப்படையில் இருவிதமான செய்திக் கூறுகளைப் பதிவு செய்யும் தன்மையுடையன. காந்த இரும்பு வளைய நினைவகமானது சுமார் 5 மி. மீ. விட்டமுடைய இரும்பு வளையமாகும். இவ்வளையத்தின் வழியாக ஒரு சிறிய மின்கம்பி செல்கிறது. இந்த மின் கம்பியில் மின்சாரம் ஒரு குறிப்பிட்ட திசையில் செல்லும் பொழுது மின்தாரைகள் ஏற்படுகின்றன. ஆனால் அதே கம்பியில் மின்சாரம் பாயும் திசையை மாற்றும் பொழுது மின்காந்தத் தாரையின் திசையும் மாறுகிறது. எனவே இதன் அடிப்படையில், செய்திகளைப் பதிவு செய்ய இயலும். இந்த அமைப்பு, படம் எண் 31—இல் காட்டப்பட்டுள்ளது.



படம்—31
காந்த வளைய நினைவகம் [Magnetic Core memory]

5.2.3.2. முதன்மை நினைவகம் (PRIMARY MEMORY)

முதன்மை நினைவகம் நாம் இடுவரல்செய்யும் வழியமைப்பு, செய்திக்கூறு [Data] முதலியவற்றைப் பதிவு செய்து வைத்துக் கொண்டு செயற்படுத்துவதற்கு மையச் செயலகத்திற்கு உதவி செய்கிறது. ஒரு கணிப்பொறியின் அமைப்புமுறை [Configuration] முதன்மை நினைவகத்தின் அளவைப் பொறுத்துத் தீர்மானிக்கப் படுகிறது. தற்கால ஆராய்ச்சிகளின் பயனாய் மிகக்குறைந்த பொருட் செலவிலேயே அதிகமான செய்திக் கூறுகளைப் பதிவு செய்யும் ஆற்றல் வாய்ந்த நினைவகங்கள் கிடைக்கக் கூடியனவாக உள்ளன. முதன்மை நினைவகத்தைக் கீழ்க்கண்டவாறு வகைப் படுத்தலாம்.

1. காந்த இரும்பு வளைய நினைவகம் [Magnetic Core memory]
2. ஒருங்கமை மின்சுற்று நினைவகம் [Integrated Circuit Memory]
3. காந்த உருளை நினைவகம் [Magnetic Drum Memory]
4. மின்காந்தக் குமிழி நினைவகம் [Magnetic Bubble Memory]

முதன்மை நினைவகத்திற்குத் துணை நினைவகத்தைக் காட்டிலும் சில குறிப்பிட்ட பண்புகள் [Properties] தேவைப் படுகின்றன. அவைகளாவன :

1. நம்பும் திறன் [Reliability]
2. அணுக்க வேகம் [**Access speed**]
3. அளவில் சிறிய அமைப்பு [Compactness]

தற்காலத்தில் உருவாக்கப்படும் நினைவகங்கள் அதிக அளவு நம்பிக்கைத் தன்மையையும் வேகத்திறனையும், மிகக் குறைந்த அணுக்க **நேரத்தையும்** பெற்றுள்ளன. மேலும், இவை அளவில் மிகவும் சிறியனவாகவும் உள்ளன. எனவே இவற்றைப் பராமரித்தலும் குறிப்பிட்ட நினைவகப் பகுதி பழுதடையும்பொழுது சீர்திருத்தலும் மிகவும் எளிதாகிறது.

5.2.3.2.1. காந்த இரும்பு வளைய நினைவகம் (MAGNETIC CORE MEMORY)

இது சிறிய புள்ளியைப் போன்ற அமைப்பையுடையதாகும். இந்தச் சிறிய புள்ளிகள் வளையம் போன்ற அமைப்பைப் பெற்றுள்ளன. இந்த வளையத்தின் விட்டம் சுமார் 5 மி. மீ. ஆகும். வளையத்தினூடே ஒரு மின்கம்பி செல்கிறது. இந்த மின்கம்பியின் மூலமாகச் செல்லும் மின்சாரத்தின் திசையை மாற்றிக் காந்தத் தாரையை [Magnetic Flux] வலச் சுழி யாகவோ [Anti Clockwise], இடச் சுழியாகவோ [Clockwise] மாற்றி அமைக்கலாம். இதை அடிப்படையாகக் கொண்டு '0' அல்லது '1' என்ற மதிப்பைப் பதிவு செய்ய இயலும்.

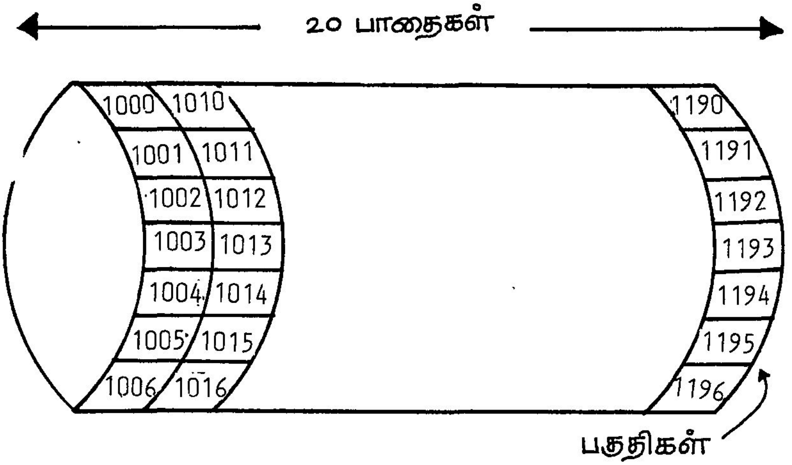
பல்லாயிரக்கணக்கான வளையங்கள் கொண்ட காந்த இரும்பு வளைய நினைவகம், ஒரு தேனடையைப் போன்ற அமைப்புடையதாகும். இவ்வாறு ஒவ்வொரு குறியும் [Character] 8 இருநிலை எண்களின் [Binary Digits-Bits] சேர்க்கையினால் நினைவகத்தில் குறிப்பிடப்படுகிறது. இந்தத் தொகுதிக்கு 'BYTE' என்று பெயர். அதாவது ஒரு குறியை நினைவகத்தில் பதிவு செய்யத் தேவையான நினைவக அளவின் அலகே [Unit], 'பைட்' [Byte] எனப்பெயர் பெறும்.

5.2.3.2.2. காந்த உருளை நினைகம் (MAGNETIC DRUM MEMORY)

காந்த உருளை நினைவகம், காந்த வளைய நினை வகத்தைப் போன்றே செயல்படுகிறது. ஆனால், இதன் அமைப்பில் வேறுபாடுகள் உள்ளன. இது ஓர் இரும்பு உருளை மீது காந்தப் பொருள் பூசப்பட்ட அமைப்பையுடையதாகும். இந்தக் காந்தப் பொருளின் மேற்பரப்பில் உள்ள, புள்ளிகள் ஒவ்வொன்றும் மின் காந்தமாகச் செயல்படுகின்றன. இதைக் கொண்டு காந்தத் தாரையை இடச்சுழியாகவோ, வலச்சுழி யாகவோ மாற்றி '0' அல்லது '1' என்ற இரு நிலை எண்களைப் பதிவு செய்யமுடிகிறது. இவ்வாறு பல இருநிலை எண்களின் தொகுப்பைக் கொண்டு செய்திக்கூறுகளை மின்காந்த உருளை நினைவகத்தில் பதிவு செய்ய இயலும். உருளையின் மேற்பரப்பு பல பாதைகளாகவும் ஒவ்வொரு பாதையும் பல பகுதிகளாகவும் பிரிக்கப்பட்டுள்ளன. ஆனால், ஒவ்வொரு

பகுதிக்கும் குறிப்பிட்ட சுட்டு எண் கொடுக்கப்பட்டுள்ளதால், அதன் மேற்பரப்பிலுள்ள பாதையில் தேவையான செய்திக் கூறுகள் எங்கு உள்ளன எனக் கண்டறிய முடிகிறது.

படம் எண். 32-இல் இதன் அமைப்பு காட்டப்பட்டுள்ளது.



படம்—32

காந்த உருளை நினைவகம் [MagneticDrum Memory]

மின்காந்த உருளை நினைவகம் செலவில் குறைந்தது. ஆனால் காந்த வளைய நினைவகத்தைவிட செயற்பாட்டுத் திறனில் குறைந்த வேகமுடையது. ஒரு மணித்துளிக்கு உருளை யானது சுமார் 7,000-இற்கும் மேற்பட்ட சுற்றுகள் சுற்றக்கூடிய வேகமுடையது. தற்கால நவீனக் கணிப்பொறிகளின் வேகத்திற்கு, இதன் வேகம் மிகவும் குறைவாகையால் பெரும்பாலும் காந்த உருளை நினைவகம் பயன்படுத்தப்படுவது இல்லை.

5.2.3.2.3. ஒருங்கமை மின்சுற்று நினைவகம் *(INTEGRATED CIRCUIT MEMORY)*

முதலாம், இரண்டாம், மூன்றாம் காலக்கட்டக் கணிப்பொறி களில் இரும்பு வளைய நினைவகமே பெரும்பாலும் பயன்பட்டு

வந்தது. ஆனால், கணிப்பொறித் துறையில், குறிப்பாகக் கருவிய விரிவாக்கத்தில் [Hardware Development] செய்யப்பட்ட பல ஆராய்ச்சிகளின் பலனாக விலை மலிவான, செயல்திறன் மிக்க புதுவகை முதன்மை நினைவகங்கள் பல உருவாக்கப்பட்டுள்ளன. அவற்றுள் ஒருங்கமை மின்சுற்று நினைவகம் ஒன்றாகும்.

இது முந்தைய நினைவகங்களைக் காட்டிலும் மிகக்குறைந்த அளவு மின்சக்தியைக்கொண்டு இயங்கக்கூடிய தன்மை வாய்ந்தது. மேலும், குறைந்த கொள்ளளவில் அதிகச் சொல் அளவு [Word Capacity] உள்ள நினைவகத்தை உள்ளடக்க முடியும். மிகை வேகக் கணிப்பொறிகளின் இயக்கத்திற்குத் தகுந்தாற்போன்று செயல்படுவதற்கு இது மிகக் குறைந்த அணுக்க நேரத்தைக் [Access Time] கொண்டது. இந்த வகை நினைவகங்கள் கையாள் வதற்கு மிகவும் இலகுவானவை. எனவே செயற்பாட்டில் குறை நேரின், இதைச் சுலபமாக நீக்கி அதற்குப் பதிலாக வேறு ஒரு பல்மின் சுற்றுச் சேர்ப்பியைப்[Chip] பொருத்திவிடலாம். இந்த வகையில் கணிப்பொறியைப் பழுது பார்த்தலும் பேணுதலும் சுலபமாகிறது. இந்த ஒருங்கமை மின்சுற்றுகள் மெல்லிய சிலிக்கான் [Silicon]எனப்படும் ஒருவகை மணலைக்கொண்டு உருவாக்கப்படுகின்றன. இவ்வாறு உருவாக்கும் பொழுது பல்லா யிரக்கணக்கான மின்னணு மூலகங்களைக்கொண்டு [Electronic Components] அதன் மேற்பரப்பில் மின்சுற்றுகள் வடிவமைக்கப் படுகின்றன. இதில் மின்சாரம் பாயும்பொழுது உயர்நிலை யாகவும் பிற நேரங்களில் தாழ்நிலையாகவும் கருதப்படுகிறது. இதைக் கொண்டு *'0'* அல்லது *'1'* என்ற இருநிலை எண்க ளைப் பதிவு செய்யலாம். தற்காலத்தில் ஒரே பல்மின் சுற்றுச் சேர்ப்பியில் கூட [Chip] 64,000 பைட்ஸ் அதாவது குறிகளைப் பதிவு செய்யும் திறன்வாய்ந்த நினைவகங்கள் கிடைக்கின்றன. இவ்வாறு நிறைவிலா மின்கடத்திகளைக்கொண்டு உருவாக்கப் பட்ட ஒருங்கமை மின்சுற்று நினைவகம் தற்கால அதிவேகக் கணிப்பொறியின் வேகத்திற்கு ஏற்ப ஈடுகொடுத்துச் செயல் படுகிறது.

5.2.3.2.4. மின்காந்தக் குமிழி நினைவகம் (*MAGNETIC BUBBLE MEMORY*)

மின்காந்தக் குமிழி நினைவகம், மின்காந்தத்தின் தன்மை யைக்கொண்டு செயல்படுகிறது. இவ்வகை நினைவகமானது

நிறைவிலா மின் கடத்தியைக்கொண்டு தயாரிக்கப்பட்ட மெல்லிய படலத்தின் மீது [Thin Film] சிறு சிறு மின்காந்தப் புள்ளிகளைக் கொண்ட அமைப்பையுடைதாகும். இப்புள்ளிகளே மின்காந்தக் குமிழிகள் [Magnetic Bubbles] என்றழைக்கப்படுகின்றன.

ஒருங்கமை மின்சுற்று நினைவகம், **நிறைவிலா மின் கடத்தி** நினைவகம் ஆகியவற்றைக் காட்டிலும் மின் காந்தக் குமிழி நினைவகத்தில் மின்சாரம் இல்லாமல் இருக்கும்பொழுதுகூட, செய்திக்கூறுகளையும் பிற விவரங்களையும் நினைவகத்தில் இருத்திவைக்க இயலும். இதுவே இதன் தனித்தன்மையாகும். இதை இழப்பிலா இருத்து முறை[Non Volatile]என்று கூறலாம்.

5.2.4. ஆளகம் (CONTROL UNIT)

ஆளகம் கணிப்பொறியின் பிற அங்கங்களின் செயற்பாட்டைக் கட்டுப்படுத்த உதவுகிறது. கணிப்பொறி செயல்படும் பொழுது இடுவரல் செய்திகளைப் பெறவும் பதிவு செய்யவும் கணிப்புச் செயல் நடைபெறவும் பின்னர் விடுவரலைப் பெறவும் தேவையான கண்காணிப்புச் செயல்களைப் புரிவதற்கேற்ற கட்டளைகளை ஆளகம் பிறப்பிக்கிறது.

5.2.5. இடுவரல், விடுவரல் ஒருங்குகள் (INPUT, OUTPUT SYSTEMS)

அட்டைத் துளைப்பொறி [Card Punching Machine], அட்டைப் படிப்பொறி [Card Reader], அச்சுப்பொறி ஆகியவற்றின் அமைப்பும் இயக்கமும் பற்றி முன்பே கவனித்தோம். இப்பகுதியில் சில முக்கியமான இடுவரல் விடுவரல் ஒருங்குகளான மின்காந்த நாடாப்பொறி [Magnetic Tape Unit], மின்காந்தத் தட்டுப்பொறி [Magnetic Disc Unit], காட்சிப்பொறி மற்றும் மின்காந்த மென்வளைத் தட்டு [Floppy Disc] ஆகியவற்றின் அமைப்பையும் அவற்றின் இயக்க முறையையும் விரிவாகக் காண்போம்.

5.2.4.1. மின்காந்த நாடாப் பொறி (MAGNETIC TAPE UNIT)

ஒரு கணிப்பொறியில் விவரங்களை ஆய்வு செய்வதற்குச் செய்திக் கூறுகளைக் கணிப்பொறியில் இடுவரல் செய்தலும் குறிப்

பிட்ட படிவத்தில் [Specified Format] செய்திகளைப் பதிவு செய்தலும் [Recording] அவசியமாகும். அவ்வாறு பதிவு செய்யப்பட்ட செய்திக்கூறுகள் நீண்ட நாள்களுக்குப் பாதுகாக்கப் பட்டுத் திரும்பத் திரும்பத் தேவைப்படும்பொழுது பயன்படுத்தப் படுகின்றன. எடுத்துக்காட்டாகக் கூறின், சம்பளப் பட்டியல் தயாரித்தல் [Payroll Preparation], தொழிற்கூடங்களில் உற்பத்தித் திட்டமிடல், பொருள் நிலையாளுகை, விற்பனையியல் [Sales Management], நிதிநிலைக் கணக்கு [Financial Accounting] முதலிய பல துறைகளில் அவை தொடர்பான செய்திகள் அடங்கிய முதன்மைக் கோவைகள் [Primary Files] பல ஆண்டுகளுக்குக் கணிப்பொறியில் பாதுகாக்கப்பட்டு அடிக்கடி பயன்படுத்தப்படுகின்றன. மேலும், இச்செய்திகளைப் பாதுகாக்க வும் கணிப்பொறியின் வேகத்திற்கு ஏற்ப இவைகளை அறிந்து உணர்ந்து ஆய்வு செய்யவும் இடுவரல், விடுவரல் ஒருங்குகளாகப் [Input--Output systems] பயன்படுத்தவும் மின்காந்த நாடாக் களும் மின்காந்தத் தட்டும் பெரிதும் பயன்படுகின்றன. மின்காந்த நாடாவிலுள்ள ஒவ்வொரு கோவையிலும் [Files] பல்லாயிரக் கணக்கான பதிப்புகளைப் [Records] பதிவு செய்து வைக்க முடியும். மேலும் துளையிடப்பட்ட அட்டைகளில் பதிவு செய் துள்ள செய்திக்கூறுகளை மின்காந்த நாடாவிற்கு மாற்றிப் பதிவு செய்வதன் மூலம் கணிப்பொறியின் செயலாக்க வேகத்திற்கு ஏற்ற வகையில் பயன்படுத்தச் சாத்தியமாகிறது.

1. அதிக அளவிலான செய்திக்கூறுகளும் வழியமைப்புகளும் பின்னர் தேவைப்படும் பொழுது பயன்படுத்தும் பொருட்டு மின்காந்த நாடாவில் பதிவு செய்யப்படு கின்றன.

2. இவ்வாறு மின்காந்த நாடாவில் பதிவு செய்யப்பட்ட ஒவ்வொரு வகைச்செய்திக்கும் இரண்டுக்கும் மேலான படிகள் [Back Ups] பதிவு செய்துவை த்தல், தேவைப் படும் பொழுது அறிக்கைகளைத் தயார் செய்ய உதவுகிறது.

3. ஒவ்வொரு தடவையும் புதிய செய்திக்கூறுகளை மின் காந்த நாடாவில் பதிவு செய்யும்பொழுது மின்காந்த நாடாவில் முன்பேயிருந்த செய்திகள் அழிக்கப்பெறு

மாதலால் தேவையான செய்திகளை முன்பே பாதுகாத்தல் அவசியமாகும்.

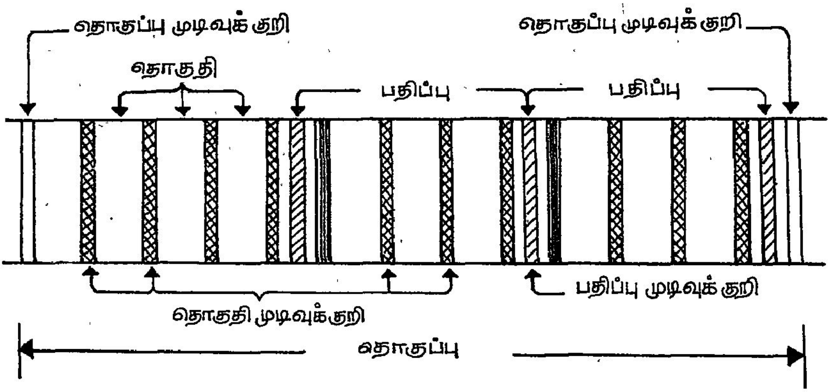
4. மின்காந்த நாடாவில், நாடாவின் நடுவில் தேவையான செய்திகளை மட்டும் முறையிலா முறையில் [Random] படித்தறிதல் இயலாது. இது வரிசைக்கிரம இடுவரல் கருவிக்கு [Sequential input device] ஓர் எடுத்துக் காட்டாகும். எனவே, மின்காந்த நாடாவில் செய்தியை வரிசையாக முதலிலிருந்து கடைசிவரை படித்தறிதலோ, எழுதுதலோ மட்டுமே இயலும்.

5. எண்ணற்ற செய்திகளைப் பற்பல பயன்பாடுகளுக்கு ஏற்ப ஆயிரக்கணக்கான மின்காந்த நாடாக்களில் பதிவு செய்து மின்காந்த நாடா நூலகம் [Tape Library] மூலம் பல ஆண்டுகளுக்குப் பாதுகாக்கலாம். பயன் பாட்டிற்கேற்ப அதனுடைய சுட்டு எண், பெயர் [Label] முதலியவற்றைக்கொண்டு தேவையான மின் காந்த நாடாவை மட்டும் தெரிவு செய்து பயன் படுத்துதல் சாத்தியமாகிறது.

5.2.5.2. மின்காந்த நாடாவில் செய்தி அமைப்புமுறை *(DATA ORGANIZATION IN MAGNETIC TAPE)*

மின்காந்த நாடாவில் செய்திக்கூறு எவ்வாறு பதிவு செய்யப்படுகிறது என்பதை நன்கு தெரிந்திருந்தால் மட்டுமே, அதைப் பின்னர் வழியமைப்பின் மூலம் படித்தறிந்து ஆய்வனை செய்ய இயலும். மின்காந்த நாடாவில் பதிவாகியுள்ள செய்திக் கூறின் அமைப்பைக் கீழ்க்கண்டவாறு விவரிக்கலாம்.

பல குறிகள் [Characters] சேர்ந்த தொகுப்பே செய்திக் கூறாக அமைகிறது. இவ்வாறு பல குறிகளின் சேர்க்கைக்குத் தொகுதி [Field] என்றும் அவ்வாறு பல தொகுதிகளின் சேர்க் கைக்குப் பதிப்பு [Record] என்றும் பெயர் கூறப்படுகிறது. அதே போன்று பல பதிப்புகளை ஒன்று சேர்த்துத் தொகுப்பு [Block] என்றும் பல தொகுப்புகளை ஒன்று சேர்த்துக் கோவை [File] என்றும் அழைக்கிறோம். படம் எண் 33—இல் தொகுதி, பதிப்பு, தொகுப்பு ஆகியவை விளக்கப்பட்டுள்ளன.



படம்—33 மின்காந்த நாடாவில் செய்தியமைப்பு முறை

5.2.5.3. மின்காந்த நாடாவின் அமைப்பு (STRUCTURE OF A MAGNETIC TAPE)

மின்காந்த நாடாக்கள் பெரும்பாலும் ½ அங்குல அகலமும், சுமார் 1200 அல்லது 2400 அடி நீளமும் உடையனவாகக் கிடைக்கின்றன. பிளாஸ்டிக்கால் ஆன நாடாவின் மீது மின்காந்தப் பொருள் கொண்ட பூச்சுப் பூசப்பெற்றுள்ளது. இந்த மின் காந்தப் பொருளின் ஒவ்வொரு புள்ளியும் மின் காந்தம் ஆகும்பொழுது மின்காந்தத் தாரைகளை உண்டாக்குகின்றன. இதன் அடிப்படையில் செய்திகள் பதிவு செய்யப்படுகின்றன. ஒரு மின் காந்த நாடாவில் சுமார் 200 இலட்சம் குறிகளைப் பதிவு செய்யும் கொள்ளளவு உள்ளது. மின் காந்த நாடாவின் இருமுனைகளிலும் உலோகக் குறிகள் இணைக்கப்பட்டுள்ளன. நாடாவின் ஆரம்பக் குறியைக் கண்டு, நாடா அனுப்பும் சுற்றுகையிலிருந்து [Sending Reel] அடையும் சுற்றுகைக்குச் [Receiving Reel] சுற்றுகிறது, படம் எண் 34—இல் சம்பளப்பட்டியல் தயாரித்தலில் பயன்படும் ஒரு பதிப்பின் [Record] அமைப்பு விளக்கப்பட்டுள்ளது.

EMP NO.	NAME	DEPT NO.	DESIGNATION	DATE OF JOINING	LAST INCREMENT DATE	NEXT INCREMENT DUE DATE	ADDRESS	LEAVES BALANCE		
								CL	EL	SL

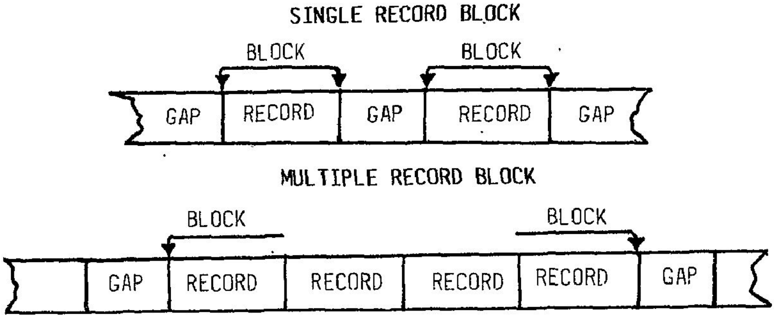
EMP NO.	BASIC	D.A	H.R.A	C.C.A	OVERTIME RATES			L.I.C	P.F.	F.P.F	I.TAX
					DAY	NIGHT	HOLIDAY				

EMP NO.	LEAVES TAKEN			OVERTIME HOURS			PRODUCTION BONUS	LOSS OF PAY-NO. OF DAYS	SPECIAL ALLOWANCE	CANTEEN CHARGES	TRANSPORT ALLOWANCE
	EL	CL	SL	DAY	NIGHT	HOLIDAY					

படம்—34

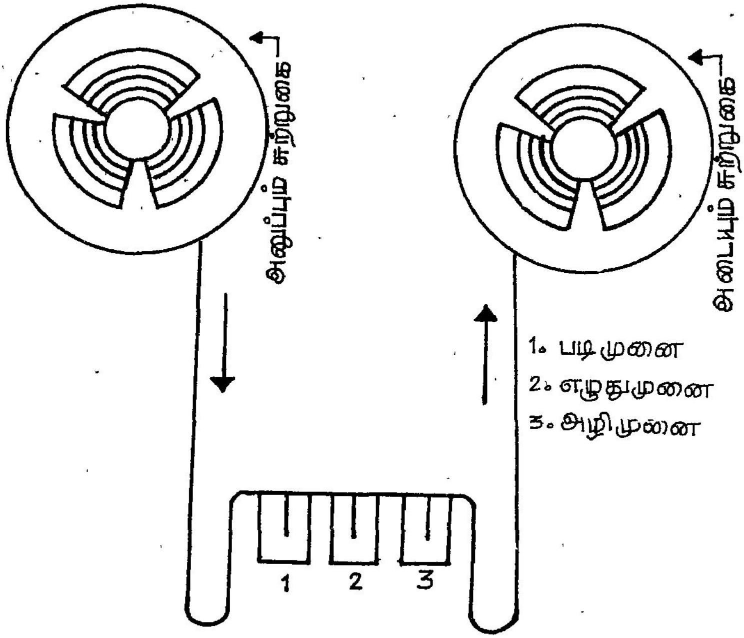
ஊதியப் பட்டியலின் செய்திப் பதிப்பு **[Payroll Records]**

செய்தியை மின்காந்த நாடாவில் படித்தறியும் பொழுது ஒவ்வொரு தொகுதி முடிந்தபிறகும் ஒரு தொகுதி முடிவுக்குறியும் **[End of Field Mark]**, *ஒவ்வொரு பதிப்பு* **[Record]** *முடிந்த பிறகும் ஒரு பதிப்பு முடிவுக்குறியும்***[EndofRecordMark]***போடப் பட்டிருக்கும். அவ்வாறே ஒவ்வொரு தொகுப்பு முடிவடைந்த வுடன் ஒரு தொகுப்பு முடிவுக்குறியும்* **[End of Block Mark]**, *அவ்வாறு ஒரு கோவை* **[File]** *முடிந்தவுடன் ஒருகோவை முடிவுக் குறியும்* **[End of File Mark]** *போடப்பட்டிருக்கும். ஒவ்வொரு தொகுதி,பதிப்பு ஒவ்வொரு பதிப்பிலும் தொகுதிகளின் எண்ணிக் கைகள், ஒவ்வொரு தொகுப்பிலும் உள்ள பதிப்புகளின் எண்ணிக்கைகள் ஆகிய குறிப்புகள் நாடாவின் ஆரம்பத்திலேயே பதிவு செய்யப்பட்டிருக்கும். மையச் செயலகம்* **[CPU]** *இதைப் படித்துணர்ந்து அதற்கேற்பச் செயற்படுகிறது. மேலும் ஒவ் வொரு தொகுப்பு, பதிப்புகளுக்கிடையே வெற்றிடமாக இடை வெளி இருக்கும். இதற்கு முறையே தொகுப்பு இடைவெளி* **[IBG Inter Block Gap]**, *பதிப்பு இடைவெளி* **[IRG Inter Record Gap]** *என்று பெயர். இவ்விடைவெளிகள் படம் எண் 35—இல் சுட்டிக்காட்டப்பட்டுள்ளன.*



படம்—35 தொகுதி, பதிப்பு இடைவெளிகள்
[Inter field and Inter Record Gaps]

நாடாவின் முடிவுக் குறியைக் கண்டதும் சுழற்சி நிற்கிறது. ஒவ்வொரு முறையும் காந்த நாடாவில் செய்திக்கூறுகள் பதிவு செய்யப்படுவதற்கு முன்பு ஆரம்பப் பதிப்பாக ஒரு தலைக்குறி [Header Label] பதிவு செய்யப்படும். இதிலுள்ள சுற்றுகையின் எண் [Number], வேலை விவரம் [Details], பணியின் ஆரம்பத் தேதி [Initial Date], முடிவுறும் தேதி [Finishing Date] ஆகிய விவரங்களை மையச் செயலகம் படித்துணரும். இரண்டாவது குறி வால்குறி [Tail Label] என்றழைக்கப்படும். இதன் மூலமாகக் கோவை முடிந்து விட்டதா, அல்லது அடுத்த நாடாவில் ஆரம்பமாகிறதா என்ற விவரங்கள் கண்டறியப்படுகின்றன. மின்காந்த நாடா சுற்றுகையிலிருந்து [Reel] செல்லும் பொழுது படிமுனை [Read Head], எழுது முனை [Write Head], அழிமுனை [Erase Head] என்ற மூன்று முனைகளின் வழியாகச் செல்கிறது. இந்த அமைப்பு, படம் எண் 36—இல் காட்டப்பட்டுள்ளது.



படம்—36 மின்காந்த நாடாப் பொறி [Magnetic Tape Unit]

மேற்குறிப்பிடப்பட்ட மூன்று முனைகளில் எழுதும் முனை [Write Head] செய்திகளை மின்காந்த நாடாவின் மேற்பரப்பில் எழுதுவதற்கும் படிமுனை [Read Head] செய்திக் கூறுகளைப் படிப்பதற்கும் பயன்படுகின்றன. அழிமுனை [Erase Head] நமக்கு நாடாவில் உள்ள செய்திக் கூறுகள் தேவைப்படாதபொழுது அழிக்கப் பயன்படுகின்றது.

5.2.5.4. மின்காந்த நாடாவில் செய்தி அடர்வு முறை (*DATA RECORDING DENSITY IN A MAGNETIC TAPE*)

மின்காந்த நாடாவில் ஒவ்வொரு குறியும் மின்காந்தமாக்கப் பட்ட பல புள்ளிகளின் [Magnetized Bits] தொகுப்புகளைக் கொண்டு குறிப்பிடப்படுகிறது. இதன் மூலமாக மின் காந்த நாடாவில் அதிக அளவு செய்திகளை நம்மால் பதிவு செய்ய முடிகிறது. இவ்வாறு 1 அங்குல நீளமுள்ள நாடாவில் எவ்வளவு

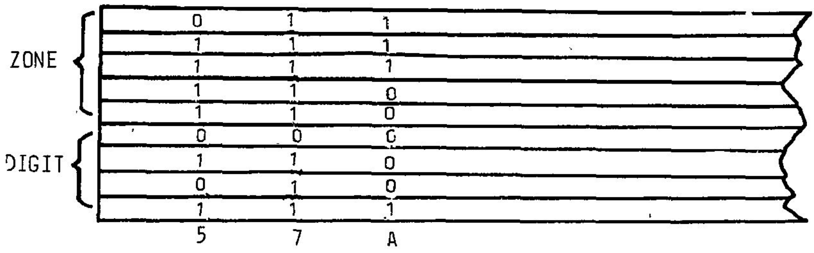
குறிகளைப் பதிவு செய்யமுடியுமோ அதற்கு அடர்வு முறை [Density] என்று பெயர். அதாவது 1 அங்குல நீள நாடாவில் பதிவாகியுள்ள Bits களின் எண்ணிக்கையால் இது அளவிடப் படுகிறது. இதற்கு [BPI-Bits Per Inch] என்று பெயர். பெரும் பாலும் மின்காந்த நாடாக்களில் Bytes Per Inch என்ற முறையில் அதாவது 1 அங்குலத்திற்கு எவ்வளவு குறிகளைப் [Bytes] பதிவு செய்யலாம் என்ற முறை பயன்படுத்தப்படுகிறது. இது 800 BPI ஆகவோ 1200 BPI ஆகவோ இருக்கும்.

5.2.5.5. மின்காந்த நாடாவில் செய்திகள் பதிவு செய்யப்படும் முறை (DATA RECORDING METHOD IN A MAGNETIC TAPE)

கணிப்பொறியின் மையச் செயலகத்தில் செய்திகள் எவ்வாறு பொறி மொழியில் குறிப்பிடப்படுகின்றனவோ அதுபோன்றே, மின்காந்த நாடாவின் மேற்பரப்பிலும் '0' அல்லது '1' என்ற முறையில் செய்திகள் பதிவு செய்யப்படுகின்றன. ஒவ்வொரு துறையின் பயன்பாட்டுக்கும் ஏற்ற முறையில் மின்காந்த நாடா ஏழு அல்லது ஒன்பது நீளவாட்டத் தாரைகளைக் [Tracks] கொண்டுள்ளது, இவ்வாறு உள்ள ஒவ்வொரு தாரையும் ஒரு மின்காந்தப் புள்ளி [Magnetized Spots] மூலம் செய்திகளைப் பதிவு செய்யும் தன்மை வாய்ந்தது. ஏழு தாரைகளை உடைய நாடாவில் ஒவ்வொரு தாரைக்கும் ஒரு படி எழுதுமுனை வீதம் ஏழு படி எழுது முனைகள் அமைக்கப்பட்டுள்ளன.

இங்ஙனமே ஒன்பது தாரைகளையுடைய நாடாவில் ஒன்பது படி எழுதுமுனைகள் அமைக்கப்பட்டுள்ளன. ஏழு தாரை நாடாக்கள் [BCD—Binary Coded Decimal] என்ற முறையில், குறிகளை நாடாவின் மேற்பரப்பில் பதிவுசெய்கின்றன. அதாவது [BA 8421] என்ற புள்ளி [Bits] முறையில் மின்காந்தமாக்கப்பட்ட புள்ளிகள் '1' ஐக் குறிப்பனவாகவும் மற்ற புள்ளிகள் '0' —ஐக் குறிப்பனவாகவும் கொள்ளப்படுகின்றன. ஏழாவது இடப்புள்ளி யானது சோதனைப் புள்ளியாகப் [Test Bit] பயன்படுத்தப்படு கிறது. இதே போன்று ஒன்பது தாரை நாடாவில் எட்டுத் தாரைகள் எட்டுப் புள்ளிகளினால் ஒரு குறியையும் [Character] ஒன்பதாவது புள்ளி சோதனைப் புள்ளியாகவும் [Test Bit] பயன் படுத்தப்படுகின்றது. இம்முறையில் விவரங்களையும் செய்தி

களையும் பதிவு செய்யும் முறைக்கு EBCDIC CODE என்று பெயர். ஒன்பது தாரை நாடாவில் ஐந்து, ஏழு,A என்ற குறிகளை எங்ஙனம் குறிப்பது என்பதைப் படம் எண் 37–இல் காணலாம்.



படம்—37 ஒன்பது தாரை நாடாவில் செய்திக் கூறமைப்பு

5.2.5.6. மின்காந்தத் தட்டு (MAGNETIC DISC)

மின்காந்த நாடாவைப் போன்றே மின்காந்தத் தட்டும் பல்லாயிரக்கணக்கான பதிப்புகள் [Records] கொண்ட செய்திகளைப் பதிவு செய்யவும் அதை ஆய்வு செய்யவும் பயன்படவல்ல ஒரு மிகை வேக இடுவரல் கருவியாகும். எடுத்துக்காட்டாகக் கூறின், ஒரு மின் காந்தத் தட்டில் சுமார் 10,00,00,000 குறிகள் கொண்ட செய்தியைப் பதிவு செய்ய இயலுமென்றால் அதன் பதிவுத் திறனுக்கு வேறு சான்றுகள் தேவையில்லை. மின்காந்தத் தட்டும் மின்காந்தப் பொருள் பூசப்பட்ட அமைப்பையுடையது. எனவே, மின்காந்த நாடாவில் செய்திகள் பதிவு செய்யப்படும் முறையிலேயே, இதிலும் செய்திகள் பதிவு செய்யப்படுகின்றன.

5.2.5.7. மின்காந்தத் தட்டின் அமைப்பு

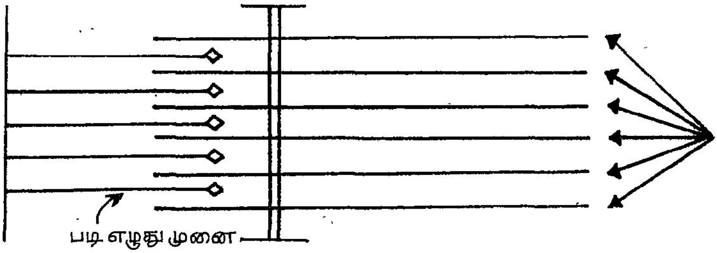
பல தட்டுகள் ஒன்றன் மேல் ஒன்றாக அடுக்கப்பட்ட நிலையில் இவ்வமைப்பு அமைந்திருக்கும். இதை இரண்டு வகைகளாகப் பிரிக்கலாம்.

1. நிலையான மின்காந்தத் தட்டடுக்கு [Stationary Magnetic Disc Pack]

2. மாற்று மின்காந்தத் தட்டடுக்கு [Changeable Magnetic Disc Pack]

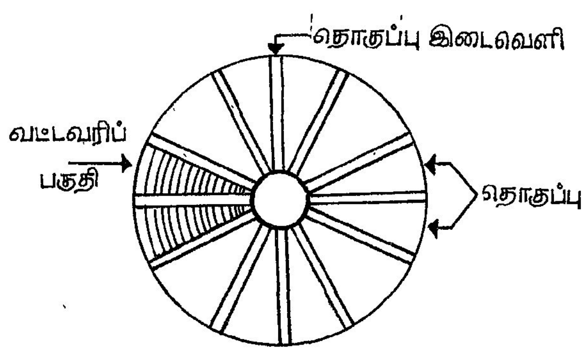
5.2.5.8. மாற்று மின்காந்தத் தட்டடுக்கு (CHANGEABLE MAGNETIC DISC PACK)

இந்த முறையில் பல தட்டடுக்குகளை [Disc Pack] வேண்டியபொழுது மாற்றிக் கொள்ளவும் செய்திகளின் கொள் திறனுக்கு [Capacity] ஏற்பப் பொருத்தவும் இயலும். இனி மின் காந்தத் தட்டின் அமைப்பைப்பற்றிச் சிறிது காண்போம். மின் காந்தத் தட்டு நமது வீடுகளில் பயன்படுத்தப்படும் இசைத் தட்டைப் [Phonograph Records] போன்ற அமைப்பை உடைய தாகும். ஆனால், இது உலோகத்தால் செய்யப்பட்டுத் தட்டின் மேல், கீழ்ப்பகுதிகளில் நுண்காந்தப் பொருள் பூசப்பட்டுள்ளது. பல தட்டுகள் ஒன்றன்மேல் ஒன்றாக அடுக்கப்பட்ட நிலையில் இரண்டு தட்டுகளுக்கு இடையில் போதிய இடைவெளி விடப் பட்டுள்ளது. இவ்வாறு அமைக்கப்பட்ட எல்லாத் தட்டுகளும் மையத்திலுள்ள உருளை வடிவச் சட்டத்தில் [Spindle] பொருத்தப்பட்டுள்ளன. தட்டுகளுக்கு இடையில் படி—எழுது முனை [Read, Write Head] பொருத்தப்பட்டுள்ளன. இந்தப் படி—எழுது முனைகளே செய்திகளைத் தட்டில் எழுதவும் அல்லது படிக்கவும் துணை செய்கின்றன. இதில் ஒன்று அசையும் படி—எழுது முனை [Moving Head Magnetic Disc] வகையும் மற்றொன்று நிலையான படி—எழுது முனையும் [Fixed Head Magnetic Disc] ஆகும். படம் எண் 38-இல் அசையும் படி—எழுது முனையைச் சேர்ந்த மின் காந்தத் தட்டின் அமைப்பு விளக்கப்பட்டுள்ளது. உருளைச் சட்டத்தில் பொருத்தப்பட்டு உள்ள மின்காந்தத் தட்டுகளின்மேல், கீழ் ஆகிய இரண்டு பரப்பு களிலும்செய்திகள் பதிவுசெய்யப்படுகின்றன.எடுத்துக்காட்டாகச் சுமார் பதினொரு அடுக்குகள் கொண்ட அமைப்பில், ஒரு தட்டுக்கு இரண்டு பக்கங்கள் வீதம் செய்திகளைப் பதிவு செய்யக் கூடிய [Recording Surface] இருபது பரப்புகள் [Surfaces] உள்ளன. காரணம் மேல் அடுக்கின் மேல் பரப்பும், கீழ் அடுக்கின்



படம்—38
அசையும் படி—எழுது முனை [Moving Read-write head]

கீழ்ப் பரப்பும் [Surface] செய்திகளைப் பதிவு செய்யப் பயன் படுத்தப்படுவது இல்லை. சுமாராக ஒவ்வொரு தட்டிலும் 100 மைய வட்ட வரிகள் [Concentric Tracks] உள்ளன. ஒவ்வொரு வட்ட வரியும் பல தொகுப்புகளாகப் பிரிக்கப்பட்டுள்ளன. ஒவ்வொரு தொகுப்பிலும் சுமார் 100 வட்ட வரிப் பகுதிகள் [Circular bands] உள்ளன. இரண்டு தொகுப்புகளுக்கு இடையே தொகுப்பு இடைவெளி [IBG—Inter Block Gap] உள்ளது. இவ்வாறு பிரிக்கப்பட்டுள்ள பகுதிகளிலுள்ள [Sectors] செய்திக் கூறுக்கோவைகளைப் படித்தறியவும் புதிய கோவைகளை உருவாக்கவும் ஏற்ற விதத்தில் மின்காந்தத் தட்டின் மேற்பரப்பின் ஒவ்வொரு வட்டவரிப் பகுதிக்கும் சுட்டு எண் [Address] கொடுக்கப்பட்டிருக்கும். படம் எண் 39—இல் ஒரு தட்டின் மேற்பரப்பு குறிப்பிடப்பட்டுள்ளது.



படம்—39 மின்காந்தத் தட்டின் மேல்பரப்பு

5.2.5.9. மின்காந்தத் தட்டின் தனி இயல்புகள்

மின்காந்தத் தட்டின் தனி இயல்புகளைக் கீழ்க்கண்டவாறு தொகுத்துக் கூறலாம்.

1. செய்திகளைப் பதிவு செய்தலிலும் படித்தறிதலிலும் மையச் செயலகத்தின் வேகத்திற்கு ஒப்ப இது செயற் படுகிறது.
2. இதில் பல்லாயிரக்கணக்கான செய்திக்கூறுப் பதிப்புகளை [Data Records] சேமித்து வைக்க முடியும்.
3. ஒரு குறிப்பிட்ட செய்தியையோ கோவையையோ பதிவு செய்யப்பட்டுள்ள மொத்தச் செய்தியிலிருந்து தெரிவு செய்ய வேண்டுமெனில் மின்காந்த நாடாவில் வரிசைக் கிரம அணுகு முறையில் [Sequential Access] நாடா முழுவதும் படிக்கவேண்டியிருக்கும்.ஆனால், மின்காந்தத் தட்டிலோ செய்தியை முறையிலா அணுக்கம் மூலம் [Random access] படித்தறிய இயலும்.
4. மின்காந்தத் தட்டில் செய்திக்கூறு, வழியமைப்பு,மற்றும் கணிக்கப்பட்ட விடை முதலியவற்றைப் பதிவு செய்து வைத்துக்கொண்டு வேண்டியபொழுது பயன்படுத்த முடியுமாதலால் இது ஒரு இடுவரல் மற்றும் விடுவரல் கருவியாகப் பயன்படுகிறது.

5.2.5.10. மின்காந்த மென்வளைத்தட்டு (FLOPPY DISC)

உயர் நுட்ப நிலை,இடைநிலைக் கணிப்பொறிகளில் அதிக அளவு உள்ள செய்திகளைப் பதிவு செய்யவும் [Recording] ஆய்வு செய்யவும் [Processing] வேண்டியுள்ளது. இதன் காரணமாக அதிக அளவு கொள்திறன் கொண்ட மின்காந்த நாடாக்களையும் மின்காந்தத் தட்டுகளையும் பயன்படுத்துகிறோம். ஆனால் சிறிய [Mini] மற்றும் மிகச் சிறிய [Micro] கணிப்பொறிகள், பெரும் பாலும் சிறிய நிறுவனங்கள் அல்லது ஆய்வுக் கூடங்களில் பயன் படுத்தப்படுகின்றன. இவற்றில் அளவில் சிறிய மின்காந்தத் தட்டு செய்திகளைப் பதிவு செய்வதற்குப் பயன்படுத்தப்படு கிறது. இதற்கு மின்காந்த மென்வளைத் தட்டு [Floppy Disc]

என்று பெயர். இதில் வரிசைக் கிரம அணுக்கம் மூலம் செய்திகளைப் படித்தறிதல் தவிர, முறையிலா அணுக்கம் மூலமும் செய்திகளைப் பெறுவதற்கு இயலும்.

