

Teletypewriter Communication Codes

Gil Smith
gil@vauxelectronics.com
2001

[\(Document Notes\)](#)

Abstract

Preliminary -- 5/01 gil smith Corrections or comments to gil@vauxelectronics.com

This information is pulled from a variety of sources, such as various emails of the greenkeys group. For more discussion of teletypewriter code development, see:

<http://www.nadcomm.com/fiveunit/fiveunits.htm>

<http://fido.wps.com/texts/codes/index.html>

<http://www.science.uva.nl/faculteit/museum/DWcodes.html>

FIVE-UNIT CODES: USTTY and ITA2 (aka BAUDOT)

There were a few variations in character codes for five-level teletypewriter machines. The two most-common character codes were ITA2 and USTTY (a variation of ITA2).

The USTTY and ITA2 5-level teletypewriter codes are commonly referred to as "Baudot" codes. While this is technically incorrect, these popular 5-level codes evolved from the work of Jean Maurice Emile Baudot of France - it seems fitting to accept the defacto reference to "Baudot" as implying USTTY or ITA2 codes, since they were the 5-level codes that saw practical use in teletypewriter systems. However, the true Baudot code dates to around 1874, when Baudot designed the "Baudot Multiplex System," a printing telegraph. The system used a 5-level code generated by a device with five keys, operated with two left-hand fingers, and three right-hand fingers -- this required great skill on the part of the operator who entered the code directly. However, it was still a major improvement in communications -- prior to Baudot's design, communication was carried out using Morse code with a telegraph key. The 5-level "Baudot" code was actually designed by Johann Gauss and Wilhelm Weber. Used primarily in France, the Baudot Multiplex System, also made inroads in Britain. The original Baudot code defined the familiar structure of a 5-level code set, using LTRS and FIGS case shifting, and became known as the International Telegraph Alphabet 1 (ITA1). Another recognition of Baudot's contribution to data communications is the term "baud," which refers to bits-per-second speed of serial data.

Around 1901, Donald Murray of New Zealand developed an automatic telegraphy system, using a typewriter-like keyboard mechanism, and a variation on the original Baudot 5-level code. While Baudot's code was designed with

finger-actuation in mind, Murray's code was designed for mechanization, to minimize machine wear for frequently-occurring characters. The Murray system employed a keyboard perforator, which allowed an operator to manually punch a paper tape, and a tape transmitter for sending the message from the punched tape. At the receiving end of the line, a printing mechanism would print on a paper tape, and/or a reperforator could be used to make a perforated copy of the message. Early British Creed machines were used with the Murray system.

Around 1930, the CCITT (International Telegraph and Telephone Consultative Committee) introduced the International Telegraph Alphabet 2 (ITA2). The United States standardized on a variation of ITA2 called the American Teletypewriter code or USTTY. ITA2 and the USTTY variant became the basis for 5-level teletypewriter codes until 7-level ASCII code debuted (in an upper-case-only form) in 1963, and finally matured in 1967 to the form still used today.

Common to all of these 5-level codes is the shifting of keys using a FIGS or LTRS code. The same 5-level (5-bit) code is used to represent a lowercase symbol (LTRS case) or an uppercase symbol (FIGS case). When the LTRS or FIGS character is transmitted, it defines whether subsequent characters are to be interpreted as lowercase or uppercase. The receiving machine must remember the case until it is next changed.

The ITA2 code was an international standard, used in applications such as Telex service. The shifted-D character could be interpreted as WRU (Who-Are-You). The shifted-F, G, and H characters were technically undefined, but had commonly-used symbols.

USTTY was a variation of ITA2, and was the prevalent code used in the United States. The letters D, F, G, H, J, S, V, and Z had different FIGS symbols than ITA2, and, for some reason, BELL and ' were opposite of ITA2. There were also variations for specific applications -- that is, machines that had primarily a USTTY or ITA2 code set, with just a few changed keys. J and H were said to have the most commonly-changed FIGS characters. USTTY was also the base code used for conversion of five-level to seven-level codes for later dial TWX service.

