

تطبيقات ميكاترونك -1-

Lecture No. 7

- Warehouse and WMR

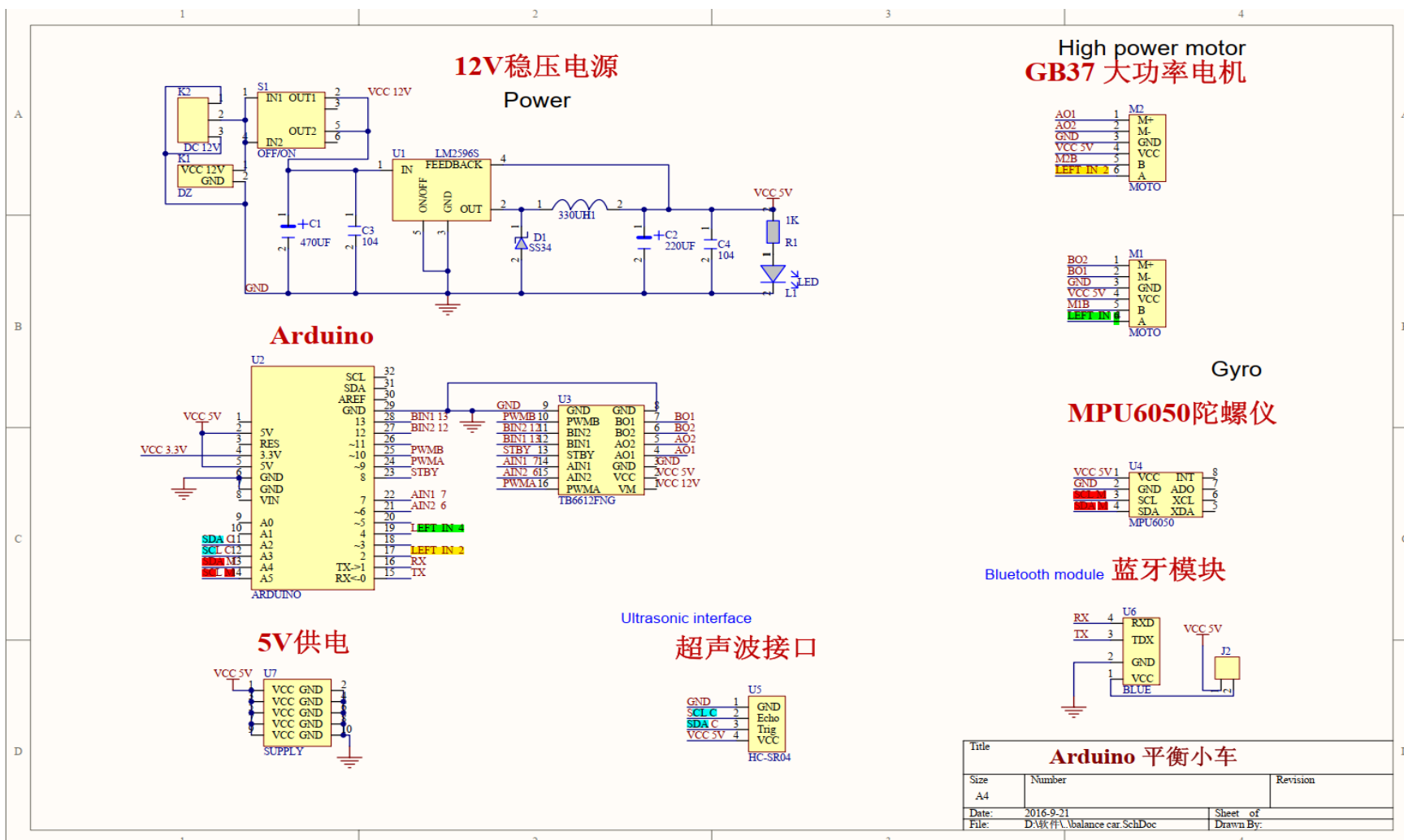
روبوت و أنظمة ذكية - سنة ثالثة

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

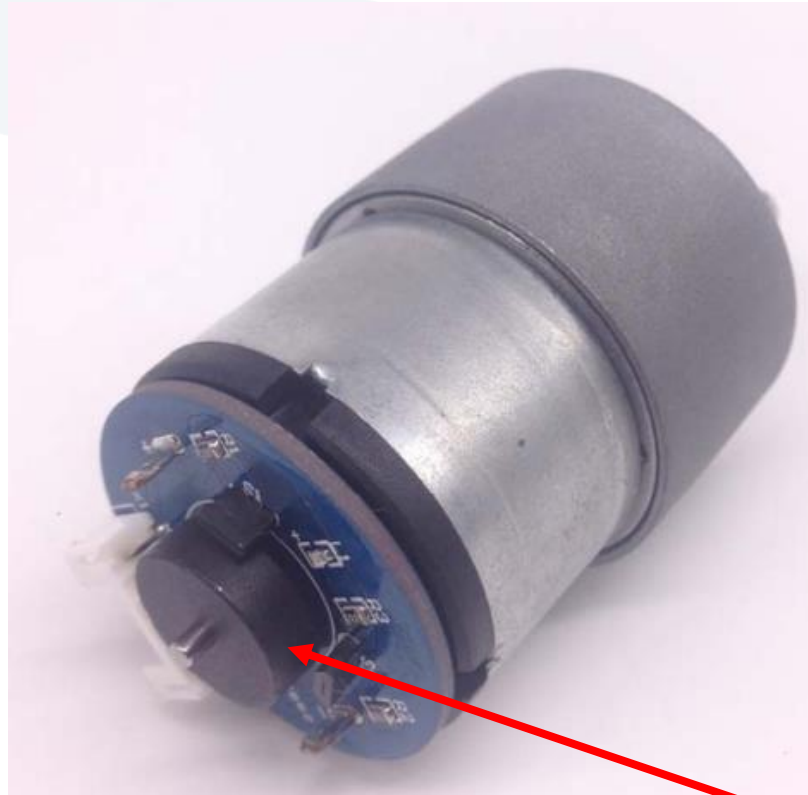
2023-2024

الانضمام الى CLASSROOM عبر الرابط التالي

<https://classroom.google.com/c/NjM5MDM5NkzNzM2?cjc=6xp3ajs>



GB37 DC gear motor



Rated voltage: 12V

gearbox Shaft Speed : 110RPM

Idle current: 250mA

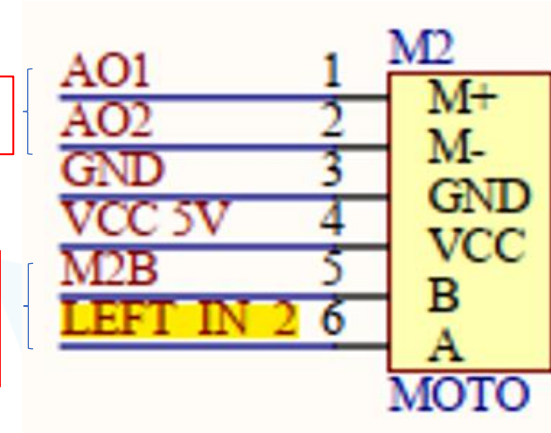
Power: 4.8w

Maximum locked rotor current: 6.5A

Reduction box length: 22mm

To driver TB6612FNG

A and B sensor
phases



جهاز استشعار حساس هال بـ 13 خط ترميز مغناطيسي وإخراج ثنائي الطور.

