Chapter - 5

अध्याय 5

कंप्यूटर से संबद्ध भाषाओं का भाषावैज्ञानिक अध्ययन

5.1 भाषा, भाषाविज्ञान तथा अभिकलनात्मक भाषाविज्ञान

5.1.1 भाषा तथा भाषाविज्ञान

5.1.2 भाषाविज्ञान तथा अभिकलनात्मक भाषाविज्ञान

5.1.3 भाषाविज्ञान : कला या विज्ञान ?

5.1.4 भाषाविज्ञान का अन्य विषयों से संबंध
(व्याकरण, भौतिक शास्त्र, शरीर विज्ञान,
तर्कशास्त्र, कंप्यूटर)

5.1.5 भाषिक संरचना और उसके विभिन्न स्तर
(भाषाविज्ञान के अंग - अर्थ, प्रोक्ति, वाक्य,
रूप, शब्द, ध्वनि, रूपस्वनिम , लेखन-लिपि)

5.2 प्राकृतिक भाषा तथा कंप्यूटर कुंजी-पटल (की-बोर्ड)

5.2.1 कुंजी-पटल की कार्य-प्रणाली
क) कुंजी-पटल का प्रकार
ख) कुंजियों की आवाज
ग) नए कुंजी-पटल
घ) कुंजी-पटल की मेमोरी
ङ) कुंजी-पटल की समस्याएँ तथा समाधान

5.2.2 कुंजियों का विवरण

5.2.3 विभिन्न कुंजी-पटलों का अध्ययन
क) देवनागरी तथा अँग्रेजी, रेमिंग्टन टाइपराइटर, ध्वन्यात्मक
ख) क्वर्टी तथा डीवोरक
ग) मानक फोंट कूट
इस्फोक, इनस्क्रिप्ट, ऑकी, इस्फा

5.2.4 प्राकृतिक भाषा तथा कुंजीपटल का भाषावैज्ञानिक अध्ययन
क) कंप्यूटर युग में देवनागरी लिपि की सार्थकता तथा गुण
ख) मानक हिंदी वर्तनी
ग) कुंजीपटल तथा कुंजियों का भाषावैज्ञानिक विश्लेषण
घ) प्रदर्श 1-6

5.3 मध्यवर्ती (इंटरफेस) भाषा

5.3.1 निम्न स्तरीय भाषाएँ (मशीनी भाषा, असेंबली भाषा)

5.3.2 उच्च स्तरीय भाषाएँ (सभी प्रोग्रामिंग भाषाएँ)

5.3.3 मध्यवर्ती भाषा का भाषावैज्ञानिक अध्ययन

5.4 द्विआधारी/द्विअंकीय (बाइनरी) भाषा

5.4.1 मानक कूट : ऑस्की तथा इस्की

5.4.2 संख्या पद्धति
(दशमलव, द्विआधारी, अष्टाधारी, षोडश आधारी पद्धति)

5.4.3 बाइनरी भाषा का भाषावैज्ञानिक अध्ययन

5.5 अध्याय 5 की संदर्भ सूची

5.1 भाषा, भाषाविज्ञान तथा अभिकलनात्मक भाषाविज्ञान

5.1.1 भाषा तथा भाषाविज्ञान

5.1.2 भाषाविज्ञान तथा अभिकलनात्मक भाषाविज्ञान

5.1.3 भाषाविज्ञान : कला या विज्ञान ?

5.1.4 भाषाविज्ञान का अन्य विषयों से संबंध
(व्याकरण, भौतिक शास्त्र, शरीर विज्ञान, तर्कशास्त्र, कंप्यूटर)

5.1.5 भाषिक संरचना और उसके विभिन्न स्तर
(भाषाविज्ञान के अंग - अर्थ, प्रोक्ति, वाक्य, रूप, शब्द, ध्वनि, रूपस्वनिम, लेखन-लिपि)

कंप्यूटर से संबद्ध भाषाओं का भाषावैज्ञानिक अध्ययन

5.1 भाषा, भाषाविज्ञान तथा अभिकलनात्मक भाषाविज्ञान

भाषा का प्रयोग परस्पर संप्रेषण के लिए किया जाता है। इसका व्यवस्थित, तार्किक तथा वैज्ञानिक अध्ययन भाषाविज्ञान करता है। कंप्यूटर के विकास के साथ ही भाषा और भाषाविज्ञान दोनों को नया आयाम प्राप्त हुआ है, जो तकनीकी है। जहाँ भाषा के रूप में अभिकलनात्मक-भाषा का विकास हुआ है, वहीं इस अभिकलनात्मक भाषा के अध्ययन के लिए अभिकलनात्मक-भाषाविज्ञान का विकास हुआ है।

5.1.1 भाषा तथा भाषाविज्ञान

(क) भाषा की परिभाषा

- **वांद्रिय हैं** - भाषा एक तरह का चिंह है। चिंह से आशय उन प्रतीकों से है जिनके द्वारा या जिनकी सहायता से मानव अपना विचार दूसरों पर प्रदर्शित करता है। ये प्रतीक कई प्रकार के होते हैं, जैसे - नेत्रग्राह्य, श्रोतग्राह्य या स्पर्शग्राह्य। वास्तव में **श्रोतग्राह्य प्रतीक भाषा की दृष्टि से सर्वश्रेष्ठ हैं।**
- **स्त्रुतेवां** - A language is a system of arbitrary vocal symbols by means of which, members of a social group co-operate & interact.
- **एनसाइक्लोपीडिया ब्रिटेनिका** - Language may be defined as arbitrary system of vocal symbols by by means of which, human beings, as members of a social group & participants in culture interact & communicate.
- **एक भारतीय विचारक** - भाषा एक विकसनशील, विश्लेषण-सापेक्ष, यादृच्छिक एवं ध्वनिमूलक सार्थक व्यवस्था है।
- **मोरियो एपाई तथा फ्रैंक ग्यानोर** - भाषा उन सार्थक और विश्लेषण समर्थ मानवोच्चारित ध्वनियों को कहते हैं जिनका प्रयोग मानव विचारों और भावों को व्यक्त करने के लिए करता है।
- **ए.ए.कार्डीनोर** - विचाराभिव्यक्ति के लिए व्यक्त ध्वनि-संकेत ही भाषा है।
- **डॉ.भोलानाथ तिवारी** - भाषा, उच्चारण अवयवों से उच्चारित मूलतः प्रायः यादृच्छिक (arbitrary) ध्वनि प्रतीकों की वह व्यवस्था है, जिसके द्वारा किसी भाषा-समाज के लोग आपस में विचारों का आदान-प्रदान करते हैं।

भाषा की इन परिभाषाओं को देखने पर भाषा की कुछ विशेषताएँ सामने आती हैं; यथा -

1) भाषा विचार-विनिमय का साधन है।
2) भाषा उच्चारण अवयवों से नि:सृत ध्वनि समष्टि होती है।
3) भाषा में प्रयुक्त ध्वनि समष्टियाँ सार्थक तो होती हैं, किंतु उनका भावों या विचारों से संबंध केवल 'यादृच्छिक' (arbitrary) या 'माना हुआ' होता है।
4) भाषा में एक व्यवस्था होती है।
5) एक भाषा का प्रयोग एक विशेष समाज या वर्ग में होता है। उसी में वह बोली और समझी जाती है।[1]

(ख) भाषाविज्ञान की परिभाषा

भाषाविज्ञान के अंतर्गत भाषा का ही अध्ययन-विशेष किया जाता है। इस संबंध में कतिपय भारतीय और विदेशी विद्वानों के भाषाविज्ञान संबंधी मत निम्नानुसार हैं :

- **आर.एच.रॉबिन्स (R.H.Robins)** - General linguistics is concerned with human languages as a universal & recognizable part of human behaviour & of the human faculties, perhaps one of the most essential of human life as we know it,& one of the most far-reaching of human capabilities in relation to the whole span of mankinds achievements.
- **चार्ल्स एफ हॉकेट (Charles F. Hockett)** - For a small group of specialists knowing about language is an end in itself. These specialists call themselves linguists & the organised body of infomation about language which their investigations produce is called linguistics.
- **ब्लॉक तथा ट्रेगर (Block & Trager)** - When he (linguist) has described the facts speech such a way as to account for all the utterances used by the members of a social group,his description is what we call the system or grammer of language.
- **डॉ.श्यामसंदर दास** - भाषाविज्ञान उस शास्त्र को कहते हैं, जिसमें भाषामात्र के भिन्न-भिन्न अंगों और स्वरूपों का विवेचन तथा निरूपण किया जाता है। ... ... सारांश यह है कि भाषाविज्ञान की सहायता से हम किसी भाषा का वैज्ञानिक दृष्टि से विवेचन, अध्ययन और अनुशीलन करना सीखते हैं। (भाषाविज्ञान)
- **डॉ.बाबूराम सक्सेना** - भाषा-तत्वों का अध्ययन भाषाविज्ञान का अध्ययन है।
- **डॉ. उदयनारायण तिवारी** - भाषाविज्ञान उस शास्त्र को कहते हैं जिसमें हम भाषा-मात्र के भिन्न-भिन्न अंगों का विवेचन, अध्ययन और अनुशीलन करना सीखते हैं। (भाषा-शास्त्र की रूपरेखा)

डॉ. भोलानाथ तिवारी - भाषाविज्ञान वह विज्ञान है जिसमें भाषा-विशिष्ट, कई और सामान्य - का समकालिक, ऐतिहासिक, तुलनात्मक और प्रायोगिक दृष्टि से अध्ययन और तद्-विषयक सिद्धांतों का निर्धारण किया जाता है।

- **डॉ. मंगलदेव शास्त्री** - भाषाविज्ञान उस विज्ञान को कहते हैं जिसमें (1) सामान्य रूप से मानवीय भाषा का ; (2) किसी विशेष भाषा की रचना और इतिहास का, और अंततः ; (3) भाषाओं या प्रादेशिक भाषाओं (या बोलियों) के वर्गों की पारस्परिक समानताओं और विशेषताओं का तुलनात्मक विचार किया जाता है।
- **प्रो. देवेंद्रनाथ शर्मा** - भाषाविज्ञान का सीधा अर्थ है भाषा का विज्ञान, और विज्ञान का अर्थ है विशिष्ट ज्ञान। इस प्रकार भाषा का विशिष्ट ज्ञान भाषाविज्ञान कहलाएगा।
- **एक विद्वान** - भाषाविज्ञान भाषाध्ययनार्थ एक गत्यात्मक विज्ञान है, जिसका विकासात्मक पल्लवन देशकाल के परिवेश में होता है।

  भाषाविज्ञान भाषा मात्र का व्यवस्थित अध्ययन है।[2]

5.1.2 भाषाविज्ञान और अभिकलनात्मक-भाषाविज्ञान में अंतर

भाषाविज्ञान के दो प्रमुख उपक्षेत्र हैं : प्रजनक भाषाविज्ञान (Generative Linguistics) और विश्लेषणात्मक भाषाविज्ञान (Analytical Linguistics)।

(क) प्रजनक भाषाविज्ञान :

प्रजनक उद्यम (Generative Enterprise) का प्रमुख लक्ष्य भाषा अर्जन की उस आरंभिक अवस्था का वर्णन करना है, जिसके अंतर्गत मानव-शिशु किसी खास भाषा-समुदाय के साहचर्य में अपने आत्मीय संबंधों के कारण भाषा का व्याकरणिक ज्ञान अर्जित करता है। भाषा अर्जन की यह आंतरिक व्यवस्था भाषा-निरपेक्ष होती है। इसी भाषा निरपेक्ष व्याकरण को ही सर्वभाषा व्याकरण (Universal Grammar) कहा गया है। प्रजनक उद्यम के अंतर्गत इन्हीं सार्वभौमिक तत्वों की खोज का प्रयास किया गया है। प्रजनक उद्यम में भाषा का अध्ययन संप्रेषण की दृष्टि से नहीं, बल्कि व्याकरणिकता की दृष्टि से किया जाता है। इसके अंतर्गत अर्थ के निरूपण पर बहुत ही कम ध्यान दिया गया है। यदि कभी अर्थ पर विचार करते भी हैं तो भी व्याकरणिकता के परिप्रेक्ष्य में ही किया जाता है। प्रजनक उद्यम का दूसरा महत्त्वपूर्ण पक्ष है, सारभौमिक तत्वों की खोज। वे भाषा-विशेष की प्रवृत्तियों का अध्ययन करने के बजाय भाषा-निरपेक्ष तत्वों की खोज में संलग्न रहते हैं। चॉम्स्की अपने नित-नवीन सिद्धांतों के माध्यम से इसी दिशा में प्रयत्नशील रहे हैं। दासगुप्ता (1991), गीता (1985) और जैन (1990) ने यह दावा किया है कि सार्वभौमिक व्याकरण का यह प्रयास विश्व की अनेक भाषाओं के संदर्भ में विशेष कर भारतीय भाषाओं के संदर्भ में सफल सिद्ध नहीं हो पाया है।

(ख) विश्लेषणात्मक भाषाविज्ञान :

भाषा-विशेष के व्याकरण के आधार पर उसके नियमों की खोज करना और आवश्यकता होने पर इन नियमों का प्रयोग उपयुक्त अभिकलनात्मक रूपवाद (Formalism) के लिए करना विश्लेषणात्मक भाषाविज्ञान के अंतर्गत आता है। भारतीय भाषाओं के संदर्भ

SMT. HANSA MEHTA LIBRARY
THE MAHARAJA SAYAJIRAO UNIVERSITY OF BARODA

में इस प्रकार का व्यवस्थित डाटा संस्कृत को छोड़ कर अन्य भारतीय भाषाओं में उपलब्ध नहीं है। इस प्रकार के डाटा के लिए मूल शब्दावली का संकलन, उनका आर्थी वर्गीकरण, वाक्यात्मक लक्षण आदि अत्यंत आवश्यक हैं। प्राचीन संस्कृत में यह कार्य निरुक्त और व्याकरण के रूप में किया गया था और अँग्रेजी में प्रजनक रूपावली (Generative Paradigm) के निर्माण से कई दशक पूर्व ही शब्द-वृत्त (Lexicon), शब्दकोश और विभिन्न प्रकार की सामग्री के रूप में डाटा एकत्र करने का कार्य आरंभ हो चुका था।

अभिकलनात्मक भाषाविज्ञान का मुख्य उद्देश्य प्राकृतिक भाषा को समझने और रचने के लिए अभिकलनात्मक मॉडल या सिद्धांतों का निर्माण करना रहा है। अभिकलनात्मक भाषाविज्ञान के अंतर्गत प्राकृतिक भाषा को संप्रेषण के माध्यम के रूप में देखा जाता है, जबकि प्रजनक उद्यम के अंतर्गत व्याकरणिकता के आधार पर प्राकृतिक भाषा का विवेचन किया जाता है। अभिकलनात्मक भाषाविज्ञान के अंतर्गत वही सिद्धांत स्वीकार्य हो सकते हैं, जिनका उपयोग प्राकृतिक भाषा के संसाधन के लिए किया जा सकता हो। इसके मुख्यतः **तीन पक्ष** हो सकते हैं। सर्वप्रथम, ये सिद्धांत प्रभावशाली हों ताकि इनकी सहायता से अपेक्षित कार्य की सिद्धि के लिए क्रियाविधियाँ तैयार की जा सकें। दूसरे, ये सिद्धांत सटीक और व्यापक हों, ताकि इनकी सहायता से कंप्यूटर मॉडल तैयार किए जा सकें। तीसरे, ये सिद्धांत व्यवस्थित हों, ताकि इनके प्रयोग में अधिक समय और कंप्यूटर-साधनों की बर्बादी न हो।

अभिकलनात्मक भाषाविज्ञान के अंतर्गत शोध कार्यों को प्रेरित करने वाली मुख्यतः दो दिशाएं रही हैं, सर्वप्रथम मेधावी कंप्यूटर प्रणाली (Intelligent Computer System) के निर्माण की प्रेरणा मुख्यतः प्रौद्योगिकीय कारणों से रही है। इस प्रणाली के अंतर्गत मशीनी अनुवाद, प्रणाली डाटा बेस के प्राकृतिक भाषाओं में अंतरापृष्ठ (Interface) सामान्य मानव-मशीन अंतरापृष्ठ, वाक् अभिज्ञान प्रणाली, पाठ विश्लेषण और बोधन प्रणाली, कंप्यूटर साधित शिक्षण, मुद्रित या हस्तलिखित पाठ को पढ़ने या समझने वाली प्रणाली आदि को रखा जा सकता है। दूसरी दिशा संज्ञानात्मक और भाषिक प्रेरणा की रही है, जिसके अंतर्गत मानव प्राकृतिक भाषा के माध्यम से कैसे संवाद स्थापित करता है, यह समझने के लिए अपेक्षित अंतर्दृष्टि को खोजने का प्रयास किया जाता है।

यही कारण है कि प्रजनक भाषाविज्ञान के अंतर्गत वाक्य-विज्ञान को अध्ययन का आधार बनाया जाता है, जबकि **अभिकलनात्मक भाषाविज्ञान में अर्थ विज्ञान और संकेत प्रयोग विज्ञान को सर्वाधिक महत्व प्रदान किया जाता है**, क्योंकि इसका उद्देश्य संप्रेषण के लिए कंप्यूटर मॉडल तैयार करना है और इसके लिए उक्ति को ग्रहण करना, निरूपित करना और प्रयोग करना अत्यंत आवश्यक है। इसी प्रकार अभिकलनात्मक भाषाविज्ञान के अंतर्गत भाषा विशिष्ट व्याकरण लिखना भी अत्यंत आवश्यक है, जबकि प्रजनक भाषाविज्ञान के अंतर्गत विभिन्न भाषाओं के सार्वभौमिक तत्त्वों की खोज करना ही प्रमुख ध्येय रहा है।

इन भिन्नताओं के बावजूद दोनों प्रकार के भाषाविज्ञानों में रूपात्मक सिद्धांत (Formal Theory) की विधि और प्रयोग में समानता है। दोनों ही क्षेत्रों में व्याकरण के नियमों की सहायता से वाक्यों का निर्माण यांत्रिक रूप में किया जा सकता है। इस संदर्भ में दोनों

शैक्षिक व्याकरण आरंभिक व्याकरणों से भिन्न हैं, क्योंकि शैक्षिक व्याकरण का स्वरूप अरूपात्मक (Non-Formal) था और इसका प्रयोग भाषा सीखने वाले मनुष्य द्वारा किया जाता था। वस्तुत: रूपात्मक सिद्धांतों के अभिकलनात्मक स्वरूप के कारण उक्त दोनों भाषाविज्ञानों में मामूली सा अंतर ही है। सामान्यत: प्रजनक व्याकरण के अंतर्गत रूपात्मक सिद्धांत का प्रयोग करते हुए वाक्यों के वास्तविक संसाधन पर अधिक बल नहीं दिया जाता है। यही कारण है कि प्रजनक भाषाविज्ञान के अंतर्गत विकसित सिद्धांतों को अभिकलन के लिए ज्यों का त्यों ग्रहण नहीं किया जा सकता। इसलिए ये सिद्धांत रूपात्मक और सटीक होने के साथ-साथ अभिकलनात्मक भी होने चाहिए। व्याकरणिकता, सार्वभौमिक तत्वों की खोज और अभिकलनात्मक स्वरूप के अलावा उक्त दोनों भाषाविज्ञानों के व्यावहारिक अनुप्रयोगों में भी भारी अंतर हैं। इसी अंतर को अभियांत्रिक भाषाविज्ञान (Engineering Linguistics) कहा जा सकता है।

अभिकलनात्मक भाषाविज्ञान के अनेक प्रत्यक्ष अनुप्रयोग हैं। इसके अंतर्गत विकसित सिद्धांतों का प्रयोग प्राकृतिक भाषाओं के संसाधन अर्थात् उन्हें समझने, रचने और अर्जित करने के लिए आवश्यक कंप्यूटर प्रणाली के निर्माण के लिए किया जाता है। इससे एक ऐसा व्यापक नेटवर्क तैयार हो जाता है, जिससे सिद्धांतों का निरंतर परीक्षण और प्रतिपुष्टि (Feed back) होती रहती है, जबकि इसके विपरीत प्रजनक भाषाविज्ञान का कोई व्यावहारिक अनुप्रयोग नहीं है। किसी भी कार्य प्रणाली के विकास के लिए पूर्ण विवरण जुटाना अत्यंत आवश्यक है। इसमें जरा भी चूक या कमी से सफलता संदिग्ध हो सकती है। किसी भी सिद्धांत की परीक्षा की यही वास्तविक कसौटी है।

भाषिक सिद्धांतों के संदर्भ में व्यापकता, समग्रता और संपूर्णता की इसी कमी के कारण इनका संसाधन ज्यों का त्यों कंप्यूटर पर संभव नहीं हो पाता। सैद्धांतिक विषयों पर चर्चा करते समय कठिन समस्याओं को बाद में निपटाने के लिए अलग रख दिया जाता है, जैसे प्राकृतिक भाषा के संसाधन के लिए वाक्य विज्ञान, अर्थ विज्ञान और संकेत विज्ञान का समन्वित अध्ययन आवश्यक है, लेकिन प्रजनक भाषाविज्ञान के अंतर्गत सरलता के कारण अर्थ विज्ञान और संकेत प्रयोग विज्ञान को छोड़ दिया जाता है। किंतु वास्तविक कार्य-प्रणाली के विकास के लिए आवश्यक है कि जटिल से जटिल समस्याओं से भी जूझने का प्रयास किया जाए। हो सकता है कि आरंभ में पाए गए समाधान तदर्थ (Ad-hoc) हों। तदर्थ समाधानों के गर्भ में नए सिद्धांतों की संभावनाएं छिपी हो सकती हैं।

कभी-कभी ऐसा भी होता है कि पूर्ण विकसित सिद्धांत भी किसी अवस्था में जा कर अनावश्यक मालूम पड़ने लगते हैं। जैसे अनेक मशीनी अनुवाद प्रणालियां आज रूपमिक विश्लेषण (Morphological Analysis) की उपेक्षा करने लगी हैं, क्योंकि शब्दार्थ के लिए शब्दों की व्युत्पत्ति का विशेष महत्त्व नहीं रह गया है।

इसी प्रकार यदि आप किसी प्रणाली के लिए 'उपयुक्त' व्याकरण को खोजने का प्रयास करते हैं तो 80-20 का नियम सामने आ जाता है। अर्थात् 20 प्रतिशत व्याकरण 80 प्रतिशत भाषा को समाहित कर लेता है, किंतु जैसे-जैसे 20 प्रतिशत व्याकरण की परतें खुलने लगती हैं तो भाषा का नया स्वरूप सामने आ जाता है, इसलिए यह याद रखना

आवश्यक है कि विश्व उतना व्यवस्थित नहीं है, जितना कि सिद्धांत उससे अपेक्षा करते हैं। हमारे लिए विस्मय का कोई कारण नहीं होना चाहिए यदि हम यह पाएं कि प्राकृतिक भाषा जैसे जटिल व्यवहार-क्षेत्र में भी सिद्धांत के अनुप्रयोग से हम सिद्धांत में कोई महत्वपूर्ण संशोधन करने के लिए विवश हो जाएं, क्योंकि इतने जटिल और विशाल व्यवहार-क्षेत्र में कार्य करते हुए सिद्धांत और प्रक्रिया दोनो में ही परिवर्तन की गुंजाइश बनी रहती है। अंततः यह भी संभव है कि अभिकलनात्मक मॉडल से संबंधित शोध कार्य प्रजनक भाषा-विज्ञान के अंतर्गत सिद्धांत-निर्माण की प्रक्रिया में सहायक सिद्ध हो।[3]

5.1.3 भाषाविज्ञान - कला या विज्ञान

सामान्यतः विज्ञान का अर्थ है-विशिष्ट ज्ञान अथवा विशिष्ट बोधयुक्त व्यवस्थित ज्ञान। दूसरी ओर, **शास्त्र** शब्द भी विज्ञान के साथ चलता है जिसका अर्थ है शासन करने वाला अर्थात क्या करणीय है और क्या अकरणीय है, यह बतलाना शास्त्र का काम है। आज सामान्य रूप से इन दोनों में भेद नहीं किया जाता है। जिस अर्थ में भौतिक, रसायन या गणित विज्ञान है उस अर्थ में भाषाविज्ञान विज्ञान नहीं है, क्योंकि **भाषाविज्ञान में कल्पना के लिए स्थान है** जबकि उपर्युक्त अन्यों में नहीं है।

भाषाविज्ञान का **आधार** विज्ञान की भाँति है, निष्कर्ष ग्रहण करने के लिए **वर्णन** भी संभव है, कुछ स्थाई परिणामों या सिद्धांतों के संदर्भ में **परीक्षण** भी संभव है। इसमें विज्ञान की भाँति **उपचारपूर्णता, अंतःसंगति,लाघव आदि** भी होते हैं। इसी आधार पर इसे विज्ञान माना जाता है, किंतु उसी सीमा तक जिस सीमा तक राजनीति या सामाजिक अध्ययन को विज्ञान माना जाता है, भौतिकविज्ञान आदि के समकक्ष नहीं।

कला को विस्तृत अर्थ में लेने पर भाषाविज्ञान को कला कहा जा सकता है। कला का क्षेत्र निश्चित नहीं है। समाजशास्त्र,राजनीतिविज्ञान आदि भौतिकशास्त्र की भाँति **निश्चित विज्ञान नहीं है,** अतः कला के अंतर्गत आते हैं।

कला का एकमात्र लक्ष्य मनोरंजन करना और सौंदर्य की सृष्टि करना है,जबकि भाषाविज्ञान का लक्ष्य इन दोनों से भिन्न भाषा से संबधित ज्ञान की वृध्दि करना है। इसके अलावा भाषा समाज की संपत्ति है, जबकि कला मनुष्य द्वारा उद्भूत है। इस प्रकार भाषा-विज्ञान कला न हो कर विज्ञान के अधिक निकट पड़ता है।

अध्ययन के विषयों को तीन वर्गों में रखा जाता है :

(क) प्राकृतिक विज्ञान (Natural Sciences), जैसे भौतिकविज्ञान ;

(ख) सामाजिक-विज्ञान (Social Sciences), जैसे समाजशास्त्र ;

(ग) मानविकी (Humanities), जैसे चित्रकला, साहित्य और संगीत कला, आदि।

भाषाविज्ञान निश्चित रूप से सामाजिक विज्ञान के निकट है, किंतु उसके विभाग प्राकृतिक विज्ञान या मानविकी के निकट भी पड़ते हैं ; जैसे - ध्वनि का अध्ययन प्राकृतिक विज्ञान के निकट ही पड़ेगा।[4]

5.1.4 भाषाविज्ञान का अन्य विषयों से संबंध

(1) भाषाविज्ञान और व्याकरण :

साम्य एवं भेद : हम भाषा विज्ञान के अंतर्गत भाषा का वैज्ञानिक अध्ययन करते हैं। इसमें भाषा जैसी है, वैसी का अध्ययन किया जाता है। इसका भाषा की विकृति अथवा परिष्कृति से कोई संबंध नहीं होता। यह तो कब, क्यों और कैसे का उत्तर देता है। इसके विपरीत व्याकरण भाषा के परिमार्जन पर बल देता है, जिसके परिणामस्वरूप भाषा में अवरोध पैदा हो जाता है। यह भाषा में अच्छे और बुरे का ध्यान केंद्रित रखता है। व्याकरण 'ऐसा होना चाहिए' कराता है जबकि भाषाविज्ञान कहता है 'ऐसा है' अथवा 'ऐसा नहीं है'। यदि व्याकरण कहता है कि अन्य पुरुष, एक वचन सर्वनाम के लिए 'वह' का प्रयोग होना चाहिए तो भाषाविज्ञान इसके संबंध में हमें बताता है कि वह प्रयोग 'कब' आरंभ हुआ तथा 'क्यों' और 'कैसे' आरंभ हुआ, आदि की जिज्ञासा को शांत करता है।

इस प्रकार **व्याकरण भाषाविज्ञान के लिए सामग्री प्रस्तुत करता है, जिसके अध्ययन-विश्लेषण के आधार पर वैज्ञानिक परिणाम और नियम निर्धारित करना भाषाविज्ञान का कार्य है;** साथ ही, व्याकरण को आगे बढाने का कार्य भी भाषाविज्ञान ही करता है; क्योंकि उसके द्वारा जीवित भाषा में परिवर्तन और विकारों के विश्लेषण के आधार पर व्याकरण को नत होकर आगे बढना पडता है। व्याकरण की सीमाएँ काल-विशेष के कटघरे में फँसी रह कर संकुचित रहती हैं, जबकि भाषाविज्ञान का संबंध काल-विशेष की सीमा से दूर जाकर कम से कम भूत और वर्तमान से रहने के कारण विस्तृत सीमा में परिभ्रमण करता है। व्याकरण का कार्य भाषा की आलोचना नियमों के आधार पर करना है जो कलात्मकता के निकट है। भाषाविज्ञान का कार्य नियम-निर्माण अर्थात् परिवर्तन की दिशा निर्धारित करना है जो भाषा के अध्ययन-विश्लेषण पर आधारित है। भाषाविज्ञान का कार्य वैज्ञानिक अधिक है। इस विवेचन के आधार पर हम दोनों के साम्य और वैषम्य (भेद) को संक्षेप में निम्न शब्दों में रख सकते हैं -

साम्य :

(क) दोनों के अध्ययन का आधआर भाषा का अध्ययन है।

(ख) दोनों के ऐतिहासिक, तुलनात्मक और समकालिक तीन भेद हो सकते हैं।

(ग) भाषा की उत्पत्ति,विकास, बनावट और विकृति का अध्ययन-विश्लेषण भाषाविज्ञान करता है, तो इसके निकट ही व्याकरण उसके शुद्ध स्वरूप और बनावट पर बल दे कर प्रकाश डालता है।

वैषम्य :

(क) भाषाविज्ञान का कार्य भाषा का वैज्ञानिक अध्ययन करना है, तो व्याकरण 'कैसा होना चाहिए' - शुद्ध बोलना, लिखना और समझना - का विश्लेषण करता है।

(ख) प्रथम का कार्य वैज्ञानिक है, तो दूसरे का कलात्मक।

(ग) एक का काम नियम-निर्धारण है, तो दूसरे का भाषा का परिष्करण और परिमार्जन।

(घ) भाषाविज्ञान का क्षेत्र कालातीत हो कर विस्तृत है, तो व्याकरण का संकुचित।

(ङ) व्याकरण सीधे नियम रख देता है। वह वर्णनात्मक है। भाषाविज्ञान निरीक्षण-परीक्षण करता है और कारण देता है।
(च) व्याकरण भाषाविज्ञान का अनुगामिनी है।
(छ) भाषाविज्ञान, भाषा की रचना, वाक्य-गठन, ध्वनि, अर्थ, शब्द-समूह और लिपि आदि का वैज्ञानिक अध्ययन प्रस्तुत करता है, जबकि व्याकरण का प्रमुख विषय भाषा की रचना और वाक्य-गठन ही माने जाते हैं।

(2) भाषाविज्ञान और भौतिक-शास्त्र :

मुँह से ध्वनि उत्पन्न और उसके कानों तक पहुँचने के मध्य ध्वनियाँ लहरों में चलती हैं जो आकाश या आकाश-लहरों के रूप में गमन करती हैं। इन लहरों के संबंध में पूरी जानकारी हमें भौतिक-शास्त्र से ही प्राप्त होती है। भाषावैज्ञानिक ध्वनियों से अन्य ध्वनियाँ किस प्रकार से भिन्न हैं, इसका उत्तर भी भौतिकविज्ञान देता है। आधुनिक युग में प्रयोगात्मक-ध्वनिशास्त्र (Experimental phonetics) के लिए भौतिकशास्त्र ने बडी सहायता की है। ध्वनियों - स्वर और व्यंजन - के तात्विक रूप पर भौतिकशास्त्र ने बड़ा प्रकाश डाला है। अब दो ध्वनियों के तात्विक अंतर का सूक्ष्मतम ज्ञान भी संभव है, जो प्रयोगों द्वारा संभव हुआ है। लैरिंगोस्कोप, काइमोग्राफ, कृत्रिम तालु, एक्स-रे आदि से ध्वनियों के अध्ययन में काफी सफलता मिली है। ध्वनिविज्ञान का पूर्ण वैज्ञानिक रूप उपलब्ध करना भौतिकी के द्वारा ही संभव हो रहा है।

(3) भाषाविज्ञान और शरीरविज्ञान :

भाषाविज्ञान में मुख से उच्चरित ध्वनियों द्वारा भाव या विचारों का अध्ययन किया जाता है। ध्वनि- उच्चारण में उच्चारण-अवयवों का मुख्य स्थान होता है। इन अवयवों की बनावट की स्थिति पर ध्वनियाँ निर्भर करती हैं। घोष-अघोष ध्वनियों के उच्चारण में अवयवों की स्थिति में पर्याप्त, किंतु सूक्ष्म अंतर हो जाता है, जिसका अध्ययन शरीरविज्ञान के माध्यम से अधिक सार्थक रूप में होता है। फिर, श्रवणेंद्रिय पर ध्वनि सुनना और उसका समझना अधिक निर्भर करता है और श्रवणेंद्रिय (कान) का ज्ञान शरीरविज्ञान ही स्पष्ट रूप में देता है।

(4) भाषाविज्ञान और तर्कशास्त्र :

भाषाविज्ञान की व्याख्यापेक्षता को तर्क से स्थायित्व मिलता है। यास्कमुनि द्वारा अपने अर्थविज्ञान-विषयक परिचित निरुक्त में तर्कप्रणाली का अपनाया जाना सर्वविदित है।

भाषाविज्ञान विज्ञान है, जिसका तर्कप्रधान होना भी एक गुण है। भारतीय तर्कशास्त्र वस्तुतः भाषा के चरम अवयव वाक्य का ही विज्ञान है। यह भाषा के सहारे ही चलता है। अतः उसे शब्दों, वाक्यों की उपयुक्तता को देख कर प्रयोग करना पड़ता है।[5]

(5) भाषाविज्ञान और कंप्यूटर :

कंप्यूटर भाषाविज्ञान को पूर्णतः विज्ञान बना देता है क्योंकि यह भाषा को ज्यों का त्यों ग्रहण करता है और उसमें मनुष्यजनित त्रुटियों की कोई गुंजाइश नहीं होती है। व्यक्ति स्वयं लिखते समय 'द्वारा' शब्द के सही क्रम में व्यतिक्रम कर के भी लिख सकता है और उसे सही मान लेता है क्योंकि भाषा की शुद्धता के प्रति व्यक्ति कामचलाऊ रवैया अपनाता है, जैसे 'सब चलता है'। लेकिन कंप्यूटर में यदि गलत इनपुट भरा जाएगा, तो गलत आउटपुट मिलेगा और सही इनपुट भरा जाएगा, तो सही आउटपुट मिलेगा। उदाहरण के लिए, 'द्वारा' शब्द हेतु यदि व+्+द+ा+र+ा टाइप करेंगे, तो आउटपुट होगा 'व्दारा'। अतः भाषाविज्ञान की दृष्टि से अध्ययन करने के लिए कंप्यूटर सबसे अधिक उपयुक्त तथा विश्वसनीय माध्यम है।

5.1.5 भाषिक संरचना और उसके विभिन्न स्तर (भाषाविज्ञान के अंग)

भाषा यादृच्छिक ध्वनि-प्रतीकों की संरचनात्मक व्यवस्था है। अर्थात् इस व्यवस्था की अपनी विशेष प्रकार की संरचना होती है। साथ ही इस संरचना में केवल एक स्तर नहीं होता। इसमें कई स्तर होते हैं। जैसे ध्वनि-स्तर, रूप-स्तर, वाक्य-स्तर, अर्थ-स्तर आदि। प्रत्येक स्तर पर भाषा की इकाइयाँ अलग-अलग होती हैं। जैसे ध्वनि इकाई ध्वनि-स्तर पर, तो वाक्य-स्तर पर वाक्य इकाई, या रूप-स्तर पर रूप इकाई। प्रत्येक स्तर की अपनी अलग संरचना होती है।

ध्वनि-स्तर की बात लें तो 'लिखना' और 'खिलना' दोनों ही में ल्, इ, ख्, न्, आ ध्वनियाँ हैं, किंतु इनकी आंतरिक संरचना अलग-अलग है, इसीलिए दोनों दो शब्द हैं तथा दोनों के दो अर्थ हैं।

'भाषिक संरचना के विभिन्न स्तर' के संबंध में विद्वानों में मतभेद है। हॉकिट पाँच स्तर मानते हैं : (1) व्याकरणिक, (2) स्वनिमिक, (3) रूपस्वनिमिक, (4) आर्थी, (5) स्वनिक। इनमें प्रथम तीन को उन्होंने केंद्रीय कहा है और अंतिम दो को परिधीय। पहली में रूप तथा वाक्य आते हैं, दूसरी में स्वनिम, तीसरी में पहली और दूसरी में संबंध-स्थापन होता है, चौथी का संबंध अर्थ से है। यहाँ वाक्य आदि आस-पास के भाषिक एवं संबद्ध भाषेतर संदर्भों से जुड़ते हैं। पाँचवी में स्वनों अर्थात् ध्वनियों का उच्चारण, प्रसरण तथा श्रवण आता है। हॉकिट 'भाषा-व्यवस्था' में इन पाँचों को 'उपव्यवस्था' का नाम देते हैं।

भाषा को ध्यान से देखें तो ध्वनियों से 'शब्द' बनते हैं, शब्दों (तथा धातुओं) से 'रूप', रूपों से 'वाक्य' और एकाधिक वाक्यों से 'प्रोक्ति'।

1) **अर्थ** - भाषा का मूलभूत काम है अर्थ की अभिव्यक्ति। वक्ता या लेखक का पूरा मंतव्य या अर्थ तो प्रोक्ति से व्यक्त होता है। यों प्रोक्ति के भीतर प्रत्येक वाक्य का, वाक्य के भीतर प्रत्येक रूप का, रूप के भीतर प्रत्येक शब्द एवं धातु का तथा कारक-चिंह और प्रत्यय का अपना अर्थ होता है।

2) **प्रोक्ति** - प्रोक्ति की संकल्पना भाषाविज्ञान में अपेक्षाकृत नई है। यों प्राचीन भारत में 'महावाक्य' द्वारा इसी संकल्पना को द्योतित किया गया है। वस्तुत: भाषा का प्रयोग किसी मंतव्य को अभिव्यक्ति देने के लिए होता है, और मंतव्य को अभिव्यक्ति देने के लिए एकाधिक वाक्यों का प्रयोग करना पडता है। **एकाधिक वाक्यों के उस समुच्चय को ही प्रोक्ति कहते हैं जो एक सुव्यवस्थित इकाई के रूप में वक्ता या लेखक के मंतव्य को अभिव्यक्ति दें।**

3) **वाक्य** - अब तक भाषाविज्ञान तथा व्याकरण में भाषा की चरम इकाई तथा सहज इकाई वाक्य माना जाता रहा है। अब प्रोक्ति को चरम और सहज इकाई मान लेने पर वाक्य प्रोक्ति के भीतर की एक इकाई माना जाने लगी है।

4) **रूप** - वाक्य रूपों से बनता है या वाक्य को विश्लेषित करने पर रूप मिलते हैं। 'रूप' को ही 'पद' भी कहते हैं। रूप में 'शब्द' तथा 'धातु' रूप में अर्थतत्व होते हैं तथा 'कारक-चिंह' और 'प्रत्यय' रूप में संबंध तत्व। 'राम ने रावण को मारा' वाक्य में 'राम ने,' 'रावण को' तथा 'मारा' तीन रूप हैं, जिनमें 'राम,' 'रावण', 'मार्' अर्थतत्व हैं और 'ने', 'को', 'आ' संबंध तत्व।

5) **शब्द** - सामान्यत: शब्द को भाषा की सर्वाधिक महत्वपूर्ण इकाई माना गया है, क्योंकि भाषा में भाव या विचार की अभिव्यक्ति मूलत: शब्द से ही होती है। शब्द में ही प्रत्ययादि जोड कर रूप बनते हैं और रूप से वाक्य और वाक्यों से प्रोक्ति। शब्द में धातु भी समाहित है। उपसर्ग, कारक-चिंह एवं प्रत्यय को भी उसी में 'बद्धशब्द' (जो अकेले न आकर किसी शब्द या धातु के साथ प्रयुक्त हो) या 'बद्धतत्व' रूप में शब्द के अंतर्गत ही रखा जा सकता है।

6) **ध्वनि** - ध्वनियाँ अलग से सार्थक नहीं होतीं, किंतु ये आपस में मिलकर सार्थक शब्द, रूप, वाक्य तथा प्रोक्ति का निर्माण करती हैं। ध्वनि-स्तर में किसी भाषा की विभिन्न ध्वनियों, उनकी स्वनिम-व्यवस्था, आक्षरिक संरचना, बलाघात, अनुतान आदि का अध्ययन होता है।

7) **रूपस्वनिम स्तर** - यह रूप और ध्वनि स्तरों को जोडने वाला स्तर है, अत: इसे अलग स्तर न मानकर दोनों की संधि माना जा सकता है। इसका क्षेत्र है शब्दों, रूपों, उपसर्गों तथा प्रत्ययों आदि के योग से होने वाले ध्वनि-परिवर्तन तथा परिवर्तन के बाद प्राप्त नए शब्द या रूप। उदाहरण के लिए घोडा + दौड = घुडदौड, पानी + घाट = पनघट। संस्कृत में संधि के अंतर्गत इन्हीं का अध्ययन होता रहा है।

