

Introduction

The K20 is a single board Morse keyer with a built in IBM AT keyboard interface. It can be used as a keyer only or as a CW keyboard. This document will describe how to interface to the asynchronous serial port of the keyer and drive it from a personal computer or other host. The interface is based on a simple set of ASCII commands. The K20 provides a single handshake line that tells when it is ready to accept data.

Command Set

The K20 only accepts asynchronous serial data at 1200 baud. The format must be 8 data bits, 1 stop bit, with no parity. Note that the K20 must be modified to accept RS232 data, see the K20 hardware modification section for details. Normal alphanumeric characters will be sent as Morse letters, commands are control characters, which are listed below. Special characters such as prosigns are listed in a separate table. When formatting a command no spaces should be added between commands and/or arguments, also numeric values must always be sent as two decimal digits with a leading zero if the value is less than 10. The only exceptions to this are the MODE register command in which the value is sent as a two digit hexadecimal value.

Immediate Commands

^A	<nn>	01h nn	Set sidetone frequency to nn where nn = 4 – 0AH (see table below)
^B	<nn>	02h nn	Set keying speed to nn WPM where nn = 05 - 99
^C	<nn>	03h nn	Set Dit/Dah Ratio to nn/10 nn = 05 - 32
^D	<nn>	04h nn	Set PTT Lead-In delay to nn*10 milliseconds nn = 00 - 99
^E	<nn>	05h nn	Set PTT Tail delay to nn*10 milliseconds nn = 00 - 99
^F		06h	Pause transmit On/Off
^G	<nn>	07h nn	Flow Control Count 01 – 30 (change not recommended)
^H		08h	Backspace buffer input pointer
^I		09h	Say Status
^J		0Ah	Clear circular buffer and abort current process
^K		0Bh	Key Down on/off
^L	<nn>	0Ch nn	Set HSCW speed to nn*100 LPM
^M	<nn>	0Dh nn	Set Extraspace to nn/10 nn = 05 - 32
^N	<hh>	0Eh h	Set Mode Register to hex value hh (see Mode Reg. Description)
^O	<list>	0Fh <....>	Load Keyer Default List (see Defaults Description)
^P	<nn>	10h nn	First bit extension to nn milliseconds nn = 0-99
^Q	<nn>	11h nn	Set Weighting Adjustment in milliseconds nn = 0 – 99, 50 = 00
^R	<nn>	12h nn	Adjust swing sensitivity nn = 0 - 99
^S	<t1>	13h t1	Start Practice t=e/r l=1,2,3,4 (see K20 User Guide)

Buffered Message Commands

^V		16h	PTT on/off
^W		17h	Key on/off
^X		18h nn	Wait for nn seconds nn=0-99
^Y	<xy>	19h xy	Merge letters x,y
^Z	<nn>	1Ah nn	Change WPM speed to nn WPM nn=5-99
^\		1Ch	Analog diagnostic
^J	<nn>	1Dh nn	Set HSCW speed to nn
^^	<nn>	1Eh nn	Set loop timer to nn nn=0-99

Special Characters

-	BT	=	BT]	QSL
/	DN	>	SK		DEL 8 dits
:	KN	@	AS		
<	AR	[QRZ		

Sidetone Values

Value	Frequency
01h	4000 hz
02h	2000 hz
03h	1333 hz
04h	1000 hz
05h	800 hz
06h	666 hz
07h	571 hz
08h	500
09h	444 hz
0Ah	400 hz

Mode Register Description

There is an eight-bit byte in the Keyer PIC that contains bit flags that control the operation modes of the keyer. Each bit's state controls a particular mode, following is a table showing the bit functions:

Bit	Function	Power Up State
7 (MSB)	Iambic A when = 1 Iambic B = 0	0
6	Farnsworth on when = 1	0
5	Bug Mode on when = 1	0
4	Softsidetone on when = 1	0
3	Swap paddles when = 1	0
2	Disable sidetone when = 1	0
1	Autospace On when = 1	0
0 (LSB)	Mute Transmit when = 1	0

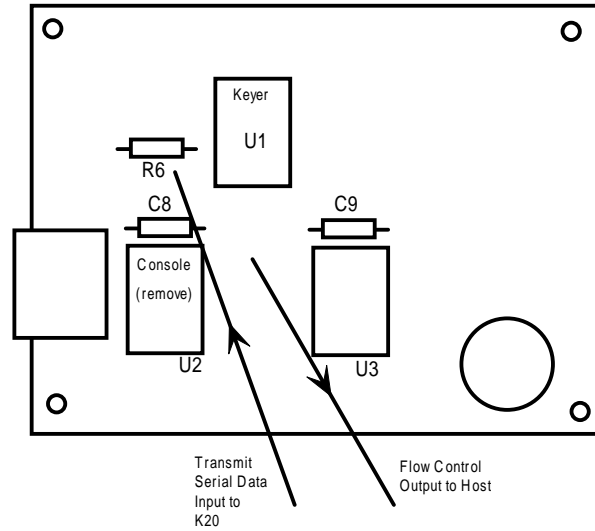
Defaults Description

All of the Keyers internal state registers can be initialized in one block transfer. This feature is useful for fast initialization or for changing the whole configuration of the keyer quickly. The following illustrates how the command is issued and the order of control words loaded.

Byte #	Value	Power On Default
0	0Fh (command)	N/A
1	Mode Register Value	00
2	WPM Speed	15
3	Sidetone Value	05
4	Dit/Dah Ratio	10
5	Lead In time	00
6	Tail Time	00
7	1 st Bit Extension	00
8	Weighting	50
9	Extra Space	10
10	Sample Adjust	10

Hardware Modification

The K20 was originally designed to have the Console PIC provide the serial input to the Keyer PIC. To operate the K20 with an external serial data source the Console PIC must be removed from its socket and wires are soldered to two unused hole positions on the K20 PCB. One wire is for serial data to the K20 the other is a flow control (ready) output. These signals are connected to a host which provides all control to the K20. The paddle inputs to the Keyer PIC are unaffected and work normally. The host must monitor the flow control line, if it is high the host must wait till the K20 drops it before sending any additional bytes.



Connections to K20 For External Serial Data Control

If the host provides true RS232 formatted signals, a level converter MUST be installed between the host and the K20 or damage will result. Standard RS232 levels transit between -12 and $+12$ volts DC while the K20 expects 0 to 5 volts. There are several RS232 level converters available from Maxim that supply all the interface details in one package. Typical ones to look at are the MAX232 and the MAX203. Following is a diagram of a typical application using the MAX 232

