جَـامعة
المَـنارة
MANARA UNIVERSITY

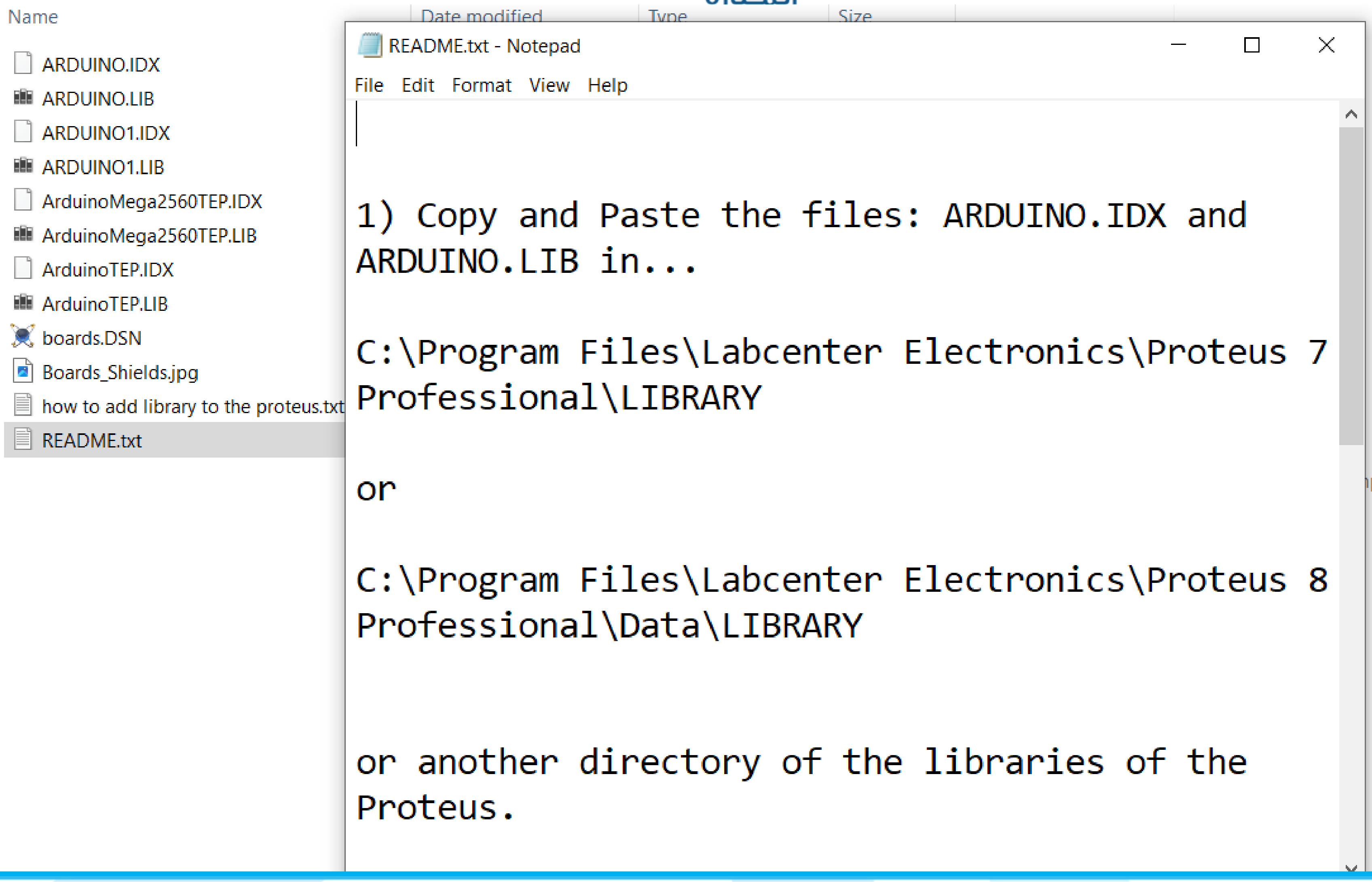
متحكمات صغرية ونظم مضمنة

Lecture No. 2+3

ميكاترونيكس- سنة رابعة

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

2025



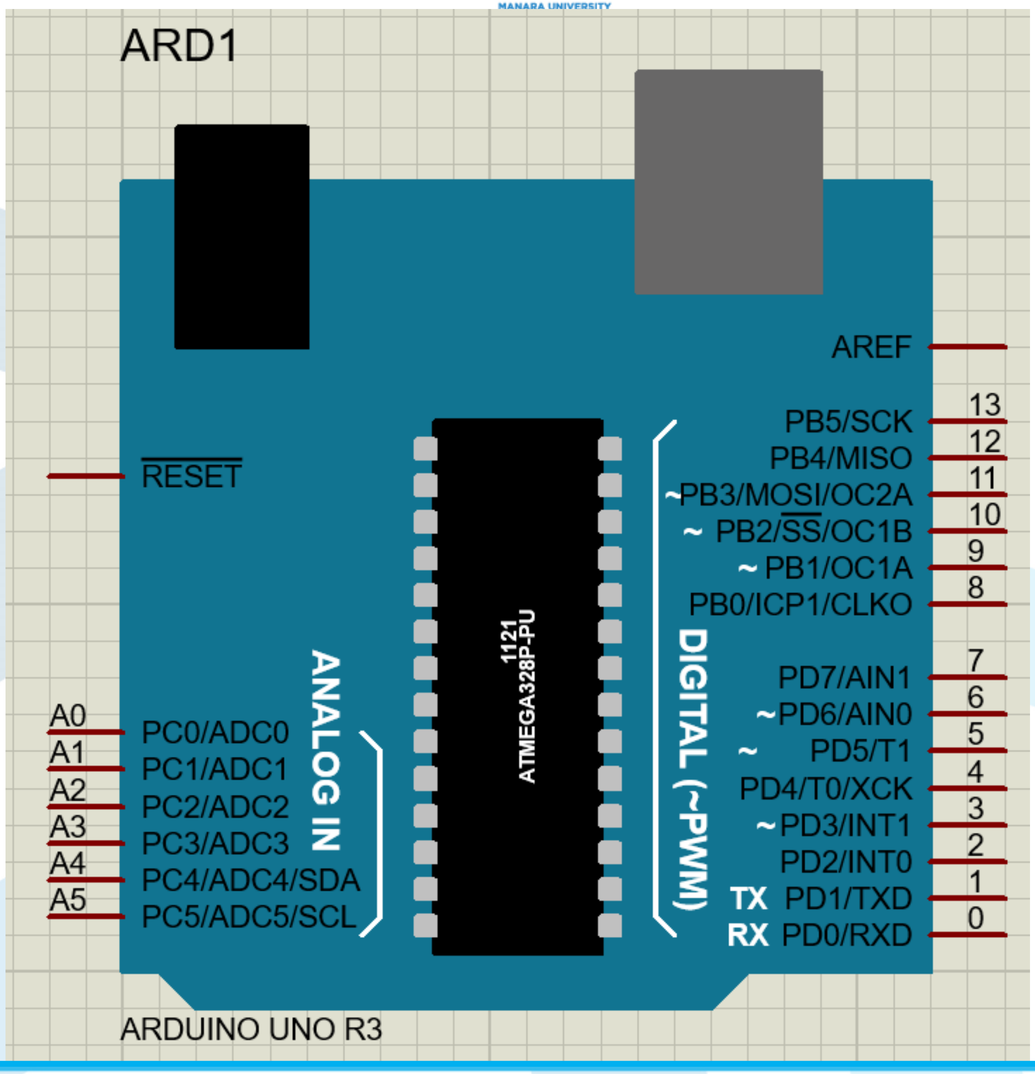
1) Copy and Paste the files: ARDUINO.IDX and ARDUINO.LIB in...

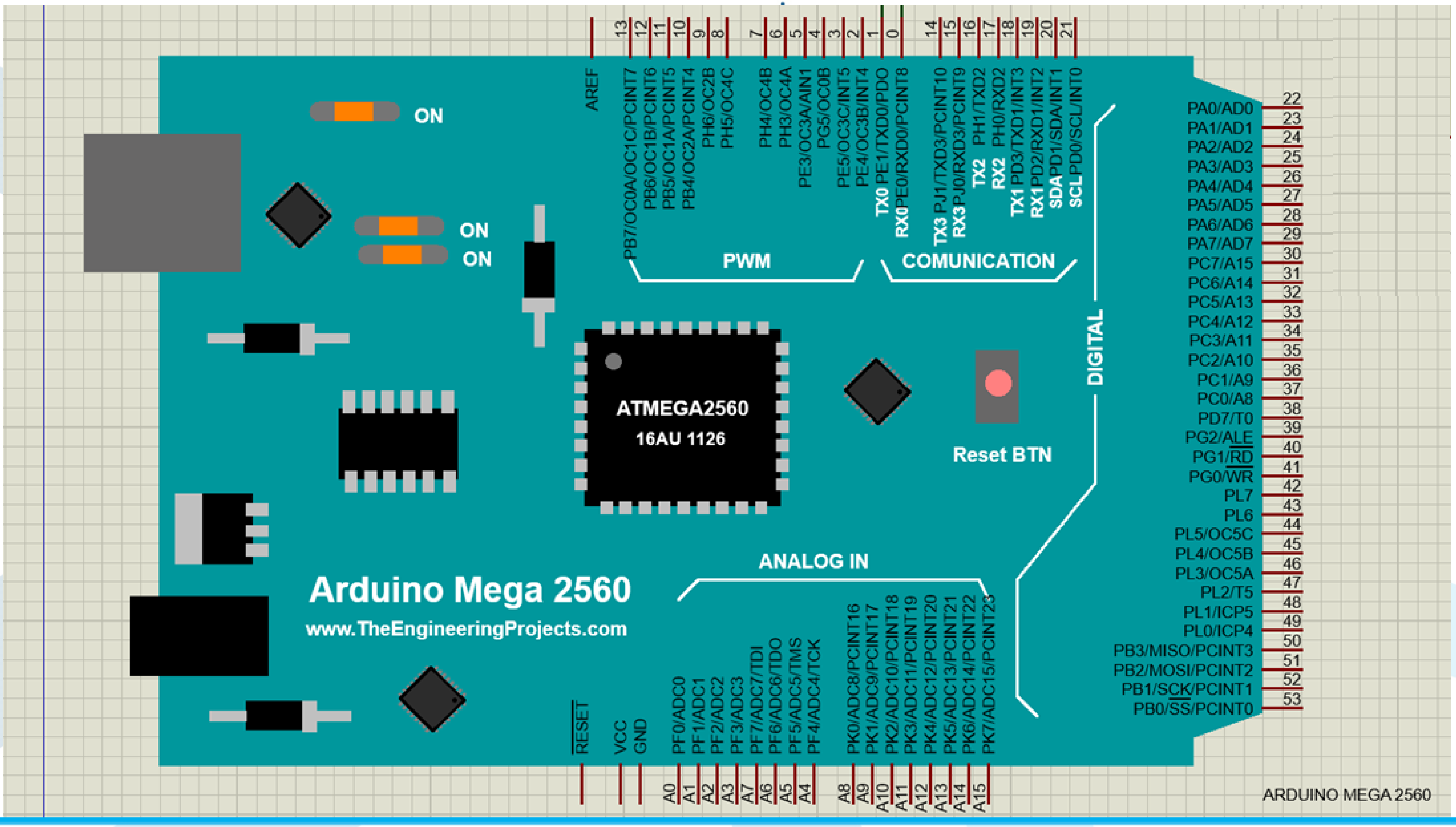
C:\Program Files\Labcenter Electronics\Proteus 7 Professional\LIBRARY

or

C:\Program Files\Labcenter Electronics\Proteus 8 Professional\Data\LIBRARY

or another directory of the libraries of the Proteus.