வழக்கமாக ஒரு படி—எழுது முனையைக் கொண்ட மின்காந்த மென்வளைத்தட்டு அமைப்பில் [Floppy Disc system] சுமார் 2,50,000 குறிகளைப் பதிவு செய்ய இயலும். மேலும் செய்தி அடர்வு முறையை இரட்டிப்பாக்கியும் இரண்டு படி-எழுது முனைகளைக் [Read-Write Head] கொண்டும் ஒரு தட்டில் சுமார் 10இலட்சம் குறிகளைப் பதிவு செய்ய இயலும். ஒவ்வொரு மென்வளைத் தட்டிலும் 77 வட்ட வரிகளும் [Tracks] ஒவ்வொரு வட்டவரியிலும் 26 பகுதிகளுமாக [Sectors] அமைந்திருக்கும். தட்டின் மேற்பரப்பிலுள்ள பல பகுதிகளின் சுட்டு எண்கள் [Address] அட்டவணையில் [Directory] பதிவு செய்யப்பட்டுள்ளதால், நமது கட்டளைக்கேற்பக் குறிப்பிடப்பட்டகோவையிலுள்ள செய்தியைத் தேடிக் கணிப்பொறியால் நமக்குப் படித்தளிக்க இயலுகிறது. இம்முறையில் படி—எழுது முனைகள் தட்டின் மேற்பரப்பின் மீது அமர்ந்து செய்திகளைப் பதிவு செய்யவோ, படிக்கவோ செய்கின்றன. இது மின்காந்த நாடா, மின்காந்தத் தட்டு ஆகியவற்றிலிருந்து செய்திகளை அணுகும் முறையிலிருந்து வேறுபட்டிருக்கிறது.

மென்வளைத் தட்டை, மின்காந்த நாடா மற்றும் மின் காந்தத் தட்டு ஆகியவற்றுடன் ஒப்பிடும்பொழுது, மென்வளைத் தட்டு கீழ்க்கண்ட வசதிகளைப் பெற்றுள்ளது.

1. மின்காந்தத் தட்டைப்போன்றே, செய்திகளை அணுகும் முறையில் மென்வளைத் தட்டிலும் முறையிலா அணுகு முறை பயன்படுகிறது.

2. மென்வளைத் தட்டு விலையில் மிகவும் குறைந்தது. எனவே, கணிப்பொறியைப் பயன்படுத்தும் ஒவ்வொருவரும் இதைத் தேவைக்கேற்ப வாங்கிப் பயன்படுத்த முடியும்.

3. இது அளவில் மிகச் சிறியது. இது பெரும்பாலும் சிறிய மற்றும் மிகச்சிறிய கணிப்பொறிகளில் பயன்படுத்த ஏற்றது. தேவையான வழியமைப்புகளின் மாற்றுப் பிரதிகளை [Backup copies] இதன்மூலம் பாதுகாத்தல் இயலும்.

5.2.5.11. காட்சித்திரை (DISPLAY SCREEN)

தொலை அச்சுத்தட்டுப் பலகை முறையில், நாம் கணிப் பொறிக்குக் கொடுக்கும் கட்டளைகள் தட்டச்சுப் பொறியைப் போன்றே வெண்தாளில் அச்சாகின்றன. மற்றொரு இடுவரல் முறையான கணிப்பொறி அட்டையில் செய்திக் கூறுகள் துளை யிடப்பட்ட நிலையில் குறிப்பிடப்படுகின்றன. இம்முறையில் வழியமைப்பு ஆணைகள், செய்திக்கூறு ஆகியவற்றில் தவறு ஏதும் இருப்பின், அதைத் திருத்தும் பொருட்டு வேறு புதிய அட்டையைத் துளை செய்யவேண்டிய அவசியம் ஏற்படுகிறது. இதனால் கணிசமான அளவு பொருள் விரயமும் கால விரயமும் ஏற்படுகிறது. மேலும் மனிதன் கணிப்பொறியுடன் நேரடியாகத் தொடர்பு கொள்ளும் முறையும் அற்றுப்போகிறது. இவை களுக்குப் பதிலாக உருவாக்கப்பட்ட இடுவரல் அமைப்பே காட்சித்திரை ஆகும்.

5.2.5.11.1. காட்சித் திரையின் அமைப்பு

இது தொலைக்காட்சிக் கருவியைப் போன்ற திரையையும் செய்திகளைத் தட்டச்சு செய்வதற்கு ஏற்றவகையில் பல குறிகள் அமையப் பெற்ற தட்டுப் பலகையையும் பெற்றுள்ளது. இக்கருவி கணிப்பொறியுடன் இணைக்கப்படும் பொழுது மின்னணு முறையில் இயங்குகிறது. இதனால் தட்டச்சு செய்யும் பொழுது செய்திக் கூறுகளும் வழியமைப்பு ஆணைகளும் காட்சித்திரையில் எழுதப்படுகின்றன. இவ்வாறான முறையில் இடுவரல் செய்யப் பட்ட செய்திக்கூறுகளில் தவறு ஏதேனும் இருப்பின், உடனுக்கு உடன் அதை இயக்குபவரால் திருத்தம் செய்துகொள்ள இயலும். இதில் செய்திக்கூறுகள் தட்டச்சு செய்யப்படும் பொழுது ஒவ்வொரு தடவையும் சட்டத்திருப்பை [CR-Carriage Return] அழுத்தினால், திரையில் செய்திகள் கீழிருந்து மேல்நோக்கி ஒவ்வொரு வரியாக நகரும். மேலும் காட்சிப் பொறி கணிப் பொறியுடன் நேராக இணைக்கப்பட்டிருக்கும் பொழுது கணிப் பொறி இயக்குபவர் மற்றும் வழியமைப்பாளர்களால் கணிப் பொறியுடன் தொடர்பு கொண்டு ஆணைகளை இடுவரல் செய்யவும் வழியமைப்புகளைச் செயலாக்கவும் [Execution] முடிகிறது.

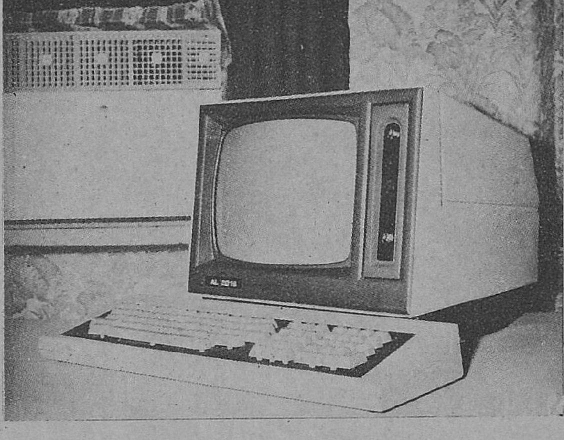
5.2.5.11.2. காட்சித்திரையின் சிறப்பியல்புப் பண்புகள்

1. இப்பொறியை இடுவரல் கருவியாகப் பயன்படுத்தும் பொழுது நேர்த்தியாகவும் திறம்படவும் செம்மையாக வும் பணியைச் செய்ய இயலுகிறது.

2. துளையிடப்பட்ட அட்டைகளின் மூலம் செய்திகளைக் கணிப்பொறிக்கு இடுவரல் செய்யும் முறையை இக்கருவி யுடன் ஒப்பு நோக்கின், கணிசமான அளவு பொருள் விரயமும் கால விரயமும் தவிர்க்கப்படுகின்றன. ஆனால் காட்சித் திரையில் எழுதப்படும் செய்திகளைப் பார்வை யிடவோ, திருத்தவோ முடியுமேயொழிய, திறப்படிவ மாகப் [Hard Copy] பெறுதல் இயலாது. அதற்கென ஓர் அச்சுப் பொறி தேவைப்படும்.

கட்டிடக் கலைத்துறை வரைபடங்கள் மற்றும் தொழிற் கூடங்களில் பல தொழில்நுட்பம் பொருந்திய கருவிகளை உரு வாக்கத் தேவையான முன்மாதிரி வடிவமைப்பு வரைபடங்கள் முதலியவற்றைக் காட்சித்திரையிலேயே வரைந்து சீர்திருத்துதல் இயலும். இதற்கெனக் கிராபிக் காட்சித்திரைகள் [Graphic Display Screens] தற்காலத்தில் கிடைக்கின்றன.

காட்சித் திரையின் அமைப்பு, படம் எண் 40–இல் காட்டப்பட்டுள்ளது.



படம்–40 காட்சித்திரை [Video Display Screen]

பாகம் – 6

6. கணிப்பொறியின் பயன்கள்

6.1. முன்னுரை

கணிப்பொறியின் தோற்றம், வரலாறு, அதன் அமைப்பு, செயற்பாட்டுமுறை முதலிய விவரங்களைப் பற்றி இதுவரை ஆராய்ந்தோம். மனிதன் தனது அறிவின் திறத்தால் உருவாக்கிய கருவிகள் பற்பல. 18-ஆம் நூற்றாண்டில் தொழில் துறையில் பெரும் புரட்சி உண்டாயிற்று. அதன் பயனாகப் பல புதிய எந்திரங்கள் தொழில் துறையில் புகுத்தப்பட்டு, மனிதனுக்குப் பேருதவி புரிந்தன. இவையாவும் மனிதனின் உடல் உழைப்பிற்கு மாற்றாக, உறுதுணையாய் விளங்கின. நமது அன்றாட வாழ்க்கைக்கு இன்றியமையாத பொருள் எங்ஙனம் நம் வாழ்க்கையில் பின்னிப் பிணைந்துள்ளதோ, அவ்வாறே அறிவியலின் பயன்கள் மனித வாழ்வில் மிக அவசியத் தேவையாக விளங்குகின்றன. எடுத்துக்காட்டாகக் கூறின், நாம் எழுதப் பயன்படுத்தும் எழுதுகோல் முதற்கொண்டு தொழிலகங்களில் பயன்படுத்தும் பற்பல கருவிகள், ஊர்திகள், வானொலி, தொலைக்காட்சி, மருத்துவக் கருவிகள் முதலியவையாகும். பன்னெடுங் காலமாக நடைபெற்ற ஆராய்ச்சிகளின் பயனாகத் தோன்றி, அறிவியல் யுகத்திலே புதியதொரு மறுமலர்ச்சியை உண்டாக்கியுள்ள ஒரு கருவி 'கம்ப்யூட்டர்' எனப்படும் கணிப்பொறியேயாகும். பற்பல துறைகளிலும் இதன் பயன் அளவிடற்கரியது. இது மற்ற கருவிகளைப் போன்று மனிதனின் உடல் உழைப்பிற்கு மட்டும் உறுதுணையின்றி, உள உழைப்பிற்கும் திறன்மிகு மாற்றுச் சாதனமாகத் திகழ்கிறது. இந்த வகையில் இது பிற கருவிகளிலிருந்து வேறு

பட்டுள்ளது. பல துறைகளில் கணிப்பொறி எங்ஙனம் பயன் படுத்தப்படுகிறது என்பதைப் பற்றிக் கீழ்க்கண்டவாறு சுருக்க மாகக் காணலாம்.

தொழிற்கூடங்களில் புதிய கருவிகள் பல பயன்படுத்தப் பட்டமையால் மக்களின் வாழ்க்கைநிலை பெரிதும் உயர்ந்தது. சமூகம், பொருளாதாரம், தொழில் முதலியவற்றில் பெரும் மாற்றங்கள் ஏற்படலாயின. பொருள் உற்பத்தியில் பன்மடங்கு பெருக்கம் ஏற்பட்டது. இதன் பொருட்டு விலைவாசிகள் குறைய வும், மக்களின் வாழ்க்கைநிலை ஓங்கவும் செய்தன. கணிப்பொறி யின் பயன் இல்லாத துறையே இல்லை என உறுதியாகக்கூறலாம். நமது பாரத நாட்டிலும் கல்வி, தொழில், வேளாண்மை, விண்வெளி ஆய்வு, ஆராய்ச்சி, போக்குவரத்து முதலான பல துறைகளில் கணிப்பொறி படிப்படியாகப் பயன்படுத்தப்பட்டு வருகிறது. கணிப்பொறியைத் தொழிலகங்களில் பயன்படுத்து வதன் மூலம், பலர் வேலைபெறும் வாய்ப்பை இழக்கிறார்கள் என்ற ஐயம் பலபேருக்கு உண்டு. ஆனால், சீரிய முறையில் பல புதிய திட்டங்களைத் திட்டமிட்டு,சிறப்பாகக் கணிப்பொறியைப் பயன்படுத்தும்பொழுது பல துறைகளில் ஏராளமான வேலை வாய்ப்பைப் பெருக்கவும் அதன்மூலம் குறிப்பிடத்தக்க அளவு பொருளாதார நிலையை உயர்த்தவும் இயலும் என்பதில் சிறிதும் ஐயமில்லை.

மனையில் உணவுப் பட்டியல் தயாரிக்க உதவுவதிலிருந்து விண்வெளி ஆய்வுப் பயணத்தைச் சிறிதும் முறைபிசகாது வெற்றி கரமாக நடத்தி முடிக்க உதவுவதுவரை, கணிப்பொறியின் பயன்கள் எண்ணிறந்தவை. கல்வி, மருத்துவம், வேளாண்மை, தொழில், தொலை தொடர்பு, போக்குவரத்து, பொறியியல், விண்வெளி ஆய்வு, பாதுகாப்புப்படை, காவல்துறை, வங்கி ஆகிய சில குறிப்பிட்ட துறைகளில், கணிப்பொறி எங்ஙனம் பயன் படுகிறது என்பதைக் கீழ்க்கண்டவாறு சுருக்கமாக ஆராயலாம்.

6.2 கல்விப் பணியில் கணிப்பொறி

பிற துறைகளில் கணிப்பொறி பயன்படுவது போலவே தற்காலக் கல்வித் துறையில் கணிப்பொறி குறிப்பிடத்தக்க அளவு மறுமலர்ச்சியை ஏற்படுத்தியுள்ளது.

நீண்ட அனுபவமும் அறிவாற்றலும் கொண்ட பல பேராசிரியர்களால் திட்டமிட்டுத் தயாரிக்கப்பட்ட பாடங்களைக் [Programmed Instructions] கணிப்பொறியில் பதிவு செய்து வைத்துக்கொள்ள வேண்டும். இப்பாடங்களைச் செயற்கைக் கோள்கள் மூலம் அனுப்பி, பல்லாயிரம் மைல்களுக்கு அப்பாலுள்ள மாணவர்களுக்குப் பாடம் கற்பிக்க இயலும். அமெரிக்காவிலுள்ள ஹார்வார்டு பல்கலைக் கழகப் பேராசிரியர் ஸ்கின்னர் [Prof. Skinner of Harvard University] என்பவர் தயாரித்த கல்வி முறைபற்றிய ஆய்வுக் கட்டுரையில், **1954—ஆம்** ஆண்டிலேயே இது பற்றி வற்புறுத்திக் கூறப்பட்டுள்ளது. மேற் குறிப்பிடப்பட்ட முறையில் பல கேள்விகளின் மூலம் மாணவனின் திறனறிந்து, அதற்கேற்பப் பாடம் கற்பிக்க முடியும்.

எடுத்துக்காட்டாகக் கணிப்பொறி ஒரு மாணவனுக்குக் கணிதம் கற்றுக்கொடுப்பதாக வைத்துக்கொள்வோம். இது வகுப்பறைக் கல்வியிலிருந்து முற்றிலும் வேறுபட்டது. கணிதத்தின் அடிப்படைக் கல்வி நிலையிலிருந்து [Elementary Mathematics] தொடங்கி, படிப்படியாக மாணவனுக்குக் கற்பித்தல் அவசியம். இது எவ்வாறு சாத்தியமாகிறது? அடிப்படைக் கணிதத் தத்துவம் முழுவதும் மாணவனுக்கு முதலில் கற்பிக்கப்படுகிறது. படிப்படியாக அதற்கு அடுத்த நிலைக்கு அவன் கற்கச் செல்ல விரும்பினால், அதுவரை அவன் கற்றுள்ள பாடத்தில் அவனது அறிவைத் தேர்ந்தறிதல் அவசியம். எனவே அதற்கேற்பவும் மாணவனின் திறனுக்கேற்பவும் கணிப்பொறி பல வினாக்களைத் தொடுத்து மாணவனின் விடையை உறுதி செய்து, தவறெனில் திருத்துகிறது; இல்லையெனில் அடுத்த பாடப்பகுதியைக் கற்பிக்கத் தொடங்குகிறது. எனவே மாணவன் பாடத்தைப் புரிந்துகொள்ளும் அறிவுத்திறன், வினாக்களுக்கு விடையளிக்கும் திறமை ஆகியவற்றைக் கொண்டு, கணிப்பொறி ஒரு தனி மாணவனின் படிப்பில் நெருங்கிய கவனம் செலுத்துகிறது. அடிப்படைப் பாடத்திலிருந்து தொடங்கி, கல்லூரி, பல்கலைக் கழகப் பாடங்கள் வரை இவ்வாறு கல்விபயிற்றுவித்தல் இயலும். இவை போன்ற பணியைச் செய்வதற்குக் கணிப்பொறிகளுக்கு ஏற்ற சிறந்த பல வழியமைப்புகளை உருவாக்குதல் அவசியம்.

மேற்கண்ட முறையில் பாடம் பயிலும்பொழுது ஓர் ஆசிரியரின் தொடர்பு மாணவனுக்குக் கிடைப்பதில்லை என்ற குறை

மட்டும் உண்டு. ஆனால், இம்முறை கட்டுப்பாடும் ஒழுங்கும் நிறைந்த கல்வி முறையாகும். இதன் மூலம் கல்விகற்கும் ஆர்வத்தை மாணவர்களிடையே தூண்டுவதுடன், முறைமையான சிந்தனையை [Logical thinking] உண்டாக்கவும் கல்வியறிவு நிறை அறிவாளிகளாக அவர்களை உருவாக்கவும் முடியும் என்பதில் சிறிதும் ஐயமில்லை மேலும் தெரிவு செய்யப்பட்ட வினாக்களையும் அதற்கு ஏற்ற விடைகளையும் [Question-Answer Bank] கணிப்பொறியில் பதிவு செய்து வைத்துக்கொள்வதன் மூலம், சீரான பாடத்திட்டத்தைக் கொண்டுள்ள கல்லூரிகளிலும் பல்கலைக் கழகங்களிலும் கணிப்பொறிவழிக் கல்வியைச் சிறந்த முறையில் செயற்படுத்த முடியும்.

தேர்வில் மாணவர்களின் விடைத்தாள் திருத்தும் பணியில் கணிப்பொறியைப் பயன்படுத்துவதன் மூலம் ஆயிரக்கணக்கான ஆசிரியர்களின் நேரமும் வேலைப்பளுவும் குறைக்கப்படுகின்றன. கணிப்பொறி மூலம் இப்பணியைத் திறம்படவும் சிறப்பாகவும் செய்யின், விரைவிலும் உரிய நேரத்திலும் தேர்வு முடிவுகளை அறிவித்தல் முடியும்.

கணிப்பொறி வழிக் கல்வி முறையில் ஓர் ஆசிரியர் தனிப்பட்ட மாணவனின் கல்வியில் முன்னேற்றம் காண எவ்வளவு சிரத்தை எடுத்துக்கொள்ள வேண்டுமோ, அதே பணியைக் கணிப்பொறி மூலம் சிறப்பாகச் செய்யலாம். பள்ளிச் சிறார்கள் முதற்கொண்டு கல்லூரி மாணவர்கள் வரை, அடிப்படைப் பாடங்களான பொது அறிவியல், கணிதம், இயற்பியல், பொறியியல் முதலியவற்றை மிகச்சிறந்த முறையில் கணிப்பொறி மூலம் [Computer Aided and Computer Managed Instructions] கற்கமுடியும் என்பதில் மேலை நாடுகள் பெரும் வெற்றி கண்டு உள்ளன.

நமது பாரதத்தில் கல்வித் துறையில் மறுமலர்ச்சி காண பள்ளிக் கல்வி நிலையிலிருந்து கணிப்பொறிப் பாடத் திட்டமும் கணிப்பொறி மூலம் பயிற்றுவிக்கும் முறையும் நடைமுறைப் படுத்தப்படவுள்ளன. இதற்கென மைய அரசு பல சிறப்பார்ந்த திட்டங்களை வகுத்துள்ளது. இத்திட்டத்திற்கு CLASS — Computer Literacy And Studies in Schools என்று பெயர். இத்திட்டத்தினடிப்படையில் 1990-இற்குள் நாடு முழுவதும் உள்ள இரண்டரை இலட்சம் பள்ளிகளில் கணிப்பொறிகள்

நிறுவப்படவுள்ளன. இதற்கென் விலையில் குறைந்த கல்விப் பணிகளுக்குகந்த சிறு கணிப்பொறிகளை நமது நாட்டிலேயே வடிவமைத்து உருவாக்கத் தேவையான திட்டங்களும் தீட்டப் பட்டுள்ளன.

தமிழ் போன்று தொன்மை வாய்ந்த மொழிகளில் கணிப்பொறி மூலம் ஆராய்ச்சிகள் பல மேற்கொண்டு கலை, அறிவியல், சமூக மாற்றங்களுக்கு ஏற்ற வகையில் மொழிக்குப் பெரும் வளம் சேர்க்கலாம். அறிவியலின் புதுமைக் கண்டு பிடிப்புகள், ஒரு மொழியின் வளர்ச்சிக்குப் பெருமளவு பயன் படுத்தப்படுதல் மிகுந்த நன்மை பயக்கும். இந்த வகையில் கணிப்பொறியின் துணைகொண்டு தானியங்கி மொழி பெயர்ப்பு [Automatic Language Translation] செய்வதன் வாயிலாகப் பல மொழி பேசப்படும் நமது பாரதத்தில், ஒரு மொழி பேசும் மக்கள் மற்ற மொழியினரின் பண்பாட்டையும் இலக்கிய வளத்தையும் அறிந்து பயன்பெறப் பெரும் வாய்ப்பு உண்டாகிறது. மேலும் ஆராய்ச்சிப் பணிகளில் ஏராளமான புள்ளி விவரங்களைத் தொகுத்துப் பதிவு செய்தல், பகுத்தலும் கணித்தலும் செய்து விடை காணல் ஆகியவற்றைக் கணிப்பொறி மூலம் சில மணி நேரங்களில் மிகவும் திறம்படச் செய்துவிடலாம். இதன் காரணமாக ஆராய்ச்சியாளர்களுக்கு வேண்டிய செய்தியைக் கணிப்பொறி மூலம் விரைவில் தந்து உதவலாம். எனவே ஆய்வாளர்கள் தங்களுடைய நேரத்தை அதிக அளவில் பயன்மிகு ஆய்வுப் பணிகளில் செலவழிக்க வழி ஏற்படுகிறது.

6.3. மருத்துவத் துறையில் கணிப்பொறி

மின்னணுவியல் வளர்ச்சியின் காரணமாகப் பல துறை களிலும் முன்னேற்றங்கள் ஏற்பட்டுள்ளன. அதே போன்று மருத்துவத்திலும் குறிப்பிடத்தக்க கண்டுபிடிப்புகளுக்குக் கணிப் பொறி பேருதவி செய்கிறது. குறிப்பாக மருத்துவத் துறையில் மருத்துவச் செய்திகளைத் தொகுத்தல், நோய்களின்தீமைகளைப் பகுத்துணர்ந்து கூறுதல் [Diagnosing], நோயின் தன்மையைக் கண்டறிதல் முதலியவற்றில் கணிப்பொறி மகத்தான பணி புரிகிறது. மேலும், நோயின் தன்மைகளுக்கேற்ற மருத்துவ முறையை விரைவில் கண்டறிந்து மருத்துவம் செய்யவும் கணிப் பொறி உதவி செய்கிறது. பெரிய மருத்துவமனைகளில் பல்லா யிரக்கணக்கான நோயாளிகள் மருத்துவ உதவி பெறுவர். இதில்

நோயாளிகளின் ஊர், வயது, பெயர் முதலான விவரங்கள், அவர்களுடைய நோயின் தன்மை ஆகிய செய்திக்கூறுகளைக் கணிப்பொறியில் சேகரித்து வைப்பதன் மூலம் குறிப்பிட்ட நோயாளியைக் கண்டறிந்து, அவசரத்திற்கேற்ப உடனுக்குடன் மருத்துவர்களால் நோயைக் குணப்படுத்த முடிகிறது. இம்முறையில் மருத்துவ நுணுக்கம் முதற்கொண்டு பிற அலுவல்களையும் கணிப்பொறி கையாள முடிவதால், மருத்துவர்கள் ஒவ்வொரு நோயாளிக்கும் அதிகக் கவனம் செலுத்த வழி ஏற்படுகிறது.

நோயாளியின் இரத்த அழுத்தம், இதயத்துடிப்பு, உடல் வெப்பநிலை முதலான விவரங்களைத் தானாகவே கண்காணித்து அதற்கேற்ப மருத்துவருக்குச் செய்தியை அனுப்பி அறிவுறுத்தும் Bed Side Monitors போன்றவைகளில் தற்காலத்தில் கணிப்பொறி பயன்படுத்தப் படுவதால், ஒரே மருத்துவர் பல நோயாளிகளைத் திறம்படக் கவனிக்க முடிகிறது. மேலும் நோயாளியின் இதய நோயை அறியப் பயன்படும் இதயமானி [Electro Cardiogram], கதிர்வீச்சுக் கருவிகள், எக்ஸ்ரே [X-Ray] முதலியவற்றைக் கொண்டு நோயின் தன்மையை அறிதல், சிகிச்சை செய்தல் ஆகியவற்றில் கணிப்பொறி மிகவும் அவசியமாகிறது.

நோயாளிகள் விபத்தில் காயமடையும் பொழுதோ, அறுவைச் சிகிச்சை நடைபெறும் பொழுதோ அவர்களுக்கு அதிக அளவு இரத்த இழப்பு ஏற்படுகிறது. இதற்கு ஒரு குறிப்பிட்ட நோயாளியின் இரத்தத்தின் தன்மையையொத்த [Blood Group] மற்றொருவருடைய இரத்தம் உட்செலுத்தப்படுதல் அவசியமாகும் எனவே இரத்தவங்கிகளில் [Blood Bank] ஒவ்வொருவகை இரத்தத்தின் கையிருப்பு, அதை வழங்கியவரின் முகவரி, இரத்தம் தேவைப்படும் நோயாளியைப் பற்றிய விவரங்கள் முதலியவற்றைக் கணிப்பொறியில் பதிவு செய்து வைத்துக்கொள்வதன் மூலம், அவசரத்திற்கு ஏற்ப இவ்வசதியைப் பயன்படுத்தி மருத்துவம் செய்ய இயலுகிறது. மேலும், கணிப்பொறியின் துணையால் ஆயிரக்கணக்கான மைல்களுக்கு அப்பாலுள்ள திறன்மிகு மருத்துவர்களின் உதவிகொண்டு வேறோர் இடத்தில் உள்ள நோயாளிக்கு மருத்துவம் செய்யமுடிகிறது. இம்முறையில் நோயாளியைப் பற்றிய விவரங்கள், நோயின் அறிகுறி [Symptom] ஆகிய செய்திகளைச் செயற்கைக்கோள் செய்தித் தொடர்பு முறையில் [Satellite Communication] அனுப்பி, அங்குள்ள கணிப்பொறியின் உதவியினால் நோயிற்கேற்ற

மருந்துகளையும் செய்யவேண்டிய மருத்துவச் செயல்முறைகளையும் திரும்பப் பெற்று, அதற்கேற்ப மருத்துவம் செய்து கொள்ளமுடியும்.

மேலை நாடுகளில் அடிப்படை மருத்துவ வசதியை மருத்துவர்களே இல்லாமல், கணிப்பொறி மூலம் மக்கள் பெறுவதற்கான வாய்ப்புகள் தற்காலத்தில் கிடைக்கின்றன. இம்முறையில் கணிப்பொறியானது நோயாளியின் பெயர், ஊர் முதலிய விவரங்களைப் பெற்றபின், அவரது நோயின் தன்மையைத் தெரிவு செய்யப்பட்ட பல வினாக்கள் மூலம் கேட்கிறது. நோயாளியின் விடையைப் பொறுத்து, நோயின் தன்மை கண்டறியப்பட்டு, அதற்கான மருத்துவ அறிவுரைகள் தகுந்த முறையில் நோயாளிக்கு அறிவுறுத்தப்படுகின்றன. இதன் ஒவ்வொரு நிலையிலும் நோயாளி எங்ஙனம் கணிப்பொறியைப் பயன்படுத்த வேண்டுமென்ற குறிப்பும் அறிவுரையும் வழங்கப் படுகின்றன. சமீபத்தில் வெளிவந்த செய்திகளின்படி, சீன நாட்டில் 80 வயதைக் கடந்த வாதநோயால் பாதிக்கப்பட்ட முதிய ஒரு மருத்துவ அறிஞர், தமது தொழிலில் கணிப் பொறியைப் புகுத்தியுள்ளார். இவர் தமது நீண்ட அனுபவத்தின் மூலம் கிடைத்த பல அரிய மருத்துவக் கண்டுபிடிப்புகளையும் கருத்துகளையும் கணிப்பொறிக்குக் கற்றுக் கொடுத்துள்ளார். எனவே இவரது, குறைந்தளவு உதவியைக் கொண்டு மருத்து வருக்குப் பதிலாகக் கணிப்பொறி நோயாளிகளைக் கவனித்து மருத்துவ உதவி வழங்க இயலுமெனில், மருத்துவத்துறையில் கணிப்பொறியின் சேவைக்கு வேறு சான்று தேவையில்லை.

6.4. வேளாண்மைத் துறையில் கணிப்பொறி

இந்தியா போன்ற நாடுகளின் பொருளாதார வளம், மக்களின் வாழ்க்கை மேம்பாடு முதலானவை, வேளாண்மையைப் பெரிதும் சார்ந்துள்ளன. நமது நாட்டின் பெரும்பகுதி நிலங்கள் நீர்ப்பாசனத்திற்குப் பயன்படுத்தப்படாமல் இருக்கின்றன. மேலும் ஆண்டுதோறும் ஆறுகளில் ஏற்படும் வெள்ளப்பெருக்கால் பெருமளவு பொருள் மற்றும் உயிர்ச்சேதங்கள் ஏற்படுவதும் அதே சமயம் நாட்டின் பல பகுதிகளில் மக்கள் நீர்ப்பாசனம் இன்றி பட்டினி, பஞ்சத்தால் துன்புறுவதும் நாம் கண்கூடாகக் காணும் நிகழ்ச்சிகளாக உள்ளன. தற்காலத்தில் வேளாண்மைத் துறையில் பல கண்டுபிடிப்புகள் நிகழ்ந்து கொண்டிருக்கின்றன.

கணிப்பொறியை வேளாண்மைத் துறையில் பயன்படுத்துவதன் மூலம் வியத்தகு செயல்கள் பல செய்ய இயலும்.

பயிர்ப்பாதுகாப்பு, நீர்ப்பாசனத் திட்டமிடல் [Planned irrigation], மண்வள ஆய்வு [Soil testing], மண்ணுக்கும் பயிருக்கும் ஏற்ற உரமிடல் முதலானவற்றைக் கணிப்பொறியைப் பயன்படுத்திச் சிறப்பாகச் செயற்படுத்தலாம். நீர்நிலைகளைக் கண்டறிதல், பருவகால நிலையைக் கண்டறிந்து உழவர்களுக்குப் பயிர் செய்ய அறிவுறுத்தல், புயல். மழை, கடல் கொந்தளிப்பு முதலியவற்றை உழவர்களுக்கு முன்கூட்டியே பெறப்பட்ட செய்திகளின் [Recorded Data] மூலம் கண்டறிந்து எச்சரிக்கை செய்தல் [Agricultural forecasting] முதலிய பணிகளைக் கணிப்பொறியால் உடனுக்குடன் திறம்படச் செய்ய இயலும். மேலும் நீர்ப்பாசன நிலைகளான ஏரி, குளம், வாய்க்கால், அணைக்கட்டு முதலியவற்றைச் செப்பனிட்டு வடிவமைத்தல் போன்ற பல்வேறு பணிகளில் கணிப்பொறி பெரும் பங்கு வகிக்கிறது.

6.5. தொழில்துறையில் கணிப்பொறி

18–ஆம் நூற்றாண்டில் ஏற்பட்ட தொழிற்புரட்சியின் காரணமாகத் தொழிலகங்களில் பற்பல எந்திரங்கள் புகுத்தப் பட்டு, தொழில் துறையில் மறுமலர்ச்சி ஏற்பட்டது. இதையே தொழில் நாகரிகம் என்று கூறுவர். அதேபோன்று கணிப்பொறியும் தொழிற்கூடங்களில் உற்பத்திப் பெருக்கத்திற்குப் பெருந்துணை புரிகிறது. தொழில் துறையில் சீரிய முறையில் உற்பத்தித் திட்டமிடல், நிதிநிலைக் கணக்கு, பொருள் நிலை யாளுகை, உற்பத்திப் பெருக்கம், விற்பனையியல், சம்பளப் பட்டியல் தயாரித்தல், பொருள்தரம் சோதித்தல் முதலிய பணி களைக் கணிப்பொறி விரைவாகவும் துல்லியமாகவும் செய்து முடிக்கும் ஆற்றல் வாய்ந்தது. விமானம், மகிழ்வுந்து (கார்), கப்பல் முதலிய ஊர்திகளை உருவாக்கும் தொழிற்கூடங்களில் பாகங்களைப் பற்ற வைத்தல் [Welding], வண்ணம் பூசுதல் [Painting] முதலான பல பணிகளில் கணிப்பொறி பெருமளவு பயன்படுத்தப்படுகிறது. மனிதன் இவை போன்ற பணிகளில் ஈடு படும்பொழுது அவனது உயிருக்கே ஆபத்து ஏற்பட வாய்ப்புண்டு. விமானம், மகிழ்வுந்து முதலான ஊர்திகளைத்தொழில் ரீதியாகப் பயன்படுத்துமுன், அவற்றின் பயன்பாட்டுத் திறனைச் சோதித்து அறிதல் மிகவும் அவசியமாகும். இதற்குப் பொய்நிகழ்வு

[Simulation] என்று கூறப்படும் சோதனையைக் கணிப்பொறி மூலம் சிறப்பாகச் செயற்படுத்த இயலும்.

தற்காலத்தில் 'ரொபாட்ஸ்' [Robots] என்று கூறப்படும் எந்திர மனிதன், மனிதனைப் போன்றே கண்களால் பார்க்கும் திறன், ஒலியை உணர்ந்தறியும் சக்தி, மனிதனின் கை கால்களைப் போன்று அசைந்து செயல்படும் உறுப்புகள், தொட்டால் உணரும் சக்தி முதலியவற்றைப் பெற்றுள்ளது. இது மனிதனின் ஐம்புலன் களுக்கு நிகராக விளங்குகிறது. மேலும் தீ, புகை முதலான வற்றை முகர்ந்து உணரும் திறன் கொண்ட எந்திர மனிதர்களும் உண்டு. இவை அமெரிக்கா, சப்பான் முதலான நாடுகளின் தொழிற் கூடங்களில், பணியாள்களுக்குப் பதிலாக மிகச் சிறந்த முறையில் பணியில் ஈடுபடுத்தப்பட்டுள்ளன.

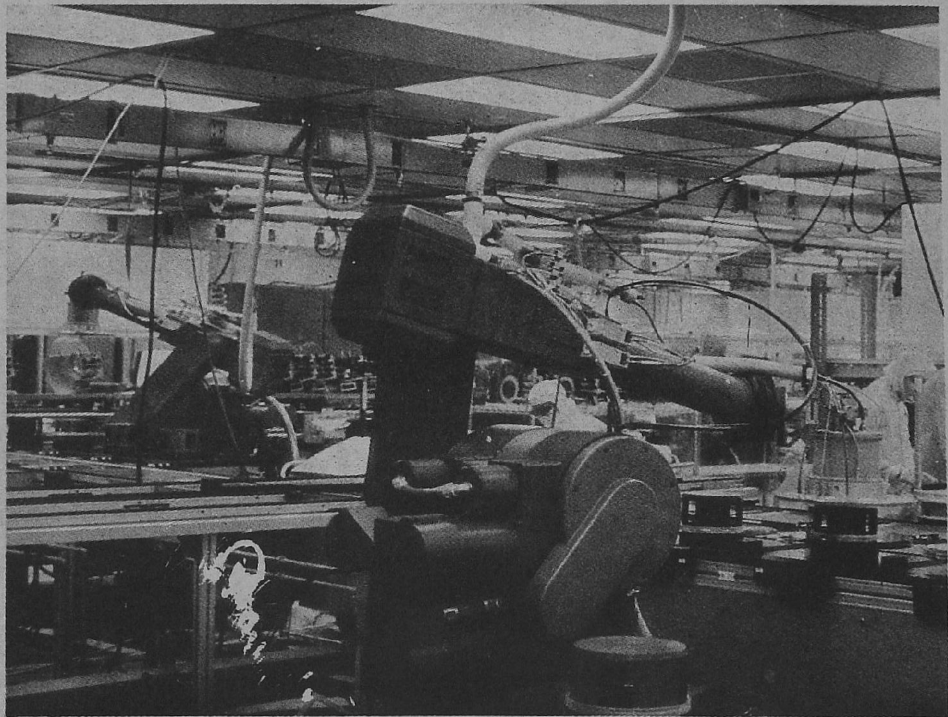
மனிதனின் உடல் நலத்திற்கு ஊறுவிளைவிக்கும் பணிகளான வண்ணம் பூசுதல் [Spray Painting], பாகங்களைப் பற்றவைத்தல் [Arc Welding] இராசாயனப் பொருள் உற்பத்தி, யுரேனியம் போன்ற கதிர்வீச்சு [Radiation] நிறைந்த தொழிற்கூடப்பணி ஆகியவற்றில் இந்த எந்திர மனிதர்கள் பெருமளவில் பயன் படுத்தப்படுகின்றன. மேலும், தொழிற்கூடப் பணித்தளத்தில் [Shop floor], பாகங்களை இணைப்பதற்காக ஓரிடம் விட்டு மற்றோர் இடத்திற்கு நகர்த்தி எடுத்தல், பளுவான பாகங்களை இடம்விட்டு இடம்நகர்த்துதல், உருவாக்கப்பட்ட பொருட்களைப் பெட்டிகளில் அடைத்தல் [Packaging] முதலான பணிகளில் எந்திர மனிதர்கள் பெரிதும் உதவுகின்றன. சப்பான் நாட்டில் பல தொழிற்சாலைகள் ஆள்களே இல்லாமல் எந்திர மனி தர்களின் துணையால் இயங்குவதை நாம் அறிவோம்.

1981-ஆம் ஆண்டு இறுதியில் மேற்கொள்ளப்பட்ட கணக்கெடுப்பின்படி, சப்பான் நாட்டில் சுமார் 70,000 எந்திர மனிதர்களும் அமெரிக்காவில் சுமார் 22,000 எந்திர மனிதர் களும் பணியில் ஈடுபடுத்தப்பட்டுள்ளதாகத் தெரிகிறது இவை யாவும் பெரிய அளவிலான குறிப்பிடத்தக்க பணிகளைச் செய்யும் திறன் பெற்ற எந்திர மனிதர்களாகும். இவையன்றி பற்பல துறைகளின் பயன்பாட்டிற்கென உலகெங்கணும் ஏராளமான சிறிய எந்திர மனிதர்களும் பயன்படுத்தப்படு கின்றன. கணிப்பொறிகளை வடிவமைத்து உருவாக்கல், கனி வள ஆய்வு, இதய அறுவைச்சிகிச்சை [Open Heart Surgery]

முதலியவற்றில் பயன்படும் எந்திர மனிதர்களை எடுத்துக் காட்டுகளாகக் கூறலாம்.

கணிப்பொறிகளை வடிவமைத்து உருவாக்கும் பணியில் ஈடுபட்டுள்ள எந்திர மனிதனைப் படம் 41-இல் காணலாம்.

மின்னணுவியல் துறையும் கணிப்பொறித் துறையும் கற்பனைக்கும் எட்டாத அளவில் வளர்ச்சி பெற்றுச் செல்வதை நாம் காண்கிறோம். மின்சுற்றுக்களையும் மின்நுண்கருவி களையும் கைகளினால் இணைத்து மின்னணுவியல் கருவிகள் உருவாக்கப்பட்ட நிலைமாறி இன்று நுணுக்கம் வாய்ந்த கருவி களனைத்தும் நுட்பம் பொருந்திய எந்திர மனிதர்கள், தொழிற் கூட எந்திரங்கள் முதலிய கருவிகளின் உதவியினாலேயே வடிவ மைத்து உருவாக்கப்படுகின்றன. இவற்றின் பயனாகக் கருவி களின் நுட்பமும் செயற்பாட்டுத் தன்மையும் மேலோங்கி நிற்பது மட்டுமன்றி அளவில் மிகவும் சிறியதாகவும் விலையில் மலிவா கவும் மின்னணுவியல்கருவிகள் கிடைக்கும் நிலை ஏற்பட்டுள்ளது. இவற்றின் வடிவமைப்பு நுட்பத்திற்கு எடுத்துக்காட்டாகக்கூறின் தற்காலத்தில் கணிப்பொறிகளில் பயன்படுத்தப்படும் RAM, ROM, MOS முதலிய நினைவகங்கள், கணிப்பொறியின் கருவியத்தில் பயன்படும் VLSI-Circuits எனப்படும் மின்சுற்றுக்கள் முதலிய வற்றில் ஒரு மி.மீ. அகலத்தில் பல்லாயிரக்கணக்கான மின்தாரை கள் செல்கின்றன. இம்மின் சுற்றுக்கள் Nano Seconds (அதாவது 1/100,00,00,000 விநாடி) வேகத்தில் செயற்படும் தன்மை வாய்ந்தவை. எனவே, தக்க தரக்கட்டுப்பாட்டுடனும் நுணுக்கத்துடனும் கணிப்பொறியின் பல மின்சுற்றுக்களை வடிவமைத்தல், கணிப்பொறியின் புற ஒருங்குகளான மின்காந்த நாடா, மின்காந்தத் தட்டு முதலியவற்றை வடிவமைத்து உரு வாக்கல், எண்ணிலக்கக் கடியாரங்கள் [Digital Watches] மற்றும் மின்னணுக் கணிப்பான்களைத் தயாரித்தல் முதலிய பணிகளில் மின்நுண் கருவிகளைத் தக்கவாறு இணைத்தல், செயற்பாட்டைச் சோதித்தல், தரக்கட்டுப்பாடு செய்தல், எண்ணிட்டு அவைகளை பெட்டிகளில் அடைத்தல் [Packing] முதலிய எல்லாப்பணிகளிலும் எந்திர மனிதன் பெரிதும் பயன்படுத்தப்படுகிறது.



படம்—41 கணிப்பொறியை வடிவமைத்து உருவாக்குவதில் எந்திரமனிதன்

விண்வெளியில் செலுத்தப்பட்டுச் சுற்றும் விண்கலங்களில் குறிப்பிட்ட பணிகளைத் தரைக்கட்டுப்பாட்டு நிலையத்தின் ஆணைக்கேற்பச் செயற்படுத்தவும் எந்திர மனிதர்கள் பெரிதும் துணை நிற்கின்றன. 1985 ஜூன் திங்களில் அட்லாண்டிக் பெருங்கடலில் விழுந்து நொருங்கிய இந்திய விமானத்தின் செய்தி பதியும் கருவியைச் சுமார் மூன்று கி. மீ. ஆழத்தில் கடலுக்கு அடியிலிருந்து தேடி எடுக்கவும் அதன் மூலம் விபத்தின் உண்மை நிலையைக் கண்டறியவும் எந்திர மனிதர்கள் பெருமளவு பயன் படுத்தப்பட்டன.

தற்காலத்தில் கணிப்பொறித்துறையில் குறிப்பிடத்தக்க பல ஆய்வுப்பணிகள் நிகழ்ந்து வருகின்றன. மனிதனைப் போன்று சிந்தித்துச் செயற்பட செயற்கை அறிவினைக் [Artificial Intelligence] கொண்டமைந்த ஐந்தாம் காலக்கட்டக் கணிப்பொறி களை உருவாக்குவதே இதன் முக்கிய நோக்கமாகும். குறிப்பிட்ட சில துறைகளின் அறிவு பூர்வமான செயற்பாடுகளைக் கணிப் பொறி மூலம் திறம்பட நிகழ்த்தும் பல அமைப்புகள் உருவாக்கப் பட்டுள்ளன. **இந்த** அமைப்பிற்கு Knowledge based System என்று பெயர். கணிப்பொறி வழி உருவாக்கப்பட்ட இந்த அமைப்பின் மூலம் கணிப்பொறி ஒரு மருத்துவ வல்லுநராகவோ, பொறியியல் அறிஞராகவோ, வழக்கறிஞராகவோ, கருவி களைப் பழுதுபார்க்கும் தொழில் வல்லுநராகவோ, கட்டிடக் கலைஞராகவோ சிறப்பான முறையில் செயலாற்ற இயலும்.

