EBCDIC

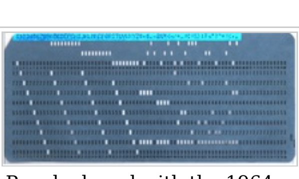
From Wikipedia, the free encyclopedia

Extended Binary Coded Decimal Interchange Code[1] (**EBCDIC**[1]) is an eight-bit character encoding used mainly on IBM mainframe and IBM midrange computer operating systems. EBCDIC descended from the code used with punched cards and the corresponding six bit binary-coded decimal code used with most of IBM's computer peripherals of the late 1950s and early 1960s.[2] It is also supported on various non-IBM platforms such as Fujitsu-Siemens' BS2000/OSD, OS-IV, MSP, and MSP-EX, the SDS Sigma series, and Unisys VS/9 and MCP.

Contents

- 1 History
- 2 Compatibility with ASCII
- 3 Code page layout
- 4 DKOI
- 5 Criticism and humor
- 6 See also
- 7 References
- 8 Further reading
- 9 External links

History



Punched card with the 1964 EBCDIC character set. Contrast at top enhanced to show the printed characters.

EBCDIC /ˈɛbsɪdɪk/ was devised in 1963 and 1964 by IBM and was announced with the release of the IBM System/360 line of mainframe computers. It is an eight-bit character encoding, developed separately from the 7-bit ASCII encoding scheme. It was created to extend the existing Binary-Coded Decimal (BCD) Interchange Code, or BCDIC, which itself was devised as an efficient means of encoding the two *zone* and *number* punches on punched cards into 6 bits. The distinct encoding of 's' and 'S' (using position 2 instead of 1) was maintained from punched cards where it was desirable to not have hole punches too close to each other to ensure the integrity of the physical card.

While IBM was a chief proponent of the ASCII standardization committee,[3] the company did not have time to prepare ASCII peripherals (such as card punch machines) to ship with its System/360 computers, so the company settled on EBCDIC.[4] The System/360 became wildly successful, together with clones such as RCA Spectra 70, ICL System 4, and Fujitsu FACOM, thus so did EBCDIC.

All IBM mainframe and midrange peripherals and operating systems use EBCDIC as their inherent encoding[5] (with toleration for ASCII, for example, ISPF in z/OS can browse and edit both EBCDIC and ASCII encoded files). Software and many hardware peripherals can translate to and from encodings, and modern mainframes (such as IBM zSeries) include processor instructions, at the hardware level, to accelerate translation between character sets.

There is an EBCDIC-oriented Unicode Transformation Format called UTF-EBCDIC proposed by the Unicode consortium, designed to allow easy updating of EBCDIC software to handle Unicode, but not intended to be used in open interchange environments. Even on systems with extensive EBCDIC support, it has not been popular. For example, z/OS supports Unicode (preferring UTF-16 specifically), but z/OS only has limited support for UTF-EBCDIC.

IBM AIX running on the RS/6000 and its descendants including the IBM Power Systems, Linux running on z Systems, and operating systems running on the IBM PC and its descendants use ASCII, as did AIX/370 and AIX/390 running on System/370 and System/390 mainframes.

Compatibility with ASCII

The fact that all the code points were different was less of a problem for inter-operating with ASCII than the fact that sorting EBCDIC put lowercase letters before uppercase letters and letters before numbers, exactly the opposite of ASCII.

Software portability and data exchange are hindered by EBCDIC's lack of codes for several symbols (such as the brace characters) commonly used in programming and in network communications.

The gaps between some letters made simple constructions that worked in ASCII fail on EBCDIC. For example, 'Z' minus 'A' was 40, not 25. This sometimes caused problems when porting software from ASCII systems.

All ASCII codes stored within an eight-bit byte had nonnegative values on systems such as the PDP-11 that treated bytes as signed quantities. Software on those platforms often took advantage of that property, causing problems when it was ported to EBCDIC-based environments where many character codes had a 1 as the "sign" bit.