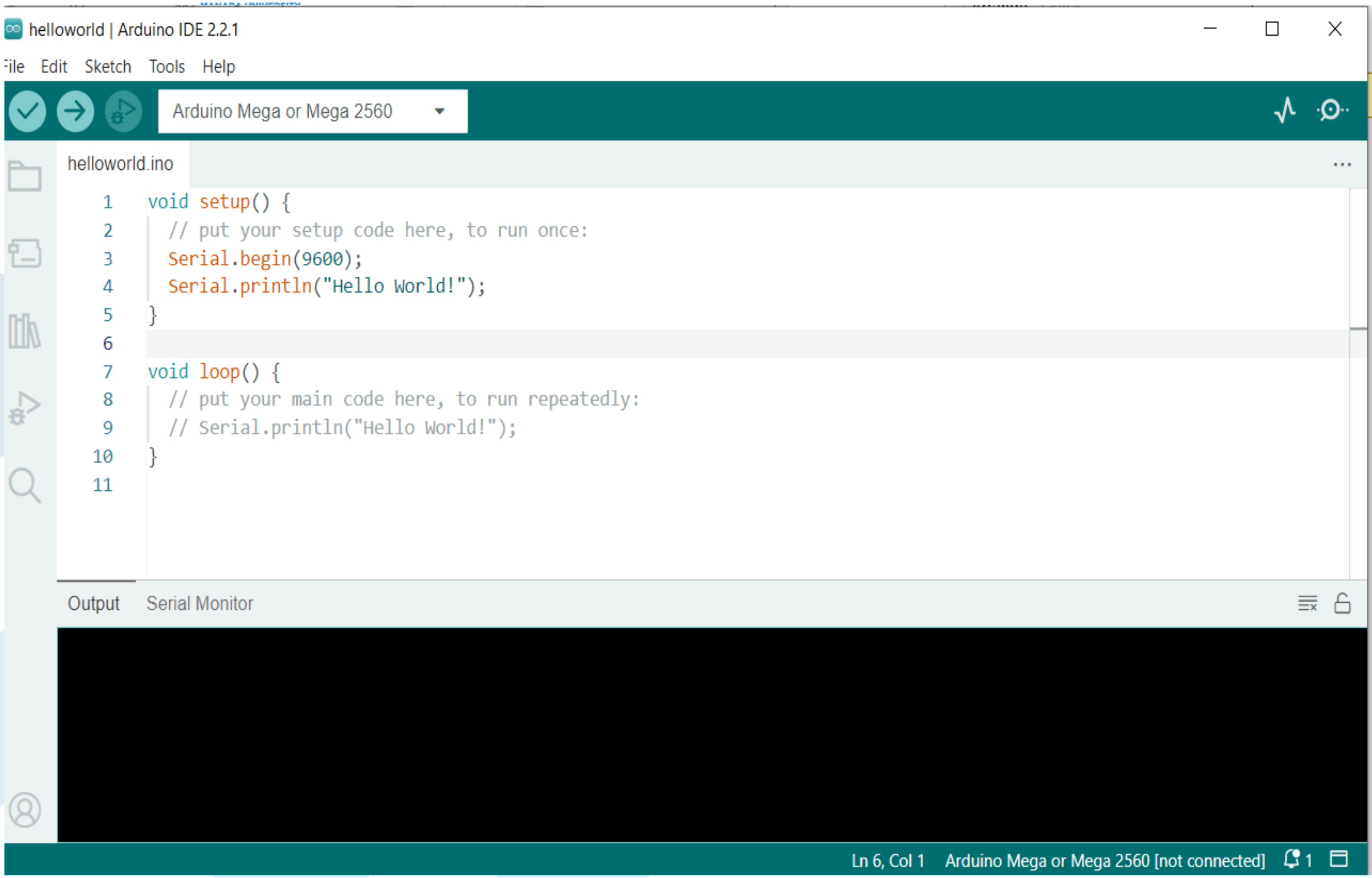

ARD1
AREF
13 PB5/SCK
12 PB4/MISO
11 ~PB3/MOSI/OC2A
10 ~ PB2/SS/OC1B
9 ~ PB1/OC1A
8 PB0/ICP1/CLKO
7 PD7/AIN1
6 ~PD6/AIN0
5 ~ PD5/T1
4 PD4/T0/XCK
3 ~PD3/INT1
2 PD2/INT0
1 TX PD1/TXD
0 RX PD0/RXD
DIGITAL (~PWM)
RESET
A0 PC0/ADC0
A1 PC1/ADC1
A2 PC2/ADC2
A3 PC3/ADC3
A4 PC4/ADC4/SDA
A5 PC5/ADC5/SCL
ANALOG IN
1121
ATMEGA328P-PU
ARDUINO UNO R3


AREF
PB7/OC0A/OC1C/PCINT7
PB6/OC1B/PCINT6
PB5/OC1A/PCINT5
PB4/OC2A/PCINT4
PH6/OC2B
PH5/OC4C
PH4/OC4B
PH3/OC4A
PE3/OC3A/AIN1
PG5/OC0B
PE5/OC3C/INT5
PE4/OC3B/INT4
TX0 PE1/TXD0/PDO
RX0 PE0/RXD0/PCINT8
TX3 PJ1/TXD3/PCINT10
RX3 PJ0/RXD3/PCINT9
TX2 PH1/TXD2
RX2 PH0/RXD2
TX1 PD3/TXD1/INT3
RX1 PD2/RXD1/INT2
SDA PD1/SDA/INT1
SCL PD0/SCL/INT0
13 12 11 10 9 8 7 6 5 4 3 2 1 0 14 15 16 17 18 19 20 21
PWM
COMUNICATION
DIGITAL
ON
ON
ON
ATMEGA2560
16AU 1126
Reset BTN
Arduino Mega 2560
www.TheEngineeringProjects.com
ANALOG IN
PA0/AD0 22
PA1/AD1 23
PA2/AD2 24
PA3/AD3 25
PA4/AD4 26
PA5/AD5 27
PA6/AD6 28
PA7/AD7 29
PC7/A15 30
PC6/A14 31
PC5/A13 32
PC4/A12 33
PC3/A11 34
PC2/A10 35
PC1/A9 36
PC0/A8 37
PD7/T0 38
PG2/ALE 39
PG1/RD 40
PG0/WR 41
PL7 42
PL6 43
PL5/OC5C 44
PL4/OC5B 45
PL3/OC5A 46
PL2/T5 47
PL1/ICP5 48
PL0/ICP4 49
PB3/MISO/PCINT3 50
PB2/MOSI/PCINT2 51
PB1/SCK/PCINT1 52
PB0/SS/PCINT0 53
RESET
VCC
GND
PF0/ADC0
PF1/ADC1
PF2/ADC2
PF3/ADC3
PF7/ADC7/TDI
PF6/ADC6/TDO
PF5/ADC5/TMS
PF4/ADC4/TCK
PK0/ADC8/PCINT16
PK1/ADC9/PCINT17
PK2/ADC10/PCINT18
PK3/ADC11/PCINT19
PK4/ADC12/PCINT20
PK5/ADC13/PCINT21
PK6/ADC14/PCINT22
PK7/ADC15/PCINT23
A0 A1 A2 A3 A7 A6 A5 A4 A8 A9 A10 A11 A12 A13 A14 A15
ARDUINO MEGA 2560

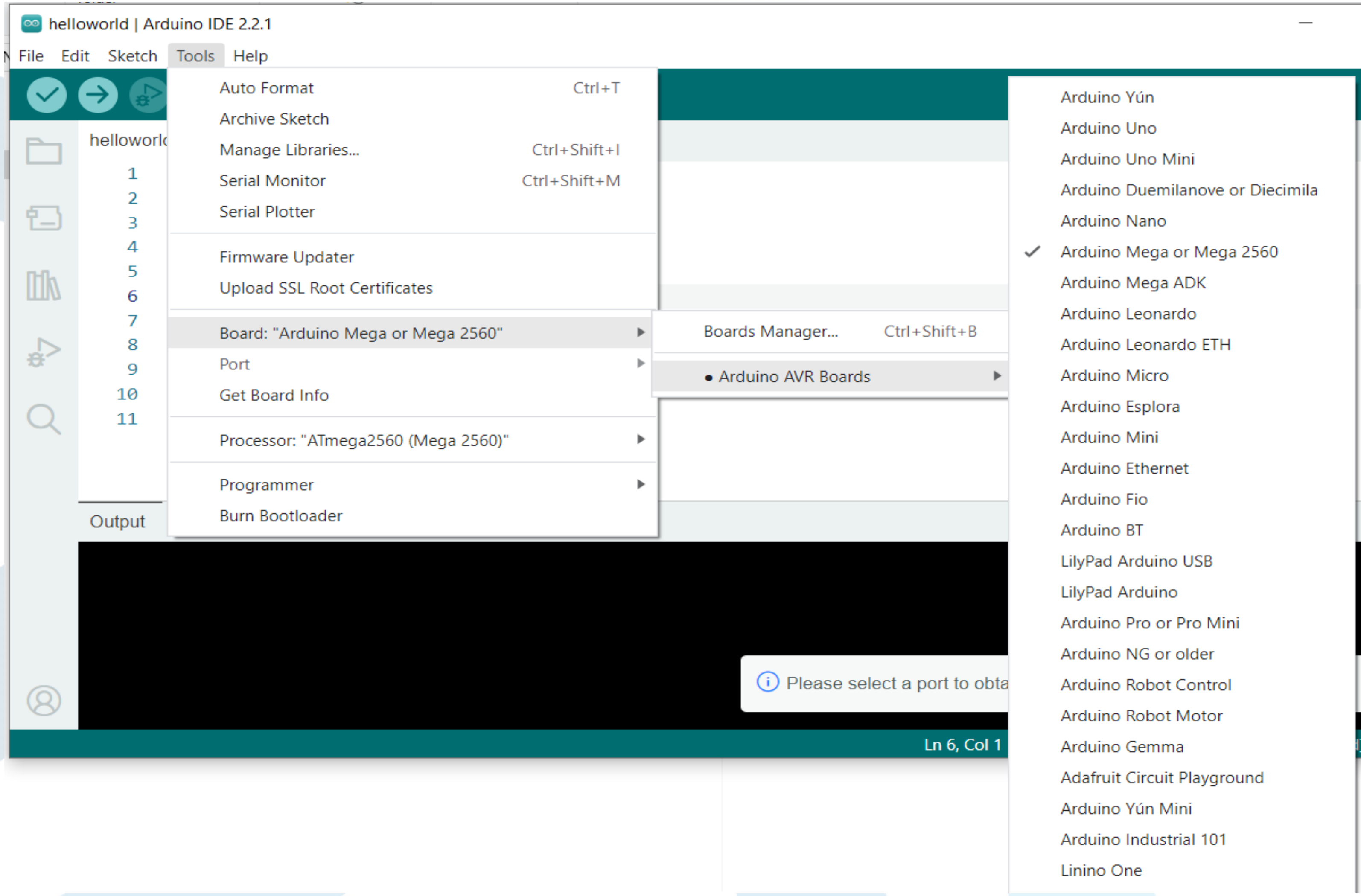
Example 1

Code.ino

```
void setup() {
  // put your setup code here, to run once:
  Serial.begin(9600);
  Serial.println("Hello World!");
}

void loop() {
  // put your main code here, to run repeatedly:
  // Serial.println("Hello World!");
}
```


helloworld | Arduino IDE 2.2.1
File Edit Sketch Tools Help
Arduino Mega or Mega 2560
helloworld.ino
Output
Serial Monitor
Ln 6, Col 1
Arduino Mega or Mega 2560 [not connected]
1

```
void setup() {
  // put your setup code here, to run once:
  Serial.begin(9600);
  Serial.println("Hello World!");
}

void loop() {
  // put your main code here, to run repeatedly:
  // Serial.println("Hello World!");
}

```



helloworld | Arduino IDE 2.2.1

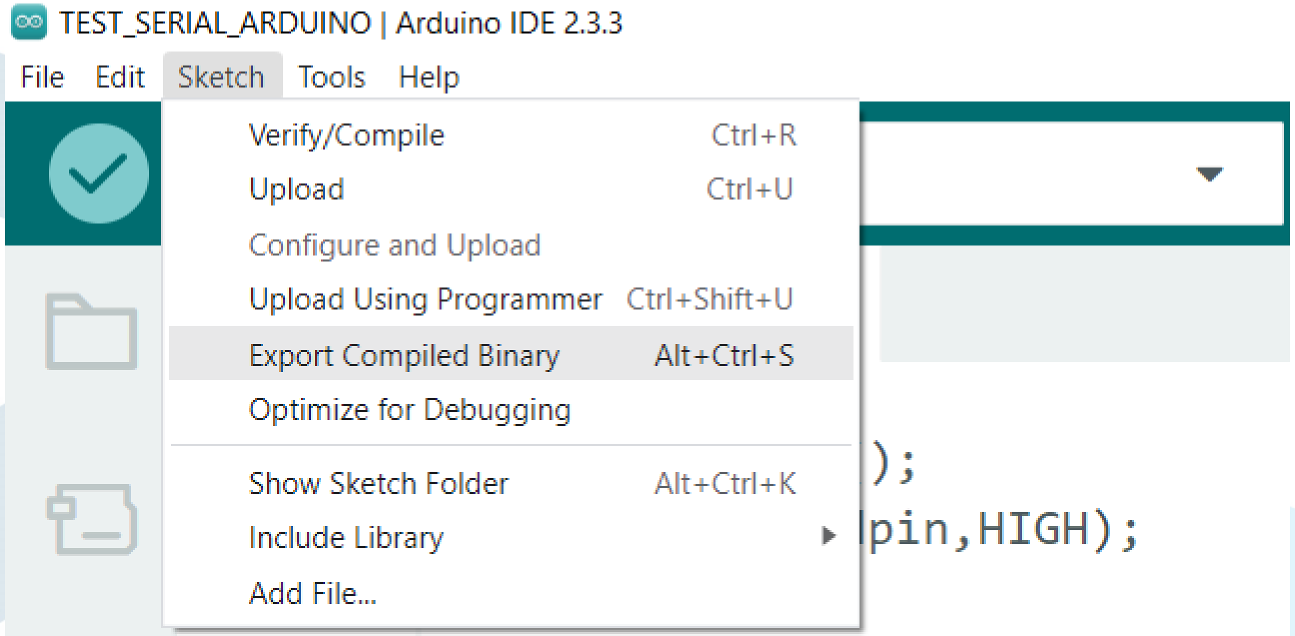
Board: "Arduino Mega or Mega 2560"

