Embedded Systems and IOT

2025/2026

Lecture 2

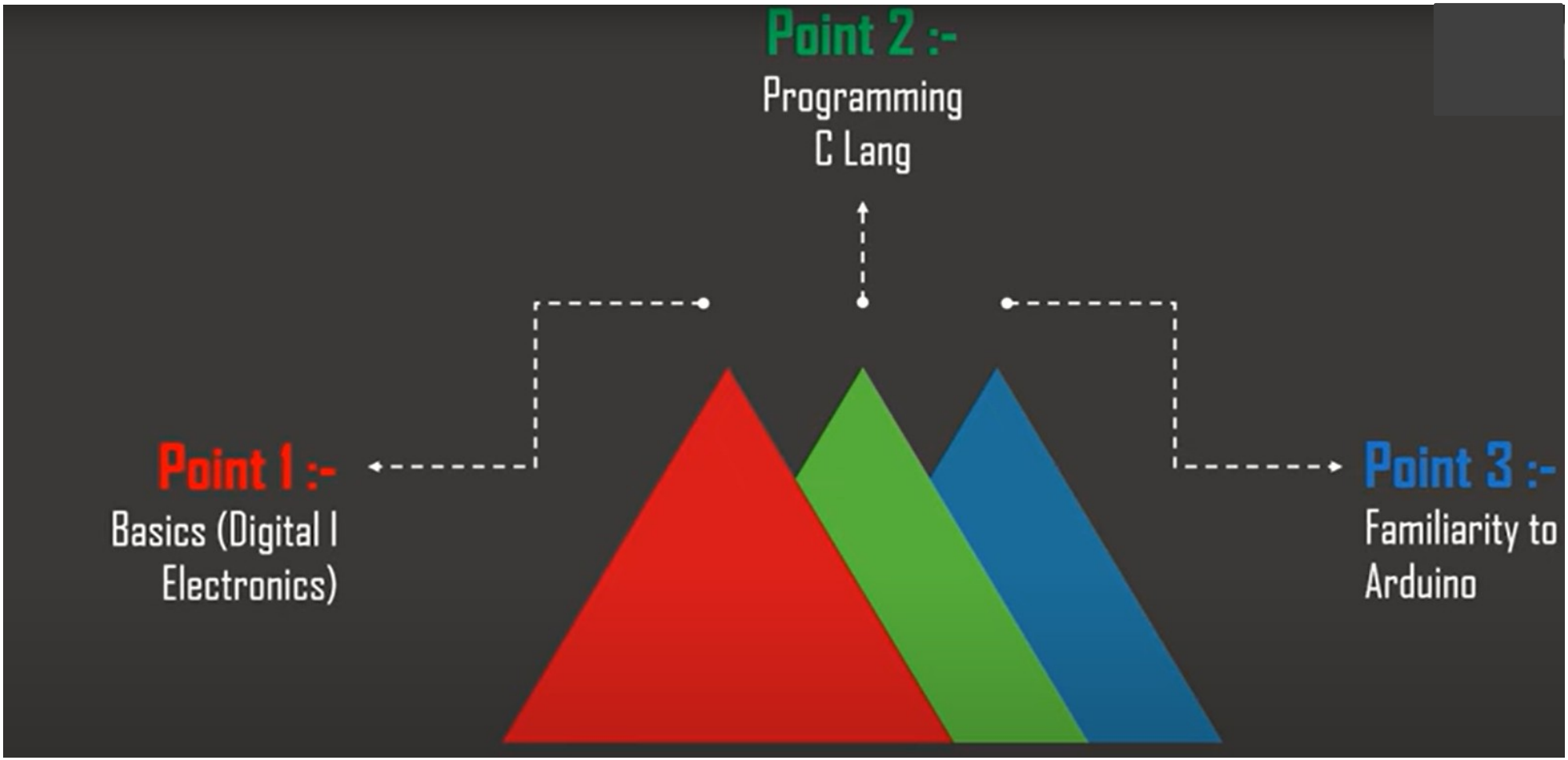
Assis. Prof. Dr. Elmahdy Maree

Embedded Systems and IOT

Introduction

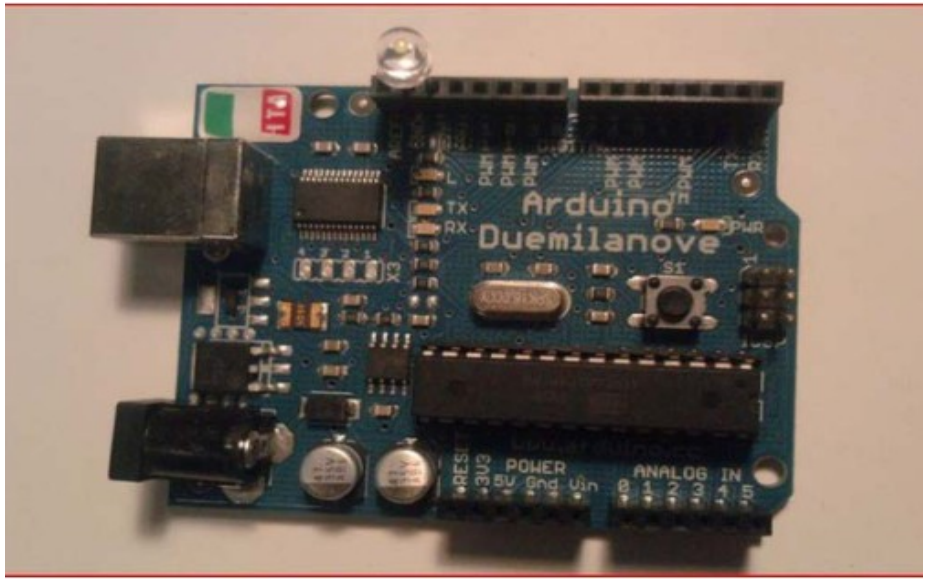
Embedded Systems and IOT

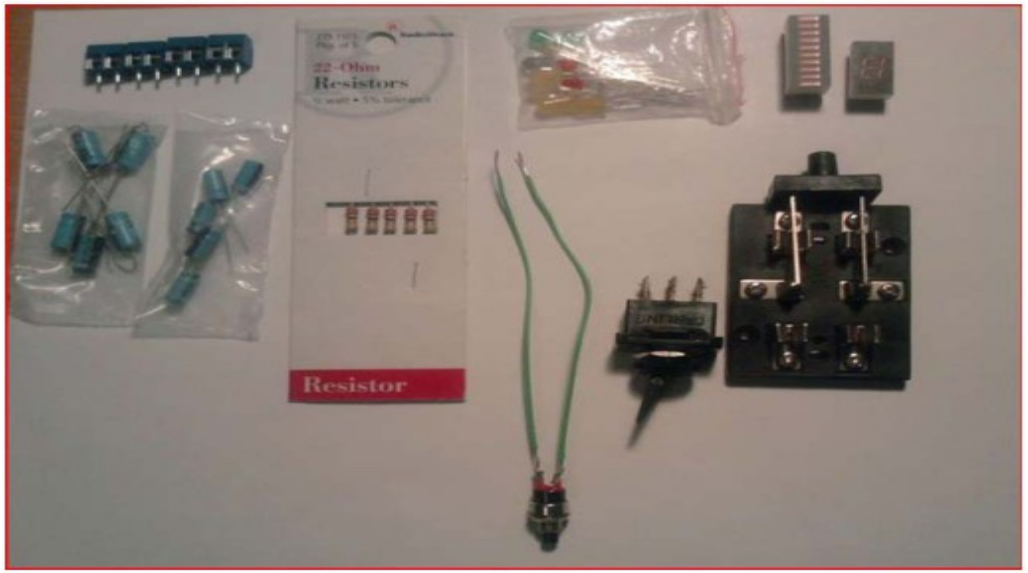
Introduction: Course prerequisite



Embedded Systems and IOT

Introduction cont'd





Embedded Systems and IOT

Embedded System Development Life cycle

1- System Design

2- Coding

3- Simulation for verification and testing

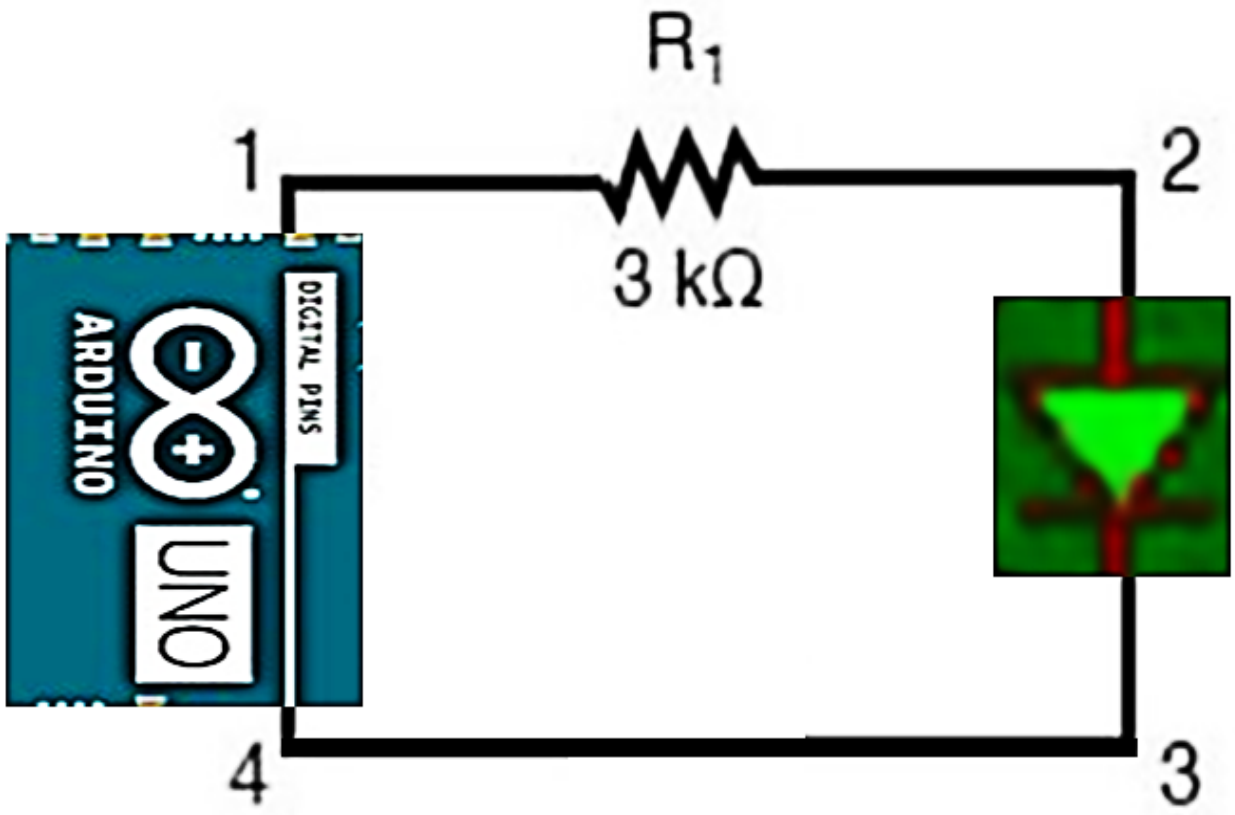
