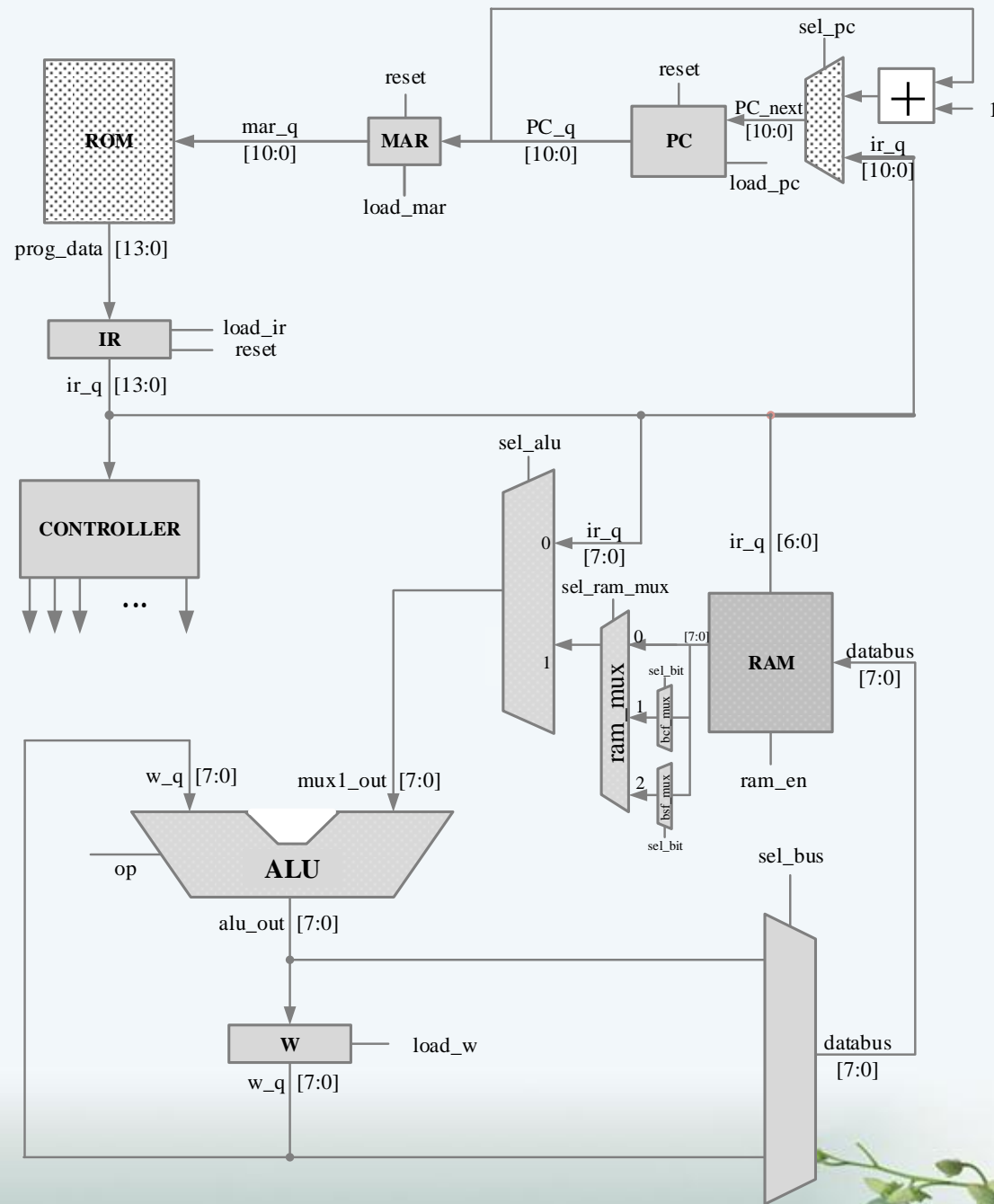


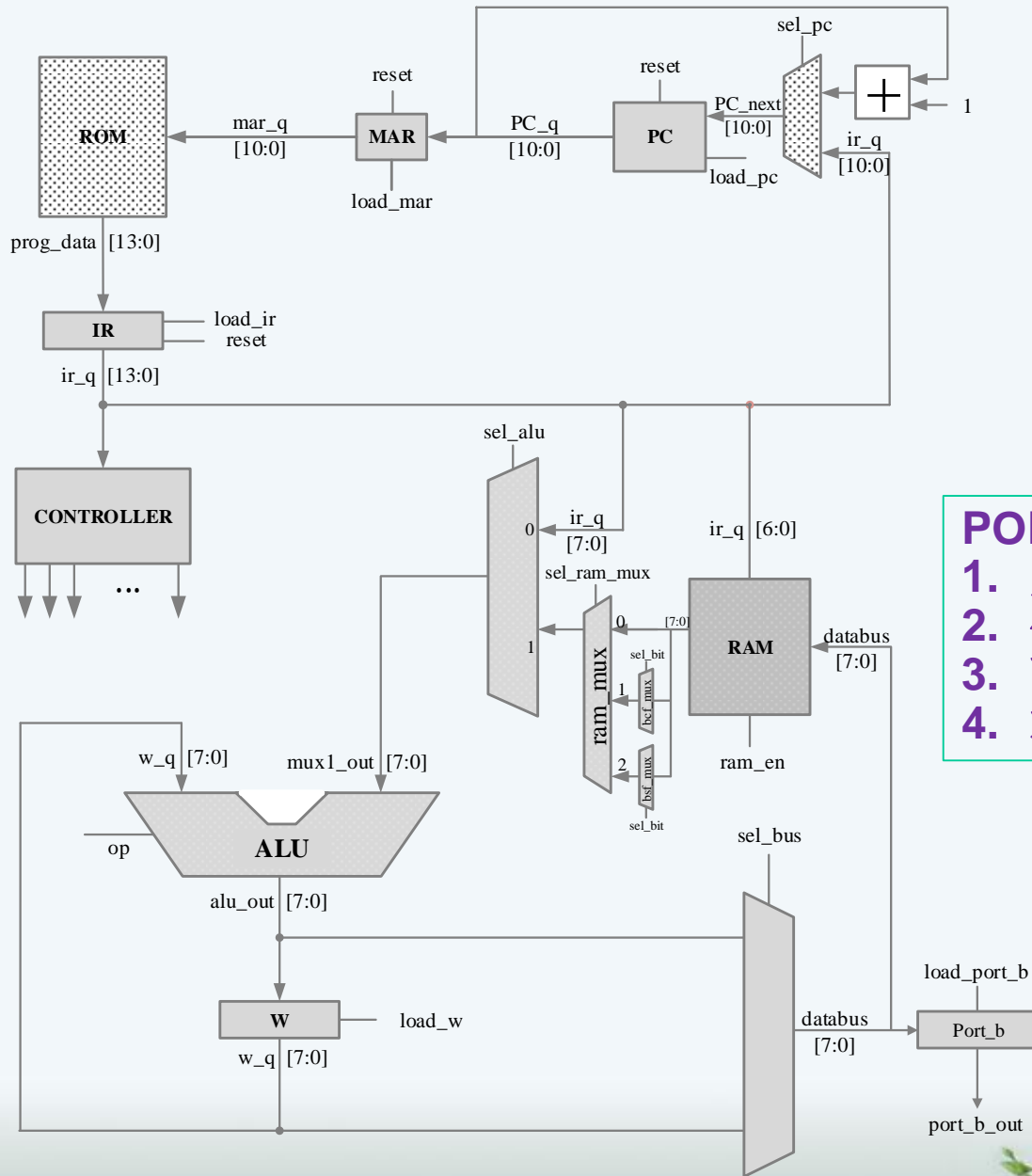
暫存器定址指令



舊架構圖



新架構圖



PORTB:

1. 為PIC對外的連接 I/O Port
2. 位址為0D
3. 可以為Input or Output
4. 本課程統一為輸出(output)



指令資料流

49個指令分成八個類別，
從八個類別中各挑出部分指令做控制訊號及資料流向範例。

各指令執行所需時間不盡相同，大致上可由類別區分：

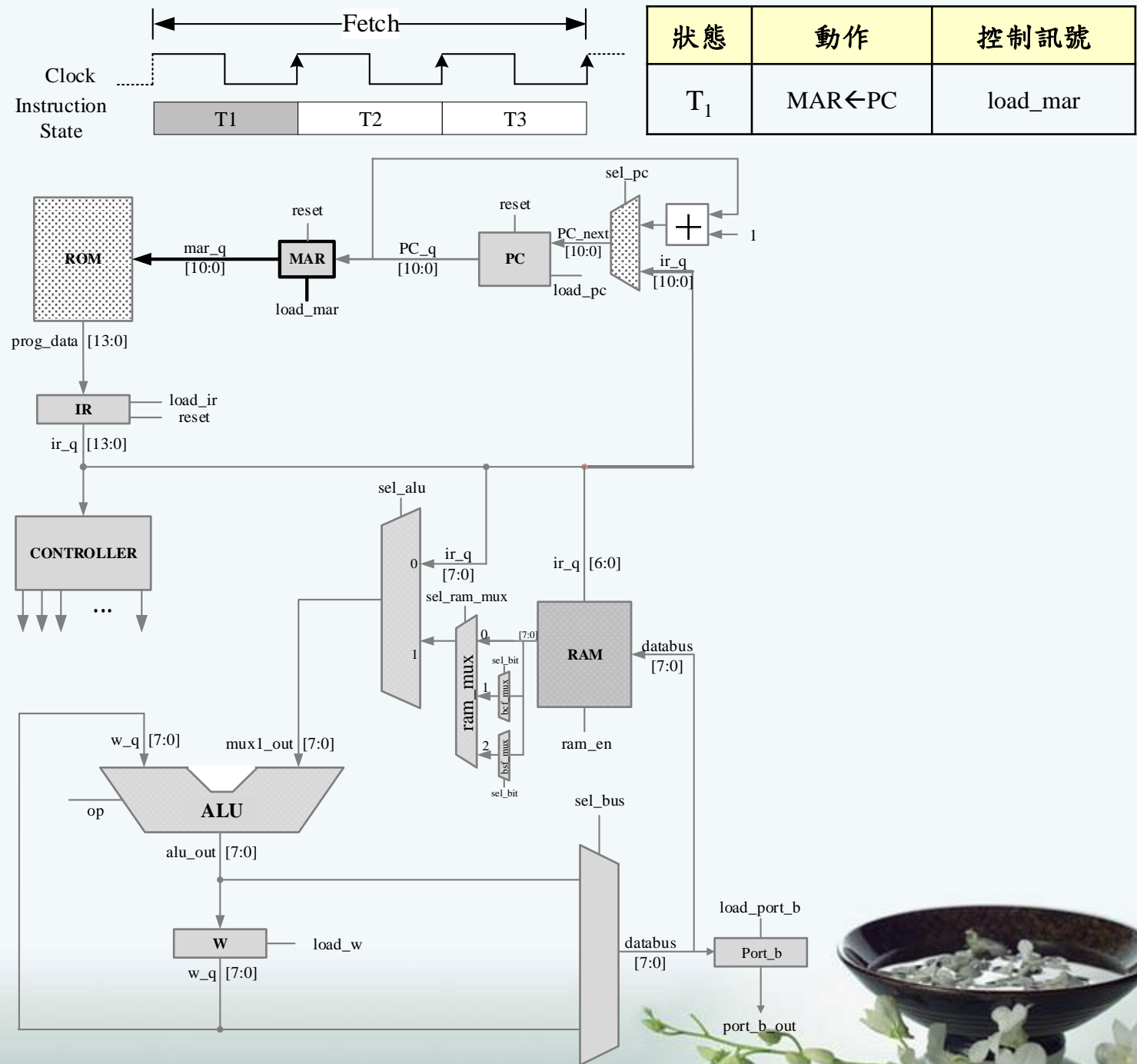
- 一個時間週期：
Literal Operations、Inherent Operations。
- 兩個時間週期：
Byte-oriented File Register Operations、Bit-oriented File Register Operations、
Bit-oriented Skip Operations。
- 三個時間週期：
Byte-oriented Skip Operations、Control Operations、C-Compiler Optimized。

T_1 、 T_2 及 T_3 擷取階段，控制訊號均相同，如下表所示。

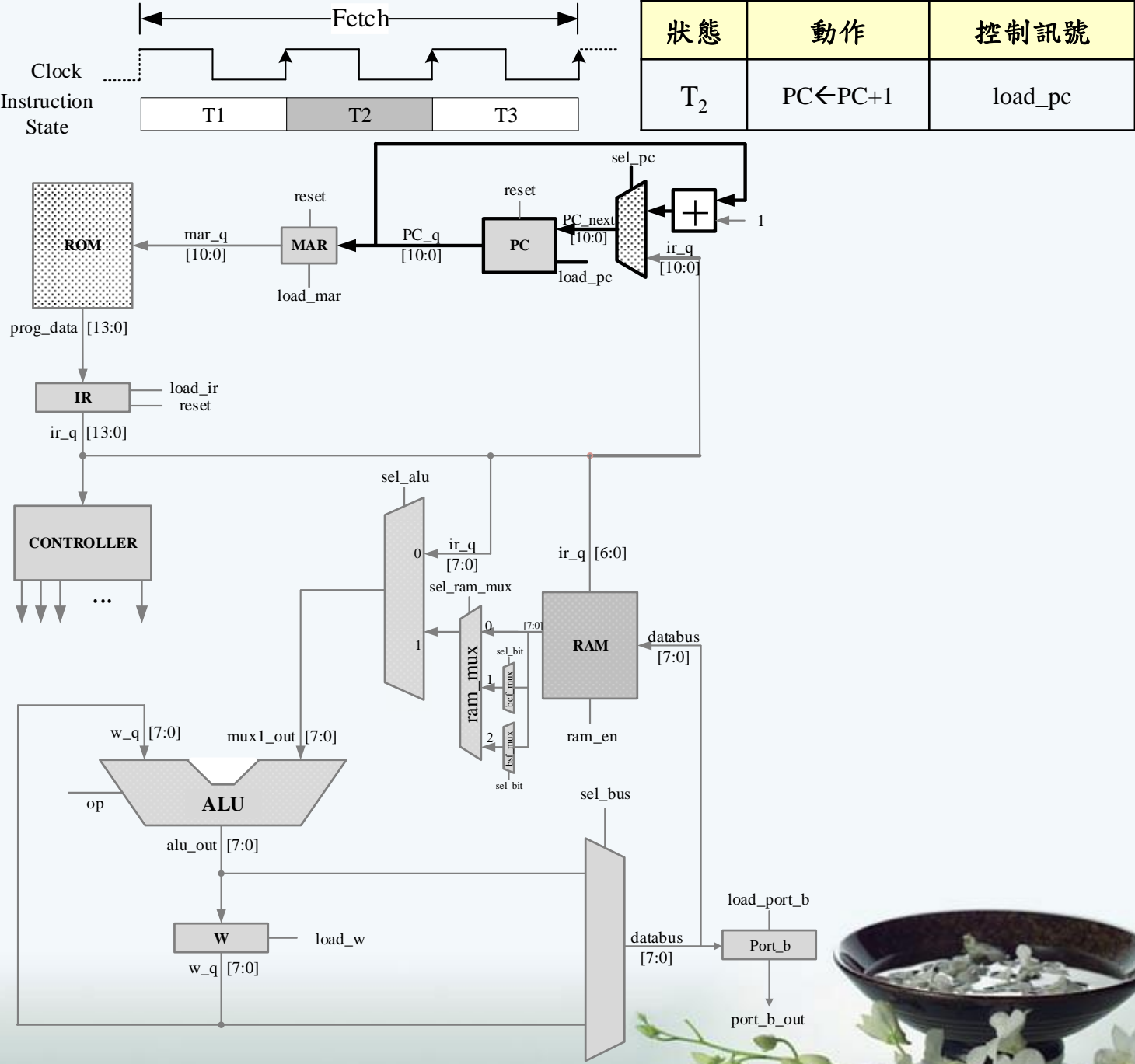
狀態	動作	控制訊號
T_1	$MAR \leftarrow PC$	load_mar
T_2	$PC \leftarrow PC + 1$	sel_pc; load_pc
T_3	$IR \leftarrow ROM[MAR]$	load_ir



