User instructions for the "Atmega_in_circuit_programmer"

Preliminary steps:

A programming pcb is required, details of which are given in the BMP and Eagle attachments.
An Atmega 168 or 328 must be programmed independently using the following configuration bytes:

| | | | |
|---|---|---|---|
| The extended fuse byte | 0xFF | Fuse byte high | 0xD7 |
| Fuse byte | 0xE2 | Lock bits | 0xFF |

Critical parameters are:
  8MHz internal RC clock,                    65ms SUT, BOD disabled
  Watch dog under program control            EEPROM preserved during chip erase
  No extra memory protection

Obviously other sets of configuration bytes will also meet these requirements.  However these bytes are suitable for both 168 and 328 devices.

Once the pcb has been assembled and the Atmega devices loaded, the programmer will output the following user prompt to a PC running a terminal program.   ?????????

Note:   The serial port settings are 56700 baud, 8 data bits, no parity, 1 stop bit and no handshaking.

The user responds by pressing any key and then following the on screen instructions.  Below is a typical copy of the dialogue:

??

ATMEGA328 detected
Press P to program target or R to run target application or -X- to escape
8 MHz internal clock.
Press -P- to send a program file or -E- to send a text file.
Send Program file.
Integer(0-FF)?  0000    **********************************************

Atmega_programmer_V1.0
Config bytes: Fuses extended, high, low and lock
00FF 00D7 00E2 003F    on-chip cal bit:  005B 005B
Hex_file_size:3840  d'loaded: 3840  in: 3840  out
???????????

Notes:
1.      The asterisks are printed out as the flash memory is read back for verification.
2.      The term "on-chip" refers to the device hosting the programmer.
3.      The "on-chip cal byte" is repeated. This means that it is user selected (see project "Using the Atmega 328 internal RC clock" posted here by osbornema on Jan 10, 2019 for details).
4.      It is normally OK to used the default calibration byte in which case the "on-chip" cal byte is only given once.
5.      The programmer code can now be copied to any number of additional chips.  Note however that if identical programmers are allowed to run on the same pcb, UART contention issues will arise.  To avoid this, slightly different user prompts must be implemented.