கணிப்பொறியானது தானாகச் சிந்தித்துச் செயற்படும் ஆற்றலற்றது என முன்பே கண்டோம். ஆனால், உயிரற்ற ஒரு கருவியான கணிப்பொறியை மனிதனைப் போன்று சிந்தித்துச் செயற்படச் செய்யவும் சூழ்நிலைக்கேற்பச் செயலாற்ற அறிவுறுத்தவும் வேண்டுமாயின் அதற்குப் போதிய அறிவினைச் செயற்கையாகப் புகுத்துதல் அவசியமாகும். இயற்கை மொழியில் மனிதன் இடும் கட்டளைகளைப் புரிந்துகொண்டு அதற்கேற்பச் செயற்படல், தானாகச் சிந்தித்து முடிவெடுத்தல், சூழ்நிலைகளின் காரணிகளைப் புரிந்து செயற்படும் செயற்கை

அறிவு முதலிய வசதிகளுடன் மின்னல் வேகத்தில் செயற்படும் செயல் வேகத்தையும் ஐந்தாம் காலக்கட்டக் கணிப்பொறிகள் பெற்றிலங்கும்.

கணிப்பொறியின் செயற்கை அறிவு பற்றிய ஆய்வு; அதாவது மனித மூளையின் செயலாற்றலைக் கருவியில் புகுத்தும் முயற்சி எனலாம். நமது கற்பனைக்கும் எட்டாத இப்பெரிய ஆய்வுப் பணித்திட்டத்தின் மூலம் அறிவுசார்ந்த கருவிகளை உருவாக்க அமெரிக்கா, சப்பான் முதலிய நாடுகளில் பல்லாயிரம் கோடி ரூபாய் பொருட்செலவில் ஆய்வுத் திட்டங்கள் செயற்படுத்தப்பட்டு வருகின்றன. 1984-ஆம் ஆண்டு அக்டோபரில் அறியப்பட்ட செய்திகளின்படி சப்பான் நாட்டில் மிட்டி [MITI – Ministry of International Trade and Industry]மற்றும் ஹிட்டாச்சி [Hitachi] முதலிய நிறுவனங்கள் ஏற்கனவே இந்தத் திட்டத்தில் 60 விழுக்காடுகள் வெற்றி பெற்றுள்ளன. இத்திட்ட ஆய்வு தொடர்ந்து நடைபெற்று வருகிறது. இத்திட்டங்கள் முழுவதும் வெற்றியடையுமாயின் புதினங்களில் விவரிக்கப்படும் கற்பனைக் கதைகளில் கருவிகள் நிகழ்த்தும் சாகசங்கள் அனைத்தும் நனவாகும் நாள் வெகுதொலைவில் இல்லை என உறுதியாகக் கூறலாம்.

6.6. தொலைதொடர்புத் துறையில் கணிப்பொறி (COMPUTERS IN TELE COMMUNICATION)

ஒரு நாட்டின் தொழில், கல்வி, சமூக நிலை, பண்பாடு ஆகியவற்றின் முன்னேற்றத்தில் செய்தித் தொடர்பு பெரும் பங்கு வகிக்கிறது. வானொலி, தொலைக்காட்சி, செய்தித் தாள்கள், தந்தி முதலான செய்தித் தொடர்புச் சாதனங்கள் உலகெங்கிலும் நிறைந்தும் பரந்தும் உள்ளன. தற்காலத் தொழில் வளர்ச்சியின் முன்னேற்றத்திற்கு ஈடுகொடுக்க வேண்டிய நிலையில், தொலை தொடர்புத் துறையின் முன்னேற்றம் மிகவும் அவசியமாகிறது. தொலை தொடர்புத்

துறையில் செய்திகளை [Messages] ஒழுங்குபடுத்துதல், செய்திகளை வாங்கி அனுப்புதல், பணிக்கணக்கீடு, அவசரகாலத் தொலைதொடர்பு முதலிய பல பணிகளில், கணிப்பொறி திறமையுடன் கடமையாற்றுகிறது. எடுத்துக்காட்டாகக் கூறின் ஒரு கணக்கெடுப்பின்படி அமெரிக்கா, சப்பான் முதலிய நாடுகளில் தொலை தொடர்பின் திறனை ஒவ்வொரு ஐந்து ஆண்டிற்கும் இரட்டிப்பாக்க வேண்டிய அவசியம் ஏற்பட்டுள்ளது. இதிலிருந்து தொலை தொடர்பின் இன்றியமையாமை நன்கு உணரப்படும். செயற்கைக்கோள் [Satellite] மூலம் தொலை தொடர்பு கொள்ளுதல், செய்திக் கட்டுப்பாடு செய்தல் முதலிய பணிகளுக்குக் கணிப்பொறி திறன்வாய்ந்த கட்டுப்பாட்டு நிலையமாக விளங்குகிறது.

6.7. போக்குவரத்துத் துறையில் கணிப்பொறி

இரயில், விமானம், கப்பல் முதலான போக்குவரத்துத் துறைகளில் கணிப்பொறி பெருமளவு பயன்படுத்தப்படுகிறது. உலகின் மக்கள் தொகைப் பெருக்கத்திற்கு ஏற்பவும் தொழில் மற்றும் பண்பாட்டுப் பரிமாற்றம் காரணமாகவும் போக்குவரத்தில் வேகமும் பாதுகாப்பும் அவசியமாகின்றன. எடுத்துக்காட்டாகப் பல்லாயிரம்அடி உயரங்களில் விண்வெளியில் பறக்கும் விமானம், தரைக்கட்டுப்பாட்டு நிலையத்துடன் தொடர்பு கொண்டு முறைப்படி விமானத்தளத்தில் இறங்குதலும்[Landing] அதே போன்று விமான நிலையத்தை விட்டுக்கிளம்பிச் செல்லுதலும் [Departure]கணிப்பொறியால் கட்டுப்படுத்தப்படுகின்றன. விமானம் வானில் பறக்கும்பொழுது விண்வெளியில் ஏற்படும் காலநிலை மாற்றம்,அதனால் ஏற்படும்அபாயம் முதலியவற்றைப் பற்றி எச்சரித்தலும் விமானத்திற்குத் தேவையான எரிபொருள் கையிருப்பு [Fuel Stock], பயணிகள் பற்றிய விவரங்கள் முதலானவற்றைப் பற்றித் தெரிவித்தலும் ஆகிய செயல்களை விமானவோட்டிக்கும் தரைக்கட்டுப்பாட்டு அலுவலர்களுக்கும் [Ground Control Officers]கணிப்பொறி ஒரு நொடியில் அறியத் தருகிறது. மேலும் பிற நாடுகளில் உள்ள விமான அனுமதிச் சீட்டுப் பதிவு செய்யும் [Airline Reservation] இடங்களுடன் தொடர்பு கொண்டு, பயணிகளுக்கு ஒரு சில நிமிடங்களில் இட ஒதுக்கீடு செய்து கொடுத்தலும் [Computerised Airlines Reservation System], பயணிகளுக்குத் தேவையான பிற வசதிகளைச் செய்து கொடுத்தலும் கணிப்பொறி மூலம் சிறப்பாகச் செயற்படுத்தப் படுகின்றன,

வானூர்த்தி இயக்குபவர் [Pilot] மனவலிமையும் அவசரத் திற்கு ஏற்பச் செய்திகளைப் புரிந்துகொண்டு செயற்படும் திறமை யும் வாய்க்கப் பெற்றவராக இருத்தல் மிகவும் அவசியமாகும். வானூர்தி இயக்குபவர்களுக்கு இத்துறையில் தேவையான பயிற்சி தரையிலேயே அளிக்கப்படுகிறது. இப்பயிற்சிக்குக் கணிப்பொறி மூலம் பொய்நிகழ்வு [Computerised Simulation] என்ற முறை பயன்படுத்தப்படுகிறது. இம்முறையின் மூலம் அதிகப் பொருட் செலவோ, பிற சேதமோ, பயமோ இல்லாமல் சீரியமுறையில் பயிற்சி அளிக்கப்படுகிறது. வானூர்தி இயக்குபவர்களுக்கு 'வானூர்திப் பொய் நிகழ்வு' [Flight Simulation Technique] என்ற முறை மூலம், அவர்கள் உண்மையாகவே வானில் விமானத்தை இயக்கிச் செல்வது போன்ற ஒரு சூழ்நிலை உரு வாக்கப்பட்டு, பயிற்சி அளிக்கப்படுகிறது. இப்பயிற்சியில் வானில் பறக்கும் பொழுது தெரிந்து கொள்ள வேண்டிய செய்திகளான தட்பவெப்ப நிலைமை, உயரம், வேகம், காற்று வீசும் திசை, காற்றழுத்தம், மழை, செய்தித் தொடர்பு முதலியவற்றிலும் அவை இயல்பாக நடப்பன போன்றே பயிற்சியளிக்கப்படுகிறது. பொருட் செலவோ, பயமோ இல்லாத இம்முறையில் திறம்படப் பயிற்சியளித்தல் கணிப்பொறியால் சாத்தியமாகிறது.

6.8. பொறியியல் துறையில் கணிப்பொறி

உலகெங்கிலும் பொருளாதார மேம்பாடு அடையவும் மக்களின் வாழ்க்கை நிலை உயரவும் பொறியியல் துறையே காரணமாக அமைந்துள்ளது. தொலை தொடர்பு, போக்கு வரத்து, கட்டிடக் கலை, கப்பல் கட்டும் துறை, தொழிற்கூடங் களில் பொருள் உற்பத்தி, கனிவளங்களைக் கண்டறிதல் மின்னியல் [Electrical], மின்னணுவியல் [Electronics] முதலிய பல துறைகள் பொறியியலில் அடங்கும். மேற்குறிப்பிடப்பட்ட பல பொறியியல் துறைகளிலும் கணிப்பொறியின் பயன் அளவிடற்கரியதாகும். தொழிற்கூடங்களில் சிறு ஆணி முதல் தொழில் நுட்பம் வாய்ந்த பல கருவிகள்வரை செய்யப்படு கின்றன. இவையாவும் செய்யப்படும் முன்னர் வடிவமைப்புச் செய்தலும் [Design], தேவைக்கேற்ப அதில் பல மாற்றங்கள் செய்தலும் வேண்டியுள்ளன. இதன் மூலமாகத்தான் குறைந்த மூலப்பொருள் செலவில் குறிப்பிட்ட காலக்கெடுவில் துல்லியமாக பல பொருட்களை உற்பத்தி செய்ய இயலுகிறது. எனவே இதுபோன்ற பணிகளின் ஒவ்வொரு நிலையிலும், கணிப்பொறி

மனிதனுக்குத் துணையாக நின்று செயற்படுகிறது. மின் உற்பத்தி செய்தலுக்குத் தேவையான மூலப்பொருட்கள் நீர் மற்றும் எரிபொருள் ஆகியவையாகும். எனவே, மின் உற்பத்தியில் அதற்குத் தேவையான கருவிகளை வடிவமைத்தல், மூலப் பொருள் பயன்பாட்டைக் கட்டுப்படுத்துதல், மின்சாரத்தைப் பயன்படுத்தும் முறைமை, தேவைக்கேற்றவாறு மின் விநியோகம் செய்தல் [Distribution of Electricity] முதலான பணிகளில் கணிப்பொறி பெருமளவு ஈடுபடுத்தப்படுகிறது. மின்சாரம் ஓரிடத்திலிருந்து மற்றோர் இடத்திற்குச் செல்லும் பொழுது ஏற்படும் போக்குவரத்து மின் இழப்பை [Line Loss] குறைப்ப தற்கும் கணிப்பொறி துணை செய்கிறது.

கட்டிடம் கட்டும் கலையில் [Architecture] நவீன முறையில் கட்டிடங்களை வடிவமைத்தல், அணைகள், பாலங்கள் கட்டுதல், கனிம வள ஆய்வு முதலியவற்றில் கணிப்பொறியின் பங்கு மகத்தானது. மின்னணுத்துறை நாள்தோறும் புதிய கண்டு பிடிப்புகளை உலகிற்கு அளித்த வண்ணம் இருக்கிறது. இந்தத் துறையில் பயன்படுத்தப்படும் நிறைவிலா மின்கடத்திகளைக் கொண்டு உருவாக்கப்பட்ட பல்மின் சுற்றுச் சேர்ப்பி [Chips], பல்லாயிரக்கணக்கான மின் இணைப்புகளை உள்ளடக்கிய பாகமாகும்.

மேலும் நுண்ணியக்கிகளைப் பயன்படுத்தி, அளவில் மிகச் சிறிய, ஆனால் செயலாற்றலில் பன்மடங்கு வேகங்கொண்ட சிறிய கணிப்பொறிகளை வடிவமைத்து உருவாக்குதலிலும் [Design and Fabrication], சோதித்தறிதலிலும் கணிப்பொறி பெரும் பங்கு வகிக்கிறது.

6.9. விண்வெளிப் பயணத்தில் கணிப்பொறியின்பங்கு

தற்காலத்தில் மனிதன் விண்வெளி ஆய்வில் நம்பவொண்ணாப் பல விந்தைச் செயல்களைச் செய்து வருகிறான். சந்திர மண்ட லத்தியல் கண்டு தெளியவும், விண்கோள்களை ஆய்ந்தறியவும் அவனுக்குத் துணைபுரியும் ஓர் இன்றியமையாச் சாதனம் கணிப் பொறியாகும் அமெரிக்க நாட்டில் 'நாசா' [NASA-National Aeronautics and Space Administration] எனப்படும் விண்வெளி ஆய்வகம் வெளியிட்டுள்ள குறிப்பின்படி, கணிப்பொறி இல்லை யேல் மனிதன் இத்தகைய பல விண்வெளி ஆய்வுகளை நிகழ்த்தி யிருத்தல் இயலாது என்பது உறுதி.

விண்வெளிப் பயணம் என்பது பல்லாயிரக்கணக்கான சிக்கலும் மிகவும் நுணுக்கம் வாய்ந்த தொழில் நுட்பமும் உள்ளடங்கியதாகும். எனவே, ஒரு விண்வெளிப் பயணம் மேற்கொள்ளும் முன்பாகத் திட்டங்கள் பல திறம்படத்தீட்ட வேண்டி உள்ளன. விண்வெளிப் பயணத்தின் அடிப்படை நோக்கம், விண்கோள் வடிவமைப்பு, வானில் மேற்கொள்ள வேண்டிய சோதனைமுறை, விண்கோளை ஏவுதலும் விண்வெளிப்பாதையில் அதை இடலும் கட்டுப்பாடுசெய்தலுமாகிய பணித்திட்டங்களைத் திட்டமிட்டு உருவாக்குதல் அவசியமாகும். மேலும் சோதனை நடைபெறும்பொழுது விவர ஆய்வனை செய்தலும் செய்தி. படங்கள் முதலானவற்றைப் பூமியிலுள்ள தரைக் கட்டுப்பாட்டு நிலையத்திற்கு [Ground Control Station] அனுப்புதலும் தொடக்கம் முதல் கடைசி வரையிலுள்ள அனைத்துப் பணிகளும் கணிப்பொறியின் துணை கொண்டே செய்யப் பெறுகின்றன. ஒரு விண்வெளிப் பயணத்தை மேற்கொள்ளும்பொழுது கணிப்பொறியில் முன்பாகவே பதிவு செய்யப்பட்ட ஆணைகளின்படி [Preplanned instructions] பயணம் சிறிதும் முறை பிசகாது அமைவதற்குக் கணிப்பொறி துணை செய்கிறது. விண்கோள் சில மணித்துளிகளில் ஆயிரக்கணக்கான மைல் தொலைவைக் கடந்து செல்லும்பொழுது குறிப்பிடப்பட்ட பணிகள் குறிப்பிட்ட நேரத்தில் மிகத் துல்லியமாகச் செய்து முடிக்கப் பெறுகின்றன. கணிப்பொறியின் துணையின்றி மனிதனால் இப்பணியைச்செய்து முடித்தல் என்பது கனவிலும் நினைக்கவொண்ணா ஒன்றாகும்.

இவைபோன்ற பணிகளில் இருவகையான கணிப்பொறிகள் பயன்படுத்தப்படுன்றன. ஒன்று, பூமியில் தரைக்கட்டுப்பாட்டு நிலையத்திலிருந்து விண்கோளுடன் தொடர்புகொண்டு பணி புரிகிறது. மற்றொரு வகைக் க்ணிப்பொறி விண்கோளில் பொருத்தப்பட்டு, தனக்கிடப்பட்ட பணியைத் திறம்படச் செய்யப் பயன்படுகிறது. இரண்டாம் வகைக் கணிப்பொறி, முதல்வகைக் கணிப்பொறியைக் காட்டிலும் அதிக நினைவகம் உடையது. மேலும் இது தன் உறுப்புகளில் பழுதேற்படா வண்ணம் கண்காணித்துப் பாதுகாக்கவும் அவ்வாறு ஏதாவது ஒரு பாகம் பழுதடைந்தால் விண்கோளிலுள்ள பதில் கணிப்பொறியிடம் [Stand by Computer] தனது பணியை மாற்றிக் கொடுத்து விண்வெளிப் பயணக் குறிக்கோளில் இடையூறு நேரா வண்ணம் செயற்படும் திறனும் வாய்க்கப் பெற்றது. மேலும்

தரைக் கட்டுப்பாட்டு நிலையத்தின் துணையின்றித் தனித்து நின்று விண்கலத்தின் இயக்க நிலையைக் கண்காணித்தல், பயணக் கணிப்பு, விண்பாதை மாற்றம், விண்கல வேகக்கட்டுப்பாடு ஆகிய பணிகளைச் செவ்வனே செய்கிறது. மேலும் விண்பாதையில் விண்கலத்தை ஏவுவதற்கான தீர்வு செய்தல், தரைக் கட்டுப்பாட்டு நிலையத்தின் ஆணையை ஏற்றுச் செயற்படுதல் முதலானவை இக்கணிப் பொறியின் சிறப்பு அம்சங்களாகும். அமெரிக்காவிலிருந்து விண்வெளிக்கு அனுப்பப்பட்ட அப்போலோ [Appollo Space Shuttle] இந்தியாவிலிருந்து அனுப்பப்பட்ட பாஸ்கரா, எஸ்.எல்.வி, ரோகிணி முதலிய விண்கலங்களை எடுத்துக்காட்டுகளாகக் கூறலாம்.

6.10. பாதுகாப்புப் படையில் கணிப்பொறி

பாதுகாப்புப் படையில் கணிப்பொறியின் பணி மிகவும் மகத்தானது. மிகப் பலம் வாய்ந்த அணு குண்டுகளும் கண்டம் விட்டுக் கண்டம்பாயும் ஏவுகணைகளும்[ICBM-Inter Continental Ballistic Missiles] தற்காலப் பாதுகாப்புப் படைகளில் மிகவும் புழக்கத்தில் உள்ளன. இக்காலத்தில் பாதுகாப்புப் படையை உரிய முறையில் எதிரியின் திறனறிந்து ஏற்ற சமயத்தில் பயன்படுத்தலே சிறந்த பாதுகாப்புப் படைக்கு அழகாகும். இதன் மூலமே பல்லாயிரம் சதுர மைல் நிலப்பரப்பைத் தகர்த்து எதிரணியினரை வீழ்த்தி வெற்றி காணல் இயலும். ஒரு நாட்டின் பாதுகாப்பினுடைய அவசியம் கருதி பாதுகாப்புப் படைப் பணிகளில் கணிப்பொறியின் பயன் மிகவும் இன்றியமையாததாகிறது. அமெரிக்கா, உருசியா முதலிய நாடுகள் தங்கள் கடற்படை, வான்படை, தரைப்படைப் பிரிவுகளில் கணிப்பொறியைப் பெரிதும் நம்பியிருக்கின்றன. அமெரிக்க ஐக்கிய நாடுகள் மிகச் சிறந்த கணிப்பொறி இணைப்பு [Computer Network] மூலம், உலகின் பல பாகங்களில் நிறுவப்பட்டுள்ள வல்லரசுகளின் படைத் திறனை உடனுக்குடன் அறிந்துகொண்டு கண்காணிக்கமுடிகிறது.

நாட்டின் பல பாகங்களில் உள்ள பாதுகாப்பு நிலையங்களைக் கணிப்பொறி மூலம் இணைத்துச் செயலாற்றும்பொழுது உடனுக்குடன் அயல்நாட்டுப் படையின் திறனறிதலும் நாட்டின் எல்லையைக் கண்காணித்தலும் உள்நாட்டு நிலைமையைக் கண்டறிந்து அதற்கேற்பப் பாதுகாப்புப் படைகளைப் பயன்படுத்துதலும் சாத்தியமாகின்றன. எடுத்துக்காட்டாகக் கூறின்,

அமெரிக்காவில் 'ரேடார்' [Radar] என்ற கருவி ஒவ்வொரு குறிப்பிட்ட வான்பரப்பையும் கண்காணிப்பு செய்கிறது. ஏதாவதொரு விமானம் வரும்பொழுது அதுபற்றிய செய்தியை 'ரேடார்' கணிப்பொறிக்கு அனுப்புகிறது. அது தன்னிடம் முன்பே பதிவு செய்யப்பட்டுள்ள செய்திக்கூறுகளை [Stored Data] கொண்டு ஆய்வு செய்து, ஒரு சில விநாடிகளில் முடிவை அறிவிக்கிறது. முடிவின்படி எதிரியின் விமானமாய் இருப்பின், மேற்கொண்டு அதைத் தாக்கும் பொருட்டு நடவடிக்கை எடுக்கவேண்டி, தரைக் கட்டுப்பாட்டு நிலையத்தை எச்சரிக்கிறது. சில சமயங்களில் அதிகாரிகள் விரும்பினால், தானாகவே இந்த முடிவைக்கூட கணிப்பொறி தீர்வு செய்து கட்டுப்பாட்டு நிலையத்திலுள்ள ஏவுகணைகளைத் தெரிந்தெடுத்து, குறிப்பிட்ட எதிரியின் விமானத்தை நோக்கி அனுப்பி அதைச் சுட்டு வீழ்த்தவும் செய்கிறது.

வான்வெளிக் கண்காணிப்பு [Air Vigilance] மட்டுமன்றி, கடற்படை மற்றும் வான் பாதுகாப்பு [Naval and Space Surveillance] முதலான பல பாதுகாப்பு-நிலையங்களிலும் பெறப்படும் ஏராளமான செய்திகளைச் சேகரித்து வகைப்படுத்தி, ஆய்ந்து, தீர்வு செய்து உரிய நேரத்தில் உரிய நடவடிக்கை எடுக்க உதவுவது கணிப்பொறியாகும். இதன் மூலம் வான்வெளிக் கண்காணிப்பு, உலகம் முழுவதும் உள்ள பிறநாடுகளின் படை பலனறிதல், செயற்கைக்கோள் நிலையறிதல் முதலிய பல சாதனைகளை, மிக விரைவில் உரிய நேரத்தில் செயலாற்ற உதவி புரிவது கணிப்பொறி என்பதில் சிறிதும் ஐயமில்லை.

6.11. காவல்துறைச் செயற்பாட்டில் கணிப்பொறியின் பங்கு

பிற துறைகளில் கணிப்பொறி எங்ஙனம் பெருந்துணை புரிந்து வருகிறதோ அது போன்றே, காவல்துறையும் சிறப்பாகக் கடமையாற்ற கணிப்பொறியின் தேவை அவசியமாகிறது. திருட்டு, கொள்ளை, கொலை, கடத்தல் முதலிய வன்செயல்கள் சமூகத்தில்நாள்தோறும் பெருகி வருகின்றன. இவற்றைக் காவல் துறையினர் தகுந்த முறையில் கண்காணிக்கவும் குற்றங்கள் நிகழாமல் தடை செய்யவும் அறிவியற் கருவிகளின் துணை இன்றியமையாத் தேவையாகும். எடுத்துக்காட்டாகக் கூறின், கடத்தல்

தொழிலில் ஈடுபட்டுள்ள நாட்டு விரோதிகளைக் கண்காணிக்கவும் அவர்களைப் பிடித்துத் தக்க முறையில் தண்டனை கொடுத்துப் பின்னர் திருத்தி நன்குடிமக்களாக்குவதற்கும் கணிப்பொறி உதவிசெய்கிறது.

நாட்டின் மையப் பகுதியில் ஒரு பெரிய கணிப்பொறியை நிறுவி, பல சிறிய கணிப்பொறிகளை நாட்டின் மற்ற பகுதிகளில் அமைத்து, பின்னர் அவைகளை இணைத்து அவைகளுக்குள் தொடர்பு வசதியை [Computer networking and Interactive operation] ஏற்படுத்துதல் வேண்டும். இதன் மூலம் கொலை, கடத்தல், கொள்ளை முதலிய வன்செயல்கள் நடைபெறும் பொழுது, காவல்துறையினர் மிகவும் விழிப்பாக இருந்து, செய்திகளைக் கணிப்பொறி மூலம் நாட்டின் பிற பகுதிகளுக்கு உடனுக்குடன் அனுப்பிக் குற்றவாளிகள் தப்பிச் செல்லாவண்ணம் தடை செய்ய இயலுகிறது. மேலும் பல்லாயிரக்கணக்கான குற்றவாளிகளின் பெயர்கள், அடையாளம், முகவரி முதலான செய்திகளைக் கணிப்பொறியில் பதிவு செய்து வைத்துக் கொள்வதன் மூலம், குற்றவாளிகள் எளிதாக இனம் கண்டு கொள்ளப்பட்டுத் தண்டனை வழங்கப்படுகிறார்கள். திருட்டு, கொலை முதலான வன்செயல்கள் நடைபெற்ற இடங்களிலிருந்து குற்றவாளிகளைக் கண்டுபிடிக்கவும் குற்றப் புலனாய்வுத்துறை [Crime Investigation Department] இரகசியப் போலீஸ் துறைகளில் குற்றவாளிகளைக் கண்டறிந்து தண்டனை வழங்கவும் கணிப்பொறி பெருந்துணை செய்கிறது.

6.11.1. கணிப்பொறியில் கைரேகைப் பகுப்பாய்வு (*THUMB IMPRESSION ANALYSIS*)

ஒவ்வொரு மனிதனின் கைரேகையும் மற்ற மனிதர்களின் கைரேகைகளிலிருந்து மாறுபட்டிருக்கிறது. எனவே, கோடிக்கணக்கான மனிதர்களின் கைரேகைகளைப் பதிவு செய்து வைத்துக்கொண்டு, அவைகளைக் கணிப்பொறி மூலம் மிகவும் கவனமாக ஒப்புநோக்கின் மேற்கூறிய உண்மை பெறப்படும். கணிப்பொறியின் துணையில்லையேல் இப்பணியைச் செய்தல் மிக மிகக் கடினமாகும். கொலை, திருட்டு முதலிய வன்செயல்களில், அவை நடைபெற்ற இடங்களில் பெறப்பட்ட தடயங்களே பின்னர் குற்றவாளிகளைக் கண்டுபிடிக்க பெரிதும் பயன்படு

கின்றன. இக்குற்றச் செயல்களைத் திரும்பத் திரும்பச் செய்யும் குற்றவாளிகளின் பெயர், முகவரி, கைரேகை முதலான விவரங்கள் முன்பே கணிப்பொறியில் பதிவு செய்யப்பட்டுள்ளன. இவற்றின் மூலம் கணிப்பொறியில் குற்றவாளிகளின் கைரேகை களை ஒப்புநோக்கி, குற்றவாளிகளைப் போலீசார் சுலபமாகக் கண்டுபிடிக்கின்றனர். படப் பகுப்பாய்வு [Image or Picture Analysis] என்ற முறையைக் கணிப்பொறியில் பயன்படுத்தி, மேற்குறிப்பிடப்பட்ட செயல்கள் செய்யப்படுகின்றன.

நாட்டின் பல பகுதிகளிலும் உள்ள நீதி மன்றங்கள் பல்லாயிரக்கணக்கான வழக்குகளை நாள்தோறும் விசாரணை செய்து நீதி வழங்கி வருகின்றன. இருப்பினும் நீதி மன்றத்தில் பல்லாயிரக்கணக்கான வழக்குகள் பல்லாண்டுகளாக நீதி விசாரணைக்கு எடுத்துக்கொள்ளப்படாமலேயே இருப்பதை அறிகிறோம். ஆகவே, இதுபோன்ற துறைகளில் வழக்கறிஞர் களுக்கு உற்ற துணையாக நின்று பெரும் சேவை புரிய, கணிப் பொறி காத்திருக்கிறது. எந்தெந்த வழக்குகளை எந்தெந்த காலத்தில் வழக்கு விசாரணைக்கு எடுத்துக் கொள்ளலாம் என்பதிலிருந்து, வழக்குக்கேற்ற சரியான தண்டனை வழங்கு வதற்கு வழக்கறிஞர்களுக்குத் துணை செய்வதுவரை உள்ள எல்லாப் பணிகளிலும் கணிப்பொறியின் சேவை மிகவும் தேவை யாகும். மிக முக்கியமான சில வழக்குகளையும், வழக்கு வரலாற்றுச் சுருக்கம் [Case History], வழக்குக்கேற்ற தண்டனை ஆகிய விவரங்களையும் மிகுந்த அனுபவமும் திறமையும் வாய்ந்த வழக்கறிஞர்களைக்கொண்டு தொகுத்து, கணிப்பொறியில் பதிவு செய்து வைத்துக்கொள்வதன் மூலம், மேற்குறிப்பிட்ட பல பணிகளையும் செப்பமுறச் செய்யலாம் என்பது திண்ணம்.

6.12. வங்கிகள் மற்றும் ஆயுட்பாதுகாப்பு நிறுவனங்களில் கணிப்பொறி

உலகில் தொழில்துறை முன்னேற்றம் காரணமாக வங்கிகள் ஆயுட் பாதுகாப்பு நிறுவனங்கள், தொழிற்கூடங்களுப் பண உதவி செய்யும் நிறுவனங்கள் [Industrial Finance Corporations] முதலியன பெருகி வளர்ந்துள்ளன. இவற்றில் புதிய தொழில் தொடங்கக் கடனளித்தல், சேமிப்பு மற்றும் வரவு செலவுக் கணக்குகள் முதலான பல பணிகள் நடைபெறுகின்றன. இவ்வித மான பணிகள் பெரும்பாலும் ஒரு குறிப்பிட்ட முறையில் ஒரே

வேலையைத் திரும்பத் திரும்பச் செய்யும் தன்மையையொத்தன வாகும். மேலும் இவைபோன்ற பணிகளில் பல்லாயிரக்கணக் கான அலுவலர்களை வேலையில் அமர்த்தியும்கூட வங்கி, ஆயுட் பாதுகாப்பு நிறுவன்ங்களில் [Bank, L.I.C.] நிருவாகத்தைச் சிறந்த முறையில் நிருவகிக்க முடிவதில்லை. அதுமட்டுமன்றி ஓரேபணியைத் திரும்பத்திரும்பச் செய்யும்பொழுது அலுவலர்கள் சோர்வு பெறுவதால் கணக்கில் பிழை ஏற்பட ஏதுவாகும். இதற் கெனக் கணிப்பொறி மூலம் குறிப்பிட்ட நெறிமுறையை [Procedures] உண்டாக்கினால், பல இலட்சக்கணக்கான மக்கள் திறம்பட வங்கி, ஆயுட்பாதுகாப்பு நிறுவனங்களின் சேவையைப் பெறமுடியும். இதன்மூலம் கணிசமான அளவு கால விரயமும் பொருள் விரயமும் தவிர்க்கப்படுகின்றன.

வங்கிகளில் காசோலை [Cheque], பணவரைவு [Demand Draft] முதலியவைகளைத் தாமாகவே படித்துணர்ந்து நினை வகத்தில் பதிவு செய்து வைக்கத் தற்காலக் கணிப்பொறிகளால் இயலும். இதற்கெனக் காந்தமைக் குறியீடு அறிமுறை [MICR-Magnetic Ink Character Recognition Technique] பயன்படுத்தப் படுகிறது.

இம்முறையில் பொருட் செலவும் காலச் செலவும் குறிப்பிடத் தக்க அளவில் மீதப்படுத்தப்படுகின்றன. தொழில் நிறுவனங் களும் வங்கிகளும் கணிப்பொறி மூலம் இணைக்கப்படும்பொழுது செய்திக்கூறுகளை நேரடியாக அனுப்புவதற்கும் அதிக வேலைப் பளுவின்றி சம்பளப்பட்டியல் முதலியன தயாரிப்பதற்கும் இயலுகிறது.

கணிப்பொறியின் பயன்கள் எண்ணிறந்தவை. தக்கமுறையில் வழியமைப்புகளை உருவாக்கிப் பயன்படுத்தினால் பல்லாயிரம் மணி நேரங்கள் மனிதர்கள் செய்யக்கூடிய பணியைச் சில மணி நேரங்களில் துல்லியமாகவும் சிறப்பாகவும் செய்து முடிக்கும் ஆற்றல் வாய்ந்தது கணிப்பொறி என்பதை முன்பே கண்டோம். கணிப்பொறி இன்றைய நவீன உலகில் எல்லாப் பணிகளுக்கும் இயைந்த கருவிஎனினும் இங்குக் குறிப்பிட்ட சில தலைப்புகளான கல்வி, மருத்துவம், வேளாண்மை, தொழில், தொலைதொடர்பு, போக்குவரத்து, பொறியியல் விண்வெளிப்பயணம், பாதுகாப்புப் படை, காவல்துறை, வங்கிகள் முதலியவற்றைப்பற்றி இதுவரை சுருக்கமாக விவரித்தோம். மின்னணுத்துறையின் வெகு வேக

வளர்ச்சி காரணமாக கணிப்பொறித்துறை முன்னேற்றம் காணவும் அதனடிப்படையில் விலையிற்குறைந்த சிறிய மற்றும் மிகச்சிறிய திறன்வாய்ந்த கணிப்பொறிகள் பல உருவாகவும் வழி ஏற்பட்டுள்ளன. பற்பல துறைகளின் வளர்ச்சிக்கும் பயன் பாட்டிற்கும் இக்கணிப்பொறிகள் உதவிபுரிந்து பொருளாதாரம் மற்றும் சமூக வாழ்வில் குறிப்பிடத்தக்க மறுமலர்ச்சியை உண்டாக்கும் என்பதில் சிறிதும் ஐயமில்லை.

பாகம் – 7

7. பேஸிக்மொழி (*BASIC LANGUAGE*)

7.1. முன்னுரை

மனிதன் தன் எண்ணங்களையோ செய்திகளையோ பிறருக்குத் தெரியப்படுத்தவும் தனது கருத்துகளைப் பிறருடன் பகிர்ந்து கொள்ளவும் மொழி ஒரு இன்றியமையாக் கருவியாக விளங்குகிறது. வாழ்வின் நாகரிகம், பண்பாடு, அறிவியல் வளர்ச்சி ஆகிய அனைத்தும் முன்னேற்றம் காண, மொழி மிகவும் அவசிய மான சாதனமாக அமைகிறது. மனிதனையும் விலங்குகளையும் வேறுபடுத்திக் காட்டும் தன்மை பொருந்திய உயரிய சாதனம் மொழி எனில், அது மிகையாகாது. உலகில் பலதரப்பட்ட இயற்கை மொழிகள் [Natural Languages] பேசப்பட்டு வரு கின்றன. நமது பாரத நாட்டில் நூற்றுக்கும் மேற்பட்ட மொழி கள் பேசப்பட்டு வருவதை நாம் அறிவோம். ஒரு மொழியின் அமைப்பும் அதன் தன்மையும் அதற்கென அமைந்துள்ள இலக்க ணத்தால் பெறப்படும். எனவே, ஒவ்வொரு மொழியும் மற்ற மொழிகளிலிருந்து வேறுபட்டுத் தனித்தன்மை வாய்ந்ததாக விளங்குகிறது. கணிப்பொறி என்பது ''மனிதர்களால் உரிய முறையில் கட்டளையிடப்படும் பொழுது கொடுக்கப்பட்ட பணியை விரைவாகவும் துல்லியமாகவும் செய்து முடிக்கும் ஆற்றல் வாய்ந்தது; தானாகச் சிந்தித்துச் செயற்படும் திறனற்ற, மனித னின் அடிமை'' என வரையறுத்துக் கூறலாம் என்பதை முன்பே பார்த்தோம்.

மேற்குறிப்பிட்டவாறு கணிப்பொறி மனிதனால் கட்டளை யிட்டபடி மட்டுமே பணி செய்ய இயலுமென்றால், அதற்கு எவ்வாறு கட்டளையிடுவது? என்ற வினா எழுகிறது. நாம் பயன்படுத்துவது போன்ற இயற்கை மொழியில், உயிரற்ற கருவி யான கணிப்பொறிக்குக் கட்டளையிட முடியுமா? அவ்வாறு கட்டளையிட்டால் அந்த மொழியைக் கணிப்பொறியால் புரிந்து கொண்டு செயல்பட முடியுமா? என்ற பல ஐயப்பாடுகள் ஏற்படு கின்றன. இயற்கை மொழிகளைப் போன்றே கணிப்பொறிகளுக் கென பலதரப்பட்ட கணிப்பொறி மொழிகள் உள்ளன. அவை ஒவ்வொன்றும் மொழிக்குரிய எல்லா அமைப்புகளையும் வகுத்து முறைப்படுத்தப்பட்ட இலக்கண அமைப்பையும் பெற்றுத் திகழ் கின்றன. கணிப்பொறிக்கு உரியமுறையில் கட்டளையிட வேண்டு மாயின், முதலில் கணிப்பொறி மொழியை நன்கு கற்றுணர்ந்தறி தல் அவசியமாகும். கணிப்பொறி மொழியில் கட்டளையிட்டு எவ்வாறு ஒரு கணக்குக்குத்தீர்வு காண்பது என்பதைக் கீழ்க்கண்ட வாறு விளக்கலாம்.

நாம் ஒரு கணக்குக்குத்தீர்வு காண வேண்டுமாயின், முதலில் அந்தக் கணக்கை நன்கு படித்து அறிந்துணர்ந்து ஆய்வு செய்ய வேண்டும். பின்னர் அதற்குத் தேவையான செய்திக்கூறுகளைச் சேகரிக்க வேண்டும். அக்கணக்குக்குத்தீர்வு காணும் முறைகளைப் படிப்படியாக வகுத்து முறைப்படுத்த வேண்டும். இந்தச் செயற் பாட்டு முறை, செய்முறை நிருணயம் என்றழைக்கப்படுகிறது. இவ்வாறு செய்முறை நிருணயம் செய்யப்பட்ட பிறகு, இதை வழியமைப்புச் செய்வழி வரைவாக [Programming Flow Chart] வரைந்துகொள்ள வேண்டும். அதன் பின்னர் கணக்கைத் தீர்வு காணும் பொருட்டு முன்பே வழிமுறைப்படுத்தப்பட்டபடி, இக் கணக்கின் செய்முறை நிருணயத்தை வழியமைப்பாக மாற்றி எழுதுதல் வேண்டும்.

வழியமைப்பு என்றால் என்ன? இவ்வினாவிற்கு விடை காணும் முன், வழியமைப்பு ஆணை என்றால் என்ன என்று பார்ப்போம். முன் குறிப்பிட்டபடி ஒரு கணக்குக்குத் தீர்வு காணும் பொழுது படிப்படியாக செய்முறைகள் வகுக்கப்படு கின்றன. ஒவ்வொரு நிலையிலும் ஒரு குறிப்பிட்ட செயற்பாடு [Activity] நிகழ்கிறது. இவ்வாறு உள்ள கணக்கியல் செயற் பாட்டினை ஒரு வழியமைப்பு ஆணையாகக் குறிப்பிடலாம். இங்ஙனம் எழுதப்பெற்ற பல வழியமைப்பு ஆணைகளின்

தொகுப்பே வழியமைப்பு என்பெயர் பெறும். வழியமைப்பு ஆணைகள் யாவும் கணக்கைத் தீர்வு செய்வதற்கான செய்முறைகளின்படி வரிசை முறையாகத் தொகுத்து வழியமைப்பாக எழுதப்படுகின்றன எனவே நமது கணக்குக்குத் தீர்வு காண்பதற்கு, கணக்கை ஒரு வழியமைப்பு மொழி மூலம் வழியமைப்பாக மாற்றுதல் அவசியமாகும்.

எடுத்துக்காட்டாக, செல்வன் கடையில் நான்கு எழுதுகோல்கள் வாங்குகிறான் என்று வைத்துக் கொள்வோம். ஒவ்வொரு எழுதுகோலின் விலையும் ரூ. ஒன்று மற்றும் ஐம்பது பைசாக்கள் எனின், மொத்தம் அவன் எவ்வளவு செலவழித்திருப்பான்?

இது ஒரு சிறிய கணக்கே எனினும் கணிப்பொறி வழி இதற்குத் தீர்வுகாணவேண்டுமெனில், கணக்கை நன்கு படித்தறிந்துணர்தல், கணிதவழிச் சமன்பாடாக்கல், வினைக்கட்ட வரைவாக்கம் செய்தல் [Programming flow Chart], கணிப்பொறி மொழியில் கணக்கின் தீர்வு நிலை [Steps] ஒவ்வொன்றையும் ஓர் ஆணை மூலம் குறிப்பிடுதல் அவசியமாகும். பின்னர் இந்த ஆணைகள் யாவும் குறிப்பிட்ட வரிசை முறையில் தொகுக்கப்பட்டு வழியமைப்பாக எழுதப்படுகின்றன.

மேற்கண்ட எடுத்துக்காட்டைச் சமன்பாடாகக் குறிப்பிடின்,

$$A = B \times C$$

A மொத்தச் செலவு

B எழுதுகோல்களின் எண்ணிக்கை

C ஒரு எழுதுகோலின் விலை

பொதுவாக ஒரு கணக்குக்குத் தீர்வு காணத் தேவையான மேற்குறிப்பிட்ட முறைகளின்படி $A = B \times C$ என்ற சமன்பாட்டைக் குறிப்பிட்ட கணிப்பொறி மொழியில் தகுந்த ஆணைகளைக் கொண்டு, குறிப்பிட்டு வழியமைப்பாக உருவாக்கலாம்.

வழியமைப்பு ஆணைகள் குறிப்பிட்ட கணிப்பொறி மொழியில் அமைதல் வேண்டும். வழியமைப்பை உருவாக்கும் முன்பாக, நமது பயன்பாட்டிற்கேற்பச் சிறந்த கணிப்பொறி மொழியைத் தேர்வு செய்தல் வேண்டும். ஏனெனில், நூற்றுக்கணக்கான

கணிப்பொறி மொழிகள் தற்காலத்தில் புழக்கத்தில் உள்ளன. குறிப்பிட்ட கணிப்பொறியில் உள்ள மொழிகளில் நமது கணக்கின் தன்மைக்கேற்பத் தேவையான மொழியைத் தேர்வு செய்ய வேண்டும். எடுத்துக்காட்டாகக் கூறின், உயர்நிலை மொழிகளான கோபால்மொழி வணிகப்பயன்பாடுகளுக்கும்[Commercial Applications], போர்ட்ரான் மொழி அறிவியல் பயன்பாடு களுக்கும் பேஸிக் மொழி மேற்கண்ட இருவகைப் பயன்பாடு களுக்கும் ஏற்ற கணிப்பொறி மொழிகளாகக் கருதப்படுகின்றன. கணிப்பொறி மொழியை அதன் தன்மையைப் பொறுத்து, உயர் நிலை மொழி [Higher Leval Language], இடைநிலை மொழி [Assembly Language], பொறி மொழி [Machine Language] என மூன்று வகைப்படுத்தலாமென்பதை முன்பே கண்டோம். பொறி மொழியில் வழியமைப்புச் செய்வது வழியமைப்பாளர் களுக்குச் சற்றுக் கடினமான பணியாகும், ஆனால் உயர்நிலைக் கணிப்பொறி மொழிகளான பேஸிக், போர்ட்ரான், கோபால் ஆகியவற்றில் வழியமைப்புச் செய்தல் எளிதாகும்.

பேஸிக் மொழியின் தன்மை, அதன் சிறப்பியல்புகள் அதன் வழியமைப்பு ஆணைகள் [Basic Programming Instructions], மற்றும் பேஸிக் மொழியில் வழியமைப்புச் செய்தல் ஆகியவற்றை பற்றி இங்குச் சிறிது விரிவாக ஆராய்வோம்.

7.2. பேஸிக் வழியமைப்பு மொழி (BASIC PROGRAMMING LANGUAGE)

பேஸிக் [BASIC] மொழி என்பது Beginners All Purpose Symbolic Instruction Code என்பதன் சுருக்கமாகும். இது கணிப்பொறி மொழியை முதன் முதலில் கற்பவருக்கு எளிமை யாகவும் சுலபத்தில் புரிந்துகொள்ளக்கூடிய வகையிலும் அமைந்த மொழியாகும்.