The shifted-H symbol (#/STOP/blank) was sometimes used in USTTY machines as a "motor off" control code; the next character would turn the motor back on. Since some characters are missed until the motor is up to speed, a BLANK (NULL) character would be sent to start the motor, with perhaps a dozen additional BLANKs sent as filler to allow the motor speed to stabilize before reception of the message.

Some teletypewriter machines "unshift-on-space," which is to say they revert to LTRS state after a space character is received. Some machines unshift at the the end of a line; this machine-specific behavior is not presumed by the code. Transmitted messages typically used a CR LF LTRS LTRS sequence at the end of a line (eg: at 72 chars) -- the CR LF positioned printing at the start of the next line, and the LTRS LTRS not only provided an explicit unshift, but it allowed the carriage time to fully return to the home position.

LTRS	USTTY FIGS	ITA2 FIGS	Fractions FIGS	other FIGS variations
----	----	-----	----	-----
A	-	-	-	
B	?	?	5/8	
C	:	:	1/8	WRU (degree?)
D	\$	# (WRU)	\$	
E	3	3	3	
F	!	@ (undef)	1/4	# % (blank)
G	&	* (undef)	&	@
H	#	\$ (undef)	#	STOP (blank) (British-pound)
I	8	8	8	
J	'	BELL	'	,
K	((1/2	
L))	3/4	
M	.	.	?	
N	,	,	7/8	\
O	9	9	9	
P	0	0	0	
Q	1	1	1	
R	4	4	4	
S	BELL	'	BELL	(blank)
T	5	5	5	
U	7	7	7	
V	;	=	3/8	
W	2	2	2	
X	/	/	/	
Y	6	6	6	
Z	"	+	"	

Translation of USTTY code to ASCII is shown below. A different ASCII character is needed depending on the current case (LTRS or FIGS):

USTTY				ASCII translation			
hex	binary	LTRS	FIGS	LTRS		FIGS	
0x00	00000	BLANK	BLANK	0x00	NUL	0x00	NUL
0x01	00001	E	3	0x45	E	0x33	3
0x02	00010	LF	LF	0x0A	LF	0x0A	LF
0x03	00011	A	-	0x41	A	0x2D	-
0x04	00100	SP	SP	0x20	SP	0x20	SP
0x05	00101	S	BELL	0x53	S	0x07	BEL
0x06	00110	I	8	0x49	I	0x38	8
0x07	00111	U	7	0x55	U	0x37	7
0x08	01000	CR	CR	0x0D	CR	0x0D	CR
0x09	01001	D	\$	0x44	D	0x24	\$
0x0A	01010	R	4	0x52	R	0x34	4
0x0B	01011	J	'	0x4A	J	0x27	'
0x0C	01100	N	,	0x4E	N	0x2C	,
0x0D	01101	F	!	0x46	F	0x21	!
0x0E	01110	C	:	0x43	C	0x3A	:
0x0F	01111	K	(0x4B	K	0x28	(
0x10	10000	T	5	0x54	T	0x35	5
0x11	10001	Z	"	0x5A	Z	0x22	"
0x12	10010	L)	0x4C	L	0x29)
0x13	10011	W	2	0x57	W	0x32	2
0x14	10100	H	#	0x48	H	0x23	#
0x15	10101	Y	6	0x59	Y	0x36	6
0x16	10110	P	0	0x50	P	0x30	0
0x17	10111	Q	1	0x51	Q	0x31	1
0x18	11000	O	9	0x4F	O	0x39	9
0x19	11001	B	?	0x42	B	0x3F	?
0x1A	11010	G	&	0x47	G	0x26	&
0x1B	11011	FIGS	FIGS	0x0E	SO	0x0E	SO
0x1C	11100	M	.	0x4D	M	0x2E	.
0x1D	11101	X	/	0x58	X	0x2F	/
0x1E	11110	V	;	0x56	V	0x3B	;
0x1F	11111	LTRS	LTRS	0x0F	SI	0x0F	SI

FIVE-UNIT WEATHER CODES

Starting in the 1940s, weather reports were sent hourly on landline circuits at 60 WPM. The weather, or "WX" symbols were used on model 15 and 19 machines with a special WX type basket. The WX code had special FIGS symbols representing cloud cover and wind direction.

