

تطبيقات ميكاترونك -1-

Lecture No. 6

- Warehouse and WMR

روبوت و أنظمة ذكية - سنة ثالثة

Dr. Eng. Essa Alghannam
Ph.D. Degree in Mechatronics
Engineering

2022

تطبيق 1

- لدينا مستودع يراد التحكم بالدخول له وإضاءته من خلال شريحة متحكم أول PIC1 من نوع 16f877a موصول على هزاز شريحة كريستالي بتردد 4MHz.
- كما يحوي المستودع في الداخل روبوت تفاضلي بعجلتين متصلتين بمحركين تيار مستمر. ويقاد من قبل متحكم ثاني PIC2 من نوع 16f877a موصول على هزاز شريحة كريستالي بتردد 4MHz.

ملاحظة: لا يوجد اتصال سلكي بين المتحكمين الأول والثاني. وتكون كل الشاشات مطفأة في البداية.

يتصل بالمتحكم الأول:

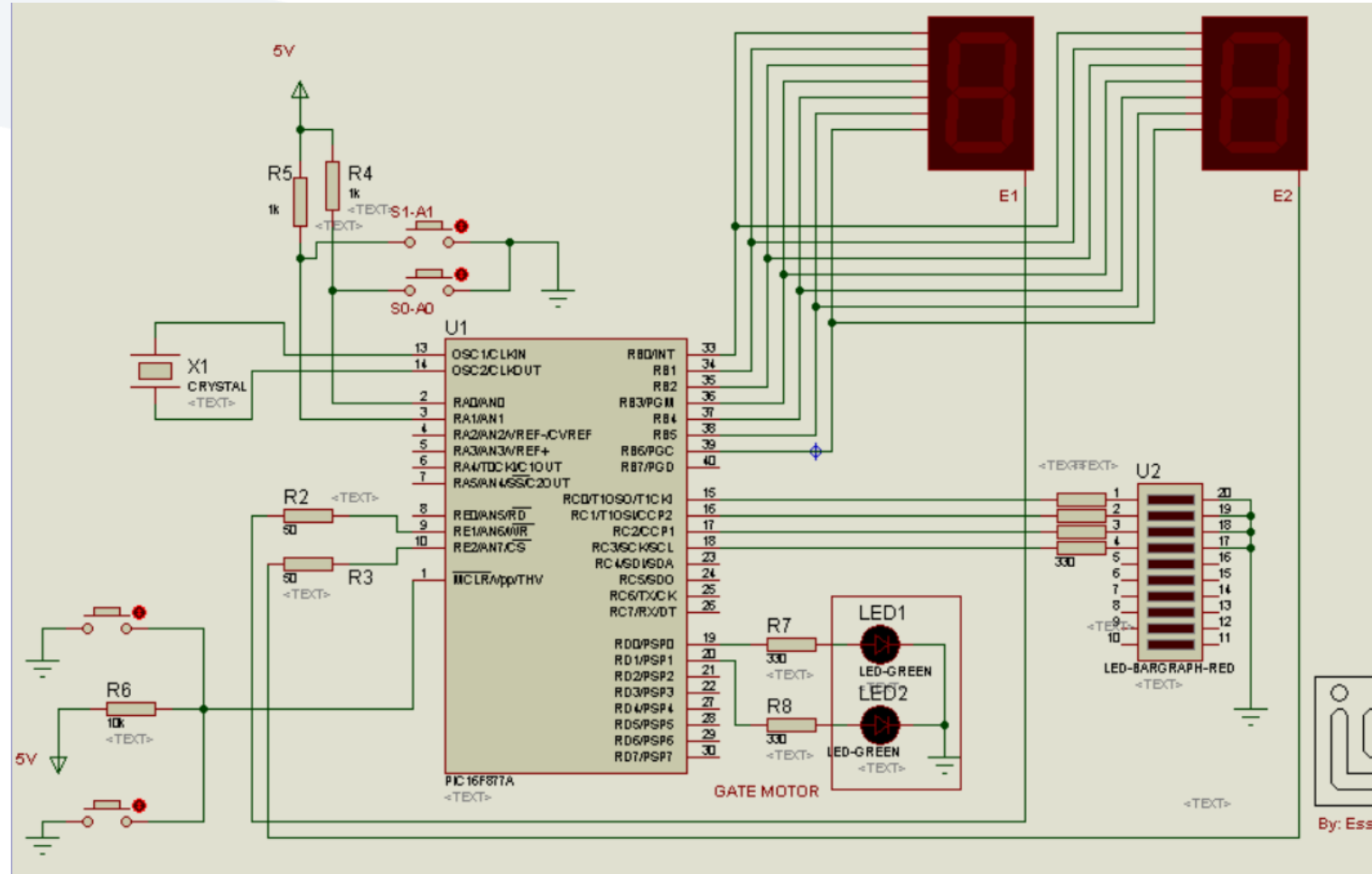
- مفتاحين s0 و s1 من نوع Push Button على الأقطاب a0 و a1 على التوالي بعد شدهما إلى المنطق العالي HIGH
- شاشات 7seg من نوع مهبط مشترك عدد اثنين من دون decoder متصلين على التفرع إلى البوابة B. (المهابط المشتركة متصلة إلى الأقطاب E1,E2 كما يلي E1-first 7seg , E2-second 7seg
- أربع ليدات متصلة إلى البوابة C0->C3
- يفتح ويغلق باب المستودع من خلال محرك تيار مستمر GATE MOTOR (تم التعبير عن هذا المحرك من خلال ديودين ضوئيين متصلين إلى القطبين D0-D1)