இந்த மொழி 1964—ஆம் ஆண்டு அமெரிக்காவிலுள்ள டார்ட் மௌத் கல்லூரிப் பேராசிரியர்களான [Dartmouth College, New Hampshire, U.S.A.] கெம்னி [Kemny], குர்ட்ஸ் [Kurtz] என்பவர்களால் உருவாக்கப்பட்டது. இம்மொழி பெரும்பாலும் நேரப்பகிர்வுச் செயற்பாட்டு முறையில் செயல் படும் வகையில் முதன்முதலில் அமைக்கப்பட்டது. அதன்பின்னர், இம்மொழியில் பலவிதமான மாற்றங்களும் திருத்தங்களும்

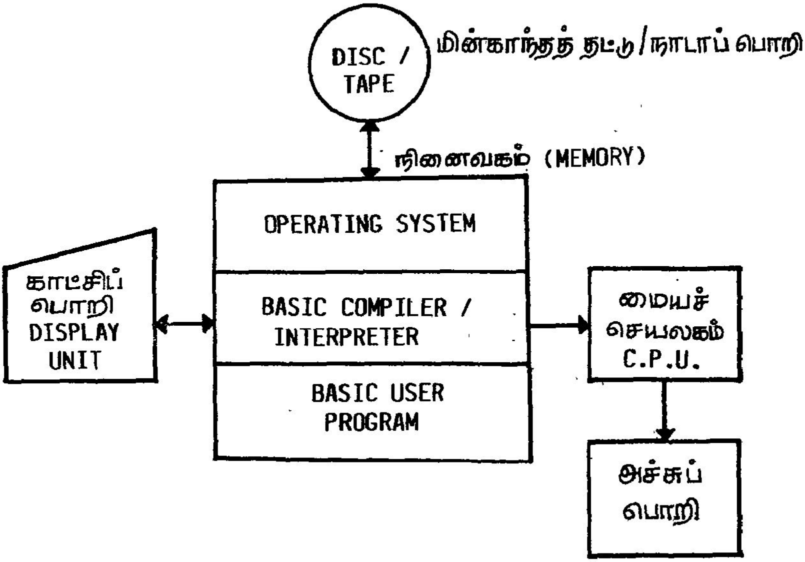
செய்யப்பட்டு, புதிய வழியமைப்பு ஆணைகள் பல சேர்க்கப் பட்டன. மேலும் இம்மொழி ஒவ்வொரு வகைக் கணிப்பொறிக் கும் ஏற்பச் சற்று வேறுபட்ட முறையில் அமைந்துள்ளது. இதன் காரணமாக ஒரு குறிப்பிட்ட இனக் [Model] கணிப்பொறியில் பயன்படும் பேஸிக் மொழிக்கும் மற்றவகைக் கணிப்பொறிகளில் பயன்படும் பேஸிக் மொழிக்கும் குறிப்பிடத்தக்க வேறுபாடுகள் உள்ளன.

தற்காலத்தில் பெரிய கணிப்பொறிகளைவிட, சிறிய மற்றும் மிகச் சிறிய கணிப்பொறிகள் பெருமளவு பயன்பாட்டுக்கு வந்துள்ளதால், பேஸிக் மொழியின் பயன்பாடும் அதிகமாகி யுள்ளது. நுண்ணியக்கிகளைக் கொண்டு உருவாக்கப்படும் மிகச் சிறிய கணிப்பொறிகள் அனைத்திலும் பேஸிக் மொழி பெரிதும் பயன்படுகிறது. இவ்வாறு பல இன நுண்ணியக்கிகளைக் கொண்ட மைந்த வெவ்வேறு கணிப்பொறிகளில் பயன்படும் பேஸிக் மொழி களின் ஆணைகள் பெரும்பாலும், ஒரே மாதிரியாக அமைந்துள்ள தால், பேஸிக் மொழியின் பயன்பாட்டுத் திறன் மேலும் அதிக மாகியுள்ளது. இம்மொழியை எளிதாகக் கற்றுக்கொண்டு பயன் படுத்தலாம். ஓர் அறிவியல் அறிஞரோ, பொறியியல் வல்லுநரோ தமது துறையில் சேகரிக்கக்கூடிய செய்திக்கூறுகளைப் பகுப்பாய்வு செய்வதற்கும் தமது பணித்திட்டங்களைக் [Project work] கண் காணிக்கவும் கணிப்பொறியைப் பயன்படுத்தும் பொழுது, பேஸிக் மொழி பெரிதும் கை கொடுத்து உதவுகிறது. கட்டிடக் கலைஞன் ஒருவன் [Architect], தனது கட்டிடக்கலைப் பணியைச் சூழ்ந்து ணரவும், இடம், பொருள், மின் சாதனங்கள் முதலியவற்றின் தேவையைக் கணக்கிட்டறியவும், கணிப்பொறியைப் பயன்படுத்த முனையும்பொழுது வழியமைப்புச் செய்வதற்குப் பேஸிக் மொழி, திறன் வாய்ந்த ஒரு கருவியாக அமைகிறது.

மேலும் வங்கி, ஆயுட் பாதுகாப்பு நிறுவனங்கள், தொழிற்கூடங்கள், ஆராய்ச்சிப் பணி மையங்கள் [Research Centers] முதலியவற்றின் பயன்பாட்டிற்கு உகந்த மொழியாக இது அமைந்துள்ளது. போர்ட்ரான், கோபால் ஆகிய மொழி களில் வழியமைப்புச் செய்வதைக் காட்டிலும் பேஸிக் மொழியில் எளிதாக வழியமைப்புச் செய்ய முடியும். சீர்மையும் முன்னேற்றமும் செய்யப்பட்ட பேஸிக் [Modified Basic] மொழி தற்காலத்தில் பெருமளவு பயன்படுத்தப்படுகிறது. மேற்கண்ட் பயன்பாடுகள் மட்டுமன்றி, வணிகத்துறைப் பயன்பாடுகளான

சம்பளப் பட்டியல் தயாரித்தல், பொருள் நிலையாளுகை, பொருள் உற்பத்தித் திட்டமிடல், நிதி நிலையாளுகை முதலான பணிகளில் எண்ணற்ற செய்திகளைத் தொகுத்து வகைப் படுத்தவும் விவர ஆய்வனை செய்யவும் நமது தேவைக்கேற்ப அறிக்கைகளைத் [Report] தயார் செய்யவும் ஏற்ற சிறந்த கணிப்பொறி மொழியாகப் பேஸிக் மொழி விளங்குகிறது.

பேஸிக் மொழி கணிப்பொறியில் எங்ஙனம் பயன்படுத்தப் படுகிறது என்பது படம் எண் 42-இல் விளக்கப்பட்டுள்ளது.



படம்—42 பேஸிக் மொழி கணிப்பொறியில் செயல்படும் விதம்

7.3. பேஸிக் மொழியின் தனிச் சிறப்பியல்புகள் (SPECIAL FEATURES OF BASIC LANGUAGE)

1. இம்மொழி கற்பதற்கு மிகவும் எளிது. இதில் வழி யமைப்புச் செய்தல் எளிதாகும். ஏனெனில், ஆங்கில மொழியைப் போன்று படித்தறியத் தக்க சொற்களைக் கொண்டு வழியமைப்பு ஆணைகள் அமைவதால், வழி யமைப்பைச் சோதித்தறிதல் மிகவும் எளிதாகிறது.

2. பேஸிக் மொழி, சிறிய மற்றும் மிகச் சிறிய கணிப்பொறிகளில் பயன்படுத்துவதற்கு மிகவும் ஏற்றது.

3. பேஸிக் மொழி,தொழில்,அறிவியல், ஆராய்ச்சி முதலிய துறைப் பணிகளுக்கு ஏற்ற மொழியாகும்.இம்மொழியில் இடுவரல், விடுவரல் ஆணைகளின் மூலம் கோவைகளை [Files] உருவாக்கவும் அவற்றைப் பயன்படுத்தவும் ஏற்ற வசதிகள் அமையப் பெற்றுள்ளன.இதன் மூலமாக தொழிற் கூடங்களில் சம்பளப் பட்டியல் தயாரித்தல், பொருள் நிலையாளுகை, நிதிநிலைக் கணக்கு முதலான பல பணிகளைச் செய்யத் தேவையான,பல வழியமைப்புகள் வடிவமைத்து உருவாக்கப்படுகின்றன.

4. பேஸிக் மொழியில் வழியமைப்புச் செய்வதன் மூலம் கோபால், போர்ட்ரான் ஆகிய மொழிகளில் வழியமைப்புச் செய்வதைவிட, பிழை உணர்தலும் [Diagnostic], பிழை உணர் திருத்தலும் [Debugging] மிகவும் எளிதாகிறது.

5. இம்மொழி கற்பதற்கு எளிதாகையால் பள்ளி, கல்லூரி மாணவர்களுக்குக் கணிப்பொறியில் பயிற்சியளிக்கப் படும் பொழுது இதைப் பயிற்று மொழியாகப் பயன் படுத்தலாம்.

6. பேஸிக் மொழியில் 'இன்டர்பிரிட்டர்' வசதியிருப்பதால் வழியமைப்பை உருவாக்கவும் உடனடியாகத் திருத்தி அதைச் செயற்படுத்தவும் மிகவும் எளிதாகிறது. வழியமைப்பை வடிவமைத்து உருவாக்கும் தொடக்க நிலையில் இன்டர்பிரிட்டரைப் பயன்படுத்துவதன் மூலம் குறிப்பிடத்தக்க அளவு வழியமைப்பாளர்களின் நேரத்தை மீதப்படுத்த இயலும். எனவே, பெரும்பணித் திட்டங்களுக்கான வழியமைப்புகளை வடிவமைக்கும் பொழுது தொடக்க நிலையில் பேஸிக்இன்டர்பிரிட்டரை பயன்படுத்தலாம். வேறு உயர்நிலை மொழிகளில் இவ் வசதியில்லாததால் வழியமைப்பைத் திருத்தவும் வடிவமைத்து உருவாக்கவும் தொடக்க நிலையில் மிகுந்த நேரம் தேவைப்படுகிறது.

7. மேற்கண்ட இன்டர்பிரிட்டர், கணிப்பொறியைப் பயன் படுத்துவோருக்குக் கணிப்பொறியுடன் நேரடித் தொடர்பு வசதியை அளிக்கிறது. இதன் மூலம் வழி யமைப்புச் செயற்படும்பொழுது இடைநிலை விடை களைச் சோதித்தறியவும் அதன் செயற்பாட்டின் ஒவ் வொரு நிலையையும் கண்காணிக்கவும் வாய்ப்பேற் படுகிறது.

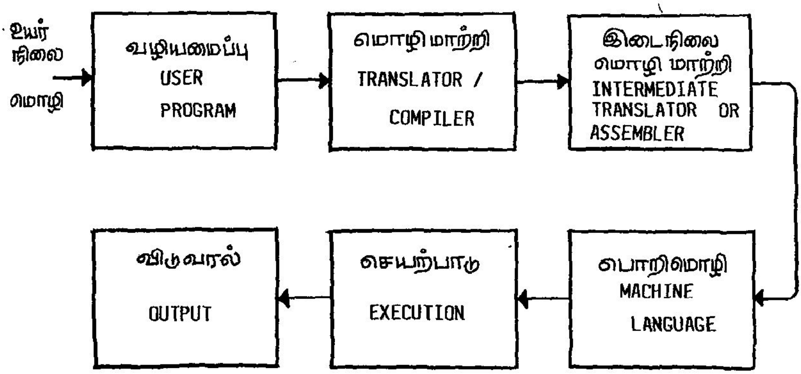
7.4. பேஸிக் மொழி மாற்றிகள் (BASIC LANGUAGE INTERPRETER COMPILER)

வழியமைப்பாளர்கள் செய்யும் வழியமைப்பு உயர் மொழி யான பேஸிக்கில் எழுதப்பட்டிருக்கும். இதைப் பொறி மொழி யாக மாற்றம் செய்து கொடுத்தால்தான், கணிப்பொறி அதை அறிந்துணர்ந்து நமது கணக்குக்கு விடை கொடுக்கமுடியும். உயர் மொழியாகிய பேஸிக்கில் உள்ள நமது கணக்கைப் பொறி மொழிக்கு மாற்றம் செய்வதற்கு மொழிமாற்றிகள் [Translators/ Compilers] தேவைப்படுகின்றன, மொழி மாற்றிகள் முதலில் உயர்மொழியிலுள்ள நமது வழியமைப்பை ஆய்வு செய்கின்றன. வழியமைப்பு இலக்கண முறைப்படி பிழையில்லாமல் இருப்பின், அது பொறிமொழியாக மாற்றம் செய்யப்படுகிறது. அவ்வாறு பிழை ஏதாயினும் இருக்குமாயின், உரியவகையில் பிழைச்செய்தி வெளிப்படுத்தப்படும். நாம் முன்பே குறிப்பிட்டபடி மொழி மாற்றி இரண்டுபடிகளில் இப்பணியைச் செய்கிறது.

1. உயர்நிலை மொழியிலிருந்து இடைநிலை மொழியாக மாற்றுதல்.
   [From Higher Level Language to Assembiy Language]

2. இடைநிலை மொழியிலிருந்து பொறிமொழிக்கு மாற்றம் செய்தல்.
   [From Assembly Language to Machine Language]

படம் எண் 43—இல் இம்முறை விளக்கப்பட்டுள்ளது.



படம்—43 மொழி மாற்றிகளின் செயற்பாட்டு முறை

பேஸிக் மொழி மாற்றிகளை இரண்டு வகைகளாகப் பிரிக்கலாம்.

1. பேஸிக் கம்பைலர் [Compiler]
2. பேஸிக் இன்டர்பிரிட்டர் [Basic Interpreter]

பேஸிக் கம்பைலருக்கும் இன்டர்பிரிட்டருக்கும் சில வேறு பாடுகள் உண்டு. பேஸிக் கம்பைலரில் வழியமைப்பைப் பயன் படுத்தும்பொழுது வழியமைப்பில் சில குறிப்பிட்ட ஆணைகளைப் பயன்படுத்தல் வேண்டும். குறிப்பாகக் கோவை ஆணைகள் [File Statements] இடுவரல், விடுவரல் ஆணைகள். இன்டர் பிரிட்டருக்கும் கம்பைலருக்கும்இடையே சிறிதுமாறுபட்டிருக்கும். கம்பைலரையோ இன்டர்பிரிட்டரையோ பயன்படுத்துவதில் சில நலன்களும் நலக்குறைகளும் உள்ளன.

பேஸிக் மொழி, மற்ற மொழிகளைக் காட்டிலும் கணிப் பொறியுடன் நேரடித் தொடர்பு வசதியில் செயல்பட மிகவும்

ஏற்றது. பேஸிக் இன்டர்பிரிட்டரில் வழியமைப்புச் செய்வதன் மூலம், விடையை உடனுக்குடன் சோதித்தறியலாம். வழியமைப்பில் பிழை ஏதேனும் இருக்குமாயின், பிழையைத் திருத்தித் திரும்பத் திரும்பச் செயலாக்கம் [Execution] செய்தல் மிகவும் எளிது. ஒரு வழியமைப்பைச் செயற்படுத்தும்பொழுது இடையில் நிறுத்தவோ, விடையின் தன்மையைச் சோதித்தறியவோ முடியும். அவ்வாறு செய்த பிறகு மீதியுள்ள வழியமைப்பைத் தொடர்ந்து செயற்படுத்துமாறு [Execution] கட்டளையிடலாம். மேலும், வழியமைப்பாளர் வழியமைப்புச் செய்யும் பொழுது தவறு திருத்திச் சோதித்தறிவதற்கு மிகவும் துணையாக இருப்பது இன்டர்பிரிட்டர் [Interpreter] ஆகும். வணிக மற்றும் அறிவியல் அல்லது ஆராய்ச்சித் துறைகளின் பயன்பாட்டிற்கென, பெரிய அளவிலான கணிம ஒருங்குகளை[Software system]வடிவமைத்து உருவாக்கும் பொழுது, தொடக்கத்தில் விரைவாகவும்தொடர்ந்து வேகமாகவும் சோதித்தறிய, கம்பைலரைவிட பேஸிக் இன்டர் பிரிட்டர் மிகவும் துணை நிற்கிறது ஆனால் இன்டர்பிரிட்டரைப் பயன்படுத்தி வழியமைப்பைச் செயற்படுத்தும் பொழுது கணிப் பொறியின் வேகம் குறைந்து விடுகிறது. மேலும் ஒவ்வொரு தடவையும் விடை தேவைப்படும்பொழுது, வழியமைப்பைச் செயல்படுத்த வேண்டிய அவசியம் ஏற்படுகிறது.

பேஸிக் கம்பைலரில் உள்ள வசதிகளைக கீழ்க்கண்டவாறு குறிப்பிடலாம்.

1. இதில் ஒரு முறை மொழிமாற்றம் [Compile] செய்து பொறி மொழியாக மாற்றிப் பதிவுசெய்து வைத்துக்கொண்டால் தேவைப்படும் பொழுது செய்திக்கூறுகளை இடுவரல் செய்து, விரைவில் விடையைப் பெறலாம். எனவே இன்டர் பிரிட்டரைக் காட்டிலும், இதன் செயற்பாட்டில் வேகம் அதிகமாகும். ஆனால் முதன் முதலில் கம்பைலர் மூலம் கணக்குக்குத் தீர்வு காணுதல் சற்றுக் கடினமான பணியாகும். இதில் வழியமைப்பு, செயலாக்க முறையில் இருக்கும்பொழுது, விடைகளைச் சோதித்தறிதலோ, பின்பு செயலாக்கத்தைத் தொடர்தலோ இயலாததாகும். இங்கு ஆராயப்படும் பேஸிக் மொழியில் கம்பைலர், இன்டர்பிரிட்டர் என்ற இருவகை மொழிமாற்றிகளும் உள்ளன. அவற்றில் பயன் படுத்தப்படும் வழியமைப்பு ஆணைகள், பின்வரும் பகுதிகளில் விவரிக்கப்பட்டுள்ளன.

இன்டர்பிரிட்டர் என்பது BBASIC என்ற பெயராலும், கம்பைலர் என்பது ABASIC என்ற பெயராலும் அழைக்கப்படு கின்றன.

ஓர் இயற்கை மொழியின் அமைப்பில் இலக்கணம், சொற் செறிவு முதலியவை எந்த அளவிற்கு மிகமிக அவசியமாக விளங்கு கின்றனவோ, அந்த அளவு கணிப்பொறி மொழிகளில் எழுதப் படும் ஒவ்வொரு வழியமைப்பு ஆணையும் அதன் இலக்கணமும் கணிப்பொறிமூலம் ஒருகணக்குக்குத்தீர்வுகாணமுற்படும்பொழுது அவசியமாகத் தோன்றுகின்றன. மாறிகள், மாறிலிகள் மற்றும் இயக்கிகள் இவைகளின் துணைகொண்டு, கணிப்பொறி வழி யமைப்பு ஆணைகள் உருவாக்கப்படுகின்றன. இவைகளைப் பற்றிக் கீழ்க்கண்டவாறு ஆராய்ந்தறியலாம்.

7.5. மாறிலிகள் (CONSTANTS)

பேஸிக் மொழியில் வழியமைப்புச் செய்யும் பொழுது, குறிப்பாகக் கணக்கியற் செயல்களைச் செய்யும்பொழுது, மாறிலிகள் [Constants] பயன்படுத்தப்படுகின்றன. எடுத்துக் காட்டாகக்கூறின், கணிதச் சமன்பாடுகளில் [Mathematical Equations] பயன்படும் π என்பது ஒரு மாறிலியாகும். வழி அமைப்பில் இதைப் பயன்படுத்தும்பொழுது இதன் மதிப்பு மாறுவதில்லை. மாறிலிகளை இரண்டு வகைகளாகப் பிரிக்கலாம்.

1. எண் மாறிலிகள் [Numeric Constants]
2. குறிக்கோவை மாறிலிகள் [String Constants]

1. எண் மாறிலிகள் (Numeric Constants)

எண் மாறிலிகளை முழு எண் [Integer], ஒற்றைத்துல்லிய எண் [Single Precision], இரட்டைத்துல்லிய எண் [Double Precision] எனப் பாகுபடுத்தலாம். முழு எண் மாறிலியின் மதிப்பு —32,768 முதல் +32,767 வரை ஆகும். ஒற்றைத் துல்லிய எண் மாறிலிகள், உண்மை எண்களாகும். அவை 7 இலக்கங்களையும் ஒரு தசமப் புள்ளியையும் [Decimal Point] பெற்று இருக்கும். அதனுடைய மதிப்பு 7 இலக்கங்களுக்குமேல் செல்லும்போது Exponential Form—இல் எண்கள் குறிக்கப்படு கின்றன. அதேபோன்று இரட்டைத்துல்லிய எண் மாறிலிகள்

8–இலிருந்து 16 இலக்கங்கள் வரையும் அதற்கு மேற்பட்ட எண்களைக் குறிக்க Exponential Form–உம் பயன்படுத்தப்படு கின்றன.

எ.கா :

முழு எண் = 5000

ஒற்றைத்துல்லிய எண் = 7641.46

Exponential Form = (7.32+E02)

இரட்டைத்துல்லிய எண் = 784269541

எடுத்துக்காட்டாக NUMBER என்பதை எண் மாறிலி என வைத்துக்கொள்வோம். அது முழு எண்ணாகவும்,ஒற்றை மற்றும் இரட்டைத்துல்லிய எண்ணாகவும் பேஸிக் மொழியில் குறிப்பிடப் படும் பொழுது கீழ்க்கண்டவாறு குறிப்பிடப்படும்.

முழு எண்	:	NUMBER %
ஒற்றைத்துல்லிய எண்	:	NUMBER !
இரட்டைத்துல்லிய எண்	:	NUMBER #

2. குறிக்கோவை மாறிலிகள் (String Constants)

குறிக்கோவை மாறிலி என்பது தொடர்ச்சியான குறிகள், இரண்டு மேற்கோள் குறிகளுக்கிடையில் இடப்பட்ட அமைப்பு ஆகும். ஒரு குறிக்கோவை மாறிலியில், 255 குறிகள் வரை இருக்கலாம். குறிகள், எண், எழுத்து அல்லது எண்ணெழுத்து [Alphanumeric] ஆகிய அமைப்பாக இருக்கலாம்.

எ.கா:

குறிக்கோவை மாறிலி (String Constant)

(i) "TAMIL UNIVERSITY"

(ii) "75123.457"

(iii) "AB763TAM"

7.6. மாறிகள் (*VARIABLES*)

பேஸிக் மொழி வழியமைப்புச் செயற்படும் பொழுது கணக்கின் தன்மைக்கேற்ப மதிப்பினைச் சுட்டும் பெயரே மாறியாகும்.

ஒரு கணிப்பொறியின் வழியமைப்பு ஆய்வு செய்யப்படும் பொழுது, மாறிலியின் மதிப்பு மாறுவதில்லை• அது நிலையாக உள்ளது. ஆனால், மாறிகள் வழியமைப்பின் ஒவ்வொரு கட்டத்திலும் கணக்கின் தன்மைக்கேற்ப, அதனுடைய மதிப்பை மாற்றிக்கொள்கின்றன. கணிப்பொறியின் நினைவகம் பல பதிவகங்களைக் கொண்டுள்ளது. ஒவ்வொரு பதிவகத்திற்கும் ஒரு குறிப்பிட்ட சுட்டு எண் [Location Number] கொடுக்கப்பட்டுள்ளது. இதற்கு நினைவக முகவரி [Memory Address] என்றுபெயர். எனவே வழியமைப்பில் உள்ள மாறிலிகள் மதிப்பு, குறிப்பிடப்பட்ட நினைவகப்பதிவகங்களில் பதிவுசெய்யப்படுகின்றன. எடுத்துக் காட்டாக, பேஸிக் மொழியில் மாறியை ஒரு வழியமைப்பு ஆணையில் பயன்படுத்தும்பொழுது LET ROSE = 100 இதில் ROSE என்ற பெயர் கொண்ட நினைவகப் பதிவகத்தில் [Memory Location] *100* என்ற எண் பதிவுசெய்யப்படுகிறது. வழியமைப்பு செயற்படும்பொழுது, கணக்கிடப்பட்ட மதிப்பு ஒரு மாறியில் பதிவுசெய்யப்படுவதற்கு முன்பு, அந்த மாறியின் மதிப்பு *0* எனக்கொள்ளப்படும்.

7.6.1. மாறிகளின் பெயர் (*VARIABLE NAME*)

மாறிகளுக்கு ஆங்கில எழுத்துக்கள் *26*—ஐயும், எண்கள் *'0'* — இலிருந்து *'9'* — வரை உள்ளவற்றையும் பயன்படுத்திப் பெயர் தரலாம். மேலும் மாறியின் தன்மையைக் குறிக்க ! # % முதலிய குறிகளையும் பயன்படுத்தலாம். மாறிகளின் ஆரம்பத்தில் எண்களைக் கொண்டு அதன் பெயரைத் துவங்குதல் தவறாகும். இங்கு நாம் ஆராயும் பேஸிக் மொழியில், ஒரு மாறியின் பெயர் எவ்வளவு குறிகளைக் கொண்டு வேண்டுமானாலும் அமையலாம். ஆனால் முதல் *40* குறிகளே அந்த மாறியைக் குறிப்பதாகக் கணிப்பொறி எடுத்துக்கொள்ளும். இந்த வசதி காரணமாக வழியமைப்பை நன்கு ஆவணப்படுத்த [Documentation] இயலும். சில பேஸிக்மொழிகளில் இரண்டு அல்லது மூன்று குறிகள் மட்டுமே மாறியின் பெயராகப் பயன்படுத்த வேண்டியிருக்கும்.

பொதுவாக மாறிகளைக் கீழ்க்கண்டவாறு இரண்டு பிரிவுகளாகப் பிரிக்கலாம்.

1. தனி மாறிகள் [Simple Variables]
2. நினைவக நிருணய மாறிகள் [Subscripted Variables]

7.6.2. தனி மாறிகள் (SIMPLE VARIABLES)

மேற்குறிப்பிட்டவாறு எண், எழுத்து மற்றும் எண்ணெழுத்துகளின் சேர்க்கைகளைக்கொண்டு, தனி மாறிகள் குறிப்பிடப்படுகின்றன. மாறிகளின் தன்மையைக் குறிப்பிட %, ! மற்றும் # ஆகிய சில தனிக்குறிகள் பயன்படுகின்றன. தனி மாறிகளை இரண்டு பிரிவுகளாகட் பிரிக்கலாம்.

1. எண் மாறிகள் [Numeric Variables]
2. குறிக்கோவை மாறிகள் [String Variables]

எண் மாறிகள் என்பன வழியமைப்பில் குறிப்பிடப்பட்டு உள்ள கணக்கின் தன்மைக்கேற்ப முழு எண், ஒற்றை அல்லது இரட்டைத்துல்லிய மதிப்பினைப் பெற்றுச் செயற்படும். எண் மாறிகள் எவ்வாறு குறிக்கப்படுகின்றன என்பதைக் கீழ்க்கண்ட எடுத்துக்காட்டுகள் மூலம் விளக்கலாம்.

எ. கா :

1. எண் மாறிகள்

```
(i) EMPNO%, OTM%, NODAY% (Integer variable)
(ii) VAL1, VAL2 ( To represent Real number)
(iii) A!, B! (To represent Single precision)
(iv) X#, Y# (To represent Double precision)
```

2. குறிக்கோவை மாறிகள்

மேற்குறிப்பிட்ட மாறியிலிருந்து இது மாறுபட்டுள்ளது. மாறியின் பெயரில் கடைசிக்குறி '' $ '' ஆக இருக்கும். எடுத்துக்காட்டாக TAM என்ற மாறி எண் மாறியாகும்; ஆனால் TAM$ என்ற பெயர் குறிக்கோவை மாறியைக் குறிப்பதாகும். பெயர்,

செய்தியுரைகளடங்கிய செய்திக்கூறுகளை வழியமைப்பு ஏற்றுச் செயற்படும்பொழுது, செய்திக்கூறுகளின் மதிப்பினைக் குறிப்பிட இந்த மாறிகள் பயன்படுகின்றன. எடுத்துக்காட்டாகக்கூறின், சம்பளப்பட்டியல் தயாரித்தலில் பணியாளர்களின் பெயர், முகவரி முதலிய விவரங்களும் பொருள் நிலையாளுகையில் பொருள் எண் [Stock Number] முதலான செய்திக்கூறுகளைக் குறிக்கவும் வழியமைப்பில் படித்தறிந்து விவர ஆய்வனை செய்யவும் குறிக்கோவை மாறிகள் பயன்படுகின்றன.

எ. கா :

```
A$="PROGRAMMING LANGUAGE"
N$, NAM$, TITLE$
```

7.6.3. நினைவக நிருணய மாறிகள் (SUBSCRIPTED VARIABLES)

கணிப்பொறியில் வழியமைப்பு செயற்படும்பொழுது, கணக்கிடப்பட்ட விடைகளையும் அல்லது விவர ஆய்வனைக்குத் தேவையான சில செய்திக்கூறுகளையும் தொடர்ச்சியான பட்டியலாக நினைவகத்தில் பதிவு செய்து வைத்திருக்கவேண்டிய அவசியம் ஏற்படும். கணக்கியற் செயற்பாடு நிகழும்பொழுது, இந்தப் பட்டியல்கள் பயன்படுத்தப்படுகின்றன. எனவே, இந்தத் தொடர்ச்சியான செய்திக்கூறுகள் நினைவகத்தில் குறிப்பிட ஒரு தனி மாறியின் பெயருடன், ஒவ்வொரு செய்திக்கூறுக்கேற்ற சுட்டு எண் அல்லது Subscript Numberஐயும் சேர்த்து நினைவக நிருணய மாறிகள் உருவாக்கப்படுகின்றன. Subscript Number முழு எண்ணாக இருத்தல் வேண்டும்.

எ. கா: AB(I, J), NR(M), TAK(K, L) மேற்கண்ட நினைவக நிருணய மாறிகளில் I, J, M, K, L என்பன முழு எண்களாகும்.

7.7. குறிகள் (CHARACTERS)

ஆங்கில எழுத்துகள் 26-உம் மற்றும் எண்கள் '0' இல் இருந்து '9' —வரையும் தனிக்குறிகளான " ! # % (? , *) : ; யாவும் குறிகள் என அழைக்கப்படுகின்றன. இவற்றின் துணை கொண்டு மாறிகளும் மாறிலிகளும் உருவாக்கப்பட்டு, பேஸிக் வழியமைப்பில் பயன்படுத்தப்படுகின்றன.

7.8. இயக்கிகள் (OPERATORS)

ஒரு வழியமைப்பில் பயன்படும் சமன்பாடோ, வழியமைப்பு ஆணையோ, எண் மாறிலி அல்லது மாறிகளைக் கொண்டோ, மாறி, மாறிலி, இயக்கி இவைகளைக் கொண்டோ அமையலாம். இயக்கிகள் இயமிகளின்மீது கணித அல்லது முறையமைச் செயற்பாட்டைச் செய்கின்றன.

இயக்கிகளைக் கணக்கியல் இயக்கிகள் [Arithmetic Operators], முறையமை இயக்கிகள் [Logical or Boolean Operators], தொடர்பமை இயக்கிகள் [Relational Operators], கூறியக்கிகள் [Functional Operators] என நான்கு பிரிவுகளாகப் பிரிக்கலாம்.

7.9. கணக்கியல் இயக்கிகள் (ARITHMETIC OPERATORS)

இயக்கிகள் என்பன ஒரு குறிப்பிட்ட செயலைச் செய்யும் பொழுது அதற்குக் காரணிகளாக அமைவனவாகும். ஓர் எண்ணையும் மற்றொரு எண்ணையும் கூட்டும்பொழுது அதில் உள்ள கூட்டல் [+]குறி, கணக்கியல் இயக்கியாகக் கருதப்படும். கணக்கியல் இயக்கிகளுக்கு எடுத்துக்காட்டுகள் கீழே கொடுக்கப் பட்டுள்ளன.

+	:	கூட்டல்	A + B
—	:	கழித்தல்	A — B
×	:	பெருக்கல்	A × B
/	:	வகுத்தல்	A / B

எண்ணடுக்கு A ↑ B [A-இன் அடுக்கு B].

7.10. முறையமை இயக்கிகள் (LOGICAL & BOOLEAN OPERATORS)

இரண்டு இயமிகள் அல்லது எண்களுக்கிடையே ஒரு குறிப்பிட்ட முறையமை செயலைச் செய்ய, முறையமை இயக்கிகள் பயன்படுகின்றன. இயமிகளுக்கிடையே உள்ள எல்லாத் தொடர்புகளையும் கொண்டு முறையமை இயக்கிகள் செயல்படுகின்றன. ஒரு கணக்குத் தொடரில் [Expression] கணக்கியல் மற்றும் தொடர்பமை இயக்கிகள் செயல்பட்ட பின்பே, முறையமை இயக்கி செயல்படுகிறது. இந்த இயமி செயல்படும்பொழுது 'உண்டு' [அ] 'இல்லை' என்ற இருநிலைகளில், கணக்கின் தன்மைக்கேற்ப ஏதாவது ஒருநிலை உண்மையாக இருக்கும். முறையமை இயக்கிகளுக்கு எடுத்துக்காட்டுகள் கீழே கொடுக்கப்பட்டுள்ளன.

NOT	:	NOT M
AND	:	M AND N
OR	:	A OR B
XOR	:	M XOR N
IMP	:	M IMP N
EQV	:	M EQV N

எ.கா: `10 IF AB>723 AND NUM<XYZ THEN 50`

7.11. தொடர்பமை இயக்கிகள் (RELATIONAL OPERATORS)

தொடர்பமை இயக்கிகள் இரண்டு எண்கள் அல்லது இரண்டு மாறிகளுக்கிடையேயுள்ள தொடர்பை ஒப்பிட்டு, பேஸிக் மொழியில் புலப்படுத்துவதற்குப் பயன்படுகின்றன. தொடர்பமை இயக்கிகள் செயற்பட்டபின் கிடைக்கும் விடையான 'உண்மை' [அ] 'பொய்' என்ற மதிப்பைப் பொறுத்து மற்ற செயல் நிகழும்.

தொடர்பமை இயக்கிகள்

=	சமமான	X = Y
<>	சமமற்ற	X<>B

<	குறைவான	X < Y
>	மிகையான	X > D
<=	குறைவான அல்லது சமமான	X<=D
>=	மிகையான அல்லது சமமான	X>=D

எ.கா: 100 IF COUNT<TEM THEN 500

7.12. கூறியக்கிகள் (FUNCTIONAL OPERATORS)

வழியமைப்பு ஆணையில் கூறுகள் பயன்படுத்தப்படுகின்றன. கூறுகள் சில வரையறுக்கப்பட்ட பணியினைச் செய்ய உருவாக்கப் பட்ட சிறிய வழியமைப்புகளாகும். இவை கணிப்பொறியின் கணிமஒருங்கில் அமைந்துள்ளதால் கூறியக்கிகளை வழியமைப்பில் பயன்படுத்துவதன் மூலம், குறிப்பிட்ட செயற்பாடு நிகழ ஏது வாகிறது. எடுத்துக்காட்டாக SQR, SIN, COS, TAN முதலிய கூறியக்கிகளைக் கூறலாம்.

7.13. சூத்திரங்களும் சமன்பாடுகளும் (FORMULA & EQUATIONS)

கணிதத்தில் பலதரப்பட்ட கணக்கியற் செயற்பாடுகள் செய்ய வேண்டியுள்ளன. எடுத்துக்காட்டாகக் கூறின், ஒரு வட்டத்தின் பரப்பளவு, சுற்றளவு முதலியவற்றைக் கண்டறிதல், சதுரம், செவ்வகம் ஆகியவற்றின் பரப்பளவைக் கண்டறிதல், உருண்டை வடிவப்பொருள் அல்லது கனபரிமாணமுள்ளபொருள் [Cubical] ஆகியவற்றின் கொள்ளளவைக் [Volume] கண்டறிதல் முதலியன ஒரு குறிப்பிட்ட சூத்திரத்தைப் பயன்படுத்தியே செய்யப்படுகின்றன. சூத்திரங்கள் என்பன முறையாக வழிப் படுத்தப்பட்ட நிலையான ஒரு கொள்கையாகும்.

மாறி, மாறிலி மற்றும் கணக்கியல் இயக்கிகள் [Arithmetic Operator] இவற்றைக் கொண்டு சமன்பாடு அமைக்கப்படுகிறது. ஓர் எண்ணிலிருந்து மற்றோர் எண்ணைக் கழிக்கவோ, கூட்டவோ, இரண்டு எண்களைப் பெருக்கவோ, வகுக்கவோ செய்யும்பொழுது −, +, *, / ஆகிய இயக்கிகள் பயன்படுகின்றன. வழியமைப்பில் சமன்பாடுகள் பயன்படுத்தப்படும் பொழுது, சில விதிமுறை களின்படி கணிப்பொறி அவற்றைக் கணக்கிடுகிறது. அவ்வாறு

கணிக்கும் பொழுது கணிப்புப் பணியானது கீழ்க்கண்டவாறு வரிசை முறையாக நடைபெறுகிறது.

1. சூத்திரத்தின் வலமிருந்து இடமாகக் கணிப்பொறி ஆய்வு செய்யும்பொழுது, பிறைக்குறிக்குள் உள்ள பகுதியை முதலில் கணிக்கிறது.

2. ஒரு சூத்திரத்தில் வலமிருந்து இடமாக வரும்பொழுது பிறைக்குறிக்குள் உள்ள செயல் முடிந்த பின் முறைப்படி அதில் காணப்படும் Exponentiation Operation செய்யப்படுகிறது.

3. மேற்குறிப்பிடப்பட்ட சூத்திரத்தைத் திரும்பவும் ஆய்வு செய்து முறைப்படி முதலில் பெருக்கலையும், பின்னர் வகுத்தலையும் செய்கிறது.

4. மேலே கண்டவாறு Exponentiation Operation, பெருக்கல், வகுத்தல் செயல்களைச் செய்து முடித்தபின் கணிப்பொறி திரும்பவும் அதே சூத்திரத்தை வலமிருந்து இடமாக ஆய்வு செய்து முறையே கூட்டல், கழித்தல் பணிகளைச் செய்கிறது.

இந்த விதிகளை விளக்கி ஒரு சூத்திரத்தைக் கணிப்பொறி எவ்வாறு கணிக்கிறது என்பதைக் கீழ்க்கண்டவாறு விளக்கலாம்.

எ.கா : (15+4*3^2/3-6)

15 + 4 × 3 ^ 2/3-6 ஐக் கணிப்பொறியில் பயன்படுத்தும் சூத்திரமாக எடுத்துக்கொள்வோம்.

கணிப்பொறி சூத்திரத்தைக் கணிக்கும் முறை

சுற்று	சூத்திரம்	செயல்	கணிக்கப்பட்ட விடை
1.	15+4*3^2/3-6	Exponentiation Operation	15+4*9/3-6
2.	15+4*9/3-6	பெருக்கல்	15+36/3-6
3.	15+36/3-6	வகுத்தல்	15+12-6
4.	15+12-6	கூட்டல்	27-6
5.	27-6	கழித்தல்	21

விடை = 21

சூத்திரங்களை வழியமைப்பில் பயன்படுத்தும்பொழுது, மேற்கண்டவாறு கணிக்கப்பட்டு விடை பெறப்படுகிறது.

சூத்திரங்கள் :

எ.கா : 1. வட்டத்தின் பரப்பளவைக் காண $= \pi r^2$
r = ஆரம் [Radius]

2. கனத்தின் கொள்ளளவைக் கண்டறிய [Volume]
= நீளம் × அகலம் × உயரம்
$a \times a \times a = a^3$

பல அறிவியற் கணக்குகளில் [Scientific Problems] சூத்திரங் களும் மாறிகளும் மாறிலிகளும் பயன்படுத்தப்படுகின்றன. மாறி, மாறிலிகளைப் பயன்படுத்தி, ஒரு கணக்குக்குத் தீர்வு காண்பதற்குச் சமன்பாடுகள் அமைக்கப்படுகின்றன. சமன்பாடு களைக்கொண்டு குறிப்பிட்ட முறையில் கணக்குக்குத் தீர்வுகண்டு விடை பெறலாம். எடுத்துக்காட்டாகக் கூறின், ஒரு Quadratic Equation இன் தீர்வைக் கண்டுபிடிப்பதற்குப் பயன்படும் ஒரு சமன்பாடு கொடுக்கப்பட்டுள்ளது. $ax^2 + bx + c = 0$ என்ற சமன்பாட்டில் a, b, c என்பன மாறிலிகள், x என்பது மாறியாகும். ஏனெனில் இந்தச் சமன்பாட்டின் மதிப்பு x இன் மதிப்பைப் பொறுத்து மாறுகிறது. இந்தச் சமன்பாட்டில் மாறி, மாறிலி, மற்றும் இயக்கிகள் அமைந்துள்ளன. இதை ஒரு பொதுச் சமன் பாடாகக் கருதின் மாறி, மற்றும் மாறிலிகளின் மதிப்பை நமது கணக்குக்கேற்ப மாற்றியமைத்துத் தீர்வு காணலாம்.

7.14. நினைவக வரிசை (*MEMORY ARRAYS*)

சில குறிப்பிட்ட செயல்களைக் கணிப்பொறி மூலம் திரும்பத் திரும்பச் செய்வதற்குச் செய்வளையைப் [Loops] பயன்படுத்து கிறோம். இம்முறையில் ஒவ்வொரு செய்வளைச் சுற்றுக்கும் புதிய மாறிகளைப் பயன்படுத்த வேண்டியுள்ளது. இதற்குப் பதிலாக, நினைவக நிருணய [Subscripted Variables] மாறியைப் பயன்படுத்தினால், செய்வளைக்குத் தகுந்தாற்போன்று மாறி களை மாற்ற முடியும். நடைமுறையில் நினைவக நிருணய மாறிகளை [Subscripted Variables] A_1 என்று எழுதுகிறோம். ஆனால் வழியமைப்பில் A (*1*) என்றுதான் குறிப்பிட வேண்டும்.

நினைவக நிருணயமாறிகளில் ஒற்றை நினைவக நிருணயமாறி [Single Subscripted Variable] அல்லது இரட்டை நினைவக நிருணயமாறி [Double Subscripted Variable] என்ற இரண்டு பிரிவுகள் உள்ளன. முன்பு குறிப்பிடப்பட்டது போல் ஒற்றை அல்லது இரட்டை நினைவக நிருணயமாறிகள் [Subscripted Variables] ஒரு குறிப்பிட்ட நினைவகப் பதிவறையின் பெயராகும். இதைக் கீழ்க்கண்ட ஓர் எடுத்துக்காட்டு மூலம் விளக்கலாம்.

ஒரு நகரத்தில் பல வீடுகளும் விடுதிகளும் உணவகங்களும் இருப்பதாக வைத்துக்கொள்வோம். ஒரு வீட்டில் குடியிருக்கும் ஒருவருக்கு ஒரு கடிதத்தைக் கொடுக்க வேண்டுமெனில், அந்த வீட்டின் பெயரைக் குறிப்பிட்டால் போதுமானதாகும். இது தனி மாறிகளுக்கு எடுத்துக்காட்டாகும்.

எ.கா : A1, B1, TON

அதே சமயம், விடுதியில் தங்கியுள்ள ஒருவருக்கு அந்தக் கடிதத்தைக் கொடுக்க வேண்டுமெனில்,விடுதியின் பெயர், அறை எண் இவற்றைக் அக்கடிதத் தலைப்பில் குறித்தல் அவசியமாகும். இது ஒற்றை நினைவக நிருணய மாறிக்கு [Single Subscripted Variable] எடுத்துக்காட்டாகும்.

எ.கா : TAMILAKAM(6)

மேலே குறிப்பிடப்பட்டுள்ள மாறியானது TAMILAKAM என்ற விடுதியிலுள்ள அறை எண் 6-ஐக் குறிப்பதாகும். அதேசமயம் TAMILAKAM என்ற ஓட்டலின் இரண்டாம் தளத்தில் உள்ள 25வது அறையைக்குறிப்பிடவேண்டுமெனில் TAMILAKAM (25,2) என்று எழுதவேண்டும். ஒவ்வொரு நினைவக நிருணய மாறியின் பெயரும் ஒரு நினைவகப் பதிவறைத் தொகுப்பைக் குறிப்பதாகும். இது வரிசை[Array or Matrix] என்று குறிப்பிடப்படும். Subscript என்று கூறப்படும் இந்த எண் +ve முழு எண்ணாக இருத்தல் வேண்டும். வழியமைப்பை உருவாக்கும் பொழுது வழியமைப்பு ஆணை மூலமாக நினைவக ஒதுக்கீடு [Memory Allocation] செய்தல் வேண்டும். இதற்குப் பயன்படும் வழியமைப்பு ஆணை DIMENSION என அழைக்கப்படும். அவ்வாறு நினைவக ஒதுக்கீடு செய்வதற்கேற்ற ஆணை வழியமைப்பில் இல்லாவிடில் கணிப்பொறி '0' இலிருந்து பத்து வரை உள்ள Subscript-க்கு மட்டும் நினைவக ஒதுக்கீடு செய்துகொண்டு

மீதியுள்ள பகுதியை விட்டுவிடும். பத்துக்கு மேல் உள்ள Subscript-ஐ வழியமைப்பில் வேறு எங்காவது பயன்படுத்தினால் பிழைச்செய்தி அறிவிக்கப்படும்.

எ. கா : DIM DON(25), ROW(20,30)

மேற்கண்ட எடுத்துக்காட்டில் 'DON' என்ற பெயரிலுள்ள நினைவக வரிசையில் 25 மாறிகளுக்கேற்ற நினைவக இட ஒதுக்கீடும், ROW என்ற பெயரில் 600 நினைவகப் பதிவறைகளும் இட ஒதுக்கீடு செய்யப்படுகின்றன.

7.15. பேஸிக் மொழிக் கட்டளைகள் (BASIC LANGUAGE COMMANDS)

பேஸிக்கில் உள்ள கட்டளைகளுக்கும், வழிமுறை ஆணைகளுக்கும் சில வேறுபாடுகள் உள்ளன. பேஸிக்கில் உள்ள சில கட்டளைகளைக் கொண்டு வழியமைப்பை வரிசையாக வெளிப்படுத்துவதற்கும் [List], வழியமைப்பை நினைவகத்திற்குக் கொணர்வதற்கும், பின் பதிவு செய்வதற்கும் [Loading & Saving], கோவைகளை [Files], உருவாக்குவதற்கும் [Creation] அல்லது துணை நினைவகத்திலிருந்து கோவைகளை நீக்குவதற்கும் [Erase] முடிகிறது. பேஸிக்கில் உள்ள கட்டளைகளைக் கீழ்க்கண்டவாறு விவரிக்கலாம்.