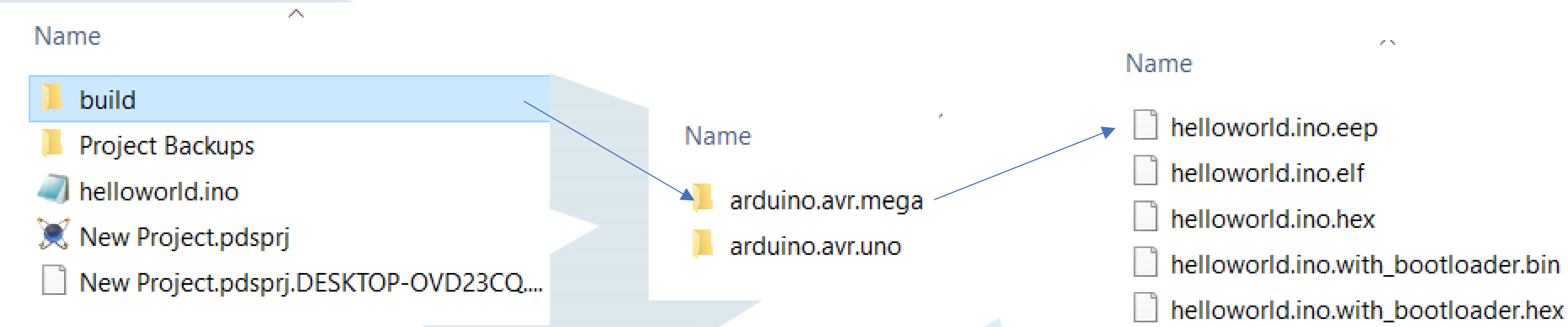
Processor: "ATmega2560 (Mega 2560)"

✓ Arduino Mega or Mega 2560

Please select a port to obta

Ln 6, Col 1


TEST_SERIAL_ARDUINO | Arduino IDE 2.3.3
File
Edit
Sketch
Tools
Help
Verify/Compile Ctrl+R
Upload Ctrl+U
Configure and Upload
Upload Using Programmer Ctrl+Shift+U
Export Compiled Binary Alt+Ctrl+S
Optimize for Debugging
Show Sketch Folder Alt+Ctrl+K
Include Library
Add File...
);
pin,HIGH);



Name

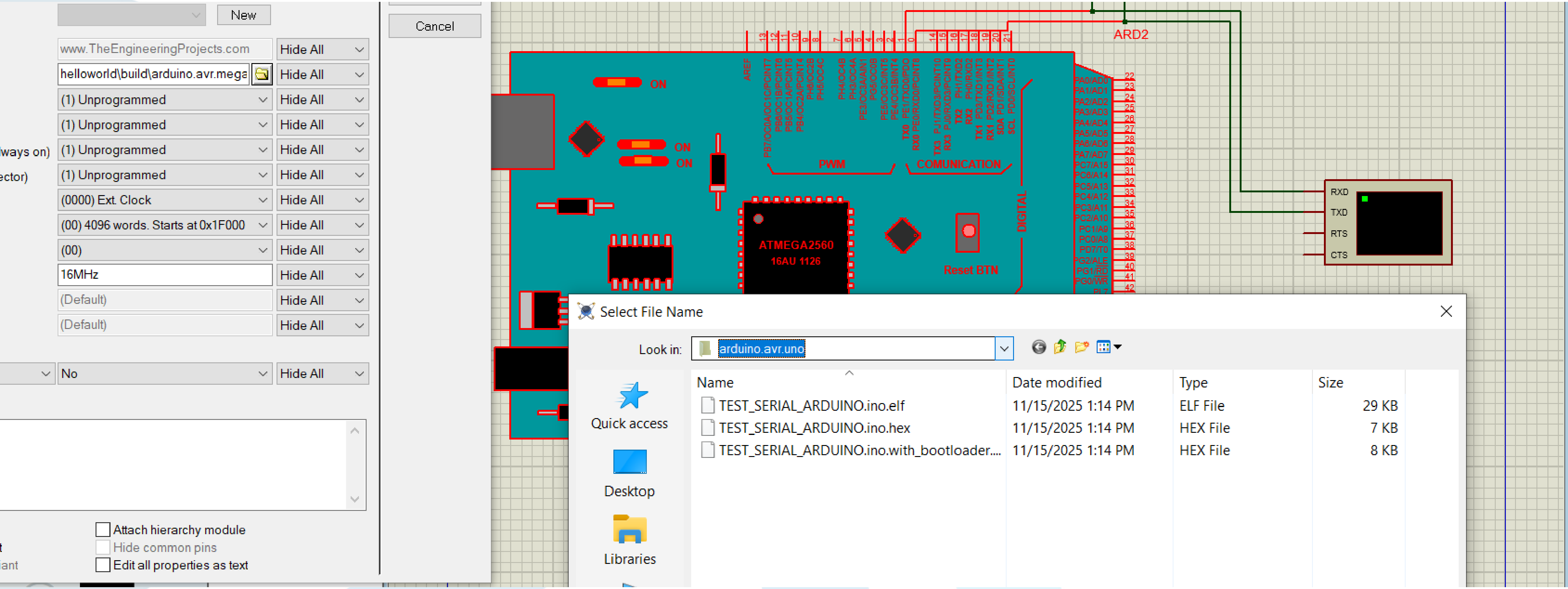
- build
- Project Backups
- helloworld.ino
- New Project.pdsprj
- New Project.pdsprj.DESKTOP-OVD23CQ....

Name

- arduino.avr.mega
- arduino.avr.uno

Name

- helloworld.ino.eep
- helloworld.ino.elf
- helloworld.ino.hex
- helloworld.ino.with_bootloader.bin
- helloworld.ino.with_bootloader.hex



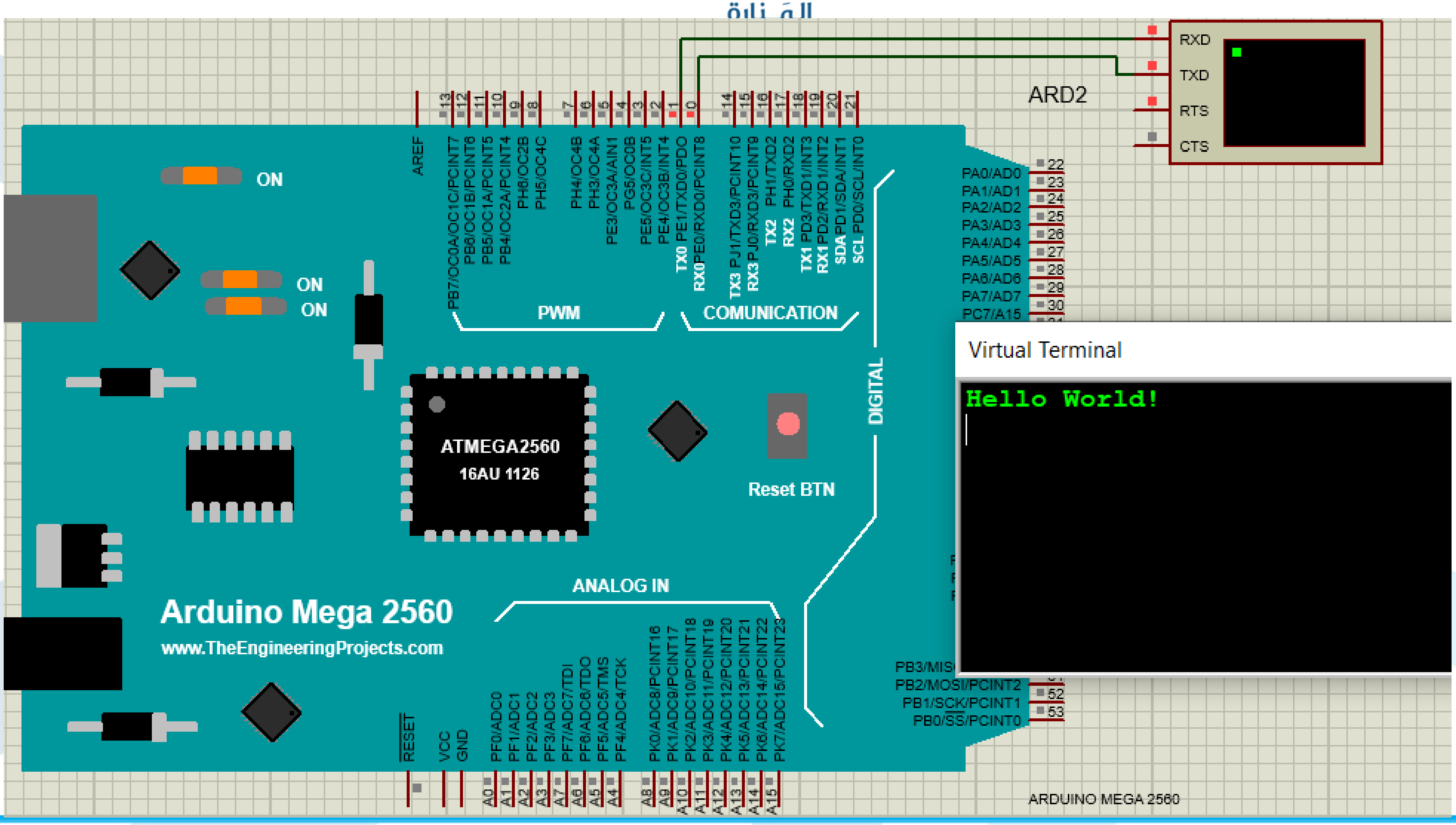
	www.TheEngineeringProjects.com	Hide All
	helloworld\build\arduino.avr.mega	Hide All
	(1) Unprogrammed	Hide All
	(1) Unprogrammed	Hide All
ways on)	(1) Unprogrammed	Hide All
ector)	(1) Unprogrammed	Hide All
	(0000) Ext. Clock	Hide All
	(00) 4096 words. Starts at 0x1F000	Hide All
	(00)	Hide All
	16MHz	Hide All
	(Default)	Hide All
	(Default)	Hide All
	No	Hide All

□ Attach hierarchy module
□ Hide common pins
□ Edit all properties as text

Select File Name

Look in: arduino.avr.uno

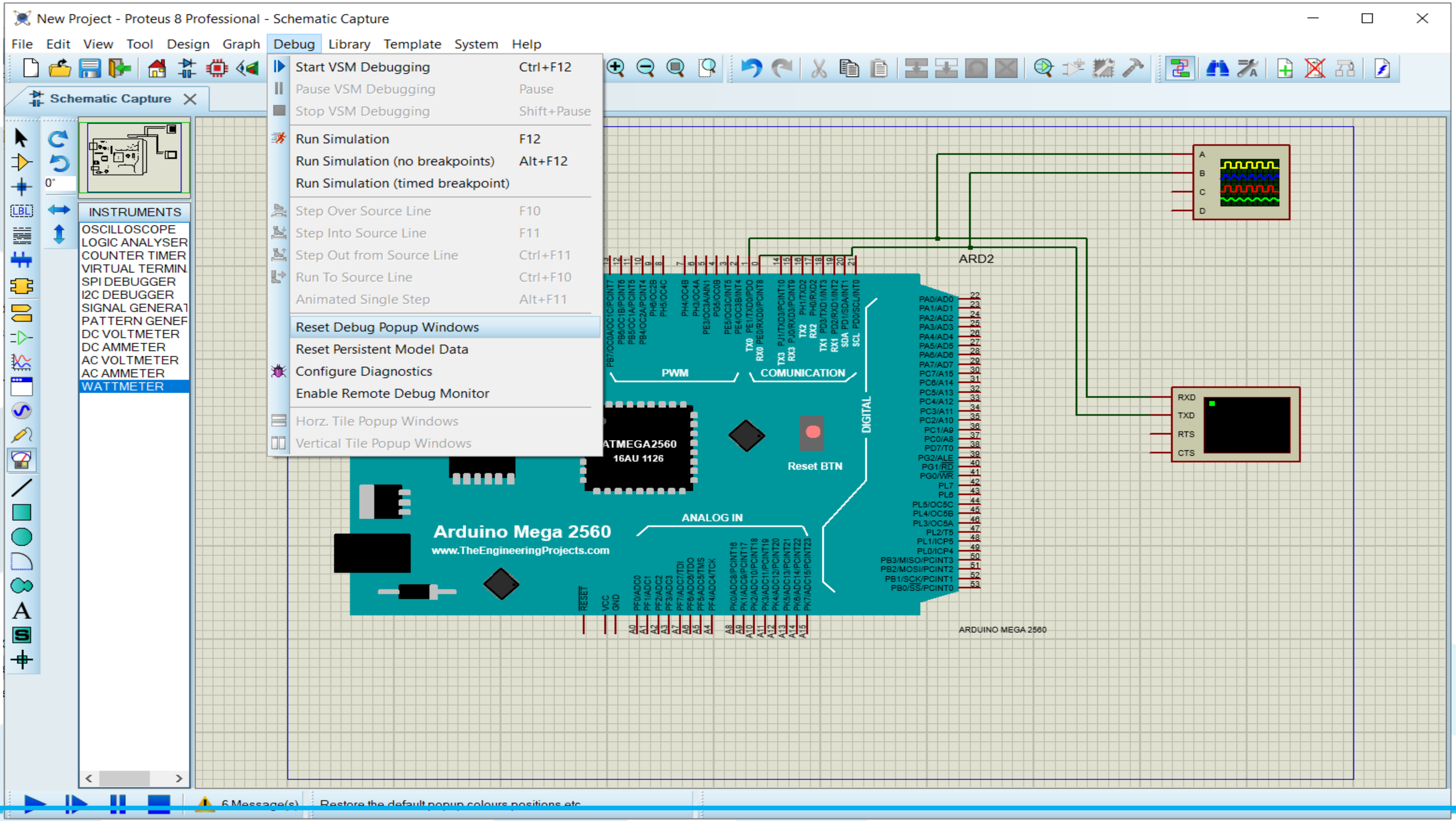
Name	Date modified	Type	Size
TEST_SERIAL_ARDUINO.ino.elf	11/15/2025 1:14 PM	ELF File	29 KB
TEST_SERIAL_ARDUINO.ino.hex	11/15/2025 1:14 PM	HEX File	7 KB
TEST_SERIAL_ARDUINO.ino.with_bootloader....	11/15/2025 1:14 PM	HEX File	8 KB