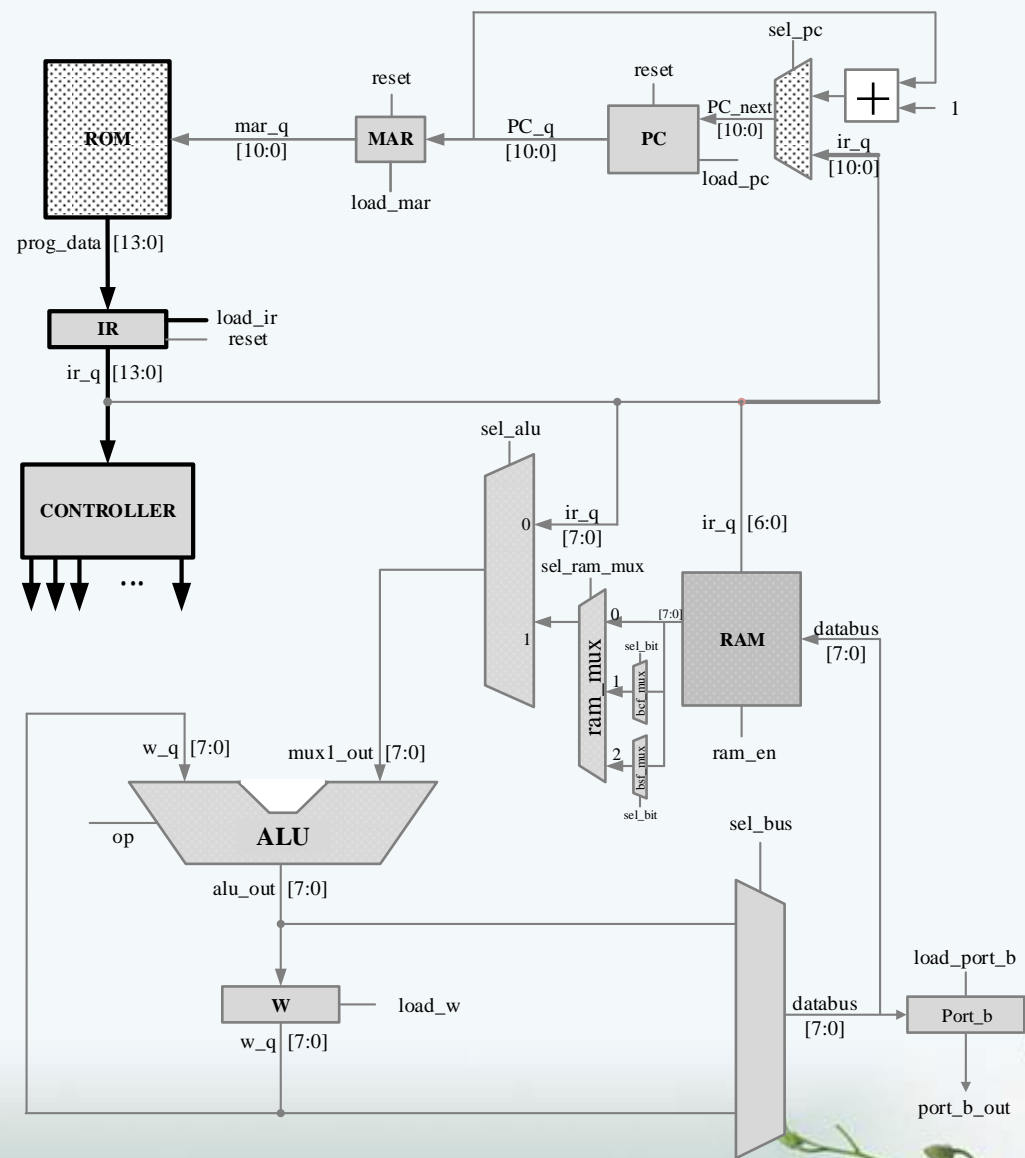
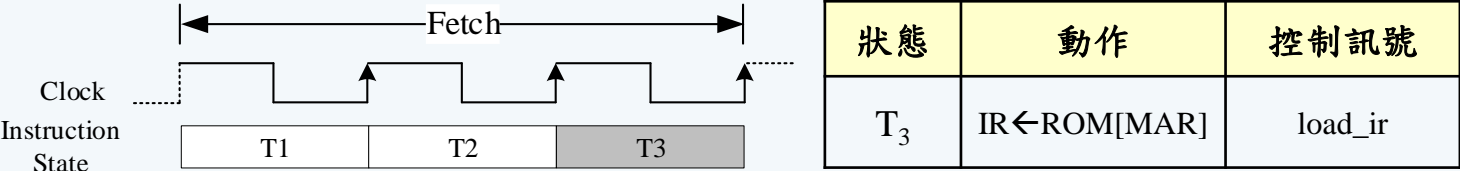
Fetch T1