motor with 30:1 reduction ratio. Motor speed is $30 \times 110 = 3300$ RPM

the number of pulses can reach $30 \times 13 \times 2 = 780$. Single phase can also reach 390 pulses for each turn of the **gearbox** shaft.

- 1--- --Motor power cord AO1
- 2--- --Motor power cord AO2
- 3--- --Sensor signal line Negative GND
- 4--- --Sensor Positive 5V VCC
- 5--- --Sensor signal line B phase
- 6--- --Sensor signal line A phase

TB6612FNG Dual Motor Driver carrier

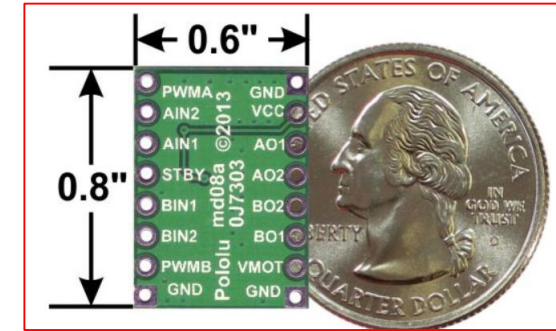
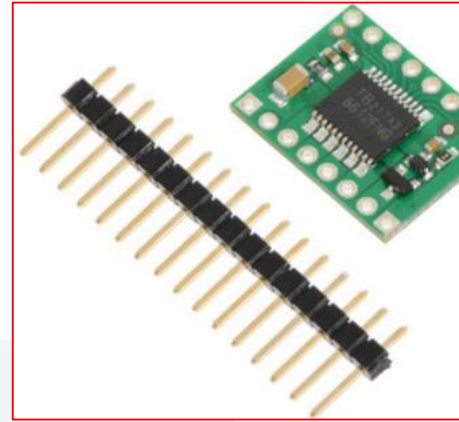


<https://www.pololu.com/product/713>

- Dual-H-bridge motor driver: can drive two DC motors or one bipolar stepper motor
- **Maximum PWM frequency: 100 kHz**