1. AUTO–இது வழியமைப்பை இடுவரல் செய்யும்பொழுது ஒவ்வொரு வரி முடிவுற்ற பின்பும் சட்டத்திருப்பை [CR] அழுத்தினால் தானாகவே வரி எண்களை நியமிக்க உதவுகிறது.

கட்டளைப்படிவம்

```
AUTO [ Line number ], [ Increment ]
```

அதாவது வழியமைப்பு ஆணையின் தொடக்க எண் Line Number ஆகவும், அதற்கடுத்த வரி எண் Line Number உடன் Increment-ஐக் கூட்டியதாகவும் நியமிக்கப்படும். எனவே ஒரு வரி எண்ணுக்கும் அதற்கடுத்த வரி எண்ணுக்கும் இடையேயுள்ள வேறுபாடு, Increment-க்குச் சமமாக இருக்கும்.

AUTO-இக் கட்டளை 10, 20, 30 ...... என்று ஒரு வரி எண்ணுக்கும் அதற்கடுத்த வரி எண்ணுக்கும் 10-ஐ வேறுபாடாகக் கொண்டு வரி எண்களை உருவாக்குகிறது.

எ.கா : `AUTO 400,20`

இது முதல் வரி எண்ணை நானூறாகவும், அதைத் தொடர்ந்த வழியமைப்பின் வரி எண்களை இருபது கூடுதலாகவும் அமைக்கும். அதாவது 400, 420, 440....

2. **CLEAR:** இக்கட்டளை மாறிகளையும், மாறிலிகளையும் ஆரம்ப நிலைப்படுத்துவதற்குப் பயன்படுகிறது. மேற்கண்ட செயலன்றி நினைவக அளவையும் [Memory Capacity], நினைவக அடுக்கிடத்தையும் [Stack] நிருணயிக்கவும் இந்தக் கட்டளை உதவுகிறது.

படிவம்:

```
CLEAR [ expression1 ],[ expression2 ]
```

மேலே குறிப்பிட்ட படிவத்தில் 'expression1' இன் மதிப்பு உயர்நினைவக [Higher Memory] அளவையும், 'expression2' இன் மதிப்பு நினைவக அடுக்கிடத்தையும் நிருணயிக்க உதவுகிறது.

எ.கா : `CLEAR 3000,256`

மாறிகள் ஆரம்ப நிலைப் படுத்தப்படுவதுடன் உயர் நினைவக [High Memory] அளவை 3000 ஆகவும், அடுக்கிடத்தை [Stack Space] 256 பைட்ஸ் ஆகவும் தீர்மானிக்கிறது.

3. **CONT:** வழியமைப்பு செயற்பாட்டு நிலையில் [Execution Mode] இருக்கும்பொழுது AC அல்லது STOP என்ற கட்டளைகளைக் கண்டவுடன் செயற்பாடு நின்றுவிடும். திரும்பவும் செயற்பாட்டைத் தொடரச் செய்வதற்கு "CONT" என்ற கட்டளை பயன்படுகிறது.

இந்தக் கட்டளை STOP என்ற ஆணையுடன் பயன்படுத்தப்படும் பொழுது, வழியமைப்பைப் சோதித்துத் தவறுகள் திருத்தப் பெரிதும் உதவுகிறது. STOP என்ற ஆணையைப் பயன்படுத்தும்பொழுது, வழியமைப்புச் செயற்பாட்டைக் குறிப்பிட்ட வழியமைப்பு வரியில் நிறுத்த இயலுகிறது. எனவே இடைநிலை விடையைச் சோதித்தறியவும் மாற்றவும் முடியும். பின்னர் வழியமைப்புச் செயற்பாட்டைத் தொடரும் பொருட்டுக் CONT கட்டளை பயன்படுத்தப்படுகிறது. வழியமைப்பில் செயற்

பாட்டுப் பிழை [Execution Error] நிகழ்ந்த பின்னர் தொடர் வதற்கும் CONT கட்டளை பயன்படுகிறது.

குறிப்பு :- வழியமைப்பைத் திருத்தியமைத்தபின் CONT கட்டளையைப் பயன்படுத்துதல் இயலாது.

4. DELETE [Begin Line]-[End Line] இந்தக் கட்டளை மூலம் வழியமைப்பில் தேவையற்ற சில வழியமைப்பு வரிகளை நீக்கி விடலாம்.

எ.கா : (i) DEL 1000

வரி எண் 1000 அகற்றப்படுகிறது.

(ii) DEL 1000-1500

இது வரி எண் 1000லிருந்து 1500 வரையுள்ள வழியமைப்பு வரிகளை அகற்றுகிறது (இரண்டு எண்களும் உட்பட).

(iii) DEL -700

இந்தக் கட்டளையைப் பயன்படுவதன் மூலம் வழியமைப்பு வரிகளைத்தொடக்கத்திலிருந்து 700 வரை(700 உட்பட)அழித்து விட முடியும்.

(iv) DEL 700-

இந்தக் கட்டளை, வழியமைப்பு வரி எண் 700இல் தொடங்கி கடைசிவரை உள்ள வழியமைப்பு வரிகளை நீக்கி விடுகிறது.

5. EDIT: இக் கட்டளை மூலம் வழியமைப்பு வரியில் சில தவறுகளைத் திருத்துவதற்கு இயலும்.

படிவம் [Format] : EDIT <Line number>

EDIT கட்டளையைப் பயன்படுத்துவதன் வாயிலாக வழியமைப்பு வரியை மீண்டும் தட்டச்சு செய்யாமலேயே தவறுகளைத் திருத்த இயலும். Edit சூழ்நிலைக்குச் சென்றவுடன் நாம் அறிவித்த வழியமைப்பு வரியில் குறிகாட்டி [Cursor] நிற்கிறது. பின்னர் Edit சூழ்நிலையில் உள்ள பிற துணைக் கட்டளைகளைக் கொண்டு வழியமைப்பு வரியை மாற்றியமைக்கலாம். Edit

கட்டளையிலுள்ள துணைக் கட்டளைகள் மூலம் குறிகாட்டியை நகர்த்தவும், புதிதாக வழியமைப்பு வரியில் குறிகளைப் புகுத்தவும் அல்லது நீக்கவும் அல்லது குறிப்பிட்ட குறிக்கோவையைக் கண்டறிந்து அதற்குப் பதிலாக வேறு ஒரு குறிக்கோவையாக மாற்றவும் இயலும்.

எ.கா: EDIT 500

வரி எண் 500ஐத் திருத்துவதற்காகக் காட்சித் திரையில் வழியமைப்புவரி வெளிப்படுத்தப்படுகிறது.

6. FILES (File Name)

இக்கட்டளைத் துணை நினைவகத்திலுள்ள [Auxiliary Storage] கோவைகளின் பட்டியலைத் [Directory] தேவையான முறையில் காட்சித் திரையில் வெளிப்படுத்தப் [Display] பயன் படுகிறது.

எ. கா: FILES

துணை நினைவகத்திலுள்ள எல்லாக் கோவைகளின் பெயர்களும் வெளிப்படுத்தப்படுகின்றன.

எ. கா: (i) FILES "*.BAS"

இந்தக் கட்டளை மூலம் துணை நினைவகத்திலுள்ள பேஸிக் வழியமைப்புக் கோவைகளின் பெயர்ப் பட்டியல் வெளிப் படுத்தப்படுகிறது.

(ii) FILES "*.DAT"

செய்திக்கூறுக் [Data Files] கோவைகளின் பெயர்ப் பட்டியல்கள் மட்டும் வெளிப்படுத்தப்படுகின்றன.

7. LIST (Line No.1) or LIST (Line No.1)-(Line No.2)

நினைவகத்திலுள்ள வழியமைப்பைக் காட்சித் திரையில் வெளிப்படுத்த இக்கட்டளை உதவுகிறது.

நினைவகத்திலுள்ள வழியமைப்பு வரிகளை வெளிப் படுத்துவதில் நான்கு விதமாக மேற்கண்ட கட்டளையைப் பயன் படுத்தலாம். இதைக் கீழ்க்கண்ட எடுத்துக்காட்டுகள் மூலம் விளக்கலாம்.

எ. கா :(i) LIST

நினைவகத்திலுள்ள முழு வழியமைப்பு வரிகளையும் வெளிப்படுத்துகிறது.

(ii) LIST 200

வரி எண் 200-ஐ மட்டும் வெளிப்படுத்துகிறது.

(iii) LIST 300–500

வரி எண் 300-லிருந்து 500 வரை உள்ள வழியமைப்பை வெளிப்படுத்துகிறது.

(iv) LIST –600

இந்தக் கட்டளை மூலம் வழியமைப்பின் தொடக்கத்திலிருந்து வழியமைப்பு வரி எண் 600 வரை (வரி எண் 600 உட்பட) காட்சித் திரையில் வெளிப்படுத்தப்படுகின்றன.

(v) LIST 600–

வழியமைப்பு வரி எண் 600-இல் தொடங்கி வழியமைப்பின் இறுதிவரை உள்ள வரிகள் வெளிப்படுத்தப்படுகின்றன.

8. LLIST: இக்கட்டளை மேற்குறிப்பிடப்பட்ட செயல்களைப் புரிகிறது. ஆனால் காட்சித் திரைக்குப் பதிலாக அச்சுப் பொறியில் [Printer] வழியமைப்பு வரிகள் வெளிப்படுத்தப் படுகின்றன.

9. LOAD : துணை நினைவகத்திலிருந்து முதன்மை நினைவகத்திற்கு, வழியமைப்பைக் கொண்டுவர இக்கட்டளை உதவுகிறது. இந்தச் செயல் நடைபெறும் பொழுது துணை நினைவகத்திலுள்ள கோவைகள் யாவும் பாதுகாக்கப்பட்டு, நினைவகத்திலுள்ள மாறிகள் அனைத்தும் ஆரம்பநிலைப்படுத்தப் படுகின்றன.

எ.கா: LOAD "PAYROLL.BAS", R

துணை நினைவகமாகிய Disc-இலிருந்து PAYROLL.BAS என்ற வழியமைப்பை முதன்மை நினைவகத்திற்குக் கொண்டு வந்து பின்னர் செயற்படுத்துகிறது.

10. MERGE [File Name] : இக்கட்டளையைப் பயன் படுத்தி நினைவகத்திலுள்ள ஒரு வழியமைப்புடன் துணை நினை வகத்திலுள்ள மற்றொரு வழியமைப்பைக் கொணர்ந்து ஒன்று சேர்த்தல் [Merge] இயலும்.

எ.கா: MERGE "SECOND.BAS"

ஆணையைப் பயன்படுத்தும் முன்பு மேற்குறிப்பிடப்பட்ட SECOND·BAS என்ற கோவை துணை நினைவகத்தில் ASCII முறையில் பதிவு செய்யப்பட்டிருத்தல் அவசியமாகும்.

குறிப்பு: நினைவகத்திலுள்ள வழியமைப்பின் சில வரி எண்களும் துணை நினைவகத்திலுள்ள வழி யமைப்பின் சிலவரி எண்களும் ஒன்றாக இருக்கு மாயின், குறிப்பிட்ட வரி எண்கள் (நினைவகத் திலுள்ள) அழிக்கப்பெற்று, பதிலாகத் துணை நினைவக வழியமைப்பு வரிகள் எழுதப்பெறும்.

11. NAME [Old File] AS [New File]

துணை நினைவகத்தில் அமைந்துள்ள வழியமைப்பு அல்லது செய்திக்கூறுக் கோவையின் [Data File] பெயரை மாற்றம் செய்வதற்கு இந்தக் கட்டளை பயன்படுகிறது.

மேற்குறிப்பிட்ட படிவத்திலுள்ள Old File என்பது துணை நினைவகத்தில் இருத்தலும், New File என்ற பெயரில் துணை நினைவகத்தில் வேறு கோவைகள் இல்லாதிருத்தலும், இக் கட்டளை செயற்படத் தேவையான நிபந்தனைகளாகும்.

எ. கா: NAME "ACCOUNT.DAT" AS "PAYROLL.DAT"

அதாவது 'ACCOUNT·DAT' என்ற செய்திக்கூறுக் கோவையின் பெயர் 'PAYROLL·DAT' என்ற பெயர் மாற்றம் செய்யப்படுகிறது.

ABASIC கம்பைலரில் மேற்குறிப்பிடப்பட்ட NAME (Old File) AS (New File) என்று கட்டளை செய்யும் பணியைக் கீழ்க்கண்டவாறு குறிப்பிடலாம்.

RENAME [ABC. DAT, XYZ. DAT] : XYZ·DAT என்ற கோவையின் பெயர் ABC. DAT என்று பெயர் மாற்றம் செய்யப்படுகிறது.

12. NEW:இது நினைவகத்திலுள்ள வழியமைப்பை முற்றிலும் அழித்துவிடப் பயன்படுகிறது. இது நினைவகத்தை ஆரம்ப நிலைப்படுத்தவும் புதிய வழியமைப்பைப் பதிவு செய்யவும் உதவுகிறது.

13. NULL: ஒவ்வொரு வரி எண் முடிவிலும் இரண்டு வெற்றுக் குறிகளை [Null Character] உருவாக்க இந்தக் கட்டளை உதவுகிறது.

எ. கா:

```
NULL 3
10 A=B-3
20 PRINT A,B
```

இங்கு ஒவ்வொரு வரியின் முடிவிலும் மூன்று வெற்றுக் குறிகள் வெளிப்படுத்தப்படுகின்றன.

14. RENUM [New Number], [Old Number], [Line Increment] இதன் மூலம் வழியமைப்பின் வரி எண்களை வெவ்வேறு எண்களாக மாற்றி அமைக்கலாம்.

மேற்கண்ட படிவத்திலுள்ள New Number என்பது வரிசை எண்களை மாற்றியமைக்கும் பொழுது முதல் எண்ணாகவும், அவ்வாறு எண் குறிப்பிடப்படாவிட்டால் முதல் எண் 10 ஆகவும் அமைகிறது. Old Number வரி எண்களை மாற்றப்பட வேண்டிய வழியமைப்பின் தொடக்க வரி எண்ணாகும். இந்த எண்ணிலிருந்து தொடங்கி முதல் எண் New Number ஆகவும் பின் ஒவ்வொரு வரி எண்ணும் அதனுடன் Increment கூடுதலாகவும் அமைகிறது. RENUM கட்டளையைச் செயற்படுத்தும் பொழுது வழியமைப்பிலுள்ள GOTO, GOSUB, ON GOTO, ON GOSUB முதலியவற்றின் வரி எண்களும் தானாகவே மாற்றியமைக்கப்படுகின்றன.

படிவம்

```
RENUM <New number>, <Old number>, <Increment>
```

எ.கா:(i) RENUM

வழியமைப்பு முழுவதையும் வரிசை எண் 10-இலிருந்து தொடங்கி ஒவ்வொரு வரிக்கும் 10 எண்கள் இடைவெளி இருக்குமாறு வரி எண்களை உருவாக்க இக்கட்டளை உதவுகிறது.

(ii) RENUM 500, 20

ஆரம்ப எண் 500-ஆகவும் அதிலிருந்து ஒவ்வொரு வரிக்கும் 20 எண்கள் இடைவெளி இருக்குமாறும் வழியமைப்பு வரி எண்கள் அமைக்கப்படுகின்றன.

15. RUN : இது நினைவகத்திலிருக்கும் வழியமைப்பைச் செயற்படுத்துவதற்குப் பயன்படுகிறது.

படிவம் : RUN <File Name>

மேற்கண்ட கட்டளை, துணை நினைவகத்திலிருந்து குறிப்பிட்ட வழியமைப்புக் கோவையை நினைவகத்திற்குக் கொணர்ந்து பின் செயற்படுத்த உதவுகிறது.

எ. கா : RUN 50

செயற்பாடு வரி எண் 50-லிருந்து மட்டும் தொடங்குகிறது.

16. SAVE [File Name], [A], [P]; இந்தக் கட்டளை நினைவகத்திலிருக்கும் குறிப்பிடப்பட்ட கோவையின் பெயரை உடைய வழியமைப்பைத் துணை நினைவகத்தில் பதிவு செய்ய உதவுகிறது. இதில் குறிப்பிடப்பட்டுள்ள "A" என்ற குறி வழியமைப்பை ASCII–முறையில் பதிவு செய்யவும், "P" என்ற குறி வழியமைப்புக் கோவையை, மாற்றம் / திருத்தத்திலிருந்து பாதுகாக்கவும் பயன்படுகின்றன. ஏனெனில் வழியமைப்பு Encoded binary படிவத்தில் பதிவு செய்யப்படுவதால் பின்னர் அதை மாற்றவோ திருத்தவோ இயலாது.

17. SYSTEM : இது BBASIC–இல் உள்ள கட்டளையாகும். இதன் மூலம் BBASIC சூழ்நிலையிலிருந்து இயக்கு வழியமைப்புச் சூழ்நிலைக்கு [Operating System] மாற முடியும். BBASIC சூழ்நிலையில் வழியமைப்பைச் சோதித்தறிந்து செயற்படுத்திய பிறகுவேறு கணிமங்களைப் பயன்படுத்தவேண்டுமாயின் இயக்க வழியமைப்புச் சூழ்நிலைக்கு மாறுவதற்கு SYSTEM என்ற கட்டளையைப் பயன்படுத்தவேண்டும்.

18 TRON: வழியமைப்பைச் சோதனை செய்து தவறு திருத்தும்பொழுது [Debugging] பணியை எளிதாக்குவதற்கு இக்கட்டளைஉதவுகிறது. செயற்பாடு நிகழும்போது வழியமைப்பு வரி எண்களை அடைப்புக்குறிக்குள் இது தெரிவிக்கிறது.

இந்தக் கட்டளை மூலம், வழியமைப்பிலுள்ள தவறுகளையும் உட்பிழைகளையும் எளிதில் கண்டுணரலாம். மேலும் தவறுதலான வழியமைப்புகளில் குறிப்பிட்ட வழியமைப்பு வரிகள் தேவையின்றி திரும்பத் திரும்பச் செயற்படுதல் முதலிய பிழைகளை எளிதில் கண்டு திருத்த முடியும்.

எ. கா : [100] [200] [220] [240] [250]......

19. TROFF மேற்குறிப்பிட்ட TRON கட்டளை செய்யும் பணியை நீக்கறவு செய்ய இது உதவுகிறது. TRON கட்டளையைப் பயன்படுத்திய பிறகு TROFF கட்டளையைப் பயன்படுத்தும் பொழுது அடைப்புக்குறிக்குள் இடப்பட்ட வழியமைப்புச் செயற்பாட்டு வரி எண்கள் நீக்கப்படுகின்றன.

20. WIDTH LPRINT [Integer] இக் கட்டளை ஒரு குறிப்பிட்ட வரியில் எவ்வளவு குறிகளை அச்சுப்பொறியில் அச்சு செய்யலாம் என்பதை நிருணயம் செய்கிறது. மேலே கண்ட படிவத்தில் LPRINT' இல்லாவிடில் எவ்வளவு குறிகளை [Integer இல் குறிப்பிட்டபடி] காட்சித் திரையில் ஒரு வரியில் வெளிப்படுத்தலாம் என்பதை நிருணயிக்கிறது. ஒரு வரிக்கு எவ்வளவு குறிகள் என்பது குறிப்பிடப்படாவிடில் கணிப்பொறி தானாகவே ஒரு வரிக்கு 72 குறிகளை நியமித்துக் கொள்ளும். Integerஇன் மதிப்பு 15 இலிருந்து 255 வரை இருக்கலாம்.

எ.கா : WIDTH 60

காட்சித் திரையில் ஒரு வரிக்கு 60 குறிகளை வெளிப்படுத்தக் கட்டளை பிறப்பிக்கிறது.

WIDTH LPRINT 100

அச்சுப்பொறியில் ஒரு வரிக்கு 100 குறிகளை அச்சாக்க ஆணையிடுகிறது.

21. **STOP :** வழியமைப்புச் செயற்படும்பொழுது இடையிலோ, முடிவிலோ அதை நிறுத்த STOP கட்டளை பயன்படுகிறது.

வழியமைப்பு செயற்படும்பொழுது சோதனையிடவும், இடைநிலை விடையைச் சோதித்தறியவும் STOP கட்டளையை வழியமைப்பில் எங்கு வேண்டுமாயினும் பயன்படுத்தலாம். STOP கட்டளை செயற்பட்டபின் செயற்பாடு நின்று கட்டளைச் சூழ்நிலைக்குச் செல்கிறது.

22. **KILL [File Name]:** இது குறிப்பிட்ட கோவையைத் துணை நினைவகமாகிய Disk—லிருந்து அகற்றப் பயன்படுகிறது. இந்தக் கட்டளையைக் கொண்டு வழியமைப்புக் கோவைகள், செய்திக்கூறுக் கோவைகள் [வரிசைக் கிரம அணுக்கக்கோவை, முறையிலா அணுக்கக்கோவை] ஆகியவற்றை Disk—லிருந்து அழிக்கமுடியும்.

எ. கா :

```
KILL "INVENT.DAT"
```

பாகம் – 8

8. பொதுவழியமைப்பு ஆணைகள்

GENERAL PROGRAMMING STATEMENTS

சென்ற பாகத்தில் பேஸிக் மொழியில் பயன்படும் சில முக்கிய மான வழியமைப்புக் கட்டளைகளைப் பற்றி ஆராய்ந்தோம். இந்தப் பாகத்தில் பொது வழியமைப்பாணைகளைப் பற்றியும், அவை எங்ஙனம் பேஸிக் மொழியில் பயன்படுத்தப்படுகின்றன என்பது பற்றியும் ஆராய்வோம்.

பொது வழியமைப்பு ஆணைகளை இரு வகைகளாகப் பிரிக்கலாம்.

1. செயலில்லா ஆணைகள் [Non executable statements]
2. செயல் ஆணைகள் [Executable statements]

எ.கா :

```
10 REM THIS IS A PAYROLL PROGRAM
```

மேற்குறிப்பிடப்பட்ட வழியமைப்பு ஆணை செயலில்லா ஆணை எனக் கூறப்படும்.

இந்த வகை ஆணைகள் வழியமைப்பில் சேர்க்கப்படா விட்டால்கூட, வழியமைப்பின் சரியான செயற்பாடு பாதிக்கப் படுவதில்லை. இவை வழியமைப்பின் தொடக்கத்திலோ, வழி யமைப்பின் இடையிலோ தேவைப்படும் இடங்களில் சேர்க்கப் படுகின்றன. இவற்றின் மூலம் ஒரு வழியமைப்பு ஆவணப்படுத்தப்

படுவதுடன் [Documentation] வழியமைப்பின் தன்மை, அதன் வெவ்வேறு பகுதிகளின் பயன்பாடு, அதில் கையாளப்பட்டுள்ள மாறி, மாறிலிகள் பற்றிய குறிப்பு, வழியமைப்பாளர் பெயர், எழுதப்பட்ட நாள், நிறுவனத்தின் பெயர் ஆகிய விவரங்களைக் குறிக்கவும் பயன்படுகின்றன.

எ.கா:

```
10 PRINT "TAMIL UNIVERSITY"
```

இது செயல் ஆணைக்கு [Executable Statement] எடுத்துக்காட்டு ஆகும். அதாவது செயல் ஆணைகள் ஒரு வழியமைப்பில் கணக்கின் செயற்பாட்டைக் குறிப்பிடும் விதத்தில் அமைந்து உள்ளன. செயல் ஆணைகள் ஒரு கணக்கிற்கான தீர்வை நிருமாணிக்கும் காரணிகளாக அமைகின்றன.

செயல் ஆணைகளைக் [Executable] கீழ்க்கண்ட பிரிவு களாகப் பிரிக்கலாம்.

1. தன்மை ஆணைகள் [Type Statements]
2. கட்டுப்பாட்டாணைகள் [Control Statements]
3. செய்வளை ஆணைகள் [Loop Statements]
4. பிறவகை ஆணைகள் [Other Statements]

8. 1. தன்மை ஆணைகள் (*TYPE STATEMENTS*)

வழியமைப்பில் [Program] சூத்திரங்களையும் [Formula] சமன்பாடுகளையும் [Equations] பயன்படுத்திக் கணக்குகளைச் செய்யவும் எண் கணக்குகளைச் [Arithmetic operation] செய்யவும் பேஸிக் மொழியில் எண்களின் தன்மை வழியமைப்பில் பயன்படும் மாறிகளின் மூலம் குறிப்பிடப்படுதல் இன்றியமையாததாகும். ஒரு கணக்கில் பயன்படும் எண்ணின் மதிப்பு முழு எண்ணாகவோ, ஒற்றைத் துல்லியமாகவோ, இரட்டைத் துல்லியமாகவோ இருக்கலாம். எனவே, கணக்கில் பயன்படும் எண்களின் தன்மையைப் பொறுத்து அதன் விடை அமைவதால், வழியமைப் பில் எண் மாறிகளின் தன்மை குறிப்பிடப்படுதல் இன்றியமையாத

தாகும். இவையன்றி வழியமைப்பில் பயன்படும் மாறிகள் குறிக்கோவை மாறியாக இருப்பின், அதன் தன்மையை வழியமைப்பின் தொடக்கத்தில் குறிப்பிடலாம்.

எ.கா: DEFINT-இது வழியமைப்பில் [Program] மாறிகளை முழு எண்களாக நிருணயம் செய்யப் பயன்படுகிறது.

DEFSNG, DEFDBL, DEFSTR-என்ற ஆணைகள் முறையே மாறிகளின் தன்மைகளான ஒற்றைத் துல்லியம் [Single Precision], இரட்டைத் துல்லியம் [Double Precision], குறிக்கோவை [String] என்பனவற்றைக் குறிப்பிட உதவுகின்றன.

எ. கா :

```
10 DEFINT A-D, M-P
20 DEFSNG X-Z
30 DEFDBL L-P
40 DEFSTR A,L,M
```

மேற்கண்ட எடுத்துக்காட்டில் குறிப்பிடப்பட்டுள்ள முதல் எழுத்தில் தொடங்கும் மாறிகள் யாவும் வழியமைப்பில் பயன்படுத்தப்படும்பொழுது அவற்றின் தன்மைகளுக்கேற்ப முழு எண்ணாகவோ, ஒற்றை அல்லது இரட்டைத் துல்லியமாகவோ, குறிக்கோவையாகவோ அமைகின்றன.

COMMON: இந்த ஆணை ஒரு வழியமைப்பிலுள்ள [Program] மாறிகளின் மதிப்பை மற்றொரு வழியமைப்பிற்கு மாற்றுவதற்குப் பயன்படுகிறது. இந்த ஆணை வழியமைப்பில் எந்த இடத்தில் வேண்டுமானாலும் இருக்கலாம்; ஏனெனில் வழக்கமாக வழியமைப்பின் தொடக்கத்தில் இந்த ஆணை எழுதப்பெறுகிறது. COMMON என்ற ஆணை CHAIN என்ற ஆணையுடன் இணைந்து செயல்படுவதால் முதல் வழியமைப்பில் உள்ள மாறிகள் அனைத்தின் மதிப்பையோ, குறிப்பிட்ட மாறிகளின் மதிப்பையோ அடுத்த வழியமைப்பிற்கு மாற்ற உதவுகிறது.

எ. கா :

```
100 COMMON A,B,C,D,SAN$
120 CHAIN "PAYROLL.BAS",500
```

இதில் A, B, C, D, SAN$ என்ற மாறிகளின் மதிப்பு PAYROLL. BAS என்ற வழியமைப்பிற்கு மாற்றம் செய்யப் படுகின்றது.

8.2. கட்டுப்பாட்டு ஆணைகள் (CONTROL STATEMENTS)

1. GOTO (Line No.)

வழியமைப்பில் [Program] நிர்ப்பந்தம் [Condition] எதுவுமின்றி வேறு ஒரு குறிப்பிடப்பட்ட வரி எண்ணுக்குச் செயற்பாடு பிரிந்து செல்வதற்குக் கட்டுப்பாட்டு ஆணைகள் பயன்படுகின்றன. Line No. இல் குறிப்பிடப்பட்டுள்ள ஆணைக் குக் கட்டுப்பாடு பிரிந்து சென்றபின், செயற்பாடு அதிலிருந்து தொடர்கிறது.

எ.கா :

```
100 GOTO 500
```

2. GOSUB (Line No.)

குறிப்பிட்ட பணியை அல்லது கணக்கியற் செயலைத் திரும்பத் திரும்பச் செய்யவேண்டுமெனில், வழியமைப்பில் தேவைப்படும் இடங்களில் ஒன்றையொத்த பல வழியமைப் பாணைகள் எழுதப் பெறல் வேண்டும். இது வழியமைப்பின் செயல்திறனைக் குறைப்பது மட்டுமன்றி, அதிக அளவு நினை வகத்தையும் எடுத்துக்கொள்ளும். எனவே, ஒரு வழியமைப்பில் குறிப்பிட்ட செயலைப் பலதடவைகள் செய்ய வேண்டுமெனில், துணை வழியமைப்புகளைப் [GOSUB] பயன்படுத்துதல் நலம் பயக்கும். குறிப்பிட்ட செயலுக்கான துணை வழியமைப்பை ஒரு பகுதியில் வைத்துக் கொண்டு, முதன்மை வழியமைப்பில் தேவைப்படும் இடங்களில் GOSUB என்ற ஆணையைப் பயன்

படுத்துவதன் வாயிலாக வழியமைப்பின் செயற்பாட்டுத் திறனை [Program efficiency] அதிகரிக்க இயலும். வழியமைப்பு செயற்படும் பொழுது [Execution], இந்த ஆணையில் குறிப்பிடப்பட்டுள்ள வரிசை எண்ணுக்குக் கட்டுப்பாடு [Control] பிரிந்து செல்கிறது. ஆணையில் குறிப்பிடப்பட்டுள்ள எண்ணிலிருந்து துணை வழியமைப்பு [Sub-Program/Subroutine] தொடர்கிறது. அதன் முடிவில் RETURN என்ற ஆணை அமைந்திருத்தல் வேண்டும். இதன் மூலமாகக் கட்டுப்பாடு [Control] முதன்மை வழியமைப்பிற்குத் [Main Program] திரும்பவும் வருகிறது. பின்னர் முதன்மை வழியமைப்பில் செயற்பாடு தொடாகிறது.

எ.கா : `50 GOSUB 100`

3. IF (Expression) THEN (Statement 1) ELSE (Statement 2)

ஒரு Expression-இன் மதிப்பைப் பொறுத்துக் கட்டாயத்தின் பெயரில் [Condition] வழியமைப்பில் குறிப்பிட்ட பணியைச் செய்யவோ, வேறு இடங்களுக்குச் செயற்பாடு பிரிந்து செல்லவோ இந்த ஆணை உத்திரவிடுகிறது. இதில் Expression-இன் மதிப்பு உண்மையாகவோ, பொய்யாகவோ [True or False] இருக்கலாம்.

எ.கா :
```
100 IF SECOND>FIRST THEN PRINT
        "GREATER" ELSE PRINT "LESSER"
```

மேற்கண்ட எடுத்துக்காட்டில் குறிப்பிடப்பட்டுள்ள SECOND-இன் மதிப்பு FIRST-ஐ விடப் பெரிதாக இருக்கும் நிலையில், 'Expression' இன் மதிப்பு உண்மையாகும். எனவே PRINT "GREATER" என்ற ஆணை செயற்படுத்தப்படும்; அவ்வாறில்லாவிடில் PRINT "LESSER" என்ற ஆணை செயற்படுத்தப்படுகிறது.

4. IF (Expression) GOTO (Line No.)

இது மேலே குறிப்பிடப்பட்ட ஆணையிலிருந்து சற்று வேறுபட்டுள்ளது. இதில் Expression-இன் மதிப்பு உண்மையாக

[True] இருக்குமாயின் குறிப்பிடப்பட்டுள்ள வரி எண்ணுக்குக் கட்டுப்பாடு [Control] செல்கிறது. மாறாக Expression-இன் மதிப்பு பொய்யாக [False] இருப்பின், செயற்பாடு அதற்கடுத்த வழியமைப்பாரணையிலிருந்து தொடர்கிறது.

எ.கா: `200 IF DOM = DIC GOTO 500`

5. IF (Expression) THEN (Line No.)

இந்த ஆணையும் மேற்குறிப்பிடப்பட்ட பணியையே செய்கிறது. ஆனால், இதன் அமைப்பில் சற்று மாறுபாடு உள்ளது. இந்த ஆணையில் GOTO என்ற சொல் விடப்பட்டு, அதற்குப் பதிலாக THEN சேர்க்கப்பட்டுள்ளது. எனினும் செயல் முறையில் இந்த ஆணை மேற்கண்ட எடுத்துக்காட்டில் உள்ள ஆணை செய்யும் பணியையே செய்கிறது.

எ. கா: `500 IF COUNT > 100 THEN 2000`

6. Nesting of IF Statements

மேலே விவரிக்கப்பட்ட IF.....THEN.....ELSE ஆணைகள் கணக்கின் செயற்பாட்டுத் திறனுக்கேற்பப் பயன்படுத்தவும், பலதரப்பட்ட நிபந்தனைகளைச் சோதித்து அதற்கேற்பக் குறிப்பிட்ட செயலை வழியமைப்பில் செய்யுமாறு கணிப்பொறிக்கு அறிவுறுத்தவும் உதவுகின்றன. இவ்வகையான வழியமைப்பாணைகள், வளைக்கட்டுப்பாட்டாணைகள் [Nested Control Statements] வகையைச் சார்ந்தனவாகும். இவ்வகை வழியமைப்பாணைகள் எவ்வாறு உருவாக்கப்படுகின்றன என்பதைக் கீழ்க்கண்ட எடுத்துக்காட்டு விளக்குகிறது.

எ. கா

```
50 IF DAN > DICK  THEN  PRINT
      "DAN IS ELDER THAN DICK" ELSE
      IF DICK > DAN THEN PRINT "DAN
      IS  YOUNGER  THAN DICK"  ELSE
      PRINT "TWINS"
```

7. ON (Expression) GOTO (N_1, N_2, N_3,.........)

ON (Expression) GOSUB (N_1,N_2,N_3, ...)

இதில் Expression-இன் மதிப்பிற்குத் தகுந்தாற்போன்று குறிப்பிட்ட வரி எண்ணுக்குக் [Line No.] கட்டுப்பாடு பிரிந்து செல்கிறது. Expression-இன் மதிப்பு ஒன்றாக இருக்கும்பொழுது வழியமைப்பின் கட்டுப்பாடு [Program Control] வரி எண் N_1 -க்குப் பிரிந்து செல்கிறது. Expression-இன் மதிப்பு இரண்டாக இருக்கும்பொழுது வழியமைப்பின் கட்டுப்பாடு [ProgramControl] வரி எண் N_2 -க்குப் பிரிந்து செல்கிறது. அதேபோன்று Expression-இன் மதிப்பைப் பொறுத்துக் கட்டுப்பாடு குறிப்பிட்ட துணை வழியமைப்புகளுக்குப் பிரிந்து செல்கிறது.

எ.கா : 100 ON VALUE GOTO 500, 600, 700

'VALUE' -இன் மதிப்பு (1, 2, 3) ஆக இருக்கும் பொழுது, முறையே கட்டுப்பாடு வரி எண் 500, 600, 700 -க்குப் பிரிந்து செல்கிறது.

எ. கா : 200 ON VALUE GOSUB 600, 700, 800

'VALUE' -இன் மதிப்பிற்கேற்பக் கட்டுப்பாடு முறையே 600, 700, 800 என்ற வரி எண்களில் துவங்கும் துணை வழியமைப்புகளுக்குச் செல்கிறது.

8. ON ERROR GOTO (Line Number)

வழியமைப்பு செயற்படும்பொழுது அதில் தவறு ஏதேனும் இருப்பின் உடனே இதில் குறிப்பிடப்பட்டுள்ள வரி எண்ணுக்குக் கட்டுப்பாடு பிரிந்துசெல்ல இந்த ஆணை உதவுகிறது. இதில் குறிப்பிடப்பட்டுள்ள வரி எண்ணிலிருந்து பிழைகளை கையாளும் துணை வழியமைப்பு [Error handling subroutine] துவங்குகிறது.

எ.கா : 20 ON ERROR GOTO 500

8.3. செய்வளை ஆணைகள் (LOOP STATEMENTS)

1. **FOR :** இந்த ஆணை "DO" என்று கூறப்படும் போர்ட்ரானிலுள்ள [Fortran] ஆணைக்குச் சமமானதாகும். வழியமைப்பில் சில குறிப்பிட்ட பணியைத் திரும்பத் திரும்பப் பல தடவைகள் செய்வதற்குச் செய்வளை ஆணைகள் பயன்படுத்தப் படுகின்றன. இந்த ஆணையின் படிவம் கீழ்க்கண்டவாறு அமையும்.

```
FOR [Variable] = A TO B [STEP C]
. . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . .
NEXT [Variable]
```

இதில் **Variable** ஒரு எண்பானாகப் [Counter] பயன்படுகிறது. இந்த ஆணையை அடுத்து NEXT [Variable] வரை இடையிலுள்ள ஆணைகள் [. . . . ஆல் குறிப்பிடப்பட்டுள்ளவை] யாவும் Variable -இன் மதிப்பு A-யில் தொடங்கி B-யாக [ஒவ்வொரு சுற்றிலும் Variable-இன் மதிப்பு C அளவு அதிகரிக்கிறது] மாறும் வரை செயற்படுத்தப்படுகின்றன. ஒவ்வொரு சுற்றுக்கும் அதன் படியளவு [Step Size] குறிப்பிடப்படாவிடில் அது ஒன்றாகக் கருதப்படும்.

2. **NEXT:** ஒரு வழியமைப்பில் FOR ஆணை இருக்கும் பொழுது, NEXT ஆணையும் இருத்தல் வேண்டும். அவ்வாறு இல்லாவிடில் வழியமைப்பில் இலக்கணப் பிழை அல்லது வெளிப் பிழை [Syntax error] இருப்பதாகக் கணிப்பொறி எடுத்துரைக்கும்.

மேலே காட்டப்பட்ட படிவத்தில் குறிப்பிடப்பட்டபடி FOR ... NEXT ஆணை வழியமைப்பில் பயன்படுத்தப்படல் வேண்டும். இந்த ஆணை, பல செயல்களைப் பல தடவைகள் செய்யப் பயன்படுவதால் இது திறமை வாய்ந்த செய்வளை ஆணையாகக் கருதப்படுகிறது. இந்த ஆணை எவ்வாறு வழியமைப்பில் பயன்படுத்தப்படுகிறது என்பதைக் கீழ்க்கண்ட எடுத்துக்காட்டு மூலம் விளக்கலாம்.

எ.கா :

```
10 LET N1# = 100:N2# = 5
20 FOR I=1 TO 5
30 RESULT = I*5+N1#-N2#
40 PRINT RESULT ; ",";
50 NEXT I
60 END
   RUN
```

விடை : 100, 105, 110, 115, 120

மேற்கண்ட எடுத்துக்காட்டில் உள்ள வழியமைப்பில் [Program] வரி எண் 30-இல் உள்ள சமன்பாட்டுக்கு 5 தடவைகள் தீர்வு காண்பதற்குச் செய்வளை பயன்படுத்தப்படுகிறது. NEXT என்ற ஆணை செய்வளை [LOOP] யின் முடிவைத் தருகிறது. FOR ஆணையில் குறிப்பிடப்பட்டுள்ள மாறியின் மதிப்பு, மொத்த செய்வளையின் மதிப்பைவிட அதிகமாகும்பொழுது செய்வளை முடிவு பெறுகிறது.

செய்வளை ஆணையைப் பல அடுக்குகள் நிலையிலும் [Nesting] பயன்படுத்தலாம். அவ்வாறு பயன்படுத்தும்பொழுது FOR ... NEXT ஆகியவற்றின் அமைப்பு சரியான முறையில் அமைய வேண்டும். மேலும் தீர்வுகாணப்பட வேண்டிய கணக்கின் தன்மை, மற்றும் அதன் விடைக்கேற்பத் தேவைப்படும் சமன் பாடுகளை FOR... NEXT செய்வளை அடுக்குகளின் [Nesting] தகுந்த இடங்களில் பொருத்துதல் இன்றியமையாததாகும். FOR ... NEXT செய்வளை ஆணையைப் பல அடுக்குகள் நிலையில் எங்ஙனம் பயன்படுத்துவது என்பதைக் கீழ்க்காணும் எடுத்துக் காட்டு விளக்குகிறது.

எ. கா :

```
┌-----> 10 FOR I=1 TO 20
|
|         20 CNT=I*4
|
|   ┌--> 30 FOR J=1 TO 100
|   |
|   |     40 RESULT = J*5+CNT-20
|   |
|   |     50 PRINT RESULT ; ",";
|   |
|   └--> 60 NEXT J
|
└-----> 70 NEXT I

          80 END
```

3. WHILE (Expression)/WEND

இந்த வழியமைப்பு ஆணை FOR...NEXT – ஆணை செய்யும் செயலையே செய்கிறது. ஆனால், இதன் அமைப்பில் சில மாறுபாடுகள் உள்ளன. இதில் குறிப்பிடப்பட்ட வழியமைப்பு ஆணைகளை [Programming Statements] Expression—இன் மதிப்பு உண்மையாக [True] இருக்கும் வரை இடையில் அமைந்துள்ள வழியமைப்பாணைகளைத் திரும்பத் திரும்பச் செயல்படுத்துகிறது. WEND-இந்த ஆணை, செய்வளையின் முடிவைக் குறிக்கும். இந்த WHILE/WEND-செய்வளைக்குள் [LOOP] குறிப்பிடப்பட்டுள்ள செயல்கள் யாவும் முதற் சுற்றில் செய்து முடிக்கப்பட்ட பின்னர், WEND என்ற தொடரைச் கண்டதும் செயற்பாடு திரும்பவும் WHILE தொடர் உள்ள வரி எண்ணுக்கு மீண்டும் வருகிறது. பின் Expression—இன் நிலை சோதிக்கப் படுகிறது. அதன் மதிப்பு உண்மையெனில் இரண்டாம் சுற்று தொடர்கிறது. அவ்வாறில்லையெனில் WHILE/WEND செய்வளையின் செயற்பாடு முடிவுறுகிறது. ஒருவழியமைப்பில் WHILE/WEND எவ்வாறு பயன்படுத்தப் படுகிறது என்பதைக் கீழே கொடுக்கப் பட்டுள்ள வழியமைப்பின் மூலம் காணலாம்.

எ. கா :

```
10 P=0 : Q=100
20 WHILE P < Q
30 PRINT "THE VALUE OF P IS"; P
40 LET P=P+4
50 WEND
60 STOP
```

8.4. பிறவகை ஆணைகள் (*OTHER STATEMENTS*)

1. CALL (Variable Name), (Argument List)

இந்த ஆணை, பேஸிக் [Basic] வழியமைப்பிலிருந்து [Program] Name—என்ற பெயரில் உள்ள இடைநிலை மொழியிலமைந்த [Assembly Language] துணை வழியமைப்பிற்குக் [Subroutine] கட்டுப்பாடு பிரிந்துசெல்லப் பயன்படுகின்றது. இதில் இடைநிலை மொழியிலுள்ள [Assembly Language] துணை வழியமைப்பின் [Subroutine] தொடக்க நினைவக எண் [Memory Location] கொடுக்கப்படல் வேண்டும்.

[Variable Name] : முதன்மை வழியமைப்பிலிருந்து கட்டுப்பாடு இடைநிலை மொழியிலுள்ள துணைவழியமைப்பிற்குப் [Assembly Language Subroutine] பிரிந்து செல்வதற்குத் தேவையான நினைவகச் சுட்டு எண் [Memory Address] இதன் மூலம் அறிவுறுத்தப்படுகிறது.

[Argument List] : முதன்மை வழியமைப்பிலிருந்து சில மாறி, மாறிலிகளின் மதிப்புகள் [Arguments] இடைநிலை மொழியிலுள்ள துணை வழியமைப்புகளுக்கு அனுப்பப்படுவதற்குரிய குறியீடுகளாக இவை பயன்படுகின்றன.

எ.கா :
```
10 SUBSHIFTS & HA000
        20 CALL, SUBSHIFT(M,N,O)
```

2. DIMENSION: DIM (List of Subscripted Variables)

இந்த ஆணையை வழியமைப்பில் [Program] பயன்படுத்துவதன் மூலம் வழியமைப்பில் பயன்படுத்தப்படும் நினைவக வரிசை மாறிகளுக்குத் [Subscripted Variables] தேவையான நினைவக ஒதுக்கீடு செய்யப்படுகிறது. இந்த ஆணையைப் பயன்படுத்தாமல் கூட ஒரு நினைவக வரிசை மாறியினுடைய வரிசை எண் [Subscript] *10* வரை இருக்கலாம். வழியமைப்பில் நினைவக வரிசையின் மதிப்பு *10* -இற்கு மேல் செல்லும்பொழுது பிழைச் செய்தி அறிவிக்கப்படும்.