8) **लेखन (लिपि) संरचना** - भाषा लिखी भी जाती है और लिखने में प्रत्येक लिपि की अपनी आंतरिक व्यवस्था और संरचना होती है। नागरी में 'लिपि' शब्द के लेखन में 'इ' के दोनों चिंह ल् तथा प् व्यंजन के पहले आएँगे, किंतु रोमन में LIPI रूप में L तथा P के बाद में।

प्रोक्ति, वाक्य, रूप, शब्द, ध्वनि, अर्थ, रूपस्वनिम ये सात तो भाषा के केंद्रीय स्तर हैं, किंतु लेखन केंद्रीय स्तर के अंतर्गत तो नहीं ही है, परिधीय भी न होकर, इन सबसे अलग है, किंतु है भाषा से पूरी तरह संबद्ध।[6]

5.2 प्राकृतिक भाषा तथा कंप्यूटर कुंजी-पटल (की-बोर्ड)

5.2.1 कुंजी-पटल की कार्य-प्रणाली

क) कुंजी-पटल का प्रकार
ख) कुंजियों की आवाज
ग) नए कुजी-पटल
घ) कुंजी-पटल की मेमोरी
ङ) कुंजी-पटल की समस्याएँ तथा समाधान

5.2.2 कुंजियों का विवरण

5.2.3 विभिन्न कुंजी-पटलों का अध्ययन

क) देवनागरी तथा अँग्रेजी, रेमिंग्टन टाइपराइटर, ध्वन्यात्मक
ख) क्वर्टी तथा डीवोरक
ग) मानक फोंट कूट
- कुंजीपटल की सीमाएँ
- फोंट का प्रतिबल/तनाव
- कुंजी संकल्पना
- नई कूट सत्ताएँ
(i) इनस्क्रिप्ट,(ii) ऑकी,(iii) इस्का,(iv) इस्फोक

5.2.4 प्राकृतिक भाषा तथा कुंजीपटल का भाषावैज्ञानिक अध्ययन

क) कंप्यूटर युग में देवनागरी लिपि की सार्थकता तथा गुण
- वर्णमाला का वर्गीकरण
- लिपिचिंहों के नाम ध्वनि के अनुरूप
- एक ध्वनि के लिए एक लिपि-चिंह
- लिपि-चिंहों की पर्याप्तता
- ह्रस्व तथा दीर्घ स्वर के लिए स्वतंत्र चिंह
- मात्राओं का प्रयोग
- नागरी के व्यंजन-चिंहों की आक्षरिकता
- नागरी के व्यंजन-चिंहों की आक्षरिकता
- सुपाठ्यता

ख) मानक हिंदी वर्तनी
ग) कुंजीपटल तथा कुंजियों का भाषावैज्ञानिक विश्लेषण
घ) प्रदर्श 1-6

5.2 मानव-भाषा तथा कंप्यूटर कुंजी-पटल (की-बोर्ड)

5.2.1 कुंजी-पटल की कार्य-प्रणाली

जब आपके पोर्टल और ब्रिज आपके कंप्यूटर के साथ संवाद स्थापित करते हैं तो की-बोर्ड ही एक ऐसा साधन है जिसके बिना यह कार्य नहीं हो सकता। की-बोर्ड की कुँजियाँ और कुछ नहीं, बस बटनों/ स्विचों की एक श्रृंखला है जिन्हें जब भी दबाया जाएगा एक विशेष प्रकार का बाइनरी कोड कंप्यूटर तक भेज दिया जाएगा। कंप्यूटर के अंदर इस कोड को संबंधित अक्षर या आदेश में परिवर्तित कर दिया जाता है। इसका प्रभाव स्क्रीन पर एक अक्षर या क्रिया रूप में देखा जा सकता है।

5.2.1 (क) कुंजी-पटल का प्रकार

विभिन्न प्रकार के की-बोर्डों का अंतर उनके 'की' में दिखाई देता है। ये मेकेनिकल, मेंब्रेन, केपेसिटिव तथा कंडक्शूसिव प्रकार के होते हैं।

मेकेनिकल स्विच वाले की-बोर्ड

सबसे आसानी से और सबसे ज्यादा उपलब्ध की-बोर्डों में से एक है। इस प्रकार के की-बोर्ड में कुंजियाँ एक प्लंगर पर लगी होती है जिसे दबाया जा सकता है। प्लंगर को घेरे रखने वाली स्प्रिंग द्वारा लगाया जाने वाला वेग, 'की' को वापस ऊपर की ओर धकेलता है, जैसा कि पुराने टाइपराइटरों में होता था। जैसे ही 'की' दबाई जाती है प्लंगर दो पत्तियों को एक दूसरे से विपरीत दिशा में धकेलता है। जैसे ही 'की' छोडी जाती है ये पत्तियाँ वापस एक दूसरे से जुड जाती हैं और सर्किट पूरा हो जाता है। परिमामस्वरूप एक विशेष विद्युत धारा प्रवाहित की जाती है जिसे की-बोर्ड कंट्रोलर के नाम से जाने जाने वाली एक विशेष चिप अनुभव करती है और इसके अनुरूप बाइनरी मूल्य कंप्यूटर को भेज दिया जाता है। हालाँकि मेकेनिकल बटनों वाले की-बोर्ड बनाने में कठिन होते हैं पर उपयोगकर्ता को शीघ्र जवाब देने जैसा महसूस करवाते हैं, इसलिए पसंद किए जाते हैं।

मेंब्रेन (Membrane) स्विच वाले की-बोर्ड

इस प्रकार के की-बोर्डों में प्रत्येक 'की' के स्विचों के नीचे की ओर झिल्लीदार स्विच होते हैं। जैसे ही कोई 'की' दबाई जाती है, प्लंगर रबर के अर्द्ध-वृत बनाता है जिससे धातु के दो छोर आपस में संपर्क में आते हैं, इससे विद्युत सर्किट पूरा होता है और की-बोर्ड कंट्रोलर इस विद्युत प्रवाह को समझ पाता है। जब 'की' वापस ऊपर आती है तो रबर के गुंबद अपने पुराने आकार में आ जाते हैं, सर्किट टूट जाता है। अपने रबर जैसे कुछ हिस्सों के कारण ये की-बोर्ड मेकेनिकल की-बोर्ड का सा प्रभाव उत्पन्न नहीं कर पाते हैं। कुछ मेंब्रेन की-बोर्डों में प्लंगर और स्प्रिंग काम मे ली जाती है जिससे वे पारंपरिक की-बोर्ड जैसा अहसास देते हैं। कुछ पीसी जो इन की-बोर्डों को प्रयोग में लेते हैं, क्लिक की आवाज

कृत्रिम तौर पर बनाते हैं जो पीसी स्पीकर में भी सुनाई देती है और मेकेनिकल की-बोर्ड जैसा भान देते हैं।

इन प्रिंटरों में उपलब्ध सस्ती क्वालिटी के प्रिंटर कभी भी फेल हो सकते हैं, इसलिए लंबे समय के लिए नहीं हैं।

केपेसिटिव (Capacitive) स्विच वाले की-बोर्ड

ये की-बोर्ड विद्युत धारा में परिवर्तन को जाँच उस पर कार्य करते हैं। एक नॉन-कंडक्टिंग पैड प्रत्येक 'की' के नीचे प्लंगर से जुड़ा होता है। जब एक 'की' को दबाया जाता है तो प्लंगर के कारण यह पैड ऐसे दो विपरीत सर्किटों के संपर्क में आता है जिनके बीच थोड़ा सा अंतराल होता है। जब नॉन-कंडक्टिंग प्लेट दो कंडक्टर को छूती है, विद्युत प्रवाह बदल जाता है। जब आप 'की' को छोड़ते हैं, इन प्लेटों के बीच का विद्युत प्रवाह वापस अपनी पुरानी अवस्था में आ जाता है।

कंडक्टिव (Conductive) स्विच वाले की-बोर्ड

ये की-बोर्ड आसानी से पहचाने जा सकते हैं। इनमें अपनी प्रत्येक 'की' के लिए रबर की एक छोटी सी बॉल होती है। हर रबर डॉट के अंदर पतली कार्बन की परत होती है। जब 'की' को दबाया जाता है तो यह 'की' के नीचे लगे धातु के दो सिरों में विद्युतीय संबंध बनाती है।

बहुत से लोगों को इन्हें प्रयोग में लेना बोझिल लगता है। इसलिए इन्हें लगातार लंबे समय तक प्रयोग में नहीं ले पाते। इस प्रकार के की-बोर्ड पर कोई तरल पदार्थ गिरने से कोई फर्क नहीं पड़ता है, ये सार्वजनिक स्थानों पर लगे कंप्यूटरों में ज्यादा अच्छे होते हैं।

मास्टर-की

एक अत्यंत महत्वपूर्ण उपकरण की-बोर्ड मे होता है जिसे **की-बोर्ड कंट्रोलर** कहा जाता है। मल्टी-प्लेक्सिंग नामक प्रक्रिया का उपयोग कर यह चिप 'की' स्थिति को स्कैन करता है और उसके बाद कंप्यूटर को सही संकेत भेजता है। ये 'की' को इनकी रो (Row) या कॉलम (Column) तथा इनके द्वारा जारी किए गए सिग्नल्स की फ्रीक्वेंसी (आवृति) के अनुसार स्कैन किया जाता है। इन कुँजियों के नीचे लगे स्विचों को एक ग्रिड (पंक्तियों और कॉलम को समूह) में बाँटा जाता है और प्रत्येक 'की' को एक नया नंबर दिया जाता है।

जब एक 'की' पर दबाव पड़ता है, तो उसे उसके रो और कॉलम के अनुसार पहचाना जाता है तथा उसी के अनुसार फ्रीक्वेंसी प्रदान की जाती है। परिणामस्वरूप मिलने वाले सिग्नलों में से पहले सिग्नल को जो एक विशेष प्रकार का सिग्नल होता है, प्रोसेसर के सभी कार्य रोक कर कंप्यूटर तक पहुँचाया जाता है। की-बोर्ड द्वारा भेजे जाने वाले बाइनरी मूल्य को आधार मान कर विशिष्ट क्रिया करी जाती है।

क्वर्टी (QWERTY) या डीवोरक (DVORAK)

प्रारूप के अनुसार की-बोर्ड को दो श्रेणियों में वर्गीकृत किया जाता है। सबसे प्रसिद्ध लेआउट है क्वर्टी (QWERTY ,की-बोर्ड की प्रथम पंक्ति की सात 'की') टाइप, और बहुत कम जाना पहचाना प्रारूप डीवोरक (DVORAK)। माना जाता है कि डीवोरक टाइप की-बोर्ड के द्वारा कोई भी व्यक्ति दोनों हाथों की उँगलियों के काम में आने के कारण अधिक शीघ्रता से कार्य करता है। इस प्रकार के की-बोर्ड में सभी स्वरों (Vowels) को एक ओर तथा सभी व्यंजनों को दूसरी ओर रखा जाता है जो तेज टाइप करने में सहायक है।

एरगोनोमिक (Ergonomic) की-बोर्ड

एरगोनोमिक की-बोर्ड इस तरह से डिजाइन किए गए हैं कि इन पर कार्य करते समय हाथ एक आरामदायक मुद्रा में रहते हैं और लंबे समय तक टाइप करने में आसानी होती है। इनकी यह विशेषता RSI (Repetitive Stress Injury) जैसी परेशानियों से दूर रखती है। उदाहरण के लिए जैसे माइक्रोसॉफ्ट नैचुरल की-बोर्ड।

वायरलैस की-बोर्ड

नए प्रकार के की-बोर्ड तारों के जरिए कंप्यूटर के अंदर डाटा पहुंचाने के लिए बाध्य नहीं हैं। यह कॉर्ड-लैस की-बोर्ड रेडियो फ्रीक्वेंसी जैसे उपायों का प्रयोग कर के की-बोर्ड सिग्नल को कंप्यूटर में ट्रांसफर करते हैं। यह की-बोर्ड की-पैड तथा एफ एम और ए एम ट्रांसमीटर या इंफ्रारेड (अवरक्त) LED हार्डवेयर रूप में ट्रांसमिट करने के लिए हार्डवेयर होते हैं। इनमें कंप्यूटर और की-बोर्ड के बीच कोई फिजीकल लिंक (भौतिक संपर्क) नहीं होता, इनका स्वयं का ऊर्जा स्रोत होता है जो सामान्यतया सूखी या दोबारा चार्ज हो सकने वाली बैटरियों के रूप में होता है। कंप्यूटर में की-बोर्ड से मिलने वाली फ्रीक्वेंसी को लेने के लिए पोर्ट पर एक इंफ्रारेड सेंसर या रेडियो रिसीवर होता है। इस प्रकार के की-बोर्ड प्रेजेंटेशन आदि में, जहाँ कई बार उपयोगकर्ता को दूरी की आवश्यकता होती है, अच्छे होते हैं, पर ये सामान्यत: महँगे होते हैं।[7]

5.2.1 (ख) बेजुबान की-बोर्ड की आवाज

कंप्यूटर के ज्यादातर यॉंत्रिक इस्तेमाल के लिए की-बोर्ड प्राथमिक इनपुट उपकरण होता है। चाहे आपको कंप्यूटर सॉफ्टवेयर से संपर्क बनाना हो, पत्र लिखना या कोई प्रोग्राम बनाना हो, की-बोर्ड के बगैर काम संभव नहीं है।

पिछले कुछ सालों में की-बोर्ड की बनावट में काफी सुधार आया है। कुछ कंप्यूटरों के साथ पहले सस्ते की-बोर्ड होते थे, जिस पर टाइप करने से न सिर्फ आपकी उँगलियों में दर्द होता था बल्कि उनमें कोई आवाज नहीं होती थी। यानी कि **कोई बटन दबा या नहीं आपको पता नहीं चल पाता था।** इस समस्या का एक इलाज तो यह है कि आप बाजार में मिलने वाले महँगे की-बोर्ड ले आएँ। लेकिन आप **खुद ही की-बोर्ड की प्रोग्रामिंग के जरिए इन्हें हल्की क्लिक की आवाज दे सकते हैं।**

एक अच्छे की-बोर्ड में हरेक बटन को दबाने पर एक यॉंत्रिक क्लिक की आवाज पैदा होती है, जबकि एक सस्ते की-बोर्ड में **बटन को दबाने पर उसमें लगा छोटा और शांत स्प्रिंग**

आपकी उँगलियों पर वापस दबाव डालता है और उसमें कोई आवाज भी नहीं होती। पर आप **KEYCLICK** से इस आवाज को पैदा कर सकते हैं।

KEYCLICK आपके कंप्यूटर के मेमोरी में बैठ कर आपके की-बोर्ड पर नजर रखता है। **हरेक बार जब आप कोई कुंजी (Key) या बटन दबाते हैं तो यह पीसी के स्पीकर को बहुत तेजी से ऑन और ऑफ करता है, जिससे क्लिक की आवाज उत्पन्न होती है।** KEYCLICK का इस्तेमाल करने से पहले इसका निर्माण करना होगा। इसके लिए **डिबग स्क्रिप्ट से KEYCLICK.SCR प्रोग्राम** बनाना होगा।

इसे डॉस कमांड लाइन पर लोड करने के लिए KEYCLICK टाइप करें। या फिर पूरे डॉस सत्र के दौरान की-क्लिक (KEYCLICK) को चालू रखने के लिए कमांड लाइन को AUTOEXEC.BAT फाइल से जोड़ें। ऐसा करते ही की-क्लिक (KEYCLICK) मेमोरी में लोड हो जाता है और तुरंत की-स्ट्रोक के लिए क्लिक की आवाज पैदा करने लगता है। क्लिक की आवाज बनाने के लिए यह पी.सी.स्पीकर में कमांड भेज कर उसे तेजी से पूरा ऑन और ऑफ करता है। ऑन और ऑफ के बीच के समय से ही यह तय होता है कि क्लिक की आवाज कितनी ऊँची है।

अगर आपको कीक्लिक (KEYCLICK) की आवाज पसंद नहीं है तो इसे बदल सकते हैं। KEYCLICK.SCR फाइल के 30वीं पंक्ति में क्लिक-डिले (CLICK DELAY) वर्णित है। यहाँ MOVCX,0258 का निर्देश दिया हुआ है। आवाज को बदलने के लिए केवल टेक्स्ट एडीटर के इस्तेमाल से इस संख्या में बदलाव होगा। क्लिक को धीमा करने के लिए संख्या को कम कर दें। आवाज को तेज करने के लिए संख्या को बढ़ा दें। बेहतर नतीजे के लिए इस संख्या को 1 और 1000 के बीच रखें।

हरेक बार जब आप बटन को दबाते और छोड़ते हैं, तो की-बोर्ड डेटा का पूरा समूह की-बोर्ड कंट्रोलर को भेजता है। कंट्रोलर तय करता है कि डेटा बिना किसी गड़बड़ी के प्राप्त किया जाए और फिर वह की-बोर्ड डेटा को एक विशेष 8-बिट के स्कैन कोड (SCAN CODE) में बदल देता है। इसके बाद कंट्रोलर हार्डवेयर इंटरप्ट (HARDWARE INTERRUPT) नाम का सिग्नल भेजता है। इंटरप्ट -9 सिस्टम के माइक्रो-प्रोसेसर पर अपना काम रोकने का दबाव डालता है और उसे तुरंत दबाए गए बटन (KEYSTROKES) को प्रोसेस करने के लिए मेमोरी में एक सबरूटीन (SUBROUTINE) एक्जीक्यूट करने को कहता है। ऐसा करने पर बायोस (BIOS) का रूटीन इंटरप्ट-9 के लिए की-बोर्ड कंट्रोलर से स्कैन कोड को प्राप्त करता है। फिर **इंटरप्ट-9 CAPS LOCK और NUM LOCK की स्थिति की जाँच करता है और यह तय करता है कि SHIFT,CTRL,ALT बटन दबाए गए हैं या नहीं क्योंकि ये सब बटन दबने पर क्लिक की आवाज नहीं होगी।** यह सब तय करने के बाद ही इंटरप्ट-9 स्कैन कोड को ऑस्की (ASCII) कोड में बदलता है, जिसके बाद कीक्लिक (KEYCLICK) से आवाज निकलती है।

5.2.1 (ग) नए कुंजी-पटल

कंप्यूटर का की-बोर्ड पीसी का एक अतिरिक्त और साधारण-सा हिस्सा नहीं है,बल्कि यह पूरी तरह व्यक्ति की सेहत, काम करने की आदत से संबंध रखता है। आज जितने भी की-बोर्ड बनाए जा रहे हैं, उनमें सबसे ज्यादा ध्यान कर्मचारियों व व्यक्तियों के काम के माहौल का रखा जाता है। नए-से-नए डिजाइन के की-बोर्ड बनाने का मकसद यही है कि आराम से काम हो सके और शरीर की मांसपेशियों पर न्यूनतम तनाव पड़े। की-बोर्डों के विकास पर कई शोध हुए हैं। बाजार में अब नए मॉडल के जितने भी की-बोर्ड आ रहे हैं, उनमें सुविधा और उत्पादकता का खासतौर से ध्यान रखा गया है। इस समय जो की-बोर्ड आ रहे हैं उनमें स्प्लिट (कई हिस्सों में बँटा हुआ), वर्टिकल, कॉर्डिंग आदि हैं।

की-बोर्ड के अब जो नए कंसेप्ट हैं,उनमें ज्यादा से ज्यादा कोशिश शरीर को आरामदायक बने रहने देने की है। जब आप की-बोर्ड का उपयोग करते हैं, तो उँगलियों के अलावा गर्दन और कंधे पर भी इसका काफी असर पड़ता है। **ऑप्टिकल स्प्लिट** में इस बात का ध्यान रखा गया है कि आपकी भुजाएँ सामने की ओर सीधी रहें, ताकि गर्दन व कंधों पर तनाव न पड़े। जबकि **ऑप्टिकल लेटेरल रिल्ट** में अँगूठों को उठाने की सुविधा पर गौर किया गया है, साथ ही पैडयुक्त पाम सपोर्ट भी है जो गति बनाए रखता है।

विभिन्न प्रकार की आवश्यकताओं को ध्यान में रख कर बनाए गए की-बोर्डों का बाजार भी छोटा नहीं है। उदाहरण के लिए, मेलट्रॉन ने **विकलांगों के लिए** विशेष प्रकार के की-बोर्ड बनाए हैं। इनमें सिंगल हेंडेड की-बोर्ड, सिंगल फिंगर और सिंगल हैड, माउथस्टिक और एक्सटेंडेड (विस्तारित) की-बोर्ड हैं, जो शारीरिक रूप से विकलांग लोगों के लिए तैयार किए गए हैं। दृष्टिहीन व्यक्ति भी इन की-बोर्डों का उपयोग कर सकते हैं। इन सबकी खूबी,इनमें खास जगह की वे कुंजियाँ होती हैं जो उपयोगकर्ता को तनाव से एकदम मुक्त रखती हैं। इस बारे में और ज्यादा जानकारी www.maltron.com साइट से ली जा सकती है।

कुछ की-बोर्ड ऐसे बनाए गए हैं, जिनका **उपयोग चलते हुए, या बाहर किसी भी परिस्थिति में** किया जा सकता है। अगर कहीं यात्रा पर जा रहे हों तो इन्हें साथ ले जा सकते हैं। इस तरह के की-बोर्ड यूनिमिसेंट सिस्टम इंकारपोरेशन ने तैयार किए हैं। ये की-बोर्ड 'बिल्डिंग ब्लॉक' कंसेप्ट पर आधारित हैं, इसलिए आप अपनी आवश्यकता के अनुरूप इसका चुनाव कर सकते हैं। इस बारे में ज्यादा जानकारी www./umsys.com से ली जा सकती है।

उद्योगों, प्रयोगशालाओं और दफ्तरों के लिए की-बोर्ड अलग तरह के हैं।देखने में ये इस तरह के लगते हैं कि एक में ही सब कुछ बना दिया हो। इनकी खूबी यह है कि ये ज्यादातर डिटरजेंटों, रोगाणुनाशकों और स्टटलाइजरों को बर्दाश्त कर सकते हैं, यानी उनसे खराब नहीं होते। इनकी कुंजियों (की) पर जो अक्षर लिखे होते हैं,वे भी आसानी से नहीं मिटते। इस प्रकार ये उच्च तकनीक वाले इंफ्रा-रेड विंडोज-95 के अनुरूप की-बोर्ड हैं। लेकिन यदि आपको ऐसा की-बोर्ड चाहिए, जो बहुउपयोगी हो तो फोकस इलेक्ट्रॉनिक्स का एफके-7200 लीजिए। ताइपेई की इस कंपनी का दावा है कि **ईजेडबाल (EzBall)**

पहली की-बोर्ड तकनीक है जो विस्तारित स्पेस बार के बीच स्थित 25mm ट्रैकबाल को इंटिग्रेट करती है और इससे उपयोगकर्ता ट्रैकबाल व की-बोर्ड को एक ही समय में एक साथ काम में ले सकता है।

यदि आप बहुत ही व्यस्त हैं और ऐसा की-बोर्ड चाहते हैं कि उससे स्कैन कर लें, टाइप कर लें, व्यवस्थित रूप से जमा दें और सूचना भेज दे, तो फिर **9600 w स्कैनर की-बोर्ड** लें। यह डिवाइस एक की-बोर्ड, एक स्कैनर से ज्यादा बढ़ कर है। इसमें पेपर मैनेजमेंट सिस्टम अलग से है, अत्याधुनिक तकनीक वाला की-बोर्ड है। आप स्कैनर की-बोर्ड में पेपर फीड करें, यह अपने आप चालू हो जाएगा, सॉफ्टवेयर अपना काम शुरु कर देंगे। इस बारे में विस्तार से जानकारी के लिए www.nmbtech.com साइट देखें। यह हर तरह से सुरक्षित और सभी तरह के कंप्यूटरों पर चलने वाला की-बोर्ड है। खास बात यह है कि बाजुओं, कलाई, पीठ, गर्दन सभी को पूरी तरह से सपोर्ट करता है।

एल्प्स इलेक्ट्रिक ने भी ग्लाइड पॉइंट जैसा उत्कृष्ट की-बोर्ड बनाया है। इसकी विशेषता यह है कि यह वजन में काफी हल्का और उचित पैडसाइज की-बोर्ड /माउस यूनिट से युक्त है। इसकी एंटर और टैब कुंजियाँ हैं, वे काफी अच्छी बनी हुई हैं, सैपेरेट नेविगेशन है, कर्सर कंट्रोल-की है और न्यूमेरिक-की पैड भी। कर्सर की स्थिति के लिए इसमें अलग से एक संपूर्ण सॉफ्टवेयर है, जिसे ग्लाइड पॉइंट पॉइंटिंग डिवाइस कहते हैं। इस की-बोर्ड में कलाई पैड भी बनाया गया है और कई केबल एडैप्टर भी हैं जिनका जरूरत के हिसाब से उपयोग किया जा सकता है। एल्प्स के इस की-बोर्ड की विशिष्ट खूबी यह है कि आप कितनी ही तेज टाइपिंग करें, शोर नहीं होगा। इसलिए ऐसी जगहों पर जहाँ सभा, मीटिंग हो रही हो, इसका उपयोग आसानी से किया जा सकता है।

इसका जो **इंटिग्रेटेड ग्लाइड पॉइंट है, वह माउस का एक पूर्ण विकल्प है**। फिर ग्लाइड पॉइंट की जो सबसे बड़ी खूबी इसे दूसरे की-बोर्डों से अलग करती है, वह यह है कि इसका डिजाइन काफी प्रेरणादायक है। यह की-बोर्ड दो भागों में बँटा है - आधा दाईं ओर, बाकी का बाईं ओर। बाएँ हिस्से में नई किस्म की कुंजियाँ हैं, जिन्हें ERASE-EASE कहा जाता है। यह दूसरे की-बोर्डों की बैक-स्पेस कुंजी की डुप्लीकेट है। बैक-स्पेस कुंजी इसकी ऐसी खूबी है, जिसकी वजह से आपको लगेगा कि पहले यह की-बोर्ड उपयोग में क्यों नहीं लिया। एल्प्स ने बाद में इस की-बोर्ड में कई ऐसे सुधार किए जो विंडोज सिस्टम के लिए काफी उपयोगी साबित हुए। जैसे - माइक्रोसॉफ्ट विंडोज लोगों के लिए अलग से ध्यान रखा गया। यह विंडोज-95 के स्टार्ट बटन की डुप्लीकेट थी। इसके अलावा राइट बटन माउस क्लिक मिनिएचर पॉपअप मेनू का डुप्लीकेट है। विंडोज-3.1 और इसके बाद के वर्जनों में यह ऑप्शनों का मेनू प्रदान करती है। ये वे विंडोज ऑब्जेक्ट हैं जिन पर आप काम कर रहे हैं। एल्प्स के इस की-बोर्ड के साथ जो सॉफ्टवेयर आते हैं, उनका उपयोग काफी आसान है। किसी भी कुंजी को रिप्रोग्राम किया जा सकता है। ग्लाइड पॉइंट माउस की गति को नियंत्रित कर सकते हैं। इस की-बोर्ड की खासियत यह है कि इसे किसी भी लैपटॉप में आसानी से लगा सकते हैं। जिस केबल एडैप्टर का उपयोग करना हो, कर सकते हैं। हालांकि यह की-बोर्ड कोई बहुत ज्यादा दक्ष नहीं है, लेकिन पूर्ण रूप से विश्वसनीय और पोर्टेबल है।[9]

5.2.1 (घ) कुंजी-पटल की मेमोरी

जब आप कंप्यूटर ऑन करते हैं तो डॉस प्रांप्ट आने से पूर्व ही आप टाइप करना शुरु कर देते हैं। लेकिन प्रांप्ट के दर्शित होने से पूर्व ही जो कुछ आपने टाइप किया था वह स्क्रीन पर कैसे आ जाता है, जबकि कंप्यूटर भी पूर्ण रूप से टाइप किए जाने के पश्चात ही बूट हुआ है।

वास्तव में कंप्यूटर में विविध डिवाइसों के लिए अलग-अलग बफर (अस्थाई मेमोरी) होते हैं। इसी प्रकार की-बोर्ड का भी एक बफर होता है। जब आप टाइप कर रहे थे उस वक्त टाइप किया गया टैक्स्ट वास्तव में की-बोर्ड बफर में स्टोर हो रहा था। जब कंप्यूटर बूट हो गया तो वह टैक्स्ट बफर से संबंधित एप्लीकेशन ले लेती है।

इस प्रकार बफर के प्रयोग से समय की बचत करते हुए सीपीयू के मूल्यवान समय को अन्य कार्यों में प्रयोग किया जा सकता है।[10]

5.2.1 (ङ) कुंजीपटल से संबंधित समस्याएँ तथा समाधान

कुंजीपटल से संबंधित प्रमुख समस्याएँ हैं -

1. कभी-कभी ऐसा होता है कि कंप्यूटर को ऑन करने के बाद मॉनीटर पर यह मैसेज दिखाई देता है। यह मैसेज इस कारण से आता है -
हमने सीपीयू में दिए हुए कुंजीपटल लॉक स्विच से कुंजीपटल को लॉक कर दिया है, ऐसी स्थिति में हमें की से लॉक को खोल कर कुंजीपटल को अनलॉक कर लेना चाहिए।

2. जब मॉनीटर पर यह मैसेज दिखाई दे, तो इसका अर्थ है कि की-बोर्ड सीपीयू से जुड़ा हुआ नहीं है। ऐसी स्थिति में हमें सीपीयू को ऑफ करके उसमें कुंजीपटल को जोड़ कर चालू करना चाहिए। यदि ऑन करने पर भी हमें यही मैसेज प्राप्त हो, तो इसके ये जिम्मेवार कारण हो सकते हैं -

- हमारा कुंजीपटल सीपीयू के साथ अनुरूप नहीं है अर्थात् सीपीयू हम एटी का प्रयोग कर रहे हैं और कुंजीपटल एक्सटी मॉडल का है।

- यदि हमारा कुंजीपटल, एक्सटी और एटी दोनों मोड में कार्य करता है, तो यह हो सकता है कि मोड बदलने की स्विच की स्थिति प्रयोग किए जा रहे हैं, जो सीपीयू के विपरीत है। अतः ऐसी स्थिति में हमें स्विच की स्थिति को ठीक कर लेना चाहिए। सामान्य रूप से यह स्विच कुंजीपटल के पीछे लगा होता है।

- किसी कारणवश कभी-कभी कुंजीपटल की कोई कुंजी स्थाई रूप से दबी रह जाती है, तो इसके परिणामस्वरूप भी उपरोक्त वर्णित संदेश प्राप्त होता है। इस समस्या के समाधान हेतु हमें की-बोर्ड को सीपीयू से अलग करके उसे ठीक प्रकार से साफ कर लेना चाहिए, जिससे कि भविष्य में ऐसे संदेश की पुनरावृत्ति न हो।

- किन्हीं विशेष परिस्थितियों में ऐसा भी होता है कि कंप्यूटर को ऑन करते ही बड़ी तेज गति से विप की ध्वनि सुनाई पडती है। इसका कारण यह भी हो सकता है कि की-बोर्ड की कोई कुंजी स्थाई रूप से दबी हुई है। यदि दबी हुई कुंजी वापस निकलकर सामान्य रूप से कार्य न करें, तो किसी हार्डवेयर इंजीनियर से उसको बदलवा लेना चाहिए।

3. कभी-कभी ऐसा होता है कि काम करते समय हम कुंजी को एक बार दबाते हैं लेकिन वह कुंजी को एक बार दबाते हैं लेकिन वह कुंजी अपने से संबंधित अक्षर को निम्न प्रकार से कई बार लिख देती है - vvvvvvbbbbbbnnnnnn । उपरोक्त वर्णित स्थिति मे भी यही करना चाहिए कि खराब कुंजी को चुनकर बदलवा देना चाहिए।

4. यदि हम माइक्रोसॉफ्ट विंडो के तहत कार्य कर रहे हैं, और हमें ऐसा प्रतीत होता है कि हमारा की-बोर्ड धीमी गति से कार्य कर रहा है, तो विंडो के कंट्रोल पैनल में दिए हुए की-बोर्ड नामक प्रतीक चिंह को प्रयोग कर सकते हैं। जब माउस से इस आइकॉन पर क्लिक करते हैं तो मॉनीटर पर इसका मीनू दिखाई देता है, इस मीनू में दिए हुए ऑप्शन का प्रयोग करके हम की-बोर्ड की गति अपनी आवश्यकता के अनुसार कम या ज्यादा तय कर सकते हैं।

5. यदि हमें अपने की-बोर्ड से इंग्लिश के अतिरिक्त किसी अन्य भाषा को लिखना है, तो कंट्री कोड को कंफिग.सिस फाइल में लिखकर यह कार्य कर सकते हैं।

सावधानियाँ

की-बोर्ड को प्रयोग करते समय हमें ये सावधानियाँ रखनी चाहिए -

1. की-बोर्ड को सीपीयू से ज्यादा दूर नहीं खींचना चाहिए।
2. यदि की-बोर्ड को सीपीयू से अलग रखना है तो इस बात का विशेष ध्यान रखना चाहिए कि सीपीयू ऑफ हो।
3. नियमित रूप से की-बोर्ड को साफ करना चाहिए।[11]

कुंजीपटल (की-बोर्ड) की सफाई

की-बोर्ड की सफाई करने के लिए निम्न वस्तुओं की आवश्यकता होगी।

- वैक्यूम क्लीनर
- ग्लास क्लीनर
- रुई

की-बोर्ड साफ करने से पूर्व अपना कंप्यूटर ऑफ कर दें। यदि आपका की-बोर्ड अधिक गंदा नहीं है तो आप इसकी कुंजियों के नीचे से कचरा बाहर निकालने के लिए वैक्यूम क्लीनर की मदद ले सकते हैं। वैक्यूम क्लीनर से धूल बाहर निकालने के लिए उससे की-बोर्ड की कुंजियों के बीच हवा इस प्रकार फैंके कि यह हवा कुंजियों के नीचे तक जाए। इससे नीचे जमी धूल उड कर बाहर आ जाएगी। यदि आपके पास वैक्यूम क्लीनर नहीं है तो की-बोर्ड को उल्टा करके इसकी धूल निकाल दें। इसकी कुंजियों को साफ

करने के लिए ग्लास साफ करने के लिए काम में लिया जाने वाला लिक्विड काम में लिया जा सकता है। रुई को इस लिक्विड में गीला कर लें। लेकिन ध्यान रहे कि रुई से लिक्विड निचोड़े जाने पर भी नीचे न गिरे। इससे लिक्विड के कुंजियों के नीचे जाने की संभावना नहीं रहेगी।

अब इस रुई से की-बोर्ड के ठोस भाग एवं कुंजियों, दोनों को साफ कर लें। अब थोड़ी देर के लिए इसे ऐसे ही पड़ा रहने दें ताकि यदि लिक्विड कुंजियों के नीचे भी गया हो तो वह सूख जाए। यदि आपका की-बोर्ड कुछ अधिक ही गंदा है या इस पर चाय कॉफी जैसा कोई द्रव्य गिर गया है तो इसकी सफाई के लिए आपको एक कदम और आगे बढ़ना होगा। आजकल आने वाले अधिकांश की-बोर्ड मेंब्रेन आधारित होते हैं। इसमें कुंजियों के नीचे एक झिल्ली होती है जो किसी कचरे या द्रव्य को नीचे तक जाने से रोकती है। इससे की-बोर्ड के खराब होने की आशंका कम रहती है।

इसके लिए आपको की-बोर्ड की असैंबली को खोलना होगा। लेकिन ध्यान रहे ऐसा करने पर इसकी गारंटी/वारंटी बेकार हो जाएगी। अत: ऐसा तभी किया जाए जबकि यह गारंटी/वारंटी समय में न हो तथा आप स्वयं यह रिस्क लेने को तैयार हों।

- की-बोर्ड को उल्टा करें। नीचे दिख रहे सभी पेंचों को खोल दें।
- जब चारों पेच खुल जाएं तो सावधानी से तीनों परतों को अलग-अलग कर दें।
- अब कुंजियों वाली परत को ग्लास क्लीनर की मदद से साफ कर दें तथा बीच वाली परत पर जमा कचरे को भी हटा दें।
- जब लिक्विड सूख जाए तो इन तीनों परतों को वापस उसी प्रकार अपनी जगह पर लगा दें। लीजिए आपके की-बोर्ड में एक नई जान आ गई है। यदि आपके पास पुराना की-बोर्ड है, तो संभव है कि उसके नीचे कोई झिल्ली न हो। ऐसे की-बोर्ड (नॉन-मेंब्रेन) में किसी द्रव्य के गिरने से यह खराब हो सकता है। यदि आप इसकी कुंजियों को साफ करना चाहते हैं तो एक छोटे पेचकस की मदद से इसकी कुंजियों को निकाल दें।

लेकिन ऐसा तभी करें जबकि आपका की-बोर्ड नॉन-मेंब्रेन हो। क्योंकि मेंब्रेन आधारित की-बोर्ड में कुंजियों को अलग नहीं किया जा सकता है। ऐसे में इन्हें बाहर निकालने की कोशिश से इनके टूटने का खतरा रहता है।

कुंजियों को बाहर निकालने से पूर्व संभव हो तो इसकी एक फोटो खींच लें अन्यथा सभी कुंजियों की पोजीशन एक कागज पर लिख लें। इससे इन्हें वापस लगाना आसान होगा। कुंजियों को निकालते वक्त बड़ी कुंजियों को अपनी जगह पर ही लगा रहने दें, क्योंकि इन्हें वापस लगाना मुश्किल कार्य है।

अब अलग हुई इन कुंजियों को अच्छे क्लीनर से साफ कर लें। जब कुंजियाँ साफ हो जाएं तो इन्हें ध्यान से किंतु थोड़ा-सा धकेलते हुए वापस अपनी जगह पर लगा दें।

माउस की सफाई

माउस को प्रयोग में लेते-लेते इसके नीचे स्थित बॉल के माध्यम से इसमें धूलकण तथा अन्य कचरा चला जाता है। इससे माउस के मूवमेंट में व्यवधान आता है। इस मूवमेंट को सही करने के लिए इसकी सफाई जरूरी हो जाती है। माउस को साफ करना काफी आसान है। इसमें जमे कचरे को बाहर निकालने के लिए आपको इसकी बॉल को बाहर निकालना होगा। बॉल को बाहर निकालने के लिए माउस को उल्टा करके इसके नीचे स्थित गोलाकार ढक्कन (Collar) को घुमाकर खोल लें। अब बॉल को बाहर निकाल लें। बॉल को बाहर निकालने पर आप के अंदर तीन रोलर पाएंगे। इन तीन रोलर में से दो तो X axis और Y axis रोलर (काला रंग) हैं और तीसरा सपोर्टिंग रोलर (सफेद रंग) कहलाता है।

यदि माउस को काफी समय बाद साफ किया जा रहा है तो संभवतः इन रोलर्स पर आपको कचरा लगा हुआ मिलेगा। इसे साफ करने के लिए आपको अपनी ऊँगली और नाखून का इस्तेमाल करना पड़ेगा। इस कचरे को नाखून की मदद से साफ करते हुए बाहर निकाल दें। बाहर निकाली गई बॉल को भी एक कपड़े से पोंछ दें। बॉल को साफ करने के लिए किसी क्लीनर का प्रयोग न करें। यह बॉल को खराब कर सकता है। अब बॉल को वापस इंसर्ट करते हुए कॉलर बंद कर दें। माउस को साफ करते वक्त आपसे इसके अलग-अलग बटन दब सकते हैं। ऐसे में यदि आपने कोई एप्लीकेशन खोल रखी है तो उसमें कुछ गड़बड़ हो सकती है। इससे बचने के लिए या तो आप कंप्यूटर को ऑफ करके यह कार्य कर सकते हैं या किसी सॉफ्टवेयर के माध्यम से कुछ देर के लिए माउस क्लिकों को डिसेबल कर दें। ऐसा ही एक सॉफ्टवेयर है : Clean n Go। यह सॉफ्टवेयर माउस क्लिकों और की-बोर्ड से दबाई गई कुंजियों को कुछ देर के लिए डिसेबल कर देता है।

अपने माउस का साफ रखने के लिए इसके लिए प्रयोग में लिए जाने वाले पैड को भी नियमित रूप से साफ करते रहें।[12]

5.2.2 कुंजीपटल का विवरण

पिछले जमाने के कंप्यूटरों में डेटा का निवेशन केवल छिद्रित पत्रकों या मेग्नेटिक टेप या डिस्कों की सहायता से किया जाता था, लेकिन माइक्रो कंप्यूटरों के विकास के साथ ही कुंजी-पटलों का बहुतायत से प्रयोग होने लगा क्योंकि इनका उपयोग बहुत आसान था एवं अन्य युक्तियों के बजाय इनका उपयोग बड़े विश्वासपूर्वक किया जा सकता था। इसका सबसे बड़ा फायदा यह है कि सीधे ही कंप्यूटर की स्मृति में डेटा का निवेश किया जा सकता है एवं क्या प्रविष्टियाँ की गई हैं, इन्हें एक क्षण में ही चाक्षुष प्रदर्शन एकक में देखा जा सकता है।

कुंजी-पटल का ऊपरी हिस्सा एक टाइपराइटर के समान ही होता है और इसमें उसी क्रम से रोमन लिपि के अक्षर, इंडो-अरेबियन पद्धति के अंक एवं टाइपिंग आदि के चिंह बने होते हैं। लेकिन इसके पटल पर सामान्य टाइपराइटर से कुछ अधिक कुंजियाँ