By using all eight bits EBCDIC may have encouraged the use of the eight-bit byte by IBM, while ASCII was more likely to be adopted by systems with 36 bits (as five seven-bit ASCII characters fit into one word). As eight-bit bytes became widespread, ASCII systems sometimes used the "unused" bit for other purposes, thus making it more difficult to transition to larger character sets.

Code page layout

The table below is based on CCSID 037, one of the code page variants of EBCDIC; it shows only the basic (English) EBCDIC characters. Characters 00–3F and FF are controls, 40 is space, 41 is no-break space (*RSP*: "Required Space"), E1 is numeric space (*NSP*: "Numeric Space"), and CA is soft hyphen. Characters are shown with their equivalent Unicode codes. Unassigned codes are typically filled with international or region-specific characters in the various EBCDIC code page variants, but the punctuation marks and other special characters, such as cent sign, are often moved around as well; only the letters and numbers and space have the same assignments in all EBCDIC code pages.

In each table cell below, the first row is an abbreviation for a control code or (for printable characters) the character itself; the second row is the Unicode code; and the third row is decimal value of the EBCDIC code.

EBCDIC

<table>
<tr><th></th><th>_0</th><th>_1</th><th>_2</th><th>_3</th><th>_4</th><th>_5</th><th>_6</th><th>_7</th><th>_8</th><th>_9</th><th>_A</th><th>_B</th><th>_C</th><th>_D</th><th>_E</th><th>_F</th></tr>
<tr><th>0_</th><td>NUL
0000
0</td><td>SOH
0001
1</td><td>STX
0002
2</td><td>ETX
0003
3</td><td>SEL
4</td><td>HT
0009
5</td><td>RNL
6</td><td>DEL
007F
7</td><td>GE
8</td><td>SPS
9</td><td>RPT
10</td><td>VT
000B
11</td><td>FF
000C
12</td><td>CR
000D
13</td><td>SO
000E
14</td><td>SI
000F
15</td></tr>
<tr><th>1_</th><td>DLE
0010
16</td><td>DC1
0011
17</td><td>DC2
0012
18</td><td>DC3
0013
19</td><td>RES ENP
20</td><td>NL
0085
21</td><td>BS
0008
22</td><td>POC
23</td><td>CAN
0018
24</td><td>EM
0019
25</td><td>UBS
26</td><td>CU1
27</td><td>IFS
001C
28</td><td>IGS
001D
29</td><td>IRS
001E
30</td><td>IUS ITB
001F
31</td></tr>
<tr><th>2_</th><td>DS
32</td><td>SOS
33</td><td>FS
34</td><td>WUS
35</td><td>BYP INP
36</td><td>LF
000A
37</td><td>ETB
0017
38</td><td>ESC
001B
39</td><td>SA
40</td><td>SFE
41</td><td>SM SW
42</td><td>CSP
43</td><td>MFA
44</td><td>ENQ
0005
45</td><td>ACK
0006
46</td><td>BEL
0007
47</td></tr>
<tr><th>3_</th><td>48</td><td>49</td><td>SYN
0016
50</td><td>IR
51</td><td>PP
52</td><td>TRN
53</td><td>NBS
54</td><td>EOT
0004
55</td><td>SBS
56</td><td>IT
57</td><td>RFF
58</td><td>CU3
59</td><td>DC4
0014
60</td><td>NAK
0015
61</td><td>62</td><td>SUB
001A
63</td></tr>
<tr><th>4_</th><td>SP
0020
64</td><td>RSP
00A0
65</td><td>66</td><td>67</td><td>68</td><td>69</td><td>70</td><td>71</td><td>72</td><td>73</td><td>74</td><td>.
002E
75</td><td><
003C
76</td><td>(
0028
77</td><td>+
002B
78</td><td>|
007C
79</td></tr>
<tr><th>5_</th><td>&
0026
80</td><td>81</td><td>82</td><td>83</td><td>84</td><td>85</td><td>86</td><td>87</td><td>88</td><td>89</td><td>!