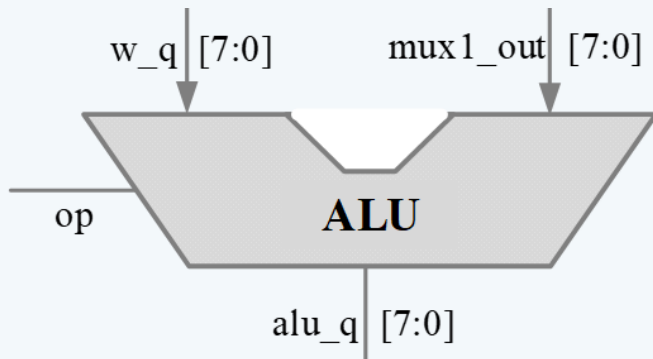
Fetch T2



Fetch T3



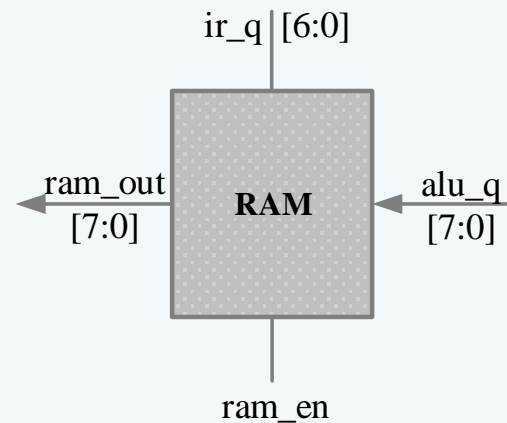
ALU



```
//ALU
always_comb
begin
    case (op)
        4'h0: alu_out = mux_1_out[7:0] + w_q;
        4'h1: alu_out = mux_1_out[7:0] - w_q;
        4'h2: alu_out = mux_1_out[7:0] & w_q;
        4'h3: alu_out = mux_1_out[7:0] | w_q;
        4'h4: alu_out = mux_1_out[7:0] ^ w_q;
        4'h5: alu_out = mux_1_out[7:0];
        4'h6: alu_out = mux_1_out[7:0]+1;
        4'h7: alu_out = mux_1_out[7:0]-1;
        4'h8: alu_out = 0;
        4'h9: alu_out = ~mux_1_out[7:0];
        4'hA: alu_out = {mux_1_out[7],mux_1_out[7:1]};
        4'hB: alu_out = {mux_1_out[6:0], 1'b0};
        4'hC: alu_out = {1'b0, mux_1_out[7:1]};
        4'hD: alu_out = {mux_1_out[6:0],mux_1_out[7]};
        4'hE: alu_out = {mux_1_out[0],mux_1_out[7:1]};
        4'hF: alu_out = {mux_1_out[3:0],mux_1_out[7:4]};
        default: alu_out = mux_1_out[7:0] + w_q;
    endcase
end
```



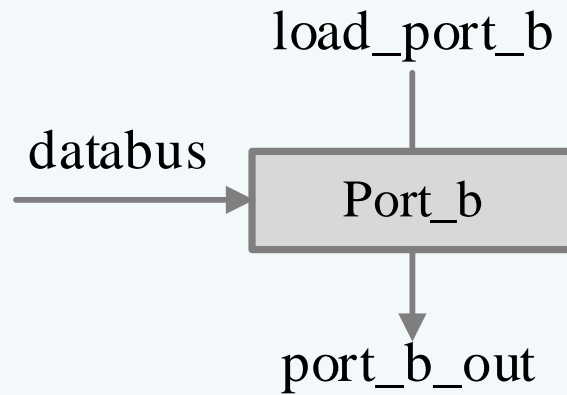
SRAM



```
module single_port_ram_128x8(  
    input [7:0]data,  
    input [6:0]addr,  
    input ram_en,  
    input clk,  
    output logic [7:0] q  
);  
  
    // Declare the RAM variable  
    //reg [DATA_WIDTH-1:0] ram[2**ADDR_WIDTH-1:0];  
    logic [7:0] ram[127:0];  
  
    always_ff @(posedge clk)  
    begin  
        // Write  
        if (ram_en)  
            ram[addr] <= data;  
    end  
  
    // Continuous assignment implies read returns NEW data.  
    // This is the natural behavior of the TriMatrix memory  
    // blocks in Single Port mode.  
  
    assign q = ram[addr];  
endmodule
```



Port_b

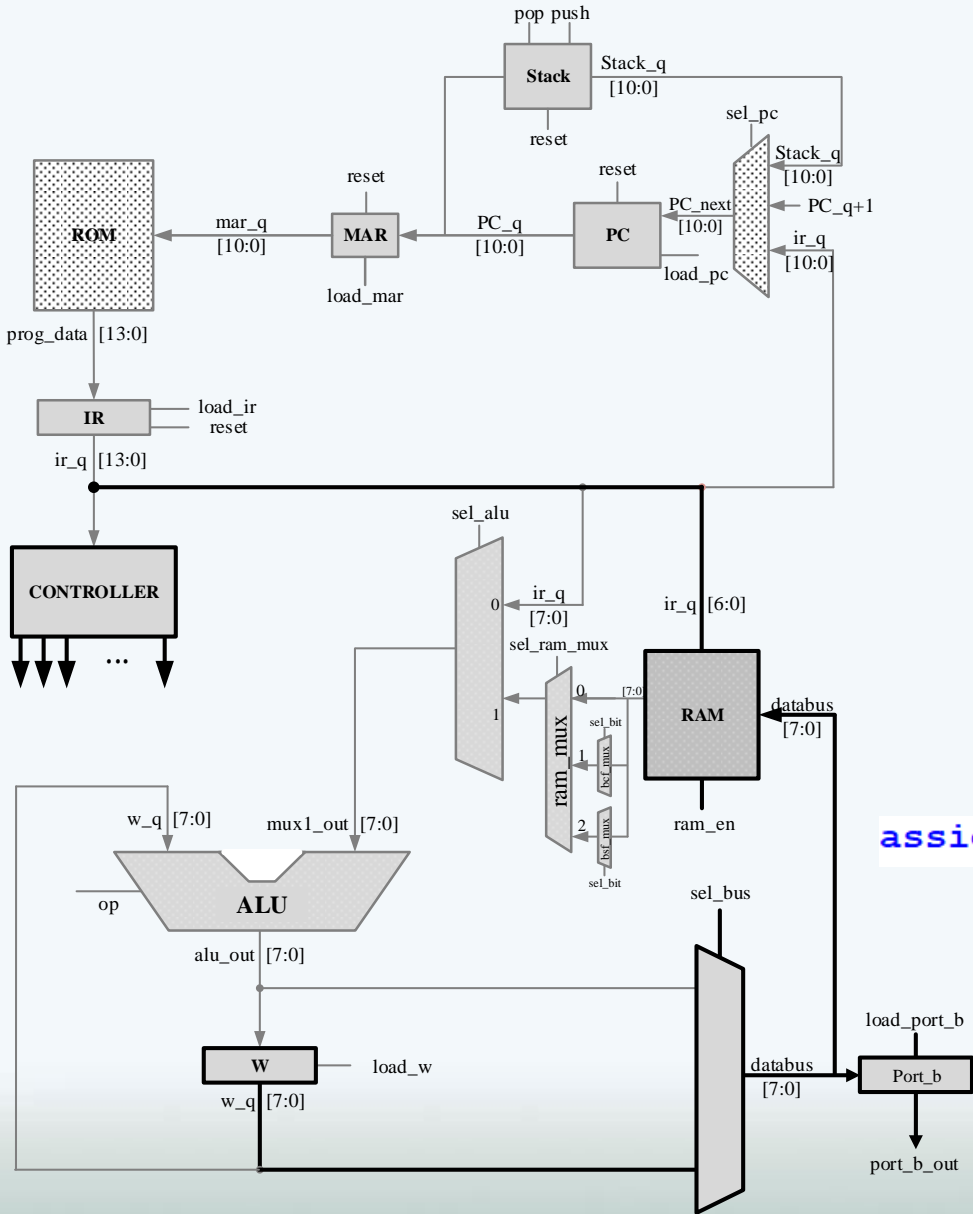


```
always_ff @(posedge clk)
  if(rst) port_b_out <= 0;
  else if(load_port_b) port_b_out <= data_bus;
```



