

Micro-Instruction Comparison

Daniel 2017/6/9 Initial Version

New PEB Board				Old PEB Board	
Decode	Instruction	Timing Set	Operand		
3'b000	NOP/RPT	6-bit	0~65535	RPT	
NOP/RPT 的 operand 為 16-bit。 當 operand 不等於 0 時，實際上此行會重複執行 operand+1 次，等同於 RPT 指令。				Repeat 次數為 2~65535，注意執行次數並未+1。	
3'b001	CJMP	6-bit	23-bit Address	Cycle On Testing	
Conditional Jump 的條件為 CPU Flag，當 Flag=1 時才會跳到 Operand 指定的 Address。反之當 Flag=0 時，則繼續執行下一行指令。					
3'b010	JSR	6-bit	23-bit Address	CALLPRE / CALL	
Jump to Subroutine，執行後會跳到 Operand 指定的 Address(副程式的開頭)，同時將目前的下一行 Address(返回的 Address)丟入堆疊中。Stack 有 16 層，即支援 16 層的 Nested 巢狀副程式呼叫。				CALLPRE / CALL 兩組指令才能完成 23-bit 的記憶體定址。 未支援巢狀呼叫副程式。	
3'b011	RTN	6-bit	XXXX	RET	
跳出副程式，回到堆疊中記錄的位置。此時 Operand 中內容無任何意義。				Operand 內容為副程式迴圈次數。	
3'b100	PLC	6-bit	0~65535	FLOOP	
Push Loop Counter Stack，將 Operand+1 的迴圈次數存於堆疊中。Stack 有 16 層，即支援 16 層的 Nested 巢狀迴圈。Loop Counter 的 Stack 與 Subroutine 的 Stack 為完全獨立不互相影響。				Loop 次數為 2~65535，注意執行次數並未+1。 未支援巢狀迴圈。	
3'b101	EOL	6-bit	23-bit Address	FLEND	
End of Loop，迴圈次數若未歸零則跳至 Operand 內指定的位置(注意不要直接指向 PLC 指令位置而造成重複 push stack)，迴圈次數若為零則執行下一行位置。				FLEND 不帶 operand，LOOP 次數未完則固定回到 FLOOP 那一行。	
3'b110	END	6-bit	XXXX	STOP	
程式結尾，結束 Burst。此時 Operand 中內容無任何意義。				程式結尾。	
3'b111	MCH	6-bit	0~65535	MATCH	
Immediate match，最多 match 次數為 Operand+1。每次發現 match site fail 時 match counter - 1，並重覆執行本指令直到 pass 或 match counter = 0。				最多 match 次數為 2~65535，注意執行次數並未+1。	

Pattern Decode Comparison

	Old PEB Board			New PEB Board				
	M	D	F	Timing Set Memory Data Format				
				Pattern Data = Timing Set Memory Address [2:0]	Pin Mode Type	Drive Marker 1 (Location 1) / Drive Marker 2 (Location 2)	Compare Marker 1 (Location 1) / Compare Marker 2 (Location 2)	IO Marker 1 (Location 1) / IO Marker 2 (Location 2)
0	0	0	0	3'b000	NF	Drive_Low(0) / Drive_Low(0)	Comp_X(0) / Comp_X(0)	Drive_En(0) / Drive_En(0)
					RZ	Drive_Low(L1) / Drive_Low(L2)	Comp_X(0) / Comp_X(0)	Drive_En(0) / Drive_En(0)
					RO	Drive_Low(L1) / Drive_High(L2)	Comp_X(0) / Comp_X(0)	Drive_En(0) / Drive_En(0)
					SBC	Drive_Fall(L1) / Drive_Rise(L2)	Comp_X(0) / Comp_X(0)	Drive_En(0) / Drive_En(0)
					NRZ	Drive_Low(L1) / Drive_Low(L2)	Comp_X(0) / Comp_X(0)	Drive_En(0) / Drive_En(0)
1	0	0	1	3'b001	NF	Drive_High(0) / Drive_High(0)	Comp_X(0) / Comp_X(0)	Drive_En(0) / Drive_En(0)
					RZ	Drive_High(L1) / Drive_Low(L2)	Comp_X(0) / Comp_X(0)	Drive_En(0) / Drive_En(0)
					RO	Drive_High(L1) / Drive_High(L2)	Comp_X(0) / Comp_X(0)	Drive_En(0) / Drive_En(0)
					SBC	Drive_Rise(L1) / Drive_Fall(L2)	Comp_X(0) / Comp_X(0)	Drive_En(0) / Drive_En(0)
					NRZ	Drive_High(L1) / Drive_High(L2)	Comp_X(0) / Comp_X(0)	Drive_En(0) / Drive_En(0)
X	0	1	0	3'b010		Don't Care	Comp_X(0) / Comp_X(0)	Drive_Dis(0) / Drive_Dis(0)
M	0	1	1	3'b011	Strobe	Don't Care	Comp_M(L1)	Drive_Dis(0) / Drive_Dis(0)
					Window	Don't Care	Comp_M(L1) / Comp_M(L2)	
L	1	0	0	3'b100	Strobe	Don't Care	Comp_L(L1)	Drive_Dis(0) / Drive_Dis(0)
					Window	Don't Care	Comp_L(L1) / Comp_L(L2)	
H	1	0	1	3'b101	Strobe	Don't Care	Comp_H(L1)	Drive_Dis(0) / Drive_Dis(0)
					Window	Don't Care	Comp_L(L1) / Comp_L(L2)	
V	1	1	0	3'b110	Not Support			
Z	1	1	1	3'b111		Don't Care	Comp_X(0) / Comp_X(0)	Drive_Dis(0) / Drive_Dis(0)