4- Hardware Implementation

5- Testing and Debugging

Embedded Systems and IOT

Embedded System Development Life cycle

1- System Design



Embedded Systems and IOT

Embedded System Development Life cycle

2- Coding

LEDblinkingCode | Arduino 1.8.15 Hourly Build 2021/05/31 10:33

File Edit Sketch Tools Help

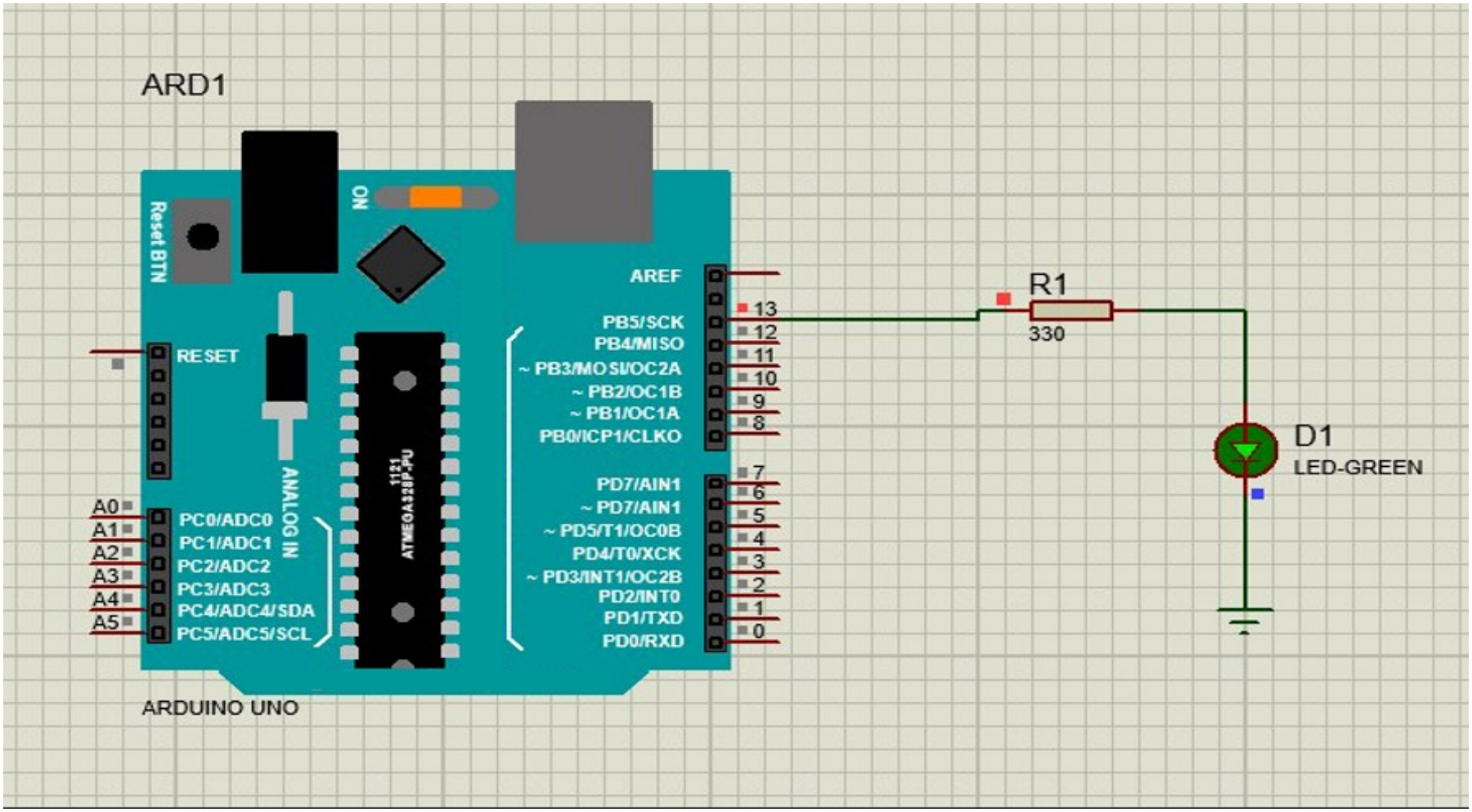
LEDblinkingCode

```
void setup() {

pinMode(13, OUTPUT);

}


void loop() {

digitalWrite(13, HIGH);

delay(1000);

digitalWrite(13, LOW);
delay(1000); // wait for a second

}
```