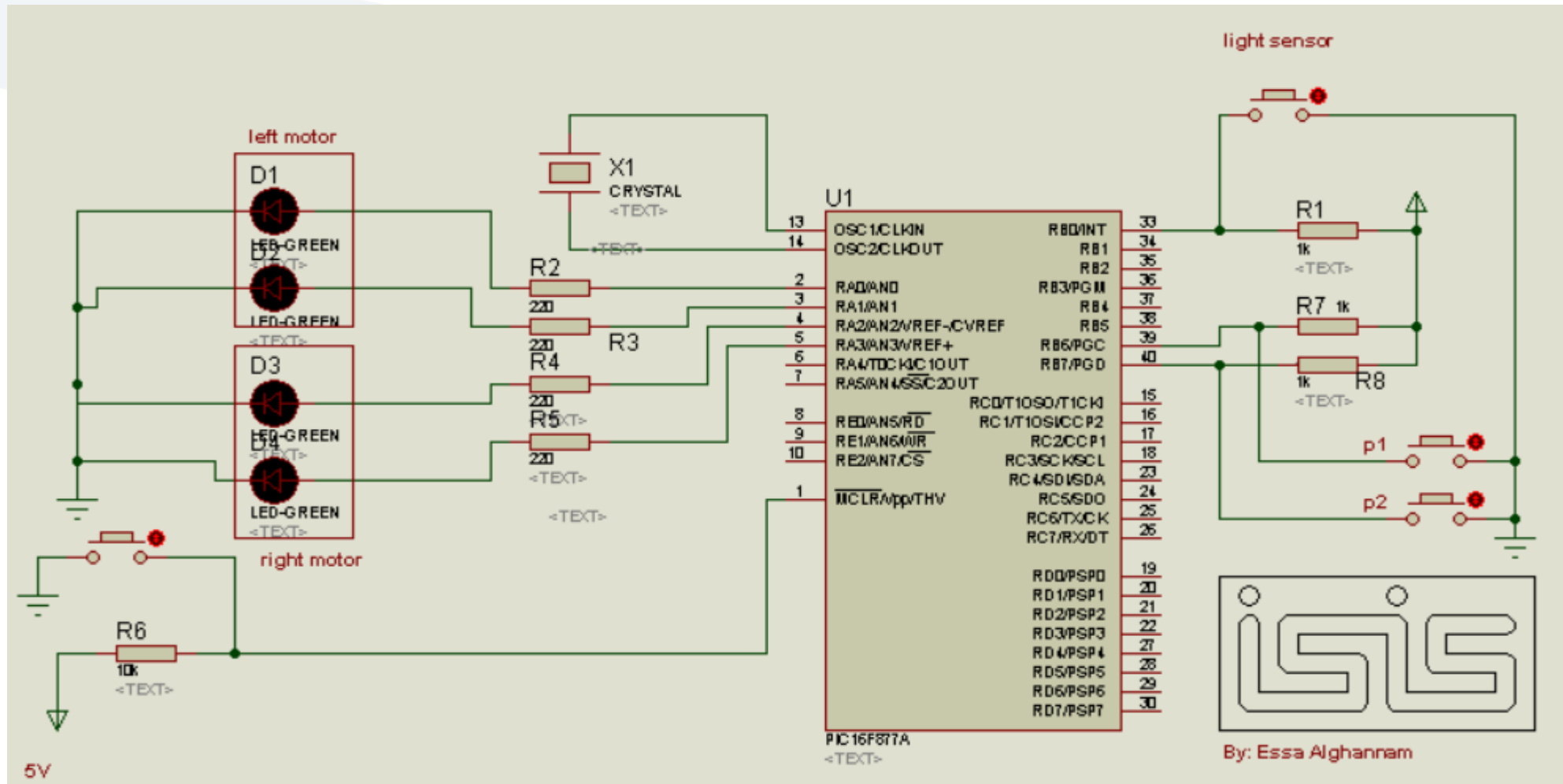
gate	LED1 - D0	LED2-D1
open	0	1
close	1	0
stop	0	0
stop	1	1