GND
VCC
AO1
AO2
BO2
BO1
VMOT
GND

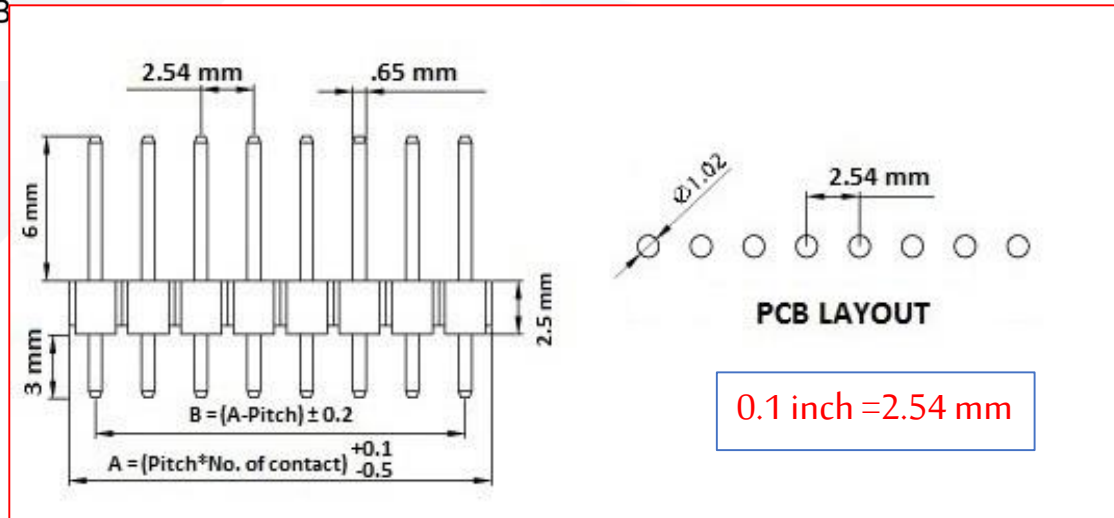
PWMA
AIN2
AIN1
STBY
BIN1
BIN2
PWMB
GND



Characteristics	Symbol	Rating	Unit	Remarks
Supply voltage	V _M	15	V	
	V _{CC}	6		
Input voltage	V _{IN}	-0.2~6	V	IN1, IN2, STBY, PWM pins
Output voltage	V _{out}	15	V	O1, O2 pins
Output current	I _{out}	1.2	A	Per 1ch
	I _{out} (peak)	2		tw=20ms Continuous pulse, Duty ≤ 20%
		3.2		tw=10ms Single pulse

Included hardware

A 1×16-pin breakaway [0.1" male header strip](#) is included with the TB6612FNG motor driver carrier. This strip can optionally be soldered to the carrier board so that it can be used with perfboards, [solderless breadboards](#), or [0.1" female connectors](#). (The headers might ship as two 1×8 pieces or as a single 1×16 piece that can be broken in half.)