Embedded Systems and IOT

Embedded System Development Life cycle

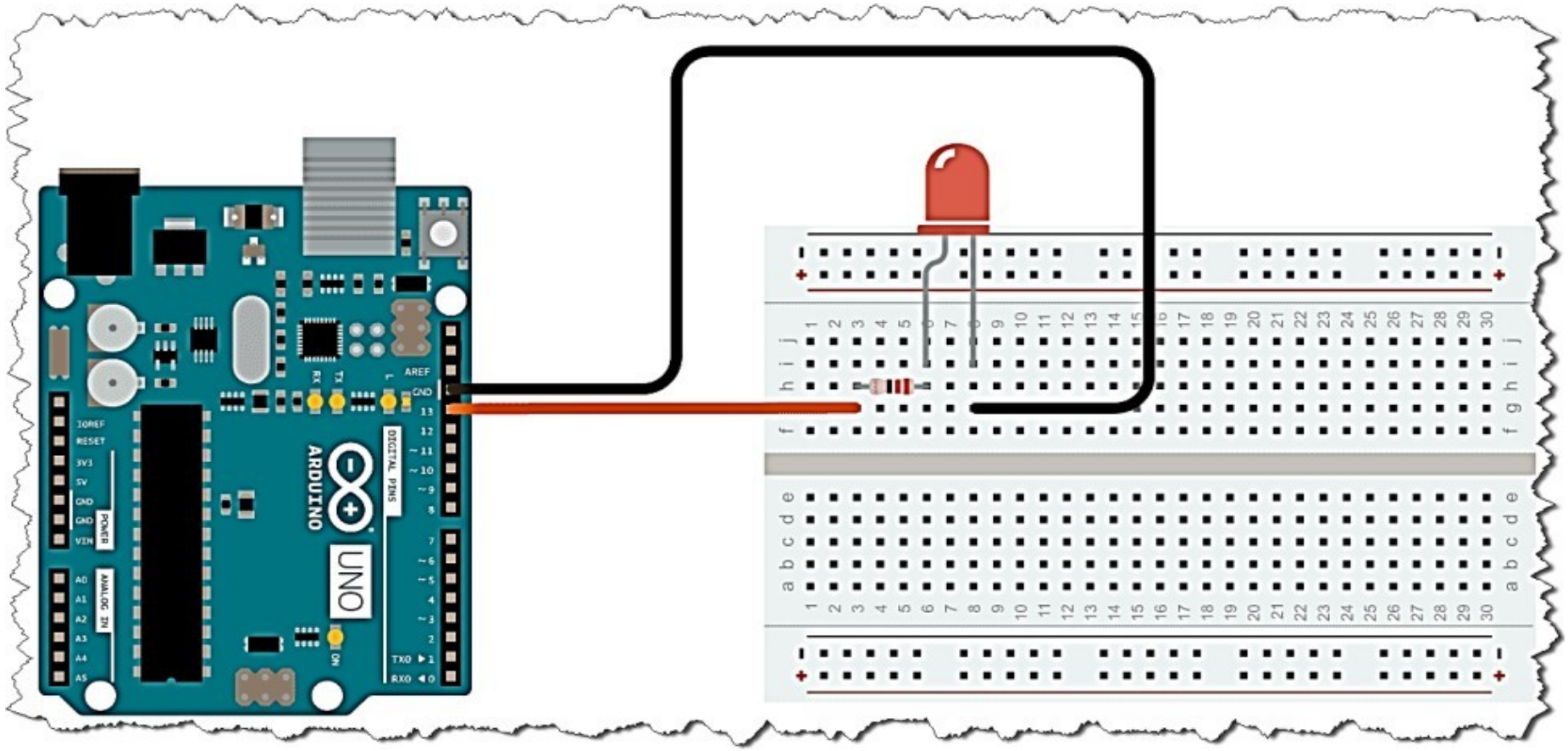
3- Simulation for verification and testing (online or offline)



Embedded Systems and IOT

Embedded System Development Life cycle

4- Hardware Implementation



5- Testing and Debugging

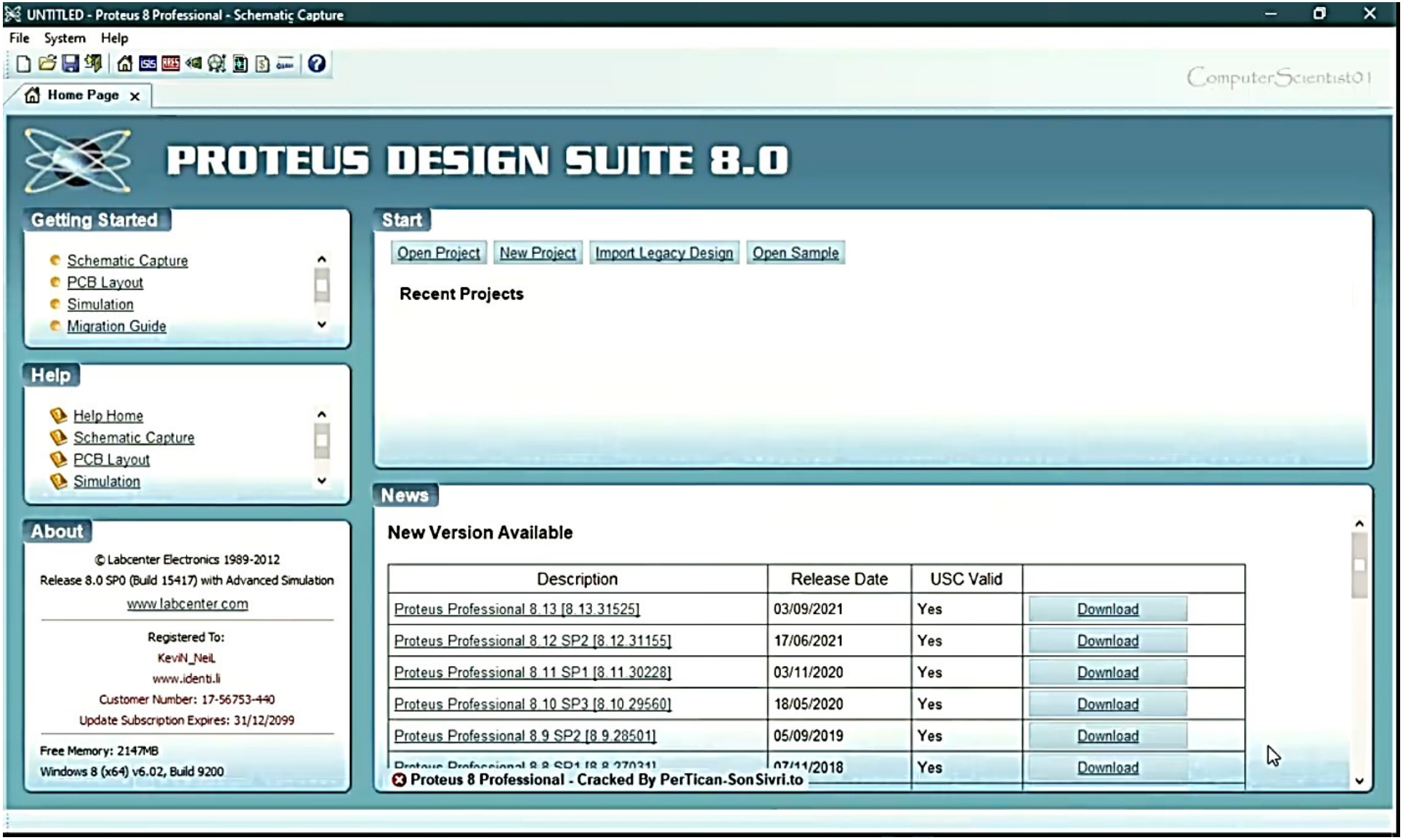
Embedded Systems and IOT

Simulation for verification and testing (Offline)