The WX service did not set their machines to unshift-on-space, which would have required a FIGS before each group of numerals (some weather codes were in groups of five numerals). The end-of-line sequence was typically LTRS CR LF LTRS, or, if the next line began with numerals, LTRS CR LF FIGS.

From a model 15 manual, the weather symbol set is defined as:

Lowercase (ltrs)	Uppercase Weather symbols (figs)	Definition
A	(up arrow)	North wind
B	(circle w/ vertical & horiz line)	Overcast -- More than 9/10 covered
C	(circle)	Sky clear -- less than 1/10 cloud cover
D	(upper-right arrow)	Northeast wind
F	(right arrow)	East wind
G	(lower-right arrow)	Southeast wind
H	(down arrow)	South wind
J	(lower-left arrow)	Southwest wind
K	(left arrow)	West wind
L	(upper-left arrow)	Northwest wind
N	(circle w/ two vertical lines)	Broken clouds -- More than 1/2 covered
V	(circle w/ one vertical line)	Scattered clouds -- less than 1/2 covered
Z	+	

The remainder of the upper case symbols are the same as USTTY.

SIX-UNIT TELETYPESETTING (TTS) CODE

The six-level TTS (Teletypesetting) code was introduced by Walter Morey, to address the need for full upper and lowercase letters for newspaper and other publishing applications. The Teletype Model 20 is a six-level teletypewriter based on a Model 15 machine.

(need table)

SEVEN/EIGHT-UNIT CODES: ASCII-63 to ASCII-67 (ITA-5)

ASCII-63, introduced in 1963, was a 7-level (7-bit) code which did away with the LTRS/FIGS case shifting of the original 5-level codes. ASCII-63 only had uppercase letters -- upper and lowercase letters were both defined in the ASCII-67 (1967) version, which is still in use today. I understand that ASCII-67 was accepted as an international version of ASCII, and called ITA-5 (International Telegraph Alphabet No. 5). The ASCII-67 code, and a translation to 5-level USTTY, are shown below.

ASCII is technically a 7-bit code, but it is often referred to as an 8-bit code, since it is usually transmitted as 8-bits, with the 8th bit either space, mark, or even-parity. "8-level" ASCII teletype machines usually sent even parity ascii from the keyboard, but some (especially early units) sent bit-8 always mark. Model 33 and 35 machines were uppercase-only, and, except for DEL, characters from col-4 printed as col-3.

ASCII-67

-----	col-1	col-2	col-3	col-4	-----
0x00	^@ NUL	0x20 SP	0x40 @	0x60 `	
0x01	^A SOH	0x21 !	0x41 A	0x61 a	
0x02	^B STX	0x22 "	0x42 B	0x62 b	
0x03	^C ETX	0x23 #	0x43 C	0x63 c	
0x04	^D EOT	0x24 \$	0x44 D	0x64 d	
0x05	^E ENQ	0x25 %	0x45 E	0x65 e	
0x06	^F ACK	0x26 &	0x46 F	0x66 f	
0x07	^G BEL	0x27 '	0x47 G	0x67 g	
0x08	^H BS	0x28 (0x48 H	0x68 h	
0x09	^I HT	0x29)	0x49 I	0x69 i	
0x0A	^J LF	0x2A *	0x4A J	0x6A j	
0x0B	^K VT	0x2B +	0x4B K	0x6B k	
0x0C	^L FF	0x2C ,	0x4C L	0x6C l	
0x0D	^M CR	0x2D -	0x4D M	0x6D m	
0x0E	^N SO	0x2E .	0x4E N	0x6E n	
0x0F	^O SI	0x2F /	0x4F O	0x6F o	
0x10	^P DLE	0x30 0	0x50 P	0x70 p	
0x11	^Q DC1	0x31 1	0x51 Q	0x71 q	
0x12	^R DC2	0x32 2	0x52 R	0x72 r	
0x13	^S DC3	0x33 3	0x53 S	0x73 s	
0x14	^T DC4	0x34 4	0x54 T	0x74 t	
0x15	^U NAK	0x35 5	0x55 U	0x75 u	
0x16	^V SYN	0x36 6	0x56 V	0x76 v	
0x17	^W ETB	0x37 7	0x57 W	0x77 w	
0x18	^X CAN	0x38 8	0x58 X	0x78 x	
0x19	^Y EM	0x39 9	0x59 Y	0x79 y	
0x1A	^Z SUB	0x3A :	0x5A Z	0x7A z	
0x1B	^[ESC	0x3B ;	0x5B [0x7B {	
0x1C	^\ FS	0x3C <	0x5E ^	0x7E ~	
0x1D	^] GS	0x3D =	0x5D]	0x7D }	
0x1E	^^ RS	0x3E >	0x5E ^	0x7E ~	
0x1F	^_ US	0x3F ?	0x5F _	0x7F DEL	