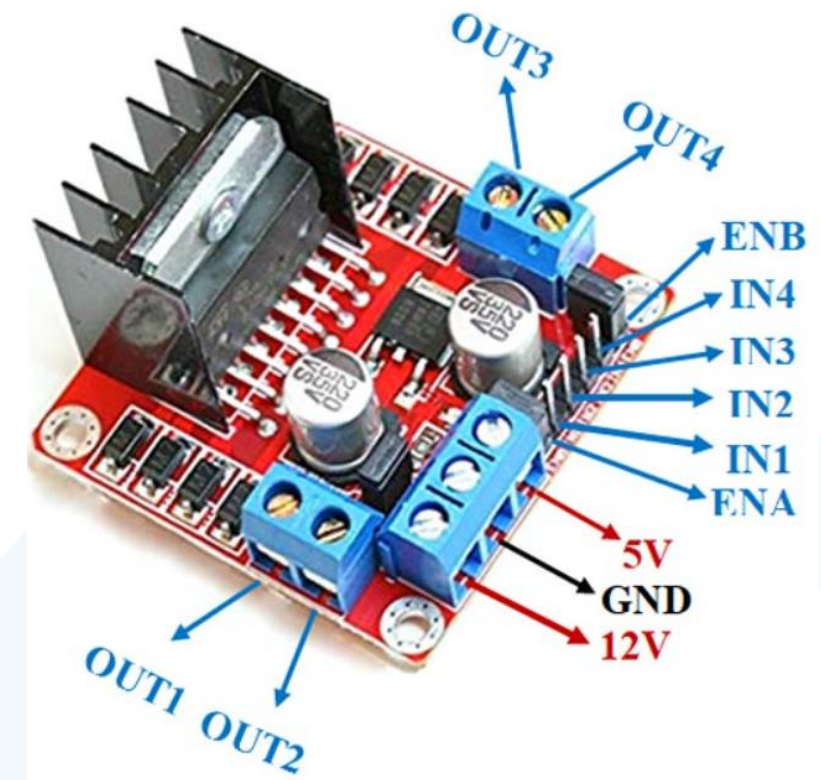
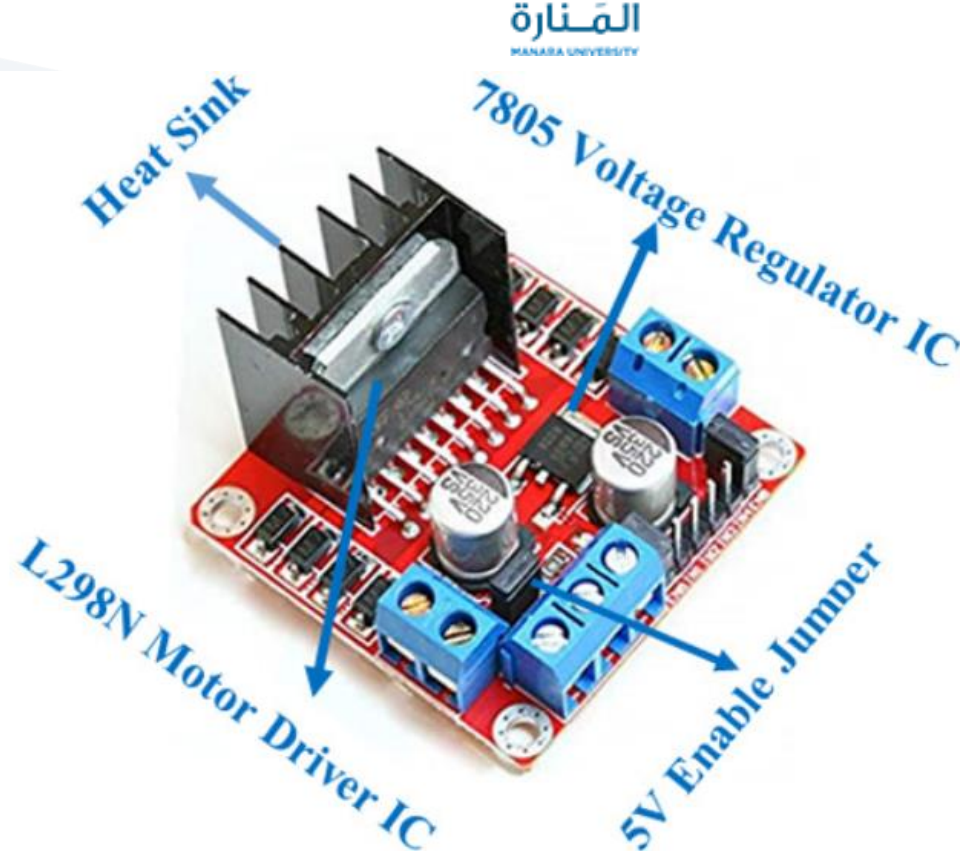


L298 Dual Motor Driver carrier



Motor driver:	L298N
Motor channels:	2
Maximum operating voltage:	50 V
Peak output current per channel:	2 A
logic voltage:	4.5-7 V

