NOTE 2007.6

:ELEMENTary GrEEK 101

;I've been told that reading the first dozen pages of the Picaxe Commands Manual is like reading Greek. Here is my TRANSLATION of the first dozen or so pages...

WHITESPACEuseplentyofwhitespacewhenwritingyourcodetomakeiteasiertoread

WHITESPACE use plenty of white space when writing

your code to make it easier to read

LABELS LABELS are names you place on LOCATIONS within a program

LABELS are defined by using a word or combination of letters & numbers at the beginning of a program line followed directly by a: (colon) when a label is referred to in another program line, no colon is used.

COMMENTS ANYTHING to the RIGHT of either:

a ; (semicolon) or a ' (apostrophe)

is IGNORED by the Editor and therefore is purely for the visual or intellectual enjoyment of the programmer

SYMBOLS SYMBOL is a command that is used to DEFINE an item

so that later in the program you can refer to the item by it's NAME instead of the actual item

WHICH program is more understandable as to what is going on?

:PROGRAM 1

High 1

Low 1

VARIABLES

* or * (using ALL the above COMMANDS)

:PROGRAM 2

SYMBOL LED=1 'LED is attached from Pin1 to +5 thru a 330ohm resistor

High LED Low LED line now high so LED is off!

Low LED Line is now low so LED is lit!

The use of WHITESPACE, COMMENTS and SYMBOLS allows the programmer to create an easy to read

AND more easily understandable program...

MOST of the time, things (numbers) are dynamically changing inside a program when it is running. Geeks refer to

these items as program VARIABLES

The PICAXE 08M has SPECIAL RAM (Random Access Memory) set aside for these VARIABLES and special NAMES for their ADDRESSES

RAM USAGE FOR THE PICAXE 08M

BITS: there are 16 special 1 bit VARIABLES NAMED Bit0 thru Bit15

these BIT VARIABLES are located INSIDE the 1st two BYTE VARIABLES

so BIT0-BIT7 are the bits INSIDE BYTE 0 and BIT8-BIT15 are the bits INSIDE BYTE 1

BYTES: there are 14 special 8 bit VARIABLES NAMED B0 thru B13

WORDS: there are 7 16 bit VARIABLES NAMED W0 thru W6

these WORD variables occupy the same RAM as the BYTE variables and simply use the RAM BYTES in PAIRS to create 16 bit values

W0 is BYTE1 & BYTE 0 (MSB/LSB)

W1 is BYTE3 & BYTE 2

etc

W6 is BYTE 13 & BYTE 12

Since there is a very LIMITED space/RAM for VARIABLES, in complicated programs GOOD GEEKS (programmers) try to use the appropriately sized VARIABLE for the data contained therein:

> a ON/OFF, YES/NO, TRUE/FALSE variable would be in a BIT VARIABLE a number between 0 & 255 would be in a BYTE VARIABLE and a number between 0 & 65335 would be in a WORD VARIABLE

;S0

This program monitors a potentiometer to see if it has EVER been turn more than half way if so, an LED turns on if not, the LED remains off

WHY? Say the potentiometer is really a shock sensor and you want to track whether the board had received a LARGE shock while in use....

SYMBOL ON OFF=Bit0

SYMBOL LED=2 SYMBOL POT=1 'LED status in Bit0 now NAMED ON OFF with 0=off & 1=on

'LED on Pin2 with 330ohm resistor to +5V

'wiper of 10K potentiometer on Pin1

with 100ohms to +5 and ground on the other legs 'potentiometer reading stored in VARIABLE B6 now NAMED OHMS

SYMBOL OHMS=B6

ON OFF=0

WHAT: READADC POT, OHMS

IF OHMS<128 THEN LITE

ON OFF=1

LITE: IF ON OFF=1 THEN ON

OFF: HIGH LED **GOTO WHAT** preset LED status to OFF

'take an A/D reading on POT (Pin1) and save in OHMS (B6)

'pot reading less than halfway?...

'change LED status to ON 'check LED status

'LED OFF

'keep watching the pot!

ON:

LOW LED

'LED ON because pot has gone more than half way at some time in history......

GOTO WHAT

'keep watching the pot!

END OF TRANSLATION!