Proteus

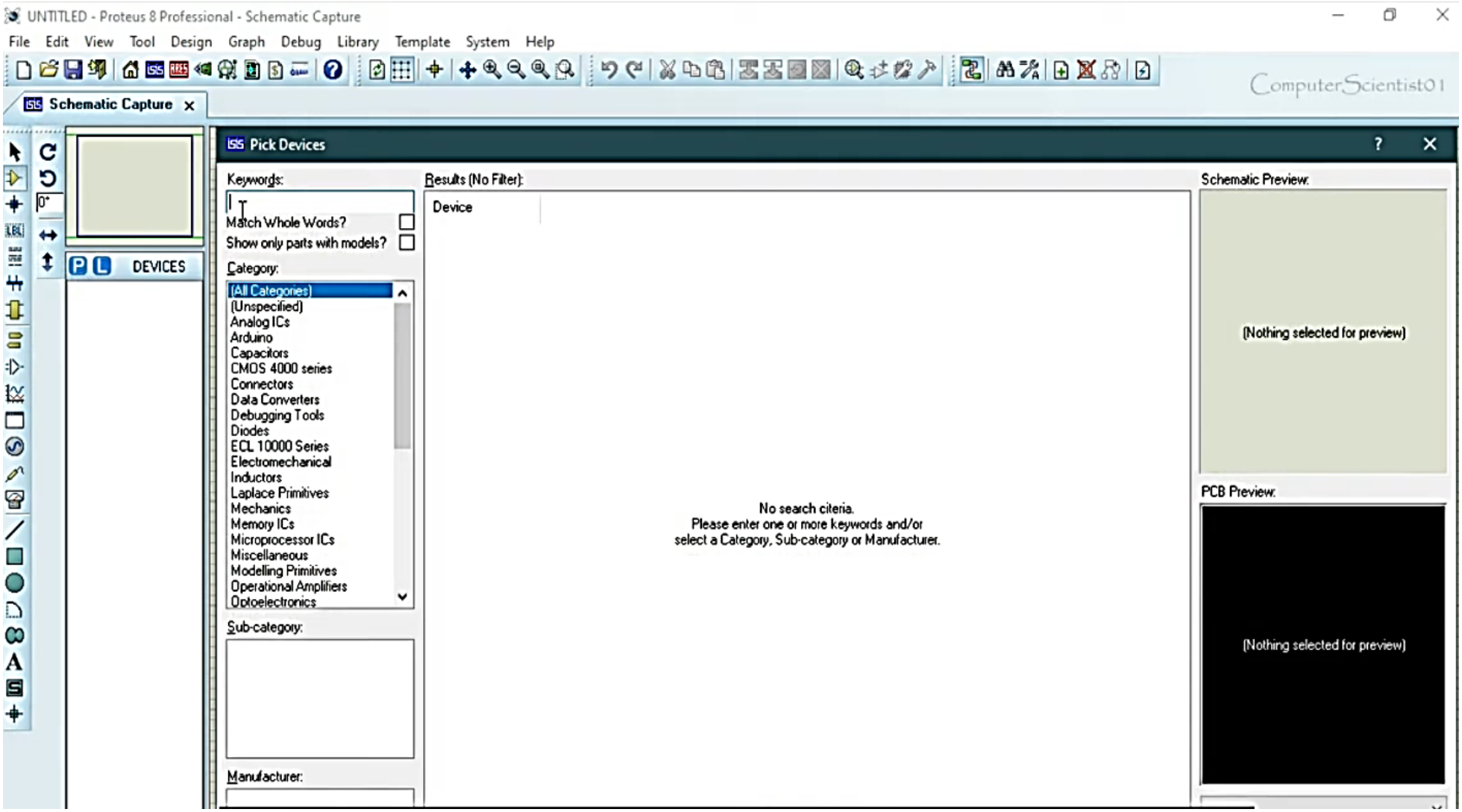
Embedded Systems and IOT

Proteus



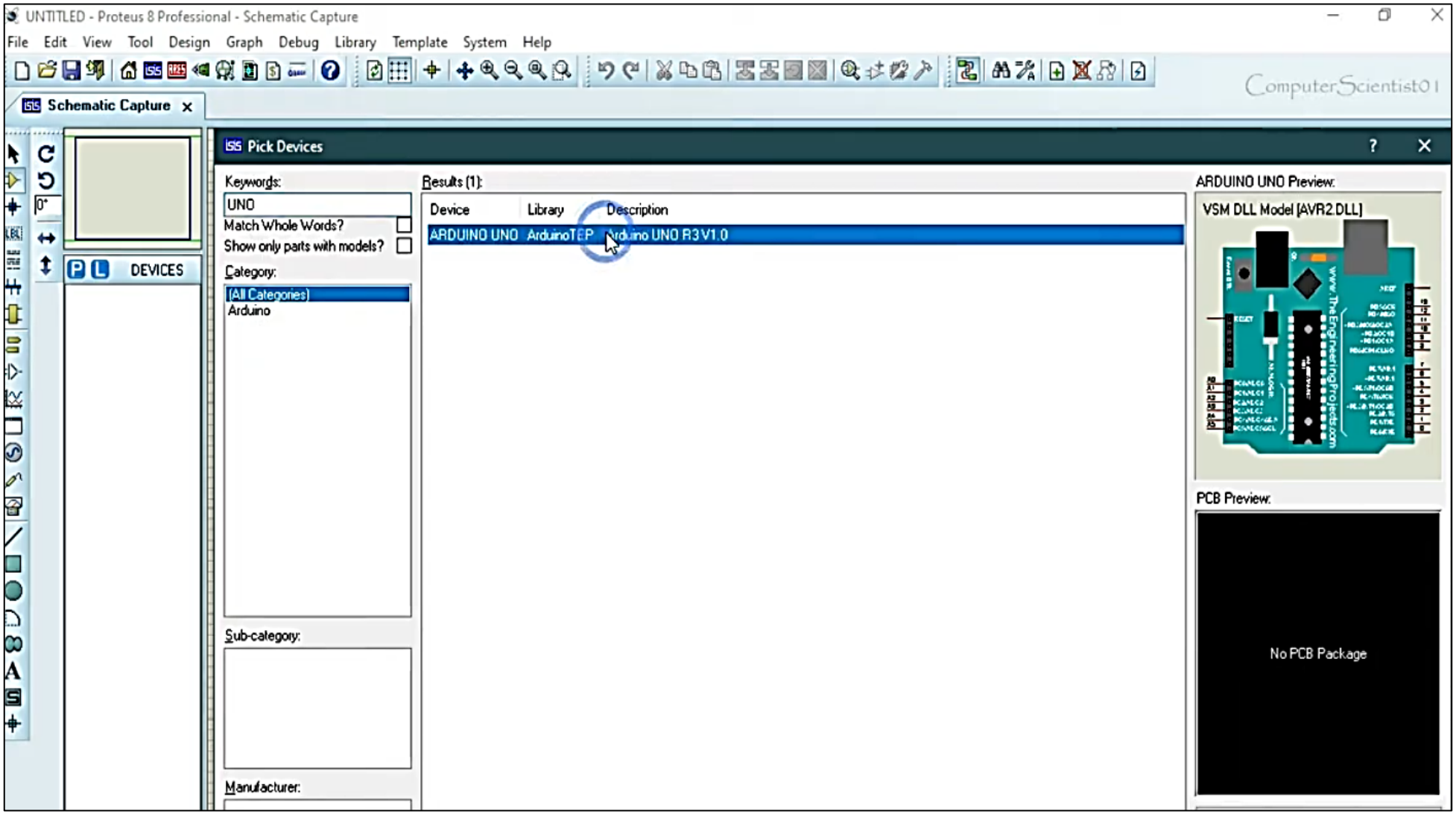
Embedded Systems and IOT

Proteus



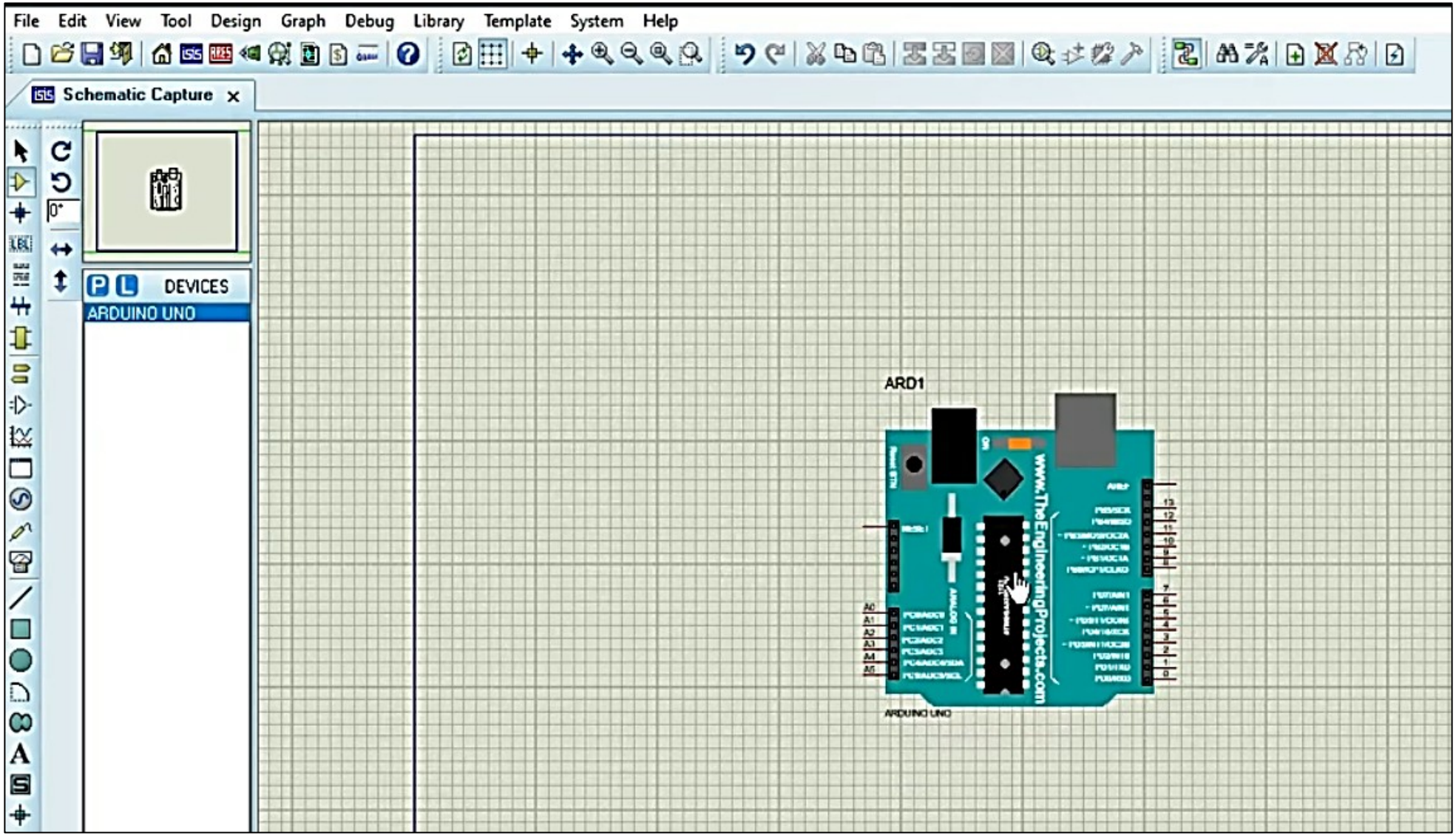
Embedded Systems and IOT