Frequency TYPICAL 25kHz - MAX40 KHz

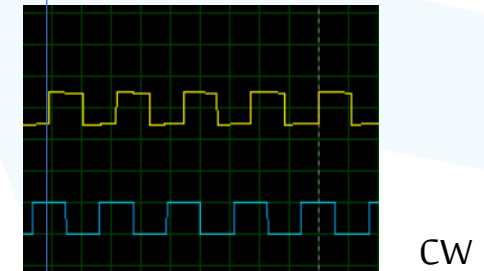
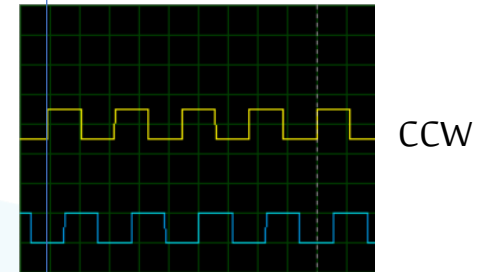
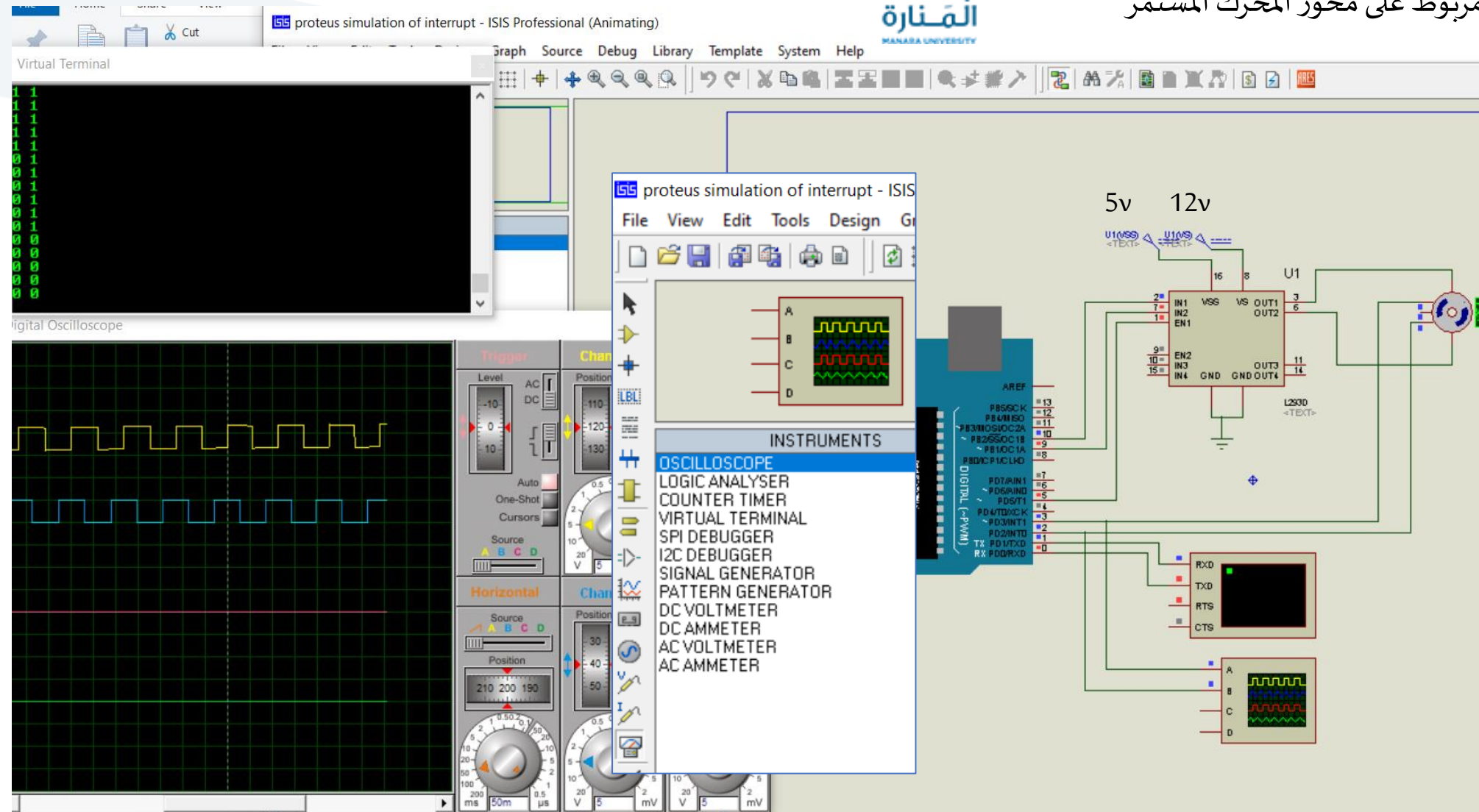


78M05 Voltage regulator will be enabled only when the jumper is placed. When the power supply is less than or equal to 12V, then the internal circuitry will be powered by the voltage regulator and the 5V pin can be used as an output pin to power the microcontroller.

The jumper should not be placed when the power supply is greater than 12V and separate 5V should be given through 5V terminal to power the internal circuitry.

DC MOTOR USING L298 AND ROTARY ENCODER

قراءة إشارة مشفر (انكودر) دوراني مربوط على محور المحرك المستمر



لحظة حدوث جبهة صاعدة
على القناة الأولى نفحص
القناة الثانية فيما اذا كانت
LOW او HIGH

```
#define A 2 // pin2 of the Arduino
#define B 3 // Pin3 of the Arduino

int A_DATA;
int B_DATA;
int in1 = 9;
int in2 = 10;
int EN = 5;

void setup() {
    Serial.begin(9600); // Activates Serial communication
    pinMode(A, INPUT); // sets pin2 as the input
    pinMode(B, INPUT); // sets pin3 as the input
    pinMode(in1, OUTPUT); |
    pinMode(in2, OUTPUT);
    pinMode(EN, OUTPUT);
    digitalWrite(in1, HIGH);
    digitalWrite(in2, LOW );
    digitalWrite(EN, HIGH);
}
```

```
void loop() {
    A_DATA = digitalRead(A);
    // We simply read Pin2 of the Arduino and store the result in variable A_DATA
    B_DATA = digitalRead(B);
    // We simply read Pin3 of the Arduino and store the result in variable b
    Serial.print(A_DATA);
    Serial.print(" ");
    Serial.print(B_DATA);
    Serial.println();
}
```

تطبيق 1

- لدينا مستودع يراد التحكم بالدخول له وإضاءته من خلال شريحة متحكم أول arduino1 نوع MEGA
- كما يحوي المستودع في الداخل روبوت تفاضلي بعجلتين متصلتين بمحركين تيار مستمر. ويقاد من قبل متحكم ثاني arduino2 نوع UNO.

ملاحظة: لا يوجد اتصال سلكي بين المتحكمين الأول والثاني.

