2018/19 S1 MST (Soln)

MCQ.

A1 – d

A2 – c

A3 – b

A4 – d

A5 – a

A6 – c

A7 – d

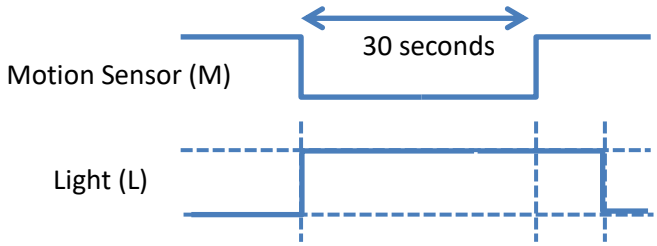
A8 – b

A9 – b

A10 – a

B1.

(a) Active low.
(b) TRISD = 0b00000001;
(c)



(d) (i) C (ii) B (iii) A (iv) D
(e) PIR sensor.

B2.

(a) Right-justified.
(b) ADCON1 = 0b 0 0 0 0 1 1 1 0;

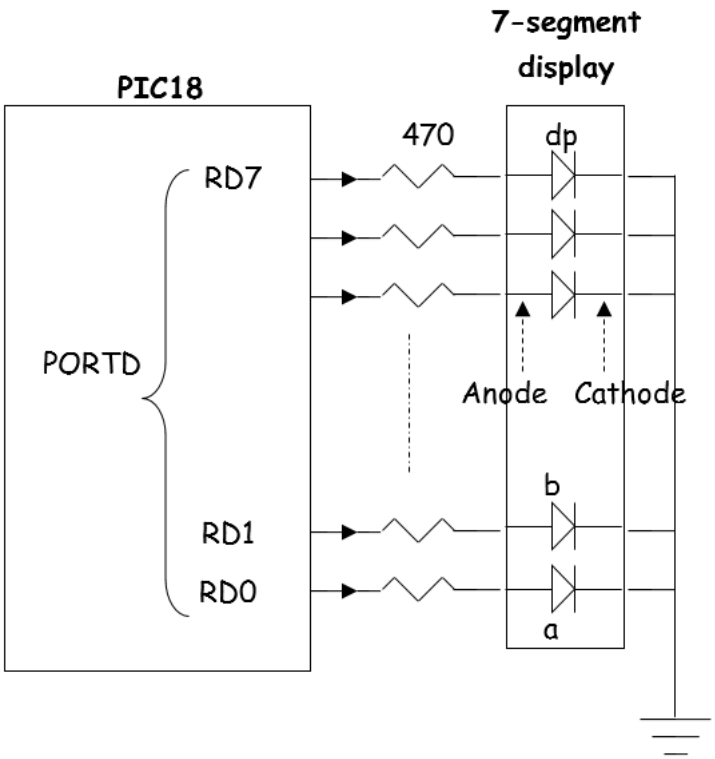
// use Vss & Vdd as voltage references, AN0 as analogue

(c) i. 00000001
ii. 00000001
iii. 00000010
iv. 00000011
(d) It is to wait for the A-D conversion to complete.

(e) Digital result = (2.8 / 5) x 1023 = 572.88 or 573 = 1000111101 i.e. 2 most significant bits are 10. So, LEDS connected to RD0 & RD1 will turn on.

B3.

(a)



(b) PORTD=0b00000110;

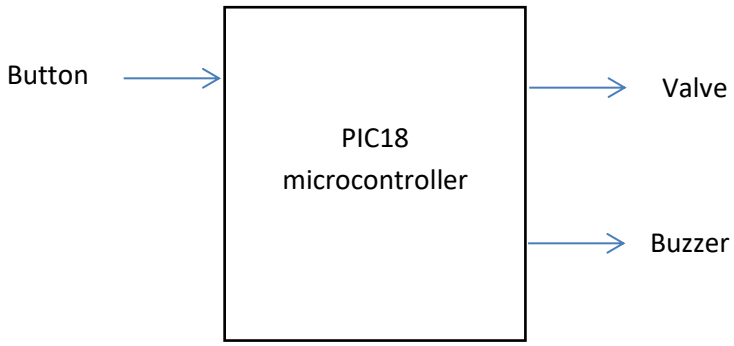
(c) else if (num == 2) // two vacant

(d)

```
unsigned char Count_vacancies (void)
{
  unsigned char vacancy = 0;
  if (PORTAbits.RA1==0) vacancy++;
  if (PORTAbits.RA2==0) vacancy++;
  if (PORTAbits.RA3==0) vacancy++;
  return(vacancy);
}
```

B4.

(a)



(b)

i. Initialise: Close Valve, Off Buzzer
ii. Button pressed?
iii. Open Valve
iv. Yes
v. No
vi. Delay 0.5 s
vii. On Buzzer
viii. Off Buzzer

(c) 0.5 Sec

B5.

(a) 0110.
(b) 4.
(c) Change the for-loop to loop more times e.g.

```
for ( k = 0;  k < 200; k++ ) {
 delay_ms (5);
 PORTCbits.RC0 = !PORTCbits.RC0;
}
```

(d)

```
while (1) {
    while ( ( PORTA & 0x0F ) == 0x0F ); // wait if bottom 4 bits all 1's
    Key = PORTA & 0x0F; // read RA3-0
    PORTB = Key; // write to RB3-0
    BUZZ (); // produce beep sound
}
```