होती हैं जिनमें क्रमांक तथा अंकगणित के चिह्न आदि लिखे होते हैं। पर्सनल कंप्यूटर एवं होम कंप्यूटरों के कुंजीपटल अलग-अलग प्रकार के होते हैं। प्रत्येक होम कंप्यूटर का कुंजी-पटल अपनी ही डिजाइन का होता है जबकि पर्सनल कंप्यूटरों में लगभग एक ही प्रकार के कुंजी-पटल होते हैं। मानक की-बोर्ड 84 एवं 101 कुंजियों वाले होते हैं।

इस कुंजी-पटल के एकदम नीचे हर कुंजी के लिए एक विद्युत संपर्क एवं अंकीय परिपथ लगा होता है, जिसकी सहायता से संबंधित संप्रतीक को द्विआधारी विद्युत संकेतों में बदल दिया जाता है। ये संकेत केंद्रीय संसाधन एकक में जा कर कंप्यूटर में अपना कार्य प्रारंभ कर देते हैं।

इस कुंजी-पटल में कुछ फलन-कुंजियाँ (Function Keys) लगी हुई होती है, जिनकी सहायता से कुछ बार-बार होने वाले कार्यों को आसानी से किया जा सकता है। **जैसे कि कोई पैराग्राफ किसी प्रोग्राम में बार-बार आता है तो उस पूरे पैराग्राफ का एक नाम दे कर उसे किसी एक फलन कुंजी से संबंधित कर दिया जाता है। अब इस फलन कुंजी को दबाते ही पूरा पैराग्राफ अपने आप चाक्षुष प्रदर्शन एकक में प्रदर्शित हो जाएगा।**

कंप्यूटरों के कुंजी-पटल के अतिरिक्त डेटा-एंटरी सिस्टम, फोटो टाइप सेटिंग मशीनों, इलैक्ट्रॉनिक टाइपराइटरों, डेस्क टॉप पब्लिशिंग सिस्टम एवं वर्ड-प्रोसेसरों में लगभग एक ही तरह के कुंजी-पटल लगाए जा रहे हैं।[13] (वीरेंद्र जैन,पृ.112-113)

कंप्यूटर परिचारक को सबसे अधिक आवश्यकता कुंजीपटल की होती है। कंप्यूटर कुंजीपटल की सबसे बड़ी विशेषता यह होती है कि इसकी प्रत्येक कुंजी मात्र एक स्विच का कार्य करती है। सभी की आंतरिक संरचना एक जैसी ही होती है तब भी प्रत्येक कुंजी पृथक प्रभाव उत्पन्न करती है। इसके विपरीत साधारण टाइपराइटर की कुंजियों की संरचनाओं में प्रत्येक से संबंधित चिंह खुदा होता है जिसके कारण प्रत्येक कुंजी पृथक प्रभाव उत्पन्न करती है।

कंप्यूटर कुंजीपटल की कुंजियाँ तारों के एक संजाल पर पृथक-पृथक स्थलों से संबंधित होती हैं। इन तारों में विद्युत धारा संकेत प्रवाहित होते हैं। प्रत्येक कुंजी संजाल पर अपनी विशिष्ट स्थिति के कारण ही पृथक प्रभाव उत्पन्न करती है।
कंप्यूटरों के प्रकार के अनुसार कुंजीपटल तीन प्रकार के होते हैं :-

1. आई.बी.एम. - पी.सी. प्रकार,
2. पी.सी.एक्स.टी. तथा ए.टी. प्रकार, तथा
3. पी.एस. 2 प्रकार

क्रमाँक 2 कुंजीपटल प्रकार के कुंजीपटल भी दो प्रकार के होते हैं जिन्हें **क्वैर्टी** प्रकार तथा **एजर्टी** प्रकार कहा जाता है। इनमें से क्वैर्टी प्रकार अधिक प्रचलित है। इन कुंजीपटल प्रकारों के ये नाम कुंजीपटल की प्रथम 6 अक्षर कुंजियों के नाम पर दिए गए हैं। जिस कुंजीपटल का हम अध्ययन करेंगे उसकी प्रथम 6 अक्षर कुंजियाँ क्रमशः **Q W E R T Y** हैं इसलिए इसे क्वैर्टी कुंजीपटल कहा जाता है।

कुंजियों की संख्याओं के अनुसार भी कुंजीपटल कई प्रकार के आते हैं। जिनमें 84 कुंजी वाले तथा 101 कुंजी वाले कुंजीपटल अधिक लोकप्रिय हैं। इनकी कुंजियों का विभाजन इस प्रकार होता है :

<table>
<tr><th>कुंजी समूह</th><th>84 कुंजी</th><th>101 कुंजी</th><th>कुंजी नाम, चरित</th></tr>
<tr><td>टंकण कुंजियाँ-टाइपिंग की</td><td>48</td><td>48</td><td>A-Z, 1-0, `,_,+,],},{,",;,?,<,>,</td></tr>
<tr><td>कार्य कुंजियाँ - फक्शन की</td><td>10</td><td>12</td><td>F1-F10, or F1-F12</td></tr>
<tr><td>कर्सर कुंजियाँ</td><td>--</td><td>04</td><td><-|->|</td></tr>
<tr><td>सांख्यिक कुंजीपट कुंजियाँ</td><td>17</td><td>16</td><td>0-9, /, *, -, +, Enter, Delete,</td></tr>
<tr><td>संशोधन कुंजियाँ</td><td>04</td><td>06</td><td>Alt, Control, Shift</td></tr>
<tr><td>विशेष क्रिया कुजियॉं</td><td>03</td><td>13</td><td>Honie, End, Page Up, Page Down, Print Screen, Scroll Lock, Pause, Tab, Backspace, Delete, Spacebar, Escape, Enter</td></tr>
<tr><td>द्विदिशी क्रिया कुजियाँ</td><td>02</td><td>03</td><td>Insert, CapsLock, Num Lock</td></tr>
<tr><td>कुल कुजियाँ</td><td>84</td><td>102</td><td></td></tr>
</table>

101-कुंजी कुंजीपटल में वास्तव में 102 कुंजियाँ होती हैं इसलिए इसे संवर्द्धित अथवा ऐनहैंस्ड कुंजीपटल कहा जाता है।

1) टंकण कुंजियाँ

टंकण कुंजियों के दो वर्ग हैं - 26 अक्षर कुंजियाँ जिन पर अँग्रेजी के ए से जेड तक के अक्षर अंकित हैं तथा 22 चरित कुंजियाँ जिन पर संख्या अंक तथा व्याकरण चिंह अंकित हैं। प्रत्येक टंकण कुंजी द्वारा दो चरितों का टंकण संभव होता है - एक अकेली कुंजी उपयोग से तथा दूसरा शिफ्ट कुंजी के साथ। अक्षर कुजियाँ अकेले ॲग्रेजी के छोटे अक्षर तथा शिफ्ट कुंजी के साथ अँग्रेजी के बडे अक्षर टंकित करती हैं। अन्य कुंजियों में से प्रत्येक पर दो चरितों का अंकन होता है। अकेली कुंजी से नीचे अंकित चरित तथा शिफ्ट कुंजी के साथ ऊपर अंकित चरित टंकित होता है। कैप्सलॉक कुंजी को एक बार दबाकर छोड़ने से अक्षर कुंजियाँ अँग्रेजी के बडे अक्षर टंकित करती हैं। इस कुंजी एक बार पुनः दबाकर छोड़ने से इसका प्रभाव समाप्त हो जाता है। कैप्सलॉक कुंजी का चरित्त कुंजियों पर कोई प्रभाव नहीं होता। हिंदी के प्रोग्राम टंकण कुंजियों के उपयोग में ही परिवर्तन करके इनसे देवनागरी लिपि के अक्षर टंकित कराते हैं।

2) टंकण सहायी कुंजियाँ

इन कुंजियों के टंकण कार्य में व्यापक उपयोग हैं जो इस प्रकार हैं :-

क- स्पेसबार अथवा रिक्ति पट्टिका

कंप्यूटर उपयोग करते समय इस कुंजी का सबसे अधिक उपयोग होता है इसलिए इसे सबसे लंबा बनाया गया है। इस कुंजी को एक बार दबाकर छोड़ने से कर्सर एक रिक्त स्थान छोड़कर दाँई ओर को आगे बढ़ जाता है।

ख - एंटर अथवा प्रवेश

कंप्यूटर उपयोग में इस कुंजी का सबसे अधिक महत्वपूर्ण योगदान होता है। कंप्यूटर को कोई अधिदेश देने में अधिदेश को अंकित करने के बाद इस कुंजी के उपयोग से ही कंप्यूटर अधिदेश को स्वीकार करता है। पाठ्य के टंकण में यह कुंजी परिच्छेद परिवर्तन के लिए उपयोग की जाती है।

ग- बैकस्पेस अथवा पश्चरिक्ति

इस कुंजी के उपयोग से कर्सर के बाँई ओर के चरित्रों को मिटाते हुए एक कदम पीछे की ओर बढाया जाता है। अत: परिचारक द्वारा की गई भूल को सुधारने में इस कुंजी का उपयोग आवश्यक होता है।

घ- शिफ्ट अथवा अंतरण कुंजी

प्रत्येक टंकण कुंजी दो मुद्रण प्रभाव उत्पन्न कर सकती है - एक अकेले तथा अन्य शिफ्ट कुंजी के साथ। इस प्रकार शिफ्ट कुंजी का टंकण में बहुत महत्व है। यह कुंजी प्रोग्राम द्वारा निर्धारित अन्य प्रकार के कार्य कर सकने में भी सक्षम है।

च- कैप्सलॉक अथवा ओद्यक कुंजी

अँग्रेजी में दो प्रकार के अक्षर हैं - ओद्य अथवा कैपिटल तथा उद्य अथवा स्मॉल। इस कुंजी को एक बार दबाकर छोड देने से अक्षर कुंजियाँ केवल ओद्य वर्णों को मुद्रित करती हैं। इसी को पुन: दबाकर छोडने से यही कुंजियाँ उद्य वर्णों को मुद्रित करने लगती हैं। एक ही कर्म से दो विपरीत प्रभाव - एक बार एक तथा अगली बार अन्य - उत्पन्न करने को टॉगल अथवा एकान्य क्रिया कहा जाता है। यहाँ कैप्सलॉक कुंजी एक टॉगल कुंजी कहलाती है।

3) कर्सर कुंजियाँ

101-कुंजी कुंजीपटल पर चार कुंजियों का एक समूह होता है जिन पर तीरों के चिंह बने होते हैं, इन कुंजियों को कर्सर कुंजियाँ कहते हैं। इनका उपयोग कर्सर को परदे पर ऊपर, नीचे, दाँए तथा बाँए ले जाने के लिए किया जाता है। कुंजियों पर तीरों के चिंह कर्सर के जाने की दिशा दर्शाते हैं।

इन कुंजियों के पृथक प्रावधान केवल 101-कुंजी कुंजीपटल में होते हैं। 84-कुंजी कुंजीपटल में ये क्रियाएँ सांखियक कुंजीपट की कुंजियों से की जाती हैं। 101-कुंजी कुंजीपटल में भी ये क्रियाएँ सांख्यिक कुंजीपट की कुंजियों से की जा सकती हैं।[13]

कुछ कुंजियाँ कंप्यूटर को अधिदेश प्रदान करने में सहायक होती हैं। कुछ कुंजियाँ एक से अधिक कार्य कर सकती हैं। इस प्रकार कुंजियों को अलग अलग वर्गों में रखा जा सकता है, जो इस प्रकार हैं :-

1) विशेष क्रिया कुंजियाँ

क- एस्केप अथवा क्रियात्याग कुंजी

किसी क्रिया के प्रारंभ हो जाने के बाद उसे यथास्थिति में छोड़ने के लिए इस कुंजी का उपयोग किया जाता है। इसका उपयोग परिचारक द्वारा की गई भूल को सुधारने के लिए अथवा शीघ्रता के लिए प्रोग्राम की कुछ अनावश्यक क्रियाओं को छोड़ने के लिए किया जाता है।

ख- डिलीट अथवा मिटाव कुंजी

कर्सर के दाँयी ओर के चरित को अथवा चुने गए भाग को मिटाने के लिए इस कुंजी का उपयोग किया जाता है।

ग- प्रिंटस्क्रीन अथवा प्रदर्शमुद्रण कुंजी

कंप्यूटर परदे पर दिखाई देने वाले भाग के यथावत मुद्रण के लिए यह कुंजी उपयोग की जाती है।

घ- स्क्रॉललॉक अथवा सरकननिषिद्ध कुंजी

इस कुंजी के उपयोग से कंप्यूटर के परदे पर दिखाई देने वाला भाग यथावत स्थिर किया जाता है।

च- पॉज़ अथवा विराम कुंजी

कंप्यूटर द्वारा की जा रही क्रिया के अस्थायी विराम के लिए इस कुंजी का उपयोग किया जाता है।

2) कर्सर संयमन कुंजियाँ

क- टैब अथवा कुदान कुंजी

यह कुंजी कर्सर को पहले से निर्धारित स्थान पर एक साथ ले जाने के लिए उपयोग की जाती है। इस क्रिया से तालिका निर्माण सरल हो जाता है। कर्सर को आगे ले जाने के लिए इस कुंजी का अकेले उपयोग किया जाता है तथा पीछे ले जाने के लिए इसका उपयोग शिफ्ट कुंजी के साथ किया जाता है।

ख- होम अथवा प्रास्थल कुंजी

कर्सर को पंक्ति के अथवा चुने गए भाग के अंतिम चरित के बाद के स्थान पर ले जाने के लिए यह कुंजी उपयोग की जाती है।

ग- एंड अथवा अंतस्थल कुंजी

कर्सर को पंक्ति में अंकित अंतिम चरित अथवा चुने गए भाग के अंतिम चरित के बाद के स्थान पर ले जाने के लिए यह कुंजी उपयोग की जाती है।

घ- पेजअप अथवा उत्तरपृष्ठ कुंजी

इस कुंजी के उपयोग से कंप्यूटर परदे पर दिखाई देने वाले भाग से आगे के भाग को परदे पर लाया जाता है।

च- पेजडाउन अथवा पूर्वपृष्ठ कुंजी

यह कुंजी पेजअप कुंजी के विपरीत दिशा में कार्य करती है।

3) द्विदिशी क्रिया कुंजियाँ अथवा टॉगल कुंजियाँ

जब किसी प्रभाव की केवल दो ही अवस्थाएँ संभव हो, जैसे स्विच की ऑन तथा ऑफ अवस्थाएँ, तथा एक ही प्रकार की बाह्य क्रिया करने से किसी एक से दूसरी अवस्था में जाया जा सकता हो तो क्रिया को टॉगल क्रिया कहा जाता है। कंप्यूटर कुंजीपटल में तीन कुंजियाँ इसी प्रकार की हैं :-

क) इन्सर्ट अथवा मध्यस्थापन कुंजी

कंप्यूटर पर पाठ्य के मध्य में कहीं चरित्त अंकन के समय दो संभावनाएँ होती हैं। या तो कर्सर पर पहले से स्थित चरित को आगे खिसका कर नया चरित अंकित होता है, या फिर पुराने चरित्त को मिटाकर उसके स्थान पर नया चरित्त अंकित होता है। प्रथम क्रिया को इन्सर्ट अथवा मध्यस्थापन विधा कहा जाता है जबकि दूसरी क्रिया को ओवरराइट अथवा ऊपरिलेखन विधा कहा जाता है। प्रारंभ में कंप्यूटर की विधा प्रोग्राम द्वारा निर्धारित होती है। इन्सर्ट अथवा मध्यस्थापन कुंजी के एक बार उपयोग से विधा परिवर्तित हो जाती है। इस कुंजी के पुन: उपयोग से विधा मूल स्थिति में आ जाती है। कुछ प्रोग्राम, जैसे पेजमेकर, इस कुंजी का उपयोग अन्य कार्यों के लिए भी करते हैं।

ख) कैप्सलॉक कुंजी

इस कुंजी के एक बार उपयोग से अक्षर टंकण कुंजियाँ अँग्रेजी के बड़े अक्षर टंकित करने लगती हैं तथा पुन: उपयोग से अँग्रेजी के छोटे अक्षर पुन: टंकित करने लगती हैं।

ग) नमलॉक कुंजी

सांख्यिक कुजीपटल अथवा न्यूमेरिक कीपैड की अधिकाँश कुंजियों के दो प्रभाव होते हैं जो उन पर अंकित होते हैं। नमलॉक कुंजी इस कीपैड की अन्य कुजियों के प्रभाव में परिवर्तन करने के काम आती है।

4) संशोधक अथवा मॉडिफायर कुंजियाँ

कंप्यूटर कुंजीपटल में तीन कुंजियाँ ऐसी हैं जो स्वयं अकेले कोई प्रभाव उत्पन्न नहीं करतीं। इनका उपयोग अन्य कुंजियों के साथ ही किया जाता है। इनमें से एक कुंजी शिफ्ट है जिसका प्रमुख कार्य अक्षर टंकण कुंजियों द्वारा अँग्रेजी के छोटे अथवा बड़े वर्णों को टंकित कराना है।

शिफ्ट, एल्ट, तथा कंट्रोल इन तीन संशोधक कुंजियों के कार्य कंप्यूटर प्रोग्राम द्वारा निर्धारित होते हैं।

5) कर्त्यन अथवा फंक्शन कुंजियाँ

एफ1 से लेकर एफ10 अथवा एफ12 तक नामांकित ये कुंजियाँ कंप्यूटर को विशिष्ट संक्रियाओं को संपन्न करने हेतु अधिदेश देने के लिए प्रयुक्त की जाती हैं। ये संक्रियाएँ कंप्यूटर प्रोग्राम द्वारा निर्धारित होती हैं।

6) सांख्यिक कुंजीपट

कुंजीपटल के सामान्यत: दाँए भाग में तीन इंगित्र ज्योतियों के नीचे 17 कुंजियों का यह समूह अनेक कार्य करता है। इनमें से कुछ कुंजियों पर दो चिंह बने हैं अत: ये कुंजियाँ दो-दो कार्य कर सकती हैं। इन दो कार्यों में से एक का चुनाव नमलॉक कुंजी द्वारा किया जाता है। नमलॉक कुंजी के एक दोल से उसके ऊपर की नमलॉक नामक ज्योति जल उठती है तथा दूसरे दोल से बुझ जाती है। जब ज्योति जलती है, तब सांख्यिक कुंजीपट की कुंजियाँ संख्या अंकों को अंकित करती हैं, अन्यथा कुंजी पर अंकित अन्य कार्य संपन्न होता है। इन कार्यों को नीचे सूची दर्शाया गया है -

कुंजी	ज्योति जली	ज्योति बुझी
		डिलीट क्रिया
0	0	इन्सर्ट क्रिया
1	1	एंड क्रिया
2	2	कर्सर क्रिया
3	3	पेजडाउन क्रिया
4	4	कर्सर बाँए
5	5	कुछ नहीं
6	6	कर्सर दाँए
7	7	होम क्रिया
8	8	कर्सर ऊपर
9	9	पेजअप
/	/	/
*	*	*
-	-	-
+	+	+
एंटर	एंटर	एंटर

इन कार्यों के अतिरिक्त नमलॉक कुंजीपटल कुंजियों का उपयोग आस्की द्वारा निर्धारित चरितों के अंकन के लिए भी होता है। यदि एल्ट कुंजी दबाकर आस्की मान की कुंजियाँ क्रमश: दोलित की जाकर एल्ट कुंजी को छोड़ने से परदे पर आस्की मान के समकक्ष चरित छप जाता है। इसके लिए एएएसआई सिस नामक प्रोग्राम सक्रिय होना चाहिए।[14]

5.2.3 विभिन्न कुँजी-पटलों का अध्ययन

5.2.3(क) देवनागरी तथा अँग्रेजी, रेमिंग्टन टाइपराइटर, ध्वन्यात्मक हिंदी की-बोर्ड

हिंदी की-बोर्ड इनपुट का एक सामान्य परंतु अत्यंत सुविधाजनक, लोकप्रिय और महत्वपूर्ण उपकरण है। इस पर अनेक बटन या कुंजियाँ (Keys) लगी होती हैं, जिन्हें दबाकर हम कोई आदेश या डाटा टाइप करते हैं, ठीक उसी तरह जिस प्रकार किसी टाइपराइटर पर टाइप किया जाता है। सामान्यतया हम जो कुछ टाइप करते हैं, वह कंप्यूटर के मॉनीटर पर भी दिखाया जाता रहता है। जब सूचना तैयार हो जाती है, तो सामान्यतया हम एक विशेष कुंजी, जिसे रिटर्न (Return) या एंटर (Enter) कहा जाता है, दबा कर उस सूचना को कंप्यूटर में भेज देते हैं।

अलग-अलग प्रकार के कंप्यूटरों के लिए अलग-अलग प्रकार के की-बोर्ड उपलब्ध हैं, लेकिन डीटीपी के कार्य में सामान्यत: हम 104 कुंजियों वाले की-बोर्ड का उपयोग करते हैं। कंप्यूटरों के साथ पाए जाने वाले सामान्य कुंजीपटलों को **अँग्रेजी (English) की-बोर्ड** कहा जाता है, क्योंकि इनसे सामान्यतया हम अँग्रेजी का पाठ्य ही टाइप करते हैं। डीटीपी के कार्य में हमें अँग्रेजी के साथ-साथ हिंदी में भी सामग्री भरने की आवश्यकता होती है। लेकिन इसका मतलब यह नहीं है कि उसके लिए हमें अलग की-बोर्ड लाना पड़ेगा। वास्तव में की-बोर्ड तो हर भाषा के कार्य के लिए एक ही होता है, परंतु **भाषा के अनुसार फोंट बदल जाते हैं और फोंट बदलने के साथ ही की-बोर्ड के मुख्य भाग की कुंजियों का अर्थ भी बदल जाता है।** उदाहरण के लिए, जब हम किसी हिंदी फोंट जैसे चाणक्य या अर्जुन का उपयोग करते हैं, तो मुख्य की-बोर्ड की 'J' कुंजी को दबाने पर 'र' टाइप होता है। इसी प्रकार अन्य बटनों के बारे में भी समझा जा सकता है।

दुर्भाग्य से हिंदी के सभी फोंटों द्वारा उपयोग किया जाने वाला कुंजीपटल एक समान नहीं है। अधिकांश फोंट **रेमिंगटन टाइपराइटर (Remington Typewriter) के की-बोर्ड** का उपयोग करते हैं। हिंदी में टाइप करना जानने वाले व्यक्ति इस तरह के फोंटों का उपयोग करके डीटीपी के कार्य के लिए हिंदी में पाठ्य सरलता से तैयार कर लेते हैं। इसमें टाइप करने की गति भी अपेक्षाकृत अधिक होती है। अन्य बहुत से फोंट सी-डैक (C-DAC) द्वारा विकसित किए गए भारतीय भाषाओं के की-बोर्ड का उपयोग करते हैं। इसे जिस्ट (GIST) की-बोर्ड भी कहा जाता है। इसमें सभी मात्राएं और स्वर एक तरफ तथा व्यंजन दूसरी तरफ लगाए गए हैं। इसमें अक्षरों की बारंबारता (Frequency) का ध्यान नहीं रखा गया है, इसलिए इसमें पाठ्य सामग्री टाइप करना बहुत समय खाने वाला कार्य है।

इनके अतिरिक्त बहुत से फोंट इन दोनों की-बोर्डों की जगह **ध्वन्यात्मक (Phonetic) की-बोर्ड** का उपयोग करते हैं, जिसमे हिंदी के शब्दों का रोमन लिपि में टाइप किया जाता है। उदाहरण के लिए 'राम' टाइप करने के लिए हमें अपने साधारण की-बोर्ड पर 'rAm' टाइप करना होगा। इसमें छापे के छोटे और बड़े दोनों प्रकार के अक्षरों का

अर्थ भिन्न होता है। बहुत से लोगों को यही सुविधाजनक लगता है, क्योंकि उन्हें हिंदी का की-बोर्ड याद नहीं रखना पड़ता। लेकिन इस की-बोर्ड में टाइप करने की गति पहले बताए गए दोनों की-बोर्डों की अपेक्षा बहुत ही कम होती है और गलतियां होने की संभावना अधिक होती है, क्योंकि इसमें शिफ्ट के उपयोग का विशेष ध्यान रखना पड़ता है।

आप हिंदी पाठ टाइप करने के लिए, जिस प्रकार के की - बोर्ड को सुविधाजनक समझते हों, उसका उपयोग करने वाले फोंट का उपयोग कर सकते हैं। यदि की-बोर्ड याद रखने में कठिनाई आती है, तो अपने कुंजीपटल के बटनों पर ऐसे स्टिकर चिपका सकते हैं, जिन पर हिंदी और अँग्रेजी दोनों के अक्षर छपे हों। आजकल ऐसे स्टिकर सरलता से उपलब्ध हैं।[15]

5.2.3(ख) क्वर्टी (QWERTY) तथा डिवोरक (DVORAK) कुंजी-पटल

की-बोर्ड ले आउट से आशय उस पर विभिन्न केरेक्टर्स की कुंजियों के क्रम से है। क्वर्टी की-बोर्ड केरेक्टर्स के क्रम को दर्शाता है। इसके अतिरिक्त डिवोरक की-बोर्ड 1930 में अगस्त डिवोरक (August Dvorak) और विलियम डिअले (william Dealy) डिजाइन की थी। पुरातन क्वर्टी की-बोर्ड से हट कर डिवोरक की-बोर्ड को इस प्रकार डिजाइन किया गया है कि बीच वाली पंक्ति में वे केरेक्टर्स हों जो अधिक काम में आते हैं। इसके साथ ही इन केरेक्टर्स को इस क्रम में लगाया गया है कि उन्हें टाइप करना आसान रहे और वे जल्दी से टाइप हो जाएँ। क्वर्टी की-बोर्ड में 8 घंटे में एक टाइपिस्ट के हाथ लगभग 16 मील का सफर तय करते हैं वहीं डिवोरक की-बोर्ड में यह 1 मील ही रह जाता ह। एक ही हाथ से काम करने में सक्षम लोगों के लिए सामान्य डिवोरक की-बोर्ड के अतिरिक्त लेफ्ट-हेंडेड की-बोर्ड और राइट-हेंडेड की-बोर्ड भी आता है।[16]

5.2.3(ग) मानक फोंट कूट

पीसी तथा मैकिंटोश पर गुणवत्तापूर्ण फोंट डिजाइन सॉफ्टवेयर की उपलब्धता ने फोंट के विकास की प्रक्रिया को सरल बना दिया है। अनेक मैकिंटोश उपयोक्ता ने अधिक जटिल लिपियों जैसे भारतीय लिपियों के लिए भी फोंट का विकास करना आरंभ कर दिया था। लेकिन शीघ्र ही उन्होंने पाया कि फोंट डिजाइन यंत्र-विन्यांस का विकास अँग्रेजी जैसी भाषा की रैखिक लिपि के लिए ही किया गया था तथा यह अरैखिक लिपियों के लिए पूर्णतया अपर्याप्त है।

अँग्रेजी भाषा में कैरेक्टर एक दूसरे के साथ जोडे नहीं जाते हैं तथा उन्हें रैखिक क्रम में प्रस्तुत किया जाता है। कुंजीपटल तथा प्रदर्शन हेतु प्रयुक्त कैरेक्टर समान हैं। फोंट-प्रदर्श-यंत्रविन्यास संसक्त कैरेक्टर की सीमित अंतिव्याप्ति (कर्निंग) उपलब्ध कराता है। जटिल लिपियों की वर्णमाला किसी पाठ को तैयार करने के लिए प्रयुक्त मूलभूत आकृतियों से पूर्णतया भिन्न हो सकती है। समुचित संयोजन के लिए आवश्यक मूलभूत

आकृतियाँ अँग्रेजी की तुलना में कहीं ज्यादा अधिक हो सकती हैं। इन आकृतियों के रूप में टंकण-कार्य अत्यंत कष्टप्रद प्रक्रिया हो सकती है।

दुर्भाग्यवश, इन कंप्यूटरों में उपलब्ध सरलतम यंत्रविन्यास के तहत उपयोक्ता फोंट कैरेक्टर के रूप में ही टंकण करने के लिए बाध्य थे। टंकण की दृष्टि से फोंट कैरेक्टर समुच्चय (सैट) के प्रारूपण (डिजाइनिंग) के लिए समझौते करने पड़ते थे।

कुंजीपटलीकरण की सीमाएँ

कैरेक्टरों को मानक अँग्रेजी कुंजीपटल पर टाइप करना पड़ता था, जो अँग्रेजी के 26 वर्णों के टंकण के लिए बनाए गए थे। जटिल लिपियों के लिए आवश्यक कैरेक्टरों की अधिक संख्याओं के कारण कुंजीपटल पर विभिन्न ओवरले (अधिचित्र) की आवश्यकता पड़ी। ओवरले की अधिकता ने टंकण में कठिनाइयाँ बढ़ा दी, जबकि ओवरले की संख्या कम करने पर कंपोजिंग की गुणवत्ता से समझौता करना पड़ता था। अधिकांश कंप्यूटरों में इन कैरेक्टरों को समाहित करने के लिए चार ओवरले उपलब्ध थे।

इन समझौतों के कारण, किसी एक लिपि के लगभग सभी फोंट कार्यान्वयन एक-दूसरे से भिन्नता लिए हुए थे। इन कुंजीपटलों में से किसी एक का प्रयोग करने में दक्षता प्राप्त करने में उपयोक्ता को कई माह लग जाते थे। इसके बाद किसी अन्य प्रकार के कुंजीपटल पर कार्य करना कठिन हो जाता था, क्योंकि इसका अर्थ किसी नई प्रणाली वाले कुंजीपटल को अपनाना होता था। इस कारण, किसी भी प्रकार का कुंजीपटल लोकप्रिय नहीं हो पाया।

जटिल लिपियों के मामले में कंप्यूटर की अक्षमता से उत्पन्न लिपि संबंधी कठिनाइयों के कारण उपयोक्ताओं को लगता था कि ये लिपियाँ समुचित कंप्यूटरीकरण के लिए उपयुक्त नहीं है। फलत: इन लिपियों का प्रयोग कदाचित ही किया जाता था जबकि अँग्रेजी लिपि अपनी सरलता के कारण मैदान जीत लेती थीं।

जटिल लिपियों से संबद्ध प्रकाशन उद्योग में विशेष रूप से प्रशिक्षित प्रचालकों की आवश्यकता पड़ती थी जो दुर्वह कुंजीपटलों पर कार्य कर सकते थे। उन्हें आपूर्तिकर्ताओं द्वारा प्रदत्त कराए गए उनके टाइपसैटिंग प्रणाली के कुछ फोंट से काम चलाना पड़ता था। संगतता के अभाव में अन्य प्रणालियों के फोंट का प्रयोग वे नहीं कर पाते थे। फलत: जटिल लिपि वाले प्रकाशन में उत्पादकता तथा गुणवत्ता पर प्रतिकूल प्रभाव पड़ता था।

फोंट का प्रतिबल/तनाव

जटिल लिपियों के लिए साधारण कुंजी योजना से समस्या का आंशिक हल होता है - यह उपयोक्ता को कुंजीपटल के प्रतिबल/तनाव से राहत देता है। लेकिन फोंट की असंगतता की समस्या फिर भी बनी रहती है। फोंट के मानक स्वरूप की आवश्यकता बनी रहती है जो विभिन्न विकासकर्ताओं द्वारा विकसित विविध प्रकार के संगत फोंटों में से उपयोक्ता को चयन करने की सुविधा दिला सके।

फोंट कैरेक्टर का कुंजीयन करने की आवश्यकता नहीं पड़ने के कारण इष्टतम कैरेक्टर समुच्चय (सैट) का निर्णय लेने में अत्यंत सहायता मिलती है। अभी तक डिजाइनरों को यह सुनिश्चित करना पड़ता था कि प्रत्येक कैरेक्टर टंकण हेतु बहुत कष्टप्रद न हो। अब कंप्यूटर कलनविधि किसी भी लिपि के लिए आधारभूत आकृतियों का उपयोग करते हुए संयोजन संबंधी जटिलताओं को निराकरण कर सकती है। प्रत्येक प्रणाली फोंट कूट समुच्चय (सैट) में उपलब्ध कैरेक्टरों की संख्या सीमित करती है। सामान्यतः अधिकतम संख्या 188 कैरेक्टर होती है। इस सीमा में किसी लिपि के लिए आवश्यक आधारभूत आकृति की व्यवस्था करने की चुनौती होती है तथा इसके साथ-साथ इसके सौंदर्य-बोध और सुरुचिमयता को भी बनाए रखना होता है।

कुंजी संकल्पना

अँग्रेजी भाषा में प्रत्येक अक्षरों को पृथक रूप से छापा जा सकता है, तथा इसीलिए छपाई के लिए केवल आधारभूत वर्णमाला की ही आवश्यकता होती है। लेकिन अधिकांशतः जटिल अरैबिक लिपियों के मामले में ऐसा नहीं होता है। इनके लिखित रूप में कई-सौ कैरेक्टर आकृतियाँ हो सकती हैं जो आधारभूत वर्णमाला समुच्चय (सैट) का संयोजन प्रस्तुत करते हैं। इनके आधारभूत वर्णमाला समुच्चय (सैट) अँग्रेजी से बहुत अधिक बड़ा नहीं है। इन जटिल लिपियों को वर्णमाला तथा फोंट के लिए पृथक कूट की आवश्यकता होती है।

अरैबिक लिपि का उपयोग इस समय के कंप्यूटरों में जटिल हो जाता है क्योंकि उपयोक्ता को फोंट कैरेक्टर में टंकण करना पड़ता है। संयोजन संबंधी ऊबाऊपन भी है। इसके अलावा, फोंट कैरेक्टर तदर्थ प्रकृति के होने के कारण, संबंधित कूट तथा कुंजीपटल प्रत्येक कार्यान्वयन में भिन्न होते हैं।

प्रदर्श या छपाई के संयोजन के लिए प्रयुक्त कूटों से कुंजीपटलों को पूर्णतया पृथक किया जाने की आवश्यकता होती है। इसी प्रकार चूँकि वर्तनी तथा छँटाई को केवल आधारभूत वर्णमाला के रूप में ही परिभाषित किया जाता है, अतः उपयोक्ता इंटरफेस की आवश्यकता पड़ती है जिसकी वजह से उपयोक्ता को वर्णमाला के अक्षर मात्र पर ही ध्यान देना होता है, तथा फोंट संबंधी विस्तृत कार्रवाई पारदर्शिता के साथ पूरी कर ली जाती हैं।

वर्णमाला कूट से फोंट कूट का पृथक्कीकरण के अनेक लाभ हैं। अब छपाई के लिए इष्टतम कैरेक्टर समुच्चय (सैट) का प्रारूपण करना संभव है, जिसके लिए लिपि की आधारभूत आकृतियों की ही आवश्यकता पड़ती है। अब कुंजीपटल से इन प्रत्येक कैरेक्टरों की टंकण-क्षमता तथा अनेक ओवरले पर इन्हें संयोजित करने की चिंता करने की आवश्यकता नहीं है। फोंट में आधारभूत कैरेक्टरों का चयन विवेकपूर्ण ढंग से करने पर, यह सुनिश्चित करना संभव है कि इसके माध्यम से लिपि के विभिन्न शैलियों को प्रस्तुत किया जा सकता है।

फोंट कूट को इस प्रकार परिभाषित किया जा सकता है कि विद्यमान सॉफ्टवेयर की सामान्य सीमाओं के तहत, यह लिपि की प्रकृति तथा सौंदर्यबोध तथा सुरुचि के साथ न्याय कर सकता है।

फॉंट कूट को बौद्धिक कलनविधि से जनित किया जा सकता है जिसे वर्णमाला कूट में प्रस्तुत किया ज़ा सकता है। आधारभूत आकृतियों का उपयोग करते हुए लिपि के समुचित संयोजन में यह बौद्धिक कलनविधि विशेषज्ञों के ज्ञान का आसवन कर सकती है।

इस प्रकार उपयोक्ता को अपने वर्णमाला-कुंजीपटल का उपयोग करते हुए आधारभूत वर्तनी के टंकण पर ही ध्यान केंद्रित करना होता है और कंप्यूटर श्रेष्ठ फॉंट का प्रयोग करते हुए तथा पूर्णतः संयोजित करते हुए, शब्द प्रदर्शित करने में सक्षम हो जाता है। कुंजीपटल से प्रविष्ट वर्णमाला-कूट प्रदर्श हेतु फॉंट-कूट में बदल जाते हैं।

नई कूट-सत्ताएँ

जटिल लिपियों की समुचित प्रस्तुति सुनिश्चित करने के लिए निम्नलिखित कूट निर्धारित किए गए हैं :

(i) इनस्क्रिप्ट - इंडियन स्क्रिप्ट (INSCRIPT - Indian Script)

जिस्ट प्रौद्योगिकी द्वारा इनस्क्रिप्ट कुंजीपटल की प्रस्तुति के साथ जटिल लिपियों का उपयोग करना सरल हो गया। इसका ओवरले मानक अँग्रेजी कुंजीपटल पर लगाया जा सकता है तथा यह समान कुंजियों का प्रयोग करते हुए समस्त भारतीय लिपियों में टंकण की सुविधा प्रदान करता है। यह इन सभी लिपियों के लिए आवश्यक आधारभूत वर्णमाला मात्र ही उपलब्ध कराता है, जिसे सरलतापूर्वक सामान्य तथा शिफ्ट ओवरले में समायोजित किया जा सकता है।

अँग्रेजी की ही भाँति, इनस्क्रिप्ट कुंजीपटल किसी शब्द की विशिष्ट तथा अनन्य वर्तनी के रूप में टंकण की सुविधा प्रदान करता है। चयनित लिपि के लिए निर्धारित संयोजन नियमों के अनुसार प्रदर्श का संयोजन स्वतः हो जाता है। प्रदर्श-संयोजन का कार्य सौंदर्यबोधी तथा सुरुचिपूर्ण ढंग से किया जाता है क्योंकि कंप्यूटर कैरेक्टरों के विशाल समुच्चय (सैट) से चयन कर सकता है। इसका अर्थ है कि उपयोक्ता प्रदर्श को देखे बिना भी वर्तनी का टंकण करते हुए, कंप्यूटर द्वारा उसे इष्टतम स्तर तक सही रूप में प्रस्तुत करने के प्रति आश्वस्त हो सकता है।

जब आधारभूत आकृतियों का प्रयोग करते हुए संयोजन कार्य में होने वाली जटिलताएँ तथा ऊबाऊपन समाप्त हो जाता है, तथा केवल आधारभूत वर्तनियों का प्रयोग करते हुए किसी लिपि में टंकण करना संभव हो जाता है, उपयोक्ता कुंजीपटल पर सीधे टंकण करते हुए भी किसी पाठ को लिख सकता है। कुंजीपटल की कार्रवाई अंतर्ज्ञात गतिविधि बन जाने के कारण, अब उपयोक्ता अपने विचार-विमान में उड़ान भरते रह सकता है।[17]

जो लोग बहुभाषी टाइपिंग में नए हैं, उनके लिए कोई समस्या नहीं है। परंतु सामान्य टाइपिंग मशीनों का उपयोग करने वाले जब कंप्यूटर पर टाइप करते हैं तो समस्या का मूल कारण टाइपराइटरों पर कुंजियों का अलग-अलग संयोजन होता है। इसलिए की-बोर्ड लेआउट के लिए भी एक मानक की आवश्यकता थी।

सभी भारतीय लिपियों में लिखने का तरीका एक जैसा ही होता है। उदाहरण के लिए देवनागरी, तमिल, पंजाबी आदि सभी लिपियों में 'पुणे' लिखने में अक्षरों और मात्राओं का क्रम एक जैसा ही होगा। इन सभी भाषाओं के लिए बनाया गया **इनस्क्रिप्ट (इंडियन स्क्रिप्ट) की-बोर्ड** एक मात्र ऐसा की-बोर्ड है जो **किसी भी एप्लीकेशन पर प्रयोग में लिया जा सकता है।** इसमें प्रयोग में लिया जाने वाला प्रमुख सिद्धांत यह है कि **अक्षर पहले टाइप किया जाए और मात्रा बाद में।** मात्रा को अक्षर से पहले लगाने का कार्य प्रोसेसर पर छोड़ दिया जाता है। पहले से टाइपिंग न जानने वाले इसमें शीघ्र ही अच्छी गति बना सकते हैं।[18]

(ii) ऑकी (ACII) - Alphabetic Code for Information Interchange.
सूचना के अंत:-अंतरण हेतु वर्णमाला कूट

यह कंप्यूटर कूट है जिससे किसी लिपि के मूलभूत वर्णों (अक्षरों) को प्रस्तुत किया जाता है। अधिकांश लिपियों में (चीनी जैसी चित्र लिपियों को छोड़ कर) प्रयुक्त मूलभूत अक्षर तथा चिंह 96 से कम हैं। किसी लिपि की सभी संभव आकृतियॉ इन मूलभूत अक्षरो के संयोजन से अभिव्यक्त की जा सकती हैं।

ऑकी कूट को कुंजीपटल **ओवरले** से टाइप किया जा सकता है। ऑकी कुंजीपटल ओवरले को किसी भी मानक अँग्रेजी कुंजीपटल पर लगाया जा सकता है। कुंजीपटल ओवरले में प्रत्येक ऑकी कैरेक्टर की अनन्य तथा भिन्न स्थिति होती है।