ARD2
RXD
TXD
RTS
CTS
ON
ON
ON
PWM
COMUNICATION
DIGITAL
ANALOG IN
ATMEGA2560
16AU 1126
Reset BTN
Arduino Mega 2560
www.TheEngineeringProjects.com
ARDUINO MEGA 2560
Virtual Terminal

```
Hello World!
```

Reopen closed windows of oscilloscope and terminal


New Project - Proteus 8 Professional - Schematic Capture
File
Edit
View
Tool
Design
Graph
Debug
Library
Template
System
Help
Start VSM Debugging Ctrl+F12
Pause VSM Debugging Pause
Stop VSM Debugging Shift+Pause
Run Simulation F12
Run Simulation (no breakpoints) Alt+F12
Run Simulation (timed breakpoint)
Step Over Source Line F10
Step Into Source Line F11
Step Out from Source Line Ctrl+F11
Run To Source Line Ctrl+F10
Animated Single Step Alt+F11
Reset Debug Popup Windows
Reset Persistent Model Data
Configure Diagnostics
Enable Remote Debug Monitor
Horz. Tile Popup Windows
Vertical Tile Popup Windows
Schematic Capture
INSTRUMENTS
OSCILLOSCOPE
LOGIC ANALYSER
COUNTER TIMER
VIRTUAL TERMIN.
SPI DEBUGGER
I2C DEBUGGER
SIGNAL GENERAT
PATTERN GENEF
DC VOLTMETER
DC AMMETER
AC VOLTMETER
AC AMMETER
WATTMETER
ARD2
ATMEGA2560 16AU 1126
Reset BTN
Arduino Mega 2560
www.TheEngineeringProjects.com
PWM
COMUNICATION
DIGITAL
ANALOG IN
ARDUINO MEGA 2560
A
B
C
D
RXD
TXD
RTS
CTS
6 Message(s)

Restore the default popup colours positions etc

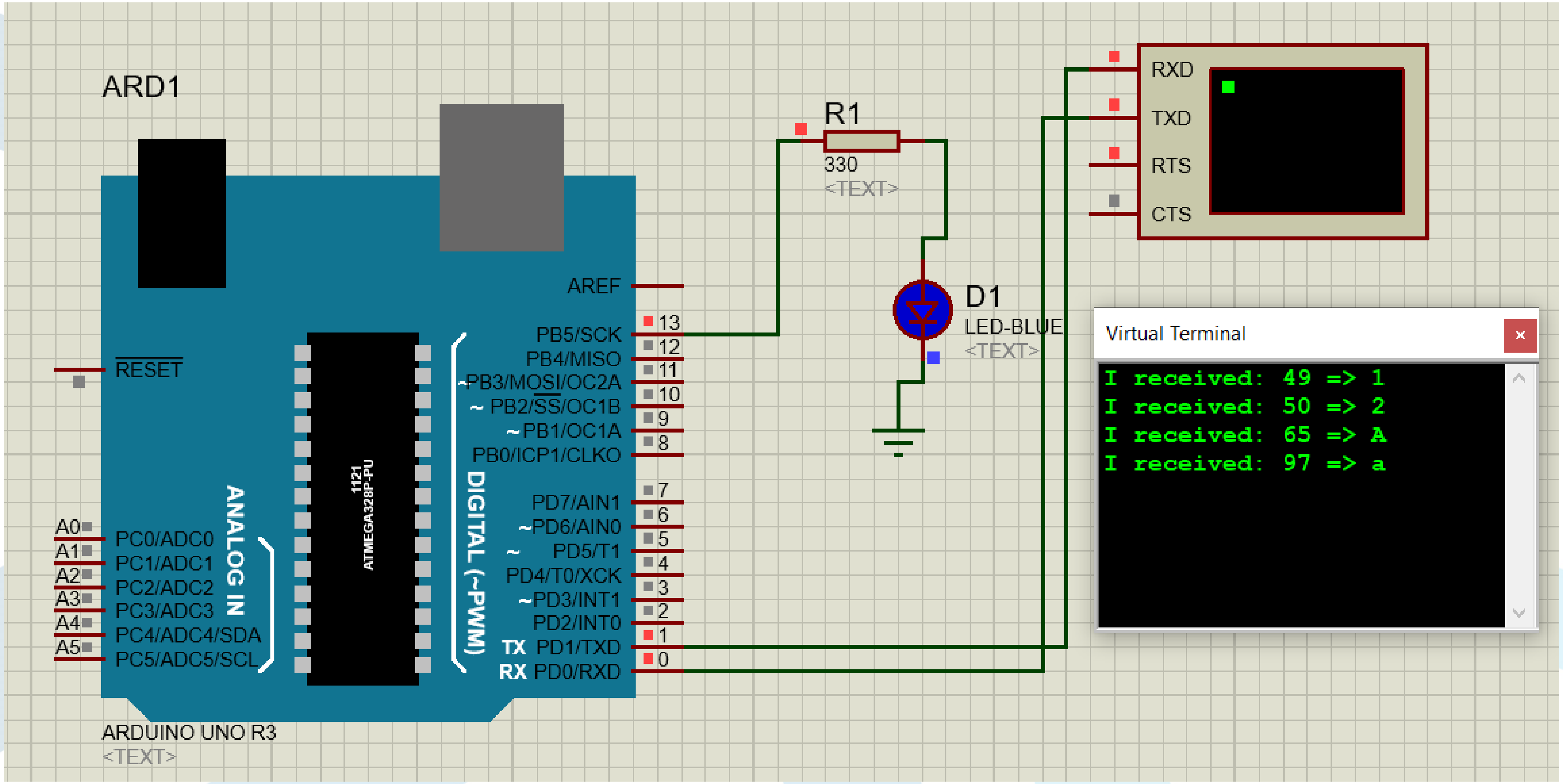
Example 2

```
// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(13, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(13, HIGH);   // turn the LED on (HIGH is the voltage level)
  delay(1000);              // wait for a second
  digitalWrite(13, LOW);    // turn the LED off by making the voltage LOW
  delay(1000);              // wait for a second
}
```

Example 3

```
int val;
int ledpin=13;
void setup()
{
Serial.begin(9600);
pinMode(ledpin,OUTPUT);
}
void loop()
{
val=Serial.read();
digitalWrite(ledpin,HIGH);
delay(500);
digitalWrite(ledpin,LOW);
delay(500);
 if (Serial.available() > 0) {
                val = Serial.read();// read the incoming byte:
                Serial.print("I received: ");
                //Serial.println(val, BIN);
                Serial.print(val, DEC);
                //Serial.println(val, HEX);
                Serial.print(" => ");
                Serial.println(char(val));}
}
```

Example 2&3



```
I received: 49 => 1
I received: 50 => 2
I received: 65 => A
I received: 97 => a
```

Example 4
digital input/output

```
#define ledPin 13 // choose the pin for the LED
#define switchPin 7 // choose the input pin (for a
pushbutton)

int val = 0; // variable for reading the pin status

void setup()
{
  pinMode(ledPin, OUTPUT); // declare LED as
output
  pinMode(switchPin, INPUT); // declare
pushbutton as input
}
```

```
void loop()
{
  val = digitalRead(switchPin); // read
input value
  if (val == HIGH)
  { // check if the input is HIGH (button
released)
    digitalWrite(ledPin, LOW); // turn
LED OFF
  } else {
    digitalWrite(ledPin, HIGH); // turn
LED ON } }
  }
}
```

شاشة الإظهار LCD

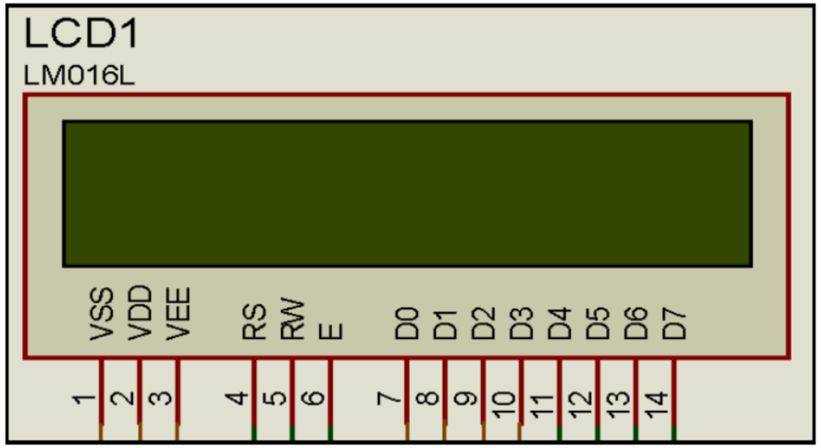
يتم التحكم بمثل هذه الشاشة بواسطة متحكم خاص ويثبت هذا المتحكم على الدارة المطبوعة خلف الشاشة و تملك الشاشة أيضا ذاكرة و مسجلات.

وتعتبر هذه الشاشات متوفرة بشكل واسع وقليلة الكلفة وتتراوح أحجامها من 8 إلى 80 محرف و تتعدد أنواعها مختلفة عن بعضها بعدد المحارف في كل سطر و عدد الأسطر في الشاشة , نذكر أن الأكثر شيوعا ًهي ذات 16 أو 20 محرف مع سطر أو سطرين أو أربعة أسطر في الشاشة.

الشاشة تعرض مجموعة محارف ASCII بالإضافة إلى رموز لاتينية و محددة و بعض الرموز الرياضية و تعمل عند جهد تغذية 5V.

تمتلك مسجلين كل منهما 8Bitهما مسجل التعليمات IR ومسجل المعطيات DR

1. يستخدم المسجل IR لتخزين التعليمات
2. في حين يخزن DR مؤقتاً المعطيات المقروءة من أو المكتوبة إلى الذاكرة التي تقوم بدورها بتخزين جميع المحارف المعروضة على الشاشة.

