

## Junior TTY Monitor



A minimized monitor with TTY support and a Intel-HEX loader. Connection with 2 free port pins PB0 (TX), PB5 (RX). Ports and pins can easily be changed.

The original monitor has no assembler and none of the old legacy code anymore. With today's compilers it's easy to generate Intel-Hex output.

Terminal setup: 2400 baud, 8 data bits, no parity, 1 stop bit (8 N 1). Set character delay to 1 ms and line delay to 10 ms.

Start the upload with a single click by pressing the PC button. In the terminal window you will see:

- **TTY>**
- **receiving...**

Now paste the hex code into the terminal — the conversion will start. On success you will see (for example):

- **Ok \$0087 bytes received!**

If loading fails, the message will be:

- **Err!**

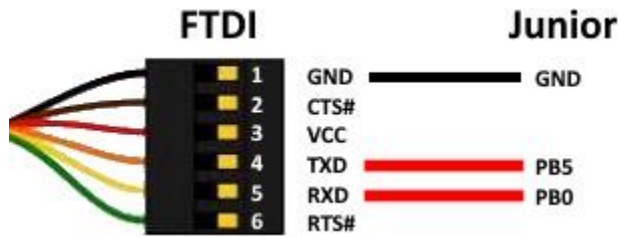
The hex code must not contain lowercase characters, nor any delimiters such as spaces, tabs, or control characters. After loading, the monitor's start address is set to the start address contained in the hex record. Simply press the GO key to start the program...

Whenever you press the RESET key, you return to the usual Junior input address \$0200, already in data mode and ready for input.

The following support routines are available.

<i>NAME</i>	<i>ADDRESS</i>	
<b>TTY_INIT</b>	\$FD3F	Call this routine before using other TTY routines
<b>TTY_PUTC</b>	\$FDA7	Display the character in A
<b>TTY_PUTS</b>	\$FDD8	Display a null terminated string pointed by by TTY_PSTR_LO = \$E3 and TTY_PSTR_HI = \$E4
<b>TTY_PUTDEC</b>	\$FDE9	Display the number in A in decimal format
<b>TTY_PUTHEX</b>	\$FE27	Display a hexadecimal number in A
<b>TTY_GETC</b>	\$FE59	Wait for a terminal input and return the key in A
<b>TTY_GETHEX</b>	\$FEA1	Wait for 2 terminal inputs and return it in A
<b>TTY_CRLF</b>	\$FE4E	Output \$0D and \$0A (CR/LF) to the terminal

Here is an example for a FTDI USB serial converter. Other converters must be wired accordingly.



The source code was written for the 64tass compiler.