ऑकी कूट का प्रयोग करते हुए पाठ की प्रस्तुति

* ऑकी कुंजीपटल पर उपलब्ध सभी आधारभूत कैरेक्टर ऑकी कूट में होते हैं। उदाहरण के लिए, ऑकी भारतीय कूट तथा कुंजीपटल में 10 भारतीय लिपियों - असमी, बांग्ला, देवनागरी, गुजराती, कन्नड, मलयालय, ओडिया, पंजाबी, तमिल तथा तेलुगु की आवश्यकताएँ पूरी की गई हैं।
* आधारभूत कैरेक्टर को इस प्रकार से क्रमवार प्रस्तुत किया जाता है कि किसी भी लिपि में समान रूप से सीधे छपाई की जा सकती है।
* ऑकी कूटों को प्रदर्श हेतु इस्फोक में परिवर्तित करना आवश्यक है। इसे चयनित लिपि के लिए इस्फा कलनविधि के माध्यम से किया जाता है।
* ऑकी पाठ को किसी भी लिपि में प्रदर्शित किया जा सकता है। दूसरी लिपि का चयन करने भर से ही उसमें लिप्यंतरण हो जाता है।
* ऑकी कूट का प्रयोग संप्रेषण मीडिया जैसे टेलेक्स में, पाठ के इष्टतम स्थानांतरण हेतु किया जाता है। एएलपी शब्द संसाधित्र ऑकी-कूट का प्रयोग आंतरिक रूप से करता है जिससे वर्णमाला के स्तर पर समुचित संपादन तथा वर्तनी की अनन्य प्रस्तुति करना संभव हो सके।
* विद्वान विंडोज अनुप्रयोग ऑकी का रखरखाव सीधे-सीधे करने में अक्षम हैं क्योंकि प्रदर्श के रखरखाव हेतु इसमें बौद्धिक कलनविधि की आवश्यकता पड़ती है। लेकिन, वे इस प्रयोजन के लिए निर्मित इस्फोक कूट का रखरखाव कर सकते हैं। इस

प्रकार, एएलपी से विंडोज अनुप्रयोग में पाठ स्थानांतरित करते समय ऑकी तथा इस्फोक के बीच परिवर्तन किया जाना आवश्यक है।

* विंडोज अनुप्रयोग में ऑकी कुंजीपटल का प्रयोग करते हुए इस्फोक पाठ का टंकण सीधे करना संभव है। यह परंपरागत कुंजीपटल ड्राइवर से किया जाता है जो ऑकी से इस्फोक में परिवर्तन आंतरिक रूप से करता है।

(iii) इस्फा - Intelligence based Script to Font Algorithm.
फोंट कलनविधि हेतु बुद्धि आधारित लिपि

शब्द सदैव अपने मूलभूत ऑकी कैरेक्टर के रूप में ही टाइप किए जाते हैं। लेकिन इन्हें मूलभूत **इस्फोक** आकृतियों का प्रयोग करते हुए प्रदर्शित करना पड़ता है। ऑकी कूटों को समुचित इस्फोक कूट में बदलने के लिए कलनविधि की आवश्यकता पड़ती है। यह कलनविधि **इस्फा कलनविधि** है।[19]

(iv) इस्फोक ISFOC -(क) Indian Standard for Font Coding
फोंट कूटीकरण के लिए भारतीय मानक
(ख) Intelligence based Script Font Code
बुद्धि आधारित लिपि फोंट कूट

स्मार्ट कंप्यूटिंग (जुलाई,2001) के अनुसार इस्फोक का अर्थ है **Indian Standard for Font Coding अर्थात फोंट कूटीकरण के लिए भारतीय मानक**। कंप्यूटर पर डेटा को चित्रित रूप में दर्शाने के लिए 1990 तक बहुत से डेवलपरों ने अनेक भारतीय भाषाओं में फोंट तैयार किए। परंतु अधिकांश फोंट अन्य पैकेज से इनके उपयोग को रोकने हेतु एक दायरे में सिमटे होते थे। सभी इस समस्या का हल करना चाहते थे क्योंकि एप्लीकेशन बनाने वाले सभी फोंट निर्माताओं के अनुसार उस एप्लीकेशन में परिवर्तन नहीं कर सकते थे। इसके हल हेतु C-DAC ने **सभी प्रचलित लिपियों को सुंदरता तथा समरूपता देने के लिए सभी वर्णों के स्लाइस या ग्लिफ मानक निर्धारित किए।** इन्हें ISFOC (इंडियन स्टैंडर्ड फॉर फोंट कोडिंग) के नाम से जाना जाता है। यह कोड चार्ट प्रत्येक लिपि के लिए भिन्न है जो कि सभी ग्राफिकल एनवायरमेंट के लिए आवश्यक बिट मैप, ट्रू टाइप फोंट आदि के लिए उपयुक्त है। आज लगभग सभी एप्लीकेशन डेवलपर इसी कोडिंग स्कीम का प्रयोग कर रहे हैं।[20]

आई.एस.एम. यूजर्स मैनुअल के अनुसार **इस्फोक (ISFOC)** से अभिप्राय है **Intelligence based Script Font Code (बुद्धि आधारित लिपि फोंट कूट)**। इस्फोक कूटित कैरेक्टर समुच्चय (सैट) है जिसमें लिपि को प्रस्तुत करने के लिए आवश्यक सभी मूलभूत आकृतियाँ निहित हैं। इन आकृतियों को लिपि के किसी शब्द को बनाने के लिए रैखिक रूप से अतिव्याप्त (ओवरलेप) किया जा सकता है। प्रत्येक इस्फोक कैरेक्टर जिगसा (Jigsaw) पहेली की तरह है जो स्वयं में पूर्ण अक्षर नहीं होता है। प्रत्येक **इस्फोक समुच्चय** (सैट) में अधिकतम 188 कैरेक्टर होते हैं। यह अधिकांश लिपियों के लिए पर्याप्त है। लेकिन, कुछ लिपियों को इससे अधिक की आवश्यकता होती है।

इस्फोक के रूप में समाधान

इस्फोक, 188-कैरेक्टर की सीमा में रहते हुए, सौंदर्य-बोध और सुरुचिपूर्ण लिप्यंतरण के लिए मानक समुच्चय (सैट) है। एक ओर इस्फोक-कूट-समुच्चय आधारभूत आकृतियों को विनिर्दिष्ट करती हैं, तो दूसरी ओर इस्फा कलनविधि इनका समुचित संयोजन करने का तरीका विनिर्दिष्ट करती है। इन इष्टतम विनिर्देशों की सहायता से, फोंट डिजाइनर जटिल आधारभूत कैरेक्टर सैट डिजाइनों की परवाह किए बिना विभिन्न प्रकार के फोंट स्टाइल प्रस्तुत कर सकते हैं।

इस्फोक योजना सुनिश्चित करती है कि विभिन्न मीडिया माध्यमों जैसे समाचारपत्र, पत्रिका, पुस्तक, शब्दकोश तथा टी.वी. के लिए ये फोंट इष्टतम हों। ये फोंट निम्न रिजोल्यूशन (मात्रक) कंप्यूटर मानीटर से ले कर अत्यंत उच्च रिजोल्यूशन फोटो-टाइपसैटर तक की जरूरतों को पूरा करते हैं। अंत में, उपयोक्ता को वर्णमाला कुंजीपटल तथा गुणवत्तापूर्ण फोंटों की विविधता के रूप में सर्वश्रेष्ठ मानवीय इंटरफेस प्राप्त होता है। इसका सर्वश्रेष्ठ सौदा यह है कि अब उपयोक्ता इन सबका उपयोग पर्सनल कंप्यूटर पर फोंट-अभिमुखी पैकेजों के साथ कर सकता है।

जटिल लिपियों का पबंधन अँग्रेजी की तरह किए जाने पर, उनका उपयोग भविष्य में उज्जवल हो सकता है। अँग्रेजी के लिए नाल प्रकार के फोंट उपलब्ध हैं। अब हम इस्फोक फोंट की सहायता से, जटिल लिपियों के लिए भी इसकी उम्मीद कर सकते हैं।

इस्फोक संकल्पना

जटिल लिपियों को प्रस्तुत करने का आधारभूत दर्शन निम्नानुसार है :

लिपि भाषांतरण दर्शन

* शब्द की वर्तनी टंकित करते समय यह अंतर्ज्ञानी तथा तार्किक होता है।
* शब्द की वर्तनी उसके उच्चारण के क्रम में आधारभूत वर्णमाला से बनती है।
* लिपि की आधारभूत वर्णमाला अपने आवश्यक विशेष प्रतीकों तथा विराम चिंहों के साथ मिल कर ऑकी का गठन करते हैं। ऑकी में अक्षरों को उनके वर्णमाला-छँटाई-क्रम के अनुसार व्यवस्थित किया गया है। ऑकी में आस्की कैरेक्टर समुच्चय (सैट) भी निहित होते हैं।
* लिपि में उपलब्ध आधारभूत आकृतियाँ होती हैं। ये सीधे टंकण करने हेतु बहुत बोझिल होते हैं।
* इस्फा द्वारा ऑकी वर्तनी प्रस्तुत की जा सकती है तथा शब्द के प्रदर्श हेतु आवश्यक इस्फोक कूट क्रम को जनित किया जा सकता है।
* अँग्रेजी जैसी सरल लिपि के लिए ऑस्की कूट स्वत: ऑकी तथा इस्फोक की पूर्ति कर देता है।
* लेकिन भारतीय लिपियों जैसे अधिकांश जटिल अरैबिक लिपियों के मामले में ऑकी, इस्फोक तथा इस्फा कलनविधि के लिए पृथक कूट की आवश्यकता पड़ती है।

इस्फोक मानक

* आधारभूत आकृतियों तथा उनके संयोजन के लिए लिपि-मानक फोंट के प्रारूपण की सुविधा प्रदान करते हैं।
* इस्फोक आवश्यक आधारभूत आकृतियों को परिभाषित करते हुए लिपि की आधुनिक भाषांतरण शैली प्रस्तुत करती है।
* आधारभूत आकृतियों का चयन इस प्रकार किया जाता है कि वे लिपि में नाना प्रकार के फोंट शैलियों को प्रस्तुत कर सकते हैं।
* लिपि के लिए निर्धारित इस्फोक, इस्फा से संबद्ध है, जो आधारभूत आकृतियों का प्रयोग करते हुए शब्द संयोजन के लिए मानक तरीका परिभाषित करता है।
* लिपि के लिए विकसित सभी फोंट परस्पर संगत होते हैं। उपयोक्ता अपनी पसंद के फोंट में पाठ देख सकता है।
* चूँकि इस्फोक फोट का संयोजन रैखिक रूप में किया जाता है, अत: इनका प्रयोग वर्तमान अँग्रेजी अनुप्रयोगों के साथ किया जा सकता है तथा वर्तमान लेसर प्रिंटर्स तथा टाइपसैटर्स पर प्रिंट किया जा सकता है।
* इस्फोक ऐसा कूट समुच्चय (सैट) उपलब्ध कराता है जो ग्राफिक-उन्मुख वातावरण जैसे एमएस विंडोज तथा मैकिंटोश में जटिल लिपियों का समावेश कराने में सहायक है।
* इस्फोक स्वच्छतम लिपि भाषांतर उपलब्ध कराते हुए, ऑकी कुंजीपटल से बौद्धिक मानवीय इंटरफेस की अनुमति भी देता है।

कैरेक्टरों की संख्या तथा उनकी प्रस्तुति

8-बिट कूट 256 कैरेक्टरों को प्रस्तुत कर सकता है। आइसो (ISO) 8-बिट कूट की परिभाषा के अनुसार, आरंभ में 32 नियंत्रण कैरेक्टर हैं। इनके अलावा कूट के निचले आधे हिस्से में दो अन्य नियंत्रण कैरेक्टर हैं। ये हैं - स्पेस कैरेक्टर (20एच) तथा डिलीट कैरेक्टर (7एफएच)। इस प्रकार निचले आधे हिस्से में प्रदर्श प्रयोजन हेतु केवल 94 कैरेक्टर उपलब्ध होते हैं।

आइसो 8-बिट कूट में, ऊपर का आधा हिस्सा एक और नियंत्रण कैरेक्टर युक्त होने के कारण निचले आधे हिस्से का दर्पण होता है। इस प्रकार प्रदर्श कैरेक्टर हेतु केवल 188 कूट उपलब्ध होते हैं।

कुछ अनुप्रयोग जैसे वैंचुरा (गोल्ड) आइसो 8-बिट संस्तुतियों का अनुपालन पूर्णतया करते हैं तथा एक कैरेक्टर समुच्चय (सैट) में केवल 188 कैरेक्टरों की ही सुविधा देते हैं। लेकिन पोस्टस्क्रिप्ट प्रिंटर्स नियंत्रण प्रयोजन के लिए केवल निचले 32 कैरेक्टरों को आरक्षित रखते हैं तथा शेष 224 कैरेक्टरों का प्रयोग करने देते हैं। इस प्रकार एक फोंट स्वयं को 188 कैरेक्टर तक सीमित कर लेता है जिससे इन दोनों प्रिंटरों के साथ इसका प्रयोग किया जा सके।

इस्फोक कैरेक्टर समुच्चय (सैट)

स्क्रिप्ट कैरेक्टर समुच्चय (सैट)

यह प्राथमिक कैरेक्टर समुच्चय (सैट) है जिसमें बारंबार प्रयुक्त अधिकांश भाषाई कैरेक्टर तथा प्रतीकों एवं संख्याओं के समुच्चय (सैट) होते हैं। यह प्रतीक समुच्चय (सैट) सभी इस्फोक कैरेक्टर समुच्चय (सैट) में समान होते हैं, लेकिन इसके कुछ अपवाद भी होते हैं।

मिलान अँग्रेजी कैरेक्टर समुच्चय (सैट)

यह अँग्रेजी फोंट से मिलान करने के लिए सह-कैरेक्टर समुच्चय (सैट) है जिसमें निचले आधे हिस्से में आस्की कैरेक्टर होते हैं तथा ऊपर के आधे हिस्से में रोमन लिप्यंतरण हेतु उच्चारण कैरेक्टर होते हैं।

संपूरक कैरेक्टर समुच्चय (सैट)

संपूरक कैरेक्टर समुच्चय (सैट), आधारभूत लिपि कैरेक्टर समुच्चय (सैट) का विस्तारित समुच्चय (सैट) है, जिसमें सामान्य प्रयोग में नहीं आने वाले समुच्चय (कंजंक्ट) तथा प्रतीक होते हैं।[21]

5.2.4 प्राकृतिक भाषा तथा कुंजीपटल का भाषा-वैज्ञानिक अध्ययन

संसार में हजारों भाषाएँ बोली, लिखी तथा पढ़ी जाती हैं और हर भाषा की अपनी-अपनी लिपि है। एक ही लिपि को कई भाषाओं में लिखा जाता है। एक ओर जहाँ संस्कृत और फ्रेंच जैसी शास्त्रीय भाषाएँ हैं, दूसरी ओर अँग्रेजी एवं हिंदी जैसी संपर्क भाषाएँ हैं। अँग्रेजी एक प्रकार से अंतरराष्ट्रीय संपर्क भाषा है लेकिन सीमित रूप में ही। विश्व के लगभग सभी देशों में प्राथमिक कंप्यूटर के कुँजीपटल में रोमन लिपि एवं इंडो-अरेबियन अंकों का प्रयोग हुआ है एवं प्रोग्रामन भाषाओं में अँग्रेजी के शब्दों का। अब सब देशों ने अपनी राजभाषाओं के शब्द-प्रक्रमकों (Word-Processors) का विकास कर लिया है, जिनमें मुद्रण, पत्रलेखन आदि कार्यों के लिए विविध लिपियों का प्रयोग किया जाता है। लेकिन उच्च स्तर पर की सभी भाषाओं में फिलहाल अँग्रेजी का प्रयोग ही हो रहा है, हाँलाकि रूस और चीन जैसे देश इस बंधन से मुक्त हैं।

कंप्यूटर में किसी भी प्रकार की कोई बुद्धि नहीं होती वरन उसे प्रोग्रामन के द्वारा बुद्धि प्रदान की जाती है। कुछ आंतरिक एवं स्थाई प्रोग्रामन के द्वारा, कुछ बाह्य प्रोग्रामों के द्वारा। ये प्रोग्राम कई स्तरों की भाषाओं में लिखे जाते हैं।[22]

5.2.4(क) कंप्यूटर युग में देवनागरी लिपि की सार्थकता तथा गुण

(I) देवनागरी लिपि की सार्थकता

यह सर्वमान्य है कि उर्दू को छोड़ कर आधुनिक भारत की सभी लिपियाँ ब्राह्मी से विकसित हुई हैं। ब्राह्मी लिपि पांचवीं-छठी सदी में अस्तित्व में आ चुकी थी। यह लिपि अपने युग की सर्वाधिक वैज्ञानिक और आदर्श लिपि थी। यही कारण था कि खरोष्ठी आदि

अनेक-अनेक लिपियों के रहते हुए भी बौद्ध तथा जैन धर्म के प्रचारकों ने इसे ही अपने धर्म के प्रचार और ग्रंथों के निर्माण का प्रमुख माध्यम बनाया। ब्राह्मी लिपि संस्कृत और प्राकृत भाषाओं के लिए लगभग 350 ई.पू. से लेकर 350 ई. तक केवल भारत में ही नहीं अन्य अनेक पड़ोसी देशों में भी प्रयुक्त होती थी। चौथी शताब्दी के उत्तरार्ध में ब्राह्मी दो शैलियों में विभक्त हो गई - उत्तरी और दक्षिणी शैली। उत्तर भारत की सभी परवर्ती लिपियाँ ब्राह्मी की उत्तरी शैली से और दक्षिण भारत की सभी परवर्ती लिपियाँ दक्षिणी शैली से विकसित हुई हैं। कालांतर में इन लिपियों में इतना अंतर आ गया कि बिना सीखे उत्तर वालों के लिए दक्षिण की किसी लिपि को पढ़ना संभव न रहा और दक्षिण वालों के लिए उत्तर की किसी लिपि को पढ़ना संभव न रहा। इस अंतर का एक प्रमुख कारण कदाचित् लेखन-सामग्री की भिन्नता भी रहा है। दक्षिण भारत में ताड़ वृक्षों की बहुलता के कारण लेखन-सामग्री के रूप में ताड़पत्रों का प्रयोग किया जाता था और उस पर सीधे लिखने से ताड़पत्रों के फट जाने की आशंका रहती थी। इसलिए नोकदार सूखी कलम से वृत्ताकार रूप में लिखने की परंपरा विकसित हुई। इसके विपरीत उत्तर भारत मे चपटी और खोखली कलम से वृत्त के बजाय ऊपर-नीचे की मात्राओं के साथ लिखने की परंपरा विकसित हुई।

चौथी शताब्दी में ब्राह्मी की उत्तरी शैली से **'गुप्त लिपि'** का जन्म हुआ। इसके अनेक वर्णों की आकृति देवनागरी के वर्णों से मिलती-जुलती है। गुप्त लिपि में वर्णों की शिरोरेखा का प्रयोग स्पष्ट रूप से मिलता है। छठी शताब्दी तक आते-आते गुप्त लिपि भी बहुत अधिक बदल गई। इस कारण इसके परिवर्तित रूप का नाम **कुटिल लिपि** रखा गया। इश लिपि का पचार छठी शताब्दी से नवीं शताब्दी तक संपूर्ण उत्तर भारत में था। इसी कुटिल लिपि से अन्य अनेक आर्यभाषाओं की लिपियों के साथ-साथ देवनागरी या नागरी का भी विकास हुआ। इस प्रकार देवनागरी लिपि का उद्‌भव आठवी-नवीं शताब्दी के आसपास कुटिल लिपि से हुआ। यह लिपि अपने उद्‌भव काल से ही सारे देश में प्रयुक्त होने लगी थी। इस प्रकार ब्राह्मी लिपि का वास्तविक राष्ट्रीय स्वरूप देवनागरी के रूप में अवतरित हुआ है। आज भी विश्व भर में प्रकाशित संपूर्ण संस्कृत साहित्य अधिकांशतः देवनागरी लिपि में ही पाया जाता है। हिंदी और उसकी अनेक क्षेत्रीय बोलियां देवनागरी में लिखी जाती हैं। नेपाली और मराठी भाषाओं की लिपि भी देवनागरी है। यह लिपि सर्वप्रथम आठवीं शताब्दी में दक्षिण में ही विकसित हुई और तदनंतर दसवीं शताब्दी के उत्तरार्ध में चालुक्य, यादव और विजयनगर के तीन राजकुलों ने भी अनेक शिलालेखों में इसी लिपि का बहुत प्रयोग किया है और दक्षिण में संस्कृत के हस्तलिखित ग्रंथों के लिए अब भी इसका प्रयोग होता है। देवनागरी इस लिपि का उत्तर भारतीय रूप है। दक्षिण में इसे **नंदिनागरी** कहते हैं।

देवनागरी का वर्तमान स्वरूप शताब्दियों के **प्रयोग पर आधारित क्रमिक विकास का प्रतिफल है।** देवनागरी लिपि **ध्वन्यात्मक लिपि है। इस प्रकार की लिपि, लिपि के अद्यतन विकास की अंतिम कड़ी है।** लिपि के इतिहास में ध्वन्यात्मक लिपि का स्थान सबसे ऊंचा है। इसकी सबसे बड़ी विशेषता है इसकी **वैज्ञानिक वर्ण-व्यवस्था। वर्णों का वर्गीकरण उच्चारण प्रक्रिया और उच्चारण स्थान को ध्यान में रखकर किया गया है।** सबसे पहले स्वर और फिर व्यंजन। स्वरों में भी सबसे पहले ह्रस्व और फिर उसके दीर्घ रूप। पहले मूलस्वर और फिर संयुक्त स्वर। इसी प्रकार व्यंजनों के भी पांचों वर्ग उच्चारण के अनुसार व्यवस्थित किए गए हैं। सबसे पहले कंठ का स्थान है, इसलिए कंठ्य ध्वनियों को पहले रखा गया

है। फिर क्रम से तालव्य, मूर्धन्य, वर्त्स्य, दंत्य और ओष्ठ्य ध्वनियों को स्थान दिया गया है। इसके विपरीत्त रोमन और अरबी लिपि में वर्णमाला का क्रम यादृच्छिक है।

देवनागरी लिपि की अन्यत्तम विशिषत्ता यह है कि इसमें **एक ध्वनि के लिए एक ही ध्वनि प्रतीक** है। यही कारण है कि इस लिपि में लिखी गई भाषा किसी दूसरे व्यक्ति द्वारा ज्यों की त्यों पढ़ी जा सकती है। इस विशेषता के कारण अन्य भाषा भाषी के लिए इसे सीखना भी कठिन नहीं है। इसके विपरीत रोमन लिपि में एक ही ध्वनि को कई ध्वनि संकेत प्रस्तुत करते हैं। जैसे 'क' के लिए 'C' और 'K'। उर्दू में भी इसी प्रकार कठिनाई होती है। जैसे जे, ज़ाल, ज़ोय, ज़्वाद, चार ध्वनि प्रतीक एक ही ध्वनि 'जे' को प्रस्तुत करते हैं।

अपनी इन तमाम स्वरूपगत विशेषताओं के बावजूद भारत के स्वतंत्र होने तक देवनागरी लिपि का प्रयोग मूलतः साहित्य सृजन तक ही सीमित रहा, किंतु स्वतंत्र भारत में हिंदी के राजभाषा के पद पर प्रतिष्ठित होने के बाद इस लिपि की संभावनाओं के द्वार खुल गए। इस समय नागरी केवल हिंदी की ही लिपि नहीं है, संस्कृत तथा अन्य कई भारतीय भाषाओं की भी लिपि है। जिस प्रकार हिंदी राष्ट्र की संपर्क भाषा के रूप में सबको एकता के सूत्र में बाँध सकती है, उसी प्रकार **नागरी संपर्क-लिपि के रूप में भारतीय भाषाओं को एकसूत्रता में बांध सकती है।** भारतीय भाषाओं की विशिष्ट ध्वनियों को नागरी लिपि में समाहित करने के लिए केंद्रीय हिंदी निदेशालय की ओर से सन् 1966 में परिवर्धित देवनागरी नामक एक पुस्तिका प्रकाशित की गई, जिसमें दक्षिण भारत की भाषाओं तथा कश्मीरी के विशिष्ट व्यंजनों के अतिरिक्त सिंधी और उर्दू की विशिष्ट ध्वनियों के लिप्यंतरण के लिए देवनागरी में अपेक्षित परिवर्धन किया गया। इसके अलावा नागरी लिपि में एकरूपता लाने के उद्देश्य से भारत सरकार के शिक्षा मंत्रालय ने सन् 1961 में एक विशेषज्ञ समिति गठित की। इस समिति की सिफारिशों के अनुरूप सन् 1967 में नागरी लिपि का मानकीकरण कर दिया गया।

देवनागरी लिपि एक वैज्ञानिक लिपि है। भारतीय भाषाएं विश्व की अनेक भाषाओं की तुलना में वाक्य-विज्ञान, ध्वनि विज्ञान और रैखिक दृष्टि से अधिक सुनियोजित हैं। अमरिकी वैज्ञानिक श्री रिक ब्रिग्ज़ की यह धारणा है कि संस्कृत भाषा

कंप्यूटर प्रोग्राम की दृष्टि से आदर्श भाषा है। इसलिए देवनागरी लिपि में कंप्यूटर पर कार्य करना कठिन नहीं है, किंतु आवश्यक सॉफ्टवेयर के अभाव में 1965 से पूर्व इसकी कल्पना भी नहीं की जा सकती थी। सर्वप्रथम अमेरिका में ज़ीरोक्स कॉरपोरेशन के सहयोग से श्री जोज़फ डी. बेकर ने इस दिशा में आरंभिक प्रयास किए। उन्होंने चीनी, जापानी, हिब्रू, हिंदी, पंजाबी, उर्दू आदि अनेक भाषाओं में सर्वप्रथम शब्द संसाधन का कार्य आरंभ किया। 'स्टार' नाम से प्रचलित इस प्रणाली के अंतर्गत **मूल पाठ का कुंजीयन रोमन लिपि में करके अपेक्षित लिपि में उसका लिप्यंतरण और मुद्रण किया जाता है।** इस प्रकार भारतीय भाषाओं से संबंधित पाठ्य सामग्री का भाषिक विश्लेषण अब तक रोमन लिपि के माध्यम से ही किया जाता रहा है।

रोमन लिपि के माध्यम से निरूपण करने से उत्पन्न अधिकांश समस्याओं का मूल कारण यह था कि रोमन लिपि ध्वन्यात्मक नहीं है, जबकि अधिकांश भारतीय लिपियाँ अपने स्वरूप में ध्वन्यात्मक हैं। इसमें संदेह नहीं कि कुछ ध्वन्यात्मक प्रतीकों को जोड़ने से हमें रोमन लिपि के माध्यम से पाठों के निवेश और निर्गम से सरलता भी मालूम पड़ेगी, क्योंकि हम अभी दैनिक जीवन में रोमन के माध्यम से विभिन्न प्रकार के कार्य करने के अभ्यस्त हो चले हैं, किंतु भारतीय भाषाओं की विशिष्ट ध्वनियों को ज्यों का त्यों व्यक्त करने में अनेक कठिनाइयाँ उत्पन्न हो सकती हैं। भारतीय वर्णमाला के क्रम के अनुसार अनुक्रमणिका तैयार करने में भी कठिनाई होगी। इसके अलावा भाषाओं के विश्लेषण और संसाधन के लिए रोमन लिपि का ज्ञान अपरिहार्य हो जाएगा, जिसके फलस्वरूप अँग्रेजी ज्ञान न रखने वाला आम भारतीय, कंप्यूटर पर अपनी भाषाओं के माध्यम से कार्य न कर सकेगा।

'विविधता में एकता' भारतीय भाषाओं और लिपियों की मूलभूत विशेषता है। इस मूलभूत एकता को कंप्यूटर वैज्ञानिकों ने भी पहचाना है और तदनुसार सभी भारतीय भाषाओं के लिए समान कुंजीपटल का विकास किया गया है। वस्तुतः किसी भी लिपि का अक्षर एक प्रतीक होता है। वह दो प्रकार के संदेशों का वहन करता है : बाहरी और आंतरिक संदेश। बाहरी संदेश इस प्रतीक का रेखीय (ग्राफिक) स्वरूप होता है और आंतरिक संदेश वर्णमाला के अक्षर के रूप में होता है। यह एक विशिष्ट ध्वनि को दर्शाता है। जैसे हिंदी के 'प्र' और 'अ' अक्षर रेखीय दृष्टि से भिन्न हैं, किंतु हम उन्हें एक ही अक्षर के रूप में पहचानते हैं जबकि उनका आंतरिक संदेश ध्वनि के रूप में एक ही है। सभी भारतीय भाषाओं में वर्णमाला का क्रम समान होने के कारण उनका आंतरिक संदेश भी समान ही होता है, किंतु रेखीय रूप में बाहरी संदेश में कमोबेश भिन्नता रहती है। इस मूलभूत आंतरिक समानता को केंद्र में रख कर ही इलैक्ट्रॉनिकी विभाग, भारत सरकार ने सभी भारतीय भाषाओं (उर्दू को छोड़कर) के लिए समान कोड स्वीकार किया है। यह कोड वस्तुतः अँग्रेजी के मानक आस्की-7 (ASCII-7) कोड का विस्तृत रूप है। इसे इस्की-8 (ISCII-8) कोड कहा जाता है। इसमें रोमन लिपि के अक्षरों को भी समाहित किया जाता है। इसका सबसे बड़ा लाभ यह है कि इस तकनीक के माध्यम से किसी भी भारतीय लिपि के ज़रिए सभी भारतीय भाषाओं के अध्ययन-विश्लेषण का कार्य किया जा सकता है। भारतीय भाषाओं में

परस्पर लिप्यंतरण इस तकनीक की अन्यतम विशेषता है। इससे पुस्तकालयों में विभिन्न भारतीय भाषाओं की पुस्तकों की अनुक्रमणिका तैयार की जा सकती है। कोश निर्माण के लिए भारतीय वर्णमाला के अनुसार अक्षरों का अनुक्रम भी बनाया जा सकता है। पर्यायवाची कोश या थिसॉरस के लिए अपेक्षित समनुक्रमणिका (Concordance) का निर्माण भी हिंदी और अन्य भारतीय भाषाओं में किया जा सकता है।

वस्तुतः देवनागरी लिपि की अनंत संभावनाओं का स्रोत उसके वैज्ञानिक स्वरूप में ही निहित है। इसका प्रमाण इस तथ्य से मिलता है कि विश्व के अनेक उन्नत देशों में आज **कंप्यूटर के माध्यम से वाक् संश्लेषण और अभिज्ञान के प्रयास** किए जा रहे हैं, किंतु इस संदर्भ **में जो आशातीत सफलता हिंदी के संदर्भ में प्राप्त हुई है**, वह कदाचित् अन्य भाषाओं और लिपियों के संदर्भ में असंभव नहीं तो कठिन अवश्य है।

इलैक्ट्रॉनिकी विभाग, भारत सरकार के सहयोग से आई.टी. मद्रास में एक प्रणाली का विकास किया जा रहा है, जिसके माध्यम से हिंदी भाषा में उच्चरित वाक्यों को कंप्यूटर की सहायता से नागरी लिपि में अंतरित किया जाएगा और इसी प्रकार नगरी लिपि में अंकित पाठों को कंप्यूटरों के मॉनिटर पर कुंजीयन करके ध्वनि में रूपांतरित किया जाएगा। इन दोनों ही परियोजनाओं में आंशिक सफलता मिलने लगी है। इस सफवत्ता का मुख्य कारण है, नागरी लिपि की वैज्ञानिकता। इससे यह स्पष्ट है कि आज के कंप्यूटर युग मे देवनागरी लिपि की सार्थकता असंदिग्ध है। (दिल्ली 1991)[23]

(II) देवनागरी लिपि के गुण

नागरी लिपि अपनी अनेक कमियों के बावजूद अनेक दृष्टियों से अपेक्षाकृत अधिक उपयोगी, गुणयुक्त तथा वैज्ञानिक है। यहाँ उसके कुछ मुख्य गुणों को रोमन तथा उर्दू की तुलना में देखा जा सकता है।

वर्णमाला का वर्गीकरण

विश्व के किसी भी कोने में प्रयुक्त वर्णमाला उतने वैज्ञानिक रूप में विभाजित या वर्गीकृत नहीं है, जितनी नागरी आदि भारतीय लिपियाँ या उनसे संबद्ध या विकसित लिपियाँ हैं। उर्दू या रोमन लिपि भी इस कमी का अपवाद नहीं है। इनमें भी स्वर एवं व्यंजन (अलिफ़, बे; डी, ई, एफ़ आदि) या व्यंजन के वैज्ञानिक और युक्तिसंगत वर्ग (लाम, मीम; सी, डी, आदि) मिले-जुले रूप में रखे गए हैं। नागरी में इस प्रकार की गड़बड़ी नहीं है। **स्वर** अलग हैं, और **व्यंजन** अलग। स्वरों में भी **ह्रस्व-दीर्घ** के युग्म (अ-आ, इ-ई, उ-ऊ) साथ-साथ हैं। समवेततः प्रारंभ में **मूल स्वर** (अ, आ, इ, ई, उ, ऊ) हैं, और उसके बाद **संयुक्त स्वर** (ए, ऐ, ओ, औ)। व्यंजनों, विशेषतः **स्पर्श** एवं **अनुनासिक** का विभाजन तो और भी वैज्ञानिक है। क, च, ट, त, प के वर्ग **स्थान पर** आधारित हैं, और हर वर्ग के व्यंजन **घोषत्व के आधार पर** दो प्रकार के हैं : प्रथम दो अघोष तथा अंतिम तीन घोष। इनके साथ ही इनके वर्गीकरण या विभाजन में **प्राणत्व** का भी ध्यान रखा गया है। पहले, तीसरे और पाँचवें, अल्पप्राण हैं तथा दूसरे, चौथे महाप्राण।

लिपिचिंहों के नाम ध्वनि के अनुरूप

नागरी में यह बहुत बड़ा गुण है कि जो लिपि-चिंह जिस ध्वनि का द्योतक है, उनका नाम भी वही है : जैसे आ, ओ, क, ब, थ, आदि। रोमन में ऐसा न होने से भाषा सीखने वाले व्यक्ति को बहुत कठिनाई होती है, और याद रखना पड़ता है कि कौन चिंह किस ध्वनि के लिए आता है। इस दृष्टि से रोमन में लिपि-चिंह तीन प्रकार के हैं : (1) लिपि-चिंह नाम का या उसकी प्रथम ध्वनि का काम करता है, जैसे ए, बी, सी, डी, जी, जे, के, ओ, पी, क्यू, टी, वी, जेड। (2) लिपि-चिंह प्रायः दूसरी ध्वनि का काम करता है, जैसे एफ़, आइ, एल, एम, एन, आर, एस, यू, एक्स। (3) लिपि-चिंह नाम में आई किसी भी ध्वनि का काम न कर किसी अन्य ध्वनि का कार्य करता है, जैसे एच (ह), ए (अ, आ, ऐ), सी (क), डब्ल्यू (व), वाइ (य)। यों तो नागरी की तुलना में तीनों ही प्रकार के नाम अवैज्ञानिक हैं, किंतु अंतिम वर्ग तो सर्वाधिक अवैज्ञानिक है। उर्दू लिपि में यह अवैज्ञानिकता इतनी अधिक तो नहीं है किंतु नागरी की तुलना में अधिक अवश्य है। उसके अधिकांश चिंह प्रायः प्रथम वर्ग में आते हैं।

एक ध्वनि के लिए एक लिपि-चिंह

अच्छी या वैज्ञानिक लिपि में यह गुण बहुत आवश्यक है। यों किसी भी लिपि में अपने आप यह गुण सर्वदा नहीं रह सकता। इसका कारण यह है कि किसी भाषा के प्रसंग में उच्चारण के स्तर पर ही हर ध्वनि के लिए एक चिंह की बात की जा सकती है, और **भाषा की ध्वनियाँ उच्चारण की दृष्टि से परिवर्तित होती रहती हैं,** जिसका आशय यह हुआ कि एक समय किसी लिपि में यह गुण हो सकता है, और दूसरे समय उच्चारण में अंतर हो जाने के कारण इस गुण का अभाव हो सकता है। इसी प्रकार एक भाषा के प्रसंग में लिपि में यह गुण हो सकता है, और दूसरी भाषा के प्रसंग में नहीं हो सकता है। इसका आशय यह हुआ कि **इस गुण का संबंध लिपि के अपने आंतरिक गुण से विशेष न होकर उसके प्रयोग से है।**

पहले हम **उर्दू लिपि** की बात लें। उर्दू लिपि में यह गुण नहीं है। उदाहरणार्थ 'स' ध्वनि के लिए उर्दू में तीन लिपि-चिंह हैं : 'से', 'स्वाद', 'सीन'। इसी प्रकार 'ज़' के लिए 'ज़े', 'ज़ाल', 'ज़ोय', 'ज़्वाद' अर्थात् चार; या यदि 'झे' को भी मिला लें तो पाँच लिपि-चिंह हैं, 'त' और 'ह' के लिए भी दो (ते, तोय) दो (बडी हे, छोटी हे) हैं। यों मूल अरबी में इन सबका स्वतंत्र उच्चारण था, किंतु उर्दू मे आकर वहाँ की वैज्ञानिकता अवैज्ञानिकता में परिवर्तित हो गई है।

अँग्रेजी में भी यह गडबडी है। उदाहरण के लिए, 'क' ध्वनि के लिए कभी K (Kite), कभी C (Coat), कभी Ch (Chemist), कभी Ck (Check), कभी Q (Cheque) तथा कभी अंशत: X (box) का प्रयोग होता है। इसी प्रकार श (Sh, si, ti आदि), फ़ (f, ph), इ (i, y - System, ee-committee, o-women, u-minute आदि), ओ, उ, अ, स (S, C अंशत: X) आदि के लिए भी एक से अधिक लिपि-चिंहों का प्रयोग होता है। इस प्रसंग में यह उल्लेख्य है कि यह गडबडी अँग्रेजी में प्रयुक्त रोमन की है, किंतु **मूल रोमन** भी इन अवगुणों (स=c, s, x; क=k, x आदि) से पूर्णत: शून्य नहीं है। **नागरी लिपि में** मूलत: यह अवगुण बिल्कुल नहीं था। अब हिंदी आदि के प्रयोग में कुछ अवगुण (ऋ-रि, श-ष) अवश्य आ गया है, किंतु उर्दू या अँग्रेजी की तुलना में यह बहुत ही कम है।

उर्दू तथा रोमन आदि में एक लिपि-चिंह एक से अधिक ध्वनियों को व्यक्त करते हैं। उदाहरणार्थ अलिफ़, 'अ' को भी व्यक्त करता है, 'आ' को भी, और कभी-कभी 'इ' (बिल्कुल) को भी। इसी प्रकार 'वाव' व, ऊ, ओ, औ; तथा 'ये' य, ए, ऐ आदि। इसका परिणाम यह हुआ है कि कभी-कभी उर्दू पढना टेढी खीर हो जाता है। जैसे 'आलू बुख़ारे' को किसी ने 'उल्लू बेचारे' पढा तो 'अब्बा अजमेर गए' को 'अब्बा आज मर गए' या 'कोडे' को 'कूडे' आदि। अँग्रेजी भी इस दृष्टि से उर्दू से पीछे नहीं है। इसमें भी बहुत से लिपि-चिंह ऐसे हैं जो एक से अधिक ध्वनियों को व्यक्त करते हैं। उदाहरणार्थ C=स, क; g=ग, ज; a=अ, आ, ऐ आदि; e=ए, इ (पैकिट); u=अ, उ,यो आदि। यही कारण है कि अँग्रेजी में उच्चारण याद करना पडता है : put = पुट, but = बट। इस प्रकार अँग्रेजी-उर्दू आदि में शब्द की वर्तनी और उसके उच्चारण का वह सहज एवं वैज्ञानिक संबंध नहीं है जो नागरी में है । इस तरह नागरी में यह लेखन और उच्चारण का अपेक्षाकृत अधिक

वैज्ञानिक संबंध निश्चय ही बहुत बड़ा गुण है। अर्थात् **अपवादों को छोड़कर** इसमें एक ध्वनि के लिए एक लिपि-चिंह है, या एक लिपि-चिंह एक ही ध्वनि को व्यक्त करता है, जबकि उर्दू या रोमन में यह बात नहीं है।

एक विशेषज्ञ ने अँग्रेजी के लिए एक नई वर्णमाला भी तैयार की है जिसका नाम **'क्विक स्क्रिप्ट'** है। इस नई वर्णमाला में 40 लिपि-चिंह हैं। अँग्रेजी के लिए, इन दो के अतिरिक्त भी कुछ अन्य वर्णमालाएँ बनाने के प्रयास भी हो चुके हैं, और ये सारे प्रयास अँग्रेजी में प्रयुक्त रोमन की अपूर्णता के ही ज्वलंत प्रमाण हैं।