MOVWF : 將W的內容傳到F

00 0000 1fff ffff



MOVWF PORTB

狀態	動作	控制訊號
T ₄	F ← W	sel_bus=1 if addr_port_b == 1 : load_port_b = 1 if addr_port_b == 0 : ram_en=1
T ₅	無動作	無
T ₆	無動作	無

```
assign addr_port_b = (ir_q[6:0] == 7'h0d);
```



BYTE-ORIENTED FILE REGISTER OPERATIONS



PIC16F1826 INSTRUCTION SET

Mnemonic, Operands	Description	Cycles	14-Bit Opcode				Status Affected	Notes
			MSB		LSB			
BYTE-ORIENTED FILE REGISTER OPERATIONS								
ASRF f, d	Arithmetic Right Shift	1	11	0111	dfff	ffff	Z	2
LSLF f, d	Logical Left Shift	1	11	0101	dfff	ffff	Z	2
LSRF f, d	Logical Right Shift	1	11	0110	dfff	ffff	Z	2
RLF f, d	Rotate Left f	1	00	1101	dfff	ffff		2
RRF f, d	Rotate Right f	1	00	1100	dfff	ffff		2
SWAPF f, d	Swap nibbles in f	1	00	1110	dfff	ffff		2

Note

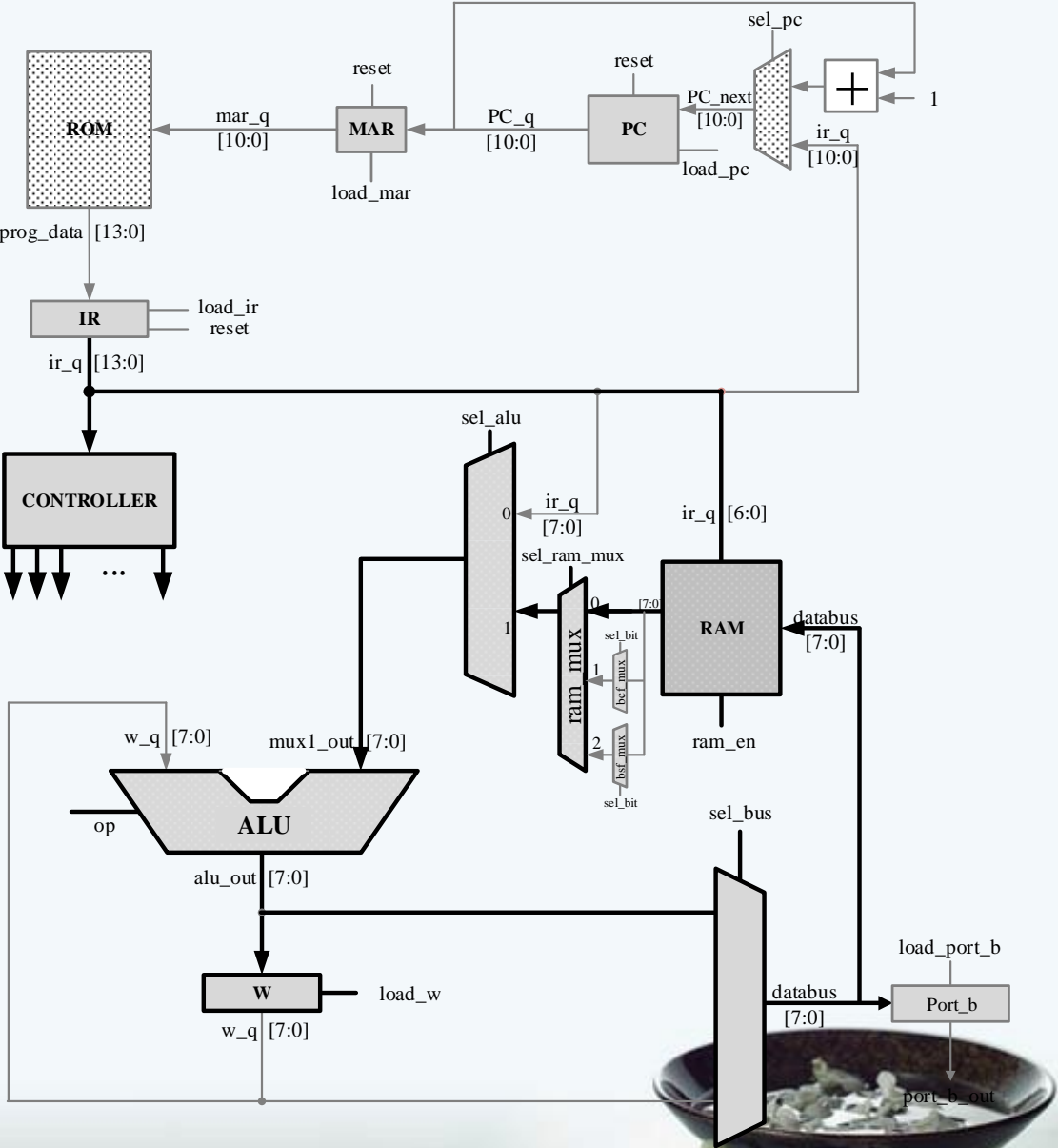
1 : If the Program Counter (PC) is modified, or a conditional test is true, the instruction requires two cycles. The second cycle is executed as a NOP.

2 : If this instruction addresses an INDF register and the MSb of the corresponding FSR is set, this instruction will require one additional instruction cycle.