^x is control key and character key together

ASCII-67				USTTY translation							
col-1	col-2	col-3	col-4	col-1		col-2		col-3		col-4	
^@ NUL	SP	@	`	N 0x00 BLANK	N 0x04 SP	--	--	--	--	--	--
^A SOH	!	A	a	--	F 0x0D !	L 0x03 A					
^B STX	"	B	b	--	F 0x11 "	L 0x19 B					
^C ETX	#	C	c	--	F 0x14 #	L 0x0E C					
^D EOT	\$	D	d	--	F 0x09 \$	L 0x09 D					
^E ENQ	%	E	e	--	--	L 0x01 E					
^F ACK	&	F	f	--	F 0x1A &	L 0x0D F					
^G BEL	'	G	g	F 0x05 BELL	F 0x0B '	L 0x1A G					
^H BS	(H	h	--	F 0x0F (L 0x14 H					
^I HT)	I	i	--	F 0x12)	L 0x06 I					
^J LF	*	J	j	N 0x02 LF	--	L 0x0B J					
^K VT	+	K	k	--	--	L 0x0F K					
^L FF	,	L	l	--	F 0x0C ,	L 0x12 L					
^M CR	-	M	m	N 0x08 CR	F 0x03 -	L 0x1C M					
^N SO	.	N	n	N 0x1B FIGS	F 0x1C .	L 0x0C N					
^O SI	/	O	o	N 0x1F LTRS	F 0x1D /	L 0x18 O					
^P DLE	0	P	p	--	F 0x16 0	L 0x16 P					
^Q DC1	1	Q	q	--	F 0x17 1	L 0x17 Q					
^R DC2	2	R	r	--	F 0x13 2	L 0x0A R					
^S DC3	3	S	s	--	F 0x01 3	L 0x05 S					
^T DC4	4	T	t	--	F 0x0A 4	L 0x10 T					
^U NAK	5	U	u	--	F 0x10 5	L 0x07 U					
^V SYN	6	V	v	--	F 0x15 6	L 0x1E V					
^W ETB	7	W	w	--	F 0x07 7	L 0x13 W					
^X CAN	8	X	x	--	F 0x06 8	L 0x1D X					
^Y EM	9	Y	y	--	F 0x18 9	L 0x15 Y					
^Z SUB	:	Z	z	--	F 0x0E :	L 0x11 Z					
^[ESC	;	[{	--	F 0x1E ;	--	--	--	--	--	--
^\ FS	<	^	~	--	--	--	--	--	--	--	--
^] GS	=]	}	--	--	--	--	--	--	--	--
^^ RS	>	^	~	--	--	--	--	--	--	--	--
^_ US	?	_	DEL	--	F 0x19 ?	--	--	--	--	--	--

One possible USTTY translation of ASCII-67 (shown above):