लिपि-चिंहों की पर्याप्तता

विश्व की अधिकांश लिपियों में चिंह पर्याप्त नहीं है। अँग्रेजी में ध्वनियाँ 40 से ऊपर हैं, किंतु केवल 26 लिपि-चिंहों से काम चलाना पड़ता है। उर्दू में भी ख, घ, छ, झ, ठ, ढ, थ, ध, फ, भ, आदि के लिए लिपि-चिंह नहीं हैं, और 'हे' से मिला कर इनका काम चलाते हैं। इस दृष्टि से नागरी तथा ब्राह्मी से विकसित अन्य कई भारतीय लिपियाँ पर्याप्त संपन्न हैं। नागरी का प्रयोग जिन-जिन भाषाओं के लिए हो रहा है, यदि अपवादतः कुछ नव विकसित ध्वनियों (ह्रस्व ऍ, ओ ॅ (ऑ), न्ह, म्ह, ल्ह,) को छोड़ दें तो इस दृष्टि से इसमें कोई कमी नहीं है। ऐसी स्थिति में यदि हिंदी आदि आधुनिक भारतीय भाषाओं को रोमन या उर्दू आदि में लिखना हो तो अनेक ध्वनियों के लिए प्रायः दो लिपि-चिंहों को मिला कर लिखने की परेशानी उठानी पड़ेगी, जबकि नागरी में उन सभी के लिए स्वतंत्र चिंह हैं।

ह्रस्व तथा दीर्घ स्वर के लिए स्वतंत्र चिंह

इस दृष्टि से रोमन और नागरी का कोई मुकाबला नहीं है। रोमन में ए (A) अक्षर से 'अ' का भी काम लेते हैं, और 'आ' का भी। **'kam'** 'कम' भी है और 'काम' भी, या **'kala'** 'कला' भी है और 'काला' भी। अँग्रेजी लेखन में यू (u) 'उ' (put) भी है और ऊ (truth) भी। नागरी में अ-आ, इ-ई, उ-ऊ, में स्पष्ट अंतर है, अतः रोमन की भाँति भ्रम की गुंजाइश नहीं है। दो 'ई' (ee) या दो 'ओ' (oo) से 'ई' 'ऊ' को व्यक्त करना वस्तुतः रोमन लिपि का अपना मूलभूत गुण नहीं है, जैसा कि नागरी में है। यह पद्धति तो एक प्रकार से बैसाखी है जिसके सहारे लँगड़ी रोमन लिपि को काम चलाने के लिए खड़ी कर लेते हैं। उर्दू में भी एक सीमा तक यह दुर्गुण है। वस्तुतः **दो लिपि-चिंहों को मिला कर किसी एक मूल ध्वनि को व्यक्त करना मूलतः अवैज्ञानिक पद्धति है। संयुक्त ध्वनि को ही एक से अधिक चिंह मिला कर प्रकट किया जाना चाहिए।**

मात्राओं का प्रयोग

नागरी लिपि में स्वर यदि स्वतंत्र रूप से आते हैं तो पूरे वर्ण-चिंह का प्रयोग होता है (आग, ईद, गाउन, आदि), किंतु व्यंजन के साथ, 'अ' के अतिरिक्त अन्य सभी का मात्रा रूप (नाग, लीद, चतुर आदि) प्रयुक्त होता है। इसके कारण चिंहों की संख्या में वृद्धि तो हो गई है, किंतु इससे यह सुविधा है कि स्थान कम घिरता है। यदि मात्राओं का प्रयोग न होता तो 'काल' को 'क्आल्अ' या 'कुली' को 'क्उल्ई' लिखना पड़ता। रोमन की

तुलना में भी इससे प्राय: कम घिरता है : काला kala बालू baloo। हाँ ऊपर-नीचे मात्रा के कारण कुछ असुविधा भी है।

नागरी के व्यंजन-चिंहों की आक्षरिकता

नागरी का हर व्यंजन-चिंह व्यंजन न होकर 'व्यंजन' और 'अ' स्वर का योग है। जैसे क=क् +अ; ख=ख्+अ आदि। लिपि का ऐसा होना 'आक्षरिकता' है। अर्थात् व्यंजन-चिंह वस्तुत: 'अक्षर' हैं। 'अक्षर' का यहाँ अर्थ है व्यंजन और स्वर का संयुक्त रूप। नागरी लिपि की यह विशेषता एक दृष्टि से तो अवगुण है, किंतु स्थान कम घेरने की दृष्टि से यह गुण भी है। उदाहरण के लिए कमल = kamala, निर्मल = Nirmala, केवल = kewala आदि। ऐसे अनेक शब्द हैं, जिसमें नागरी में रोमन की तुलना में कम स्थान घिरता है।

उच्चारण की दृष्टि से समान लिपि-चिंहों में आकृति की समानता

वैज्ञानिकता की दृष्टि से लिपि में यह गुण भी होना चाहिए। यों विश्व में कोई भी ऐसी लिपि नहीं है जो इस दृष्टि से पूर्ण हो। रोमन में तो यह गुण प्राय: नहीं के बराबर (P-B अपवाद है) है, उर्दू में कुछ है (बे-पे, काफ़-गाफ़, जीम-चे), किंतु नागरी में अपेक्षाकृत बहुत अधिक है : ट-ठ-ड-ढ-ङ-ढ़, प-फ-ब-म-भ, च-ज-भ, घ-ध, झ-भ, त-न आदि।

सुपाठ्यता

लिखते हैं पढ़ने के लिए। अत: सुपाठ्यता किसी भी लिपि के लिए अनिवार्यत: आवश्यक गुण है। इस दृष्टि से देवनागरी बहुत वैज्ञानिक लिपि है। रोमन की तरह उसमें mal को मल, माल, मैल या aghan को अघन, अगहन पढ़ने की परेशानी उठाने की संभावना ही नहीं है। उर्दू में भी 'अगहन' को 'अघन' या 'तर' को 'तर', 'तिर' 'तुर' या जूता को 'जोता' 'जूता' आदि कई रूपों में पढ़ने की गलती प्राय: हो जाती है, किंतु देवनागरी में यह अवैज्ञानिकत्ता नहीं है।[24]

5.2.4(ख) मानक हिंदी वर्तनी

(i) मानक हिंदी वर्णमाला तथा अंक

भारतीय संघ तथा कुछ राज्यों की राजभाषा स्वीकृत हो जाने के फलस्वरूप हिंदी का मानक रूप निर्धारित करना बहुत आवश्यक था, ताकि वर्णमाला में सर्वत्र एकरूपता रहे और टाइपराइटर आदि आधुनिक यंत्रों के उपयोग में लिपि की अनेकरूपता बाधक न हो।

इन सभी बातों को ध्यान में रखकर केंद्रीय हिंदी निदेशालय ने शीर्षस्थ विद्वानों आदि के साथ वर्षों के विचार-विमर्श के पश्चात् हिंदी वर्णमाला तथा अंकों का जो मानक स्वरूप निर्धारित किया, वह इस प्रकार है :

मानक हिंदी वर्णमाला

स्वर	अ	आ	इ	ई	उ	ऊ	ऋ	ए	ऐ	ओ	औ	
मात्राएँ		ा	ि	ी	ु	ू	ृ	े	ै	ो	ौ	

अनुस्वार	ं	(अं)
विसर्ग	:	(अ:)
अनुनासिकता चिंह	ँ	

व्यंजन	क	ख	ग	घ	ङ		
	च	छ	ज	झ	ञ		
	ट	ठ	ड	ढ	ण	ड़	ढ़
	त	थ	द	ध	न		
	प	फ	ब	भ	म		
	य	र	ल	व		ळ	
	श	ष	स	ह			
संयुक्त व्यंजन	क्ष	त्र	ज्ञ	श्र			
हल् चिंह	्	(ड्)					
ग्रहीत स्वन	ऑ	(ॉ),	ख़,	ज़,	फ़		
देवनागरी अंक	१	२	३	४	५		
	६	७	८	९	0		
भारतीय अंकों का अंतरराष्ट्रीय रूप	1	2	3	4	5		
	6	7	8	9	0		

संस्कृत के लिए प्रयुक्त देवनागरी वर्णमाला में तो ऋ, ऌ तथा ॡ भी हैं, किंतु हिंदी में इन वर्णों का प्रयोग न होने के कारण इन्हें हिंदी की मानक वर्णमाला में स्थान नहीं दिया गया है।

संविधान के अनुच्छेद 343(1) के अनुसार संघ के राजकीय प्रयोजनों के लिए प्रयुक्त होने वाले अंकों का रूप भारतीय अंकों का अंतरराष्ट्रीय रूप होगा, परंतु राष्ट्रपति संघ के किसी भी राजकीय प्रयोजन के लिए भारतीय अंकों के अंतरराष्ट्रीय रूप के साथ-साथ देवनागरी रूप का प्रयोग भी प्राधिकृत कर सकते हैं।

परिवर्धित देवनागरी वर्णमाला

केंद्रीय हिंदी निदेशालय ने उपर्युक्त मानक हिंदी वर्णमाला के साथ ही परिवर्धित देवनागरी वर्णमाला भी विकसित की है, ताकि उसके माध्यम से सभी भारतीय भाषाओं का लिप्यंतरण देवनागरी में हो सके।

परिवर्धित देवनागरी वर्णमाला

स्वर :	अ	आ	इ	ई	उ	ऊ	ऋ	ए	ऐ	ओ	औ
मात्राएँ :		ा	ि	ी	ु	ू	ृ	े	ै	ो	ौ

अनुस्वार : ं (अं)

विसर्ग : ः (अः)

अनुनासिकता चिंह : ँ

व्यंजन :	क	ख	ग	घ	ङ		
	च	छ	ज	झ	ञ		
	ट	ठ	ड	ढ	ण	ड़	ढ़
	त	थ	द	ध	न		
	प	फ	ब	भ	म		
	य	र	ल	व		ळ	
	श	ष	स	ह			
संयुक्त व्यंजन :	क्ष	त्र	ज्ञ	श्र			

हल् चिंह : ्

विशेषक चिंह :

स्वर :	(1)	ह्रस्व ए और ओ	ऎ ऒ
		मात्राएँ	ॆ ॊ
	(2)	कश्मीरी के विशिष्ट स्वर	उॆ ऊॆ अ॒ ओ॒ ऑ ऑ॒
		मात्राएँ	ॆ ॆ ा ा ॅ ॉ

व्यंजन :	(1)	कश्मीरी चवर्ग	च़ छ़
	(2)	सिंधी अंत:स्फोटी व्यंजन	ग॒ ज॒ ड॒ ब॒
	(3)	तमिल ழ और मलयालम ഴ	ऴ
	(4)	बंगला-असमिया	य़
	(5)	दक्षिण भारतीय भाषाओं के 'र' का कठोर उच्चारण	ऱ
	(6)	तमिल	ऩ
	(7)	मलयालम ഩ का वर्त्स्य उच्चारण	ऩ
	(8)	फ़ारसी-अरबी और अँग्रेजी से गृहीत स्वन	क़, ख़, ग़, ज़, झ़, फ़
	(9)	उर्दू अैन (ع) स्थिति के अनुसार	अ़ आ़ (आदत), अ़ि (अिबादत), ई़ (ईद), अ़ (अमर), अ़ै (अ़ैब), औ़ (औरत), आदि

(ii) हिंदी वर्तनी का मानकीकरण

किसी भी भाषा के सीखने-सिखाने में सहायक या बाधक बनने वाले दो प्रमुख तत्त्व हैं, उसका व्याकरण और लिपि। लिपि का एक पक्ष है, सामान्य और विशिष्ट स्वनों के पृथक प्रतीक-वर्णों की समृद्धि, उनका परस्पर स्पष्ट आकार-भेद, लिखावट में सरलता तथा स्थान-लाघव एवं प्रयत्न-लाघव।

लिपि का दूसरा पक्ष है, वर्तनी। एक ही स्वन को प्रकट करने के लिए विविध वर्णों का प्रयोग वर्तनी को जटिल बना देता है और यह लिपि का एक सामान्य दोष माना जाता है। यद्यपि देवनागरी लिपि में यह दोष न्यूनतम है, फिर भी उसकी कुछ अपनी विशिष्ट कठिनाइयाँ भी हैं।

इन सभी कठिनाइयों को दूर कर हिंदी वर्तनी में एकरूपता लाने के लिए भारत सरकार के शिक्षा मंत्रालय ने सन् 1961 में एक विशेषज्ञ समिति नियुक्त की थी। समिति ने अप्रैल, 1962 में अपनी अंतिम सिफारिशें प्रस्तुत कीं, जिन्हें सरकार ने स्वीकृत किया। इन्हें 1967 में हिंदी वर्तनी का मानकीकरण शीर्षक पुस्तिका में व्याख्या तथा उदाहरण सहित प्रकाशित किया गया था।

वर्तनी संबंधी अद्यतन नियम इस प्रकार हैं :

(1) संयुक्त वर्ण

(क) **खड़ी पाई वाले व्यंजन**

खड़ी पाई वाले व्यंजनों का संयुक्त रूप खड़ी पाई को हटाकर ही बनाया जाना चाहिए, यथा :

ख्याति, लग्न, विघ्न,	व्यास,
कच्चा, छज्जा,	श्लोक,
नगण्य,	राष्ट्रीय,
कुत्ता, पथ्य, ध्वनि, न्यास,	स्वीकृत,
प्यास, डिब्बा, सभ्य, रम्य,	यक्ष्मा,
शय्या,	त्र्यंबक,
उल्लेख	

(ख) **अन्य व्यंजन**

(अ) 'क' और 'फ' के संयुक्ताक्षर :
संयुक्त, पक्का, दफ़्तर, आदि की तरह बनाए जाएँ,
न कि संयुक्त, पक्का, दफ़्तर की तरह।

(आ) ङ, छ, ट, ठ, ड, ढ, द और ह के संयुक्ताक्षर हल् चिह्न लगाकर ही बनाए जाएँ, यथा :
वाङ्मय, लट्टू, बुड्ढा, विद्या, चिह्न, ब्रह्मा आदि
(वाङ्मय, लट्टू, बुड्ढा, विद्या, चिह्न, ब्रह्मा नहीं)।

(इ) संयुक्त 'र' के प्रचलित तीनों रूप यथावत् रहेंगे, यथा : प्रकार, धर्म, राष्ट्र।

(ई) 'श्र' का प्रचलित रूप ही मान्य होगा। इसे 'श' के रूप में नहीं लिखा जाएगा। त् + र् के संयुक्त रूप के लिए त्र और त्र दोनों रूपों में से किसी एक के प्रयोग की छूट होगी। किंतु 'क्र' को 'क्र' के रूप में नहीं लिखा जाएगा।

(उ) हल् चिह्न युक्त वर्ण से बनने वाले संयुक्ताक्षर के द्वितीय व्यंजन के साथ 'इ' की मात्रा का प्रयोग संबंधित व्यंजन के तत्काल पूर्व ही किया जाएगा, न कि पूरे युग्म से पूर्व, यथा : कुट्टिम, द्वितीय, बुद्धिमान, चिह्नित आदि। (कुट्टिम, द्वितीय, बुद्धिमान, चिह्नित नहीं)।

(ऊ) संस्कृत में संयुक्ताक्षर पुरानी शैली से भी लिखे जा सकेंगे, उदाहरणार्थ : संयुक्त, चिह्न, विद्या, चञ्चल, विद्वान, वृद्ध, अङ्ग, द्वितीय, बुद्धि आदि।

(2) विभक्ति-चिंह

(क) हिंदी के विभक्ति-चिंह सभी प्रकार के संज्ञा शब्दों में प्रातिपदिक से पृथक लिखे जाएँ, जैसे - राम ने, राम को, राम से आदि तथा स्त्री ने, स्त्री को, स्त्री से आदि। सर्वनाम शब्दों में ये चिह्न प्रातिपदिक के साथ मिला कर लिखे जाएँ, जैसे - उसने, उसको, उससे, उसपर आदि।

(ख) सर्वनामों के साथ यदि दो विभक्ति-चिंह हों तो उनमें से पहला मिलाकर और दूसरा पृथक लिखा जाए, जैसे - उसके लिए, इसमें से।

(ग) सर्वनाम और विभक्ति के बीच 'ही', 'तक' आदि का निपात हो तो विभक्ति को पृथक लिखा जाए, जैसे - आप ही के लिए, मुझ तक को।

(3) क्रियापद

संयुक्त क्रियाओं में सभी अंगभूत क्रियाएँ पृथक्-पृथक् लिखी जाएँ, जैसे - पढ़ा करता है, आ सकता है, जाया करता है, खाया करता है, जा सकता है, कर सकता है, किया करता था, पढ़ा करता था, खेला करेगा, घूमता रहेगा, बढ़ते चले जा रहे हैं, आदि।

(4) हाइफन

हाइफन का विधान स्पष्टता के लिए किया गया है।

(क) द्वंद्व समास में पदों के बीच हाइफन रखा जाए, जैसे - राम-लक्ष्मण, शिव-पार्वती-संवाद, देख-रेख, चाल-चलन, हँसी-मज़ाक, लेन-देन, पढ़ना-लिखना, खाना-पीना, खेलना-कूदना आदि।

(ख) सा, जैसा आदि से पूर्व हाइफन रखा जाए, जैसे - तुम-सा, राम-जैसा, चाकू-से तीखे।

(ग) तत्पुरुष समास में हाइफन का प्रयोग केवल वहीं किया जाए, जहाँ उसके बिना भ्रम होने की संभावना हो, अन्यथा नहीं, जैसे भू-तत्व। सामान्यतः तत्पुरुष समासों में हाइफन लगाने की आवश्यकता नहीं है, जैसे रामराज्य, राजकुमार, गंगाजल, ग्रामवासी, आत्महत्या आदि।

इसी तरह यदि 'अ-नख' (बिना नख का) समस्त पद में हाइफन न लगाया जाए तो उसे 'अनख' पढे जाने से 'क्रोध' का अर्थ भी निकल सकता है।

अ-नति (नम्रता का अभाव) : अनति (थोड़ा),

अ-परस (जिसे किसी ने न छुआ हो) : अपरस (एक चर्म रोग),

भू-तत्व (पृथ्वी तत्व) : भूतत्व (भूत होने का भाव),

आदि समस्त पदों की भी यही स्थिति है। ये सभी युग्म वर्तनी और अर्थ दोनों दृष्टियों से भिन्न-भिन्न शब्द हैं।

(घ) कठिन संधियों से बचने के लिए भी हाइफन का प्रयोग किया जा सकता है, जैसे : द्वि-अक्षर, द्वि-अर्थक आदि।

(5) अव्यय

'तक', 'साथ' आदि अव्यय सदा पृथक् लिखे जाएँ, जैसे - आपके साथ, यहाँ तक।

इस नियम को कुछ और उदाहरण देकर स्पष्ट करना आवश्यक है। हिंदी में आह, ओह, ऐ, ही, तो, सो, भी, न, जब, तब, कब, यहाँ, वहाँ, कहाँ, सदा, क्या, श्री, जी, तक, भर, मात्र, साथ, कि, किंतु, मगर, लेकिन, चाहे, या, अथवा, तथा, यथा और आदि अनेक प्रकार के भावों का बोध कराने वाले अव्यय हैं। कुछ अवयवों के आगे विभक्ति चिंह भी आते हैं, जैसे - अब से, तब से, यहाँ से, वहाँ से, सदा से आदि। नियम के अनुसार अव्यय सदा पृथक लिखे जाने चाहिए, जैसे - आप ही के लिए, मुझ तक को, आपके साथ, गज भर कपड़ा, देश भर, रात भर, दिन भर, वह इतना भर कर दे, मुझे जाने तो दो, काम भी नहीं बना, पचास रुपए मात्र आदि। सम्मानार्थक 'श्री' और 'जी' अव्यय भी पृथक लिखे जाएं, जैसे- श्री श्रीराम, कन्हैयालाल जी, महात्मा जी आदि।

समस्त पदों में प्रति, मात्र, यथा आदि अव्यय पृथक नहीं लिखे जाएँगे, जैसे- प्रतिदिन, प्रतिशत, मानवमात्र, निमित्तमात्र, यथासमय, यथोचित आदि। यह सर्वविदित नियम है कि समास होने पर समस्त पद एक माना जाता है। अतः उसे व्यस्त रूप में न लिखकर एक साथ लिखना ही संगत है।

(6) श्रुतिमूलक 'य', 'व'

(क) जहाँ श्रुतिमूलक 'य', 'व' का प्रयोग विकल्प से होता है, वहाँ न किया जाए, अर्थात किए-किये, नई-नयी, हुआ-हुवा आदि में से पहले (स्वरात्मक) रूपों का ही प्रयोग किया जाए। यह नियम क्रिया, विशेषण, अव्यय आदि सभी रूपों और स्थितियों में लागू माना जाए, जैसे - दिखाए गए, राम के लिए, पुस्तक लिए हुए, नई दिल्ली आदि।

(ख) जहाँ 'य' श्रुतिमूलक व्याकरणिक परिवर्तन न होकर शब्द का ही मूल तत्व हो वहाँ वैकल्पिक श्रुतिमूलक स्वरात्मक परिवर्तन की आवश्यकता नहीं है, जैसे - स्थायी, अव्ययी भाव, दायित्व आदि। यहाँ स्थाई, अव्यईभाव, दाइत्व नहीं लिखा जाएगा।

(7) **अनुस्वार तथा अनुनासिकता -चिंह (चंद्रबिंदु)**

अनुस्वार (ं) और अनुनासिकता चिंह (ँ) दोनों प्रचलित रहेंगे।

(क) संयुक्त व्यंजन के रूप में जहाँ पंचमाक्षर के बाद सवर्गीय शेष चार वर्णों में से कोई वर्ण हो तो एकरूपता और मुद्रण/लेखन की सुविधा के लिए अनुस्वार का ही प्रयोग करना चाहिए, जैसे - गंगा, चंचल, ठंडा, संध्या, संपादक आदि में पंचमाक्षर के बाद उसी वर्ग का वर्ण आगे आता है, अत: पंचमाक्षर के स्थान पर अनुस्वार का प्रयोग होगा। (गङ्गा, चञ्चल, ठण्डा, सन्ध्या, सम्पादक का नहीं)। यदि पंचमाक्षर के बाद किसी अन्य वर्ग का कोई वर्ण आए अथवा वही पंचमाक्षर दुबारा आए तो पंचमाक्षर अनुस्वार के रूप में परिवर्तित नहीं होगा, जैसे - वाङ्मय, अन्य, अन्न, सम्मेलन, सम्मति, चिन्मय, उन्मुख आदि। अत: वांमय, अंय, अंन, संमेलन, संमति, चिंमय, उंमुख आदि रूप ग्राह्य नहीं हैं।

(ख) चंद्रबिंदु के बिना प्राय: अर्थ में भ्रम की गुंजाइश रहती है, जैसे - हंस : हँस, अंगना : अँगना आदि में। अतएव ऐसे भ्रम को दूर करने के लिए चंद्रबिंदु का प्रयोग अवश्य किया जाना चाहिए। किंतु जहाँ (विशेषकर शिरोरेखा के ऊपर जुड़ने वाली मात्रा के साथ) चंद्रबिंदु के प्रयोग से छपाई आदि में बहुत कठिनाई हो और चंद्रबिंदु के स्थान पर बिंदु (अनुस्वार चिंह) का प्रयोग किसी प्रकार का भ्रम उत्पन्न न करे, वहाँ चंद्रबिंदु के स्थान पर बिंदु के प्रयोग की छूट दी जा सकती है, जैसे नहीं, में, मैं। कविता आदि के प्रसंग में छंद की दृष्टि से चंद्रबिंदु का यथास्थान अवश्य प्रयोग किया जाए। इसी प्रकार छोटे बच्चों की प्रवेशिकाओं में जहाँ चंद्रबिंदु का उच्चारण सिखाना अभीष्ट हो, वहाँ उसका यथास्थान सर्वत्र प्रयोग किया जाए, जैसे - कहाँ, हँसना, आँगन, सँवारना, मैं, में, नहीं, आदि।

(8) **विदेशी ध्वनियाँ**

(क) - अरबी-फारसी या अँग्रेजी मूलक वे शब्द जो हिंदी के अंग बन चुके हैं और जिनकी विदेशी ध्वनियों का हिंदी ध्वनियों में रूपांतर हो चुका है, हिंदी के रूप में ही स्वीकार किए जा सकते हैं, जैसे - कलम, किला, दाग आदि (क़लम, क़िला, दाग़ नहीं)।

- पर जहाँ उनका शुद्ध विदेशी रूप में प्रयोग अभीष्ट हो अथवा उच्चारणगत भेद बताना आवश्यक हो वहाँ उनके हिंदी में उनके हिंदी में प्रचलित रूपों में यथास्थान नुक्ते लगाए जाएँ, जैसे - खाना : ख़ाना, राज : राज़, फन : फ़न।
- सारांश रूप में यह कहा जा सकता है कि अरबी-फारसी एवं अंग्रेजी की मुख्यत: पाँच ध्वनियाँ (क़, ग़, ख़, ज़ और फ़) हिंदी में आई हैं जिनमें से दो (क़ और ग़) तो हिंदी उच्चारण (क, ग) में परिवर्तित हो गई हैं, एक (ख़) लगभग हिंदी 'ख' में खपने की प्रक्रिया में है और शेष दो (ज़, फ़) धीरे-धीरे अपना अस्तित्व खोने/बनाए रखने के लिए संघर्षरत हैं।

(ख) - अँग्रेजी के जिन शब्दों में अर्धविवृत्त 'ओ' ध्वनि का प्रयोग होता है, उनके शुद्ध रूप का हिंदी में प्रयोग अभीष्ट होने पर 'आ' की मात्रा (ा) के ऊपर अर्धचंद्र का प्रयोग किया जाए (ऑ, ॉ)।

- जहाँ तक अँग्रेजी और अन्य विदेशी भाषाओं से नए शब्द ग्रहण करने और उनके देवनागरी लिप्यंतरण का संबंध है, अगस्त - सितंबर, 1962 में वैज्ञानिक तथा तकनीकी शब्दावली आयोग द्वारा वैज्ञानिक शब्दावली पर आयोजित भाषाविदों की संगोष्ठी में अंतरराष्ट्रीय शब्दावली के देवनागरी लिप्यंतरण के संबंध में की गई सिफ़ारिश उल्लेखनीय है। उसमें यह कहा गया है कि

- अँग्रेजी शब्दों का देवनागरी लिप्यंतरण इतना क्लिष्ट नहीं होना चाहिए कि उसके लिए वर्तमान देवनागरी वर्णों में अनेक नए संकेत-चिंह लगाने पड़ें।
- अँग्रेजी शब्दों का देवनागरी लिप्यंतरण मानक अँग्रेजी उच्चारण के अधिक-से-अधिक निकट होना चाहिए। उसमें भारतीय शिक्षित समाज में प्रचलित उच्चारण संबंधी थोड़े-बहुत परिवर्तन किए जा सकते हैं।
- अन्य भाषाओं के शब्दों के संबंध में भी यही नियम लागू होना चाहिए।

(ग) हिंदी में कुछ शब्द ऐसे हैं, जिनके दो-दो रूप बराबर चल रहे हैं। विद्वत्समाज में दोनों रूपों की एक-सी मान्यता है। फ़िलहाल इनकी एकरूपता आवश्यक नहीं समझी गई है। कुछ उदाहरण हैं - गरदन/गर्दन, गरमी/गर्मी, बरफ़/बर्फ़, बिलकुल/बिल्कुल, सरदी/सर्दी, कुरसी/कुर्सी, भरती/भर्ती, फ़ुरसत/फ़ुर्सत, बरदाश्त/बर्दाश्त, वापिस/वापस, आखीर/आखिर, बरतन/बर्तन, दोबारा/दुबारा, दूकान/दुकान, बीमारी/बिमारी आदि।

(9) हल् चिंह

संस्कृतमूलक तत्सम शब्दों की वर्तनी में सामान्यत: संस्कृत रूप ही रखा जाए, परंतु जिन शब्दों के प्रयोग में हिंदी में हल् चिंह लुप्त हो चुका है, उनमें उसको फिर से लगाने का यत्न न किया जाए, जैसे - 'महान्', 'विद्वान' आदि के 'न' में।

(10) स्वन-परिवर्तन

संस्कृतमूलक तत्सम शब्दों की वर्तनी को ज्यों-का-त्यों ग्रहण किया जाए। अत: ब्रह्मा को ब्रम्हा, चिह्न को चिन्ह, उऋण को उरिण में बदलना उचित नहीं होगा। इसी प्रकार ग्रहीत, दृष्टव्य, प्रदर्शिनी, अत्याधिक, अनाधिकार आदि अशुद्ध प्रयोग ग्राह्य नहीं हैं। इनके स्थान पर क्रमश: गृहीत, द्रष्टव्य, प्रदर्शनी, अत्यधिक, अनधिकार ही लिखना चाहिए। जिन तत्सम शब्दों में तीन व्यंजनों के संयोग की स्थिति में एक द्वित्वमूलक व्यंजन लुप्त हो गया है उसे न लिखने की छूट है, जैसे अर्द्ध/अर्ध, उज्ज्वल/उज्वल, तत्त्व/तत्व आदि।

(11) विसर्ग

संस्कृत के जिन शब्दों में विसर्ग का प्रयोग होता है, वे यदि तत्सम रूप में प्रयुक्त हों तो विसर्ग का प्रयोग अवश्य किया जाए, जैसे - 'दु:खानुभूति' में। यदि उस शब्द के तद्भव रूप में विसर्ग का लोप हो चुका हो तो उस रूप में विसर्ग के बिना भी काम चल जाएगा, जैसे - दुख-सुख के साथी।

(12) 'ऐ' 'औ' का प्रयोग

हिंदी में ऐ (ै), औ (ौ) का प्रयोग दो प्रकार की ध्वनियों को व्यक्त करने के लिए होता है। पहले प्रकार की ध्वनियाँ 'है', 'और' आदि में हैं तथा दूसरे प्रकार की 'गवैया', 'कौवा' आदि में। इन दोनों ही प्रकार की ध्वनियों को व्यक्त करने के लिए इन्हीं चिह्नों (ऐ, ै, औ, ौ) का प्रयोग किया जाए। 'गवय्या', 'कव्वा' आदि संशोधनों की आवश्यकता नहीं है।

(13) **पूर्वकालिक प्रत्यय**

पूर्वकालिक प्रत्यय 'कर' क्रिया से मिलाकर लिखा जाए, जैसे - मिलाकर, खा-पीकर, रो-रोकर आदि।

(14) **अन्य नियम**

(क) शिरोरेखा का प्रयोग प्रचलित रहेगा।

(ख) फुलस्टॉप को छोडकर शेष विराम आदि चिंह वही ग्रहण कर लिए जाएं, जो अँग्रेजी में प्रचलित हैं, यथा - (- , ; ? !: =) (विसर्ग के चिंह को ही कोलन का चिंह मान लिया जाए)।

(ग) पूर्ण विराम के लिए खड़ी पाई (।) का प्रयोग किया जाए।[25]

5.2.4 (ग) कुंजीपटल तथा कुंजियों का भाषा-वैज्ञानिक विश्लेषण

कंप्यूटर के लिए देवनागरी लिपि को सर्वाधिक उपयुक्त पाया गया है। इसके बावजूद हिंदी में कंप्यूटर के कुंजीपटल तथा सॉफ्टवेयर तैयार करने का प्रयास बहुत कम किया गया है। फिर भी जितने प्रयास किए गए हैं, उनका विश्लेषण निम्नानुसार किया जा सकता है :

1) अँग्रेजी कुंजीपटल पर हिंदी की व्यवस्था

- अँग्रेजी कुंजीपटल को ही द्विभाषी अर्थात अँग्रेजी तथा हिंदी दोनों भाषाओं में कार्य करने के योग्य बनाया गया है। हिंदी के लिए पृथक से कुंजीपटल उपलब्ध नहीं है।

- हिंदी के कुंजीपटल द्विभाषी (हिंदी-अँग्रेजी) अथवा बहुभाषी (अँग्रेजी तथा भारतीय भाषाएँ) होने के कारण बहुपयोगी होते हैं।

- (क) चूँकि अँग्रेजी वर्णमाला में 26 अक्षर होते हैं, अतः अँग्रेजी कुंजीपटल में तदनुसार ही व्यवस्था की जाती है। दूसरी ओर, हिंदी वर्णमाला में 47 अक्षर होते हैं। इन्हें अँग्रेजी के 26 अक्षरों वाले कुंजीपटल में ही समेकित करना पडता है।

  (ख) कुंजीपटल पर हिंदी वर्णमाला के अक्षरों के साथ-साथ मात्राओं की व्यवस्था भी उसी सीमित कुंजियों वाले कुंजीपटल पर करनी पड़ती है।

  राम बंसल "विज्ञाचार्य" के अनुसार, "वर्तमान कंप्यूटर रचना अँग्रेजी वर्णमाला एवं शब्द रचना नियमों को ध्यान में रखकर की गई है, जो हिंदी के प्रचलित लगभग 200 ध्वनि प्रतीकों को समाहित करने में अक्षम है।"

- एक ही कुंजीपटल में भिन्न प्रकृति वाली दो भाषाओं की व्यवस्था होने के कारण, जब उपयोक्ता एक भाषा में काम करने के बाद दूसरी भाषा में काम करना शुरु करता है, तब उसे कुंजीपटल के साथ व्यवस्थित होने में समय लगता है।

2) विशेष क्रिया संबंधी कुंजियाँ

कुंजीपटल पर उपलब्ध अन्य कुंजियों जैसे फंक्शन कुंजी, मेक्रो कुंजी आदि पर अंकन अँग्रेजी में किया जाता है। अतः हिंदी-उपयोक्ता को कंप्यूटर में विशेष क्रिया आदि के लिए अँग्रेजी पर निर्भर होना पडता है।

3) विविध कुंजीपटल लेआउट

(क) टाइपराइटर पर काम करने वाले की सुविधा के लिए **रेमिंग्टन टाइपराइटर** की कुंजियों की सुविधा हिंदी सॉफ्टवेयर में उपलब्ध कराई जाती है, लेकिन यह सुविधा सभी कंप्यूटरों पर नहीं है। हालांकि प्रतिस्पर्धा के कारण आजकल हिंदी सॉफ्टवेयर बनाने वाली कंपनियां अपने उत्पाद के नए संस्करणों में यह सुविधा देने लग गई हैं। लेकिन उपयोक्ता एक बार हिंदी सॉफ्टवेयर लोड करा लेने के बाद सामान्यतः उसका उन्नयन (अपग्रेडेशन) नहीं कराता है।

(ख) हिंदी टाइपिंग न जानने वाले उपयोक्ता की सुविधा के लिए **ध्वन्यात्मक (फोनेटिक) कुंजीपटल** की सुविधा भी उपलब्ध है। इसके अंतर्गत व्यक्ति हिंदी के शब्दों को अँग्रेजी में टाइप करता है और कंप्यूटर उसे हिंदी में उपलब्ध कराता है। लेकिन कई बार हिंदी के शब्दो को अँग्रेजी में ज्यों का त्यो लिखना संभव नहीं हो पाता है क्योंकि अँग्रेजी की वर्णमाला में अक्षर हिंदी की तुलना में आधे हैं। अतः हिंदी के शेष अक्षरों को अँग्रेजी के सीमित अक्षरों में समेकित करना पडता है, जिससे कंप्यूटर हिंदी के मूल शब्द को उसकी सही वर्तनी के साथ प्रस्तुत नहीं कर पाता है। उदाहरण के लिए **त** और **ट** के लिए अँग्रेजी के **टी** अक्षर को टाइप करना पडता है, जो कंप्यूटर को भ्रमित करने के लिए पर्याप्त होता है।

(ग) हिंदी सॉफ्टवेयर के विकास हेतु हिंदी के स्वरूप के मानकीकरण के बाद, **इनस्क्रिप्ट कुंजीपटल** उपलब्ध हो गए हैं, जो भारतीय भाषाओं के अनुकूल है।

(घ) हिंदी सॉफ्टवेयर के नए संस्करणों में अब **फ्लोटिंग कुंजीपटल** की सुविधा शामिल कर ली गई है। इसके अंतर्गत स्क्रीन पर कुंजीपटल का ले-आउट प्रदर्शित होता है। इसे देख कर हिंदी टाइपिंग न जानने वाला उपयोक्ता हिंदी टंकण कर सकता है। कुछ फ्लोटिंग कुंजीपटलों में **माउस** के माध्यम से टाइपिंग करने की भी सुविधा होती है। जिस क्रम में अक्षर टाइप करने होते हैं, उसी क्रम में माउस से ले-आउट में प्रदर्शित अक्षरों को क्लिक करना होता है। क्लिक करने पर वांछित अक्षर टाइप हो जाता है।

(ङ) यदि उपयोक्ता उपलब्ध कुंजीपटलों के स्थान पर अपना कुंजीपटल तैयार करना चाहता है जिसमें वह अपनी इच्छानुसार कुंजियों को व्यवस्थित कर सके, तो एसे उपयोक्ता की सहायता के लिए **यूजर्स डिफाइंड** की सुविधा उपलब्ध है।

4) हिंदी अक्षरों के स्टीकर

अनेक कुंजीपटलों में हिंदी के अक्षरों को अंकित नहीं किया जाता है तथा अँग्रेजी कुंजीपटल की कुंजियों पर हिंदी अक्षरों के स्टीकर चिपकाने पड़ते हैं। ऐसे कुंजीपटल का अधिक उपयोग करने पर ये स्टीकर निकल भी जाते हैं। स्टीकरों के निकल जाने पर उस कुंजी-विशेष के मान को समझना मुश्किल हो जाता है कि वह कुंजी किस अक्षर के लिए

निर्धारित की गई है। स्टीकर का प्रयोग लंबे समय तक करने के लिए की-बोर्ड पर पारदर्शी स्किन (कवर) लगाया जा सकता है।

5) कुंजीपटल पर वर्णमाला की प्रस्तुति

(क) हिंदी कुंजीपटल पर कुंजियों को इस प्रकार व्यवस्थित किया गया है कि बाँई ओर **स्वर** रहे और दाँई ओर **व्यंजन**।

(ख) चूँकि अँग्रेजी कुंजीपटल पर ही हिंदी के अक्षरों को व्यवस्थित करने का प्रयास किया गया है, इसलिए स्वर के सभी अक्षर तथा मात्राएँ बाँई ओर उपलब्ध नहीं कराई जा सकी हैं। ऑ, ऋ तथा इनकी मात्राओं को कुंजीपटल पर दाँई ओर ऊपर की तरफ और ऍ तथा इसकी मात्रा को बाँई ओर ऊपर की तरफ स्थान दिया गया है।

(ग) इसी प्रकार संयुक्त अक्षर ज्ञ, त्र, श्र तथा क्ष को ऊपर की तरफ मध्य में अंकों तथा चिंहों के साथ स्थान दिया गया है।

6) मात्रा

(क) कंप्यूटर दो मात्राओं को एक साथ कभी भी स्वीकार नहीं करता है। एक व्यंजन पर दो स्वर- चिंह नहीं लगाए जा सकते।

(ख) यदि किसी शब्द की मात्रा मे परिवर्तन करना हो, तो मात्रा के साथ-साथ अक्षर भी हटाना पडता है।

(ग) टाइप करते समय अक्षर पहले टाइप किया जाता है और फिर उसकी मात्रा लगाई जाती है। लेकिन जब किसी शब्द को 'बेक स्पेस' कुंजी की सहायता से मिटाया जाता है, तो अक्षर पहले मिटता है और उसकी मात्रा बाद में, जबकि बेकस्पेस करते समय अंतिम टंकित सामग्री पहले मिटनी चाहिए। वैसे इस समस्या का समाधान 'अनडू' के माध्यम से संभव है।

7) अक्षर

(क) **ड़, ढ़** - हिंदी में शब्दांत में आने वाले **ड़** तथा **ढ़** का उच्चारण करते समय ध्यान देना आवश्यक होता है क्योंकि हिंदी ध्वन्यात्मक भाषा है। जैसे - पड़, पढ़, पेड़, बाढ़। ISM जैसे हिंदी सॉफ्टवेयर में ड़ / ढ़ को टाइप करते समय एक के स्थान पर दो कुंजियाँ दबानी पड़ती हैं। इन दोनों कुंजियों के मान के स्थान निर्धारण कम होने की वजह से नीचे लगाई जाने वाली बिंदी अक्षर ड / ढ के अंतिम गोलाई में मिल जाती है और बिंदी लगे होने के बावजूद ड़ / ढ़ के स्थान पर ड / ढ का आभास होता है क्योंकि कंप्यूटर में तैयार होने वाले सभी अक्षर बिंदुओं का समायोजित स्वरूप होते हैं और उसमें एक बिंदु और मिल जाने पर उसकी पृथक सत्ता पता नहीं चल पाती है।

(ख) संयुक्त अक्षर टाइप करने के लिए पृथक कुंजी की व्यवस्था है, लेकिन यदि संयुक्त अक्षरों को टंकित करने के बाद मिटाना हो, तो बेकस्पेस-कुंजी को एक के स्थान पर दो या दो से अधिक बार दबाना पडता है, जिसमें समय अधिक लगता है। केवल एक बटन दबाने पर पूरा अक्षर नहीं मिटता है बल्कि संयुक्त अक्षर में निहित अक्षरों को एक-एक करके मिटाना पडता है। जैसे - क्ष > क> क् > । इस प्रकार **क्ष** के लिए तीन बार डिलीट करना पडता है।

वस्तुत: संयुक्त अक्षरों को एक ही स्ट्रोक में टाइप करने की व्यवस्था पृथक कुंजी के रूप में की गई है तथा इसमें निहित सभी स्वर तथा व्यंजन को संयुक्त रूप से एक साथ टाइप कर दिया जाता है। लेकिन इन्हें मिटाते समय इसमें निहित सभी स्वर तथा व्यंजन को पृथक-पृथक मानते हुए एक-एक कर के मिटाया जाता है।