எ.கா :
```
60 DIM ROW(55), COLUMN(25), BOX(35,100)
```

3. END : வழியமைப்பின் இயக்கச் செயல் [Execution] நிறுத்தப்படுவதற்கு இந்த ஆணை பயன்படுகிறது. இந்த ஆணை வழியமைப்பில் செயற்படும்பொழுது செய்திக் கூறுக்கோவைகள் [Data files] அனைத்தையும் பாதுகாக்கிறது. பிறகு கட்டளைச் சூழலுக்கு வருகிறது. ஒரு வழியமைப்பில் இது பயன்படுத்தப் படாதுபோனாலும் செயற்பாட்டில் இடையூறு நேர்வதில்லை.

எ.கா :
```
400 IF ENDBOOK > 1700  THEN  PRINT
       "BOOK READ":STOP  ELSE GOTO 800
  - - - -
  - - - -
  - - - -

800 REM CONTINUE READING THE BOOK
```

4. ERASE (List of Variables) : மேற்குறிப்பிடப்பட்ட DIMENSION - ஆணையில் நினைவக ஒதுக்கீடு செய்யப்படுகிறது. ERASE ஆணையைப் பயன்படுத்தும்பொழுது அவ்வாறு ஒதுக்கீடு செய்யப்பட்ட நினைவகம் விடுவிக்கப்பட்டு வழியமைப்பில்

[Program] வேறுவகையில் பயன்படுத்தப்படுவதற்கு உதவுகிறது. இந்த ஆணையைப் பயன்படுத்துவதன் மூலம் வழியமைப்பில் முன்பே பயன்படுத்தப்பட்ட சில மாறிகளை நீக்க முடிகிறது. இதன் வாயிலாக நினைவக அளவு விடுவிக்கப்படுவதால் வழியமைப்பைச் செயற்படுத்துவதற்குத் தேவையான அளவு நினைவகம் கிடைக்கிறது.

எ.கா : `10 ERASE ONE, TWO`

5. **ERR:** தவறுகளைக் கையாளும் துணை வழியமைப்புகள் [Error Handling Subroutines] செயற்படுத்தப்படும்பொழுது [Execution], ERR -இல் பிழை எண் [Error Code] பதிவு செய்யப்பட்டிருக்கும். ERL என்ற மாறியில் பிழை காணப்பட்ட வரி எண் பதிவு செய்யப்படுகிறது. ERR மற்றும் ERL ஆணைகள் நேரடியான IF...THEN ஆணையுடன் இணைந்து செயற்படுகின்றன.

எ.கா : `100 IF ERR=300 THEN GOSUB 6000`

6. **LET :** இது கணக்கியற் செயல்களின் [Arithmetic Operations] மாறிகளுக்குக் [Variables] குறிப்பிட்ட மாறிலியின் மதிப்பைச் சமன்படுத்தவும் அல்லது கணக்கிட்ட விடையை மாறிகளில் சேமிக்கவும் பயன்படுகிறது. மாறிகளையோ, விடையையும் மாறியையுமோ சமன்படுத்தும் பொழுது LET கட்டாயமாகப் பயன்படுத்தப்பட வேண்டிய தேவையில்லை. பயன்படுத்தப் படாது போனலும் விடை ஒன்றே. i.e, LET is only optional.

எ. கா :
```
10 LET NUM = 100

20 LET COUNT = 10.5 * NUM/3.6
```

7. **OPTION BASE n :** ஒரு array Subscript-இன் குறைந்த அளவு மதிப்பை இவ்வாணை மூலம் குறிப்பிடலாம், அதாவது n-இன் மதிப்பே Array Subscript-இன் குறைந்த அளவு மதிப்பாகும்.

எ. கா : `10 OPTION BASE 3`

8. **OUT I, J :** இந்த ஆணையைப் பயன்படுத்துவதன் மூலம் J யில் உள்ள செய்தியை [Data] PORT I இற்கு மாற்ற

முடியும். I, J-யின் மதிப்பு முழு எண்ணாகவும் 0 இலிருந்து 255 க்கு மிகாமலும் இருக்க வேண்டும்.

எ.கா : 50 OUT 30, 165

9. POKE X. Y: POKE ஆணை நினைவக எண் X-இல் Y-இன் மதிப்பை இருத்துகிறது. நினைவக முகவரியைச் சுட்டும் எண்ணான [Memory address] X-உம், செய்திக்கூறான [Data] Y-உம் முழு எண்ணாக இருத்தல் வேண்டும். Y-இன் மதிப்பு 0-விலிருந்து 255 வரையும் X-இன் மதிப்பு 0-விலிருந்து 65,536 வரையும் இருக்கலாம். இந்த ஆணை இடைநிலை வழியமைப்பு மொழியில் [Assembly Language] பெரிதும் பயன்படுகிறது.

எ.கா : 300 POKE 12235, 225

10. WAIT (Port Number), (I), (J)

இந்த ஆணை பெரும்பாலும் இடுவரல், விடுவரல் Port களின் இயக்க நிலையைச் சோதிக்க உதவுகிறது. வழியமைப்புச் செயற்பாட்டினைத் [Program execution] தற்காலிகமாக நிறுத்தி வைக்கவும், இடுவரல், விடுவரல் Port களின் நிலையைக் கண் காணிக்கவும் WAIT ஆணை பயன்படுகிறது. செயற்பாடு தற்காலி மாகத் தடைப்படும்பொழுது I/O Port-லிருந்து செய்திக் கூறு களைப் (Data) பெறவும், பின்னர் அதைக் கொண்டு J,I உடன் AND,OR முதலிய முறையமைப் பணிகளைச் (Logical Operation) செய்யவும் பயன்படுகிறது.

எ.கா : 200 WAIT 50, 10, 5

11. RANDOMIZE (Expression)

முறையிலா எண்களை [Random Number] உருவாக்கு வதற்கு இவ்வாணை பயன்படுகிறது. ஒவ்வொரு தடவையும் இந்த ஆணையைப் பயன்படுத்தும் பொழுது இந்த எண் உருவாக்க வரிசையில் (Creation Sequence) மாற்றங்கள் ஏற்படுகின்றன. இந்த ஆணையைப் பயன்படுத்தும் பொழுது Expression-விடுபட்டிருந்தால் செயற்பாட்டின் இடையில் தடை ஏற்பட்டு Random Number [−32,768 to 32,767]? என்ற வினா வினவப் படும். Expression—இன் மதிப்பு −32,768 முதல் + 32,767

வரை இருக்கலாம். வழியமைப்பின் தொடக்கத்தில் RANDOMIZE என்ற ஆணையைப் பயன்படுத்தி, பின்னர் RND என்ற ஆணையைப் பயன்படுத்துவதன் மூலம் வெவ்வேறு விதமான முறையிலா எண்களை உருவாக்க இயலும்.

12. RESUME ஆணை

படிவம்—(Format)

[i] RESUME or RESUME 0
[ii] RESUME NEXT
[iii] RESUME (Line Number)

வழியமைப்பின் தவறிலிருந்து விடுபட்டு வழியமைப்புச் செயற்பாடு [Program Execution] தொடர்வதற்கு இந்த ஆணை உதவுகிறது. முதல்வகை ஆணை பிழை நிகழ்ந்த வரி எண்ணிலேயே செயற்பாடு தொடர்வதற்கும், இரண்டாம் வகை ஆணை பிழை நிகழ்ந்த வழியமைப்பு வரிக்கு அடுத்த வரியில் செயற்பாடு தொடர்வதற்கும் பயன்படுகின்றன. மூன்றாம் வகை ஆணை, வழியமைப்புச் செயற்பாட்டில் பிழை நிகழ்ந்த பின்னர் குறிப்பிடப்பட்ட வரி எண்ணுக்குக் [Line Number] கட்டுப்பாடு பிரிந்து செல்வதற்கு உதவுகிறது.

எ.கா: 600 RESUME 500

இதில் தவறிலிருந்து விடுபட்டபின் வழியமைப்பின் செயற்பாடு வரி எண் 500 லிருந்து தொடங்குகிறது.

13. SWAP (Variable 1), (Varriable 2)

Variable 1, Variable 2 -என்ற மாறிகளின் மதிப்பை ஒன்றுக்கொன்று மாற்றிக் கொள்வதற்கு இந்த ஆணை உதவுகிறது. இவ்வாறு மாறிகளின் மதிப்பு ஒன்றிலிருந்து மற்றொன்றுக்கு மாற்றப்படும் பொழுது இரண்டு மாறிகளின் தன்மையும் ஒரே மாதிரி இருத்தல் வேண்டும். இல்லையெனில், 'Type Mismatch' என்ற பிழைச் செய்தி வெளிப்படுத்தப்படும். மாறிகளின் தன்மை முழு எண்ணாகவோ, ஒற்றை அல்லது இரட்டைத் துல்லியமாகவோ, குறிக்கோவையாகவோ இருக்கலாம்.

எ.கா : 60 SWAP N1, M1

N1 -இன் மதிப்பை M1 -இற்குமாற்றவும் M1 -இன் மதிப்பை N1-இற்கு மாற்றவும் இந்த ஆணைப் பயன்படுகிறது.

14. STOP

வழியமைப்பின் செயல்பாட்டை [Program Execution] இந்த ஆணை தற்காலிகமாக நிறுத்துகிறது. "Break" -என்ற செய்தியைத் தெரிவித்தபின் கட்டளை நிலைக்கு [Command Mode] வழியமைப்பு செல்கிறது. இந்த ஆணையைப் பயன் படுத்துவதன் மூலம் வழியமைப்புச் செயற்பாட்டை இடையில் தற்காலிகமாக நிறுத்தவும் இடைநிலை விடைகளைச் சோதித் தறியவும் முடியும். STOP ஆணை பயன்படுத்தப்படும் முறை கீழ்க்கண்ட எடுத்துக் காட்டில் விளக்கப்பட்டுள்ளது.

எ. கா :
```
10 INPUT A$, B$
20 PRINT "ADDRESS"
30 STOP
40 INPUT PIN
50 STOP
RUN
 TAMIL UNIVERSITY, THANJAVUR
ADDRESS
PRINT A$, B$
TAMIL UNIVERSITY, THANJAVUR
 613001
PRINT PIN
613001
```

கணிப்பொறி ஒருங்கும்
பேஸிக் மொழியும்

பாகம் – 9

9. துணை நினைவகக் கோவை ஆணைகளும் கூறுகளும்

(DISK BASED INPUT/OUTPUT STATEMENTS AND FUNCTIONS)

பேஸிக் மொழியில் பயன்படுத்தப்படும் சில முக்கியமான வழியமைப்பு ஆணைகளைப் பற்றி முன்பே கவனித்தோம். அவை இடுவரல், விடுவரல் [Input-Output] மற்றும் கூறு ஆணைகள் [Functions], கோவை ஆணைகள் [File Statements] இவற்றிலிருந்து வேறுபட்டுள்ளன. இந்தப் பகுதியில் ஆராயப் படும் BBASIC மற்றும் ABASIC -இல் உள்ள வழியமைப்பு ஆணைகளைக் [Programming Statements] கீழ்க்கண்டவாறு பிரிக்கலாம்.

1. இடுவரல் / விடுவரல் ஆணைகள்
   [Input/Output Statements]

2. ABASIC, BBASIC -இன் கோவை ஆணைகள்
   [Disk based Input Output Statements]

3. வழியமைப்புக் கூறுகள் [Functions]

4. கட்டுப்பாட்டுக் கட்டளைகள் [Control Statements]

5. திருத்து முறைக்கட்டளைகள் [Edit Sub-Commands]

9.1. இடுவரல்/விடுவரல் ஆணைகள் (INPUT/OUTPUT STATEMENTS)

பேஸிக் [Basic] மொழியில் வழியமைப்புச் [Program] செய்யும்பொழுது, வழியமைப்பாளர்கள் பல வழியமைப்பு ஆணைகளைப் பயன்படுத்துகின்றனர். ஒரு கணக்குக்குத் தீர்வு காண்பதற்குத் தேவையான செய்திக் கூறுகளைக் [Data] கணிப் பொறியில் இடுவரல் [Input] செய்வதற்கும் விடையைத் தகுந்த படிவங்களில் [Formats] பெறுவதற்கும், கணிப்பொறி யுடன் வழியமைப்பாளர் நேரடித் தொடர்பு கொள்வதற்கும் [Interactive Facility], இடுவரல்/விடுவரல் ஆணைகள் பெரிதும் உதவுகின்றன. மேலும் இடுவரல், விடுவரல் [Input and Output] ஆணைகள் செய்திக்கூறுகளைத் [Data] துணை நினைவக மாகிய மின்காந்தத்தட்டு [Magnetic Disk], மென்வளைத் தட்டு [Floppy Disk], இயக்குநிலையகம் [Console], அச்சுப் பொறி [Printer], முதன்மை நினைவகம் [Memory] ஆகிய வற்றிற்கிடையே பரிமாற்றம் [Transfer] செய்துகொள்ளப் பெரிதும் உதவுகின்றன.

9.1.1. இடுவரல் ஆணைகள் (INPUT STATEMENTS)

இடுவரல் ஆணைகளின் உதவியுடன் மட்டுமே நாம் கணிப் பொறியுடன் தொடர்பு கொள்ளவும், உரியமுறையில் கட்டளை யிடவும் முடியும். வழியமைப்பின் மூலமோ, நேரடித் தொடர்பு முறையில் [Interactive Operation] கட்டளைச் சூழ்நிலையிலோ [Command mode] இடுவரல் ஆணைகளைப் பயன்படுத்திச் செய்திக் கூறுகளையோ, தேவையான கட்டளைகளையோ கணிப்பொறிக்கு அறிவுறுத்த இயலும். பேஸிக் மொழி நேரடித் தொடர்பு முறை [Interactive Facility] வசதி வாய்க்கப் பெற்ற மொழியாதலால், அதில் பயன்படும் இடுவரல் ஆணைகள் யாவும் கையாள மிகவும் எளியனவாகவும், சக்தி வாய்ந்தனவாக வும் உள்ளன. இவற்றைப் பற்றிக் கீழ்க்கண்டவாறு விளக்கிக் கூறலாம்.

1. INPUT (Variable List)

இந்த ஆணை மூலம் மாறிகளுக்கான மதிப்பை நேரடித் தொடர்பு முறையில் [Interactive Facility] இடுவரல் செய்ய

முடிகிறது. இதில் குறிப்பிடப்பட்டுள்ள மாறிகளின் எண்ணிக்கை மற்றும் அதன் தன்மைக்கேற்ப இடுவரல் செய்யப்படல் வேண்டும். இல்லையெனில் தக்க பிழைச் செய்தி அறிவிக்கப்பட்டு மீண்டும் செய்திக் கூறுகளை இடுவரல் செய்யுமாறு கணிப்பொறி பணிக்கும்.

எ.கா :

```
80 INPUT ANY, NAN, TAM$
```

2. INPUT (Prompt String), (List of Variables)

இது மாறிகளின் [Variables] மதிப்பைக் காட்சித் திரையின் மூலம் பெறுவதற்குப் பயன்படுகிறது. இடுவரல் ஆணை செயற்படும் பொழுது காட்சித் திரையில் ஒரு வினாக்குறி (?) வெளிப்படுத்தப்பட்டுச் செய்திக்கூறுகளை [Data] இடுவரல் செய்யும்படி பணிக்கும். ஆனால், இதில் குறிப்பிடப்பட்டுள்ள Prompt String இருக்குமேயானால், வினாக் குறிக்கு முன்பாகச் செய்தி காட்சித் திரையில் வெளிப்படுத்தப்படும். இந்த ஆணையில் ஒன்றுக்கும் மேலான மாறிகள் [Variables] இருக்கும் பொழுது, காற்புள்ளி (,) களால் அவை பிரிக்கப்படும். மாறிகள் எண்களாகவோ குறிக்கோவை [String] யாகவோ இருக்கலாம்.

எ.கா :

```
1000 INPUT " EMPLOYEE NUMBER, NAME, DESIGNATION ";EMP,NAM$,DES$
```

செய்தியும் இடுவரல் செய்திக் கூறும்

```
EMPLOYEE NUMBER, NAME, DESIGNATION ? 500, SELVAN, MANAGER
```

3. LINE INPUT (;) (Prompt String) (;) (String Variable)

இந்த ஆணை மூலம் 255 எண்ணிக்கையுள்ள குறிகளை இடையில் Delimiter எதுவுமின்றித் தொடர்ச்சியாக இடுவரல் செய்ய இயலும். இதில் குறிப்பிடப்பட்டுள்ள Prompt String பயன்படுத்துவோர்க்குச் செய்திக் கூறுகளின் தன்மை, மற்றும் எவ்வாறு இடுவரல் செய்வது முதலிய குறிப்புகளை அறிவுறுத்து

கிறது. Prompt செய்தியின் முடிவிலிருந்து சட்டத்திருப்பை [CR] அழுத்தும்வரை, இடுவரல் செய்திக்கூறு, குறிக்கோவை மாறியில் இருத்தப்படுகிறது.

எ. கா : `100 LINE INPUT " ADDRESS ";ADD$`

4. READ (List of Variables)

READ ஆணை வழியமைப்பின் செயற்பாட்டின்பொழுது DATA ஆணையில் கொடுக்கப்பட்டுள்ள செய்திக் கூறுகளைக் [Data] குறிப்பிடப்பட்ட மாறிகளுக்குப் படித்தளிக்கும். READ ஆணை எப்பொழுதும் DATA ஆணையுடன் மட்டுமே செயற்படும். READ-DATA செயற்படும்பொழுது செய்திக் கூறுகளை, ஒவ்வொன்றாக படித்து மாறிகளுக்குச் சமன்படுத்துகிறது. READ ஆணையில் மாறிகளின் பட்டியலிலுள்ள மாறிகளின் தன்மை, மற்றும் மாறிகளின் எண்ணிக்கை, DATA ஆணையில் கொடுக்கப் பட்டுள்ள செய்திக்கூறின் தன்மை, மற்றும் எண்ணிக்கையை ஒத்திருத்தல் மிகவும் இன்றியமையாததாகும். இல்லையெனில் பிழைச் செய்தி அறிவிக்கப்படும். DATA ஆணையில் குறிப்பிடப் பட்ட செய்திக் கூறைத் திரும்பப் படிக்க வேண்டுமாயின், RESTORE என்ற ஆணையைப் பயன்படுத்தல் வேண்டும்.

எ. கா :
```
10 READ NUM%, NN, XY$

20 DATA 100, 25.67, "NET SALARY"
```

5. DATA (List of Values)

DATA : மேற்குறிப்பிடப்பட்ட READ ஆணையில் கொடுக்கப்பட்ட மாறிகளின் [Variables] மதிப்பு, DATA ஆணையில் செய்திக் கூறாகக் [Data] கொடுக்கப்பட்டிருக்கும். வழியமைப்பில் [Program] READ ஆணையைக் கண்டவுடன் கணிப்பொறி DATA ஆணையில் கொடுக்கப்பட்டிருக்கும் செய்தியைப் படித்து மாறிகளுக்குச் சமன்படுத்துகிறது. DATA ஆணையில் கொடுக்கப்பட்டிருக்கும் செய்திக்கூறுகள் ஒவ்வொன்றும் காற்புள்ளியால் பிரிக்கப்பட்டிருக்கும். இதிலுள்ள செய்திக் கூறுகள் [Data] எண்களாகவோ, குறிக்கோவையாகவோ,

READ ஆணையில் குறிப்பிடப்பட்டுள்ள மாறிகளின் தன்மைக் கேற்ப அமைந்திருக்கும். செய்திக் கூறுகளைக் குறிப்பிடும் பொழுது, ஒரு வழியமைப்பு வரியில் முழுவதையும் குறிப்பிட முடியவில்லையெனில் அடுத்த வரியில் வரி எண்ணும் DATA -வும் கொடுத்துப் பின்னர், செய்திக் கூறுகளைத் தொடர்ந்து குறிப்பிடலாம்.

எ. கா : 50 DATA 100, 25.67, "NET SALARY"

60 DATA 755, "TAMIL UNIVERSITY"

6. INPUT (Prompt String) LINE (Variable 1)

இந்த ஆணை ABASIC மொழியில் மட்டும் உள்ளது. இது BBASIC மொழியில் பயன்படுத்தப்படும் இடுவரல் ஆணையை ஒத்ததாகும். ஆனால் இதன் மூலம் 255 குறிகளை இடுவரல் செய்யமுடியும்.

எ.கா : 10 INPUT "NAME"; LINE NAM$

7. RESTORE (Line Number)

READ—DATA ஆணைகளைப் பயன்படுத்திச் செய்திக் கூறுகளைப் [Data] படித்தறிந்த பிறகு, திரும்பவும் அதே செய்திக் கூறுகளைப் [Data] படித்தறிவதற்கு RESTORE பயன்படுகிறது. வழக்கமாக RESTORE ஆணையை வழியமைப்பில் செயற்படுத்தும் பொழுது, அடுத்த READ ஆணையில் குறிப்பிடப்பட்டுள்ள மாறிகளுக்கு முதலிலுள்ள DATA ஆணையில் அமைந்துள்ள செய்திக்கூறுகளை [Data] வரிசையாகப் படித்தளிக்கும்.

எ.கா : 900 RESTORE 1000

9.1.2. விடுவரல் ஆணைகள் (*OUTPUT STATEMENTS*)

மேலே கண்ட இடுவரலாணைகள், மூலம் செய்திக் கூறுகளும் [Data], கட்டளைகளும் [Commands] இடுவரல்

செய்யப்படுகின்றன. பின்னர் கணிப்பொறியின் மையச் செயலகம் வழியமைப்பைச் செயற்படுத்தியபின் நமக்குத் தேவையான விடையை அளிக்கிறது.

கணிப்பொறி மூலம் கணிக்கப்பட்ட விடை, பகுக்கப்பட்ட செய்திகள் ஆகியவை கணிப்பொறியின் காட்சி திரையிலோ, அச்சுபொறியிலோ நமக்குக் கிடைக்கின்றன.

1. PRINT or WRITE (List of Variables)

இதன் மூலம் செய்திகள் காட்சித் திரையில் வெளிப் படுத்தப்படுகின்றன. மாறிகளுக்கு [Variables] இடையே அரைப் புள்ளிகள் [;] இருக்குமாயின் விடைகள் தொடர்ச்சியாகக் காட்சித் திரையில் வெளிப்படுத்தப்படும். அவ்வாறன்றி மாறிகள் காற்புள்ளிகளால் பிரிக்கப்பட்டிருப்பின் பதினான்கு வெற்றிடங் கள் இடைவெளியில் [14 Spaces] விடை வெளிப்படுத்தப்படும்.

PRINT-ஆணையைத் தொடர்ந்து பல மாறிகள் கொடுக்கப் பட்டிருக்கும் பொழுது, அவற்றைக் காட்சித் திரையில் 80 குறி நிலைகளுக்குள் வெளிப்படுத்த முடியவில்லையெனில் மீதமுள்ள விடை அதற்கடுத்த வரியில் வெளிப்படுத்தப்படும். PRINT — ஆணையை எங்ஙனம் பயன்படுத்துவது என்பதைக் கீழ்க்காணும் சிறிய வழியமைப்பு விளக்குகிறது.

எ.கா :

```
10 A=50:B=60:C=70
20 PRINT A,B,C
30 PRINT A;B;C
RUN
```

விடை :

```
50            60            70
50 60 70
```

வழியமைப்பில் PRINT ஆணைமட்டும் இருக்கும்பொழுது ஒரு வரி வெற்றாகக் காட்சித் திரையில் நகருகிறது [Prints a blank Line].

2. (PRINT or WRITE) TAB1, Variable, TAB2...

இந்த ஆணையைப் பயன்படுத்துவதன் மூலம் காட்சித் திரையில் குறிப்பிட்ட படிவத்தில் [Specified Format] விடையை வெளிப்படுத்தலாம். வணிகப் பயன்பாடுகளான சம்பளப் பட்டியல் தயாரித்தல், நிதிநிலைக் கணக்கு, பொருள் நிலை யாளுகை முதலியவற்றில் தயார் செய்யப்படும் அறிக்கைகள், தேவையான படிவத்திலிருத்தல் மிகவும் இன்றியமையாததாகும். எனவே, இவை போன்ற பணிகளில் பேஸிக் மொழியைப் பயன் படுத்தும்பொழுது அறிக்கைகளைத் தயார் செய்ய TAB பெரு மளவு உதவுகிறது.

எ.கா : `10 PRINT TAB(10);NUM;TAB(50);RUPEE`

3. LPRINT (List of Variables)

இது BBASIC இல் பயன்படும் ஆணையாகும். இது PRINT ஆணை செய்யும் பணியைச் செய்கிறது. ஆனால், விடை காட்சித் திரைக்குப்பதிலாக அச்சுப்பொறியில் [PRINTER] அச்சாகிறது.

எ. கா :

```
40 A=50:B=60:C=70

50 LPRINT A,B,C

RUN
```

அச்சுப்பொறியில் விடை

```
50 60 70
```

4. LPRINTER

இது ABASIC-இல் உள்ள ஆணையாகும் இந்த ஆணை மூலம் கணிக்கப்படும் விடைகளும் செய்திகளும் காட்சித் திரைக்குப் பதிலாக, அச்சு பொறியில் [Printer] கிடைக்கின்றன.

5. CONSOLE

இது ABASIC இல் உள்ள ஆணையாகும். விடையைக் காட்சித் திரையில் பெறுவதற்கு இது பயன்படுகிறது. ABASIC இல் LPRINTER-ஐப் பயன்படுத்திய பிறகு, விடையைக் காட்சித் திரையில் வெளிப்படுத்த வேண்டுமாயின், CONSOLE பயன்படுத்தப் படுகிறது.

9.1.3. விடுவரல் எண் படிவ ஆணைகள் (*NUMERIC FORMAT STATEMENTS*

PRINT USING (String Exp) ; (List of Variables)

இந்த ஆணையைப் பயன்படுத்துவதன் மூலம் எண் விடைகளை [Numeric] நாம் விரும்பிய படிவத்தில் பெறமுடியும். String Exp என்பது ' # ' என்ற குறியைக் கொண்டு உருவாக்கப்படும். ஒவ்வொரு ' # ' குறியும் எண்ணின் இலக்கத்தை வெளிப்படுத்த உதவும். நமது விடையின் தன்மை, மற்றும் மொத்தம் எவ்வளவு இலக்கங்கள் இருக்கலாம் என்பனவற்றைக் கருத்தில் கொண்டு String Exp உருவாக்கப்படுகின்றன. தசமப் புள்ளியைத் தேவையான இடத்தில் இடுவதற்கு ஏற்ப இந்த String Exp அமைதல் வேண்டும். இதில் PRINT -ஆணைக்குப் பதிலாக LPRINT ஆணை இருக்குமாயின் விடை அச்சுப்பொறியில் [Printer] அச்சாகிறது.

எ. கா :

```
10 A=350.352:B=25

20 PRINT USING "####.##";A;"##";B

RUN
```

விடை

```
Ø350.35   25
```

வெளிப்படுத்தப்படும் விடையின் தொடக்கத்தில் -, +, $, * போன்ற குறிகள் அமைய வேண்டுமாயின் கீழ்க்கண்ட எடுத்துக் காட்டிலுள்ளவாறு [String Exp] -ஐ உருவாக்கல் வேண்டும்.

எ. கா :

```
10 PRINT USING "###.##";-15.27,25.46

20 PRINT USING "$$##.##";576.49

30 PRINT USING "**##.##";765.37

40 STOP

RUN
 -15.27        25.46
$576.49
*765.37
```

9.1.4. குறிக்கோவைப் படிவ ஆணைகள்

(STRING FORMAT STATEMENTS)

குறிக்கோவைப் படிவ ஆணைகளில் இரண்டு வகைகள் உள்ளன. அவைகளாவன :

1. Fixed Length Field : இதில் தேவையான வெற்றிடங்களால் [Blanks] -பிரிக்கப்பட்ட இரண்டு "\" -கள் பயன்படுகின்றன. இதைக் கீழ்க்கண்ட எடுத்துக்காட்டு மூலம் விளக்கலாம்.

எ. கா :

```
10 FOR$ = "THE VALUE IS"

20 VAL$ = "TOTAL AMOUNT"

30 PRINT USING "\bbbbb...\";FOR$;VAL$

RUN
```

விடுவரல் : THE VALUE IS TOTAL AMOUNT

மேற்கண்ட வகைப் படிவம் [n+2] அளவுள்ள குறிகளை வெளிப்படுத்தும். n என்பது, அவ்வளவு எண்ணிக்கையுள்ள வெற்றிடத்தைக் குறிக்கும். வெளிப்படுத்தப்படவேண்டிய குறிக் கோவையின் நீளம் குறிப்பிடப்பட்டுள்ள படிவத்தை விட, கூடுதலாக இருக்குமாயின் படிவத்தில் [Format] கொடுக்கப் பட்டுள்ள எண்ணிக்கை அளவுள்ள குறிகள் மட்டுமே வெளிப் படுத்தப்படும்; மீதமுள்ள குறிகள் வெளிப்படுத்தப்படா.

2. Variable Length Field : இந்த வகை இடுவரல் ஆணையில் ''&'' என்ற குறியைப் படிவத்திற்காகப் பயன் படுத்துவதன் மூலம் தகுந்த படிவத்தில் குறிக்கோவை [String] குறிப்பிடப்படுகிறது.

எ.கா. :

```
10 NAME$="SYSTEM":DES$="IS A MULTI-USER"

20 PRINT USING "&";NAME$;DES$

RUN
```

விடுவரல் : SYSTEM IS A MULTI-USER

9.2. கோவை ஆணைகள் (*DISK BASED FILE STATEMENTS*)

நாம் இதுவரை சாதாரண இடுவரல். விடுவரல் ஆணை களைப் பற்றியும் அவற்றைப் பயன்படுத்திச் செய்திக் கூறுகளைக் கணிப்பொறிக்கு இடுவரல் செய்தல் மற்றும் விடைகளை எங்ஙனம் காட்சித்திரை அல்லது அச்சுப் பொறியில் பெறுதல் ஆகிய விவரங்களைப் பற்றியும் ஆராய்ந்தோம். பெரும்பாலான வணிக முறைப் பயன்பாடுகளில் ஏராளமான செய்திக் கூறுகளைக் கோவைகளின் மூலம் துணை நினைவகத்தில் பதிவு செய்தலும் பின்னர் அவற்றைப் பயன்படுத்துதலும் மிகமிக இன்றியமையா தனவாகின்றன.

கணிப்பொறியில் பேஸிக் வழியமைப்புச் [Program] சூழ்நிலையில் [Mode] பணிபுரியும்பொழுது செய்திக் கூறுகளை [Data] இடுவரல் செய்யவும், பின்னர் அவற்றைத் துணை நினைவகங்களில் குறிப்பிட்ட கோவைகளின் பெயரில் பதிவு

செய்துவைக்கவும் வேண்டியுள்ளது. இவ்வாறு பதிவு செய்யப் பட்ட செய்திக்கூறுகள் [Data] ஆய்வு செய்வதற்குத் தேவைப் படும்பொழுது எடுத்துக்கொள்ளப்படுகின்றன. ABASIC, BBASIC ஆகிய இரண்டு மொழிகளிலும் வழியமைப்பு செய்யும்பொழுது பயன்படுத்தக் கூடிய கோவை ஆணைகளைப் பற்றி இங்குக் காண்போம்.

9.2.1. BBASIC

இதில் செய்திக் கூறுகளைப் பதிவு செய்து இயக்கக் கையாளப்படும் கோவைகளை இரண்டு வகைகளாகப் பிரிக்கலாம்.

(அ) முறையிலா அணுகுமுறைக் கோவை
(Random access file)

(ஆ) வரிசைக்கிரம அணுகுமுறைக் கோவை
(Sequential access file)

(a) முறையிலா அணுகுமுறைக் கோவை (RANDOM ACCESS FILE)

(1) OPEN "Mode", (Channel No), (File name), (Record Length)

மேற்கண்ட ஆணை ஒரு கோவையிலிருந்து செய்திக்கூறு களைப் [Data] படித்தறிவதற்கு முன்பு, அந்தக் கோவையை அணுகுவதற்கு ஏற்ற முறையைத் தெரிவிக்கிறது. கோவையின் 'Mode' கீழ்க்கண்டவாறு அமையலாம் :

"O" -ஆக இருக்கும்பொழுது, செய்திக்கூறுகளை [Data] வரிசைக் கிரம அணுகுமுறைக் கோவையில் [Sequential Access File] எழுத [Write] மட்டுமே இயலும். இந்நிலையில் செய்திக் கூறுகளைப் படித்தறிதல் இயலாது.

" I " -ஆக இருக்கும்பொழுது வரிசைக்கிரம அணுகு முறைக் கோவையில்[Sequential Access File] செய்திக்கூறுகளை [Data] படித்தறிய மட்டுமே இயலும்; எழுதுதல் இயலாது.

"R" – இது முறையிலா அணுகுமுறைக் கோவை [Random Access File] என்பதைக் குறிப்பதாகும். செய்திக் கூறுகளை [Data] முறையிலா அணுகுமுறைக் கோவையில் [Random Access File] இடுவரல் மற்றும் விடுவரல் செய்யப் பயன்படுகிறது.

Record Length : பேஸிக் மொழியில் முறையிலா அணுகு முறைக் கோவை பயன்படுத்தப்படும்பொழுது, அதிலுள்ள ஒரு பதிப்பின் அளவு எவ்வளவு குறிகள் கொண்டது எனக் குறிப்பிடப் படல் வேண்டும். அவ்வாறு குறிப்பிடப்படாவிடில் ஒரு பதிப்பின் அளவு 128 bytes அதாவது ஒரு sector ஆகக் கொள்ளப்படும்.

2. CLOSE Channel No.

வழியமைப்பில் செயற்பாடுகள் முடிந்தபிறகு செய்திக் கூறுக் கோவையைப் [Data file] பாதுகாத்தற்கு [Close] இந்த ஆணை பயன்படுகிறது. கோவையில் செய்திக்கூறுகளைப் படித்தறிந்த அல்லது எழுதிய பிறகு CLOSE ஆணை பயன் படுத்தப்படுகிறது.

Channel No : இது ஒரு கோவையின் சுட்டு எண்ணைக் [Channel Number] குறிப்பதாகும். வழியமைப்பில் பயன்படுத்தப் படுமுன் OPEN கட்டளை மூலமாக ஒரு செய்திக் கூறுக் கோவை, பேஸிக் மொழிக்கு அறிமுகப்படுத்தப்படுகிறது. எனவே இவ்வாறு திறவு செய்யப்பட்ட கோவையை வழியமைப்பின் மற்ற பகுதி களில் குறிப்பிடவேண்டுமெனில், அந்தக் கோவைக்குரிய சுட்டு எண்ணைக் [Channel Number] குறிப்பிட்டால் போதுமான தாகும். BBASIC-இல் மூன்று கோவைகளை ஒரே நேரத்தில் செயற்படச் [active] செய்யலாம். பேஸிக் மொழி வழி யமைப்பில் தனிப்பட்ட வழியில் [Special Switches] பதினைந்து கோவைகள் வரை பயன்படுத்த இயலும்.

3. FIELD # (Channel No.), (Field width) AS (String Variable)

முறையிலா அணுகுமுறைக்கோவையைப் பேஸிக் வழி யமைப்பு மொழியில் பயன்படுத்தும்பொழுது நினைவகத்தில் Random Buffer -க்குத் தேவையான இடத்தை ஒதுக்குவதற்கு இந்த ஆணை பயன்படுகிறது.

Channel No. : கோவையின் சுட்டு எண்ணைக் குறிப்பதாகும்.

Field width : இது ஒரு மாறிக்குத் தேவையான பதிவக அளவை ஒதுக்கீடு செய்ய உதவுகிறது. முறையிலா அணுகுமுறைக் கோவையின் [Random Access File] பதிவிலிருந்து செய்திக் கூறுகளைப்பெற இது உதவுகிறது. மேலும் நினைவகத்தில் முறையிலா அணுகுமுறைக் கோவையின் [Random Access File] Buffer -க்குப் போதிய அளவு நினைவகத்தை ஒதுக்குகிறது.

குறிப்பு :

இதில் பயன்படுத்தப்படும் மாறியை [Variable] வழியமைப்பின் மற்ற இடங்களில் பயன்படுத்தவோ, இடையூறு செய்யவோ கூடாது.

எ. கா :

```
10 FIELD #1,25 AS A$, 30 AS B$

20 INPUT C$,D$

30 LSET A$=C$:LSET B$=D$
```

4. LSET (String Variable) = (String Expression)
   RSET (String Variable) = (String Expression)

LSET என்பது குறிக்கோவைச் செய்தியை [String Data] முறையிலா அணுகுமுறைப் பதிவகத்திற்கு மாற்றுவதுடன் செய்திக்கூறை மாறியின் இடப்பக்கமாக அமைக்கிறது. RSET என்ற ஆணை முறையிலா அணுகுமுறைப் பதிவகத்திற்குச் [Random Buffer] செய்திக்கூறை மாற்றி வலப்பக்கமாக அமைக்கிறது. விடுவரல் செய்யப்படும் மாறிகளின் மதிப்பைக் குறிக்கோவையாக மாற்றிப் பதிவு செய்யவும் இந்த ஆணை பயன்படுகிறது.

எ. கா :

```
10 LSET NAM$=YES$

20 RSET N$ = MKD$(ACCOUNT#)
```

5. GET # (Channel No), (Record No)

முறையிலா அணுகு முறைக் கோவையிலிருந்து [Random Access File] குறிப்பிட்ட பதிப்பைப [Record] படித்தறிவதற்கு இது உதவுகிறது.

எ.கா : 1000 GET#2, 100

சுட்டு எண் [Channel Number] 2ஐ உடைய முறையிலா அணுகுமுறைக் கோவையில் 100 ஆவது பதிப்பைப் படித்தறிகிறது

6 PUT # (Channel No) , (Record No)

இந்த ஆணை மூலம் முறையிலா அணுகுமுறைக் கோவையில் [Random Access File] ஒரு குறிப்பிட்ட பதிப்பு [Record] எண்ணில் நினைவகத்தில் உள்ள பதிப்பைப் [Record] பதிவு செய்ய முடிகிறது.

எ. கா : 1100 PUT#2, 100

2 ஐச் சுட்டு எணணாக [Channel Number] உடைய கோவையில் 100ஆவது பதிப்பில் செய்திக்கூறு எழுதப்பெறுகிறது.

(b) வரிசைக் கிரம அணுகுமுறைக்கோவை (SEQUENTIAL ACCESS FILE)

1 INPUT# (Channel No), (Variable List)

இதன் மூலம் வரிசைககிரம அணுகுமுறைக் [Sequential Access File] கோவையிலிருநது செய்திகளைப் படித்துணர முடிகிறது வரிசைககிரம அணுகுமுறைக கோவையில செய்திக கூறுகள் வரிசையாக எழுதப்பெறுகின்றன. பின்னர் இந்த ஆணையைப பயன்படுத்தும் பொழுது அதே வரிசையில் செய்திக் கூறுகள் படித்தறியப்படுகின்றன.

எ கா . 1000 INPUT#1, N1, N2

2. LINE INPUT#,(File Number), (String Variable)

DISK-இல் பதிவு செய்யப்பட்டுள்ள செய்திகளை வரிசைக கிரம அணுகுமுறைக கோவையிலிருந்து குறிக்கோவை மாறிக்கு [String Variable], ஒரே நேரத்தில 255 குறிகள [Characters] அதாவது ஒரு வரி முழுவதுமுள்ள செய்திககூறை இந்த ஆணையைப் பயன்படுத்திப் படித்தறியலாம.

எ. கா · `200 LINE INPUT#1, TAM$`

3. PRINT # (Channel No.), (Variable List)

இந்த ஆணை, துணை நினைவகததிலுள்ள வரிசைககிரம அணுகுமுறைக் கோவையில [Sequential Access File] செய்திகளைப் பதிவு செய்ய உதவுகிறது மாறிகள அரைபபுள்ளியால் (,) பிரிக்கபபட்டிருத்தல் வேணடும. இநத ஆணையைப் பயன்படுத்துவதன் மூலம காட்சிததிரையில் வெளிப்படுததப்படும் படிவத்திலுள்ள செய்திககூறுப பதிப்புகள வரிசைககிரம அணுகுமுறைக் கோவையில் பதிவு செய்யப்படுகினறன

4 WRITE # (Channel No.), (Variable List)

இந்த ஆணை மேற்குறிபபிட்ட PRINT -ஆணை செயத பணியையே செய்கிறது. ஆனால், தானாகவே இரண்டு மாறிகளுக்கு இடையில் ஒரு காற புள்ளியை (,) இட்டுக் கொள்கிறது. இந்த ஆணையைப பயன்படுதத வேண்டுமாயின், Channel No -ஐ கோவையின் சுட்டு எண்ணாகவுடைய கோவை 'O' mode -இல திறவு செய்யப்பட்டிருத்தல் இன்றியமையாததாகும.

எ கா .

```
10 LET N$="TAMIL"
20 LET M$="UNIVERSITY"
30 WRITE#1, N$, M$
```

5. KILL (File Name)

துணை நினைவகத்திலுள்ள [Disk] தேவையற்ற வழியமைப்பு அல்லது செய்திக்கூறுக் கோவைகளை [Program or Data Files] நீக்குவதற்கு [Erase] இந்த ஆணை பயன்படுகிறது. கோவையானது வரிசைக்கிரம அணுகுமுறை அல்லது முறையிலா அணுகுமுறைக் கோவையாக இருக்கலாம்.

எ. கா :

```
200 KILL "PAYROLL.BAS"
```

6. EOF (Channel No.)

இந்த ஆணை, செய்திக்கூறுக் கோவைகளிலிருந்து (Data Files) செய்தியைப் படித்தறியப் பயன்படுகிறது. அவ்வாறு செய்தியைப் படிக்கும்பொழுது கோவையில் செய்தி முடிவுற்றதா இல்லையா என்பதை, இந்த ஆணை உணர்த்துகிறது. செய்திக் கூறுகளைப் படிக்கும்பொழுது கோவையில் செய்தி முடிவுறவில்லையெனில், EOF- இன் மதிப்பு-1 ஆகவும், கோவையில் செய்தி முடிவு அடைந்தவுடன் 0 ஆகவும் இருக்கும்.

மேற்கண்ட இடுவரல், விடுவரல் ஆணைகளைக் கொண்டு அமைக்கப்பட்ட ஒரு வழியமைப்பு [Program] கீழே கொடுக்கப்பட்டுள்ளது.

எ. கா :

```
10 OPEN "O",2,"PAYROLL.DAT"
20 LINE INPUT "COMPONENTS DETAILS";C$
30 PRINT#2,C$
40 CLOSE#2
50 OPEN "I",2,"PAYROLL.DAT"
60 LINE INPUT#2,C$
70 PRINT C$
```

```
80 CLOSE#2

90 END
```

9.2.2. ABASIC

ABASIC -இல் உள்ள கோவை ஆணைகள் BBASIC - இல் பயன்படுத்தப்படும் இடுவரல், விடுவரல் ஆணைகளிலிருந்து சிறிது மாறுபட்டுள்ளன.

1. OPEN (File Name 1), (Exp), RECL (Exp) AS (Exp 1) BUFF (Exp 2) RECS (Exp 3)

இந்த ஆணை, துணை நினைவகமாகிய Disk -இல் உள்ள செய்திக்கூறுக் கோவையிலிருந்து [Data File] செய்தியைப் படித்தறிய உதவுகிறது. துணை நினைவகத்தில் முன்பே உருவாக்கப்பட்ட கோவையை அணுகுவதற்கு இந்த ஆணை பயன்படுகிறது.

[File Name 1]	:	கோவையின் பெயர்
Exp 1	:	கோவை எண் [Channel Number]
RECS [Exp 3]	:	பதிப்பின் அளவு [Record Length in Bytes]
BUFF [Exp 2]	:	ஒரே சமயத்தில் செய்திகளைக் கோவையிலிருந்து படித்தறியவோ ஒரே சமயத்தில் நினைவகத்தில் வைத்திருக்கவோ தேவையான பதிவகத் [Buffer] திறனைக் குறிப்பதாகும்.
RECL	:	மாறாத பதிப்பின் அளவைக் குறிப்பதாகும்.

எ. கா :

```
100 OPEN "A:PAYROLL.DAT" AS 5 BUFF 3 RECS 96 RECL
```

2. CREATE (File Name 1) (Exp), RECL(Exp) AS (Exp 1) BUFF (Exp2) RECS (Exp 3)

இந்த ஆணையும் மேற்குறிப்பிடப்பட்ட ஆணை செய்த பணியையே செய்கிறது. ஆனால், ஒவ்வொரு தடவையும் இந்த ஆணை செயற்படும் பொழுது, புதிய கோவையை உருவாக்கு கிறது.

எ.கா : 100 CREATE "ACCOUNT.DAT" AS 10 BUFF 4 RECS 128

3. FILE (Exp 1), (Exp 2)

இந்த ஆணை, குறிப்பிடப்பட்ட கோவையின் பெயர் [File Name] முன்பே துணை நினைவகத்தில் இருப்பின், அதைத் திறவு [Open] செய்கிறது. இல்லையெனில், [Exp 1] என்ற பெயரில் புதிய கோவையை உருவாக்கி [Exp 2], என்று பதிப்பின் அளவைக் [Record Length] குறிக்கிறது.

எ.கா : 10 FILE NAM$(REC.A%)

4. CLOSE (Exp 1)

இது [Exp 1] என்ற எண்ணைச் சுட்டு எண்ணாக உடைய கோவையைப் பாதுகாக்க [Close] உதவுகிறது. இதன் பிறகு கோவையில் செய்திக்கூறைப் படிக்கவோ, எழுதவோ நேரின், திரும்பவும் கோவையைத் திறவுசெய்ய வேண்டும். இந்த ஆணையைச் செயற்படுத்தியவுடன் கோவை எண்ணும், கோவைக்கென ஒதுக்கப்பட்ட நினைவகமும் [Buffer] விடு விக்கப்படுகின்றன.