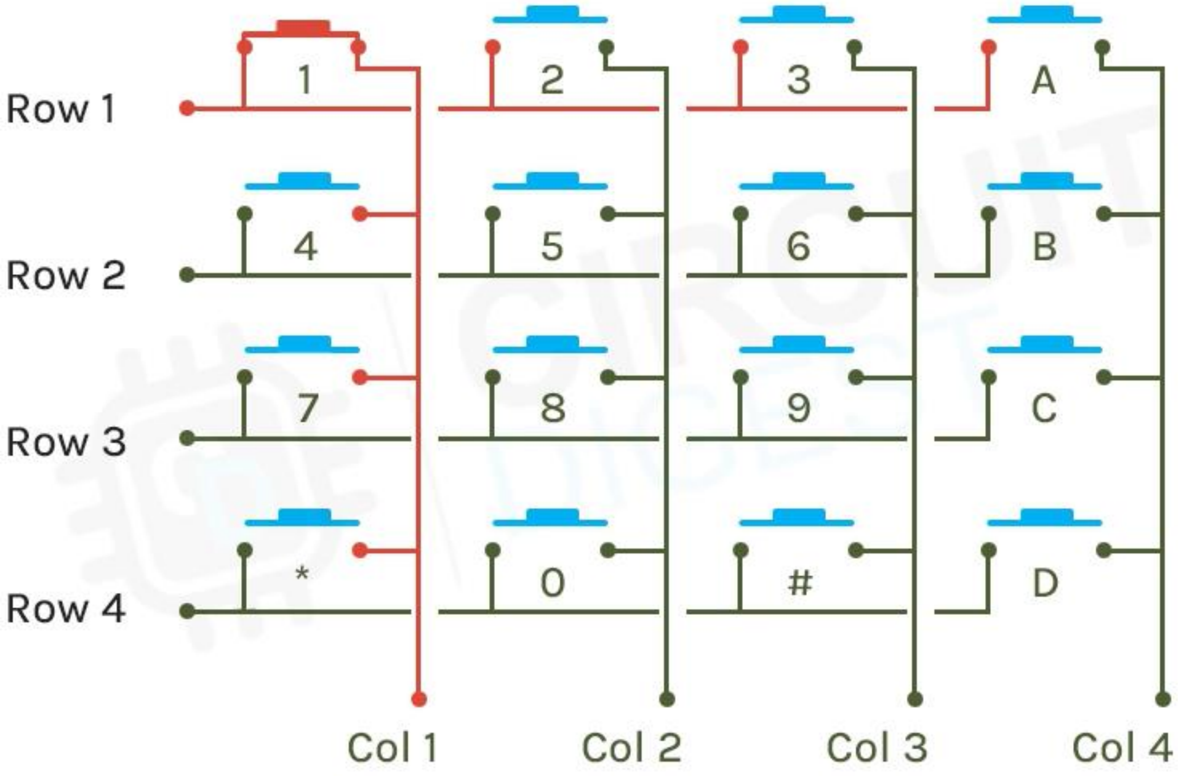


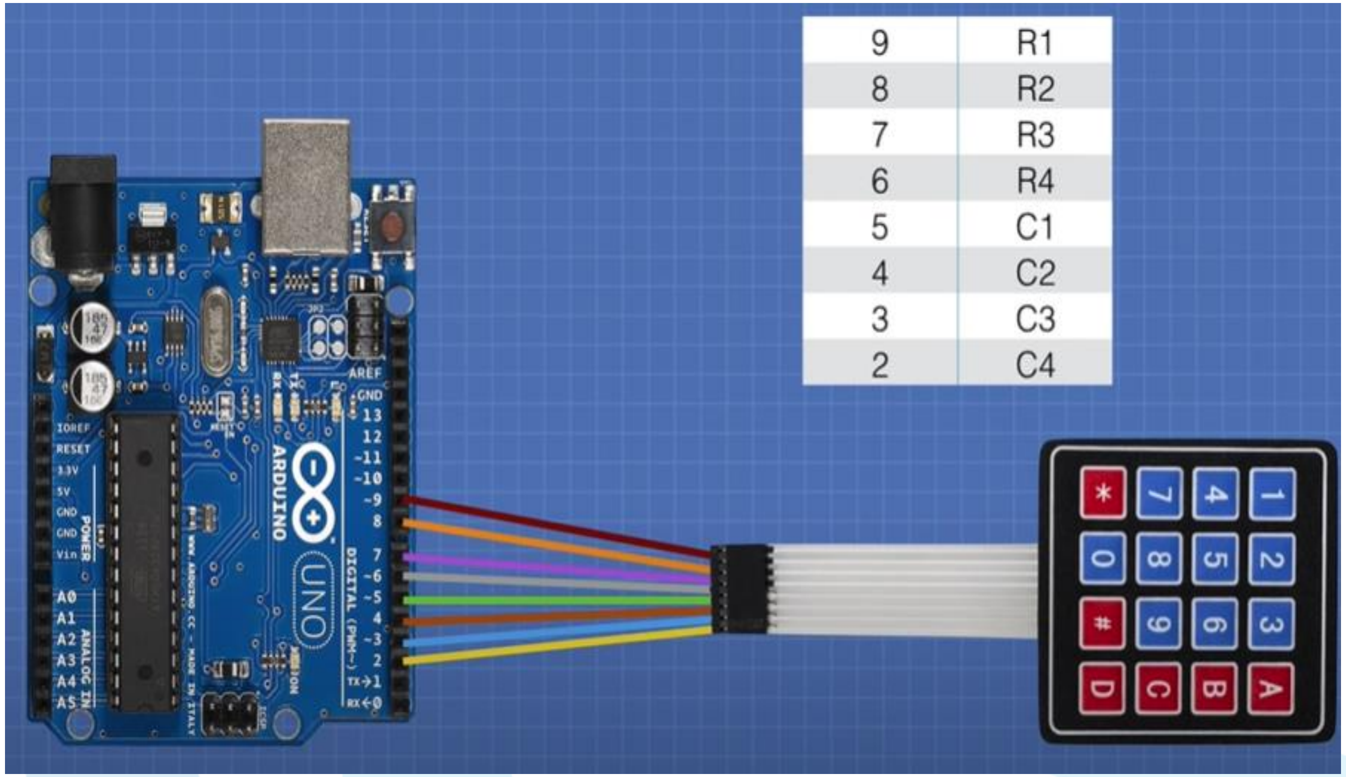
يتم التحكم بمثل هذه الشاشة بواسطة متحكم خاص ويثبت هذا المتحكم على الدارة المطبوعة خلف الشاشة و تملك الشاشة أيضا ذاكرة و مسجلات.

خط التحكم	الوظيفة	
خط التمكين E	0	الـ LCDغير مفعلة أو DISABLE
	1	الـ LCD مفعلة أو ENABLE
خط القراءة/الكتابة R/W	0	العملية هي كتابة إلى الـ LCD
	1	العملية هي قراءة من الـ LCD
خط اختيار المسجل RS	0	مسجل التعليمة IR
	1	مسجل المعطيات DR

الشاشة	نقاط الوصل المقابلة
Vdd	5 volts
Vee	مقاومة متغيرة للتحكم بسطوع الشاشة
Vss	GND
RS	D1 from Port D
R / W	D2 from Port D
E	D0 from Port D
D4 إلى D7	D4 إلى D7 from Port D