(ग) सामान्य अक्षरों को मिटाने के लिए भी यही व्यवस्था है। पहले स्वर मिटता है और फिर व्यंजन। इसका फायदा संपादन के समय मिलता है और स्वर रहित अक्षरों को दुबारा से टाइप नहीं करना पडता है।

8) हलंत

(क) स्वर रहित अक्षर को कंप्यूटर हलंत लगे होने पर ही स्वीकार करता है। अत: स्वर रहित अक्षर के बाद कोई अक्षर नहीं होने पर कंप्यूटर उसे उसके आधे स्वरूप में प्रस्तुत करने के स्थान पर हलंत लगा कर प्रस्तुत करता है। उदाहरण के लिए **वाक्** तथा **वाक्य** शब्दों को लिया जा सकता है।

(ख) हलंत युक्त अक्षर पूर्ण इकाई नहीं होती हैं। अत: कंप्यूटर ऐसे अक्षरों पर मात्रा नहीं लगाएगा, जबकि व्यक्ति हाथ से लिखते समय लगा सकता है। उदाहरण के लिए **वाक्ी** को लिया जा सकता है।

9) मानकीकरण

व्याकरण और भाषाविज्ञान के अन्योन्य पारस्परिक संबंध के कारण हिंदी कुंजीपटल तथा हिंदी वर्णमाला दोनों का मानकीकरण किया जाना अत्यावश्यक है। इस मानकीकरण के अभाव में इन दोनों ही क्षेत्रों में अराजकता व्याप्त हो जाती हैं। इससे इसके उपयोक्ता को इसका प्रयोग करने में अनेक असुविधाओं का सामना करना पडता है, जिससे हिंदी के प्रति विकर्षण उत्पन्न होता है तथा भ्रम की स्थिति निर्मित होती है। उदाहरण के लिए **जावे/जाए/जाये/जाय** में से किसे मानक तथा सही माना जाए, यह निर्णय करना कठिन है।

(क) **हिंदी कुंजीपटलों का मानकीकरण** : विभिन्न कंपनियों द्वारा विभिन्न प्रकार के कुंजीपटल बनाए जाने के कारण कुंजीपटलों में एकरूपता बनाए रखने की आवश्यकता उत्पन्न हुई। इस आवश्यकता की पूर्ति के लिए **इस्फोक, इनस्क्रिप्ट** जैसे मानक फॉंट कूट निर्धारित किए गए हैं। इस्फोक मानक कूट के दो नाम प्रचलन में प्राप्त होते हैं : (i) Indian Standard for Font Coding तथा (ii) Intelligence based Script Font Code (सी-डेक के ISM-2000 यूजर्स मैनुअल से संदर्भित)। इस प्रकार मानक कोड का नाम मानक नहीं रह गया है।

(ख) **हिंदी वर्णमाला का मानकीकरण** :

- विश्व की अवैज्ञानिक भाषाओं ने अपना स्वरूप निर्धारित कर लिया है लेकिन हिंदी जैसी वैज्ञानिक भाषा ने अभी तक ऐसा नहीं किया है।

- **म्ह, न्ह, ल्ह** - ये तीन अदृश्य महाप्राण ध्वनियाँ हैं, जो संस्कृत में नहीं थीं लेकिन हिंदी में विकसित हुई है। अत: हिंदी में चिंह बनाए गए। इसे बाँटा नहीं जा सकता क्योंकि ये एक ध्वनि हैं। जैसे - तुम्हारा, कान्हा, कुल्हाड़ी, आल्हा, चूल्हा। इन ध्वनियों को अभी तक हिंदी वर्णमाला में स्थान प्राप्त नहीं हुआ है।

- ष - अकसर यह तर्क दिया जाता है कि हिंदी भाषा में मूर्धन्य **ष** प्रचलन में नहीं रह गया है तथा यह तालव्य **श** में समाहित हो गया है। अत: इसे हिंदी वर्णमाला में स्थान देने की आवश्यकता नहीं है। लेकिन यह तर्क विज्ञान-सम्मत नहीं है क्योंकि भाषा बिना मतलब के बोझ नहीं ढोती है। इष्ट, ईषत्, अनिष्ट, विशेषण, उष्ण, वार्ष्णेय जैसे शब्दों को उदाहरण-स्वरूप लिया जा सकता है। इन उदाहरणों में मूर्धन्य **ष** के बाद के अक्षर का उच्चारण मूर्धन्य **ष** का उच्चारण करने के बाद ही, उस शब्द-विशेष के अंग के रूप में किया जा सकता है क्योंकि उच्चारण अपने आसपास के वातावरण से प्रभावित होता है। अत: मूर्धन्य **ष** का उच्चारण तालव्य **श** के रूप में नहीं किया जा सकता। भाषा में शार्ट-कट नहीं होते।

- हिंदी के कई अक्षरों के एक से अधिक रूप प्रचलित हैं। जैसे - प्र से मिलता-जुलता पुराना अ (ऋ) तथा नया अ, भ से मिलता-जुलता पुराना झ (ꣿ) तथा नया झ, ळ तथा ल, अनेक प्रकार के छ तथा श्र के विविध रूप।

हिंदी वर्णमाला की इन भिन्नताओं तथा अनेकरूपता को ध्यान में रखते हुए **केंद्रीय हिंदी निदेशालय** द्वारा **मानकीकरण के लिए सिफारिश** की गई है,जिसमें से प्रमुख हैं :

- **अनुस्वार** - भिन्न पंचम वर्ण के स्थान पर अनुस्वार लिखें। जैसे - सम्बंध, संबन्ध, सम्बन्ध के स्थान पर **संबंध** लिखा जाए। इसी प्रकार, सम्यंत्र, संयन्त्र, सम्यन्त्र के स्थान पर **संयंत्र** लिखा जाए।

- **हलंत वाले अक्षर** - परंपरागत तरीके से हिंदी में हलंत वाले अक्षरों को उनके बाद में आने वाले अक्षर के साथ जोड कर लिखा जाता है। इस प्रकार लिखने पर ये अक्षर नए संयुक्त अक्षर का रूप ले लेते हैं, जिससे अहिंदी-भाषियों तथा नौसिखियों को असुविधा का सामना करना पड़ता है। जैसे - द्वंद्व > द्‌वंद्‌व, द्विवेदी > द्‌विवेदी, बुद्धि > बुद्‌धि

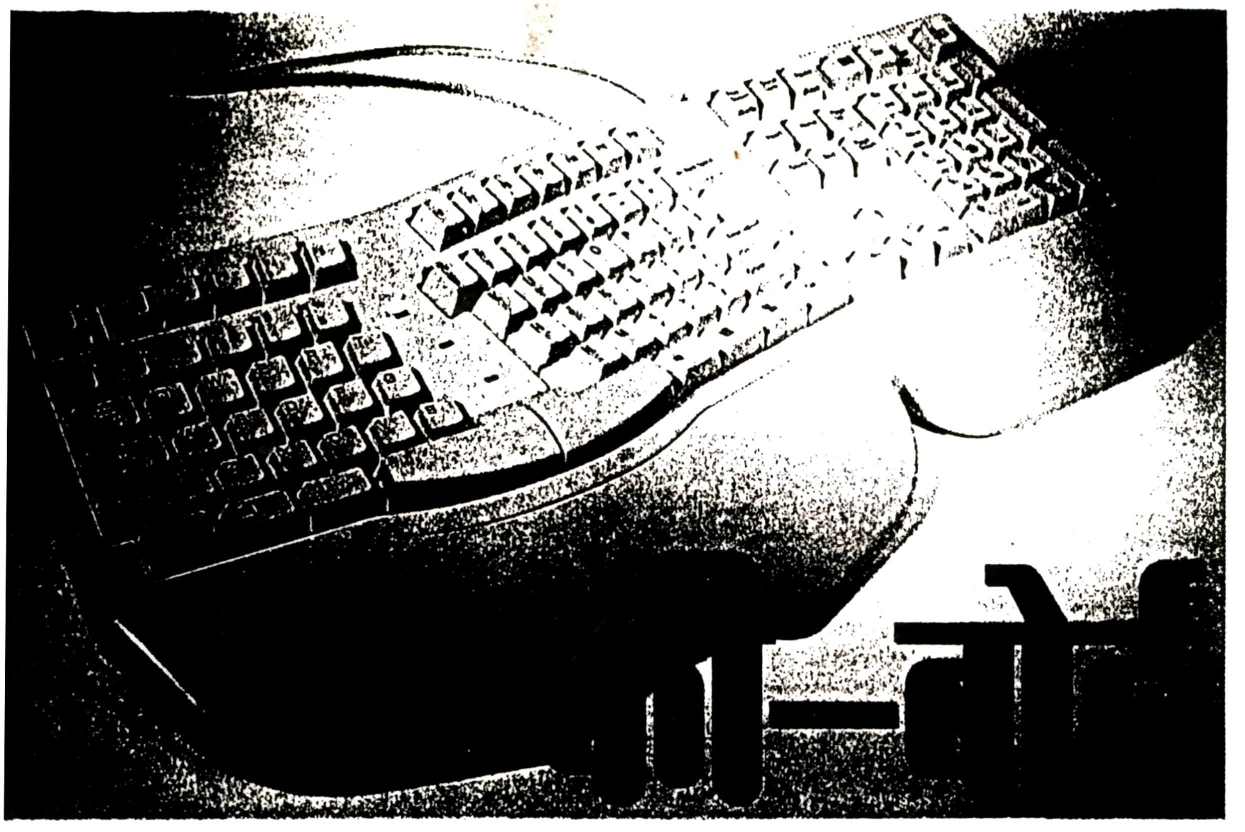
- **र अक्षर** - लखनऊ समिति के अनुसार **स्वर रहित र** को हलंत के साथ लिखा जाना चाहिए। लेकिन स्वर रहित **र** के सभी रूप पूर्ववत रहेंगे। जैसे - रु, रू, श्र, श्रृ, क्र, र्क, ट्र।

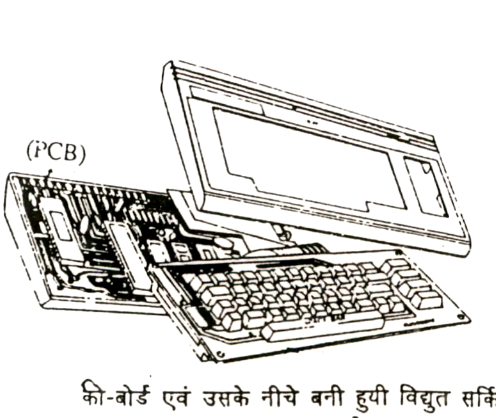
राम बंसल "विज्ञाचार्य" के अनुसार, "**हिंदी को प्रचलित कंप्यूटर कुंजीपटल में ठूँसने से इसकी उक्त वर्तनी विशेषता का नष्ट होना अपेक्षित है।** विश्व के अन्य भाषाविद् अपनी भाषा को इस उद्देश्य से विकृत करने के पक्षधर नहीं हैं, वे कुंजीपटल संरचना को अपनी भाषा के अनुकूल बनाने के प्रयास कर रहे हैं। हमारे यहाँ ऐसा कोई प्रयास नहीं किया जा रहा।

हिंदी संस्कृत से अधिक प्राचीन लोकभाषा के रूप में विकसित हुई है, कंप्यूटरी सुविधा के लिए इसका अविकास खेदजनक होगा। **इसकी सुविधा के लिए कंप्यूटर को विकसित करना हमारा लक्ष्य होना चाहिए।** यह संभव तो है ही वांछित भी है। **कंप्यूटर अपने वर्तमान स्वरूप में हिंदी को विकसित अथवा समृद्ध करने में अक्षम है।** हिंदी कंप्यूटर के विकास से हिंदी में वर्तनी दोष पनपे हैं।

हिंदी तथा कंप्यूटर का कोई पारस्परिक संबंध नहीं है क्योंकि दोनों का विकास पूर्णतः पृथक दृष्टिकोणों से किया गया है। **हिंदी के माध्यम से सर्वांगीण कंप्यूटरी विकास के लिए अनिवार्य रूप से कुंजीपटल की पुनर्संरचना हो तथा हिंदी में कंप्यूटरी भाषा की रचना हो।"**

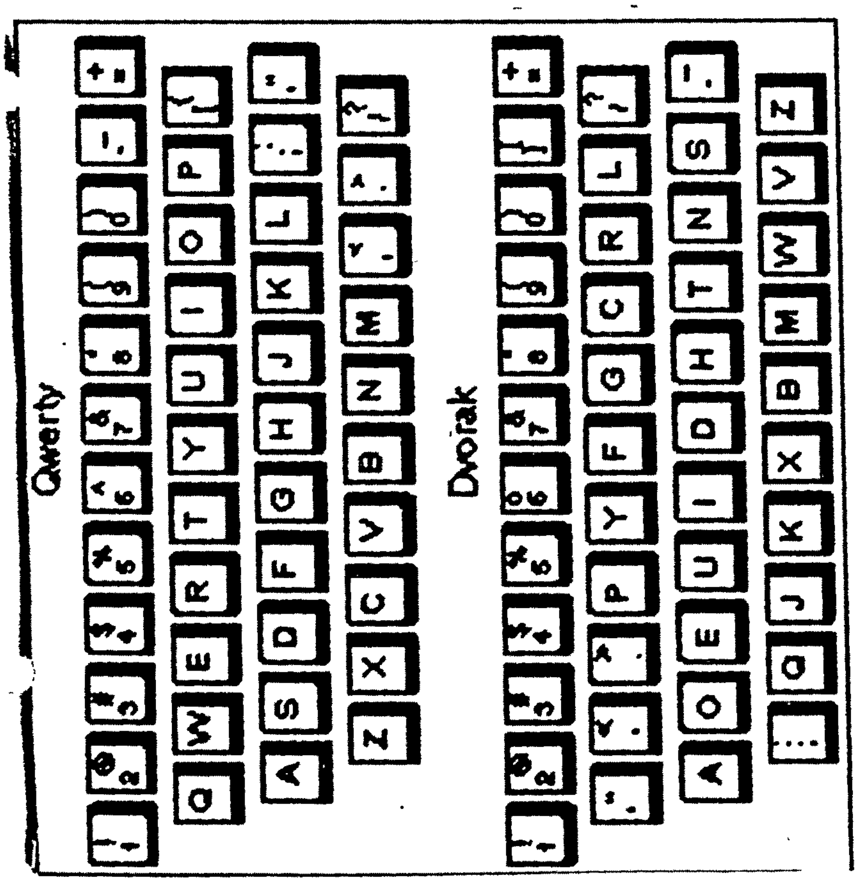
lingstdy\Keyboard.doc



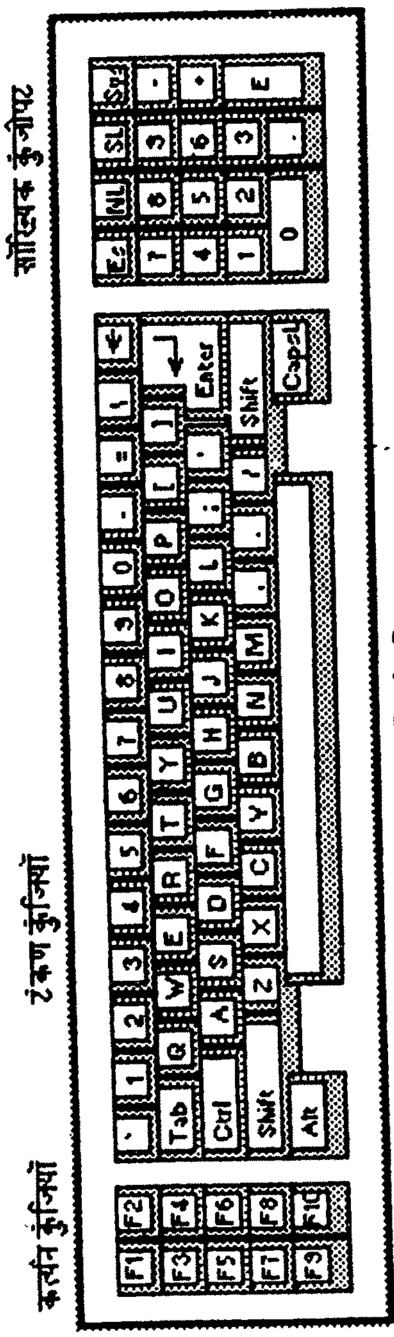


की-बोर्ड एवं उसके नीचे बनी हुयी विद्युत सर्किट

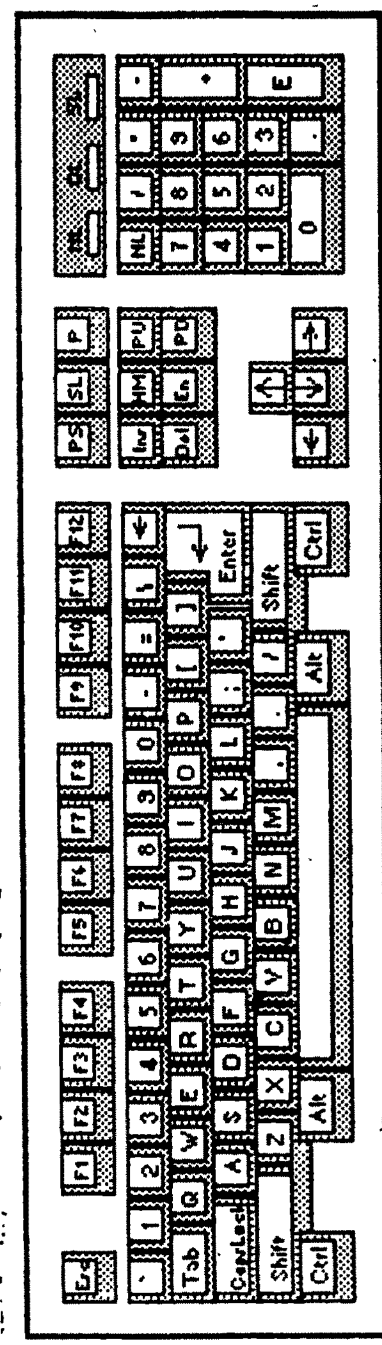
प्रदर्श - 1



प्रदर्श - 2

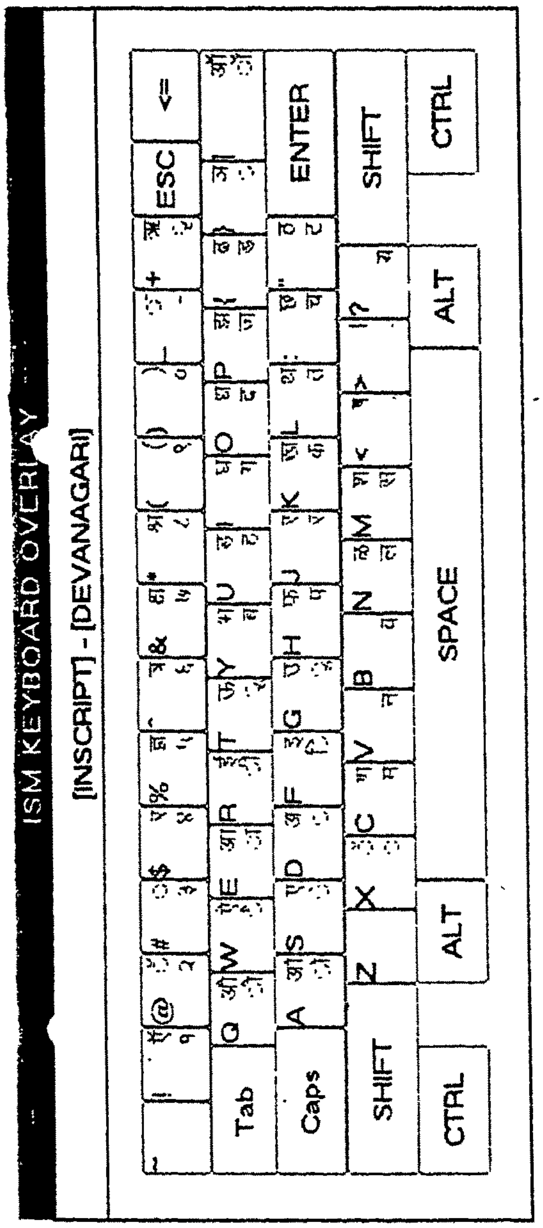


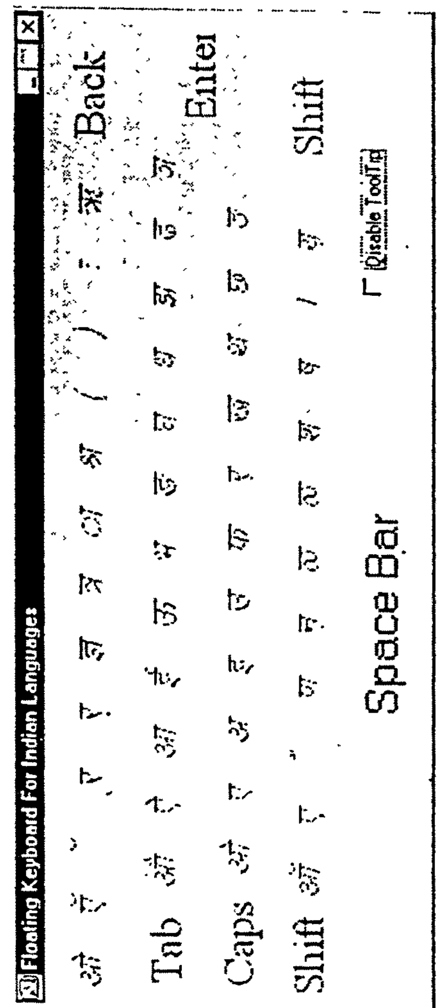
84-कुंजी कुंजीपटल



संवर्द्धित 101-कुंजी कुंजीपटल

प्रदर्श - 3

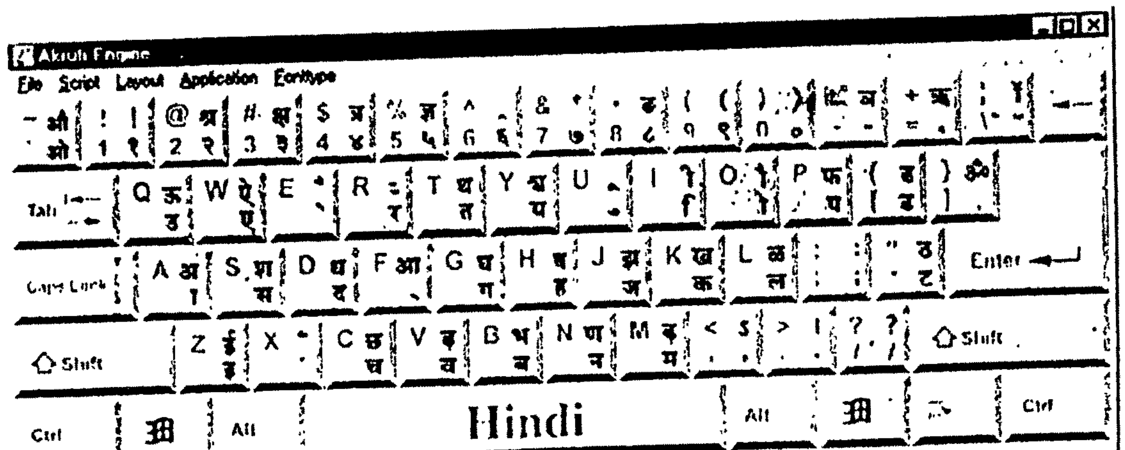

ISM KEYBOARD OVERLAY
[INSCRIPT] - [DEVANAGARI]
ESC
Tab
Caps
ENTER
SHIFT
SHIFT
CTRL
ALT
SPACE
ALT
CTRL


Floating Keyboard For Indian Languages
Back
Tab
Enter
Caps
Shift
Shift
Space Bar
Disable ToolTip

प्रदर्श - 4

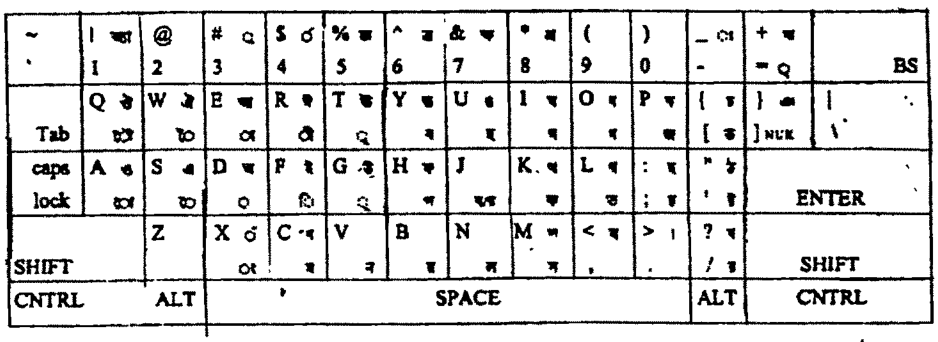
Roman Indian - **Shusha** version 1.5

Key Board Layout - Hindi Shusha

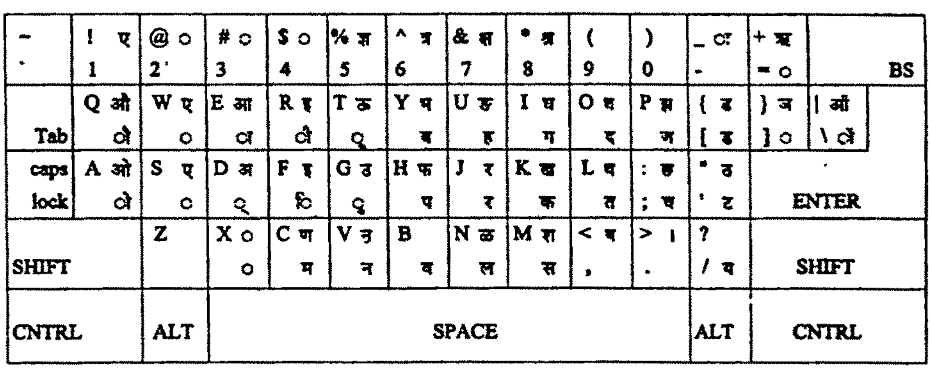


Keyboard Layout - Akruti English Phonetic *Halant = f Nukta =]*

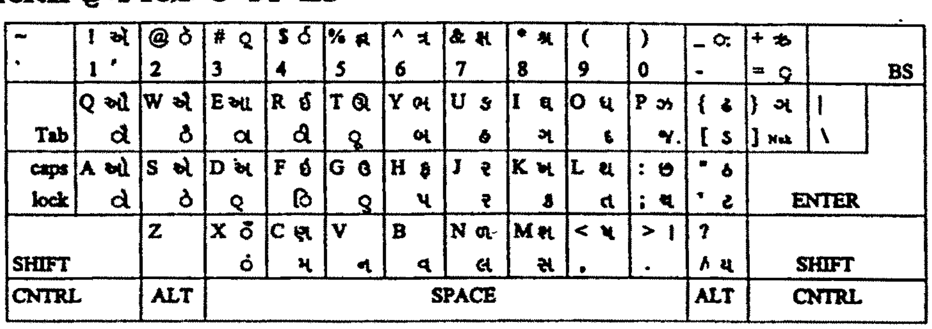
असमिया और बंगाली इन-स्क्रिप्ट की-बोर्ड



देवनागरी इन-स्क्रिप्ट की-बोर्ड



गुजराती इन-स्क्रिप्ट की-बोर्ड



5.3 मध्यवर्ती (इंटरफेस) भाषा

5.3.1 निम्न स्तरीय भाषाएँ
- (क) मशीनी भाषा
- (ख) असेंबली भाषा

5.3.2 उच्च स्तरीय भाषाएँ
- (सभी प्रोग्रामिंग भाषाएँ)

5.3.3 मध्यवर्ती भाषा का भाषावैज्ञानिक अध्ययन
- (i) प्राकृतिक और अप्राकृतिक भाषाओं में अंतर
- (ii) मध्यवर्ती (इंटरफेस) भाषाओं का भाषावैज्ञानिक अध्ययन

5.3 मध्यवर्ती (इंटरफेस) भाषा

कंप्यूटर से कोई काम कराने अथवा किसी समस्या को हल कराने के लिए प्रोग्राम लिखना पड़ता है। कोई प्रोग्राम (Program) कुछ निर्देशों या कथनों (Statements) का ऐसा क्रमबद्ध (Ordered) समूह है, जिसका ठीक-ठीक पालन करने पर कोई काम पूरा हो जाता है या कोई समस्या हल हो जाती है। किसी अल्गोरिथ्म (Algorithm) को कंप्यूटर संबंधी किसी भाषा में लिखने से प्रोग्राम बन जाता है। कंप्यूटर जो बड़े-बड़े काम करता मालूम पड़ता है, वह कंप्यूटर की नहीं बल्कि उसको दिए गए प्रोग्राम की करामात है और प्रोग्राम लिखने वाले अर्थात प्रोग्रामर (Programmer) की बुद्धि का कमाल है। कंप्यूटर केवल दिए प्रोग्राम का ठीक-ठीक पालन कर के काम जल्दी कर देता है।[26]

कंप्यूटर केवल 0 एवं 1 को ही समझता है जो कि किसी स्विच (Switch) के ऑन व ऑफ होने पर निर्धारित होते हैं। अतः हमें कंप्यूटर को निर्देश द्वि-आधारी अंकों 0 व 1 में ही देना होता है। परंतु किन्हीं निदेशों को 0 व 1 के समूह में बनाना या उसमें परिवर्तित करना हमारे लिए अत्यंत ही जटिल कार्य है। इसके लिए एक दुभाषिए या इंटरप्रेटर (Interpreter) की आवश्यकता होती है। यह इंटरप्रेटर एक इलैक्ट्रिकल सर्किट (Electrical circuit) के रूप में होता है।[27]

कंप्यूटर के लिए प्रोग्राम ऐसी भाषा में लिखा जाता है, जिसको कंप्यूटर समझ सके। जिन भाषाओं में प्रोग्राम लिखे जाते हैं, उन्हें प्रोग्रामिंग भाषाएँ (Programming Languages) कहा जाता है। प्रोग्रामिंग भाषाओं को दो श्रेणियों में बाँटा जाता है :

I) निम्नस्तरीय भाषाएँ (Low Level Languages)

ii) उच्चस्तरीय भाषाएँ (High Level Languages)[28]

वीरेंद्र जैन ने इन्हें निम्न स्तरीय भाषा (मशीन भाषा या मशीन असेंबली कोड), मध्य स्तरीय भाषा (सांकेतिक भाषा या असेंबली कोड), तथा उच्च स्तरीय भाषा (फोरट्रान, बेसिक, आदि) में विभक्त किया है।[29]

5.3.1 निम्नस्तरीय भाषाएँ (Low Level Languages)

निम्न स्तरीय प्रोग्रामिंग भाषाएँ उनको कहा जाता है, जो किसी कंप्यूटर पर पूरी तरह निर्भर करती हैं और हर कंप्यूटर के लिए अलग-अलग होती हैं। निम्न स्तरीय भाषाएँ मुख्यतः दो होती हैं :

(क) मशीनी भाषा (Machine Language)

(ख) असेंबली भाषा (Assembly Language)[30]

(क) मशीनी भाषा (Machine Language/Code) :

मशीन भाषा में प्रत्येक निर्देश के दो भाग होते हैं। एक आदेश कोड व दूसरा स्थिति कोड। आदेश कोड व स्थिति कोड 0 व 1 के क्रमों में ही व्यक्त किए जाते हैं।[31]

किसी कंप्यूटर में सारा कार्य बाइनरी अंकों 0 और 1 के माध्यम से किया जाता है। उसकी भाषा भी 0 और 1 से बनी होती है। इस भाषा को कंप्यूटर की **मशीनी भाषा** कहा जाता है। कंप्यूटर केवल अपनी मशीनी भाषा को ही समझ सकता है, इसलिए उसको दिए जाने वाले सभी आदेश (Command) और प्रोग्राम (Program) मशीनी भाषा में ही होने चाहिए।[32]

प्रारंभ में कंप्यूटर प्रोग्रामर मशीन भाषा में ही प्रोग्राम लिखते हैं, जिन्हें कंप्यूटर सीधे ही समझ लेता था। परंतु 0 व 1 के समूह में प्रोग्राम लिखना एक जटिल कार्य था।[33]

"विज्ञाचार्य" के अनुसार निम्नस्तरीय भाषाएँ (Low Level Languages) जो कंप्यूटरों के अवयवों की वास्तविक प्रक्रिया पर आधारित होती हैं। ये शक्तिशाली भाषाएँ होती हैं, किंतु इन्हें सीखने से पूर्व कंप्यूटर के अवयवों की कार्यविधि का ज्ञान आवश्यक है; इसलिए इनका सीखना थोड़ा कठिन कार्य है।[34]

कंप्यूटर मानव का सबसे विश्वसनीय नौकर है जिसे अपने कार्य संपादन के लिए अपने स्वामी से केवल आदेश चाहिए। लेकिन मशीन (यंत्र) होने के कारण उसकी अपनी पद्धति है जिसे मशीनी भाषा से ही संचालित कर सकते हैं।

कंप्यूटर विद्युत स्पंदनों की भाषा समझता है। वह स्पंदन के होने या न होने की स्थिति को भाँप लेता है, जिसे द्विआधारी कोड कहते हैं। इस द्विआधारी कोड को प्रतीकात्मक रूप से 0 और 1 से लिखा जाता है, जिसका मशीन के लिए अर्थ है, विद्युत स्पंदन (Pulse) का होना या न होना। इसे मशीनी भाषा कहते हैं। प्रत्येक कंप्यूटर के डिजाइन के अनुसार मशीन कोड अलग-अलग होता है। वास्तव में प्रत्येक चिप्पड के लिए एक मशीन कोड होता है जो उसके निर्माता द्वारा तय किया जाता है। मशीन कोड के अनुसार प्रत्येक आदेश या अनुदेश को 0 एवं 1 में बनी संख्या के रूप में लिखा जाता है।

कंप्यूटर का केंद्रीय संसाधन एकक इन द्विआधारी संकेतों को विद्युतीय स्पंदनों के माध्यम से ग्रहण करता है। ये स्पंदन कंप्यूटर के अंदर लगी हुई अंकीय घड़ी से जारी होते हैं एवं इन्हें इलैक्ट्रॉनिक द्वारों (Gates) से गुजरना होता है। एक सेकेंड में, औसतन कंप्यूटर मे 8 मेगा हट्र्ज की गति से स्पंदन जारी होते हैं यानि 80 लाख स्पंदन प्रति सेकेंड। स्पंदनों को द्विआधारी कोड के अनुसार जारी किया जाता है। स्पंदन होने का अर्थ है 1 एवं स्पंदन नहीं होने का अर्थ है 0।

मशीनी भाषा में अनुदेशों के दो स्पष्ट भाग होते हैं :

- संक्रिया-कूट (ऑपरेशन कोड) या फलन कूट (फंक्शन कूट), तथा
- संकार्य-पता (ओपरेंड- Operand एड्रेस)।

ऑपरेशन कोड को संक्षिप्त रूप में ओ.पी. कोड भी कहा जाता है। संक्रिया या फलन कूट को आधुनिक कंप्यूटरों में हेक्स पद्धति से लिखा जाता है।[35]

(ख) असेंबली/ संयोजन भाषा (Assembly Language/Code) :

बाइनरी संख्याओं में प्रोग्राम लिखना असंभव नहीं तो कठिन है,क्योंकि वह प्रोग्राम 0 और 1 की लंबी श्रृंखला के रूप में होगा जैसे 10011000110000110111... इत्यादि। यदि इसको लिखने में कोई गलती हो जाए तो उसे पकड़ना और ठीक करना तो लगभग असंभव ही है। इसलिए एक ऐसी भाषा का विकास किया गया, जिसमें बाइनरी संख्या को अक्षरों या चिंहों में लिखा जाता है। इस भाषा को **असेंबली भाषा** कहा जाता है। असेंबली भाषा में प्रोग्राम लिखना मशीनी भाषा की तुलना में सरल होता है, परंतु कंप्यूटर पर चलाने से पहले उसका अनुवाद (Translation) मशीनी भाषा में कराना पड़ता है। यह अनुवाद खुद कंप्यूटर एक प्रोग्राम के द्वारा कर लेता है, जिसे असेंबलर(Assembler) कहा जाता है। नीचे इस प्रक्रिया को दिखाया गया है :

असेंबली भाषा प्रोग्राम --------------> असेंबलर ----------------> मशीनी भाषा प्रोग्राम[36]

असेंबली भाषा, मशीन भाषा में आ रही कठिनाइयों को दूर करने के लिए बनाई गई थी। इसमें मशीन कोड के स्थान पर 'नेमोनिक कोड' (Nemonic Code) का प्रयोग किया गया।

जिस प्रकार यदि किसी व्यक्ति का नाम राजेंद्र कुमार है और उसे हम घरेलू नाम 'राजू' कह सकते हैं तो घरेलू नाम राजू उस व्यक्ति का नेमोनिक कोड होगा। इसी तरह कंप्यूटर के लिए भी इसी प्रकार के कोड उपयोग में लाए जाते थे। उदाहरण के लिए SUB (Subtrct) इत्यादि। प्रोग्राम इन्हीं संकेतों में लिखा जाने लगा। इनमें में से प्रत्येक संकेत (Code) के लिए एक मशीन कोड भी निर्धारित किया गया, पर असेंबली कोड से मशीन कोड में परिवर्तन का कार्य, कंप्यूटर में ही स्थित एक प्रोग्राम के द्वारा किया जाने लगा। इस प्रकार के प्रोग्राम को असेंबलर कहा जाता है।

मशीन एवं असेंबली भाषाएँ कंप्यूटर की मूल संरचनाओं से संबंधित है एवं ये अलग-अलग संरचना वाले कंप्यूटर के लिए अलग-अलग होती हैं। अत: इन्हें निम्न स्तरीय भाषाएँ कहा जाता है।[37]

एक कंप्यूटर के लिए लिखा गया प्रोग्राम दूसरे कंप्यूटरों के लिए बेकार होता है। इस कारण ऐसी प्रोग्रामिंग भाषाओं की जरूरत हुई,जो हर कंप्यूटर के लिए समान हों। परिमामस्वरूप उच्च स्तरीय भाषाओं का निर्माण हुआ।[38]

"विज्ञाचार्य" के अनुसार उच्चस्तरीय भाषाओं (High Level Languages) के उपयोग का कंप्यूटर के अवयवों की कार्यविधि से कोई संबंध नहीं है। ये सामान्य लोगों द्वारा प्रयुक्त तर्कों पर आधारित होती हैं।[39] ("विज्ञाचार्य" पृ.18)

मशीनी भाषा में प्रोग्राम बहुत क्लिष्ट और याद रखने में दुष्कर होते हैं,क्योंकि इन्हें सिर्फ द्विआधारी अंकों के रूप में लिखा जाता है। सभी अनुदेश सिर्फ दो अंकों की दुरूह श्रृंखला में लिखे जाते हैं। मशीनी भाषा में छोटे से छोटे कार्य के लिए काफी लंबा प्रोग्राम

लिखना पड़ता है। यह बहुत ही क्लिष्ट कार्य है, जिसे माइक्रो कंप्यूटर या उसके द्वारा अन्य सामग्री उत्पादित करने वाले वैज्ञानिक ही समझ सकते हैं।

प्रत्येक माइक्रो प्रोसेसर के लिए अनुदेश अलग अलग प्रकार के हैं, जबकि स्मृति पते एक ही हैं। ऐसा उन्हें उत्पादक जान-बूझ कर बनाते हैं, जिससे उनकी नकल न की जा सके।

आजकल के कंप्यूटरों के साथ इतनी अधिक निवेश या निर्गम इकाइयों एवं स्मृति आदि का प्रयोग होता है कि कम से कम द्विआधारी संख्याओं में लिखे 200 फलन कूटों की आवश्यकता हो जाती है। लेकिन इतने अधिक फलन कूटों को आसानी से याद नहीं रखा जा सकता है, इसलिए फलन कूटों को अंकों के अलावा शब्दों में लिखा जाता है, ऐसे शब्दों में, जिन्हें आसानी से याद रखा जा सके। शब्द भी पूरे न लिखे जा कर प्रतीकात्मक छोटे रूप में लिखे जाते हैं, जैसे ADDITION के लिए ADD, LOAD ACCUMULATOR के लिए LDA और STORE ACCUMULATOR के लिए STA आदि। इन प्रतीकात्मक शब्दों को **स्मृति सहायक (Mnemonic)** शब्द कहा जाता है।

इन प्रतीकात्मक शब्दों LDA, ADD, STA आदि को आसानी से याद रख सकते हैं या भूल जाने पर सामान्य बुद्धि से पुनः स्मृति में ला सकते हैं। इसी प्रकार संकार्य-संख्याओं की बजाय X, Y, NUM आदि को भी छोटे में लिखने से प्रोग्राम लिखने में लगने वाले समय में बचत कर सकते हैं। इन प्रतीकात्मक शब्दों के अर्थ मशीनी भाषा में असेंबलर नामक सॉफ्टवेयर प्रोग्राम से बदल दिए जाते हैं।

इन प्रतीकात्मक शब्दों के कई व्यक्तियों द्वारा तरह-तरह के लघु रूप बनाए जा सकते हैं। लेकिन ज्यादातर व्यक्ति थोड़ी सी मेहनत के बाद इनके प्रामाणिक स्वरूपों को याद करने में समर्थ हो जाते हैं।