எ.கா : 10 CLOSE#XYZ%

5. DELETE (Exp 1)

இதன் மூலம் [Exp 1] என்ற சுட்டு எண்ணுடைய [Channel No] கோவையை, கோவைப்பட்டியலிலிருந்து [Directory] அகற்ற முடியும். Exp 1 -இன் மதிப்பு 0 -விலிருந்து 20 -க்குள்

இருக்க வேண்டும். குறிப்பிட்ட கோவை [வழியமைப்பு அல்லது செய்திக்கூறுக் கோவை] தேவையற்றதெனில், இந்த ஆணையைக் கொண்டு துணை நினைவகத்திலிருந்து அவற்றை அகற்ற முடியும்.

எ.கா: 10 DELETE 5

6. IF END # (Exp 1) THEN (Statement 1)

இந்த வழியமைப்பு ஆணை, கோவையின் முடிவைத் தெரிந்த பிறகு ஒரு குறிப்பிட்ட செயற்பாட்டினைச் செயற்படுத்தப் பயன்படுகிறது.

எ.கா: 50 IF END#10 THEN PRINT"EOF REACHED"

7. READ # (Exp 1), (Var 1, Var 2)

இந்த ஆணையைப் பயன்படுத்தி நினைவகத்தில் கோவையிலுள்ள செய்திக்கூறுகளை [Data] ஒவ்வொரு தொகுதியாகப் படித்தறிந்து, மாறிகளுக்குச் [Variables] சமன்படுத்த முடிகிறது. இதைக் கீழ்க்கண்டவாறு வேறு வகையாகவும் எழுதலாம்.

READ # (Exp 1), (Exp 2) ; (Var 1), (Var 2)

இந்த ஆணை மூலம் குறிப்பிட்ட கோவையிலிருந்து குறிப்பிட்ட பதிப்பைப் படித்துணர முடிகிறது. Exp 2 பதிப்பு எண்ணைக் குறிப்பதாகும்.

எ.கா :

```
10 READ # 10;NAM$,ADS$,CITY$,NO$

20 READ # (exp1), (exp2);LINE (var1)
```

இந்த முறையில் ஒரு கோவையில் உள்ள செய்திகளின் ஒவ்வொரு வரியின் முடிவிலும் CR [Carriage Return] மற்றும் LF [Line Feed] கொடுக்கப்பட்டிருக்கும். இதன் மூலமாக

முழுவரிச் செய்தியைக் கணிப்பொறி, துணை நினைவகத்திலுள்ள கோவையிலிருந்து படித்துணர்கிறது.

எ.கா :
```
100 READ#10;LINE ADDRESS$

200 READ#10,REC%;LINE DTL$

300 READ#XYZ;LINE C$
```

8 (i) PRINT # (Exp 1) ; (Var 1, Var 2)
(ii) PRINT # (Exp 1), (Exp 2); (Var 1, Var 2)

செய்திக்கூறுகளைக் கோவையில் பதிவு செய்வதற்கு இந்த ஆணைகள் உதவுகின்றன. முதலில் உள்ள ஆணை, வரிசைக் கிரம அணுகுமுறைக் கோவையில் செய்திக்கூறுகளை எழுதவும் [Sequential access file], இரண்டாவதிலுள்ள ஆணை, முறையிலா அணுகுமுறைக் கோவையில் [Random Access File] செய்திக்கூறுகளைப் பதிவு செய்யவும் பயன்படுகின்றன.

Exp 1 : கோவையின் சுட்டு எண் (Channel No.)

Var 1, Var 2 : மாறிகள் (Variables)

Exp 2 : பதிப்பு எண் (Record No.)

எ. கா :
```
100 PRINT#4, 55;NAME$,CITY$

200 PRINT#FILE%;N$,C$
```

9. (i) PRINT USING (Exp 1); # (Exp 2); (Var 1, Var 2)
(ii) PRINT USING (Exp 1); # (Exp 2), (Exp 3) ; (Var 1, Var 2)

BBASIC-இல் பயன்படுத்தப்பட்ட PRINT USING என்ற ஆணை செய்த பணியையே இந்த ஆணையும் செய்கிறது. ஆனால் இதில் Exp 1 என்ற படிவத்தில் [Format] செய்திக்கூறுகள் Exp 2 என்ற எண்ணுள்ள கோவையில் பதிவு செய்யப்படுகின்றன.

இரண்டாவது ஆணை [ii] முறையிலா அணுகுமுறைக் கோவையில் [Random Access File] செய்திகளை இடுவரல் [Input] செய்யப் பயன்படுகிறது. இந்த முறையில் தெரிவு செய்யப்பட்ட பதிப்பு எண்ணில் பதிப்பை எழுத இயலும்.

9.3. வழியமைப்புக் கூறுகள் (*PROGRAMMING FUNCTIONS*)

பேஸிக் [Basic] மொழியில் வழியமைப்புச் செய்வதற்குத் தேவையான கணக்கியல் ஆணைகள் [Arithmetic Statements], இடுவரல்/விடுவரல் ஆணைகள், கட்டுப்பாட்டு ஆணைகள் [Control Statements] முதலியவற்றைப் பற்றி இதுவரை ஆராய்ந்தோம். வழியமைப்புக் கூறுகள் [Functions] வழியமைப்பு ஆணைகளிலிருந்து [Instructions] சற்று மாறுபட்டுள்ளன. வழியமைப்பில் அவை பயன்படுத்தப்படும் பொழுது, ஒரு குறிப்பிட்ட பணியைச் செய்யும் விதத்தில் அவை உருவாக்கப் பட்டுள்ளன. BBASIC -இல் பயன்படும் கூறுகள் [Functions] அனைத்தும் ABASIC மொழியிலும் செயல்படுகின்றன. இவையன்றி ABASIC-இல் இன்னும் கூடுதலாகச் சில கூறுகள்[Functions] அமைந்துள்ளன. பொதுவாகக் கூறுகளைக் கீழ்க்கண்டவாறு மூன்று பிரிவுகளாகப் பிரிக்கலாம்.

(I) க்ணக்கியற் கூறுகள் (Arithmetic Functions)

(II) குறிக்கோவைக் கூறுகள் (String Functions)

(III) தனிக்கூறுகள் (Special Functions)

இவைகளைப் பற்றி கீழ்க்கண்டவாறு ஆராயலாம்.

9.3.1. கணக்கியற் கூறுகள் (*ARITHMETIC FUNCTIONS*)

1. ABS(A) : இக்கூறு A-யின் சரியான எண் மதிப்பை அளிக்கிறது [Absolute Value].

எ.கா :
```
150 Y=ABS(-25.31)
RUN
```

விடை 25.31

2. ATN(A) : A -யின் மதிப்பு 'Radians' -இல் இருக்கும்பொழுது, இந்தக்கூறு அதன் 'arctangent' -இன் மதிப்பைக் காண உதவுகிறது.

எ. கா :

```
10 LET B=3
20 Y=ATN(B)
30 PRINT Y,B
```

3. COS(A) : A -யின் மதிப்பு 'Radians' -இல் இருக்கும் பொழுது COSINE -னின் மதிப்பைக் கண்டறிகிறது. COS[A] -யின் மதிப்பு ஒற்றைத் துல்லியமாகக் [Single Precision] கிடைக்கிறது.

எ. கா :

```
20 LET NUM=COS(A)
```

4. CSNG(A) : A-யின்மதிப்பை ஒற்றைத் துல்லியமாக [Single Precision] மாற்றுகிறது.

எ. கா :

```
10 N# = 726.3678
20 PRINT N#;CSNG(N#)
RUN
726.3678   726.367
```

5. CDBL(A) : A-யின் மதிப்பை இரட்டைத் துல்லியமாக [Double Precision] மாற்றுகிறது.

6. EXP(A) : இது e^A -யின் [Exponentiation] மதிப்பைக் கண்டறிய உதவுகிறது. A- யின் மதிப்பு 87க்குமிகாமலிருக்க வேண்டும்.

எ. கா :
```
10 A=5
20 PRINT EXP(A-1)
RUN
54.5982
```

7. FIX(A) : இந்தக்கூறு A- யின் மதிப்பை முழு எண்ணாகத் தசமப்புள்ளி மதிப்பைத் தவிர்த்து [Truncate Decimal value] விடையளிக்கிறது.

எ. கா :
```
20 LET NUM=107.821
30 PRINT FIX(NUM)
RUN
107
```

8. FRE : இது வழியமைப்பு மற்றும் செய்திக் கூறுகள் [Data] போக மீதமுள்ள [Free Memory space] நினைவக அளவைத் தெரிவிக்கிறது. இதற்கு Garbage Collection என்ற பெயர் உண்டு.

எ.கா :
```
10 PRINT FRE ("")
RUN
27,632 Bytes
```

9. FLOAT(X%): X% -இன் மதிப்பை உண்மை எண்ணாக [Real Number] மாற்று

கிறது. இது ABASIC -இல் உள்ள கூறாகும்.

10. INT(A) A -யின் மதிப்புக்கு அருகிலுள்ள சமமான அல்லது குறைவான முழு எண்ணைக் கண்டறியப் பயன்படுகிறது. i.e., முழு எண் <= A

எ.கா :
```
10 PRINT INT(79.67)
RUN
79
```

11. LOG(A) : A -யின் மதிப்பு 0 -ஐ விடப் பெரியதாகவும் உண்மை எண்ணாகவும் [Real Number] இருக்கும் பொழுது, அதன் Natural Logarithm கண்டறியப் படுகிறது.

12. RND(A) : இந்தக்கூறு முறையிலா முறையில் எண்களை [Random Numbers] உருவாக்குகிறது. A -யின் மதிப்பு 0 -விலிருந்து 1 -க்குள் இருத்தல் வேண்டும்.

13. SIN(A) : A-யின் மதிப்பு 'Radian' இல் இருக்கும் பொழுது SINE -னின் மதிப்பைக் கண்டறிய இந்தக்கூறு உதவுகிறது.

14. SQR(A) : ஒரு முழு [+ve] எண்ணின் வர்க்க மூலத்தைக் [Square Root] கண்டறிய இந்தக்கூறு பயன்படுகிறது.

15. TAN(A) : இந்தக் கூறைப் பயன்படுத்தி A -யின் மதிப்பு 'Radian' -இல் இருக்கும் பொழுது அதன் TANGENT -இன் மதிப்பைக் கண்டறிய இயலும்.

9.3.2. குறிக்கோவைக் கூறுகள் (STRING FUNCTIONS)

1. ASC(A $) : குறிக்கோவை [String] A$ இல் உள்ள முதற்குறியின் ASCII -எண்ணைக் கண்டறியப் பயன்படுகிறது.

எ.கா :

```
20 A$="ACCOUNT"

30 PRINT "THE ASCII VALUE OF";A$;" IS:",ASC(A$)

RUN

THE ASCII VALUE OF ACCOUNT IS:65
```

2. CHR $ (Exp): Exp -னில் குறிப்பிட்டுள்ள ASCII மதிப்பைக் கொண்டு குறியைக் [Character] கண்டறிய இந்தக்கூறு உதவுகிறது. Exp -னில் மதிப்பு முழு எண்ணாக இருத்தல் இன்றியமையாததாகும்.

எ.கா :

```
10 PRINT CHR$(65)

A
```

3. HEX $ (Exp) : தசம மதிப்பான Exp ஐ Hexadecimal [Base of 16] க்குச் சமமான குறிக்கோவையாக மாற்றுகிறது.

எ.கா :

```
10 A$=HEX$(32)

20 PRINT A$

RUN

20
```

4. **INKEY$** : இந்தக்கூறு இயக்கு நிலையகத்து விருந்து [Console] ஒரு குறியை அல்லது Null குறியை இடுவரலாகப் பெறுகிறது. ஆனால், அது பிரதி பலிக்கப்படுவதில்லை [Not echoed].

5. INPUT $ (length, #F)

INPUT$ என்ற கூறு குறிப்பிட்ட [length] எண்ணிக்கையுள்ள குறித் தொடர்ச்சியை [Character String] இயக்கு நிலையகத்திலிருந்தோ [Console], [#F] என்ற சுட்டு எண்ணைக் கொண்ட கோவையிலிருந்தோ பெற்றுத் தருகிறது. இயக்குநிலையகமாயின் குறிகள் பிரதிபலிக்கப்படா.

எ. கா : `30 A$=INPUT$(6, 2)`

6. INSTR (Exp, String 1, String 2)

இந்தக்கூறு செயற்படும்பொழுது String 2 என்ற குறிக் கோவையை String 1 என்றகுறிக்கோவையிலிருந்து தேடி, அவ்வாறு இருக்குமாயின், அது அமைந்துள்ள இடத்தை தெரிவிக்கிறது. முதல் குறிக்கோவையின் எந்தக் குறியிலிருந்து இச்செயல் தொடங்க வேண்டுமென்பதை Exp உணர்த்துகிறது.

எ. கா :

```
10 CTR$="TO TEST THE PROGRAM"
20 A$="P"
30 PRINT "THE POSITION OF P IS"
```

```
40 PRINT INSTR(CTR$,A$)
50 END
RUN
13
```

7. LEFT $ (String, Length)

இதன் மூலம் 'Length' எண்ணிக்கையுள்ள குறிகளைக் கொண்ட குறிக்கோவையை 'String' -இன் இடப்பக்கத்திலிருந்து பிரித்தெடுக்கலாம். 'Length' -இன் மதிப்பு 255-க்கு மேலிருத்தல் கூடாது.

எ. கா:

```
10 A$="PAYROLL PROCESSING"
20 B$=LEFT$(A$,7)
30 PRINT B$
RUN
PAYROLL
```

8. LEN (String)

குறிக்கோவையிலுள்ள மொத்தக் குறிகளின் எண்ணிக்கையைக் கணக்கிடுகிறது. இதில் வெற்றிடங்களும் உள்ளடங்கும்.

எ. கா:

```
10 A$="TAMIL UNIVERSITY"
20 PRINT LEN(A$)
RUN
16
```

9. MID$ (String, Start, Length)

இந்த ஆணை மூலம் ஒரு குறிக்கோவையின் இடையில் தொடங்கி Start என்ற குறி எண்ணிலிருந்து 'Length' எண்ணிக்கையுள்ள குறிகளைக் கொண்ட குறிக்கோவையை உருவாக்க இயலும். இதில் Start, Length இவற்றின் மதிப்பு 1 லிருந்து 255 க்குள் இருத்தல் மிகவும் அவசியமாகும்.

எ. கா :
```
10 A$="TAMIL UNIVERSITY LIBRARY"
20 X$=MID$ (A$,7,10)
30 PRINT X$
RUN
UNIVERSITY
```

10. MATCH (X$ Y$ Z%)

இது ABASIC -இல் உள்ள கூறாகும். BBASIC -இல் உள்ள INSTR என்ற கூறு செய்யும் பணியிலிருந்து இதன் செயற்பாடு சற்று மாறுபட்டுள்ளது. இதில் X$ என்ற குறிக்கோவையின் இருப்பிடத்தை Y$ -இல் Z% என்ற குறிநிலையில் தொடங்கிக் கண்டறிவிக்கும்.

a. " ## " : இக்குறி எல்லா எண்களுக்கும் மற்றும் " ! " குறி, கீழ்நிலை, மேல்நிலைக் குறிகளுக்கும் [Lower or Upper Case Characters] பொருந்தும்.

b. " ? " : எல்லாவகைக் குறிகளுக்கும் பொருந்தும்.

c. " " : ஒரு குறிக்கோவையில் உள்ள தனிப் பொருளற்ற [Special Meaning] குறிக்குப் பொருந்தும்.

எ.கா :
```
MATCH ("####","DATE: 23-07-1983",1)
```

7 - (அதாவது " #### " பொருந்தும் வகையில் 23, என்ற எண் ஏழாவது குறிநிலையில் அமைந்துள்ளது).

11. OCT $ (Exp)

குறிப்பிட்ட எண்ணை [Exp] Octal [Base of 8] ஆக மாற்றப் பயன்படுகிறது.

எ. கா :

```
10 LET A$=OCT$(48)
20 PRINT A$
RUN
60
```

12. RIGHT $ (X$, Y)

X$ - என்ற குறிக்கோவையின் வலப்பக்கமிருந்து Y எண்ணிக்கையுள்ள குறிகளைக் கொண்ட புதிய குறிக்கோவையைப் பிரிக்க இது பயன்படுகிறது.

எ. கா :

```
10 LET A$="TAMIL NADU"
20 LET B$=RIGHT$(A$,4)
30 PRINT B$
RUN
NADU
```

13. SPACE $ (Exp)

Exp எண்ணிக்கையுள்ள வெற்றிடக்குறிகளை (Blank Spaces) உருவாக்குகிறது.

எ. கா :
```
10 LET C$="FIVE BLANKS"
20 LET D$=SPACE$(5)
30 PRINT D$;C$
RUN
bbbbbFIVE BLANKS
```

14. STR $ (Exp)

ஒரு குறிப்பிட்ட எண்ணைக் (Exp) குறிக்கோவையாக (String) மாற்ற உதவுகிறது.

எ. கா :

```
10 LET A=175.32
20 LET B$=STR$(A)
30 PRINT B$
RUN
175.32
```

15. STRING $ (Length 1, String 1) or
STRING $ (Length1, A)

இந்தக் கூறைப் பயன்படுத்தி length 1 எண்ணிக்கையுள்ள String1 இன் முதல் குறியைக் கொண்ட அல்லது ASCII—A யின் மதிப்பைக் (Characters) கொண்ட குறிக்கோவையை உருவாக்கலாம்.

எ.கா :

```
10 LET C$=STRING$(14,42)
20 PRINT C$;"TAMIL NADU";C$
RUN
**************TAMIL NADU**************
```

16. UCASE $ (X $)

கீழ்நிலைக் குறியிலுள்ள [Lower case Character] X$ என்ற குறிக் கோவையை மேல்நிலைக் குறிக் [Upper case Character] கோவை யாக மாற்றுகிறது.

எ. கா :

```
10 A$="tamil book"
20 PRINT UCASE$(A$)
RUN
TAMIL BOOK
```

17. VAL(String) : String என்ற குறிக்கோவையை எண்ணாக மாற்றுகிறது.

எ. கா :

```
10 A$="1055.25"
20 PRINT VAL(A$)
RUN
1055.25
```

9.3.3. தனிக்கூறுகள் (*SPECIAL FUNCTIONS*)

1. CVI [String 1], CVS [String 1], CVD [String 1]

இந்தக்கூறுகள் யாவும் செய்திக்கூறுக் கோவையிலிருந்து படித்துணரப்பட்ட குறிக்கோவை வடிவிலுள்ள [String] செய்திக் கூறுகளை [Data], முறையே முழு எண், ஒற்றைத் துல்லிய, இரட்டைத் துல்லிய எண்களாகவும் [Integer, Single Precision, Double Precision] மாற்ற உதவுகின்றன.

எ. கா :

```
10 OPEN "R",1,"A:PAYROLL.DAT",22

20 FIELD#1,10 AS EMP$,6 AS SAL$,6 AS DA$

30 GET#1

40 LET A=CVI(EMP$)

50 LET B=CVS(SAL$)

60 LET C=CVD(DA$)

70 PRINT A,B,C

RUN
```

விடை : 600   696.55   77.62

2. LOC (FILE #)

GET, PUT என்ற முறையிலா அணுகு முறைக் [Random Access] கோவையின் இடுவரல் / விடுவரல் ஆணைகளைப் பயன் படுத்தும் பொழுது ஒவ்வொரு பதிப்பையும் படித்த பிறகு அதற் கடுத்த பதிப்பின் பதிப்பு எண்ணை LOC குறிக்கிறது. வரிசைக்கிரம அணுகுமுறைக் கோவையில் [Sequential Access File] அதுவரை படித்த (அ) எழுதிய செய்தியை 128 குறிகளைக் கொண்ட பகுதிகளாகத் (Sector) தெரிவிக்கிறது.

எ.கா: `100 IF LOC(2) > 55 THEN PRINT "RECORD REACHED"`

3. MKI$, MKS$, MKD$

இவையாவும் இடுவரல் செய்யப்பட்ட எண் செய்திக்கூறை [Numeric Data] முறையிலா அணுகுமுறைக் கோவையில் எழுதுமுன் குறிக்கோவையாக [String] மாற்ற உதவுகின்றன.

MKI$: 2 பைட்ஸ் [bytes] நினைவக அளவுடைய முழு எண்ணைக் [Integer] குறிக்கோவையாக மாற்றுகிறது.

MKS$: 4 பைட்ஸ் [bytes] நினைவக அளவையுடைய ஒற்றைத் துல்லிய எண்ணைக் [Single Precision] குறிக்கோவையாக மாற்றுகிறது.

MKD$: 8 பைட்ஸ் [bytes] அளவு நினைவகத்தைக் கொண்ட இரட்டைத் துல்லிய எண்ணைக் [Double Precision] குறிக்கோவையாக [String] மாற்றுகிறது.

எ. கா:

```
10 LET X=10
20 LET Y=X+X^2+X^3+X^4
30 OPEN"R",2,"COMPUT.DAT",30
40 FIELD#2, 10 AS A$, 20 AS B$
50 LSET A$=MKI$(X)
60 LSET B$=MKS$(Y)
70 PUT#2:CLOSE#2
80 STOP
```

4. PEEK (Exp) : இதைப் பயன்படுத்தி '' Exp '' என்ற நினைவகப் பதிவகத்திலிருந்து செய்திக்கூறைப் [Data] படித்தறிய முடியும்.

எ. கா : 10 X=PEEK(&HD010)

5. POS (X) : குறிகாட்டியின் [Cursor Position] நிலையைக் காட்சித்திரையில் காட்ட உதவுகிறது.

6. USER (n) (Arg) : பொறிமொழி வடிவிலுள்ள [Machine Language] துணை வழியமைப்பை [Subroutine] அழைக்க [Call] இது உதவுகிறது.

எ.கா : 10 A=USR 5 (X)

7. VARPTR (X) : மாறி ''X'' இன் நினைவக எண்ணைக் [Memory Address] கண்டறிய உதவுகிறது.

9.4. கட்டுப்பாட்டுக் கட்டளைகள் (*CONTROL COMMANDS*)

இவை BBASIC–இல் மட்டும் அதன் வழியமைப்பைத் [Program] திருத்துவதற்கும் மாற்றியமைப்பதற்கும் பயன்படுகின்றன.

குறிப்பு : '' ∧ '' என்ற குறி CONTROL ஐக் குறிக்கின்றது.

∧ A — BBASIC இல் வழியமைப்பு வரியைத் [Program Lines] திருத்துவதற்குத் தொடக்க நிலையை [Re–edit] உருவாக்குகிறது.

∧ C — இது வழியமைப்பைச் [Program] செயற்பாட்டிலிருந்து [Execution] நிறுத்துவதற்குப் பயன்படுகிறது.

∧ G — காட்சிப் பொறியிலுள்ள மணியை [BELL] ஒலிக்கச் செய்கிறது.

∧ H — முன்பு இடுவரல் செய்யப்பட்ட குறியை [Character] அழிக்கிறது.

^I — இது ஒவ்வொரு 8-ஆவது குத்து வரிசைக்கும் [Column] இடையில் இடைவெளி கொடுக்கிறது.

^O — இது வழியமைப்பைச் செயற்பாட்டிலிருந்து [Execution] நிறுத்துவதற்கும், பின்னர் தொடர்வதற்கும் பயன்படுகிறது.

^R — Edit - செய்யப்படும் வழியமைப்பு வரியைத் [Program Lines] திரும்பவும் வெளிப்படுத்துகிறது [Display again] .

^S — வழியமைப்புச் செயற்பாட்டைத் [Programming Execution] தற்காலிகமாக நிறுத்துகிறது.

^Q — மேற்குறிப்பிட்ட செயலுக்குப் பிறகு வழியமைப்பின் செயற்பாடு [Execution] தொடர்வதற்கு இக்கட்டளை பயன்படுத்தப்படுகிறது.

^U — முன்பே பதிவு செய்யப்பட்ட வழியமைப்பு வரி [Program Line] அழிக்கப்படுகிறது.

^X — முன்பே பதிவு செய்யப்பட்ட வழியமைப்பு வரியின் கடைசிக் குறி அழிக்கப்பட்டு BBASIC கட்டளை Mode -க்குச் செல்கிறது.

9.5. திருத்து முறைக் கட்டளைகள் (*EDIT COMMANDS*)

BBASIC –சூழ்நிலையில் [Environment] வழியமைப்பைத் திருத்துவதற்குப் பல வசதிகள் தேவைப்படுகின்றன. வழியமைப்பின் ஒரு வரியிலுள்ள வழியமைப்பு ஆணையில் உள்ள பிழையைத் திருத்தவும் புதிய குறிக்கோவையையோ [String] எண்ணையோ [Number] புகுத்தவும் தேவையற்ற வழியமைப்பு ஆணைகளை நீக்கவும் சில திருத்து முறைக் கட்டளைகள் தேவைப்படுகின்றன. அவை கீழ்க்கண்டவாறு விளக்கப்பட்டுள்ளன.

1. X [String] : வழியமைப்பின் ஆணை வரிசையை [Statement Number] நீட்டிப்பதற்கும் [To Extend and Insert], மேலும் புதிய ஆணைகளைப் புகுத்துவதற்கும் இது பயன்படுகிறது.

2. nD : குறிகாட்டி [Cursor] உள்ள நிலையிலிருந்து 'n' எண்ணிக்கையுள்ள குறிகளை இது அழிக்கிறது.

3. H [String] [Escape]

வரியின் மீதமுள்ள வழியமைப்பாணைகளை அழிக்கவும் குறிப்பிட்ட குறிக்கோவையை எழுதவும்பயன்படுகிறது.

4. nSC : வழியமைப்பாணையில் 'C' என்ற குறியின் 'n' நிகழ்வைத் தேடி, அங்கு குறிகாட்டியைக் [Cursor] கொண்டு நிறுத்துகிறது.

5. [Carriage Return]

திருத்துமுறைச் சூழ்நிலை [Edit Mode] யிலிருந்து விடுபட்டு, வழியமைப்பாணையில் செய்த மாற்றங்களைப் பதிவு செய்கிறது.

6. nCd : குறிப்பிடப்பட்ட 'n' எண்ணிக்கையுள்ள குறிகளைக் கொண்டு குறிகாட்டியைத் [Cursor] தொடர்ந்த குறிகளை மாற்றுகிறது.

7. Q : Edit சூழ்நிலையில் செய்யப்பட்ட திருத்தங்களைப் பதிவு செய்யாமல் கட்டளைச் சூழ்நிலைக்கு [Command Mode] மாறுகிறது.

8. L : செய்யப்பட்ட திருத்தங்களுடன் வழியமைப்பு வரியை முழுவதும் தெரியப் படுத்துகிறது [Display Line]. அதன் பிறகு திருத்தம் செய்யும் பொருட்டு, குறிகாட்டி [Cursor] வரியின் தொடக்கத்தில் இருத்தப் படுகிறது.

9. A : முன்பே திருத்தம் செய்யப்பட்ட வழியமைப்பு வரியைத் [Program Line] திரும்பவும் திருத்தம் [Edit] செய்வதற்கு இந்தக் கட்டளை உதவுகிறது.

9.6. ABASIC மற்றும் BBASIC-இல் வழியமைப்பு செயற்படும் முறை

(*PROGRAM EXECUTION UNDER ABASIC & BBASIC*)

9.6.1. ABASIC

Text Editor மூலம் வழியமைப்பு [Program] இடுவரல் [Input] செய்யப்படுகிறது. இதில் வழியமைப்பைச் செயற்படுத்துவதற்கு இரண்டு நிலைகள் உள்ளன. ஒன்று மொழி மாற்றம் செய்தல், மற்றொன்று இடைநிலை மொழியிலுள்ள வழியமைப்புக் கோவையைச் [Intermediate File] செயற்படுத்தல். வழியமைப்புக் கோவையின் பெயர் " ACCOUNT. BAS " என்று வைத்துக்கொள்வோம். இந்தக் கோவையை எங்ஙனம் செயற்படுத்துவது என்பதைக் கீழ்க்கண்ட கட்டளைகள் மூலம் விளக்கிக் கூறலாம்.

எ. கா :
```
OA>ABASIC ACCOUNT.BAS
( Compilation Step )
```

இதன் மூலம் மொழிமாற்றம் செய்யப்பட்ட வழியமைப்பு ACCOUNT. INT என்ற கோவையில் பதிவு செய்யப்பட்டிருக்கும். மொழிமாற்றம் செய்யப்படும்பொழுது வழியமைப்பில் தவறு ஏதேனும் இருப்பின், பின்பு பிழைச் செய்தி மூலம் [Error Mes-age] அறிவிப்பாளருக்கு அறிவிக்கப்படும். பின்னர் திருத்தப்

பட்டு மீண்டும் மொழிமாற்றம் [Compilation] செய்தல் வேண்டும். தவறு ஏதும் இல்லாவிடில் அது கீழ்க்கண்டவாறு செயற்படுத்தப்படுகிறது.

எ. கா : OA} ARUN ACCOUNT

9.6. 2. BBASIC

இது ஒரு 'இன்டர்பிரிட்டர்' ஆகும். எனவே இதன் வழியமைப்பு [Program] பேஸிக் வழியமைப்புச் சூழ்நிலையிலேயே இடுவரல் செய்யப்பட்டுத்தவறுகள் திருத்தப்படுகின்றன. வெளிப்பிழை[Syntax Error] ஏதேனும் இருப்பின், செயற்படுத்தப் படும்பொழுது பிழைச் செய்தி [Error Message] அறிவிக்கப் படுகிறது. பின்னர் அது திருத்தப்பட்டு [Debugging] மீண்டும் செயற்படுத்தப்படும். வழியமைப்பு செயற்படும்பொழுது, இடையில் அதை நிறுத்தவோ, மாறி, மாறிலிகளின் மதிப்பைச் சோதிக்கவோ [Interaction] பின்னர் செயற்பாட்டைத்தொடரவோ, BBASIC 'இன்டர்பிரிட்டரில்' வசதிகள் உள்ளன.

இவற்றைக் கீழ்க்கண்டஆணைகளின் மூலம் குறிப்பிடலாம்.

எ. கா : OA}BBASIC COMPUT.BAS ‹ C R ›

இதன் பிறகு கணிப்பொறி வழியமைப்புச் சூழலுக்குச் [Programming Environment] செல்கிறது. பின்னர் தேவையான வழியமைப்புக் கோவையைத்[Program File] துணை நினைவகத்திலிருந்து முதன்மை நினைவகத்திற்குக் கொண்டுவருகிறது. [Loading the Program File from Disk to Memory]. பின்னர் வழியமைப்பு, கீழ்க்கண்டவாறு செயற்படுத்தப்படுகிறது.

```
LOAD "ACCOUNT.BAS"

RUN  ‹ CR ›
```

பின்னிணைப்பு
(APPENDIX)

பின்னிணைப்பு-1 (APPENDIX-A)

மாதிரி வழியமைப்புகள் [SAMPLE PROGRAMS]

PROGRAM NO. 1

```
10 REM ***** PROGRAM FOR RANDOM NUMBER GENERATION **********
20 CNT=0
30 A=RND:CNT=CNT+1
40 LPRINT A
50 LPRINT
60 IF CNT>5 THEN STOP
70 GOTO 30
80 END
RUN
 .245121

 .305003

 .311866

 .515163

 .0583136
```

PROGRAM NO. 2

```
10 REM ************* FIBONACCI SERIES **************************
20 LPRINT
30 LPRINT TAB(15);"FIBONACCI    SERIES"
40 LET K=0
50 LET N1=1                'LAST
60 LET N2=0                'LAST BUT 1
70 LPRINT
80 LPRINT TAB(15);"NUMBER","TERM"
90 LPRINT
100 LPRINT TAB(15);0,0                     'LINE ZERO ALREADY KNOWN
110 LET N=N1+N2            'FORM NEXT TERM
120 LET K=K+1               'INCREMENT COUNTER
130 LPRINT TAB(15);K,N
140 IF N>1000! THEN 180               'STOP IF OVER 1,00,000
150 LET N2=N1             'REPLACE LAST BUT 1
160 LET N1=N               'REPLACE LAST
170 GOTO 110
180 STOP
190 END
```

FIBONACCI SERIES

NUMBER	TERM
0	0
1	1
2	2
3	3
4	5
5	8
6	13
7	21
8	34
9	55
10	89
11	144
12	233
13	377
14	610
15	987
16	1597

PROGRAM NO. 3

```
10 REM ***** PROGRAM TO FIND STANDARD DEVIATION **********
20 DIM X(25)
30 FOR I=1 TO 25:READ X(I):NEXT I
40 N=25:SUM=0
```

```
50 FOR I=1 TO 25:SUM=SUM+X(I):NEXT I
60 MEAN=SUM/N:SIGX=0
70 FOR I=1 TO 25:SIGX=SIGX+(X(I)-MEAN)^2:NEXT I
80 STDEV=SQR(SIGX/N)
90 LPRINT"MEAN VALUE OF THE OBSERVATION IS = ";MEAN :LPRINT
100 LPRINT"STANDARD DEVIATION OF OBSERVATION  IS";STDEV:STOP:END
110 DATA 1,3.5,2.1,40,5,30,5.9,90,10,2,55,34,78,5.678,9.1234
120 DATA 123,24,61,78,100,21,23,53,67,23

RUN
MEAN VALUE OF THE OBSERVATION IS =  37.7721

STANDARD DEVIATION OF OBSERVATION  IS 34.5258
```

PROGRAM NO. 4

```
10 REM ***** SOLUTION OF AN EQUATION BY  NEWTON RAPHSON METHOD ******
20 REM **** Y=3X-2LOG(X)-5 ************
30 LPRINT"ITERATION";TAB(14);"X0";TAB(29);"X1":LPRINT
40 READ X0,ER
50 FOR K=1 TO 20
60 FX0=3*X0-2*LOG(X0)-5
70 DFX0=3-2/X0:X1=X0-FX0/DFX0
80 LPRINT K;TAB(10);X0;TAB(25);X1
90 IF (ABS(X1-X0)-ER)<=0 THEN 120
100 X0=X1
```

```
110 NEXT K
120 LPRINT:LPRINT"ROOT OF THE EQUATION IS ";X1:STOP:END
130 DATA 5,.0001

RUN
ITERATION     XO               X1

 1          5                2.39188
 2          2.39188          2.19247
 3          2.19247          2.18895
 4          2.18895          2.18895

ROOT OF THE EQUATION IS  2.18895
```

PROGRAM NO. 5

```
10 REM ***** ITERATION METHOD OF SOLVING SIMULTANEOUS EQUATIONS ******
20 REM 5X1+X2+X3=10:X1+4X2+X3=12:X1+2X2+6X3=23
30 LPRINT TAB(7)"X1";TAB(21);"X2";TAB(33);"X3":LPRINT
40 FOR I=1 TO 3
50 FOR J=1 TO 3
60 READ A(I,J)
70 NEXT J,I
80 FOR I=1 TO 3:READ C(I):NEXT I
90 FOR K=1 TO 3:X(K)=0:NEXT K
100 FOR N=1 TO 20
110 FOR I=1 TO 3
```

```
120 S=0
130 FOR J=1 TO 3
140 IF (I-J)=0 THEN 160
150 S=S+A(I,J)*X(J)
160 NEXT J
170 X(I)=(C(I)-S)/A(I,I)
180 NEXT I
190 LPRINT TAB(6);X(1);TAB(20);X(2);TAB(32);X(3)
200 IF N=8 THEN STOP
210 NEXT N
220 END
230 DATA 5,1,1,1,4,1,1,2,6,10,12,23
```

RUN

```
     X1            X2          X3

     2             2.5         2.66667
     .966667       2.09167     2.975
     .986667       2.00958     2.99903
     .998278       2.00067     3.00006
     .999853       2.00002     3.00002
     .999992       2           3
     1             2           3
     1             2           3
```

PROGRAM NO. 6

```
10 REM ********* LAGRANGES INTERPOLATION  FOR A GIVEN VALUE OF X ******
20 REM ********* Y=L1(X)Y1+L2(X)Y2+L3(X)Y3 ... LN(X)YN ********
30 N=7:FOR I=1 TO 7:READ X(I),Y(I):NEXT I
40 XINT=1.03
50 LPRINT "X VALUE","INTERPOLATION VALUE":LPRINT
60 P=0
70 FOR J=1 TO N
80 A(J)=1
90 FOR JJ=1 TO N
100 IF (JJ-J)=0 THEN 120
110 A(J)=A(J)*(XINT-X(JJ))/(X(J)-X(JJ))
120 NEXT JJ
130 P=P+A(J)*Y(J)
140 NEXT J
150 LPRINT XINT,P
160 XINT=XINT+.02:IF (XINT-1.15)<0 THEN 60 ELSE STOP
170 DATA 1,.76159,1.05,.78181,1.1,.8005,1.15,.81775,1.2,.83365
180 DATA 1.25,.84828,1.3,.86512

RUN

X VALUE         INTERPOLATION VALUE

 1.03            .773853
 1.05            .78181
```

```
 1.07          .789489
 1.09          .796892
 1.11          .804052
 1.13          .811001
 1.15          .81775
```

PROGRAM NO. 7

```
10 REM ***** PROGRAM FOR BEST FITTING OF STRAIGHT LINE *****
20 FOR I=1 TO 8:READ X(I),Y(I):NEXT I
30 LPRINT TAB(5);"X";TAB(15);"Y";TAB(20);"X^2";TAB(27);"XY":LPRINT
40 SX=0:SY=0:SX2=0:SXY=0
50 FOR I=1 TO 8
60 A=X(I)*X(I):B=X(I)*Y(I):SX=SX+X(I):SY=SY+Y(I):SX2=SX2+A:SXY=SXY+B
70 LPRINT TAB(4);X(I);TAB(12);Y(I);TAB(20);A;TAB(25);B
80 NEXT I
90 LPRINT:LPRINT TAB(4);SX;TAB(12);SY;TAB(20);SX2;TAB(25);SXY
100 D=1/(8*SX2-SX^2)
110 SLOPE=(8*SXY-SX*SY)*D
120 DINTER=(SX2*SY-SX*SXY)*D
130 LPRINT:LPRINT "SLOPE =  ";SLOPE;SPC(3);"CONSTANT =  ";DINTER:LPRINT
140 LPRINT "EQUATION OF THE LINE IS Y =";SLOPE;"X +";DINTER
150 DATA 1,3.249,2,3.522,3,4.026,4,4.332,5,4.9,6,5.121,7,5.601,8,5.893
160 STOP:END

RUN
```

X	Y	X^2	XY
1	3.249	1	3.249
2	3.522	4	7.044
3	4.026	9	12.078
4	4.332	16	17.328
5	4.9	25	24.5
6	5.121	36	30.726
7	5.601	49	39.207
8	5.893	64	47.144
36	36.644	204	181.276

SLOPE =   .389953   CONSTANT =   2.82572

EQUATION OF THE LINE IS Y = .389953 X + 2.82572

PROGRAM NO. 8

```
10 REM *** PROGRAM FOR NUMERICAL INTEGRATION USING SIMPSONS AND
20 REM *** TRAPEZOIDAL RULE ********* (X^2+X+1)dx ********
30 DIM F(301)
40 A=1:B=3:N=101
50 AN=N:IN=N-1
60 H=(B-A)/(AN-1)
70 FOR I=1 TO N:X1=I:X=A+(X1-1)*H
```

```
80 F(I)=X*X+X+1
90 NEXT I:S=0
100 FOR I=1 TO N:S=S+F(I):NEXT I
110 S1=S-(F(1)+F(N))/2
120 TRAPEZ=S1*H:SUM=0
130 FOR I=3 TO N STEP 2
140 SUM=SUM+F(I-2)+4*F(I-1)+F(I)
150 NEXT I:SIMPS=SUM*H/3
160 LPRINT "AREA IS DIVIDED INTO ";IN;"EQUAL PARTS":LPRINT
170 LPRINT "AREA BY TRAPEZOIDAL RULE IS";TRAPEZ
180 LPRINT "AREA BY SIMPSONS RULE IS";SIMPS
190 N=N+100
200 LPRINT:LPRINT
210 IF (N-301)<=0 THEN 50 ELSE STOP:END

RUN

AREA IS DIVIDED INTO  100 EQUAL PARTS

AREA BY TRAPEZOIDAL RULE IS 14.6668
AREA BY SIMPSONS RULE IS 14.6667


AREA IS DIVIDED INTO  200 EQUAL PARTS

AREA BY TRAPEZOIDAL RULE IS 14.6667
AREA BY SIMPSONS RULE IS 14.6667
```

```
AREA IS DIVIDED INTO  300 EQUAL PARTS

AREA BY TRAPEZOIDAL RULE IS 14.6667
AREA BY SIMPSONS RULE IS 14.6667
```

PROGRAM NO. 9

```
10 REM **** THIS PROGRAM PRODUCES HISTOGRAM (SIDEWAYS) ****
20 DIM X(100)
30 REM **** READ NUMBER OF SCORES ****
40 READ N
50 IF N<1 THEN 80
60 IF N>100 THEN 80
70 GOTO 110
80 LPRINT "ONLY 1 TO 100 SCORES PLEASE"
90 LPRINT "RE-EXAMINE DATA STATEMENTS."
100 STOP
110 REM *** FIND MAX FOR SCALING
120 F9=0               'MAX.FREQUENCY
130 FOR I=1 TO N       'READ AND CHECK FREQUENCIES
140 READ F
150 IF F<0 THEN 180
160 IF F>999999! THEN 180
170 GOTO 220
180 REM *** DATA ERROR
```

```
190 LPRINT"ERROR IN DATA ITEM";I
200 LPRINT"FREQUENCY OF";F;"IS OUT OF BOUNDS(0-99999)"
210 STOP
220 IF F>F9 THEN 240
230 GOTO 250
240 F9=F             'NEW MAX.
250 X(I)=F:NEXT I
260 REM *** SCALING FOR WIDTH
270 F1=0:LPRINT
280 LPRINT"SCALED FROM 0 TO ";F9:LPRINT
290 LPRINT"1 STAR=";F9/66;"UNITS."
300 LPRINT
310 FOR I=1 TO N
320 T=X(I)/F9
330 LPRINT I;TAB(5);"!";
340 FOR J=0 TO (T*50-.5)
350 LPRINT"*";:NEXT J:LPRINT
360 NEXT I
370 DATA 15,20,13,36,30,27,13,29,21,23,24,20,20
380 DATA 20,11,15,14,12,10,18,10,11,17,08,10
390 END


RUN
```

```
SCALED FROM 0 TO  36

1 STAR= .545455 UNITS.

 1  !****************************
 2  !******************
 3  !**************************************************
 4  !******************************************
 5  !**************************************
 6  !******************
 7  !****************************************
 8  !*****************************
 9  !********************************
 10 !*********************************
 11 !****************************
 12 !****************************
 13 !***********************************
 14 !***************
 15 !*********************
```

PROGRAM NO. 10

```
10 REM **** PROGRAM FOR THE BIAS CIRCUIT DESIGN PROBLEM ****
20 PRINT"FOR SILICON TRANSISTOR, ENTER 1, FOR GERMANIUM ,ENTER 0 ";
30 INPUT S
40 PRINT"ENTER MIN & MAX, DESIGN TEMPERATURES ,MIN & MAX ,BETA "
```

```
50 INPUT T1,T2,H1,H2
60 PRINT"ENTER SUPPLY VOLTAGE ,EMITTOR RESISTOR,RES TOLERANCE IN %"
70 INPUT VO,R4,P
80 PRINT"ENTER V(CE),I(CBO) IN MA,MIN & MAX.I(E) IN MA"
90 INPUT V1,IO,I1,I2
100 I1=I1*(1+.03*P)
110 I2=I2*(1-.03*P)
120 H1=H1*.865*EXP(.00575*T1)
130 T3=T2-25
140 H2=H2*(.865*EXP(.00575*T2)-(S-1)*(.00895-.00565*T3+.00048*T3^2))
150 IF S=1 THEN 170
160 IO=IO*EXP(.075*T3)
170 R3=2000*(VO-V1)/(I1+I2)-R4
180 R6=((I2-I1)*R4+2.5*(T2-T1))/(IO+I1/(H1+1)-I2/(H2+1))
190 IF R6>O THEN 210
200 LPRINT"I(E) RANGE TOO NARROW":STOP
210 V6=I1*.001*(R6/(H1+1)+R4)+.2+.5*S-.0025*(T1-25):LPRINT
220 LPRINT"BIAS VOLTAGE =";V6;"VOLTS"
230 LPRINT"BIAS RESISTOR=";R6;"OHMS"
240 LPRINT LPRINT"FOR STABILIZED BIAS CIRCUIT"
250 LPRINT:LPRINT "R(B-VCC)=";VO*R6/V6
260 LPRINT"R(B-GND)=";VO*R6/(VO-V6)
270 LPRINT"R(COLL.)=";R3
280 LPRINT"R(EMITTER)=";R4
290 STOP:END

RUN
```

```
BIAS VOLTAGE = 2.35224 VOLTS
BIAS RESISTOR= 9202.8 OHMS

FOR STABILIZED BIAS CIRCUIT

R(B-VCC)=39123.7
R(B-GND)=12033.3
R(COLL.)=3263.16
R(EMITTER)=2000
```

