

Istruzioni del PicBasicPlus

Mostriamo in questo capitolo una tabella con le istruzioni del PicBasicPlus, il terzo dei linguaggi con cui stiamo lavorando.

MNEMONICO	PARAMETRI	DESCRIZIONE
Operatori aritmetici (16 bit)		
+		Somma
-		Sottrazione
*		Parte bassa della moltiplicazione
**		Parte alta della moltiplicazione
*		Moltiplicazione con decimali
/		Divisione
//		Resto della divisione
Operatori logici (livello bit)		
&		AND
		OR
^		XOR
Operatori di rotazione		
<<		Rotazione a sinistra
>>		Rotazione a destra
Comparatori (livello byte)		
=		Uguale a
<		Minore di
>		Maggiore di
◇		Diverso da
≤		Minore o uguale a
≥		Maggiore o uguale a
Strutture di controllo		
IF condizione THEN istruzione: {istruzione}		Se si compie la condizione si eseguono le istruzioni
IF condizione THEN istruzione: {istruzione}; ELSE istruzione: {istruzione}		Se si compie la condizione si eseguono le prime istruzioni. Altrimenti le seconde
IF condizione THEN istruzione {istruzione} ELSE istruzione {istruzione} ENDIF		Se si compie la condizione si eseguono le prime istruzioni. Altrimenti le seconde
BRANCH var, [etich1 {, ... etichn}]		Salta a un'etichetta secondo la variabile
BRANCHL var, [etich1 {, ... etichn}]		Salta a un'etichetta secondo la variabile. Per memorie di istruzioni superiori a una pagina
GOTO {etichetta} GOSUB Etichetta ... Etichetta ... RETURN		Salta alla parte del programma indicata dall'etichetta. Ingresso alla subroutine indicata nell'etichetta e uscita dalla medesima
CALL Etichetta ...		Ingresso alla subroutine indicata dall'etichetta e uscita dalla medesima. Per subroutine in assembler



ASM Etichetta ... RETURN ENDASM		
FOR variabile=valore_iniziale TO valore_finale [STEP passo_salto] istruzioni NEXT		Si eseguono le istruzioni dal valore_iniziale sino al valore_finale di una variabile, con incrementi automatici di quest'ultima ogni passo_salto
WHILE condizione istruzioni WEND		Si ripete l'esecuzione delle istruzioni mentre si compie la condizione
REPEAT istruzioni UNTIL condizione		Si ripete l'esecuzione delle istruzioni sino a quando si compie la condizione
Istruzioni di ingresso/uscita		
CLEAR	Porta.pin , Simbolo o variabile	Impostazione a 0 di un bit di una Porta, o Impostazione a 0 di una variabile
LOW	Porta.pin , Simbolo o variabile	Impostazione a 0 di un bit di una Porta, o Impostazione a 0 di una variabile
SET	Porta.pin , Simbolo o variabile	Impostazione a 1 di un bit di una Porta, o Impostazione a 1 di una variabile
HIGH	Porta.pin , Simbolo o variabile	Impostazione a 1 di un bit di una Porta, o Impostazione a 1 di una variabile
Gestione della memoria dei dati		
variabile=PEEK	registro_RAM	Acquisisce il valore contenuto in un registro_RAM e lo carica su una variabile
POKE	registro_RAM, valore	Deposita un valore su un registro_RAM
variabile=EREAD	indirizzo_EEPROM	Acquisisce il valore contenuto in un indirizzo_EEPROM e lo carica su una variabile
EWRITE	indirizzo_EEPROM, [val1 {...valn}]	Si scrivono i valori a partire dall'indirizzo_EEPROM
EDATA	val1 ... valn	Si scrivono i valori a partire dall'indirizzo 0 della EEPROM
Tabelle		
DATA	dato_alfanumerico1, dato_alfanumerico2,...	Crea una tabella di dati alfanumerici
RESTORE READ	indirizzo_tabella variabile	L'indice punta all'indirizzo_tabella Sulla variabile si memorizza il valore puntato dall'indice
CDATA	dato_alfanumerico1, dato alfanumerico2,...	Crea una tabella di dati alfanumerici nella memoria di programma
variabile=CREAD	indirizzo	Sulla variabile si memorizza il valore contenuto nell'indirizzo della memoria di programma
CWRITE	indirizzo, [val1,...valn]	Si scrivono valori nella memoria di programma a partire dall'indirizzo specificato
Lavoro con dati semplici		
variabile=INKEY		Si acquisisce su una variabile il valore del tasto premuto
variabile=LOOKUP	tasto, [val1,val2,... val16]	Trasforma il tasto acquisito dall'istruzione INKEY nel valore che ha sulla tastiera e lo scrive su una variabile
variabile=COUNTER	Porta.pin, tempo	Conta il N° di impulsi positivi che entrano su un pin per un determinato periodo, e lo scrive su una variabile
variabile=PULSIN	Porta.pin, tempo	Inizia a contare il tempo all'arrivo dell'impulso e termina quando si ripristina il livello iniziale, memorizzando il tempo su una variabile
PULSOUT	Porta.pin, tempo, polarità	Produce un impulso su un pin della durata e polarità {HIGH, LOW} specificate