T4

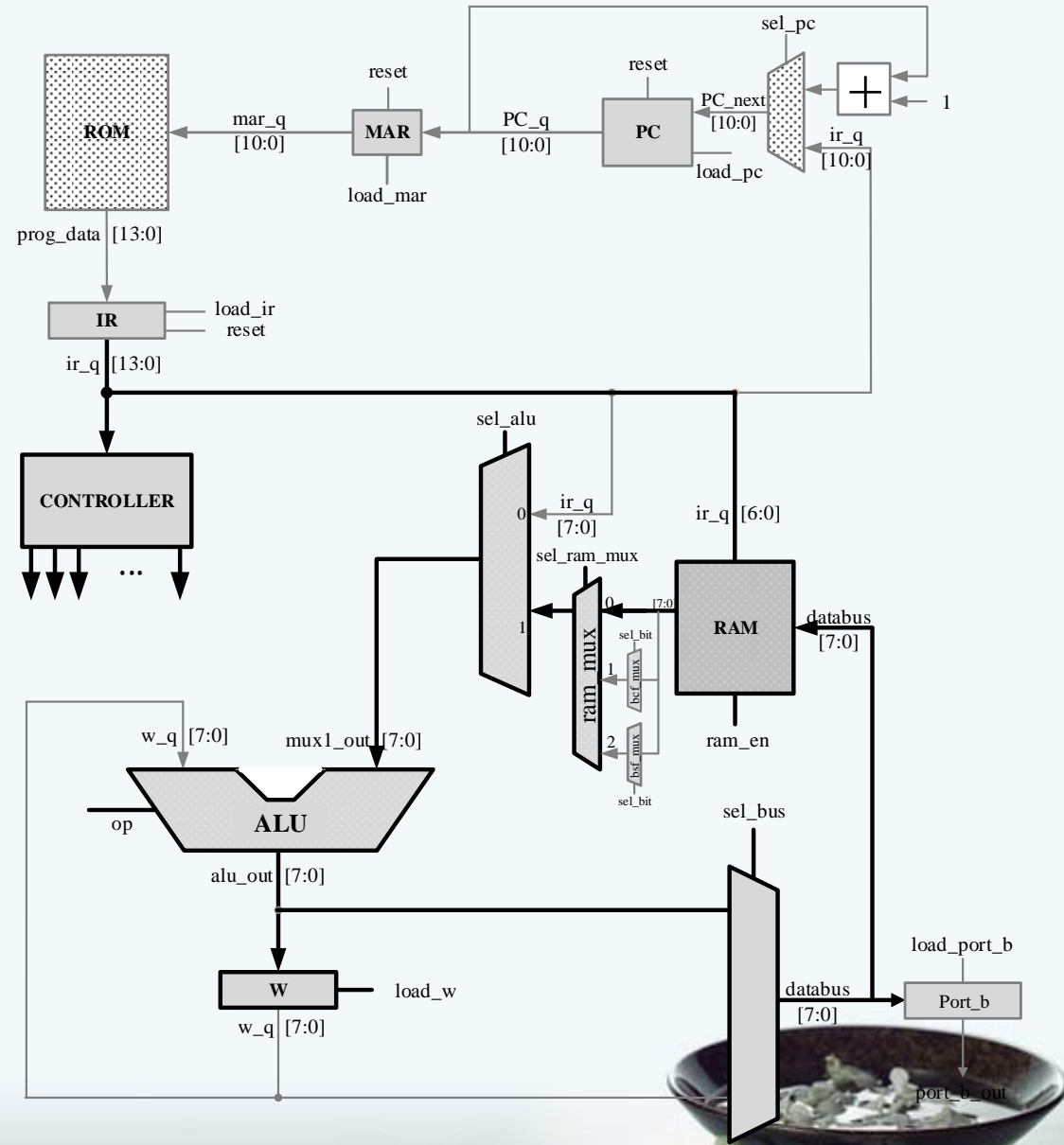
状態	動作	制御訊號
T ₄	<div>addr = ir[6:0]</div> <div>d == 0: W <= alu_out</div> <div>d == 1: RAM[addr] <= alu_out</div>	<div>sel_alu = 1 sel_ram_mux = 0 op = 4'hA</div> <div>d == 0: load_w = 1</div> <div>d == 1: sel_bus = 0 ram_en = 1</div>



LSLF

T4

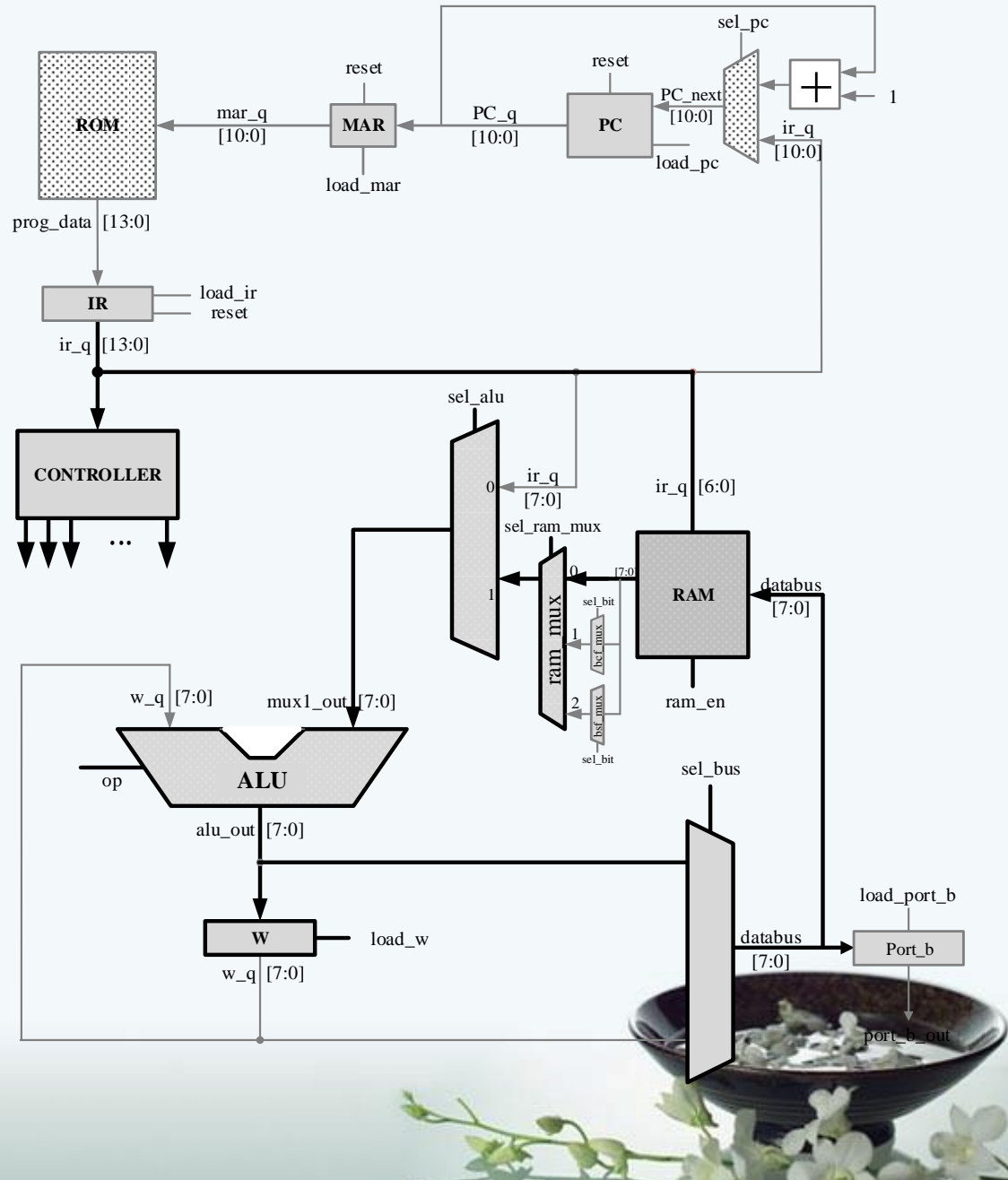
状態	動作	制御訊號
T ₄	$addr = ir[6:0]$ d == 0: $W \leftarrow alu_out$ d == 1: $RAM[addr] \leftarrow alu_out$	$sel_alu = 1$ $sel_ram_mux = 0$ $op = 4'hB$ d == 0: $load_w = 1$ d == 1: $sel_bus = 0$ $ram_en = 1$