Proteus



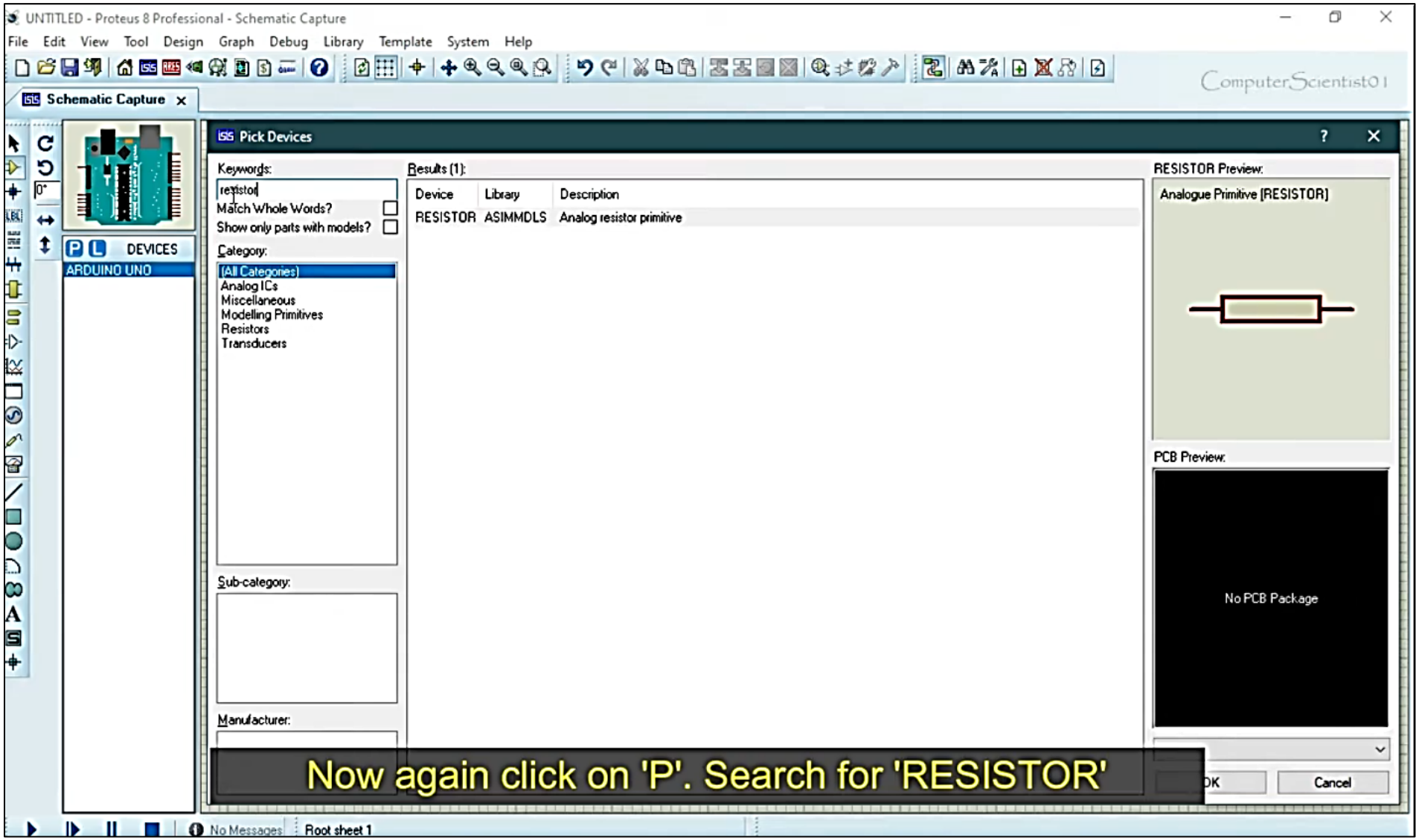
Embedded Systems and IOT

Proteus



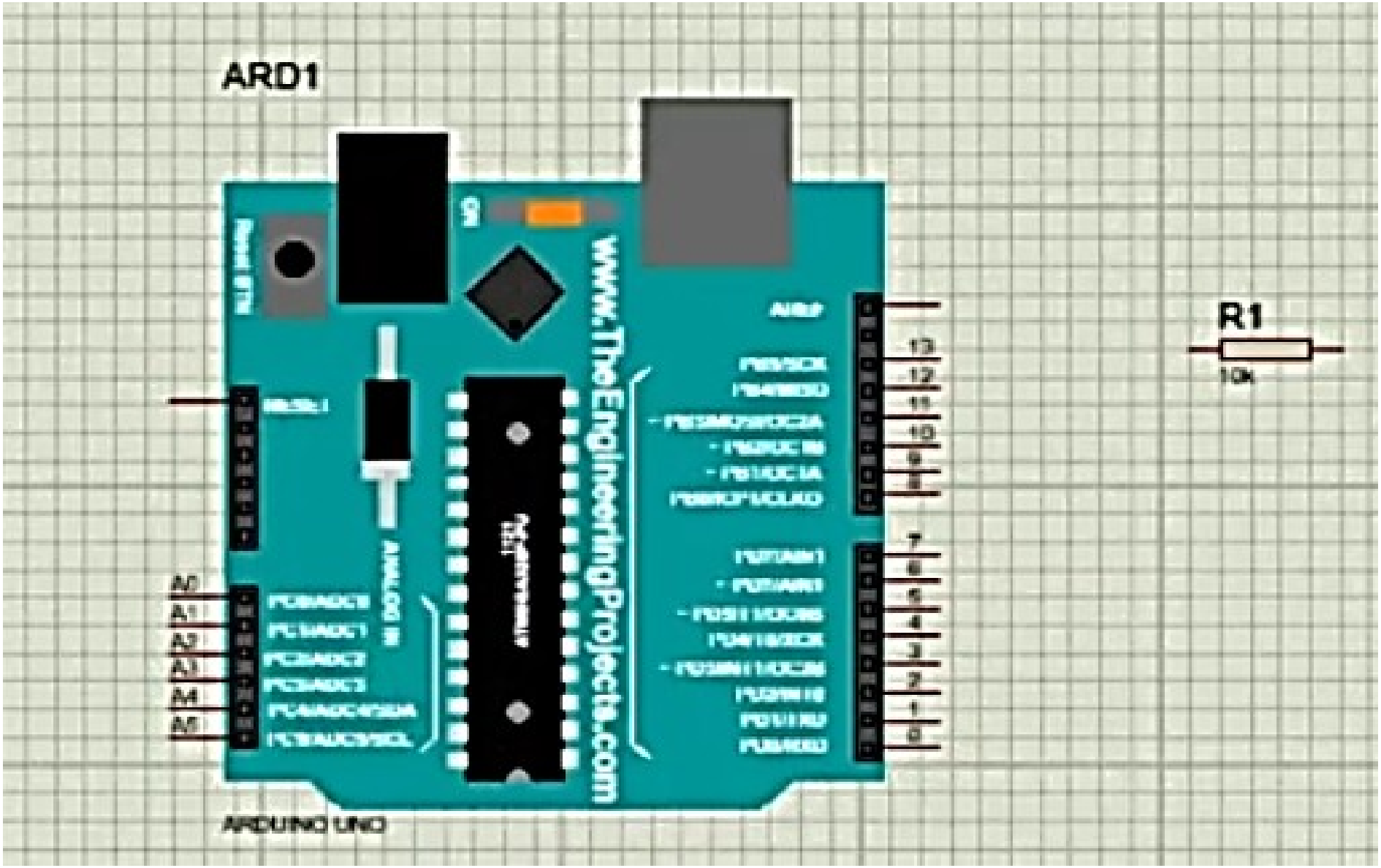
Embedded Systems and IOT

Proteus



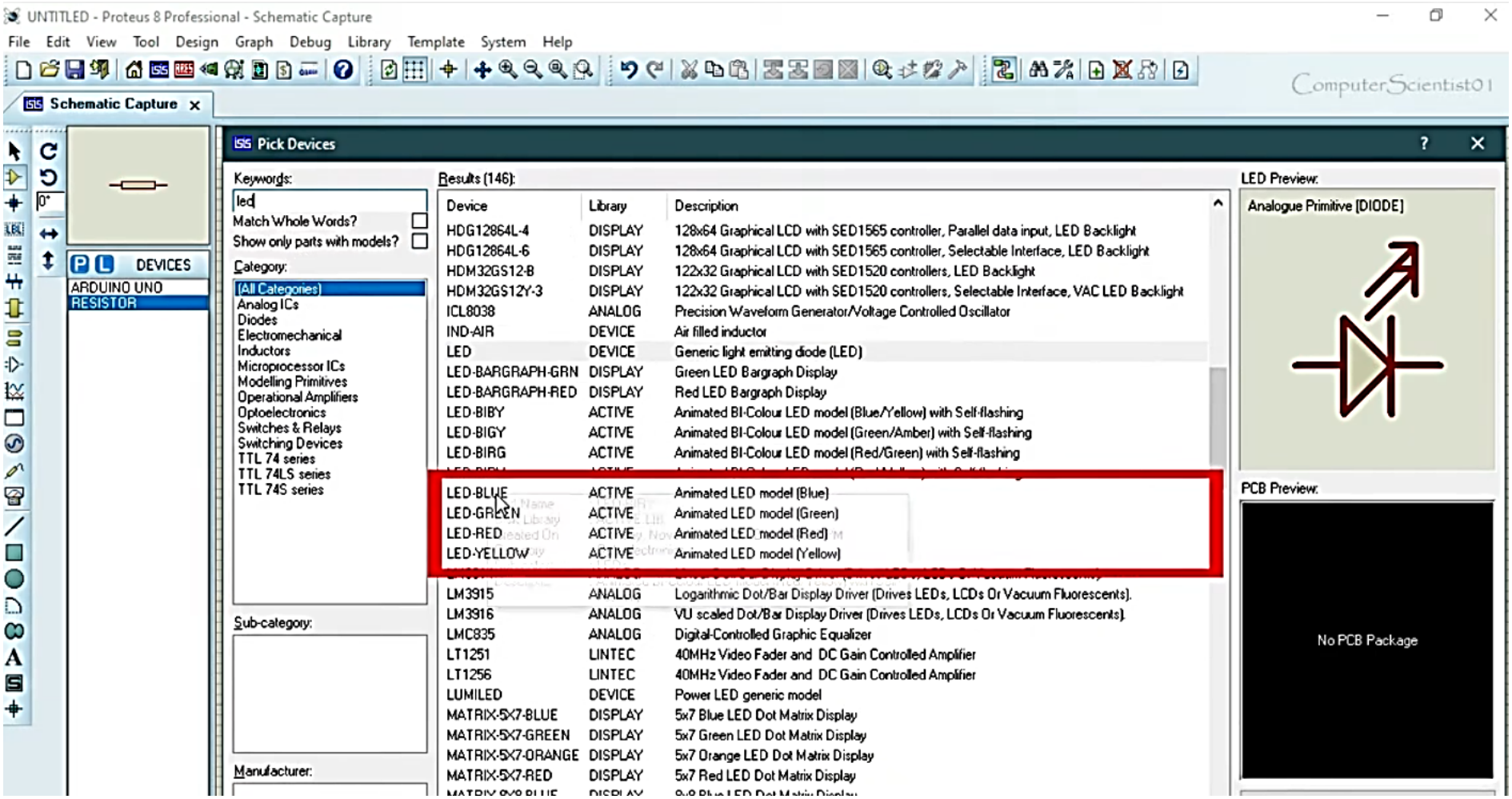
Embedded Systems and IOT