يتصل بالمتحكم الثاني:

- حساسين موقع بداية P1 ونهاية P2 افتراضهما من نوع Push Button موصولين إلى الأقطاب B6 و B7. القطبين مشدودين إلى المنطق العالي HIGH.
- حساس إضاءة افترضه من نوع Push Button موصول إلى القطب B0. القطب مشدود إلى المنطق العالي HIGH.
- محرك تيار مستمر للعجلة اليسار حيث تم التعبير عن هذا المحرك من خلال ديودين ضوئيين متصلين إلى القطبين A0 و A1.
- محرك تيار مستمر للعجلة اليمين حيث تم التعبير عن هذا المحرك من خلال ديودين ضوئيين متصلين إلى القطبين A2 و A3.
- الروبوت مقاد بمحركين مستمرين كما يلي:

حالة أقطاب المتحكم	Left Motor قطبين A0-A1		Right Motor قطبين A2-A3	
	LED1-A0	LED2-A1	LED11-A2	LED12-A3
STOP المحرك متوقف وكلا الليدين مطفأين	0	0	0	0
CW المحرك يدور مع عقارب الساعة أحد الليدين يعمل والآخر مطفئ	0	1	0	1
CCW المحرك يدور عكس عقارب الساعة وحالة الليدين المكافأة تعاكس الحالة السابقة	1	0	1	0
STOP المحرك متوقف وكل الليدين يعملان	1	1	1	1





1. عند وصول شخص إلى باب المستودع يضغط مفتاح **S0** من نوع **PUSH BUTTON** عدد محدد من المرات وليكن ثلاث حيث يمثل هذا الرقم رمز الدخول للمستودع.
2. عند كل ضغطة للمفتاح **S0** يظهر رقم الضغطة على الشاشتين المتصلتين مع الشريحة الأولى مثلا بعد أول ضغطة يظهر الرقم 01 على الشاشتين ثم 02 عند ثاني ضغطة وهكذا
3. عند ضغط **S1** وإذا كانت عدد مرات ضغط **S0** صحيحة يظهر **on** على الشاشتين وتعمل الإضاءة المتصلة مع الشريحة الأولى ويفتح باب المستودع لمدة خمس ثواني ثم يغلق. وإذا كان العدد خاطئا يتم إطفاء الشاشتين ونعود لنقطة البداية.
4. عند عمل الإضاءة يتحسس حساس إضاءة رقمي متصل إلى **B0** في المتحكم الثاني وتصبح حالة القطب منطوق منخفض **LOW** فيبدأ الروبوت بالتحرك وفق خط مستقيم ذهابا وإيابا من دون التفاف بين نقطتين **P1** و **P2** حيث تمثل كل منهما حساس رقمي يغير الروبوت اتجاه سيره عند وصوله إليها.
5. عند غياب الإضاءة داخل المستودع يتابع الروبوت طريقه نحو النقطة التي يسير إليها سواء ذهابا أو إيابا ويتوقف عندها حتى عودة الإضاءة.

افتراض وجود الروبوت في النقطة **P1** بداية ودوران محركاته **CW** يؤمن الحركة من **P1** إلى **P2**.
المطلوب: حاكمي دائرة المتحكم الثاني واكتب الكود البرمجي لكل متحكم باستخدام **PICC Compiler**.
ملاحظة أهمل أي حالة لم تذكر في المسألة و افترض ما تراه مناسباً لإنجاز الحل وفق الشروط المحددة مسبقاً.