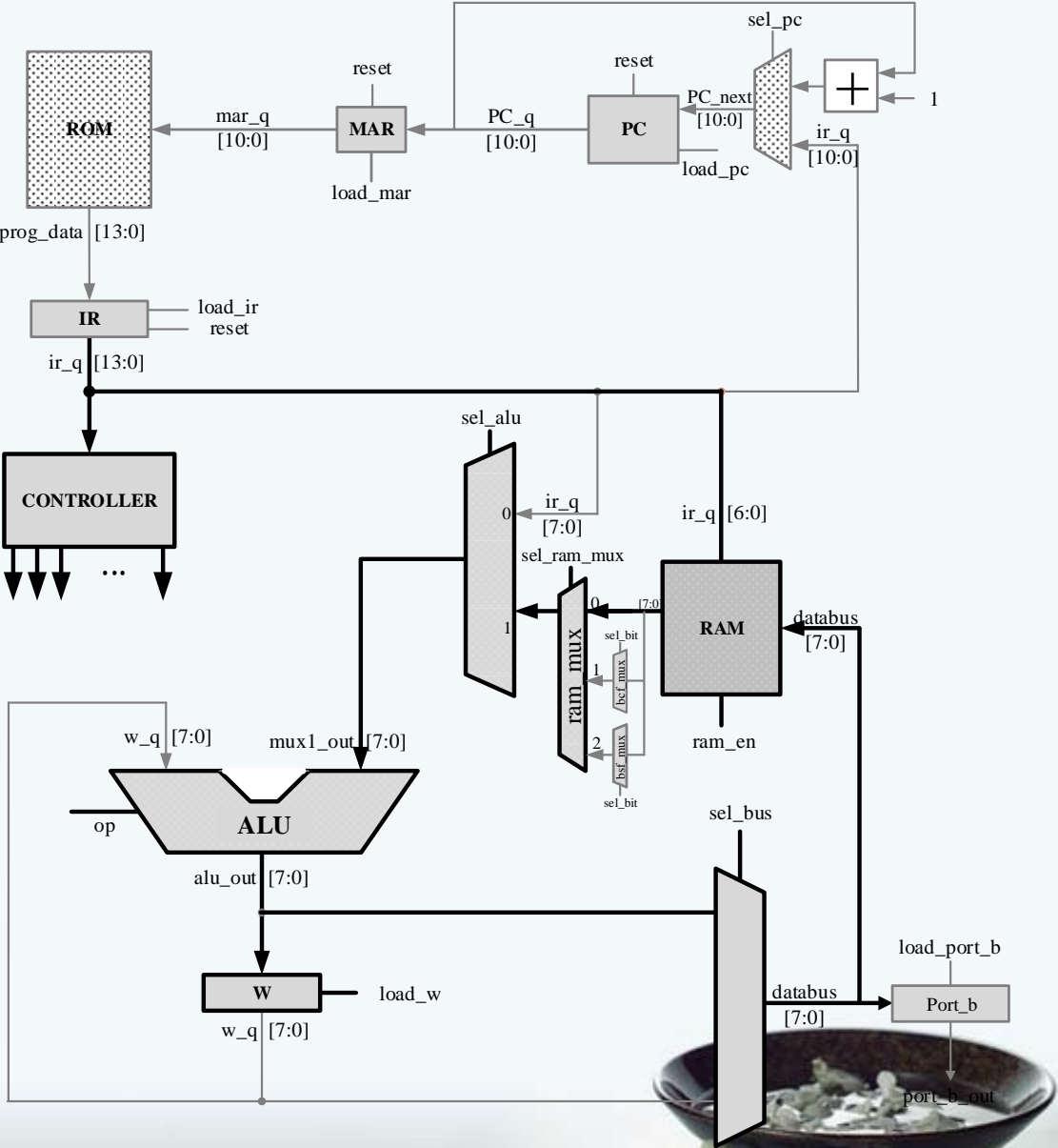
LSRF

T4

状態	動作	制御訊號
T ₄	$\text{addr} = \text{ir}[6:0]$ d == 0: $W \leftarrow \text{alu_out}$ d == 1: $\text{RAM}[\text{addr}] \leftarrow \text{alu_out}$	$\text{sel_alu} = 1$ $\text{sel_ram_mux} = 0$ $\text{op} = 4'hC$ d == 0: $\text{load_w} = 1$ d == 1: $\text{sel_bus} = 0$ $\text{ram_en} = 1$



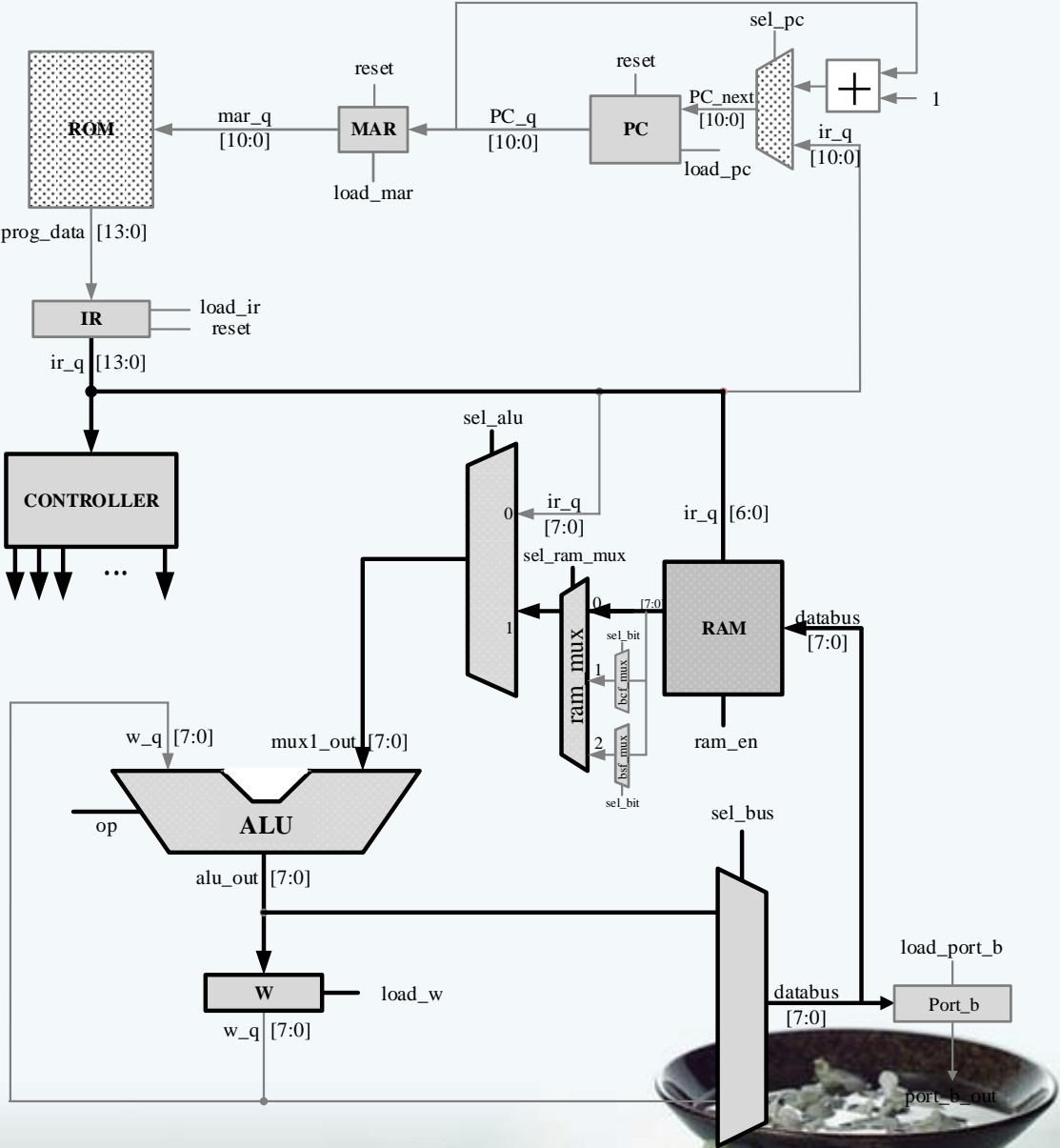
状態	動作	制御訊號
T ₄	<div>addr = ir[6:0]</div> <div>d == 0: W <= alu_out</div> <div>d == 1: RAM[addr] <= alu_out</div>	<div>sel_alu = 1 sel_ram_mux = 0 op = 4'hD</div> <div>d == 0: load_w = 1</div> <div>d == 1: sel_bus = 0 ram_en = 1</div>