Proteus



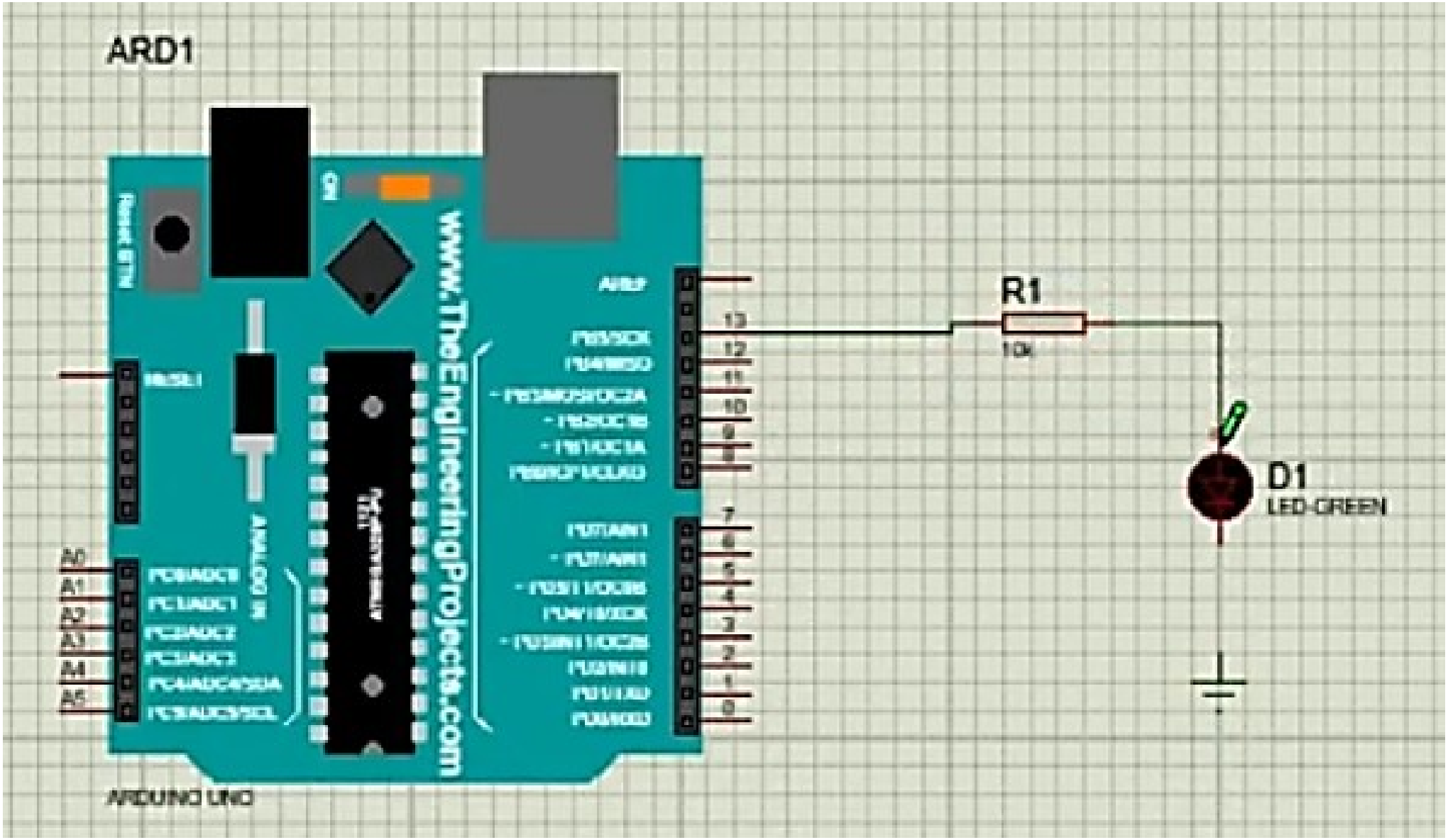
Embedded Systems and IOT

Proteus



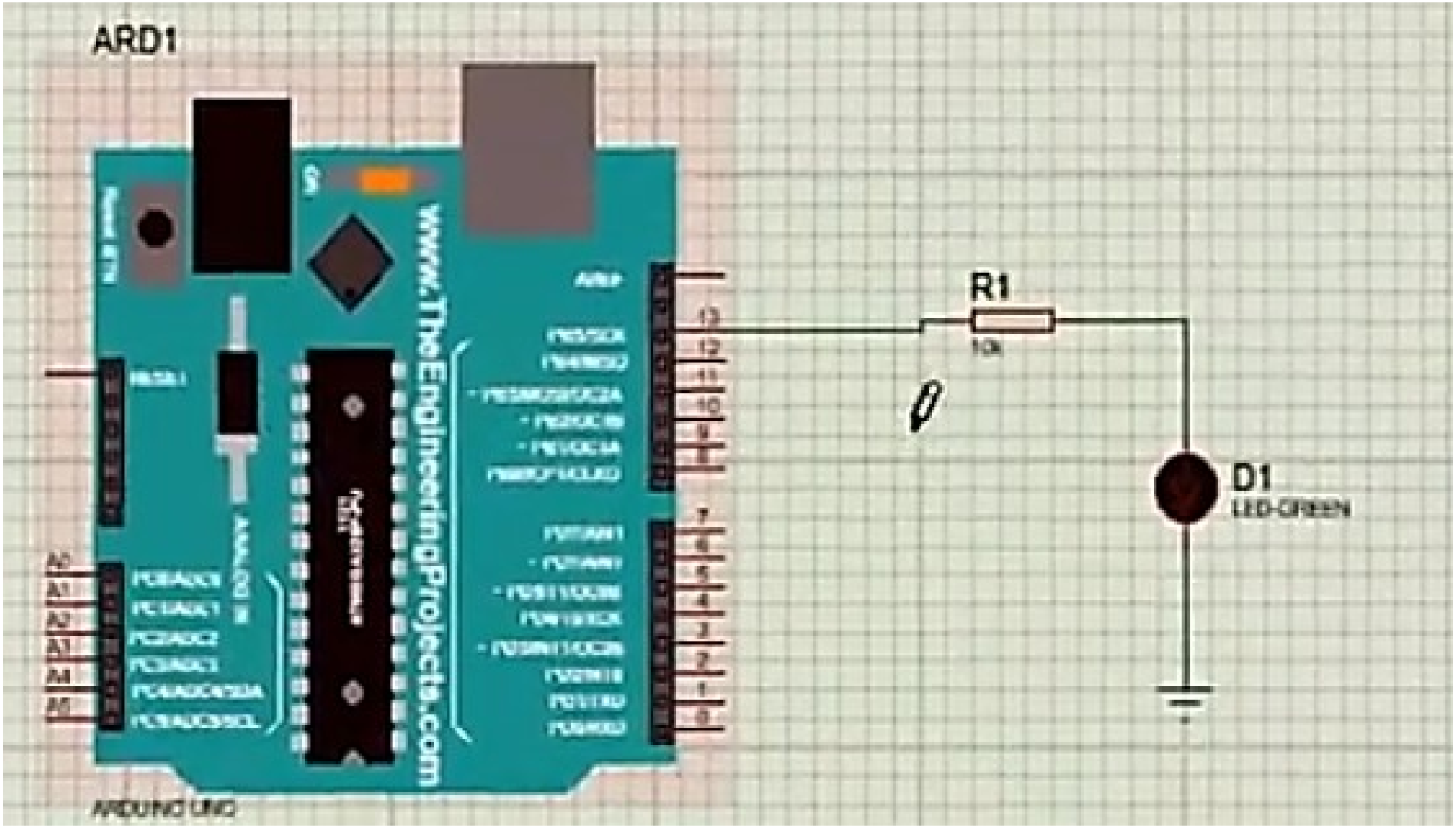
Embedded Systems and IOT

Proteus



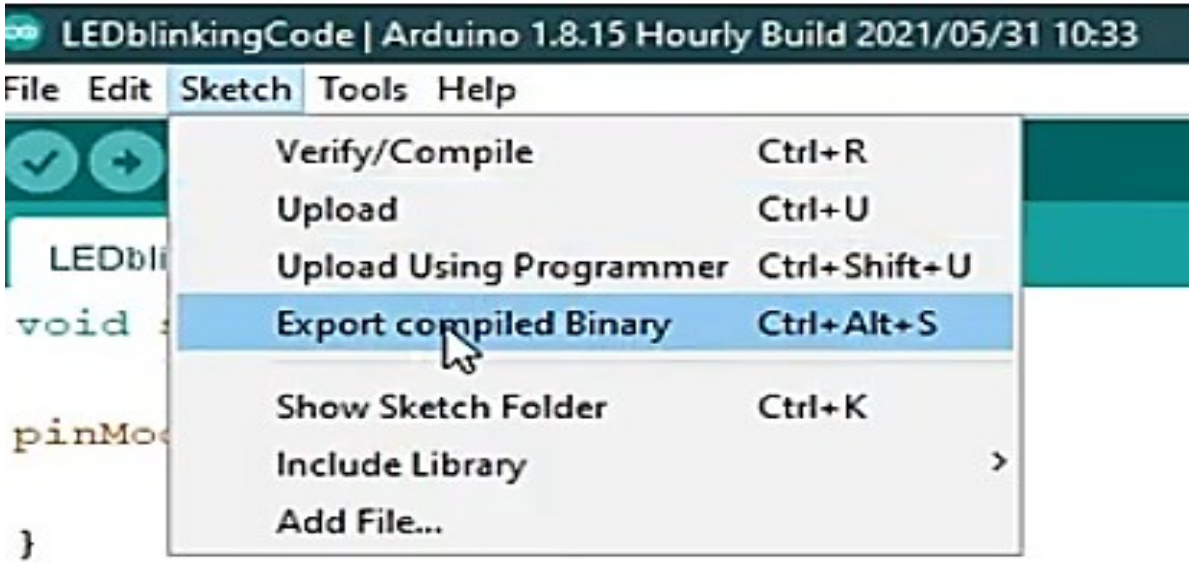