0021
90</td><td>$
0024
91</td><td>*
002A
92</td><td>)
0029
93</td><td>;
003B
94</td><td>¬
00AC
95</td></tr>
<tr><th>6_</th><td>-
002D
96</td><td>/
002F
97</td><td>98</td><td>99</td><td>100</td><td>101</td><td>102</td><td>103</td><td>104</td><td>105</td><td>¦
00A6
106</td><td>,
002C
107</td><td>%
0025
108</td><td>_
005F
109</td><td>>
003E
110</td><td>?
003F
111</td></tr>
<tr><th>7_</th><td>112</td><td>113</td><td>114</td><td>115</td><td>116</td><td>117</td><td>118</td><td>119</td><td>120</td><td>`
0060
121</td><td>:
003A
122</td><td>#
0023
123</td><td>@
0040
124</td><td>'
0027
125</td><td>=
003D
126</td><td>"
0022
127</td></tr>
<tr><th>8_</th><td>128</td><td>a
0061
129</td><td>b
0062
130</td><td>c
0063
131</td><td>d
0064
132</td><td>e
0065
133</td><td>f
0066
134</td><td>g
0067
135</td><td>h
0068
136</td><td>i
0069
137</td><td>138</td><td>139</td><td>140</td><td>141</td><td>142</td><td>±
00B1
143</td></tr>
<tr><th>9_</th><td>144</td><td>j
006A
145</td><td>k
006B
146</td><td>l
006C
147</td><td>m
006D
148</td><td>n
006E
149</td><td>o
006F
150</td><td>p
0070
151</td><td>q
0071
152</td><td>r
0072
153</td><td>154</td><td>155</td><td>156</td><td>157</td><td>158</td><td>159</td></tr>
<tr><th>A_</th><td>160</td><td>~
007E
161</td><td>s
0073
162</td><td>t
0074
163</td><td>u
0075
164</td><td>v
0076
165</td><td>w
0077
166</td><td>x
0078
167</td><td>y
0079
168</td><td>z
007A
169</td><td>170</td><td>171</td><td>172</td><td>173</td><td>174</td><td>175</td></tr>
<tr><th>B_</th><td>^
005E
176</td><td>177</td><td>178</td><td>179</td><td>180</td><td>181</td><td>182</td><td>183</td><td>184</td><td>185</td><td>[
005B
186</td><td>]
005D
187</td><td>188</td><td>189</td><td>190</td><td>191</td></tr>
<tr><th>C_</th><td>{
007B
192</td><td>A
0041
193</td><td>B
0042
194</td><td>C
0043
195</td><td>D
0044
196</td><td>E
0045
197</td><td>F
0046
198</td><td>G
0047
199</td><td>H
0048
200</td><td>I
0049
201</td><td>SHY
00AD
202</td><td>203</td><td>204</td><td>205</td><td>206</td><td>207</td></tr>
<tr><th>D_</th><td>}
007D
208</td><td>J
004A
209</td><td>K
004B
210</td><td>L
004C
211</td><td>M
004D
212</td><td>N
004E
213</td><td>O
004F
214</td><td>P
0050
215</td><td>Q
0051
216</td><td>R
0052
217</td><td>218</td><td>219</td><td>220</td><td>221</td><td>222</td><td>223</td></tr>
<tr><th>E_</th><td>\
005C
224</td><td>NSP
2007
225</td><td>S
0053
226</td><td>T
0054
227</td><td>U
0055
228</td><td>V
0056
229</td><td>W
0057
230</td><td>X
0058
231</td><td>Y
0059
232</td><td>Z
005A
233</td><td>234</td><td>235</td><td>236</td><td>237</td><td>238</td><td>239</td></tr>
<tr><th>F_</th><td>0
0030
240</td><td>1
0031
241</td><td>2
0032
242</td><td>3
0033
243</td><td>4
0034
244</td><td>5
0035
245</td><td>6
0036
246</td><td>7
0037
247</td><td>8
0038
248</td><td>9
0039
249</td><td>250</td><td>251</td><td>252</td><td>253</td><td>254</td><td>EO
255</td></tr>
<tr><th></th><th>_0</th><th>_1</th><th>_2</th><th>_3</th><th>_4</th><th>_5</th><th>_6</th><th>_7</th><th>_8</th><th>_9</th><th>_A</th><th>_B</th><th>_C</th><th>_D</th><th>_E</th><th>_F</th></tr>
</table>