असेंबली भाषा में बनाए गए प्रोग्राम के की-बोर्ड की सहायता से ऐसे कंप्यूटर में प्रविष्ट कराया जाएगा, जिसमें असेंबलर लगा हो। यह असेंबलर स्वचालित विधि से असेंबली प्रोग्राम को मशीनी प्रोग्राम में बदल देगा। मशीनी प्रोग्राम को **ऑब्जेक्ट कोड या मशीनी कोड** भी कहा जाता है। मशीनी प्रोग्राम में दिए गए फलन-कूट वास्तव में दाशमिक फलन कूटों के द्विआधारी तुल्य-संख्या के रूप में है।

असेंबली भाषा के गुण-दोष

(क) मशीनी भाषा में लिखे हुए आदेश 0 एवं 1 की लंबी श्रृंखलाओं में होते हैं, जिन्हें याद रखना बहुत दुष्कर कार्य होता है, जबकि असेंबली भाषा में स्मृति-सहायक कोड (Mnemonic) वाले प्रतीकात्मक शब्द बनाए जाते हैं, जिन्हें याद रखना बहुत आसान होता है।

(ख) प्रोग्रामर को मशीनी भाषा में प्रोग्राम लिखते समय स्मृति-स्थानों की संख्या को याद रखना होता है, जबकि असेंबली भाषा में चर-संज्ञाओं (Variable Names) की सहायता से यह कार्य किया जाता है, जो कि आसान है।

(ग) इस भाषा में लिखे आदेशों में संशोधन हेतु विलोपन (Deletion) एवं निवेशन (Insertion) कार्य सरल होता है। पूर्ण लिखे हुए प्रोग्राम को संशोधित करना भी आसान होता है। असेंबली भाषा में प्रोग्राम लिखने से समय की भी बचत होती है।
(घ) इस भाषा के इस्तेमाल में मशीनी भाषा से कम तथा उच्च स्तरीय भाषाओं से अधिक बोधगम्यता की जरूरत होती है।
(च) इसे छोटे माइक्रो प्रोसेसरों या कंप्यूटरों में इस्तेमाल नहीं किया जा सकता।
(छ) असेंबली भाषा में लिखे प्रोग्राम की दक्षता मशीनी कोड प्रोग्राम की दक्षता से कम होती है।[40]

5.3.2 उच्च स्तरीय भाषाएँ (High Level Languages)

निम्न स्तरीय भाषाओं में प्रोग्रामिंग कार्य बहुत बड़ा एवं अत्यंत जटिल कार्य था। अतः जब कंप्यूटर का प्रयोग अधिक बढ़ा तो एक ऐसी भाषा की आवश्यकता महसूस हुई जो कि प्राकृतिक भाषाओं के निकट हो ताकि सामान्य व्यक्ति भी कंप्यूटर का प्रयोग कर सके। अतः इस कारण उच्च स्तरीय भाषाओं का प्रयोग होने लगा। उच्च स्तरीय भाषाओं का प्रोग्राम लिखते समय प्रोग्रामर प्राकृतिक भाषा (अँग्रेजी) के कुछ चुने हुए शब्दों और गणित के स्वीकृत चिंहों को एक निश्चित रूप में प्रयोग कर सकता है जो भाषा को व्याकरण (Syntax) के अंतर्गत पूर्व निर्धारित होते हैं। चूँकि सीमित शब्दावली का प्रयोग एक मुख्य आवश्यकता थी, इस कारण भिन्न-भिन्न कंप्यूटर भाषाओं की रचना की गई।[41]

उच्च स्तरीय प्रोग्रामिंग भाषाएँ ऐसी प्रोग्रामिंग भाषाओं को कहा जाता है,जो हर प्रकार के कंप्यूटर के लिए समान होती हैं। दूसरे शब्दों में, इन भाषाओं का व्याकरण मशीनों (कंप्यूटरों) पर निर्भर नहीं होता और इनमें लिखा गया प्रोग्राम किसी भी ऐसे कंप्यूटर पर चलाया जा सकता है जिसमें उस भाषा का अनुवादक या कंपाइलर लगा हो। असेंबली भाषा की तरह ही किसी उच्च स्तरीय प्रोग्रामिंग भाषा में लिखे गए प्रोग्राम को भी किसी कंप्यूटर पर चलाने से पहले उसका अनुवाद कंप्यूटर की मशीनी भाषा में कराना पड़ता है। यह कार्य जिस सॉफ्टवेयर (प्रोग्राम) द्वारा किया जाता है,उसे कंपाइलर(Compiler) या संकलक कहा जाता है।
स्रोत प्रोग्राम----------> कंपाइलर -----------> ऑब्जेक्ट प्रोग्राम

किसी उच्च स्तरीय भाषा में लिखे गए प्रोग्राम को स्रोत प्रोग्राम (Source Program) कहा जाता है और मशीनी भाषा में किए गए अनुवाद को ऑब्जेक्ट प्रोग्राम (Object Program) कहते हैं। कंपाइलर किसी स्रोत प्रोग्राम को ऑब्जेक्ट प्रोग्राम में
बदल देता है,यह क्रिया चित्र मे समझाई गई है। वास्तव में कंपाइलर का कार्य कुछ जटिल होता है। उसके दो मुख्य कार्य हैं -
1. स्रोत प्रोग्राम की जाँच कर के उसमें व्याकरण (Syntax) की गलतियों का पता लगाना।
2. यदि कोई गलती नहीं है तो उसका मशीनी भाषा में अनुवाद करना।

कंपाइलर का छोटा भाई होता है - **इंटरप्रिटर** (Interpreter) या प्रतिपादक। यह भी स्रोत प्रोग्राम का अनुवाद मशीनी भाषा में करता है, परंतु अंतर यह है कि कंपाइलर जहाँ

पूरे स्रोत प्रोग्राम का एक साथ मशीनी भाषा में अनुवाद करता है, वहीं इंटरप्रिटर एक बार में केवल एक कथन (Statement) या एक आदेश का अनुवाद मशीनी भाषा में करता है और उसका पालन हो जाने के बाद ही अगले कथन का अनुवाद करता है। यह कार्य तब तक चलता रहता है जब तक कि खत्म नहीं हो जाता। स्पष्ट है कि इंटरप्रिटर कोई ऑब्जेक्ट प्रोग्राम तैयार नहीं करता, बल्कि सीधे स्रोत प्रोग्राम को ही पालित कराता है।[42]

मशीनी भाषा की क्लिष्टता से बचने के लिए असेंबली भाषा बनाई गई थी,लेकिन असेंबली भाषा में प्रोग्राम को हर कोई प्रोग्रामर नहीं बना सकता है क्योंकि इस भाषा में प्रोग्राम बनाने के लिए सभी प्रकार के हार्डवेयरों तथा माइक्रोप्रोसेसर सर्किट की पूर्ण जानकारी होना जरूरी है। इस कार्यविधि को सीखने के लिए कंप्यूटर विज्ञान में स्नातकोत्तर स्तर की जानकारी होना आवश्यक है।इसलिए कंप्यूटर को बहु-प्रचलित बनाने के लिए इस प्रकार की भाषा बनाना जरूरी हो गया है जिसमें अँग्रेजी जैसे शब्दों के प्रयोग से प्रोग्राम बनाए जा सकें और बाकी कार्य एक कंपायलर या असेंबलर करें। यह कंपायलर अँग्रेजी में उच्च स्तरीय भाषा के आदेशों /शब्दों को सीधे ही मशीन कोड में बदल देता है। कंपायलर में हर प्रकार के कार्य के लिए अँग्रेजी जैसे शब्दो के दिए गए आदेशो को मशीन-कोड में बदलने की व्यवस्था होती है। इस प्रकार उच्च स्तर की भाषा एवं इस भाषा से संबंधित कंपायलर प्रोग्राम की सहायता से प्रोग्रामिंग की जा सकती है। उच्च स्तर की भाषा में प्रोग्राम बनाना कोई भी व्यक्ति 6 महीने में सीख सकता है।

सबसे पहले उच्च स्तर की भाषा के लिए आई बी एम ने 1957 में कोशिश की और उसका परिणाम फोरट्रान के रूप में सामने आया। इसके पश्चात कोबोल, बेसिक, अल्गोल, पास्कल, पी एल-1, पी एल-2, कोराल, लोगो, सी, लिस्प, प्रोलाग आदि सैंकड़ों उच्च स्तरीय भाषाओं का आविष्कार हो चुका है।

इन भाषाओं को उच्च स्तरीय भाषा इसलिए नहीं कहा जाता है कि इन्हें सीखने के लिए उच्च स्तरीय ज्ञान की जरूरत होती है बल्कि इन भाषाओं का ग्राहाता स्तर इतने उच्च स्तर का है कि कोई भी पढा-लिखा व्यक्ति इन्हें आसानी से सीख सकता है। इन भाषाओं में इतना अधिक लचीलापन होता है कि थोड़े से आदेशों की सहायता से बहुत से कार्य कराए जा सकते हैं, इसलिए भी इन्हें उच्च स्तरीय भाषा कहा जाता है।

इन सभी उच्च स्तरीय भाषाओं में एक समानता है कि लगभग सभी में अँग्रेजी के वर्णों एवं इंडो-अरेबियन अंकों का प्रयोग किया जाता है।

उच्च स्तरीय भाषाओं के गुण-दोष

गुण :

1) इन्हें हाई स्कूल तक शिक्षा प्राप्त व्यक्ति आसानी से सीख सकता है।

2) इन भाषाओं में प्रोग्राम लिखने वाले व्यक्ति को कंप्यूटर की आंतरिक संरचना तथा परिपथ की जानकारी होना जरूरी नहीं है।

3) डेटा एवं जानकारी को किस स्थान में संचयित कर के रखना है, यह कार्य कंप्यूटर अपने आप करता है।

4)कंप्यूटरों को दिए जाने वाले आदेशों की संख्या एवं क्लिष्टता अपने आप कम हो जाती है। उच्च स्तरीय भाषा में दिया गया सिर्फ एक आदेश 5 से 10 असेंबली आदेशों के बराबर होता है।

5) अधिकांश उच्च स्तरीय भाषाओं को प्रायः सभी कंप्यूटरों पर चालित किया जा सकता है। इस प्रकार से ये किसी भी मशीन पर निर्भर नहीं होती हैं।

6) मशीन या असेंबली कोड में प्रोग्राम लिखने के लिए बहुत अधिक स्मरण शक्ति एवं संदर्भ सारणियों के देखने की जरूरत होती है एवं प्रोग्राम बहुत लंबे-लंबे लिखने होते हैं। मशीन कोड में इस कार्य के लिए 11 एवं असेंबली कोड में कम से कम तीन अनुदेश लिखने होते हैं, जबकि किसी भी उच्च स्तरीय भाषा में केवल 1 अनुदेश यह कार्य कर सकता है। औसतन एक उच्च स्तरीय अनुदेश को मशीन कोड में कम से कम 10 अनुदेशों के माध्यम से व्यक्त किया जाता है। सिर्फ यही नहीं हर कंप्यूटर के लिए ये अनुदेश अलग-अलग प्रकार के होंगे।

7) कई उच्च स्तरीय भाषाओ में लिखे गए प्रोग्रामों में वाक्य रचना एवं अन्य प्रकार की त्रुटियों को स्वचालित रूप से व्यक्त कर देता है।

8) कई उच्च स्तरीय भाषाओं में कंप्यूटर के द्वारा इंटरएक्टिव मोड में वार्तालाप किया जा सकता है।

दोष :

1) लगभग शत प्रतिशत उच्च स्तरीय भाषाओं में अँग्रेजी के शब्दों का इस्तेमाल होता है जो अन्य मातृभाषायी व्यक्तियों के लिए दुविधाजनक है।

2) इन भाषाओं में से लगभग सभी भाषाओं के कई-कई संस्करण प्रचलित हैं,जिनके कारण इन्हें सीखने वाले व्यक्तियों को बड़ी परेशानी का सामना करना होता है। बेसिक और लोगो के अनगिनत रूप प्रचलित हैं। हालाँकि अमेरिका राष्ट्रीय मानक संस्थान के द्वारा इनमें से कई भाषाओं के मानक रूप निर्धारित कर दिए गए हैं,लेकिन अधिकांश निर्माताओं और उपभोक्ताओं के द्वारा इन मानक स्वरूपों का प्रयोग नहीं किया जाता।

3) ये भाषाएँ कंप्यूटर के ऑपरेटिंग सिस्टम और माइक्रोप्रोसेसर की डिजाइन पर भी निर्भर हैं। उदाहरणार्थ सी (C) नामक भाषा सिर्फ यूनिक्स ऑपरेटिंग सिस्टम के साथ उपयोग में लाई जा सकती है और फोरट्रान, बेसिक आदि जिन्हें एम. एस. डॉस ऑपरेटिंग सिस्टम के लिए बनाया गया है, को यूनिक्स पर नहीं चलाया जा सकता है। अब धीरे-धीरे इन भाषाओं को एक दूसरे सिस्टमों पर चलने (परिवहन होने) लायक बनाया जा रहा है।

4) ये भाषाएँ कंप्यूटर की स्मृति क्षमता पर भी आधारित होती है, जैसे कि बेसिक के परिवर्धित रूप को होम कंप्यूटर पर नहीं चलाया जा सकता।

5) कुछ माइक्रो कंप्यूटर के साथ उनकी स्मृति के रोम-क्षेत्र में बेसिक (या इसी प्रकार की अन्य भाषाओं) का कंपायलर प्रोग्राम स्थाई रूप से भर दिया जाता है। इसलिए इनमें अन्य कंपायलरों को इंटरफेस नहीं किया जा सकता।

6) उच्च स्तरीय भाषाओं को सीखना भी उतना आसान नहीं है जितना समझा जाता है। स्नातक स्तर तक पढ़े हुए विज्ञान के विद्यार्थी को भी फोरट्रान-IV को पूरी तरह समझने और इसके प्रयोग में निपुण होने में कम से कम 6 माह लग जाते हैं। इन भाषाओं में निपुणता प्राप्त करना व्यक्ति की अपनी क्षमता और बुद्धिमत्ता पर निर्भर होता है।

7) किसी उच्च स्तरीय भाषा का हर प्रकार के उपयोग के लिए प्रयोग नहीं किया जा सकता। जैसे शब्द-संसाधन (Word Processing) के लिए फोरट्रान-IV का तथा अत्यंत जटिल गणितीय संक्रियाओं के लिए कोबोल का प्रयोग नहीं किया जा सकता।

8) हर भाषा का अपना एक कंपायलर या इंटरप्रटर होता है, जिसकी कीमत 5000/- से कम नहीं होती है। ये कंपायलर इंटरप्रेटर प्रोग्राम एक फ्लापी डिस्क के रूप मे मिलते हैं, जिन्हें बाद में हार्ड डिस्क में स्टोर कर लिया जाता है और फिर इस भाषा को प्रोग्राम में प्रयोग कर लिया जाता है। अक्सर इन कंपायलरों को आसानी से टेप कर के इनकी प्रतिलिपि बनाई जा सकती है, जिसके कारण उपभोक्ताओं में चोरी की आदत पड़ जाती है।[43]

राम बंसल "विज्ञाचार्य" ने वर्तमान में प्रचलित उच्च स्तरीय भाषाओं की उनके **उपयोग क्षेत्र के आधार पर वर्गीकृत** निम्नानुसार किया है :

उपयोग क्षेत्र	भाषाएँ
व्यवसाय	कोबोल, स्प्रैडशीट, सी, पीएल-आई
विज्ञान	फोर्ट्रान, सी, सी++ ,बेसिक, पॉस्कल
कंप्यूटर परिचालन तंत्र	सी, सी++, पॉस्कल, ऐडा, बेसिक,मोड्यूला
कृत्रिम बोध	लिस्प, प्रोलॉग
प्रकाशन	टैक्स, पोस्ट स्क्रिप्ट
प्रक्रिया संयमन	यूनिक्स शैल, टीसीएल, पर्ल, मार्वेल
नवीन उपयोग	एम एल, स्माल टॉक, ऐफेल[44]

5.3.3 मध्यवर्ती (इंटरफेस) भाषाओं का भाषावैज्ञानिक अध्ययन

(i) प्राकृतिक और अप्राकृतिक भाषाओं में अंतर

प्राकृतिक भाषा मानव-जीवन का अभिन्न अंग है। मानवों के बीच परस्पर संवाद स्थापित करने और सूचनाओं को लिपिबद्ध करके अभिलिखित करने का यह एक प्रमुख साधन है। इसके माध्यम से मानव जटिल, सूक्ष्म, गहन और व्यापक विचारों को अभिव्यक्त करने में सफल हो जाता है। **भाषाविज्ञान में इसे व्यवस्थाओं की व्यवस्था कहा गया है।** किंतु हमारे जीवन के साथ यह इस कदर गुंथी है कि हम इसकी शक्ति और प्रभाव को अनदेखा कर देते हैं। **अभिकलनात्मक भाषाविज्ञान इसी शक्ति को आत्मसात करने के लिए प्रयत्नशील है। प्राकृतिक भाषा संसाधन (प्राभासं) अभिकलनात्मक भाषाविज्ञान का ही एक अंग है।** इसका उद्देश्य कंप्यूटर के ऐसे व्यापक मॉडल और डिज़ाइन तैयार करना है, जिनकी सहायता से मानव-मशीन के बीच संवाद स्थापित हो सके। अधिकांश मानव-संवाद भाषा के लिखित या मौखिक रूप मे होता है। इसलिए प्राभासं का सर्वप्रथम लक्ष्य तो यही है कि **इसकी सहायता से लिखित पाठ या मौखिक भाषा का विश्लेषण किया जा सके** किंतु यह विश्लेषण भी मात्र स्वनिमों को पहचानने तक सीमित नहीं रह सकता। इसके लिए प्राकृतिक भाषा को समझना आवश्यक है। वस्तुतः **कंप्यूटर प्रोग्राम रूपबद्ध प्रणाली (Formal System) का ही मूर्त रूप है।** कंप्यूटर के भीतर जो भी परिचालन होता है, वह द्वयंक (binary) सिद्धांत पर ही आधारित होता है। इस प्रकार के निर्देश और डाटा अंततः द्वयंक रूप में ही परिवर्तित

होने चाहिए। कंप्यूटर की आंतरिक भाषा, जो द्वयंक जगत का संचालन करती है, मशीनी भाषा (Machine language) कहलाती है।

प्रोग्रामन भाषाओं को प्राकृतिक भाषा की कोटि में नहीं रखा जा सकता, क्योंकि अनेक दृष्टियों से ये भाषाएं हिंदी, अँग्रेजी, चीनी, रूसी आदि वास्तविक भाषाओं (real languages) से भिन्न होती हैं। इसलिए इन्हें हम रूपबद्ध भाषा (Formal language) कह सकते हैं। अप्राकृतिक प्रोग्रामन भाषाओं का प्रयोग मुख्यत: कंप्यूटर विशेषज्ञों द्वारा ही किया जाता है। **इन भाषाओं के सिद्धांत गणित, तर्कशास्त्र आदि के मॉडल पर आधारित होते हैं।** इन सिद्धांतो का उद्देश्य सुसंबद्ध ढांचा (Coherent framework) तैयार करना है। रूपबद्ध भाषा का सिद्धांत आश्चर्यजनक रूप से मुख्यत: **तीन क्षेत्रों में सफल हुआ है** : व्याकरण को परिभाषित करने की जटिलता के आधार पर भाषाओं का वर्गीकरण, पदनिरूपकों (Parsers) का अर्था विश्लेषण (explication), सरलीकरण (simplification) और विशिष्ट संरचना (Specific Construction) तथा वाक्यपरक व संरचनात्मक रूपांतरणों का सैद्धांतिक अध्ययन, वाक्य विन्यास को अर्थ विज्ञान के साथ संबद्ध करने में इसे विशेष सफलता नहीं मिल पाई है।

प्राकृतिक भाषा संसाधन

प्राकृतिक भाषा संसाधन का कार्य अपने आप में अत्यंत जटिल है, किंतु इस क्षेत्र में सर्वाधिक कठिन कार्य है - विभिन्न प्रकार के ज्ञान, अर्थपरक ज्ञान और संदर्भपरक ज्ञान को समन्वित करके भाषा संसाधन का कार्य करना। इस प्रकार के कार्य को सुगम बनाने के लिए **कुछ सामान्य तकनीकें** भी हैं। इनमें से एक तकनीक है **मॉड्यूल बनाना** (Modularity) अर्थात् अपनी प्रणाली के ज्ञान को अपेक्षाकृत अधिक स्वायत्त उपादानों में विभाजित करना। जटिल प्रणाली को सुगम बनाने की दूसरी तकनीक है **रूपबद्ध मॉडलों** (Formal Models) का प्रयोग। बड़े प्रोग्राम डिजाइन करना, उनमें संशोधन करना ज्यादा कठिन होता है इसलिए आरंभ में सरल और छोटा मॉडल बनाया जाता है और बाद में अपनी प्रणाली को उस मॉडल के कार्यान्वयन के लिए बढ़ाते चले जाते हैं।

अधिकांश प्राकृतिक भाषा प्रणालियों में भाषा-विश्लेषण और भाषा-रचना दोनों क्षमताओं की आवश्यकता होती है। व्यावहारिक प्रणालियों के लिए विश्लेषण का महत्त्व सर्वाधिक है। प्राकृतिक भाषा में प्रयुक्त वाक्य को समझने के लिए वाक्य के साथ-साथ प्रोक्ति (Discourse) स्तर पर विश्लेषण भी आवश्यक है, लेकिन कुछ छोटे अनुप्रयोगों में वाक्य स्तर तक विश्लेषण ही पर्याप्त रहता है।[45]

(ii) मध्यवर्ती (इंटरफेस) भाषाओं का भाषावैज्ञानिक अध्ययन

1) मध्यवर्ती भाषा का निर्माण प्रोग्रामर द्वारा किया जाता है। अत: इस भाषा के माध्यम से केवल वही व्यक्त होता है जो प्रोग्रामर अभिव्यक्त करना चाहता है। इस कारण यह भाषा प्रोग्रामर का माउथ-पीस होता है और प्रोग्रामर की जुबान ही बोलता है।

2) प्रोग्रामर जिस भाषा में कंप्यूटर को प्रशिक्षित करेगा अर्थात् मध्यवर्ती भाषा का निर्माण करेगा, कंप्यूटर उसी भाषा को अपना लेगा, क्योंकि कंप्यूटर में किसी भी प्रकार का पूर्वाग्रह

नहीं होता है। अभी तक कंप्यूटर के लिए प्रोग्रामन का कार्य अँग्रेजी भाषा में ही किया जाता रहा है, इसीलिए प्रोग्रामन की भाषा के रूप में अँग्रेजी प्रतिष्ठित हो गई हैं। विश्व की कुछ अन्य भाषाओं में भी इस दिशा में प्रयास किए गए हैं। लेकिन हिंदी में अभी तक ऐसे प्रयास नहीं किए गए हैं। अतः प्रोग्रामन की भाषा के रूप में हिंदी विकसित नहीं हुई है। इसका कोई तकनीकी कारण नहीं है। तकनीकी दृष्टि से हिंदी भाषा में भी प्रोग्रामन किया जा सकता है।

3) मध्यवर्ती भाषा के रूप में प्रोग्रामर की क्षमता और कौशल सामने आती हैं। प्रोग्रामर जितना मेधावी होगा, मध्यवर्ती भाषा उतनी ही अधिक सक्षम तथा शक्तिशाली होगी। मध्यवर्ती भाषा के रूप में हिंदी के विकसित न होने का अर्थ है कि प्रोग्रामर की क्षमता और कौशल को निखारने की आवश्यकता है।

4) कंप्यूटर के प्रोग्रामन का निर्धारण उसके स्मृति-स्थान से भी किया जाता है। स्मृति-स्थान की सीमाओं को देखते हुए प्रोग्रामन-भाषा का संक्षेपण आवश्यक हो जाता है। जैसे Addition के लिए ADD का प्रयोग किया जाता है। इसी प्रकार अनेक आदेशों तथा अनुदेशों का संक्षेपण करते हुए प्रोग्रामन तैयार किए जाते हैं।

5) प्रोग्रामन भाषा का अपना व्याकरण होता है, अपना वाक्य-विन्यास होता है। इसके अंतर्गत प्राकृतिक भाषा (मानव-भाषा) और कंप्यूटर की बाइनरी भाषा में तालमेल बिठाया जाता है। जिस प्रकार देवनागरी में लिखने पर जो भाषा हिंदी कहलाती है तथा अरबी-फारसी में लिखे जाने पर उर्दू कही जाती है, उसी प्रकार प्रोग्रामन भाषा का मिला-जुला रूप कंप्यूटर से रूबरू होते समय बाइनरी भाषा के करीब पहुँच जाता है और मनुष्य से मुखातिब होते समय प्राकृतिक भाषा के नजदीक आ जाता है।

6) मध्यवर्ती भाषा को दो भागों में वर्गीकृत किया जाता है - निम्नस्तरीय भाषा तथा उच्च-स्तरीय भाषा। निम्नस्तरीय भाषाओं में अधिक अनुदेश लिखने पडते हैं जबकि उच्चस्तरीय भाषा में उतने ही कथन के लिए कम अनुदेश लिखने पडते हैं।

7) निम्नस्तरीय भाषा का प्रोग्राम कंप्यूटर-विशेष के लिए लिखा जाता है। अतः उसका प्रयोग दूसरे कंप्यूटर पर नहीं किया जा सकता है। इस कारण निम्नस्तरीय भाषा कंप्यूटर जगत में सर्वव्यापी-भाषा के रूप में विकसित नहीं हुई है।

8) उच्चस्तरीय भाषाएँ विशेष प्रयोजन हेतु तैयार की जाती है, जैसे फोरट्रान का प्रयोग विज्ञान संबंधी कार्यों के लिए किया जाता है। उन्हें अन्य प्रयोजन हेतु प्रयुक्त नहीं किया जा सकता। इस प्रकार इन भाषाओं का कार्यक्षेत्र सीमित होता है और इन भाषाओं में प्राकृतिक भाषाओं जैसी व्यापकता तथा विविधता नहीं होती है। प्राकृतिक भाषा का प्रयोग सभी प्रयोजन के लिए किया जा सकता है।

5.4 द्विआधारी/द्विअंकीय (बाइनरी) भाषा

5.4.1 मानक कूट
- (क) ऑस्की
- (ख) इस्की
  - (i) इस्की कोड
  - (ii) इस्की-7
  - (iii) 7-बिट मोड

5.4.2 संख्या पद्धति
- (क) दाशमिक /दशमलव पद्धति
- (ख) द्विआधारी पद्धति
  - (i) दाशमिक से बाइनरी में बदलना
  - (ii) बाइनरी से दाशमिक में बदलना
  - (iii) बाइनरी अंकगणित (योग, घटाव, गुणन, विभाजन)
  - (iv) बाइनरी संख्या के लाभ
- (ग) अष्टाधारी पद्धति
  - (i) दाशमिक से अष्टमिक में बदलना
  - (ii) अष्टमिक से दाशमिक में बदलना
  - (iii) बाइनरी से अष्टमिक में बदलना तथा इसका उल्टा
- (घ) षोडश आधारी पद्धति
  - (i) बाइनरी से षट्दाशमिक तथा इसका उल्टा
  - (ii) अष्टमिक से षट्दाशमिक तथा इसका उल्टा

5.4.3 बाइनरी भाषा का भाषावैज्ञानिक अध्ययन

5.4 मशीनी भाषा/द्विअंकीय (बाइनरी) भाषा

कंप्यूटर विद्युत धारा प्रवाह द्वारा संचालित होने वाली मशीन है। कंप्यूटर में विद्युत धारा प्रवाह के दो प्रकार के उपयोग हैं, एक - शक्ति स्रोत के रूप में तथा दूसरा - परिचालन संकेत के रूप में। शक्ति स्रोत के रूप में विद्युत धारा इसके विभिन्न ऑपरेटिंग सिस्टमों की मोटरों आदि का परिचालन करती है। सामान्यतः यह कार्य 12 वोल्ट पर संपादित होते हैं।

कंप्यूटर में विद्युत धारा का दूसरा कार्य अधिक महत्वपूर्ण है, यद्यपि इसमें विद्युत शक्ति की मामूली सी मात्रा ही प्रयुक्त होती है। यह कार्य कंप्यूटर की कार्य-प्रणाली के अनुसार संकेतों का उत्पादन करना है। इन्हीं संकेतों के आधार पर कंप्यूटर सभी प्रकार के कार्य संपन्न करता है। यह कार्य 5 वोल्ट पर संपादित होता है।

कंप्यूटर के परिचालन में केवल दो संकेतों का प्रयोग किया जाता है - विद्युत धारा की उपस्थिति अथवा अनुपस्थिति। इन्हीं दो संकेतों की संभावनाओं में श्रृंखला संयोजन से कंप्यूटर अपने सभी कार्य संपादित करता है। सुविधा के लिए इन दो संकेतों को 0 तथा 1 से दर्शाया जाता है। इसी कारण कंप्यूटर परिचालन को बाइनरी परिचालन कहा जाता है।

कंप्यूटर की इस प्रणाली के प्रत्येक अंक को बिट (BIT - BInary Digit) कहा जाता हैं। 8 बिटों के समूह को एक बाइट (BYTE) कहा जाता है। प्रारंभिक कंप्यूटर 8-8 बिटों के समूह में सूचना प्रवाह करते थे, किंतु आधुनिक कंप्यूटर 32 बिटों के समूहों में सूचना प्रवाह करते हैं।

बाइनरी सिस्टम के आधार पर सभी प्रकार की गणितीय प्रक्रियाएँ भी की जा सकती हैं। इसे बाइनरी गणित (Binary Mathematics) कहा जाता है।

5.4.1 मानक कूट

5.4.1(क) आस्की (ASCII)

(सूचनाओं के आंतरिक अंतरण के लिए अमेरिकन मानक कूट

American Standard Code for Information Interchange)

प्रत्येक कंप्यूटर पयोगकर्ता अंकों, अक्षरों तथा संकेतों के लिए बाइनरी सिस्टम पर आधारित कूट या कोड (Code) निर्माण कर कंप्यूटरों को परिचालित कर सकता है। किंतु उसके कोड केवल उसी के द्वारा निर्मित प्रोग्रामों व आदेशों के लिए लागू होंगे। इससे कंप्यूटरों के प्रयोगकर्ता परस्पर सूचनाओं का आदान-प्रदान तब तक नहीं कर सकते जब तक कि वे एक-दूसरे द्वारा प्रयुक्त कोड संकेतों से परिचित न हों।सूचनाओं के आदान-प्रदान में सुविधा के लिए अमेरिका में **अमेरिका राष्ट्रीय मानक संस्थान द्वारा** एक मानक कोड तैयार किया गया,जिसे अब पूरे विश्व में मान्यता प्राप्त है। इसे आस्की (ASCII) के नाम से जाना जाता है।[46]

आस्की के दो स्वरूप हैं जिन्हें क्रमशः आस्की तथा संवर्धित आस्की कहा जाता है। आस्की 7 बिटों का कूट है तथा संवर्धित आस्की 8 बिटों का। इस कारण से आस्की में 2 = 128 कूट संकेत हैं जबकि संवर्धित आस्की में 2 = 256 कूट संकेत हैं।

आस्की कूटों की सूची को एक से प्रारंभ कर क्रम देने से प्रत्येक कूट की क्रम संख्या को उस कूट का आस्की मान कहते हैं। इसका उपयोग कुंजीपटल के उपयोग से कंप्यूटर परदे पर आस्की अक्षरों को दर्शाने हेतु किया जाता है। आस्की मान बाइनरी कूट के दशमलवीय मान से एक अधिक होता है क्योंकि बाइनरी कूट शून्य से प्रारंभ होता है तथा आस्की मान एक से।[47]

इसमें प्रत्येक रोमन अक्षर, अंक व संकेत को 8 बिट, अर्थात एक बाइट से दर्शाया गया है। 8 स्थानों पर **0 तथा 1 के स्थापन से 256 विविध समूह निर्मित होते हैं**। इनमें से शून्य से 127 तक की 128 संभावनाओं को विभिन्न कार्यों, अक्षरों, अंकों व संकेतों के लिए नियुक्त कर दिया गया है। शेष 128 को विभिन्न प्रोग्रामरों द्वारा अन्य उपयोगों के लिए स्वतंत्र रखा गया है।

इन 128 संख्याओं में से 0 से 31 तक की 32 संख्या कोड कंप्यूटर के आंतरिक कार्यों के आदेशों के आदान-प्रदान के लिए आरक्षित हैं तथा 33 से 127 तक के संख्या कोड शेष संकेतों व अक्षरों के लिए निर्धारित किए गए हैं।

आस्की कोड सभी प्रकार के कंप्यूटर सिस्टमों के लिए सामान्य हैं। किंतु IBM-PC के समतुल्य कंप्यूटरों में 256 सिंबलों का प्रयोग किया गया है। इन आंतरिक सिंबलों में विदेशी सिंबलों के अतिरिक्त रेखाएँ, आयत, वृत्त आदि के चित्रण की भी व्यवस्था है। 256 कोडों वाली सूची को संवर्धित आस्की (Extended ASCII) कहा जाता है।[48]

5.4.1 (ख) इस्की (ISCII)
(सूचनाओं के आंतरिक अंतरण के लिए भारतीय लिपि संबंधी मानक कूट) Indian Script Standard Code for Information Interchange)

कंप्यूटर में दुनिया भर में किए जाने वाले कार्यों को एक सामान्य प्लेटफार्म की आवश्यकता प्रारंभ से ही महसूस की जा रही थी। इसलिए कोडिंग स्टैंडर्ड्स का प्रादर्भाव हुआ। BCD (बाइनरी कोडेड डैसिमल), EBCDIC (एक्सटेंडेड बाइनरी कोडेड डेसिमल इंटरचेंज कोड), के बाद आज **ASCII** (अमरीकन स्टैंडर्ड कोडिंग फॉर इंफॉर्मेशन इंटरचेंज) हमारे सामने है।

ऐसी ही किसी कोडिंग प्रणाली की आवश्यकता 80 के दशक में हमारे देश मे महसूस की गई, जब अधिकांश सॉफ्टवेयर डेवलपर अपनी-अपनी स्थानीय भाषाओं में वर्ड प्रोसेसर, स्प्रैडशीट और अन्य एप्लीकेशन बनाने में लगे थे।

उस समय ये डेवलपर भारतीय भाषाओं के लिए अलग-अलग कोडिंग तैयार कर अपनी एप्लीकेशनों में इनका प्रयोग कर रहे थे। इन एप्लीकेशनों द्वारा तैयार किए गए डेटा

को अन्य एप्लीकेशनों में स्थानांतरित करना लगभग असंभव होता था। अपने पक्ष को सही सिद्ध करने के लिए इन डेवलपरों का तर्क यह होता था कि ये अपने डेटा के अन्य एप्लीकेशनों में खुल सकने की संभावनाओं को कम कर डेटा के अन्य एप्लीकेशनों में खुल सकने की संभावनाओं को कम कर डेटा सिक्योरिटी प्रदान कर रहे हैं। परंतु वास्तव में ऐसा नहीं था। इनका मुख्य उद्देश्य अपने यूजर्स को अन्य डेवलपरों से अलग रख कर उन्हें पूरी तरह अपने पर आश्रित करना होता था। परंतु यूजर्स की आवश्यकता एक ऐसी कोडिंग की थी जिसके आधार पर बिना किसी परेशानी के डेटा को दूसरी एप्लीकेशनों में उपयोग किया जा सके। स्मार्ट कंप्यूटिंग (जुलाई,2001) नामक पत्रिका में प्रकाशित जानकारी के अनुसार जुलाई 1983 में डिपार्टमेंट ऑफ इलेक्ट्रॉनिक्स (DOE) ने ISCII (इंडियन स्क्रिप्ट कोड फॉर इंफॉर्मेशन इंटरचेंज) कोड पद्धति को जारी किया। 8-बिट की इस प्रणाली में नीचे के 128 अक्षर 7-बिट वातावरण के आस्की सैट के समान ही थे। ऊपर के 128 कैरेक्टरों में 10 भारतीय लिपियों के अक्षरों का समावेश किया गया था। इन 10 लिपियों में देवनागरी, पंजाबी, गुजराती, उड़िया, बंगाली, असमी, तेलुगु, कन्नड, मलयालम तथा तमिल थीं। इन सभी लिपियों के मूल में ब्राह्मी ही थी। देवनागरी के मूल से जो भाषाएँ लिखी जाती हैं, उनमें हिंदी, संस्कृत, मराठी और नेपाली भी हैं।

सौभाग्यवश इन सभी लिपियों में फॉनेटिक संरचना भी एक सी ही है जिसके कारण एक ही कॉमन कैरेक्टर सैट बनाना संभव हो सका है। इसी कारण मात्र उनके दिखाई देने वाले रूप को परिवर्तित कर के ही उनके बीच भाषानुवाद संभव हो जाता है। इसी कारण आज मिलने वाले सभी बहुभाषी सॉफ्टवेयरों में फॉनेटिक की-बोर्ड भी प्रदान किया जाता है। आज सभी भारतीय भाषाओं की वेब साइटों पर भी टाइप करने के लिए फॉनेटिक प्रणाली का उपयोग किया जाता है।[49]

यह 8 बिटों का कूट होने के कारण 256 अक्षरों के कूट निर्धारित करता है। एक से 128 तक के कूट संकेत संवर्धित आस्की के ही रखे गए हैं। शेष 128 कूट संकेतों को देवनागरी लिपि के वर्णों एवं अंकों के लिए उपयोग किया गया है।[50]

i) इस्की कोड

सूचनाओं के आदान-प्रदान के लिए भारतीय मानक कोड को 1986 में DOE ने मानकीकृत किया था, जिसे 1988 में संशोधित किया गया। यह 8-बिट चरित (कैरेक्टर कोड हेतु आई एस ओ की संस्तुतियों के अनुरूप है।)

इस्की ने अपने निचले आधे हिस्से में मानक आस्की को स्थान दिया है। ऊपर के आधे हिस्से में सभी भारतीय लिपियों के लिए आवश्यक चरित (कैरेक्टर) दिए गए हैं। इन मूल चरितों को मिला कर भारतीय भाषाओं में संयुक्त अक्षर/चरित बनाए जा सकते हैं।

नुक्ता कोड का प्रयोग नुक्ता युक्त अक्षरों /चरितों तथा कदाचित प्रयोग में आने वाले कुछ अन्य अक्षरों /चरितों के लिए किया जाता है।

हलंत कोड का प्रयोग "आधे व्यंजन" दर्शाने के लिए किया जाता है। आधे व्यंजन को किसी व्यंजन के साथ जोड कर संयुक्त रूप में प्रस्तुत किया जाता है।

एटीआर (एट्रीब्यूट /लक्षण विशेष) के बाद किसी अँग्रेजी अक्षर के आने पर, इसका प्रयोग किसी विशेष डिस्प्ले या फोंट एट्रीब्यूट को दर्शाने के लिए किया जाता है।

ईएक्सटी कोड (एक्सटेंडेड /विस्तारित) इस्की अक्षर के बाद आता है तथा विस्तारित इस्की अक्षर के विस्तारित स्वरुप को दर्शाता है।

इन मूल अक्षरों के माध्यम से प्रत्येक शब्द अनन्य रूप से प्रस्तुत किया जाता है। इसका अर्थ है कि किसी शब्द की वर्तनी प्रभावित नहीं होती है जिसके माध्यम से संयुक्ताक्षरों को प्रदर्शित किया जाता है।

चूँकि अक्षर लिपियों से मुक्त होते हैं, अत: विभिन्न भारतीय लिपियों में तत्काल लिप्यंतरण प्राप्त किया जा सकता है।

वर्ण-माला का क्रम ऐसा है कि भारतीय शब्दकोश क्रम के अनुसार अँग्रेजी छँटाई पैकेज से यथोचित छँटाई संभव है।

ii) इस्की-7

7-बिट इस्की उन कंप्यूटरों तथा पैकेजों के लिए है जो 8-बिट कोड के अनुसार कार्य नहीं करते हैं। यहाँ भारतीय लिपि वाले अक्षर /चरित को 7-बिट आस्की सैट के तहत प्राप्त किया जाता है। यह अँग्रेजी अक्षरों के स्थान पर भारतीय लिपि वाले अक्षर परिभाषित कर के प्राप्त किया जा सकता है।

चूँकि सभी इस्की चरितों (कैरेक्टर) को सीधे-सीधे स्थान नहीं दिया जा सकता है, अत: नुक्ता चरित (कैरेक्टर) का प्रयोग कम प्रयुक्त होने वाले कुछ इस्की चरितों को प्राप्त करने के लिए किया जाता है। केवल 'a' स्वर ही सीधे-सीधे दिया गया है, अन्य के मामले में संगत मात्रा के बाद 'a' लगाया जाता है।[51]

भारतीय लिपियों के लिए केवल अँग्रेजी अक्षरों के कोडों का प्रयोग सुनिश्चित करता है कि उनका प्रयोग अँग्रेजी के सभी पैकेजों के साथ किया जा सकता है। इस्की जैसे समान समतुल्य चरित (कैरेक्टर) सैट होने के अलावा, इस्की-7 भी समान छँटाई क्रम व्यवस्था उपलब्ध कराता है। इस्की-7 कोड को इनस्क्रिप्ट कुँजीपटल पर कार्यान्वित किया जा सकता है, जिससे प्रयोक्ता को भिन्न प्रकार से टाइप करने की आवश्यकता नहीं पडती है।[52]

iii) 7-बिट मोड

7-बिट मोड उन पैकेज तथा कंप्यूटर पर भारतीय लिपियों की सुविधा प्रदान करता है, जो 8वें बिट को स्वीकार नहीं करते हैं। यह केवल डेटा पेज पर ही उपलब्ध है। यह इस्की-7 कोड का प्रयोग करता है,जो कार्य के स्तर पर इस्की कोड के समरुप है। भारतीय लिपि वाले चरितों /कैरेक्टरों को अँग्रेजी के बडे तथा छोटे 52 चरितों के स्थान पर