يتصل بالمتحكم الأول:

- مفتاحين s0 و s1 من نوع Push Button على الأقطاب D2 D3 على التوالي بعد شدهما إلى المنطق العالي HIGH
- شاشات 7seg من نوع مهيّط مشترك عدد اثنين من دون decoder متصلين على التفرع إلى d7-d13. (المهابط المشتركة متصلة إلى الأقطاب d5,d6 كما يلي d6-second 7seg , d5-first 7seg
- أربع ليدات متصلة إلى d24
- يفتح ويغلق باب المستودع من خلال محرك تيار مستمر GATE MOTOR (متصل عبردارة قيادة من نوع L298 عبرالأقطاب D42-D44)

D44 IS ENABLE

gate	D42	D43
open	0	1
close	1	0
stop	0	0
stop	1	1

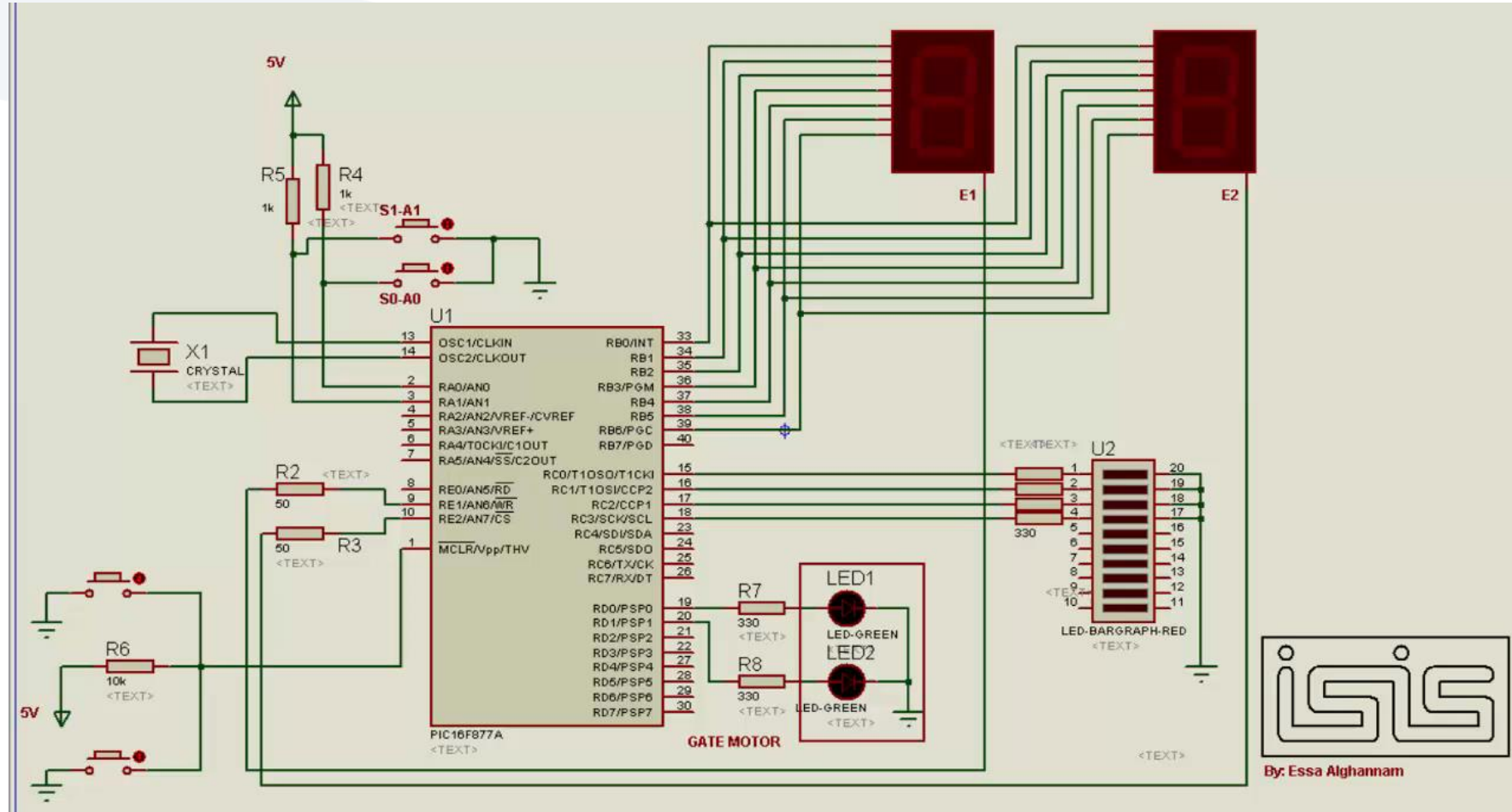
يتصل بالمتحكم الثاني:

- حساسين موقع بداية P1 ونهاية P2 افتراضهما من نوع Push Button موصولين إلى الأقطاب D2 و D3. القطبين مشدودين إلى المنطق العالي HIGH.
- حساس إضاءة افتراضه من نوع Push Button موصول إلى القطب A0. القطب مشدود إلى المنطق العالي HIGH.
- محرك تيار مستمر للعجلة اليسار متصل إلى القطبين D9 و D8 عبردارة قيادة D10 IS ENABLE L298
- محرك تيار مستمر للعجلة اليمين متصل إلى القطبين D12 و D13 عبردارة قيادة D11 IS ENABLE L298
- الروبوت مقاد بمحركين مستمرين كما يلي:

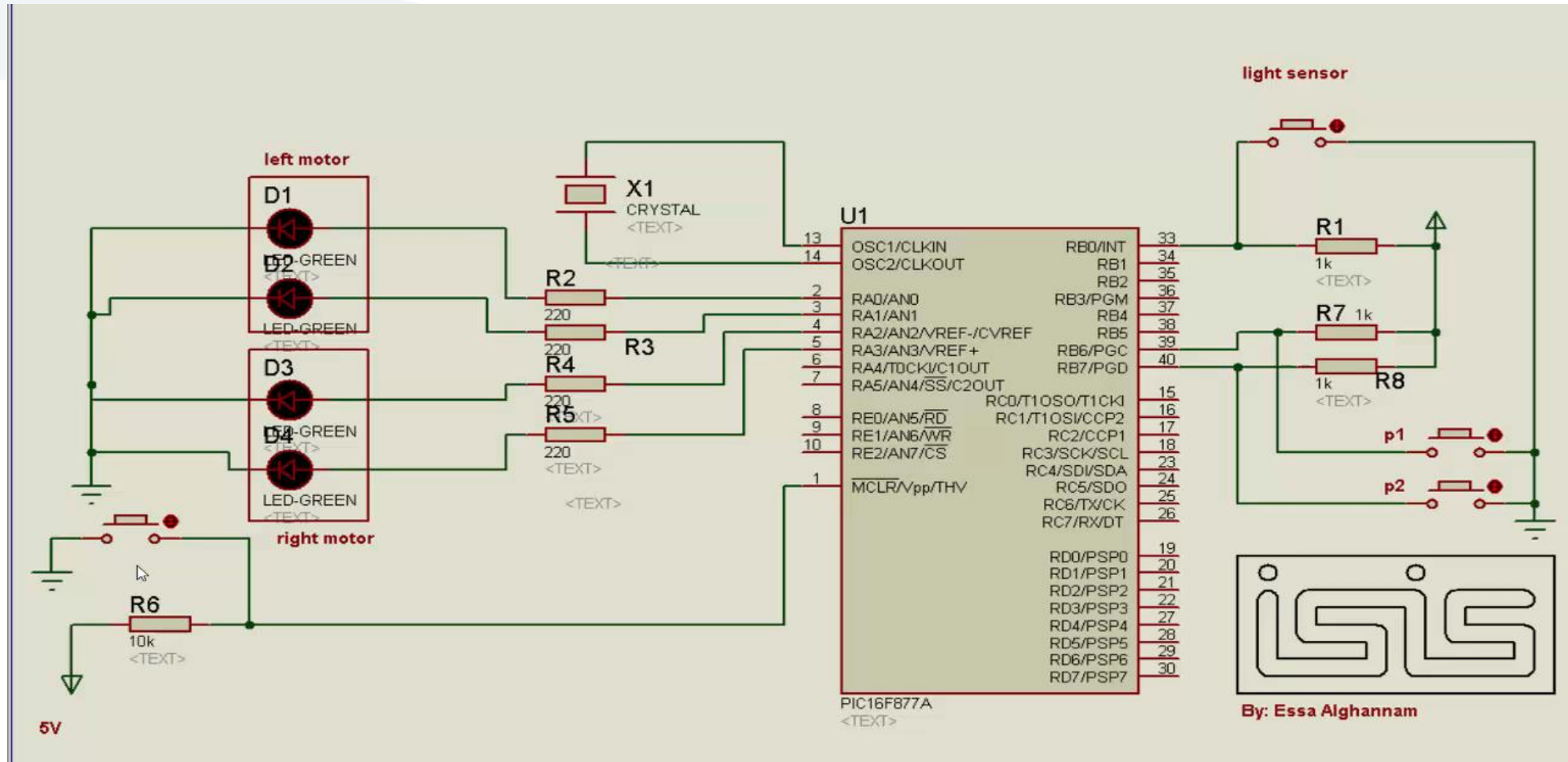
حالة أقطاب المتحكم	Left Motor		Right Motor	
	D8	D9	D12	D13
STOP المحرك متوقف وكلا الليدين مطفأين	0	0	0	0
CW المحرك يدور مع عقارب الساعة أحد الليدين يعمل والآخر مطفئ	0	1	0	1
CCW المحرك يدور عكس عقارب الساعة وحالة الليدين المكافأة تعاكس الحالة السابقة	1	0	1	0
STOP المحرك متوقف وكل الليدين يعملان	1	1	1	1

