PIC16F15344 Bootloader

Communication: terminator= <lf>, 2000000,N,8,1

Pin usage:

rb5=rtx bidirectional 2 Mbit (analog disabled)

Config:

```
; This encoder uses the 32 Mhz internal clock PLL combination
 __CONFIG _CONFIG1, _RSTOSC_HFINTPLL & _CLKOUTEN_OFF & _CSWEN_OFF & _FCMEN_OFF &
_FEXTOSC_OFF
 __CONFIG _CONFIG2, _MCLRE_ON & _PWRTE_ON & _LPBOREN_OFF & _BOREN_ON & _BORV_HI &
_ZCD_OFF & _PPS1WAY_OFF & _STVREN_ON
 __CONFIG _CONFIG3, _WDTE_OFF
 __CONFIG _CONFIG4, _BBEN_OFF & _SAFEN_OFF & _WRTAPP_OFF & _WRTB_OFF & _WRTSAF_OFF
& _LVP_OFF
 __CONFIG _CONFIG5, _CP_OFF
 #include "p16f15344.inc"
```

Commands:

bhhll: blank (erase) a 32 word line of memory starting hhll
  eg. b0840 : erase 32 words at 0x0840

i: reply 1 byte devcode, 1 byte rev. (0x1301)

jbbmmnn: read locations bb=bank(8..1), mm=offset 7 bits, nn=# bytes to read
        nn 0-ff where 0 represents 256.
  eg. j002010: read 16 bytes bank 0 offset 20h
      NOTE: program memory can be read from upper bank.

Kcccc... : echo all characters up to <lf> as a test of the serial port.
  eg. Kecho this line

mnnxxx...: write 16 bytes (xxx...) into bank 2 buffer starting at offset nn
        nn= 20,30,40 or 50. This is for filling the buffer to write to PGM memory.
  eg. m2001020304050607080 90a0b0c0d0e0f

REA: unlock

Rhhll: read a 32 word line of program memory into bank 2 64 byte buffer

Q: do soft reset

uhhll: program a line (32 words) of NV memory offset hhll

Notes Overleaf==>

NOTES:

Locations E20 (1E20 for '345) to end of memory are reserved for the bootloader.
Locations 0,1 are reserved and the host program must copy code from 0,1 to E1E,E1F (+1000 for '345).
It is assumed the 0,1 location contains: [movlb,goto] instructions pointing to init code.
The bootloader does not use interrupts.
If the rx line is high the bootloader will exit to the appvector. If the rx line is low the bootloader will loop until rx goes high then enter the bootloader monitor.

The host program is responsible for verifying the programmed data, the data does not include check information. However the bootloader will reject strings with the wrong length or unknown commands.
The bootloader replies ‘*’ when the command is accepted, ‘?’ when it is not.