- change case as needed by sending LTRS or FIGS: L=LTRS, F=FIGS, N=No-Change
 - ascii lowercase chars converted to uppercase
 - map control chars NUL, BEL, LF and CR
 - append LTRS LTRS after each CR for mechanical TTY machines to return fully
 - use SO to send FIGS, SI to send LTRS (not needed unless testing shifts)
 - no translation for other control chars, or () <> [] { } ? % * + = @ ` _ \ ^ | ~
- (send BLANK or nothing)

ASCII-63 was mostly identical to the current ASCII-67 version. The definitions of the control characters (col-1 above) varied between the two versions, as defined below. Also, in ASCII-63, the upper 32 positions (col-4) were undefined, except for three: RUB (0x7F), ACK (0x7C), and ESC (0x7E). There are inconsistent references to an ALT-MODE char (0x7D) in ASCII-63. In the 1967 version, RUB became DEL and stayed in the same position, but ACK and ESC moved into the control character area (col-1). In ASCII-67, ^ replaced the up-arrow symbol, and _ replaced the left-arrow.

ASCII-63 and ASCII-67 are the common variants of ASCII, but there appear to have been some transitional versions as well: in the Teletype Model 33 manual, there are references to a 1965 version of ASCII, that had SS in place of SUB (0x1A), \ for @ (0x40), ~ for \ (0x5C), an odd character in place of | (0x7C), and | for ~ (0x7E). A Teletype code card for M33 and M35 machines indicates a 1966 version of ASCII, though the printable characters shown on the card were identical in all versions.

The DC1 and DC3 control characters have been used as aux-device start/stop commands (eg: auto-tape-reader option in Teletype M33asr). DC2/DC4 have also been used as auto-tape-punch on/off commands in the M33asr.

DC1 and DC3 are also referred to as XON and XOFF respectively. XON and XOFF are widely used today as software-handshaking flow-control characters for throttling data transfer into a serial buffer.

ASCII-63 Control Characters

NULL = Null char
SOM = Start of Message
EOA = End of Address
EOM = End of Message
EOT = End of Transmission
WRU = Who Are You
RU = Are You
BELL = Signal Bell
FE0 = Format Effector 0
HT/SK = Horizontal Tab
LF = Line Feed
VTAB = Vertical Tab
FF = Form Feed
CR = Carriage Return
SO = Shift Out
SI = Shift In
DC0 = Device Control 0
DC1 = Device Control 1
DC2 = Device Control 2
DC3 = Device Control 3
DC4 = Device Control 4
ERR = Error
SYNC = Synchronous
LEM = Logical End of Medium
S0 = Special Application 0
S1 = Special Application 1
S2 = Special Application 2
S3 = Special Application 3
S4 = Special Application 4
S5 = Special Application 5
S6 = Special Application 6
S7 = Special Application 7
RUB = Rubout
ACK = Acknowledge
ALT-MODE = Alternate Mode
ESC = Escape

ASCII-67 Control Characters

NUL = Null char
SOH = Start of Heading
STX = Start of Text
ETX = End of Text
EOT = End of Transmission
ENQ = Enquiry
ACK = Acknowledge
BEL = Signal Bell
BS = Back Space
HT = Horizontal Tab
LF = Line Feed (aka NL = New Line)
VT = Vertical Tab
FF = Form Feed
CR = Carriage Return
SO = Shift Out (special)
SI = Shift In (normal)
DLE = Data Link Escape
DC1 = Device Control 1 (aka XON) (M33 tape-reader-on)
DC2 = Device Control 2 (M33 tape-punch-on)
DC3 = Device Control 3 (aka XOFF) (M33 tape-reader-off)
DC4 = Device Control 4 (M33 tape-punch-off)
NAK = Negative Acknowledge
SYN = Synchronous Idle
ETB = End of Transmission Block
CAN = Cancel
EM = End of Medium (Media?)
SUB = Substitute
ESC = Escape
FS = Field Separator
GS = Group Separator
RS = Record Separator
US = Unit Separator
DEL = Delete
(position 0x7C became |)
(position 0x7D became })
(position 0x7E became ~)

Document Notes

Created by [Gil Smith](#), July 2001. The original file is [smith--teletype-codes.txt](#).