Program Description Language (PDL) of PIC1

START

Configure PORTs pins status as output or input

Set output PORTs pins as Low

Set INT Variable COUNT = 0

DEFINE MATRIX

DEFINE DISPLAY

DEFINE GATE

DO FOREVER

IF A0-S0 IS PRESSED AND RELEASED THEN

IF COUNT <=10

INCREASE COUNT BY 1

ELSE CLEAR COUNT

ENDIF

ENDIF

DISPLAY COUNT

IF A1-S1 IS PRESSED AND RELEASED THEN

IF COUNT ==3

TURN LIGHT

OPEN Gate and close it

else

RESET

ENDIF

ENDIF

ENDDO

END

<https://manara.edu.sy/>

Code of pic1

```
void main()
{int count=0;
  io_set();
  output_b(0x00);
  output_c(0x00);
  output_d(0x00);
  output_e(0x00);

  while (1)
  {
    if(!input(pin_a0))
    {
      if(count>=10)
      count=0;
      else
      {count++;seg=1;}
      while(!input(pin_a0));
    }
    display(count,seg);
    if(!input(pin_a1))
    {
      while(!input(pin_a1));
      if(count==3)
      {
        output_c(0x0f);
        seg=2;
        gate();
      }
    }
    else
    {seg=0;count=0;}
  }
}
```

```
# include <16f877a.h>
# use delay (clock=4000000)
```

```
void io_set()
{set_tris_b(0x00);
 set_tris_c(0x00);
 set_tris_d(0x00);
 set_tris_e(0x00);
 set_tris_a(0xff); //0x03 a1&e0->in
}
```

```
int disp(int i)
void display(int i, int1 seg)
```



```
void display(int i, int1 seg)
{
    int digit;
    if(seg==0)
    {
        output_b(0x00);
        output_e(0x00);
        output_c(0x00);
    }
    if(seg==1)
    {
        digit=i%10;
        output_high(pin_e1);
        output_low(pin_e2);
        output_b(disp(digit));
        delay_ms(10);
        digit=i/10;
        output_high(pin_e2);
        output_low(pin_e1);
        output_b(disp(digit));
        delay_ms(10);
    }
    if(seg==2)
    {
        output_high(pin_e1);
        output_low(pin_e2);
        output_b(disp(11)); //n
        delay_ms(10);
        output_high(pin_e2);
        output_low(pin_e1);
        output_b(disp(10)); //o
        delay_ms(10);
    }
}
```

```
int seg=0;
```

```
int disp(int i)
{int val[12]={0x3f,0x06,0x5b,0x4f,0x66,0x6d,0x7d,0x07,0x7f,0x6f,0x5c,0x54};
return val[i];
}
```

```
void gate()
{
    output_high(pin_e2);
    output_low(pin_e1);
    output_b(disp(10));
    delay_ms(200);
    output_high(pin_e1);
    output_low(pin_e2);
    output_b(disp(11));
    output_d(0x02);
    delay_ms(5000);
    output_d(0x00);
    delay_ms(200);
    output_d(0x01);
    delay_ms(5000);
    output_d(0x00);
    delay_ms(200);
}
```

Code of pic2

Program Description Language (PDL) of PIC2



START

Configure PORTs pins status as output or input

Set output PORT A pins as Low

Set Boolean Variable DIR1 = 0

DO FOREVER

IF THERE IS LIGHT THEN

DO FOREVER

IF DIR1 = 0 THEN

GO FROM P1 TO P2

DIR1 = 1

WAIT UNTIL reach p2

IF THERE IS no LIGHT THEN BREAK

ENDIF

IF DIR1 = 1 THEN

GO FROM P2 TO P1

DIR1 = 0

WAIT UNTIL reach p1

IF THERE IS no LIGHT THEN BREAK

ENDIF

ENDDO

ENDIF

Set output PORT A pins as Low

ENDDO

END

```
# include <16f877a.h>
# use delay (clock=4000000)
# byte porta=5
# byte portb=6

void io_set()
{set_tris_b(0xff); //0xc1 //0b11000001
 set_tris_a(0x00);
}

void main()
{int1 dir =0;
 io_set();
 output_a(0x00);
 while(true)
 {if(!input(pin_b0))
 {
 while(true)
 {
 if (dir==0)
 {
 output_a(0x0A); // p1 to p2
 dir=1;
 while(input(pin_b7)); //not reaching p2
 if (input(pin_b0)) break; // no light
 }
 if (dir==1)
 {
 output_a(0x05); // p2 to p1
 dir=0;
 while(input(pin_b6)); //not reaching p1
 if (input(pin_b0)) break; // no light
 }
 }
 }
 }
 }
 }
 }

} //if (!input(pin_b0))
output_a(0x00);
} //while (true)
} //main
```

انتهت المحاضرة