मानचित्रित किया गया है। चूँकि अँग्रेजी और हिंदी दोनों लिपियों के अक्षरों को प्रस्तुत करने के लिए समान अक्षरों का प्रयोग किया जाता है, अत: उनके अंतर को स्पष्ट करने की आवश्यकता पड़ती है।

'**x**' चरित (कैरेक्टर) का प्रयोग शब्द के आरंभ में किया जाता है जिससे उस शब्द के भारतीय लिपि वाला शब्द होना प्रकट हो जाए अन्यथा वह शब्द अँग्रेजी का मान लिया जाएगा। छोटे '**x**' चरित (कैरेक्टर) का प्रयोग अँग्रेजी में बहुत कम शब्दों जैसे "जेनोफोबिया (Xenophobia) " आदि के आरंभ में किया जाता है,अत: इसके प्रयोग से किसी प्रकार की व्यावहारिक कठिनाई नहीं होती है।

इस्की-7 कोड में '**x**' के स्थान पर नुक्ता चरित (कैरेक्टर) को परिभाषित किया गया है। नुक्ता किसी भी भारतीय लिपि वाले शब्दके आरंभ में नहीं आ सकता है। '**x**' को नीचे रेखांकन के रूप में निरूपित किया गया है।

डबल '**xx**' सिंगल '**x**' के समतुल्य है।इसकी आवश्यकता '**x**' से आरंभ होने वाले अँग्रेजी शब्द लिखते समय पड़ती है।

अकेले प्रयुक्त '**x**' के बाद रिक्त स्थान या अनाक्षर होने पर, इसका अर्थ '**x**' ही होता है, न कि नुक्ता।

हाइफ़न से जुड़े अँग्रेजी शब्द '**-x**' भी अँग्रेजी शब्द ही रहेंगे और इसे भारतीय लिपि में बदला नहीं जाएगा।

7-बिट मोड में इनस्क्रिप्ट अधिचित्र (ओवरले), टंकण पैटर्न समान रखते हुए समतुल्य इस्की-7 चरित (कैरेक्टर) भेजता है। भारतीय लिपि वाले शब्द के आरंभ में '**x**' स्वत: जुड़ जाता है। यह जिस्ट टर्मिनल में फर्मवेयर द्वारा किया जाता है।[53]

5.4.2 संख्या प्रणालियाँ /पद्धतियाँ (Number Systems)

कंप्यूटर मुख्यत: एक परिकलन यंत्र है एवं प्रत्येक कार्य गिन कर करता है। इसका सारा कार्य संख्याओं पर आधारित है। कंप्यूटर प्रणाली के अभिकल्पन एवं उपयोग में कई प्रकार की संख्या पद्धतियों का उपयोग किया जाता है। कंप्यूटर की कार्यप्रणाली में दशमलव पद्धति का प्रयोग नहीं होता वरन् द्विआधारी (बाइनरी) या षोडश (हेक्सा डेसीमल) पद्धति का प्रयोग होता है।

दशमलव पद्धति के अलावा हम एक और पद्धति का उपयोग करते हैं जिसे रोमन पद्धति कहा जाता है, जो पाँच अंकों पर आधारित अपूर्ण पद्धति है।

कंप्यूटर में द्विआधारी पद्धति का ज्यादातर उपयोग होता है, क्योंकि कंप्यूटर का मुख्य गणना अवयव एक प्रकार का स्विच होता है, जिसकी दो स्थितियाँ होती हैं : बंद

(Closed) या खुला (Open)। स्विच बंद होने की स्थिति में विद्युत विभव या स्पंदन (पल्स) उत्पन्न होता है और खुला होने पर विद्युत स्पंदन उत्पन्न नहीं होता। इस प्रकार विद्युत स्पंदन होने या न होने की दो स्थितियों से गणना कार्य किया जा सकता है। कृत्रिम स्मृतियों में भी इस द्विआधारी पद्धति का प्रयोग किया जाता है, जिसमें दो विभिन्न स्थितियाँ निम्नलिखित हो सकती हैं :

- चुंबकत्व का होना या न होना (चुंबकीय फीते एवं डिस्क)
- विद्युत आवेश का होना या न होना (चार्ज्ड कपल डिवाइसेस)
- छिद्र का होना या न होना (छिद्रित पत्रक)
- स्विच का खुला होना या बंद होना (गेट)
- स्पंदन का होना या न होना (स्विचिंग सर्किट)

बूले के द्विआधारी तार्किक बीजगणित एवं क्लाड्ड शैनान के द्वारा विकसित स्विचन परिपथों की मदद से सभी प्रकार के परिकलन-कार्य कंप्यूटरों में किए जा सकते हैं।[54]

जिस प्रणाली के अनुसार संख्याएँ लिखी जाती हैं और उनका मान निकाला जाता है, उसे संख्या प्रणाली कहते हैं।

5.4.2 (क) दाशमिक/दशमलव प्रणाली (Decimal Number System)

हम अपने दैनिक कार्यों में जिस संख्या प्रणाली का प्रयोग करते हैं, उसका नाम दाशमिक संख्या प्रणाली है।[55]

इस पद्धति में कुल मिला कर दस अंक होते हैं जिन्हें आजकल भारतीय-अरब पद्धति में लिखा जाता है।[56]

संख्या 3235 में 3 का अंक दो बार प्रयोग किया गया है, पहली बार इसका मान 3000 है और दूसरी बार 30 है। किसी संख्या में प्रत्येक अंक के दो मान होते हैं - **मुख मान** (Face Value) तथा **स्थान मान** (Place Value)। अंक 3 का मुख मान सदा 3 ही रहता है, परंतु स्थान मान उसके स्थान मान के अनुसार बदलता रहता है। दाशमिक प्रणाली में स्थान मान दाईं ओर से बाईं ओर क्रमशः 10 गुने होते जाते हैं,जैसे इकाई, दहाई, सैंकड़ा, हजार, दस हजार, लाख, दस लाख आदि। 10 गुना होने का कारण यह है कि हमारी संख्या प्रणाली में 0 से 9 तक कुल 10 अंक हैं अर्थात इस संख्या प्रणाली का आधार (Base) 10 है।

दाशमिक प्रणाली के स्थान मानों को निम्न प्रकार समझा जा सकता है :

• स्थान	- 7	6	5	4	3	2	1
• स्थान मान	- दस लाख	लाख	दस हजार	हजार	सैंकड़ा	दहाई	इकाई
• संख्या में	- 1000,000	100,000	10,000	1,000	100	10	1
• की घात के रूप में	- 10^6	10^5	10^4	10^3	10^2	10^1	10^0

स्पष्ट है कि दाशमिक प्रणाली में स्थान मान दाईं ओर से बाईं ओर 10 के घात के रूप में बढ़ता जाता है। इसी प्रकार दशमलव बिंदु के दाईं ओर स्थान मान के घातों के रूप में घटता जाता है। अर्थात 1/10, 1/100, 1/1000, 1/10000 आदि। किसी दाशमिक संख्या का वास्तविक मान पता करने के लिए उसके प्रत्येक अंक के मुख-मान को उसके स्थान-मान से गुणा करते हैं और उन्हें जोड़ लेते हैं। उदाहरण के लिए :

$$3235 = 3x10^3 + 2x10^2 + 3x10^1 + 5x10^0$$
$$= 3x1000 + 2x100 + 3x10 + 5x1$$
$$= 3000 + 200 + 30 + 5$$
$$= 3235$$

5.4.2 (ख) बाइनरी संख्या प्रणाली (Binary Number System)

10 से कम और अधिक अंकों वाली संख्या प्रणाली भी हो सकती हैं। सभी कंप्यूटर एक ऐसी संख्या प्रणाली का उपयोग करते हैं, जिसका आधार 2 है अर्थात् जिसमें केवल 2 अंक है। कंप्यूटर 10 अंकों अर्थात् 10 स्थितियों को नहीं दिखा सकता।

ट्रांजिस्टर और चिप, जिनसे कंप्यूटर की स्मृति बनती है, स्विच की तरह कार्य करते हैं। स्विच की केवल 2 स्थितियाँ हो सकती हैं - ऑफ (Off) तथा ऑन (On)। इनके अलावा कोई तीसरी स्थिति संभव नहीं है। अतः कंप्यूटर 2 ही अंक दिखा सकता है। सुविधा के लिए ऑफ को अंक 0 मानते हैं और ऑन को 1।

अतः कंप्यूटर एक ऐसी संख्या प्रणाली में कार्य करते हैं, जिसमें केवल दो अंक हैं - 0 तथा 1, अर्थात् जिसका आधार 2 है। इस प्रणाली को बाइनरी (Binary) या द्विक या द्वि-आधारी संख्या प्रणाली कहते हैं। इस प्रणाली के अंकों को बाइनरी अंक (Binary Digit) या संक्षेप में बिट (Bit) कहा जाता है। अर्थात् किसी बिट के 2 मान हो सकते हैं - 0 और 1। अब क्योंकि बाइनरी प्रणाली का आधार 2 है, इसलिए उसके स्थान मान दाईं ओर से बाईं ओर क्रमशः 2 गुने होते जाते हैं। अर्थात् 1, 2, 4, 8, 16, 32, 64 आदि। ये संख्याएँ 2 की घात के रूप में निम्न प्रकार लिखी जा सकती हैं - 2^0, 2^1, 2^2, 2^3, 2^4, 2^5, 2^6 आदि।

किसी भी संख्या को बाइनरी में भी लिख सकते हैं। पहली 16 संख्याओं को बाइनरी में निम्न प्रकार लिखा जाता है -

दाशमिक संख्या	बाइनरी संख्या
0	0
1	1
2	10
3	11
4	100
5	101
6	110
7	111

8	1000
9	1001
10	1010
11	1011
12	1100
13	1101
14	1110
15	1111

जिस प्रकार दाशमिक में 9 के अंक के बाद बाईं ओर एक स्थान बढ जाता है, उसी प्रकार बाइनरी में 1 के अंक के बाद बायीं ओर एक बिट बढ जाती है।

(i) दाशमिक से बाइनरी में बदलना (Decimal to Binary Conversion)

किसी दाशमिक संख्या को बाइनरी में बदलने के लिए उसमें लगातार 2 का भाग देना होता है और शेषफल को क्रमशः बाईं ओर लिखते जाते हैं। यह प्रक्रिया उस समय रुक जाती है, जब भागफल शून्य हो जाता है।

उदाहरण - दाशमिक संख्या 92 को बाइनरी में बदलना

2	92	शेषफल
2	46	0
2	23	0
2	11	1
2	5	1
2	2	1
2	1	0
	0	1

1 0 1 1 1 0 0

अतः $(92)_{10} = (1011100)_2$ अर्थात् दाशमिक 92 = बाइनरी 1011100

यह पहचानने के लिए कि कोई संख्या किस प्रणाली में है, उसे कोष्ठक में रखकर उसका आधार नीचे लिख देते हैं।

उदाहरण - दाशमिक संख्या 53.6875 को बाइनरी में लिखना।

यहाँ दाशमिक संख्या के दो भाग हैं - पूर्णांक 53 तथा भिन्नांक (Fraction) .6875। इस संख्या को बाइनरी में बदलने के लिए दोनों भागों को अलग-अलग बाइनरी में बदलना पड़ेगा। पूर्णांक 53 को ऊपर के उदाहरण में बताई गई पद्धति से बाइनरी में बदल सकते हैं।

$(53)_{10} = (110101)_2$

दाशमिक .6875 को बाइनरी में बदलने के लिए इसमें 2 का गुणा बार-बार करना होगा और प्राप्त पूर्णांक को दाईं ओर बिटों के रूप मे लिखते जाना होगा। यह क्रिया तब तक चलती रहेगी जब तक या तो संख्या 0 न रह जाए या इच्छित स्थानों तक बिट न भर जाएं।

	बिट
.6875 x 2 = 1.375	1
.375 x 2 = 0.75	0
.75 x 2 1.5	1
. 5 x 2 = 1.0	1

अत: $(.6875)_{10} = (.1011)_2$

इस प्रकार $(53.6875)_{10} = (110101.1011)_2$

(ii) बाइनरी से दाशमिक में बदलना (Binary to Decimal Conversion)

किसी बाइनरी संख्या को दाशमिक में बदलने के लिए प्रत्येक बिट को उसके स्थान मान से गुणा करके जोड लेते हैं। पूर्णांक बिटों के स्थान-मान दाईं ओर से बाईं ओर क्रमश: 1, 2, 4, 8, 16, 32, 64 आदि होते हैं। इसी प्रकार बिंदु से दाईं ओर उनके स्थान-मान क्रमश: 1/2, 1/4, 1/8, 1/16, 1/32 आदि होते हैं, अर्थात् .5, .25, .125, .0625, .03125 आदि। किसी संख्या का गुणा एक से करने पर उतना ही आता है ओर शून्य से करने पर शून्य आता है। अत: वास्तव में केवल उन बिटों का स्थान मान जोडा जाता है, जिनका मुखमान 1 है।

उदाहरण - बाइनरी संख्या 1011010.10101 को दाशमिक में बदलना। हम पूर्णांक तथा भिन्नांक को अलग-अलग दाशमिक में बदलेंगे।

$(1011010)_2 = 1 \times 2^6 + 0 \times 2^5 + 1 \times 2^4 + 1 + 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0$

$= 1 \times 64 + 0 \times 32 + 1 \times 16 + 1 \times 8 + 0 \times 4 + 1 \times 2 + 0 \times 1$

$= 64 + 0 + 16 + 8 + 0 + 2 + 0$

$= (90)_{10}$

$(.10101)_2 = 1 \times 1/2 + 0 \times 1/2^2 + 1 \times 1/2^3 + 0 \times 1/2^4 + 1 \times 1/2^5$

$= 1 \times 1/2 + 0 \times 1/4 + 1 \times 1/8 + 0 \times 1/16 + 1 \times 1/32$

$= 1 \times .5 + 0 \times .25 + 1 \times .125 + 0 \times .0625 + 1 \times .03125$

$= .5 + 0 + .125 + 0 + .03125$

$= (.65625)_{10}$

अत: $(1011010.10101)_2 = (90.65625)_{10}$

(iii) बाइनरी अंकगणित (Binary Arithmetic)

किसी भी संख्या प्रणाली में गणितीय क्रियाएँ - जोड़ना, घटाना, गुणा करना और भाग देना - उसी प्रकार की जा सकती हैं, जिस प्रकार दाशमिक प्रणाली में करते हैं। बाइनरी में भी ये क्रियाएँ सरलता से की जा सकती हैं।

बाइनरी योग (Binary Addition) - जिस प्रकार दाशमिक प्रणाली मे हासिल (carry) प्राप्त होते हैं उसी प्रकार बाइनरी में भी हासिल से गणनाएं की जाती हैं। दो बाइनरी संख्याओं को जोड़ने में निम्नलिखित **चार नियमों** को ध्यान में रखना पड़ता है -

0 + 0 = 0 (हासिल कुछ नहीं)
0 + 1 = 1 (हासिल कुछ नहीं)
1 + 0 = 1 (हासिल कुछ नहीं)
1 + 1 = 0 (हासिल 1)

बाइनरी योग भी दाशमिक की तरह दाईं ओर से बाईं ओर किया जाता है और यदि कोई हासिल प्राप्त होता है, तो उसे बाईं बिट में जोड़ा जाता है।

उदाहरण - बाइनरी संख्याओं 1011100 तथा 1011010 को जोड़ना।

```
   11          (हासिल, यदि कोई हो)
  1011100
+ 1011010
---------
 10110110
```

ध्यान दीजिए कि चौथे स्थान की बिटों 1 और 1 को जोड़ने पर 10 प्राप्त होता है, जिसमें 1 हासिल है, जिसे पाँचवें स्थान की बिटों में जोड़ा जाता है। पाँचवे स्थान में हासिल सहित तीन 1 बिटें आती हैं, जिनका योग 11 होता है, जिसमें 1 हासिल है।

बाइनरी संख्याओं को दाशमिक में बदलकर और उन्हें जोड़कर हम अपने परिणाम की पुष्टि कर सकते हैं।

बाइनरी घटाव (Binary Substraction) - यह भी दाशमिक की तरह ही किया जाता है। आवश्यकता होने पर बाईं ओर से एक अंक उधार ले लिया जाता है, जिसका दाशमिक मान 2 होता है।

उदाहरण - 1011100 में से 1000111 घटाना।

```
  1011100
- 1000111
---------
  0010101
```

पहले स्थान पर बाईं ओर से उधार लेने से 10 हुए, जिसमें से 1 घटाने पर 1 बचा। दूसरे स्थान पर पुनः उधार लेना पड़ा जिससे 10 हुए, जिसमें से 10 घटाने पर शून्य बचा। तीसरे स्थान पर पुनः उधार लेना पड़ा, जिसे मिलाकर 11 हुए उसमें से 10 घटाने पर 1 बचा। चौथे स्थान पर उधार नहीं लेना पड़ा और 1 में से 1 ही घटाया गया,

जिससे 0 बचा। इसी प्रकार आगे तक समझा जा सकता है। इन संख्याओं को दाशमिक में बदलकर उत्तर की पुष्टि की जा सकती है।

बाइनरी गुणन (Binary Multiplication) - यह भी दाशमिक गुणन की तरह सरलता से किया जा सकता है। वास्तव में यह दाशमिक से अधिक सरल है, क्योंकि हमें या तो 1 या 0 से गुणा करना पडता है। केवल जोडते समय अधिक सावधानी रखनी पडती है।

उदाहरण - बाइनरी संख्याओं 1101 तथा 1011 का गुणा करना।

```
1101 * 1011
-----------
      1101
     1101
     0000
    1101
-----------
  10001111
```

(संदर्भ :[57])

बाइनरी विभाजन - यहाँ भी दाशमिक पद्धति वाले नियम लागू होते हैं।

(क) 1 : 1 = 1 (ख) 0 : 1 = 0 (ग) 0 : 0 = अर्थहीन (घ) 1 : 0= अर्थहीन

उदाहरण - $(101100)_2$ को $(100)_2$ से विभाजित करना।

```
100 ( 101100 ( 1011
      -101
      ----
        110
       -100
       ----
         100
        -100
        ----
         xxx
```

उपरोक्त विभाजन के विभिन्न चरणों की व्याख्या निम्न है :

(क) पहले 100 से 1 बार विभाजित किया इसलिए भागफल 1 लिख लिया।

(ख) 101 में से 100 को घटा लिया तो शेषफल 1 आया। अब हासिल में विभाजित होने वाली संख्या से चौथा अंक ले लिया यानि 1, जिससे यह संख्या 11 हो गई। इसमें 100 का भाग नहीं जा सकता इसलिए भागफल 0 आया एवं एक और संख्या ऊपर से हासिल में ले ली।

(ग) अब यह संख्या 110 हो गई। इसमें 100 का भाग देने पर भागफल 1 आया एवं शेषफल 10 आया।

(घ) अब विभाजित होने वाली संख्या में एक अंक और हासिल किया एवं प्राप्त संख्या 100 में 100 का भाग दिया जिससे भागफल 1 आया एवं शेष कुछ नहीं बचा।

जाँच :

$(101100)_2 = 1x2^5 + 0X2^4 + 1x2^3 + 1x2^2 + 0x2^1 + 0x2^0$
$= 32+0+8+4+0+0$
$= (44)_{10}$

$(100)_2 = 1x2^2 + 0x2^1 + 0x2^0$
$= (4)_{10}$

$(1011)_2 = 1x2^3 + 0x2^2 + 1x2^1 + 1x2^0$
$= 8+2+1$
$= (11)_{10}$
$= 44 : 4 = 11$

इसलिए $(101100)_2 : (100)_2 = 1011_2$
(संदर्भ : [58])

(iv) बाइनरी संख्या प्रणाली के लाभ (Advantages of Binary Number System)

1) संख्या प्रणालियों का आधार कुछ भी हो, उनमें प्रत्येक में वह कार्य किया जा सकता है, जो किसी दूसरी प्रणाली में किया जा सकता हो। दूसरे शब्दों में, दाशमिक संख्या प्रणाली में जो कुछ भी किया जा सकता है, वह सब बाइनरी प्रणाली में भी उतनी ही अच्छी तरह किया जा सकता है, क्योंकि प्रत्येक दाशमिक संख्या को उसकी समतुल्य (Equivalent) बाइनरी संख्या में बदल सकते हैं। इस प्रकार किसी संख्या प्रणाली के सभी लाभ बाइनरी में भी मिल जाते हैं।

2) कंप्यूटर इलैक्ट्रॉनिक पुर्जों (Electronic Components) से बना होता है, जो स्वाभाविक रूप से ऑफ और ऑन में कार्य करते हैं, अर्थात् बाइनरी में कार्य करते हैं। अतः बाइनरी प्रणाली में की जाने वाली सभी क्रियाओं को बहुत सरलता से इलैक्ट्रॉनिक सर्किटों द्वारा किया जा सकता है। दूसरे शब्दों में, इन क्रियाओं के लिए सर्किट डिजाइन करना बहुत सुविधाजनक रहता है।

3) इलैक्ट्रॉनिक सर्किट सरलता से बनाए जा सकने के कारण कंप्यूटर का मूल्य कम होता है। साथ ही उनकी विश्वसनीयता अधिक होती है।

4) बाइनरी प्रणाली में स्टोर किए गए डाटा को चुंबकीय साधनों जैसे टेप, डिस्क आदि में स्टोर करना भी सरल होता है, क्योंकि उनमें भी दो स्थितियाँ होती हैं - उत्तर-दक्षिण (North-South) तथा दक्षिण-उत्तर (South-North)।

5.4.2 (ग) अष्टमिक संख्या प्रणाली (Octal Number System)

अष्टमिक संख्या प्रणाली का आधार 8 होता है, इसलिए इसमें 0 से 7 तक कुल आठ अंक प्रयोग किए जाते हैं। अष्टमिक संख्या प्रणाली का ज्ञान होना इसलिए आवश्यक है

आधार 8 होने के कारण अष्टमिक प्रणाली के स्थान मान पूर्णांक में दाईं ओर से बाईं ओर आठ गुने होते हैं, अर्थात् क्रमशः 1, 8, 64, 512 आदि होते हैं तथा बिंदु से दाईं ओर क्रमशः .125, .015625 आदि होते हैं।

(i) दाशमिक से अष्टमिक में बदलना (Decimal to Octal Conversion)

किसी दाशमिक पूर्ण संख्या को अष्टमिक में बदलने के लिए उसमें बार-बार 8 का भाग दिया जाता है और शेषफलों को क्रम से बाईं ओर लिखते जाते हैं। जब भागफल शून्य रह जाता है तो यह काम पूरा हो जाता है। भिन्नांक दाशमिक को अष्टमिक में बदलने के लिए उसमें 8 गुणा बार-बार किया जाता है। इससे जो पूर्णांक बनते हैं उन्हें बिंदु के दाहिनी ओर लिखते जाते हैं। जरूरी संख्या में अंक मिल जाने पर या भिन्नांक शून्य हो जाने पर यह कार्य रुक जाता है।

(ii) अष्टमिक से दाशमिक में बदलना (Octal to Decimal Conversion)

किसी अष्टमिक संख्या को दाशमिक में बदलने के लिए उसके सभी अंकों को उनके स्थान-मानों से गुणा करके जोड़ लिया जाता है।

(iii) बाइनरी से अष्टमिक तथा अष्टमिक से बाइनरी में बदलना (Binary to Octal and Vice-Versa)

बाइनरी संख्या प्रणाली का आधार 2 होता है तथा अष्टमिक का आधार 8 होता है, परंतु $8 = 2 \times 2 \times 2 = 2^3$, इसलिए प्रत्येक अष्टमिक अंक को तीन बाइनरी अंकों (या बिटों) में बदला जा सकता है और प्रत्येक 3 बिटों को एक अष्टमिक अंक में बदला जा सकता है। नीचे सभी अष्टमिक अंकों के बराबर बाइनरी अंक दिए गए हैं -

अष्टमिक	बाइनरी
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

5.4.2 (घ) षट्दाशमिक संख्या प्रणाली (Hexadecimal Number System)

आजकल की ज्यादातर कंप्यूटर कंपनियाँ षट्दाशमिक संख्या प्रणाली का उपयोग करती हैं और अधिकांश कंप्यूटरों में बहुत सी सूचनाएँ इसी प्रणाली में दी जाती हैं। इसकी लोकप्रियता का कारण यह है कि इसमें बाइनरी संख्याओं (अर्थात् बाइनरी डाटा) को संक्षेप में अष्टमिक संख्या प्रणाली से ज्यादा अच्छी तरह लिखा जा सकता है और याद भी रखा जा सकता है, क्योंकि बाइनरी संख्या के 4 अंक (यानि 4 बिट) षट्दाशमिक संख्या के 1 अंक के बराबर होते हैं।

जैसा कि इसके नाम से स्पष्ट है, इस प्रणाली का आधार 16 होता है, अर्थात् इसमें 16 अंक होते हैं, जो 0 से 15 तक होते हैं, परंतु 10, 11, 12 आदि को 2 अलग-अलग अंक न समझ लिया जाए इसलिए 10, 11, 12, 13, 14 और 15 को क्रमश: A, B, C, D, E तथा F लिखते हैं। दूसरे शब्दों में, इस प्रणाली में अक्षरों A से F तक का प्रयोग अंकों के रूप में किया जाता है, जिनके मुख-मान (Face Values) क्रमश: 10, 11, 12, 13, 14 तथा 15 होते हैं। इस प्रकार षट्दाशिक संख्या प्रणाली में निम्नलिखित 16 अंकों का प्रयोग किया जाता है -

0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E तथा F

षट्दाशमिक प्रणाली में अंकों के स्थान-मान दाईं ओर से बाईं ओर 16 के घात के अनुपात में बढ़ते हैं जैसे 16^0, 16^1, 16^2, अर्थात् 1, 16, 256 आदि। इसलिए किसी दाशमिक संख्या में बार-बार 16 का भाग देकर उसे षट्दाशमिक में बदला जा सकता है।

उदाहरण - दाशमिक संख्या 23778 को षट्दाशमिक में बदलना

16	23778	शेषफल
16	1486	2
16	92	14 (अर्थात् E)
16	5	12 (अर्थात् C)
	0	5

अत: $(23778)_{10} = (5CE2)_{16}$

इसी तरह किसी षट्दाशमिक संख्या को दाशमिक में बदलने के लिए उसके सभी अंकों के मुख-मानों को स्थान-मानों से गुणा करके जोड लेते हैं। यह क्रिया निम्नलिखित उदाहरण से स्पष्ट हो जाएगी।

उदाहरण - षट्दाशमिक संख्या A30D को दाशमिक में बदलना

$$
\begin{aligned}
(A30D)16 &= A \times 16^3 + 3 \times 16^2 + 0 \times 16^1 + D \times 16^0 \\
&= 10 \times 4096 + 3 \times 256 + 0 X 16 + 13 X 1 \\
&= 40960 + 768 + 0 + 13 \\
&= (41741)_{10}
\end{aligned}
$$

(i) बाइनरी से षट्दाशमिक तथा उसका उल्टा
(Binary to Hexadecimal & Vice-Versa)

षट्दाशमिक का आधार 16 है तथा $16 = 2^4$ । इसका अर्थ है कि हर षट्दाशमिक अंक को चार बिटों के रूप में लिखा जा सकता है तथा हर चार बिटों के समूह को एक षट्दाशमिक अंक में बदला जा सकता है। इस प्रकार आठ बिटों की बाइनरी संख्या को केवल 2 षट्दाशमिक अंकों में लिख सकते हैं और उसे याद भी रख सकते हैं, जबकि किसी बाइनरी संख्या को याद करना बहुत कठिन है। नीचे प्रत्येक षट्दाशमिक अंक के बराबर दाशमिक और बाइनरी संख्याएँ प्रस्तुत हैं :

षट्दाशमिक	दाशमिक	बाइनरी
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
A	10	1010
B	11	1011
C	12	1100
D	13	1101
E	14	1110
F	15	1111

किसी बाइनरी संख्या को षट्दाशमिक में बदलने के लिए हम दाईं ओर से चार-चार बिटों के समूह बनाते हैं। जरूरी होने पर बाईं तरफ 0 बिटें बढ़ाई जा सकती हैं। समूह बना लेने के बाद प्रत्येक समूह के बदले उसके बराबर षट्दाशमिक अंक लिख दिया जाता है। इसी प्रकार षट्दाशमिक को बाइनरी में बदलने के लिए उसके प्रत्येक अंक के बदले चार बिटें लिख दी जाती हैं।

उदाहरण - बाइनरी संख्या 10110101 को षट्दाशमिक में लिखना।

$$(10110101)_2 = (\overline{1011}\ \ \overline{0101})_2 = (\ B\ \ \ 5\ )_{16} = (B5)_{16}$$

उदाहरण - षट्दाशमिक संख्या E9A को बाइनरी में लिखना

$(E9A)_{16} = (1110\ 1001\ 1010)_2$

(ii) अष्टमिक से षट्दाशमिक तथा उल्टा
(Octal to Hexadecimal and Vice-Versa)

अष्टमिक संख्याओं को षट्दाशमिक में बदलना या षट्दाशमिक को अष्टमिक में बदलना जरा टेढ़ा मामला है। इसका कारण यह है कि हमें अष्टमिक या षट्दाशमिक में गुणा भाग करने का अभ्यास नहीं है। अतः हम सीधे एक तरह की संख्या को दूसरी में नहीं बदल सकते। इसका एक सरल उपाय यह है कि पहले हम एक तरह की संख्या को बाइनरी में बदल लें, फिर बाइनरी संख्या को दूसरी तरह की संख्या में बदल लें। यह बात नीचे के उदाहरणों से अच्छी तरह स्पष्ट हो जाएगी।

उदाहरण - अष्टमिक संख्या 576 को षट्दाशमिक में बदलना

$(576)_8 = (101\ 111\ 110)_2$

$= (0001\ 0111\ 1110)_2$ (बाईं ओर 3 शून्य बढाने पर)

$= (17\ E)_{16}$

उदाहरण - $(765)_8$ में से $(123)_{16}$ को बाइनरी में घटाना।

$(765)_8 = (111\ 110\ 101)_2$

$(123)_{16} = (0001\ 0010\ 0011)_2$

$= (100100011)_2$ (बाईं ओर के शून्यों को छोड़ने पर)

अत:

```
 111110101
-100100011
 011010010
```

(संदर्भ : [59])

5.4.3 बाइनरी भाषा का भाषावैज्ञानिक अध्ययन

मानव-भाषा की लघुतम इकाई **ध्वनि** में स्वर तथा व्यंजन होते हैं, जिनकी संख्या विभिन्न भाषाओं में भिन्न-भिन्न होती हैं। लेकिन कंप्यूटर की लघुत्तम इकाई में सिर्फ 0 तथा 1 ही होते हैं। कंप्यूटर की सभी भाषाओं - प्रणाली (सिस्टम) तथा अनुप्रयोग (एप्लीकेशन) भाषाओं को अंतत: 0 तथा 1 में सिमट जाना पड़ता है। अष्टमिक तथा षट्दाशमिक संख्या पद्धतियाँ भी स्वयं को 0 तथा 1 में ही ढालती हैं।

चूँकि कंप्यूटर तकनीकी क्षेत्र से संबंधित है, अत: उसकी भाषा का स्वरूप भी तकनीकी है। अपने तकनीकी स्वरूप में यह भाषा संप्रेषण के लिए दो अंकों का प्रयोग करती है - 0 तथा 1।

वस्तुत: मानव-भाषा की ध्वनि तथा कंप्यूटर के बीच सांख्यिकीय सहसंबंध है।

लिपि के दो रूप प्रचलित हैं - अक्षर लिपि तथा चित्र लिपि। कंप्यूटर ने इसे एक नया आयाम दिया है - **अंक-लिपि**। कंप्यूटर का समस्त संप्रेषण-व्यापार अंकों के माध्यम से ही होता है। अत: लिपिबद्ध करने के लिए भी कंप्यूटर के पास केवल अंक ही होते हैं। अंकबद्ध कथन ही अंक-लिपि का विकास करते हैं।

जिस प्रकार संपूर्ण मानव-भाषा को कुछ निर्धारित अक्षर-चिंहों या चित्र-चिंहों में समेकित कर के प्रस्तुत किया जाता है, उसी प्रकार अंक-लिपि के तहत संपूर्ण मानव-भाषा को कंप्यूटर की बाइनरी भाषा में परिवर्तित करते हुए अंक-लिपि में दर्ज किया जाता है।

मानव-भाषा को बाइनरी भाषा में परिवर्तित या अनूदित करने का कार्य संख्या-पद्धति के माध्यम से किया जाता है। ये संख्या पद्धतियाँ द्विअंकीय, अष्टमिक तथा षट्दाशमिक होती हैं। इनकी सहायता से मानव-भाषा की दाशमिक पद्धति की संख्याओं तथा कंप्यूटर की बाइनरी भाषा में परस्पर संवाद कायम होता है। अष्टमिक (2x2x2) तथा षट्दाशमिक

(2x2x2x2) संख्या-पद्धतियाँ द्विअंकीय संख्या-पद्धति का ही बड़ा तथा विकसित रूप है तथा इनके माध्यम से कंप्यूटर से कार्य कराना अधिक सुविधाजनक रहता है। दूसरे शब्दों में अष्टमिक थ्री-इन-वन है तथा षट्दाशमिक फोर-इन-वन है।

आम धारणा है कि कंप्यूटर से संवाद करने के लिए केवल अंकों का ही प्रयोग किया जाता है। लेकिन षट्दाशमिक संख्या-पद्धति में अंकों के अलावा अँग्रेजी वर्णमाला **ए** से **एफ** तक का भी प्रयोग किया जाता है।

इन संख्या-पद्धतियों में मानव-भाषा को परिवर्तित करने के लिए एकरूपता बनाए रखने हेतु **आस्की** तथा **इस्की** कोड निर्धारित किए गए हैं। इसमें 256 कूट-संकेत का प्रावधान किया गया है, जो 8-बिट मोड (2^8) पर आधारित है।

5.5 अध्याय 5 की संदर्भ सूची

5.1

1. भाषा विज्ञान : भारत भूषण सरोज + डॉ.कृष्णदेव शर्मा, पृ. 1-3
2. भाषा विज्ञान : भारत भूषण सरोज + डॉ.कृष्णदेव शर्मा, पृ. 3-5
3. अक्षर भारती, विनीत चैतन्य, राजीव संगल, 1991
   (पुनर्प्रस्तुति : कंप्यूटर के भाषिक अनुप्रयोग, विजय मल्होत्रा, पृ. 159-163)
4. भाषा विज्ञान : भारत भूषण सरोज + डॉ.कृष्णदेव शर्मा, पृ. 5-7
5. भाषा विज्ञान : भारत भूषण सरोज + डॉ.कृष्णदेव शर्मा, पृ. 7-16
6. भाषाविज्ञान, डॉ.भोलानाथ तिवारी, पृ. 9-11

5.2

5.2.1

7. की-बोर्ड कैसे काम करते हैं ?, Chipहिंदी, अप्रैल,2000, पृ.39-40
8. सुनें अपने की-बोर्ड की आवाज, कंप्यूटर संचार सूचना, मई 2003, पृ.70-71
9. नए कीबोर्ड क्यों ?,कंप्यूटर संचार सूचना, जनवरी 2003, पृ.70-71
10. क्या आप जानते हैं?,तकनीकी तथ्य,चिपहिंदी(पी.सी.विशेषाँक),फरवरी2001,पृ.38
11. टैली 6.3, बादल कुमार शर्मा, पृ. 21-22
12. धो डाला !,स्मार्ट कंप्यूटिंग, जुलाई 2001, पृ. 40-41

5.2.2

13. प्रारंभिक कंप्यूटर शिक्षा, भाग-I, राम बंसल "विज्ञाचार्य", पृ. 28-31
14. प्रारंभिक कंप्यूटर शिक्षा,भाग-II, राम बंसल "विज्ञाचार्य", पृ.21-24

5.2.3

15. रैपिडैक्स कंप्यूटर कोर्स,विजय सिंघल+अमित गुप्ता+डॉ.अशोक गुप्ता,पृ.441-443
16. क्या आप जानते हैं ?, स्मार्ट कंप्यूटिंग,जुलाई 2001,पृ.26
17. ISM-2000 यूजर्स मैनुअल
18. समस्या गैर स्टैंडर्ड की, स्मार्ट कंप्यूटिंग,जुलाई 2001,पृ.62
19. ISM-2000 यूजर्स मैनुअल
20. समस्या गैर स्टैंडर्ड की, स्मार्ट कंप्यूटिंग,जुलाई 2001,पृ.62
21. ISM-2000 यूजर्स मैनुअल

5.2.4

22. आधुनिक कंप्यूटर विज्ञान, वीरेंद्र जैन,पृ. 133
23. कंप्यूटर के भाषिक अनुप्रयोग, विजय कुमार मल्होत्रा, पृष्ठ 77-81
24. हिंदी भाषा की लिपि-संरचना, डॉ.भोलानाथ तिवारी, पृ. 49-54
25. केंद्रीय हिंदी निदेशालय, नई दिल्ली द्वारा मानकीकृत मानक हिंदी वर्तनी

5.3

5.3.1

26. कंप्यूटर विज्ञान के सिद्धांत,विजय कुमार सिंघल,पृ.63
27. कंप्यूटर विज्ञान संख्या प्रणाली सहित, डॉ. सुरेंद्र चौकसे,पृ.96

28. कंप्यूटर विज्ञान के सिद्धांत,विजय कुमार सिंघल,पृ.63
29. आधुनिक कंप्यूटर विज्ञान, वीरेंद्र कुमार जैन,पृ.133
30. कंप्यूटर विज्ञान के सिद्धांत,विजय कुमार सिंघल,पृ.63
31. कंप्यूटर विज्ञान संख्या प्रणाली सहित, डॉ. सुरेंद्र चौकसे,पृ.97
32. कंप्यूटर विज्ञान के सिद्धांत,विजय कुमार सिंघल,पृ.63
33. कंप्यूटर विज्ञान संख्या प्रणाली सहित, डॉ. सुरेंद्र चौकसे,पृ.97
34. बेसिक प्रोग्रामिंग, राम बंसल "विज्ञाचार्य" पृ.18
35. आधुनिक कंप्यूटर विज्ञान, वीरेंद्र कुमार जैन,पृ.133-145
36. कंप्यूटर विज्ञान के सिद्धांत,विजय कुमार सिंघल,पृ.63-64
37. कंप्यूटर विज्ञान संख्या प्रणाली सहित, डॉ. सुरेंद्र चौकसे,पृ.97
38. कंप्यूटर विज्ञान के सिद्धांत,विजय कुमार सिंघल,पृ.64
39. बेसिक प्रोग्रामिंग, राम बंसल "विज्ञाचार्य" पृ.18
40. आधुनिक कंप्यूटर विज्ञान, वीरेंद्र कुमार जैन,पृ.141-145

5.3.2

41. कंप्यूटर विज्ञान संख्या प्रणाली सहित, डॉ. सुरेंद्र चौकसे,पृ.97
42. कंप्यूटर विज्ञान के सिद्धांत,विजय कुमार सिंघल,पृ.64-65
43. आधुनिक कंप्यूटर विज्ञान, वीरेंद्र कुमार जैन, पृ.145-146
44. कंप्यूटर सामान्य ज्ञान एवं यूज़र गाइड, राम बंसल "विज्ञाचार्य" पृ.29

5.3.3

45. कंप्यूटर के भाषिक अनुप्रयोग, विजय कुमार मल्होत्रा, पृ. 167-173

5.4

46. कंप्यूटर परिचालन तत्व, राम बंसल "विज्ञाचार्य", पृ.56
47. कंप्यूटर क्या, क्यों और कैसे, राम बंसल "विज्ञाचार्य", पृ.14
48. कंप्यूटर परिचालन तत्व, राम बंसल "विज्ञाचार्य", पृ.56-60
49. समस्या गैर स्टैंडर्ड की,स्मार्ट कंप्यूटिंग,जुलाई 2001,पृ.62
50. कंप्यूटर क्या, क्यों और कैसे, राम बंसल "विज्ञाचार्य", पृ.16
51. वीएम 4005 मल्टी-लिंगुअल वीडियो डिस्प्ले टर्मिनल-रिलीज नोट्स, पृ.9-10
52. एचसीटी 900 टर्मिनल, यूजर्स मैनुअल, पृ. 4-2
53. वीएम 4005 मल्टी-लिंगुअल वीडियो डिस्प्ले टर्मिनल-रिलीज नोट्स, पृ.10
54. आधुनिक कंप्यूटर विज्ञान, वीरेंद्र जैन, पृ.62
55. कंप्यूटर विज्ञान के सिद्धांत, विजय सिंघल,पृ.27
56. आधुनिक कंप्यूटर विज्ञान, वीरेंद्र जैन,पृ.62
57. कंप्यूटर विज्ञान के सिद्धांत, विजय सिंघल,पृ.27-32
58. आधुनिक कंप्यूटर विज्ञान, वीरेंद्र जैन, पृ.90-91
59. कंप्यूटर विज्ञान के सिद्धांत, विजय कुमार सिंघल, पृ. 32- 37

lngstcs\binrylan.doc