RRF

T4

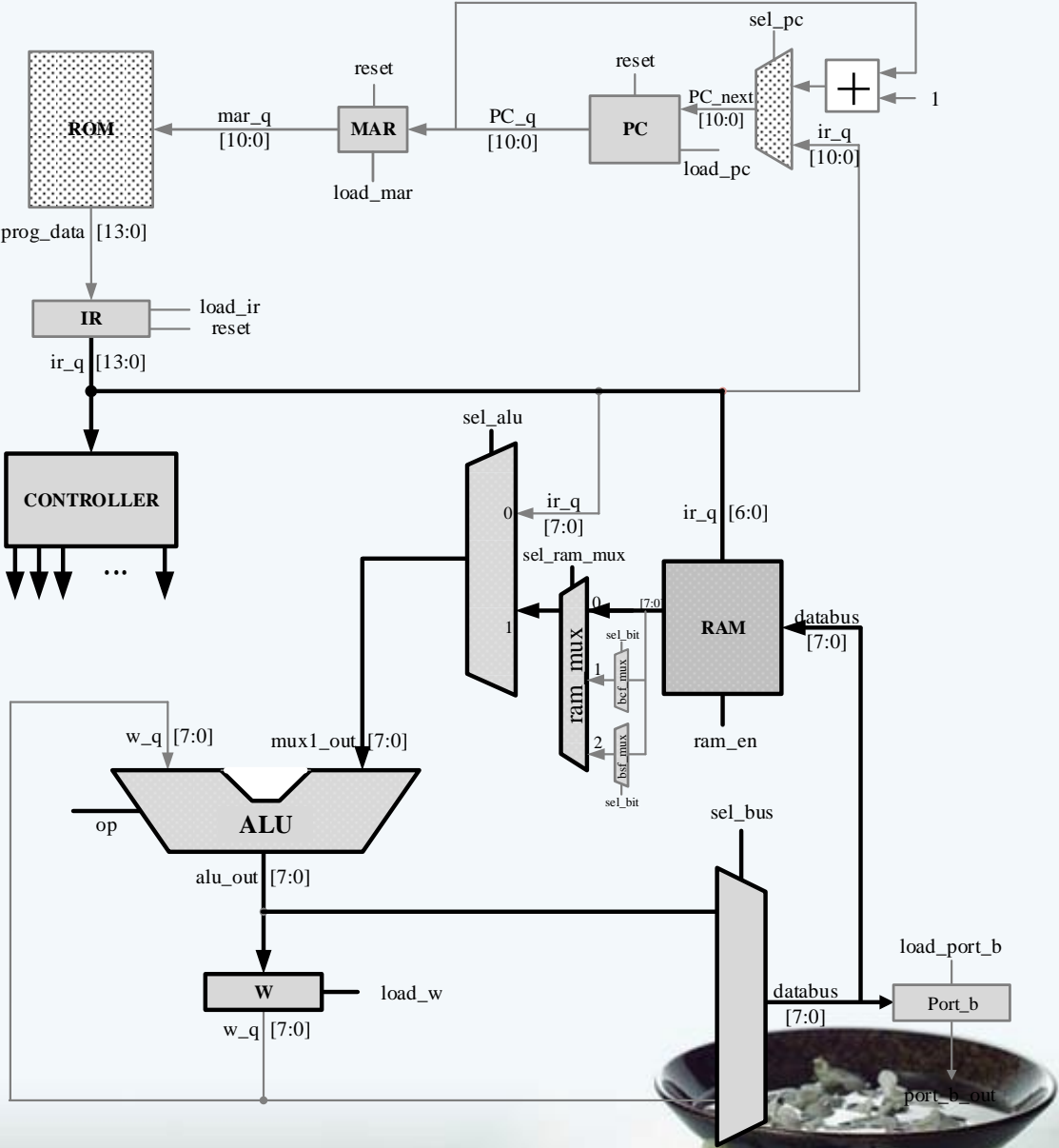
状態	動作	制御訊號
T ₄	$\text{addr} = \text{ir}[6:0]$ d == 0: $W \leftarrow \text{alu_out}$ d == 1: $\text{RAM}[\text{addr}] \leftarrow \text{alu_out}$	$\text{sel_alu} = 1$ $\text{sel_ram_mux} = 0$ $\text{op} = 4'hE$ d == 0: $\text{load_w} = 1$ d == 1: $\text{sel_bus} = 0$ $\text{ram_en} = 1$



SWAPF

T4

状態	動作	制御訊號
T ₄	$\text{addr} = \text{ir}[6:0]$ d == 0: $W \leftarrow \text{alu_out}$ d == 1: $\text{RAM}[\text{addr}] \leftarrow \text{alu_out}$	$\text{sel_alu} = 1$ $\text{sel_ram_mux} = 0$ $\text{op} = 4'hF$ d == 0: $\text{load_w} = 1$ d == 1: $\text{sel_bus} = 0$ $\text{ram_en} = 1$



上課實作：霹靂燈

燒錄

模擬

Port_b的值：

0000_0001 => 0000_0010=>0000_0100 =>
0000_1000 => 0001_0000=>0010_0000 =>
0100_0000 => 1000_0000=>0100_0000 =>
0010_0000 => 0001_0000=>0000_1000 =>
0000_0100 => 0000_0010=>0000_0001

不斷重覆

Module的output請換成port_b_out

TOP LEVEL INPUTS																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																			
------------------	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

```
#include <pl6Lf1826.inc> ; Include file locate at de
;

temp      equ      0x25
count1    equ      h'20'
count2    equ      h'21'
count3    equ      h'22'
;*****
;          Program start          *
;*****
org        0x00          ; reset vector

clr        temp          ; //ram[37]<=0
clrw       ; //w<=0
movlw     1              ; //w<=1
movwf     temp          ; //ram[37]<=0000_0001
loop1     lslf     temp,1 ; //ram[37]左移並存到w
movf      temp,0
movwf     PORTB
movlw     .3
movwf     count1
delay1    clrf     count2
delay2    clrf     count3
delay3    decfsz   count3,1
goto      delay3
decfsz    count2,1
goto      delay2
decfsz    count1,1
goto      delay1
btfss     temp,7
goto      loop1          ; //從0000_0001到1000_0000
loop2     lsr     temp,1 ; //ram[37]右移並存到w
movf      temp,0
movwf     PORTB
movlw     .3
movwf     count1
delay1    clrf     count2
delay2    clrf     count3
delay3    decfsz   count3,1
goto      delay3
decfsz    count2,1
goto      delay2
decfsz    count1,1
goto      delay1
btfss     temp,0
goto      loop2          ; //從10000_0000到0000_0001
goto      loop1
end
```