DKOI

EBCDIC-based DKOI character sets:[6]

- DKOI CS2 (defined in CSN 36 9103)[7]
- DKOI K1 (defined in CSN 36 9103, ST SEV 358-88)[7]
- DKOI K2 (defined in CSN 36 9103, ST SEV 358-88)[7]
- DKOI L2 (defined in CSN 36 9103)[7]

Criticism and humor

Open-source software advocate and hacker Eric S. Raymond writes in his *Jargon File* that EBCDIC was almost universally loathed by early hackers and programmers. The Jargon File 4.4.7 gives the following definition:[8]

> EBCDIC: /eb´s@·dik/, /eb´see`dik/, /eb´k@·dik/, n. [abbreviation, Extended Binary Coded Decimal Interchange Code] An alleged character set used on IBM dinosaurs. It exists in at least six mutually incompatible versions, all featuring such delights as non-contiguous letter sequences and the absence of several ASCII punctuation characters fairly important for modern computer languages (exactly which characters are absent varies according to which version of EBCDIC you're looking at). IBM adapted EBCDIC from punched card code in the early 1960s and promulgated it as a customer-control tactic (see *connector conspiracy*), spurning the already established ASCII standard. Today, IBM claims to be an open-systems company, but IBM's own description of the EBCDIC variants and how to convert between them is still internally classified top-secret, burn-before-reading. Hackers blanch at the very *name* of EBCDIC and consider it a manifestation of purest evil.
>
> — The Jargon file 4.4.7

EBCDIC design was also the source of many jokes. One such joke went:

> *Professor: "So the American government went to IBM to come up with an encryption standard, and they came up with—"*
> *Student: "EBCDIC!"*

References to the EBCDIC character set are made in the classic Infocom adventure game series *Zork*. In the "Machine Room" in *Zork II*, EBCDIC is used to imply an incomprehensible language:

> This is a large room full of assorted heavy machinery, whirring noisily. The room smells of burned resistors. Along one wall are three buttons which are, respectively, round, triangular, and square. Naturally, above these buttons are instructions written in EBCDIC...

A similar description can be found in the "Maintenance Room" in Zork:

> This is what appears to have been the maintenance room for Flood Control Dam #3, judging by the assortment of tool chests around the room. Apparently, this room has been ransacked recently, for most of the valuable equipment is gone. On the wall in front of you is a group of buttons, which are labelled in EBCDIC. However, they are of different colors: Blue, Yellow, Brown, and Red.

See also

- List of EBCDIC code pages with Latin-1 character set
- Code page 037 (English, Portuguese)
- Code page 285 (Ireland, United Kingdom)
- UTF-EBCDIC