Embedded Systems and IOT

Proteus



Embedded Systems and IOT

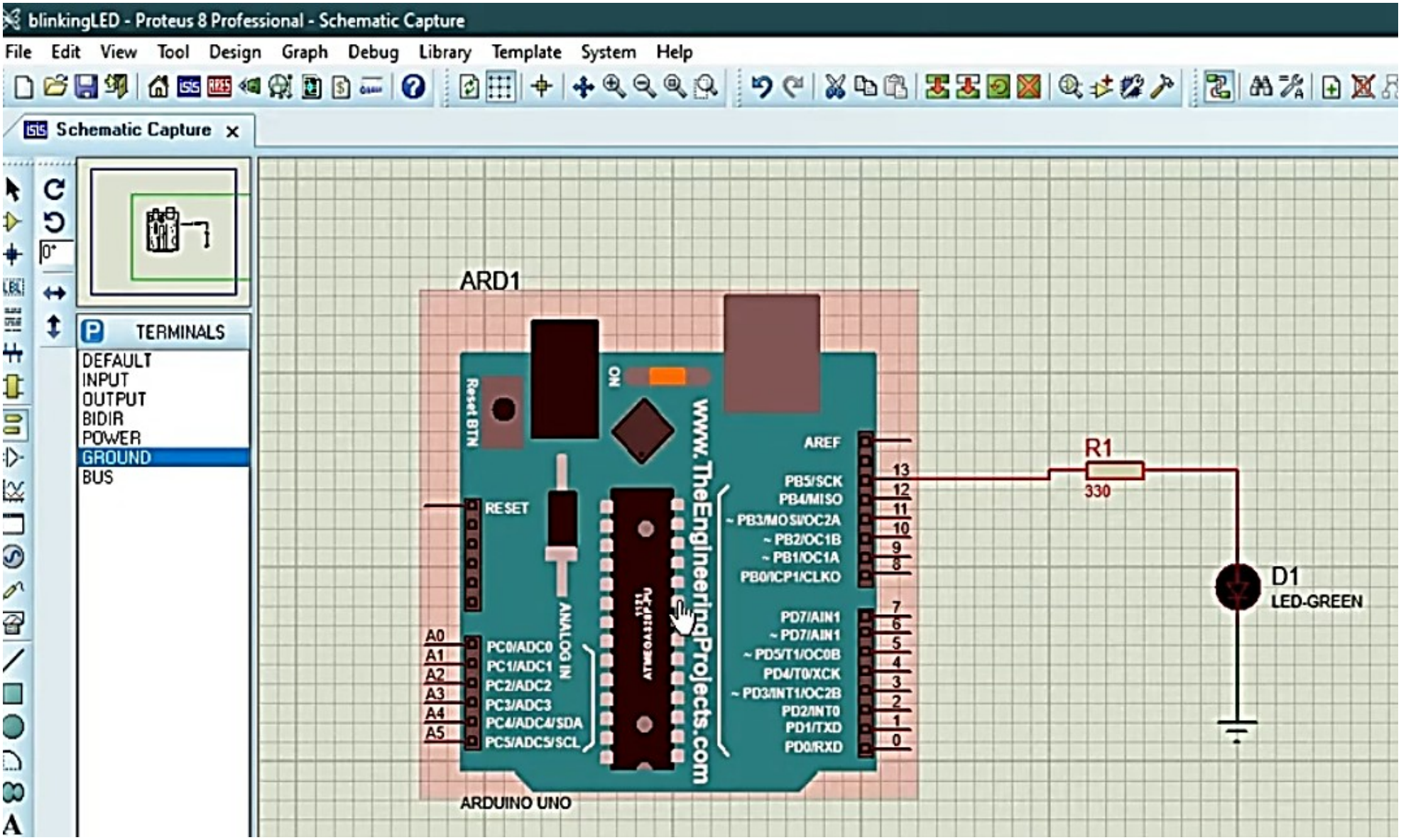
Proteus




```
void loop() {

digitalWrite(13, HIGH);

delay(1000);

digitalWrite(13, LOW);
delay(1000); // wait for a second

}
```