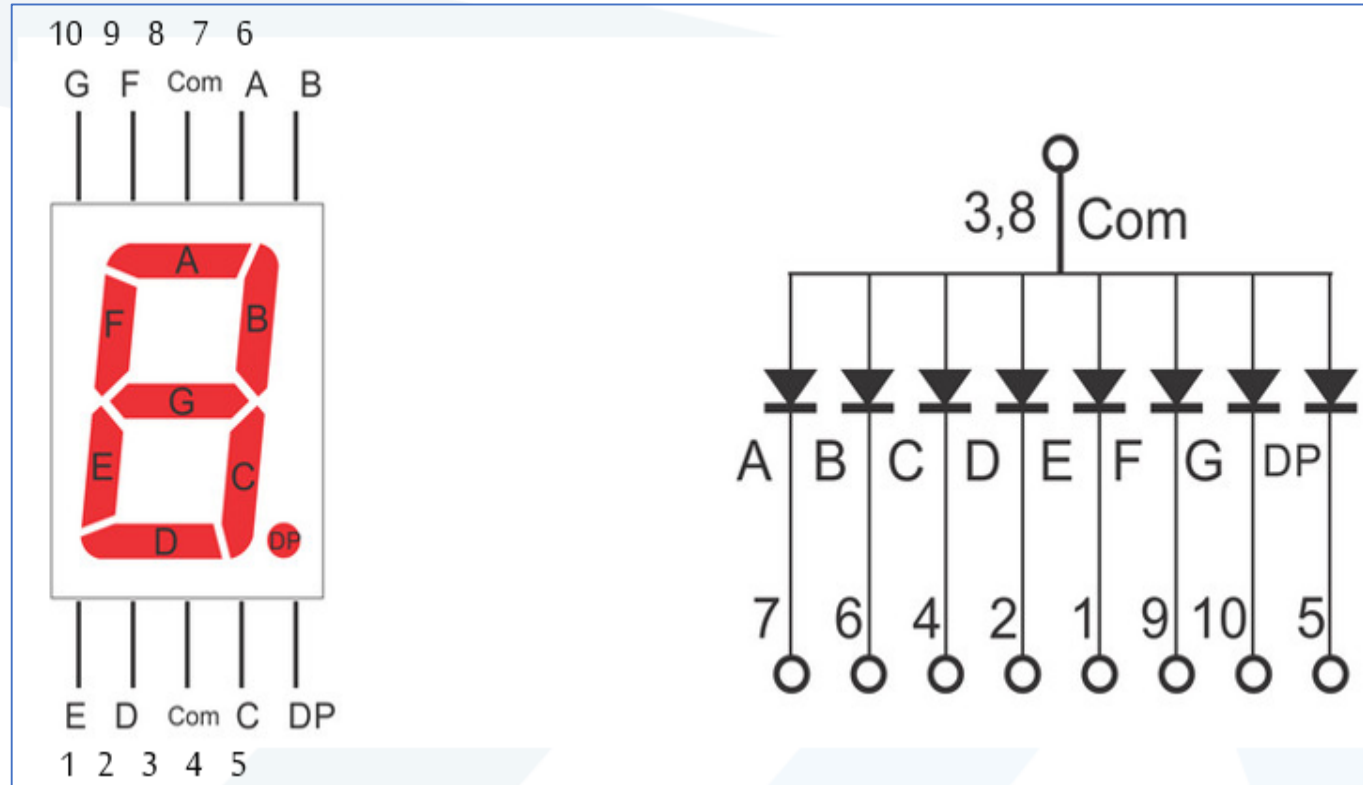
1. عند وصول شخص إلى باب المستودع يضغط مفتاح **S0** من نوع **PUSH BUTTON** عدد محدد من المرات وليكن ثلاث حيث يمثل هذا الرقم رمز الدخول للمستودع.
2. عند كل ضغطة للمفتاح **S0** يظهر رقم الضغطة على الشاشتين المتصلتين مع الشريحة الأولى مثلاً بعد أول ضغطة يظهر الرقم 01 على الشاشتين ثم 02 عند ثاني ضغطة وهكذا
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 خاطئاً يتم إطفاء الشاشتين ونعود لنقطة البداية.
4. عند عمل الإضاءة يتحسس حساس إضاءة رقمي في المتحكم الثاني وتصبح حالة القطب منطق منخفض **LOW** فيبدأ الروبوت بالتحرك وفق خط مستقيم ذهاباً وإياباً 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 ذهاباً أو إياباً ويتوقف عندها حتى عودة الإضاءة.

افترض 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
ملاحظة أهمل أي حالة لم تذكر في المسألة و افترض ما تراه مناسباً لإنجاز الحل وفق الشروط المحددة مسبقاً.



دائرة المحاكاة للروبوت التفاضلي







Program Description Language (PDL) of PIC1

Code 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

```

```

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>
#define _XTAL_FREQ 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