PROGRAM NO. 11

```
10 REM **** PROGRAM FOR CALCULATING COMPOUND INTEREST ****
20 DIM A(12)
30 INPUT "WHAT IS YOUR INITIAL AMOUNT (1 TO 5000)";A
40 IF A<1 THEN 80
50 IF A<=5000 THEN 100
60 LPRINT "TOO LARGE!"
70 GOTO 30
80 LPRINT "TOO SMALL!"
90 GOTO 30
100 FOR I=2 TO 12 STEP 2
110 LET A(I)=A      'SET INITIAL VALUES
120 NEXT I
130 LPRINT "COMPOUND INTEREST TABLE, FOR AN INITIAL AMOUNT OF Rs. ";A
140 LPRINT:LPRINT "PERIOD"
```

```
150 LPRINT "(YEARS)";
160 FOR I=2 TO 12 STEP 2
170 LPRINT TAB(5*I-2);I;"%";:NEXT I:LPRINT:LPRINT
180 FOR P=1 TO 25
190 FOR I=2 TO 12 STEP 2
200 LET A(I)=A(I)+(A(I)*I/100):NEXT I
210 LPRINT P;
220 FOR I=2 TO 12 STEP 2
230 LPRINT TAB(5*I-2);A(I);
240 NEXT I:LPRINT
250 NEXT P:STOP:END
```

RUN

COMPOUND INTEREST TABLE, FOR AN INITIAL AMOUNT OF Rs. 200

PERIOD (YEARS)	2 %	4 %	6 %	8 %	10 %	12 %
1	204	208	212	216	220	224
2	208.08	216.32	224.72	233.28	242	250.88
3	212.242	224.973	238.203	251.942	266.2	280.986
4	216.486	233.972	252.495	272.098	292.82	314.704
5	220.816	243.331	267.645	293.866	322.102	352.468
6	225.233	253.064	283.704	317.375	354.312	394.765
7	229.737	263.186	300.726	342.765	389.743	442.136
8	234.332	273.714	318.77	370.186	428.718	495.193

9	239.019	284.662	337.896	399.801	471.59	554.616
10	243.799	296.049	358.17	431.785	518.748	621.17
11	248.675	307.891	379.66	466.328	570.623	695.71
12	253.648	320.207	402.439	503.634	627.686	779.195
13	258.721	333.015	426.586	543.925	690.454	872.699
14	263.896	346.335	452.181	587.439	759.5	977.423
15	269.174	360.189	479.312	634.434	835.45	1094.71
16	274.557	374.596	508.07	685.189	918.995	1226.08
17	280.048	389.58	538.555	740.004	1010.89	1373.21
18	285.649	405.163	570.868	799.204	1111.98	1537.99
19	291.362	421.37	605.12	863.14	1223.18	1722.55
20	297.189	438.225	641.427	932.191	1345.5	1929.26
21	303.133	455.754	679.913	1006.77	1480.05	2160.77
22	309.196	473.984	720.708	1087.31	1628.06	2420.06
23	315.38	492.943	763.95	1174.29	1790.86	2710.47
24	321.687	512.661	809.787	1268.24	1969.95	3035.73
25	328.121	533.167	858.374	1369.69	2166.94	3400.01

PROGRAM NO. 12

```
10 REM **** THIS PROGRAM IS TO CREATE A SEQUENTIAL ACCESS FILE ****
20 INPUT"TYPE THE NAME OF THE FILE : ",FIL$
30 OPEN "O",1,FIL$
40 INPUT"TYPE THE STOCK NUMBER,QTY ARRIVAL,QTY ISSUED ",STK$,QTYR,QTYS
50 WRITE#1,STK$,QTYR,QTYS
60 INPUT"WANT TO ENTER MORE ?(Y/N) :",X$
```

```
70 IF X$="Y" THEN GOTO 40
80 IF X$<>"N" THEN INPUT"INCORRECT ENTRY...TYPE Y or N ONLY ",X$:GOTO 70
90 CLOSE#1 'The following routine will display the data
100 OPEN "I",1,FIL$:GOSUB 160:CNT=0
110 LPRINT"SL.NO.";TAB(18);"STOCK NO.";TAB(40);"QTY-ARRIVAL";
120 LPRINT TAB(60);"QTY-ISSUED":GOSUB 160
130 INPUT#1,STK$,QTY1,QTY2:CNT=CNT+1
140 LPRINT CNT;TAB(18);STK$;TAB(40);QTY1;TAB(60);QTY2
150 IF EOF(1) THEN GOSUB 160:CLOSE#1:STOP ELSE 130
160 LPRINT STRING$(80,"-"):RETURN
```

RUN

*** DATA LIST ***

SL.NO.	STOCK NO.	QTY-ARRIVAL	QTY-ISSUED
1	EC12378	750	275
2	EC12458	500	300
3	EC12765	5200	3100
4	EC15435	320	180
5	EC17456	480	100
6	EC18546	550	200
7	EC17548	4200	3700
8	EC16578	490	280
9	EC15746	580	70
10	EC16986	2000	1200

PROGRAM NO. 13

```
10 REM **** THIS IS TO CREATE A RANDOM ACCESS FILE FOR DATA ENTRY ****
20 GOSUB 190
30 INPUT"Type the Author's name :",A$:INPUT"Type the Title's Name :",B$
40 INPUT"Type the Imprint :",C$:INPUT"Type the Accession No/Call No : ",D$
50 INPUT"Type the Subject Heading Code :",SD
60 LSET B$(1)=A$:LSET B$(2)=B$:LSET B$(3)=C$:LSET B$(4)=D$
70 LSET B$(5)=MKI$(SD)
80 PUT#1:PRINT:INPUT"More to enter?(Y/N) :",X$
90 IF X$="Y" THEN 30
100 IF X$<>"N" THEN INPUT"TYPE Y or N ONLY ",X$:GOTO 90
110 CLOSE#1
120 GOSUB 190 'This routine is to Display the Data
130 GOSUB 230
140 LPRINT"SL.NO.";TAB(8);"AUTHOR";TAB(38);"TITLE";TAB(78);"IMPRINT";
150 LPRINT TAB(113);"ACC.NO";TAB(123);"SUB-CODE":GOSUB 230:CNT=0:LPRINT
160 CNT=CNT+1:GET#1:FOR I=1 TO 4:A$(I)=B$(I):NEXT I:SN=CVI(B$(5))
170 LPRINT CNT;TAB(8);A$(1);TAB(38);A$(2);TAB(78);A$(3);TAB(113);A$(4);
180 LPRINT TAB(123);SN
190 IF EOF(1) THEN CLOSE#1:LPRINT:GOSUB 230:STOP ELSE 160
200 OPEN "R",1,"CATALOG.DAT",120
210 FIELD#1,30 AS B$(1),40 AS B$(2),35 AS B$(3),10 AS B$(4),5 AS B$(5)
220 RETURN
230 LPRINT STRING$(132,"*"):RETURN

RUN
```

*** DATA LIST ***

SL.NO.	AUTHOR	TITLE	IMPRINT	ACC.NO	SUB-CODE
1	RAJARAM (K)	ERI KARKAL	TAMILPUTHTHAKALAYAM\MADRAS	236	1
2	VITHALREDDI (K)	MANA CITRELEKHANA SASTRAMU	TERLUGU ACADEMY\HYDERABAD	239	3
3	CHIDAMBARA RAO G V	BALDEEKA VIGNANALMU	VENKATARAMA AND CO\VIJAYAVADA	241	4
4	KENAPPA (BELLA)	PRAYOGA DEEPIKA	TELUGU ACADEMY\HYDERABAD	345	7
5	KRSNACAITNYA	VIJNANATTINTEE MAULIKATAVATRAYAM	SAHITYAPRARISHAD\ERNAKULAM	65	3
6	SAMBASIVAM V	PENNUTAATTA CHONNA KATHAIGAL	PUTHU ILAKKIYA\MADRAS	8765	87
7	DATTA (P K)	INTARMIDIYAT	TELUGU ACADEMY\HYDERABAD	4567	74
8	MUNIYAPPA (P T)	ARIMURAI VEDHIYIYAL	ORIENT LONGMAN\MADRAS	654	27
9	SESHADRI (H S)	ARGYANIK RASA VANAVIJANANYA	INST OF KANNADA\MYSORE	3456	32
10	SUBBABHAT K	RASAYANA SASTRA	VISVAVIDYA NILAYA\MYSORE	765	59
11	VIRUPAKASAPPA B	NITYA JIVANADALLIRASAYANA SASTRA	VISVAVIDYALAYA\DARWAR	154	102
12	APPUSAMY IYER PN	BHOOMIYINUM ULLAE	KALAIMAL\MADRAS	754	22
13	NAGANNA	BHUVIJNANA	BANGALORE UNIV\BANGALORE	3445	67
14	RANGAYYA	BHUVIJNANA PARICAYA	KANNADA ADHYAYAN\MYSORE	1245	29
15	VIDYANATHAN T S	NAVARATHIANGAL	ADHEENAM\TIRUVAVADUTHURAI	167	73
16	DEVARAJA	JEEVASASTRA PARICAYA	VISVAVIDYALA\MYSORE	3457	37
17	GOLDMAN	ADIMANUSYAR	PRABHATAM PRINTERS\ERNAKULAM	177	92
18	SINDHE	JIVA JIVANA	MAIITRIPRAKASHANA\MANGALORE	6543	54
19	VENKARAYAPPA	MANAVA SASTRA	VISVAVIDYA NILAYA\MYSORE	7662	81
20	CAMAYYA	SASYA SASTRA	VISVAVIDYANILAYA\MYSORE	7552	28
21	CIKKANNAYYA P S	SASYASASTRA	VISVAVIDYALAYA\DHARWAR	2443	83
22					0

PROGRAM NO. 14

```
10 REM **** A MODEL PAYROLL PROGRAM ****
20 REM **** THIS PROGRAM IS TO ENTER THE DATA FOR THE PAYOLL SOFTWARE
30 DIM B(20),C(20),A$(20),B$(20),C$(20),MON$(20),MONTH$(20)
40 PRINT CHR$(26):ANR$="PAYROLL & PERSONNEL INFORMATION SYSTEM":GOSUB 180
50 ANR$="MANAGEMENT CONSULTANCY HOUSE":GOSUB 180:ANR$="THANJAVUR,TAMIL NADU"
60 GOSUB 180:ANR$=STRING$(L,"-"):GOSUB 180
70 PRINT"You may Choose any one of the following Activity at a time :"
80 RESTORE 160:PRINT:PRINT:FOR I=1 TO 4:READ AB$(I):ANR$=AB$(I)
90 PRINT TAB(25);ANR$:NEXT I:L=LEN(ANR$):ANR$=STRING$(L,"-")
100 PRINT TAB(25);ANR$:PRINT:PRINT
110 INPUT"Your Choice Please :",CHS
120 ON CHS GOSUB 200,1420,920,140
130 GOTO 40
140 PRINT STRING$(80,"*"):PRINT:PRINT TAB(30);"**** THANK YOU ***"
150 PRINT:PRINT STRING$(80,"*"):SYSTEM
160 DATA "01 ----- DATA ENTRY " ,"02 ----- PAYROLL PROCESSING "
170 DATA "03 ----- PRINT PAYSLIP "."04 ----- JOB OVER"
180 L=LEN(ANR$):TAB=(80-L)/2
190 PRINT TAB(TAB);ANR$:RETURN
200 REM ****** DATA ENTRY FOR FILE PERSON1.DAT ***********
210 INPUT "WANT TO DO DATA ENTRY IN PERSON1.DAT FILE  ? (Y/N) :";X$
220 IF X$="N" THEN GOTO 380
230 IF X$<> "Y" THEN GOSUB 1670:GOTO 220
240 GOSUB 760
250 KK=1:LL=0:S=0:INPUT "TYPE THE EMPLOYEE CODE :",EMP
```

```
260 INPUT "NAME ",NAM$
270 INPUT "DESIGNATION:",DES$
280 INPUT "DEPT# ",DEP:INPUT "DATE OF JOINING :",DAT$:INPUT "ADDRESS :",ADD$
290 INPUT "TYPE LEAVE BALANCE CL,EL,SL ",CL,EL,SL
300 LSET A$(1)=MKI$(EMP):LSET A$(2)=NAM$:LSET A$(3)=DES$:LSET A$(4)=MKI$(DEP)
310 LSET A$(5)=DAT$:LSET A$(6)=ADD$:LSET A$(7)=MKI$(CL):LSET A$(8)=MKI$(EL)
320 LSET A$(9)=MKI$(SL)
330 PUT#1
340 INPUT "MORE TO DO (Y/N) ? :",X$
350 IF X$="Y" THEN GOTO 250
360 IF X$<>"N" THEN GOSUB 1670:GOTO 340
370 CLOSE#1
380 INPUT "WANT TO DO DATA ENTRY IN PAYMAS.DAT ? (Y/N)  :",X$
390 IF X$="N" THEN GOTO 570
400 IF X$<> "Y" THEN GOSUB 1670:GOTO 380
410 GOSUB 800
420 INPUT "TYPE THE EMPLOYE CODE :",EMP
430 INPUT "BASIC,  D.A,  H.R.A,  C.C.A ",BSC,DA,HRA,CCA
440 INPUT "TYPE THE OVERTIME RATES DAY, NIGHT, HOLIDAY:",DAY,NIG,HOL
450 INPUT "TYPE L.I.C, P.F, F.P.F, INCOME TAX ",LIC,PF,FPF,ITAX
460 LSET B$(1)=MKI$(EMP):LSET B$(2)=MKS$(BSC)
470 LSET B$(3)=MKS$(DA):LSET B$(4)=MKS$(HRA)
480 LSET B$(5)=MKS$(CCA):LSET B$(6)=MKS$(DAY)
490 LSET B$(7)=MKS$(NIG):LSET B$(8)=MKS$(HOL)
500 LSET B$(9)=MKS$(LIC):LSET B$(10)=MKS$(PF)
510 LSET B$(11)=MKS$(FPF):LSET B$(12)=MKS$(ITAX)
520 PUT#2
```

```
530 INPUT "MORE TO DO ?  <Y/N>  :",X$
540 IF X$="Y" THEN GOTO 420
550 IF X$<>"N" THEN GOSUB 1670:GOTO 530
560 CLOSE#2
570 INPUT "WANT TO DO DATA ENTRY IN TRANSACTION FILE ? <Y/N> :",X$
580 IF X$="N". THEN  RETURN
590 IF X$<>"Y" THEN GOSUB 1670:GOTO 580
600 GOSUB 850
610 INPUT "TYPE EMPLOYEE CODE:",EMP:INPUT "LEAVES TAKEN EL,CL,SL :",EL,CL,SL
620 INPUT "OVERTIME HOURS-DAY ,NIGHT,HILIDAY :",DAY,NIG,HOL
630 INPUT "PRODUCTION BONUS :",PROD:INPUT "LOSS OF PAY-NO OF DAYS ",NUM
640 INPUT "SPECIAL ALLOWANCE :",ALLO:INPUT "CANTEEN CHARGE :",CAN
650 INPUT "TRANSPORT CHARGE :",TRS
660 LSET C$(1)=MKI$(EMP):LSET C$(2)=MKS$(EL)
670 LSET C$(3)=MKS$(CL):LSET C$(4)=MKS$(SL)
680 LSET C$(5)=MKS$(DAY):LSET C$(6)=MKS$(NIG):LSET C$(7)=MKS$(HOL)
690 LSET C$(8)=MKS$(PROD):LSET C$(9)=MKS$(NUM):LSET C$(10)=MKS$(ALLO)
700 LSET C$(11)=MKS$(CAN):LSET C$(12)=MKS$(TRS)
710 PUT #3
720 INPUT "MORE TO DO ? <Y/N> :",X$
730 IF X$="Y" THEN GOTO 720
740 IF X$<> "N" THEN GOSUB 1670:GOTO 610
750 CLOSE#3:GOSUB 1630:RETURN
760 OPEN "R",1,"PERSON1.DAT",100
770 FIELD#1,5 AS A$(1),20 AS A$(2),15 AS A$(3),5 AS A$(4),10 AS A$(5),
780 FIELD#1,30 AS A$(6),5 AS A$(7),5 AS A$(8),5 AS A$(9)
790 RETURN
```

```
800 OPEN "R",2,"PAYMAS.DAT",49
810 FIELD#2,5 AS B$(1),4 AS B$(2),4 AS B$(3),4 AS B$(4),4 AS B$(5),
820 FIELD#2,4 AS B$(6),4 AS B$(7),4 AS B$(8),4 AS B$(9),4 AS B$(10),
830 FIELD#2,4 AS B$(11),4 AS B$(12)
840 RETURN
850 OPEN "R",3,"TRANSACT.DAT",49
860 FIELD#3,5 AS C$(1),4 AS C$(2),4 AS C$(3),4 AS C$(4),4 AS C$(5),
870 FIELD#3,4 AS C$(6),4 AS C$(7),4 AS C$(8),4 AS C$(9),4 AS C$(10),
880 FIELD#3,4 AS C$(11),4 AS C$(12)
890 RETURN
900 OPEN "O",4,"PAYOUT.DAT":RETURN
910 OPEN"R",5,"DEPART.DAT",25:FIELD#5,25 AS DD$:RETURN
920 ANR$="MANAGEMENT CONSULTANCY HOUSE":GOSUB 950:ANR$="THANJAVUR":GOSUB 950
930 ANR$="PAYSLIP FOR THE MONTH OF "+MONTH$+" 1985":GOSUB 950
940 LPRINT :GOTO 960
950 L=LEN(ANR$):TT=(80-L)/2:LPRINT TAB(TT);ANR$:RETURN
960 OPEN "I",1,"PAYOUT.DAT"
970 GOSUB 1690   'To Read the Data from the File
980 LPRINT"EMP#:  ";EMP;TAB(35);"DESIGNATION:";DES$
990 LPRINT "NAME :  ";NAM$; TAB(35);"DEPARTMENT :";DEP$
1000 LPRINT STRING$(80,"-")
1010 LPRINT "EARNINGS:";:LPRINT TAB(60);"Rs   .Np"
1020 LPRINT"--------";TAB(60);"--------"
1030 LPRINT"BASIC                .....   .......   .......   .......";TAB(60);
1040 LPRINT USING"#####.##";BSC
1050 LPRINT"D.A                  .....   .......   ......    .......";TAB(60);
1060 LPRINT USING"#####.##";DA
```

```
1070 LPRINT"H.R.A                ..... ....... ...... .......";TAB(60);
1080 LPRINT USING"#####.##";HRA
1090 LPRINT"C.C.A                ..... ....... ...... .......";TAB(60);
1100 LPRINT USING"#####.##";CCA
1110 LPRINT"OVERTIME             ..... ....... ...... .......";TAB(60);
1120 LPRINT USING"#####.##";OT
1130 LPRINT"PROD.BONUS           ..... ....... ...... .......";TAB(60);
1140 LPRINT USING"#####.##";PROD
1150 LPRINT"SPECIAL ALLOWANCE    ..... ....... ....... .......";TAB(60);
1160 LPRINT USING"#####.##";SPC
1170 LPRINT TAB(60);"---------":LPRINT TAB(40);"TOTAL .......";TAB(60);
1180 LPRINT USING "#####.##";TOTAL:LPRINT TAB(60);"---------"
1190 LPRINT TAB(5);"DEDUCTION:":LPRINT TAB(5);"---------"
1200 LPRINT TAB(5);"P.F              ..... .......... ............";
1210 LPRINT TAB(60);:LPRINT USING"#####.##";PF
1220 LPRINT TAB(5);"F.P.F            ..... .......... ............";
1230 LPRINT TAB(60);:LPRINT USING"#####.##";FPF
1240 LPRINT TAB(5);"L.I.C            ..... .......... ............";
1250 LPRINT TAB(60);:LPRINT USING"#####.##";LIC
1260 LPRINT TAB(5);"I.TAX            ..... .......... ............";
1270 LPRINT TAB(60);:LPRINT USING"#####.##";ITAX
1280 LPRINT TAB(5);"LOSS OF PAY      ..... .......... ............";
1290 LPRINT TAB(60);:LPRINT USING"#####.##";LOSS1
1300 LPRINT TAB(5);"CANTEEN CHARGE   ..... .......... ............";
1310 LPRINT TAB(60);:LPRINT USING"#####.##";CAN
1320 LPRINT TAB(5);"TRANSPORT CHARGE ..... .......... ............";
1330 LPRINT TAB(60);:LPRINT USING"#####.##";TRN
```

```
1340 LPRINT TAB(60);"----------"
1350 LPRINT TAB(40);"TOTAL .......";TAB(60);
1360 LPRINT USING "#####.##";DED
1370 LPRINT TAB(60);"----------"
1380 LPRINT TAB(10);"NET SALARY ...... Rs ";:LPRINT USING"#####.##";TOTAL-DED
1390 LPRINT TAB(40);"Signature:"
1400 IF EOF(1) THEN CLOSE#1:RETURN ELSE 970
1410 IF EOF(1) THEN CLOSE#1:RETURN ELSE 970
1420 INPUT "TYPE THE DATE OF PROCESSING AS (DAY/MONTH/YEAR):",A$
1430 FOR I=1 TO 12:READ MON$(I):NEXT I
1440 DATA JANUARY,FEB,MARCH,APRIL,MAY,JUNE,JULY,AUGUST,SEP,OCT,NOV,DEC
1450 C$=MID$(A$,4,2):D=VAL(C$):MONTH$=MON$(D)
1460 GOSUB 760:GOSUB 800:GOSUB 850:GOSUB 900:GOSUB 910
1470 GET#1:IF EOF(1) THEN CLOSE#1:CLOSE#2:CLOSE#3:CLOSE#4:CLOSE#5:RETURN
1480 A(1)=CVI(A$(1)):AA$(2)=A$(2):AA$(3)=A$(3):A(4)=CVI(A$(4)):GET#5,A(4)
1490 AA$(4)=DD$
1500 GET#2:B(1)=CVI(B$(1)):FOR J=2 TO 12:B(J)=CVS(B$(J)):NEXT J
1510 GET#3:C(1)=CVI(C$(1)):FOR J=2 TO 12:C(J)=CVS(C$(J)):NEXT J
1520 OT=C(5)*B(6)+C(6)*B(7)+C(7)*B(8)
1530 EAR=OT+B(2)+B(3)+B(4)+B(5)+C(8)+C(10)
1540 EAR1=B(2)+B(3)+B(4)+B(5)
1550 LOSS=EAR1/30*C(9)
1560 DED=B(9)+B(10)+B(11)+B(12)+C(11)+C(12)+LOSS
1570 NET=EAR-DEDUCTION
1580 WRITE#4,A(1)
1590 WRITE#4,AA$(2),AA$(3),AA$(4)
1600 WRITE#4,B(2),B(3),B(4),B(5),OT,C(8),C(10),EAR,B(10),B(11)
```

```
1610 WRITE#4, B(9),B(12),C(9),C(11),C(12),DED,NET
1620 GOTO 1470
1630 GOSUB 910
1640 FOR I=1 TO 13:INPUT "TYPE THE DEPARTMENT NAME:",DEP$
1650 LSET DD$=DEP$:PUT#5:NEXT I
1660 CLOSE#5:RETURN
1670 INPUT "TYPE Y OR N ONLY :",X$:RETURN
1680 GOSUB 760:GOSUB 800:GOSUB 850
1690 INPUT#1,EMP
1700 INPUT#1,NAM$,DES$,DEP$
1710 INPUT#1,BSC,DA,HRA,CCA,OT,PROD,SPC,TOTAL,PF,FPF,LIC,ITAX
1720 INPUT#1,LOSS,CAN,TRN,DED,NET
1730 RETURN

RUN
```

MANAGEMENT CONSULTANCY HOUSE
THANJAVUR

PAYSLIP FOR THE MONTH OF SEP. 1985

EMP#: 500 DESIGNATION:SYSTEM ANALYST
NAME : TAMIL SELVAN S. DEPARTMENT :COMPUTER CENTRE

EARNINGS:	Rs .Np
BASIC	1500.00
D.A	800.00
H.R.A	200.00
C.C.A	100.00
OVERTIME	0.00
PROD.BONUS	200.00
SPECIAL ALLOWANCE	100.00
TOTAL	2900.00

DEDUCTION:

P.F	200.00
F.P.F	100.00
L.I.C	150.00
I.TAX	150.00
LOSS OF PAY	0.00
CANTEEN CHARGE	75.00
TRANSPORT CHARGE	60.00
TOTAL	735.00

NET SALARY ...... Rs 2165.00

Signature:

பின்னிணைப்பு-2 (APPENDIX-B)

கலைச் சொற்பட்டியல் (List of Technical Terms)

A

Abacus	மணிச்சட்டம்
Access	அணுக்கம்
Access Time	அணுகு நேரம்
Account	கணக்கு
Accounting Machine	கணக்கிடு கருவி
Accumulator	சேர்ப்பிடம்
Accuracy	நுட்பம், முடிவுறுதி
Adding Machine	கூட்டுப் பொறி
Air Conditioned	காற்றுச் சீரடைந்த, வெப்பக் கட்டுப்பாட்டுடைய
Algebra	குறிக்கணக்கியல்
Algorithm	செய்முறை நிருணயம் / செய்வழி நிருணயம்
Alphabet	எழுத்து
Alphanumeric	எண்ணெழுத்து
Analog Computer	ஒப்புமைக் கணிப்பொறி, ஒப்பியல் கணிப்பொறி
Analytical Engine	பகுப்பாய்வுக் கருவி
AND Gate	உம்மிணைக் கதவு
Appendix	பின்னிணைப்பு
Arithmetic Operation	கணக்கியல் செயல்
Arithmetic Unit	கணக்ககம்
Arrays	நினைவக வரிசை
Assembler	இடைநிலை மொழிமாற்றி
Auxiliary Storage	துணை நினைவகம், துணைக் கொள்ளகம்

B

Back Space	பின்னிடம்
Bar Printer	சட்ட அச்சுப் பொறி
Basic	அடிப்படை
Binary	இருநிலை
Binary Coded Decimal	இருநிலைக்குறியீட்டுத் தசம எண்
Bistable	இருநிலையமை
Bistable Circuit	இருநிலையமைச் சுற்று
Bit	இருநிலைக்குறி, துண்டு
Block	தொகுப்பு
Block Diagram	வினைக்கட்ட வரைவு
Block Data	செய்திக்கூறுத் தொகுப்பு
Boolean Algebra	பூலியன் குறிக்கணக்கியல்
Branch Instruction	வழிமாற்றி ஆணைகள்
Brush	துடைப்பான்
Budget	மதிப்பீடு, வரவு செலவுத் திட்டம், அரசிறைக் கணக்கு
Buffer Register	தாங்கிப் பதிவறை

C

Calculator	கணிப்பான்
Carbon Ribbon	கரிநாடா
Card	அட்டை
Card Code	அட்டைக் குறியீடு
Card Deck	அட்டைக் கொத்து
Card Punch	அட்டைத் துளைப் பொறி
Card Punch Control	அட்டைத் துளைப் பொறி இயக்கி
Card Reader	அட்டைப் படிப்பொறி
Card Reader Control	அட்டைப் படிப்பொறி இயக்கி
Carry	மிகுதி
Cell	சிற்றறை

Central Processing Unit	மையச் செயலகம்
Chain Printer	சங்கிலி அச்சுப் பொறி
Character	குறி
Check Bit	தணிக்கைக் குறி
Cheque	காசோலை
Chip	பல்மின் சுற்றடக்கி / பல்மின் சுற்றுச் சேர்ப்பி
Clear and Add	அழித்துச் சேர்
Clerical	எழுத்து சம்பந்தமான
Code	குறியீடு
Column	குத்து வரிசை
Common	பொது
Compiler	மொழி மாற்றி
Complement	மாற்று, மாறுபட்டு
Complementary Arithmetic	மாற்றுமுறைக் கணக்கு
Complementary Number	மாற்று எண்
Computer	கணிப்பொறி
Computer Language	கணிப்பொறி மொழி
Computerization	கணிப்பொறி மயமாக்கு
Concentric Tracks	மையவட்ட வரிகள்
Concurrent Processing	ஒருமித்த இயக்க வசதி
Console	இயக்கு நிலையகம்
Constant	மாறிலி
Control	கட்டுப்பாடு
Control Unit	இயக்ககம், கட்டுப்பாட்டகம், ஆளகம்
Control Statements	கட்டுப்பாட்டாணைகள்
Core	உள்
Core Storage	உள் நினைவகம்
Cost	செலவு
Count	எண்ணுக, எண்ணிக்கை
Counter	எண்பான்
Creative Thinking	ஆக்க நினைவு

Cursor	குறிகாட்டி
Cycle	சுற்று

D

Data	செய்திக் கூறு
Data Organization	செய்திக் கூறமைப்பு
Data Division	செய்திக் கூறுப் பிரிவு
Data Entry Terminal	விவரம் பதியும் கருவி
Data Processing	விவர ஆய்வனை
Debugging	பிழை திருத்தல்
Decimal Code	தசமக் குறி
Decimal Point	தசமப் புள்ளி
Decimal Number	தசம எண்
Definition	வரையறை
Delay	தாமதி
Design	வடிவமைப்பு
Destructive	அழிவுள்ள
Development	முன்னேற்றம்
Diagnostic	பிழையுணர்தல், பிழை காணல்
Difference Engine	வேறுபாட்டுக் கருவி
Digital	எண்ணிலக்கம்
Digital Computer	எண்ணிலக்கக் கணிப்பொறி
Diode	இருங்கடத்திகள்
Disc Storage	தட்டு நினைவகம்
Display Unit	காட்சிப்பொறி
Display Unit Control	காட்சிப்பொறி இயக்கி
Double precision	இரட்டைத் துல்லியம்
Drum Printer	உருளை அச்சுப் பொறி
Drum Storage	உருளை நினைவகம்
Duplicate	இருபடியாக்கு

E

Edit Sub Command	திருத்துமுறைக் கட்டளைகள்
Electrical	மின்னியல்
Electronic	மின்னணுவியல்
Electronic Components	மின் நுண்கருவிகள்
Electronic Flipflop	மின்னணு நிலை மாறி
Electronic Pulse	மின்னணுத் துடிப்பு
Electronic Tubes	மின்னணுக் குமிழ்கள்
Element	கருப்பொருள்
Encoding	செய்திக்கூறு மாற்றல்
End	முடிவு
End of Block Marker	தொகுப்பு முடிவுக் குறி
End of Field Marker	தொகுதி முடிவுக் குறி
End of Record Marker	பதிப்பு முடிவுக் குறி
Environment Design	சூழ்நிலை வடிவமைப்பு
Equivalent	சம மதிப்புள்ள, சமமான
Erasing Head	அழிமுனை
Error	தவறு, பிழை
Error Detecting Code	பிழையறி குறியீடு
Excess	அதிகமாக
Execute	நிறைவேற்று, செய்துகாட்டு
External Memory	வெளி நினைவகம்
External Plug In / Out Wiring System	வெளிப்புறக் கம்பியிணைப்பு முறை

F

Fast Access Storage	வேக அணுக்கப் பதிவு முறை
Fast Memory	மிகைவேக நினைவகம்
Fault Time	பிழை நேரம்
Field	தொகுதி
File	கோவை

Fixed Program Computer	நிலையான வழியமைப்புக் கணிப்பொறி
Fixed Point Constants	நிலைப்புள்ளி மாறிலிகள்
Flip Flop	நிலை மாறி
Floating Point Constants	அலைப்புள்ளி மாறிலிகள், அலை எண்
Flow Chart	வழிமுறைப்படம் / செய்வழி வரைவு
Format	படிவம்
Formula	விதி வாய்ப்பாடு, சூத்திரம்
Full Adder	முழுக் கூட்டல் பொறி
Full Adder Circuit	முழுக் கூட்டல் மின் சுற்று
Function Table	கூறு அட்டவணை
Functions	கூறுகள்

G

Gate	கதவு
General File	பொதுக்கோவை
General Purpose Computer	பொதுச் செயல் கணிப்பொறி
Generalised Control	பொது இயக்கம்
Generalised Registers	பொதுவாக்கிய பதிவறைகள்
Grammar	இலக்கணம்

H

Half Adder	அரைக் கூட்டல் பொறி
Half Select Pulse	பாதி தெரிவுத் துடிப்பு
Halt	நிறுத்து
Hardware	கருவியம்
Header Label	தலைக் குறி
High Speed Memory	மிகை வேக நினைவகம்
Higher Level Language	உயர்நிலை மொழி
History of Computer	கணிப்பொறியின் வரலாறு

Home Computer	மனைக் கணிப்பொறி
Hybrid Computer	கலப்பினக் கணிப்பொறி
Hydrogen Bomb	நீர்வாயுக் குண்டு
Hypothetical Computer	கற்பனைக் கணிப்பொறி (புனைவுக் கணிப்பொறி)

I

Ideal	நேர்த்தி
Identification Division	அறிமுகப் பிரிவு
Image	உருவம்
Image Analysis	படப்பகுப்பாய்வு
Immediate	உடனடியாக
Implication	உட்குறிப்பீடு
Imply	உட்குறிப்பீடு
Incompatible	முரண்பட்ட
Incremental	கூடுதலான
Independent	கட்டற்ற, தனித்த
Index Register	சுட்டுப் பதிவறை
Indexing	சுட்டுதல்
Indicator	காட்டி
Indirect	மறைமுகமாக
Infinite	எல்லையில்லா
Information	செய்தி
Inhibit	தடை செய்
Initialise	ஆரம்ப நிலைப்படுத்து, தொடங்கு
Inner Loop	உள்வளை
Inner Memory	உள்நினைவகம்
Input	இடுவரல்
Input Data	இடுவரல் செய்திக் கூறு
Input Unit	இடுவரலகம்
Instruction	ஆணை, கட்டளை
Integer	முழுஎண்

Integer Constant	முழு எண் மாறிலி
Integer Variable	முழு எண் மாறி
Inter Record Gap	பதிப்பு இடைவெளி
Interactive Facility	நேரடித் தொடர்பு வசதி
Interchange	இடமாற்று
Intermediate	இடைநிலை
Internal Memory	உள் நினைவகம்
Internal State	உள் நிலை
Interpreter	மொழிமாற்றி
Intersection	ஊடறு பகுதி
Interstate	இடைநிலை
Interval	இடைவெளி
Inventory	பொருள் நிலை
Inverse	தலை கீழ், தலை கீழான
Inverter	தலைகீழாக்கி
Irredundant	மிகையற்ற
Iterate	திரும்பிச் செய்

J

Jet Plane	காற்றுந்து, வானூர்தி
Job Card	பணிக் குறியட்டை
Jump Instructions	மாற்று ஆணைகள்

K

Key Operations	உயர்நிலை இயக்கங்கள்
Kinematic	இயக்கம் சார்ந்த
Kinetic	இயக்கத்துக்குரிய

L

Label	குறிப்பிணை
Language	மொழி
Level	எல்லை நிலை
Library	சேர்ப்பகம், [படிப்பகம்]

Life Insurance Policies	ஆயுள் பாதுகாப்புப் படிவங்கள் [காப்புறுதிகள்]
Limit	எல்லை
Line Printer	வரி அச்சுப்பொறி
Linear	நேர்முகமான
Linkage	இணைப்பு
Load	நிரப்பு, பாரமேற்று
Location	இடம்
Logarithm	எண் சுருக்கீட்டு முறை
Logic	முறையமை
Logical	முறையமை
Logical Operation	முறையமை இயக்கம்
Logical Definition	முறையமை வரையறுப்பு
Logical Operator	முறையமை இயக்கி
Loops	செய்வளைகள்
Lower Case Character	கீழ் நிலைக்குறி

M

Machine Language	பொறி மொழி
Machine Word	பொறிச் சொல்
Magnetic Ink Character Recognition Technique	காந்தமைக் குறியீட்டு அறிமுறை
Magnetic Core Memory	காந்த உள் நினைவகம்
Magnetic Disc Pack	காந்தத் தட்டடுக்கு
Magnetic Drum	காந்த உருளை
Magnetic Flux	காந்தத் தாரை
Magnetic Tape	காந்த நாடா
Magnetic Tape Control	காந்த நாடா இயக்கி
Main	முதன்மை
Mechanisation	பொறிமயமாக்கு
Medical Record	மருத்துவத் தொகுப்பு
Memory	நினைவகம்

Memory Address Register	நினைவகக் குறியீட்டுப் பதிவறை
Microprocessor	நுண்ணியக்கி
Missiles	எறிபடைகள்
Multiplication	பெருக்கல்
Multiprocessing	பன் மைய இயக்கம்
Multiprogramming	பன் வழியமைப்புச் செயற்பாடு

N

Natural Number	இயற்கை எண், இயலெண்
Negation	எதிர்மாற்று, எதிர்மறை
Nested Loops	உள்ளடங்கு வளைகள்
Non Destructive Read Head	அழியாப்படிமுனை
Non Executable	நிறைவேற்றவொண்ணா
Non-numerical	எண்முறையில்லா
NOT Gate	எதிர்மறைக் கதவு
Number Systems	எண் முறைகள்
Numeric Constants	எண் மாறிலிகள்

O

Object Program	பொறிமொழி வழியமைப்பு
Octal System	எட்டிலக்க முறை
Off Line	இணைப்பற்ற
On Line	நேரியக்க முறை
Open Subroutine	திறந்த துணை வழியமைப்பு
Operand	இயமி
Operating Instructions	செய்முறை ஆணைகள்
Operating System	இயக்க வழியமைப்பு ஒருங்கு
Operation	இயக்கம்
Operation Code	இயக்கக் குறியீடு
Operation Register	இயக்கப் பதிவறை
Operational Research	நடைமுறை ஆய்வு
Operator	இயக்குபவர்

Optical Character Recognition Technique	கண்ணாடிக் குறி அறி முறை
OR Gate	அல்லதிணைக் கதவு
Organization	அமைப்பு
Output	விடுவரல்
Output Unit	விடுவரலகம்

P

Paper Tape	காகித நாடா
Parallel Transfer	இணைந்த மாற்றம்
Parity Bit	ஒப்புமைக் குறி
Pattern	வடிவமைப்பு
Peripheral	புற ஒருங்கு
Photo Electric Method	புகைமின் முறை
Planning	சூழ்ந்துணர்வு
Pointer	சுட்டு முள், இணைப்பு முனை
Postulate	முற்கோள்
Printer	அச்சுப் பொறி
Problem	கணக்கு, சிக்கல் வினா, புதிர்மானம்
Procedure Division	வழிவகைப் பிரிவு
Process	செயல்
Process Control	இயக்க ஆளுகை
Processor	செயலகம்
Program	வழியமைப்பு, வழிமுறை
Programmer	வழியமைப்பாளர்
Programming Flow Chart	வழியமைப்புச் செய்வழி வரைவு
Programming Language	வழியமைப்பு மொழி
Properties	பண்புகள்
Pulse	துடிப்பு
Punch	துளையிடு, துளைக்கருவி
Punched Card	துளையிட்ட அட்டை

R

Radix	அடிப்படை
Random	கட்டற்ற, முறையிலா
Random Access	கட்டற்ற அணுக்கம், முறையிலா அணுக்கம்
Random Access Memory	முறையிலா அணுக்க நினைவகம்
Read Instructions	படிப்பாணைகள்
Reading Head	படி முனை
Real Number	உண்மை எண்
Realtime	உண்மை நேரம்
Receiving Reel	அடையும் சுற்றுகை
Record	பதிப்பு
Recording	பதிப்பு,பதிவு செய்தல்
Rectangle	செவ்வகம்
Redundant	மிகையான
Register	பதிவறை
Relational Operator	தொடர்பமை இயக்கி
Relay	இயங்கு தடை
Reservation	ஒதுக்கீடு
Resource Sharing	வசதியைப் பகிர்ந்தியங்கல்
Result	முடிவு, விடை
Ring	வளையம்
Row	நீள் வரிசை, நீட்டு வரிசை

S

Sales Management	விற்பனை இயல்
Satellite	விண்வெளிக் கலம்
Scale	அளவு வீதம்
Selected Register	தேர்ந்த பதிவறை
Semantic Error	உட்பிழை
Semi Conductor	நிறைவிலா மின் கடத்தி
Sending Reel	அனுப்பும் சுற்றுகை
Sequence	கோர்வை
Sequential Access	வரிசை அணுக்கம்

Serial Adder	வரிசைக் கூட்டல் பொறி
Set	அடைவு
Seven Track	ஏழு பாதை
Shift Register	இடம், பெயர்ப் பதிவறை
Simple Variable	தனி மாறிகள்
Single Precision	ஒற்றைத் துல்லியம்
Slide Rule	கணக்குச் சட்டம்
Slow Memory	குறைவேக நினைவகம்
Software	கணிமம்
Space Craft	விண்வெளியூர்தி
Specification	தகுதிறன்
Speed	வேகம்
Spindle	உருளாணி, அச்சாணி
Stack Space	அடுக்கிடம்
Storage Register	நினைவகப் பதிவறை
Store	அகத்திலிடு
Stored Program Concept	நினைவிருத்த வழிமுறை
String	குறிக்கோவை
String Constants	குறிக்கோவை மாறிலிகள்
String Variables	குறிக்கோவை மாறிகள்
Sub Program	துணை வழியமைப்பு
Subscripted Variable	நினைவக நிருணய மாறிகள்
Subtraction	கழித்தல்
Sum	கூட்டுத் தொகை
Super Computer	உயர்நிலை நுட்பக்கணிப்பொறி
Supervisor	மேற்பார்வையாளர்
Symptoms	அறிகுறிகள்
Syntactic Error	வெளிப்பிழை
System	ஒருங்கு

T

Table	அட்டவணை
Tag	தொங்கு முனை
Tail Label	வால் குறி

Tape	நாடா
Tape Reader	நாடாப் படிப் பொறி
Tele Type Key Board	தொலை அச்சுத் தட்டுப் பலகை
Thought	எண்ணம்
Time Sharing Facility	நேரப் பகிர்வுச் செயற்பாட்டு முறை
Tracks	நீள வாட்டத் தாரைகள்
Transfer	மாற்று
Transfer Operation	மாற்றுச் செய்கை, பிரிவுஇயக்கம்
Transistor	முக்கடத்தி
Type Statements	தன்மை ஆணைகள்
Typewriter	தட்டெழுத்துப் பொறி

U

Unconditional	கட்டுப்பாடற்ற
Undefined Variable	வரையறுக்காத மாறி
Universal	முழுமொத்தமான, எல்லாமுட்பட்ட
Unspecified States	தெளிவுபடுத்தா நிலைகள்
Upper Case Character	மேநிலைக் குறி

V

Variable	மாறி
Verify	சரி பார்

W

Wheel Printer	சக்கர அச்சுப் பொறி
Wire	கம்பி
Word	சொல்
Word Length	சொல் நீளம்
Write	எழுது
Writing Head	எழுது முனை

Z

Zero	சுன்னம், சுழி

பின்னிணைப்பு—3(APPENDIX—C)
நூற்பட்டியல்

குலோத்துங்கன் : வளர்க தமிழ் தியாகராய நகர், சென்னை : பாரதி பதிப்பகம், (டிசம்பர் 1982).

குழந்தைசாமி வா. செ., வெங்கடாசலம் பி. ஏ., செல்லமுத்து கா.செ. கணிப்பொறி. (வெளியிடப்பட உள்ள தட்டச்சுப்படி).

Allan Lytel. ABC's of Computers. U.S.A: W.Foulsham & Co Ltd, (1966).

Barnett E.H. Programming time-shared computers in BASIC. New York: Wiley-Interscience, (1972).

Bunt R.B. An introduction to computer science. An algorithmic approach. Jean-Paul Tremblay.

Cay weitzman. Distributed Micro/Minicomputer Systems. New Jersey: Prentice-Hall,Inc., (1980).

Dana Gibson E. An introduction to automated data processing. New York: Harper & Row, Publishers Inc., (1966).

Davis G.B. Computer data processing. Tokyo: McGraw-Hill Kogakusha,Ltd. (1969).

Davis G.B Computers and information processing. Tokyo: McGraw-Hill Kogakusha,Ltd. (1978).

Davis W.S. Computers and business information processing. London: Addison-wesley publishing company, (1978).

Don Cassel and Martin Jackson. Introduction to computers and information processing. Reston publishing company, Inc.

Frank Herbert and Max Barnard. The home computer book. U.S.A: St Edmendsburg Press, (1982).

Gottfried B.S. Theory and problems of programming with BASIC. Asian student edition, (1982).

Marateck S.L. 2nd Edition BASIC. New York: Academic Press,Inc. (1982).

Miller R. BASIC programs for Scientists and Engineers. U.S.A: SYBEX Inc., (1981).

Pragati Computers. BBASIC user's manual. Pondicherry: Pragati Computers.

Pragati Computers. ABASIC user's manual. Pondicherry: Pragati Computers.

Rajaraman V. Sahasrabuddhe H.V. Computer programming in COBOL. New Delhi: Prentice-Hall of India Private Limited, (1981).

Richard Forsyth. The Basic Idea. London: Chapman and Hall, (1978).

Richard Mateosian. Inside BASIC games. U.S.A: SYBEX Inc., (1981).

Ruckdeschel F.R. Basic Scientific subroutines Vol I. New York: McGraw-Hill publications Co., (1981).

Stern R.A and Nancy Stern. An introduction to computers and information processing. New york: John Wiley & Sons, (1979).

Thomas Bartee. Introduction to computer science. Tokyo: McGraw-Hill kogakusha Ltd, (1975).

Wakerly J.F. Microcomputer Architecture and programming. U.S.A: John Wiley & Sons,Inc, (1981).

Weiss E.A. Computer usage fundamentals. New York: Computer usage company Inc., (1969).

William Barden JR. Guide book to small computers. Indiana (U.S.A): Howard W.Sams and co.,Inc., (1980).