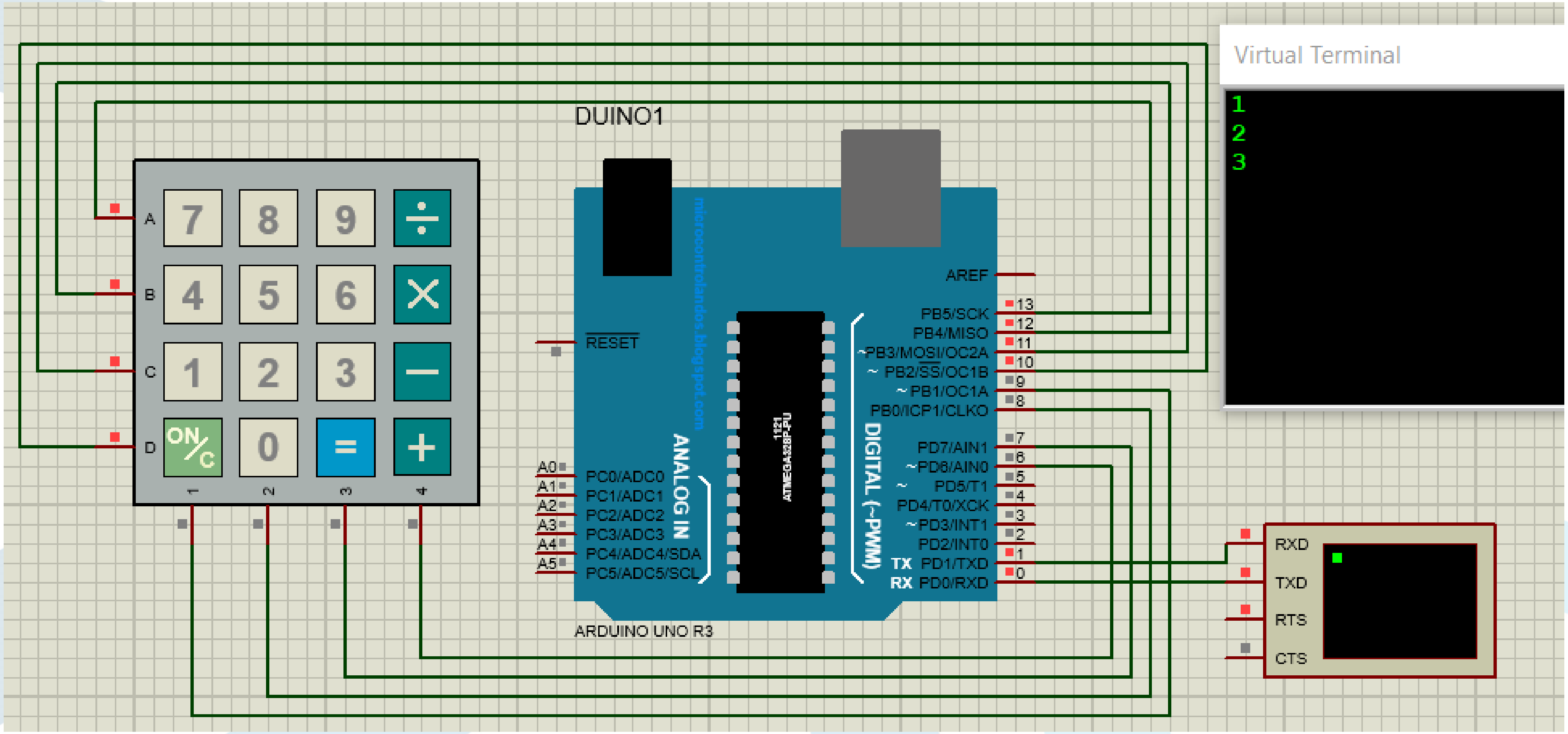
Using Keypads with Arduino





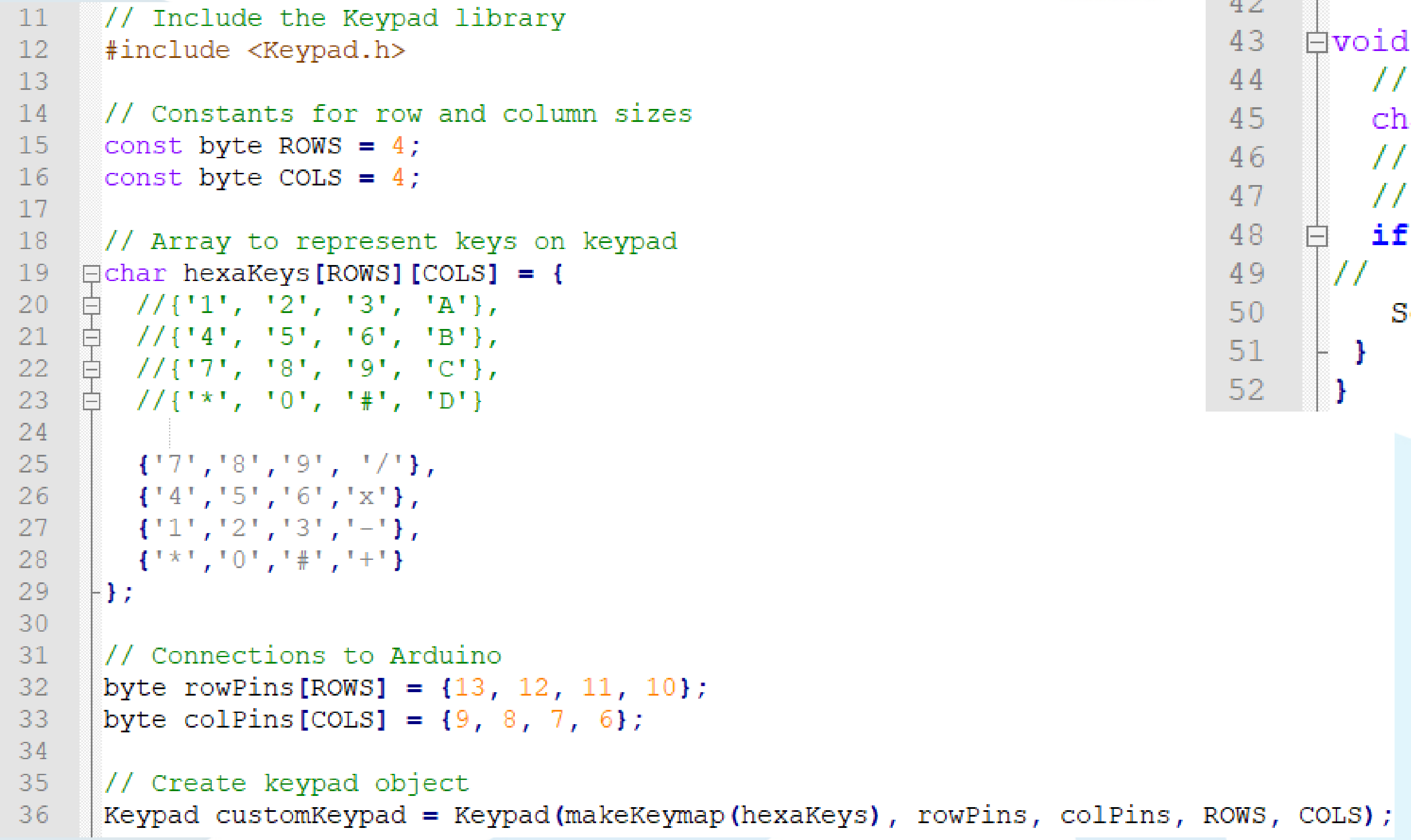
9	R1
8	R2
7	R3
6	R4
5	C1
4	C2
3	C3
2	C4

Example 3

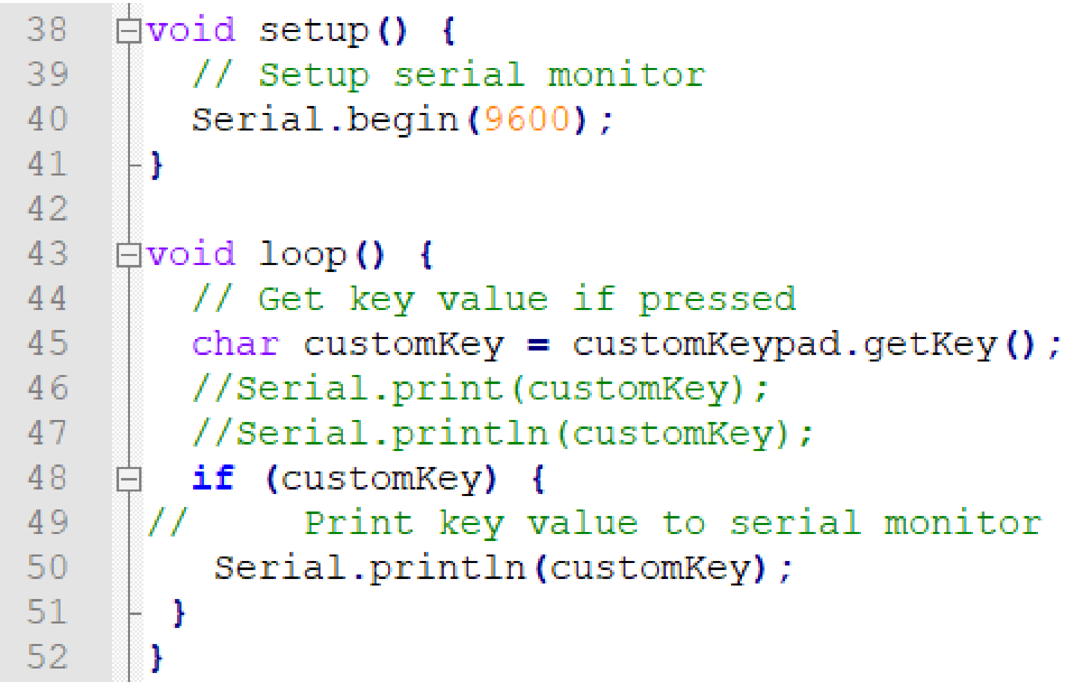


```
1
2
3
```

Example 3

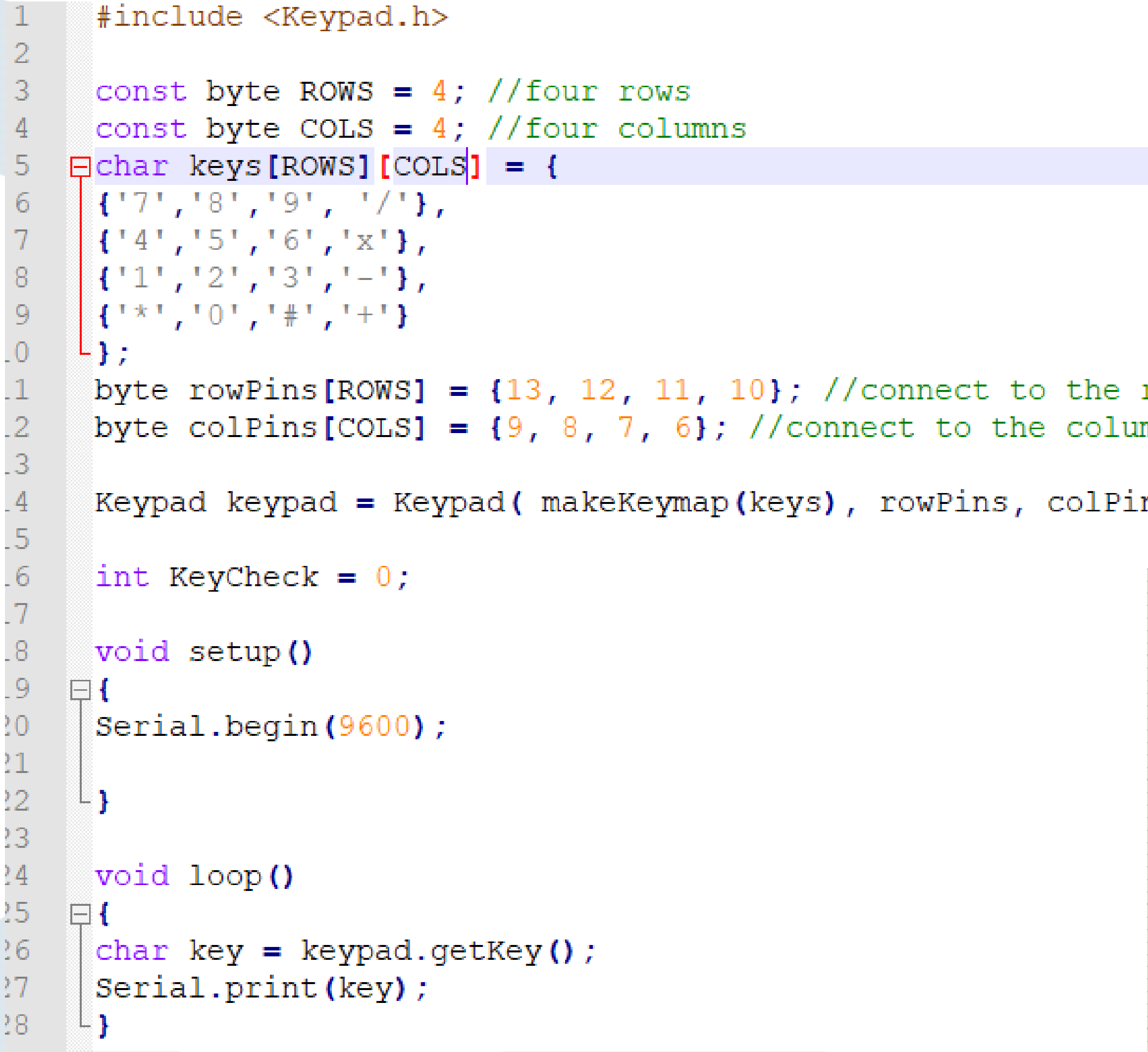


```
// Include the Keypad library
#include <Keypad.h>

// Constants for row and column sizes
const byte ROWS = 4;
const byte COLS = 4;

// Array to represent keys on keypad
char hexaKeys[ROWS][COLS] = {
  //{'1', '2', '3', 'A'},
  //{'4', '5', '6', 'B'},
  //{'7', '8', '9', 'C'},
  //{'*', '0', '#', 'D'}

  {'7','8','9', '/'},
  {'4','5','6','x'},
  {'1','2','3','-'},
  {'*','0','#','+'}
};

// Connections to Arduino
byte rowPins[ROWS] = {13, 12, 11, 10};
byte colPins[COLS] = {9, 8, 7, 6};

// Create keypad object
Keypad customKeypad = Keypad(makeKeymap(hexaKeys), rowPins, colPins, ROWS, COLS);
```



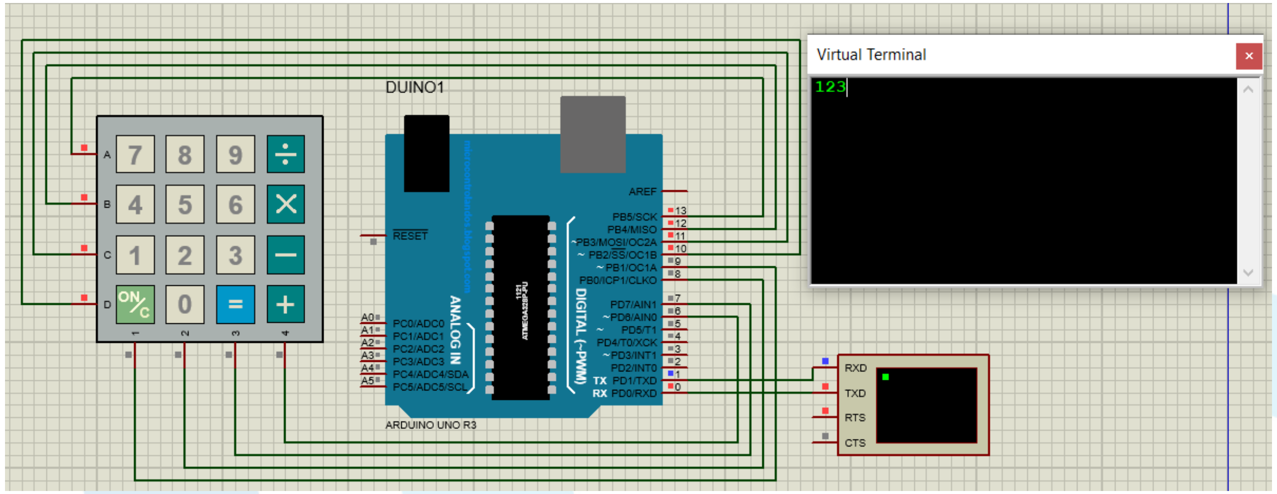
```
void setup() {
  // Setup serial monitor
  Serial.begin(9600);
}

void loop() {
  // Get key value if pressed
  char customKey = customKeypad.getKey();
  //Serial.print(customKey);
  //Serial.println(customKey);
  if (customKey) {
//      Print key value to serial monitor
   Serial.println(customKey);
  }
}
```

The Loop is very simple. We use the *getKey* method of the keypad library to get a key value when it detects a keypress. Then we simply print that to the serial monitor.



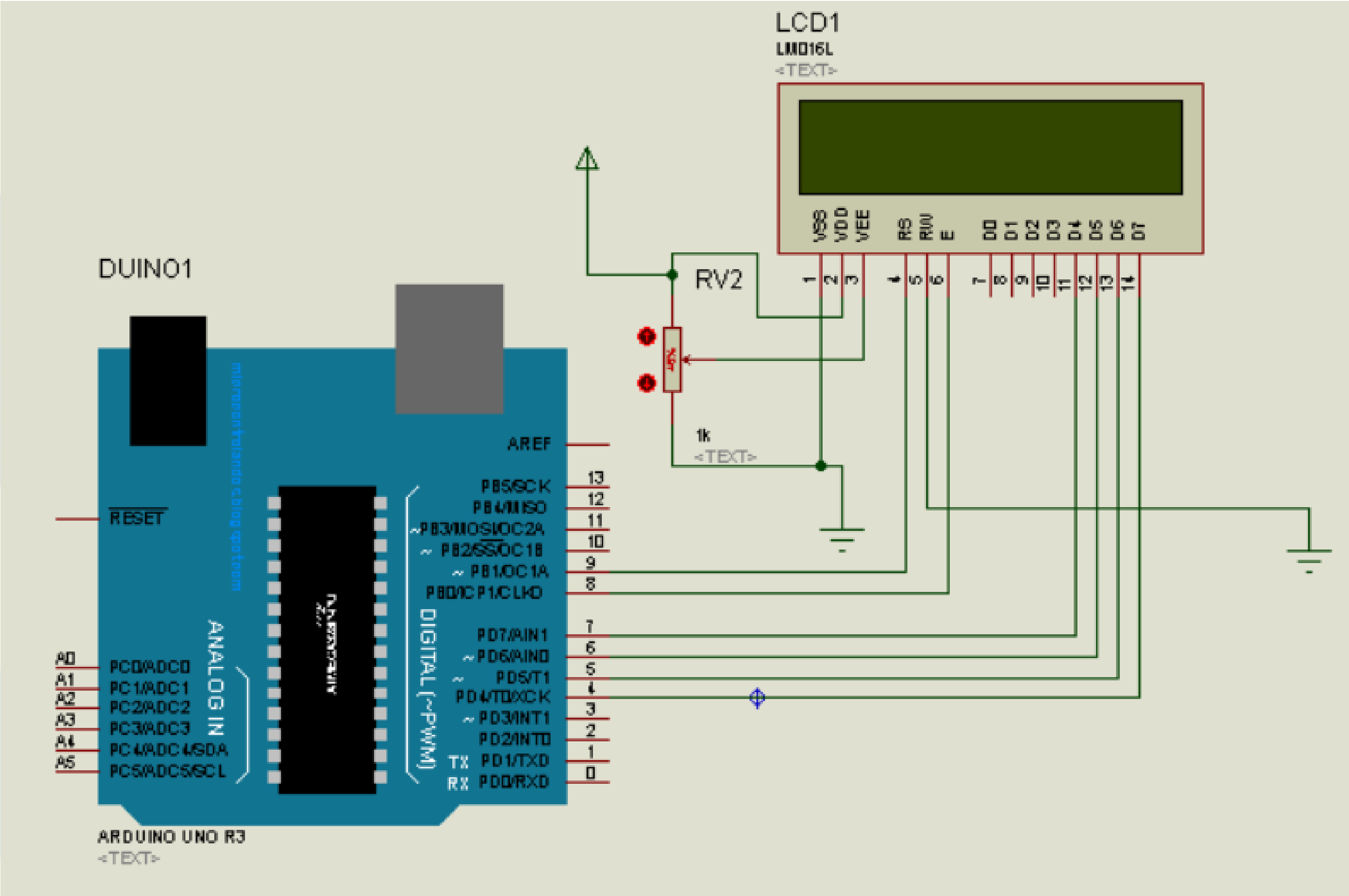
```
#include <Keypad.h>

const byte ROWS = 4; //four rows
const byte COLS = 4; //four columns
char keys[ROWS][COLS] = {
{'7','8','9', '/'},
{'4','5','6','x'},
{'1','2','3','-'},
{'*','0','#','+'}
};
byte rowPins[ROWS] = {13, 12, 11, 10}; //connect to the row pinouts of the keypad
byte colPins[COLS] = {9, 8, 7, 6}; //connect to the column pinouts of the keypad

Keypad keypad = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS );

int KeyCheck = 0;

void setup()
{
Serial.begin(9600);

}

void loop()
{
char key = keypad.getKey();
Serial.print(key);
}
```

The Loop is very simple. We use the getKey method of the keypad library to get a key value when it detects a keypress. Then we simply print that to the serial monitor.