References

1. Mackenzie, Charles E. (1980). *Coded Character Sets, History and Development* (https://books.google.com/books?id=6-tQAAAAMAAJ). *The Systems Programming Series* (1 ed.). Addison-Wesley Publishing Company, Inc. ISBN 0-201-14460-3. LCCN 77-90165 (https://lccn.loc.gov/77-90165). ISBN 978-0-201-14460-4. Retrieved 2016-05-22. [1] (https://web.archive.org/web/20160526172151/https://textfiles.meulie.net/bitsaved/Books/Mackenzie_CodedCharSets.pdf)
2. Bemer, Bob. "EBCDIC and the P-Bit" (http://www.bobbemer.com/P-BIT.HTM). Retrieved July 2, 2013.
3. They had 4 staff on the final 21-member ASA X3.2 sub-committee (http://worldpowersystems.com/projects/codes/X3.4-1963/page4.JPG)
4. "*...but their printers and punches were not ready to handle ASCII, and IBM just HAD to announce.*" (http://www.trailing-edge.com/~bobbemer/P-BIT.HTM), EBCDIC and the P-Bit, Bob Bemer
5. IBMnt (2008). "IBM confirms the use of EBCDIC in their mainframes as a default practice" (https://archive.is/20130103091717/http://publib.boulder.ibm.com/infocenter/zos/v1r9/index.jsp?topic=/com.ibm.zos.r9.adms700/adms7a05158.htm). Archived from the original (http://publib.boulder.ibm.com/infocenter/zos/v1r9/index.jsp?topic=/com.ibm.zos.r9.adms700/adms7a05158.htm) on 2013-01-03. Retrieved 2008-06-16.
6. Kornai, Andras; Birnbaum, David J.; da Cruz, Frank; Davis, Bur; Fowler, George; Paine, Richard B.; Paperno, Slava; Simonsen, Keld J.; Thobe, Glenn E.; Vulis, Dimitri; van Wingen, Johan W. (1993-03-13). "CYRILLIC ENCODING FAQ Version 1.3" (ftp://kermit.columbia.edu/kermit/charsets/cyrillic-summary.txt). 1.3. Retrieved 2017-02-18.
7. Petrlik, Lukas (1996-06-19). "The Czech and Slovak Character Encoding Mess Explained" (http://ftp.fi.muni.cz/pub/localization/charsets/cs-encodings-faq). *cs-encodings-faq*. 1.10. Archived (https://web.archive.org/web/20160621092901/http://ftp.fi.muni.cz/pub/localization/charsets/cs-encodings-faq) from the original on 2016-06-21. Retrieved 2016-06-21.
8. "EBCDIC" (http://catb.org/jargon/html/E/EBCDIC.html). *Jargon File*.

Further reading

- EBCDIC and the P-Bit (The Biggest Computer Goof Ever) (http://www.bobbemer.com/P-BIT.HTM)

External links

- Character Data Representation Architecture (CDRA) from IBM (http://www.ibm.com/software/globalization/g11n-res.jsp) Contains IBM's official information on code pages and character sets.
  - Code page 37 (ftp://ftp.software.ibm.com/software/globalization/gcoc/attachments/CP00037.pdf)
  - Code page 1047 (ftp://ftp.software.ibm.com/software/globalization/gcoc/attachments/CP01047.pdf)
- F.0 Appendix F. Code Pages (http://publib.boulder.ibm.com/cgi-bin/bookmgr/BOOKS/QB3AQ501/F.0) from AS/400 International Application Development V4R2 (http://publib.boulder.ibm.com/cgi-bin/bookmgr/BOOKS/QB3AQ501/CCONTENTS)
- ICU Converter Explorer (http://demo.icu-project.org/icu-bin/convexp) Contains more information about EBCDIC derived from IBM's CDRA, including DBCS EBCDIC (Double Byte Character Set EBCDIC)
- ICU Charset Mapping Tables (http://www.icu-project.org/charts/charset/) Contains computer readable Unicode mapping tables for EBCDIC and many other character sets
- EBCDIC (https://web.archive.org/web/20160303222819/http://www.hansenb.pdx.edu/DMKB/dict/tutorials/ebcdic.php) character list, including decimal and hex values, symbolic name, and character/function
- iconv.com (http://www.iconv.com/asciiebcdic.htm) Online tool to convert from ASCII to/from EBCDIC
- EBCDIC-code pages with Latin-1-charset (JavaScript) (http://wzwz.de/prog/ebcdic/cc_en.htm)
- All EBCDIC code pages and 3270 graphics escape codes (http://mainframe.wiki/ebcdicTbl.php)

Retrieved from "https://en.wikipedia.org/w/index.php?title=EBCDIC&oldid=786967721"

Categories: IBM mainframe operating systems | EBCDIC code pages

- This page was last edited on 22 June 2017, at 17:07.
- Text is available under the Creative Commons Attribution-ShareAlike License; additional terms may apply. By using this site, you agree to the Terms of Use and Privacy Policy. Wikipedia® is a registered trademark of the Wikimedia Foundation, Inc., a non-profit organization.