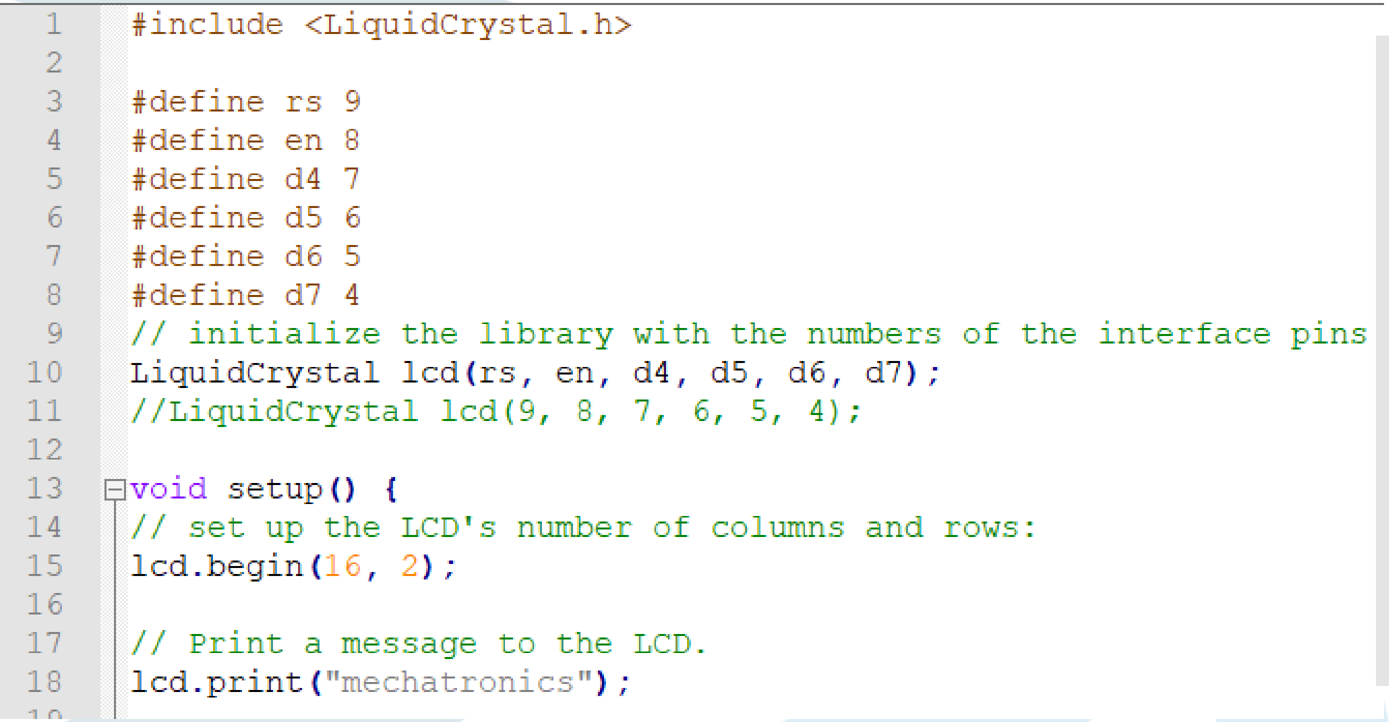


```
123
```

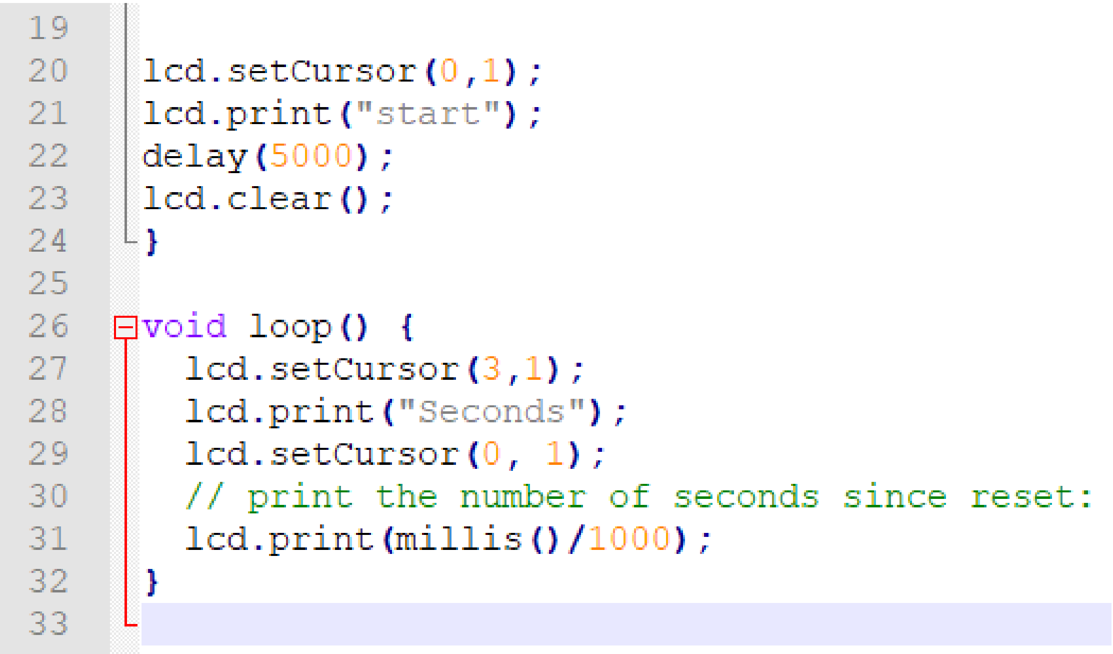
Connect LCD TO Arduino using LiquidCrystal library



Connect LCD TO Arduino using LiquidCrystal library

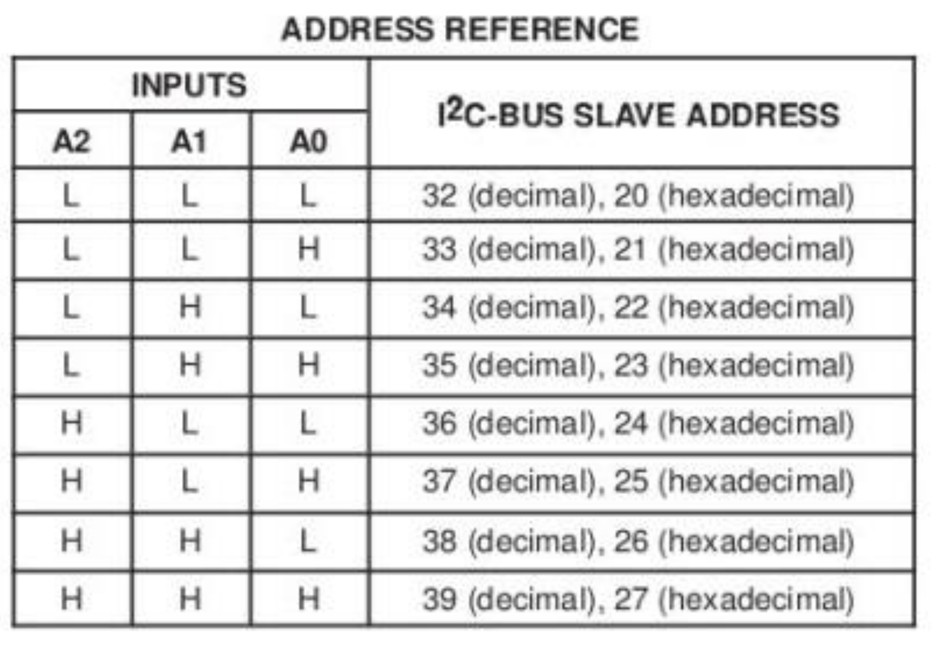


```
#include <LiquidCrystal.h>

#define rs 9
#define en 8
#define d4 7
#define d5 6
#define d6 5
#define d7 4
// initialize the library with the numbers of the interface pins
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
//LiquidCrystal lcd(9, 8, 7, 6, 5, 4);

void setup() {
// set up the LCD's number of columns and rows:
lcd.begin(16, 2);

// Print a message to the LCD.
lcd.print("mechatronics");
```



```

lcd.setCursor(0,1);
lcd.print("start");
delay(5000);
lcd.clear();
}

void loop() {
  lcd.setCursor(3,1);
  lcd.print("Seconds");
  lcd.setCursor(0, 1);
  // print the number of seconds since reset:
  lcd.print(millis()/1000);
}
```

The PCF8574/74A provides general-purpose remote I/O expansion via the two-wire bidirectional I2C-bus (serial clock (SCL), serial data (SDA)).



ADDRESS REFERENCE

INPUTS			I2C-BUS SLAVE ADDRESS
A2	A1	A0	
L	L	L	32 (decimal), 20 (hexadecimal)
L	L	H	33 (decimal), 21 (hexadecimal)
L	H	L	34 (decimal), 22 (hexadecimal)
L	H	H	35 (decimal), 23 (hexadecimal)
H	L	L	36 (decimal), 24 (hexadecimal)
H	L	H	37 (decimal), 25 (hexadecimal)
H	H	L	38 (decimal), 26 (hexadecimal)
H	H	H	39 (decimal), 27 (hexadecimal)

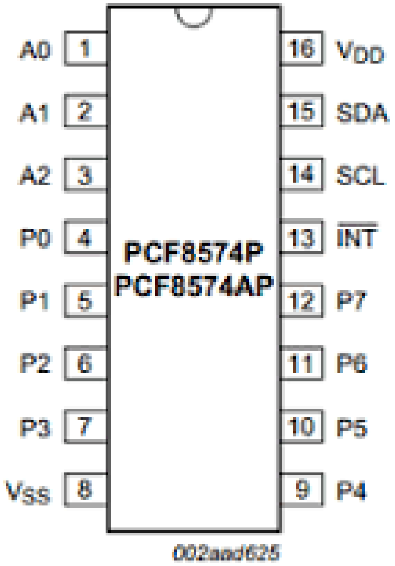
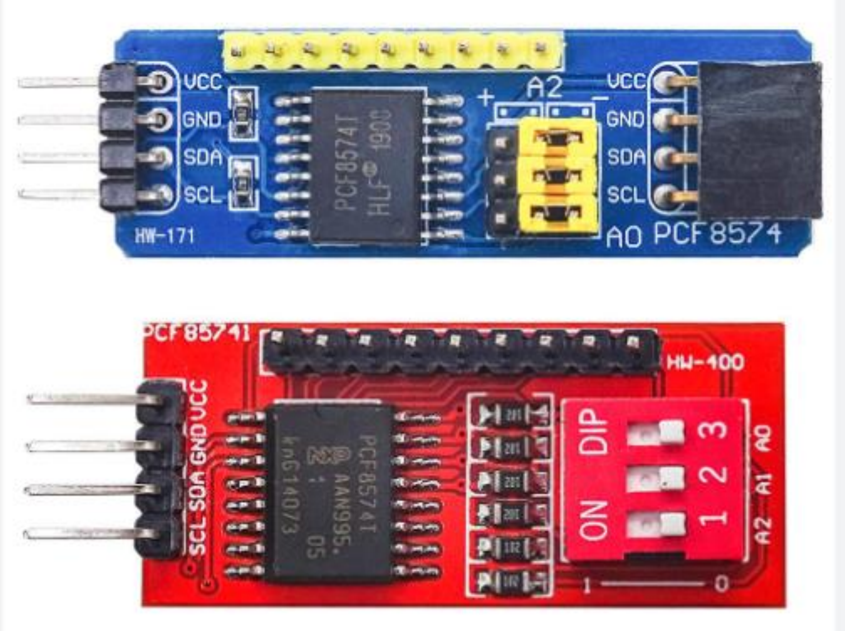
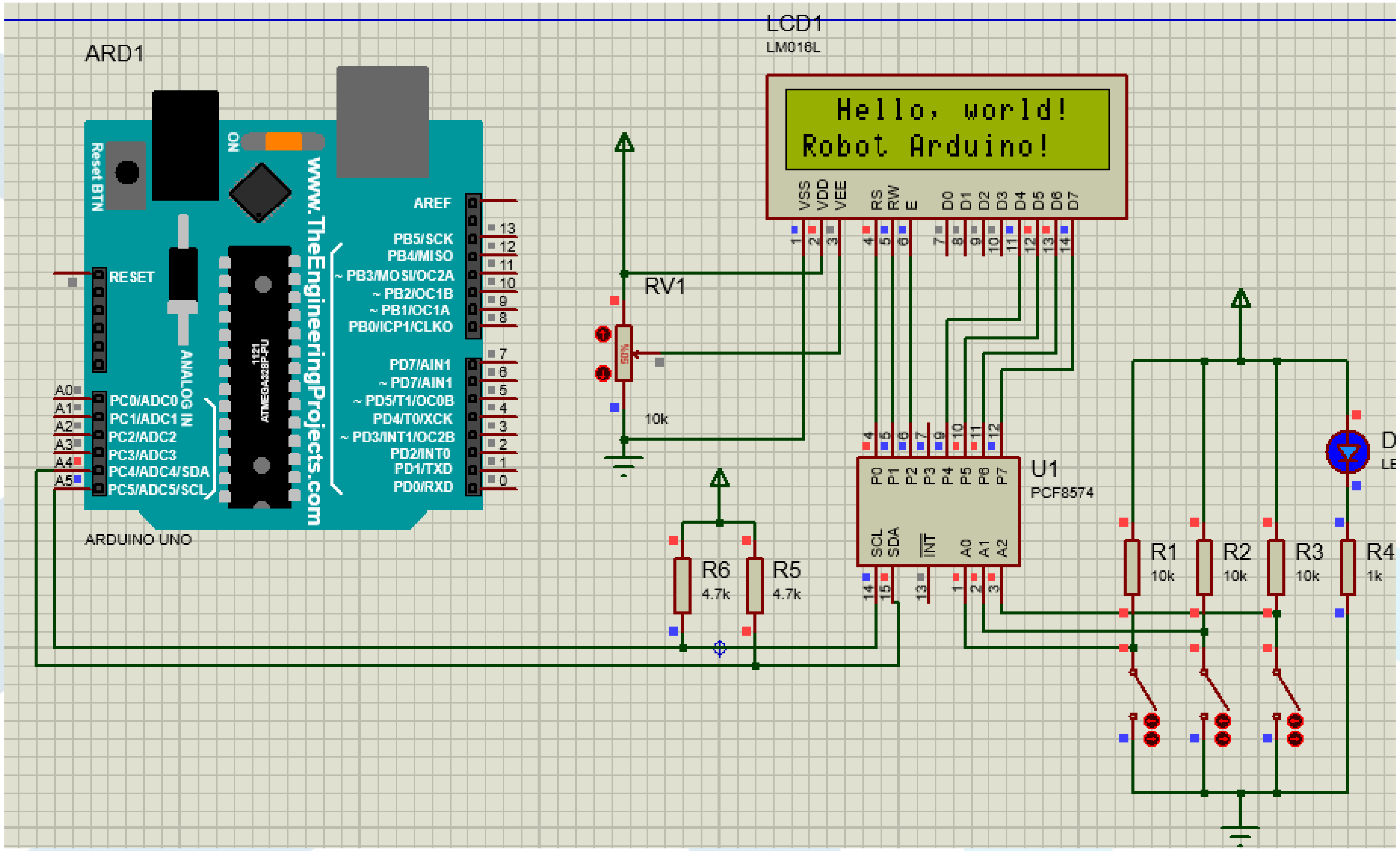


Fig 3. Pin configuration for DIP16

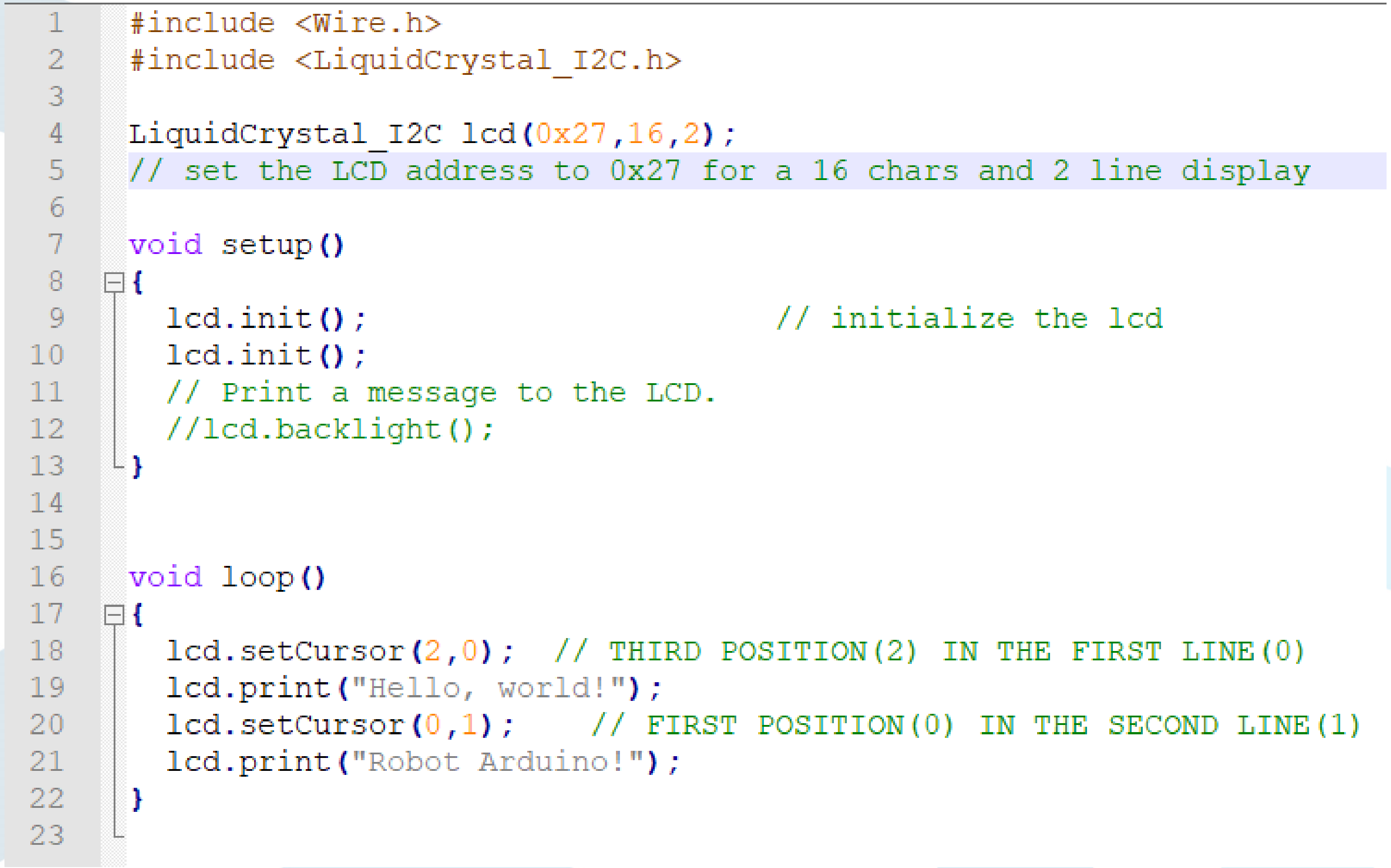


PCF8574 or MCP23008

Connect LCD TO Arduino using i2c



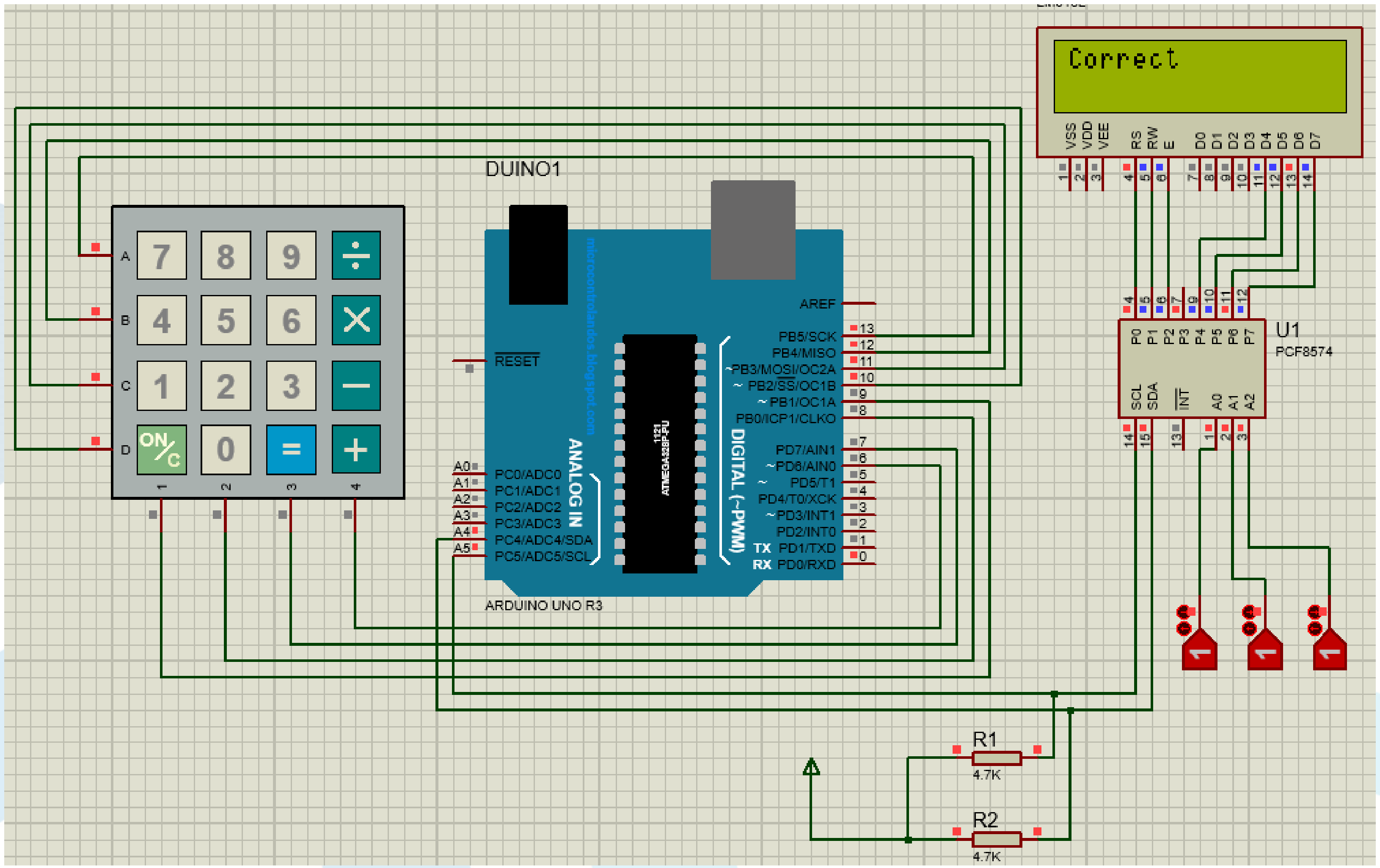
Connect LCD TO Arduino using i2c



```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27,16,2);
// set the LCD address to 0x27 for a 16 chars and 2 line display

void setup()
{
  lcd.init();                      // initialize the lcd
  lcd.init();
  // Print a message to the LCD.
  //lcd.backlight();
}


void loop()
{
  lcd.setCursor(2,0);  // THIRD POSITION(2) IN THE FIRST LINE(0)
  lcd.print("Hello, world!");
  lcd.setCursor(0,1);    // FIRST POSITION(0) IN THE SECOND LINE(1)
  lcd.print("Robot Arduino!");
}

```

Homework

Develop an Arduino Uno program that implements a basic password-entry system using a 4x4 keypad and a 16x2 LCD display ((can be parallel or I2C)).



جَـامعة
المَـنارة
MANARA UNIVERSITY

انتهت المحاضرة