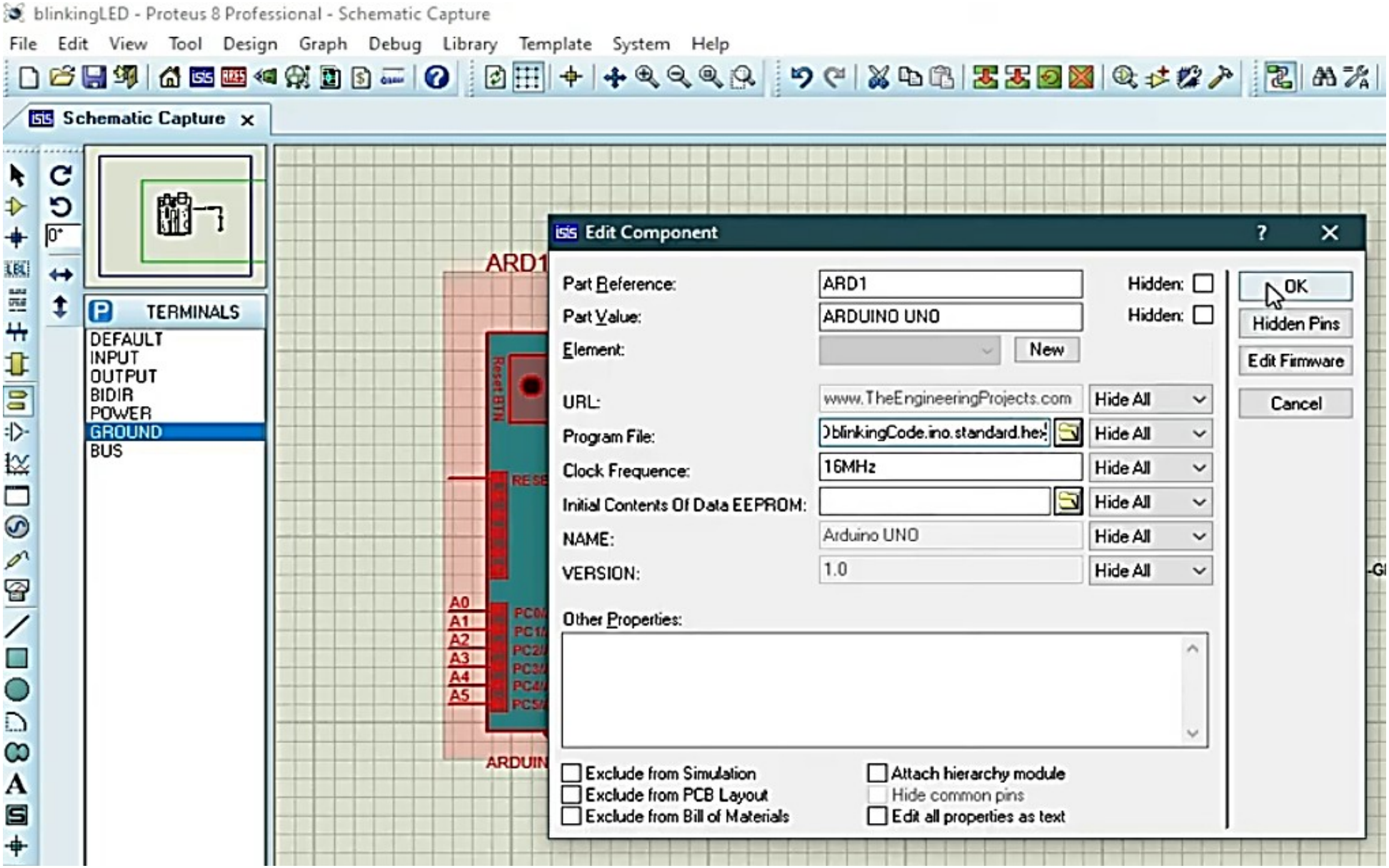
Embedded Systems and IOT

Proteus



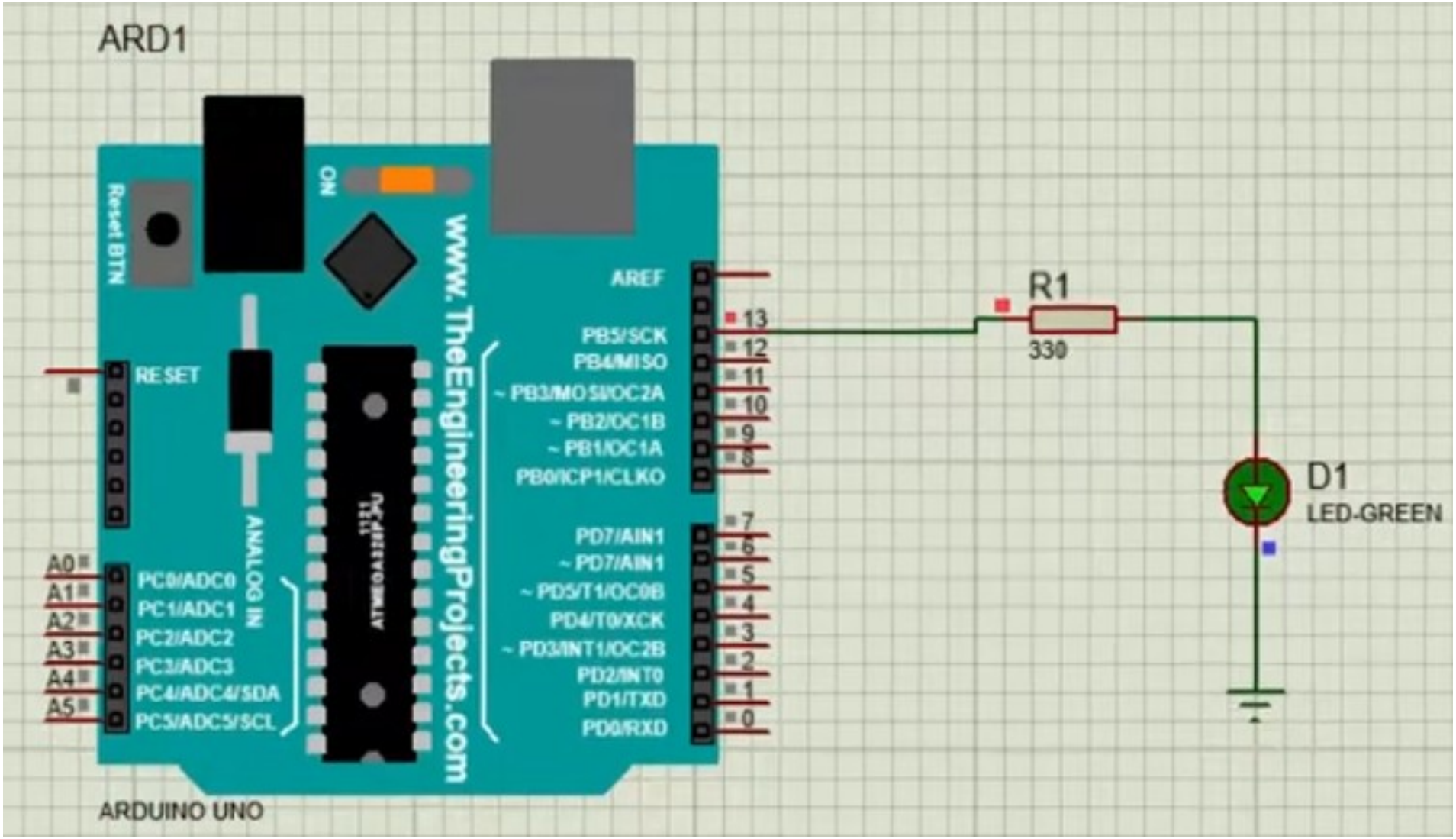
Embedded Systems and IOT

Proteus



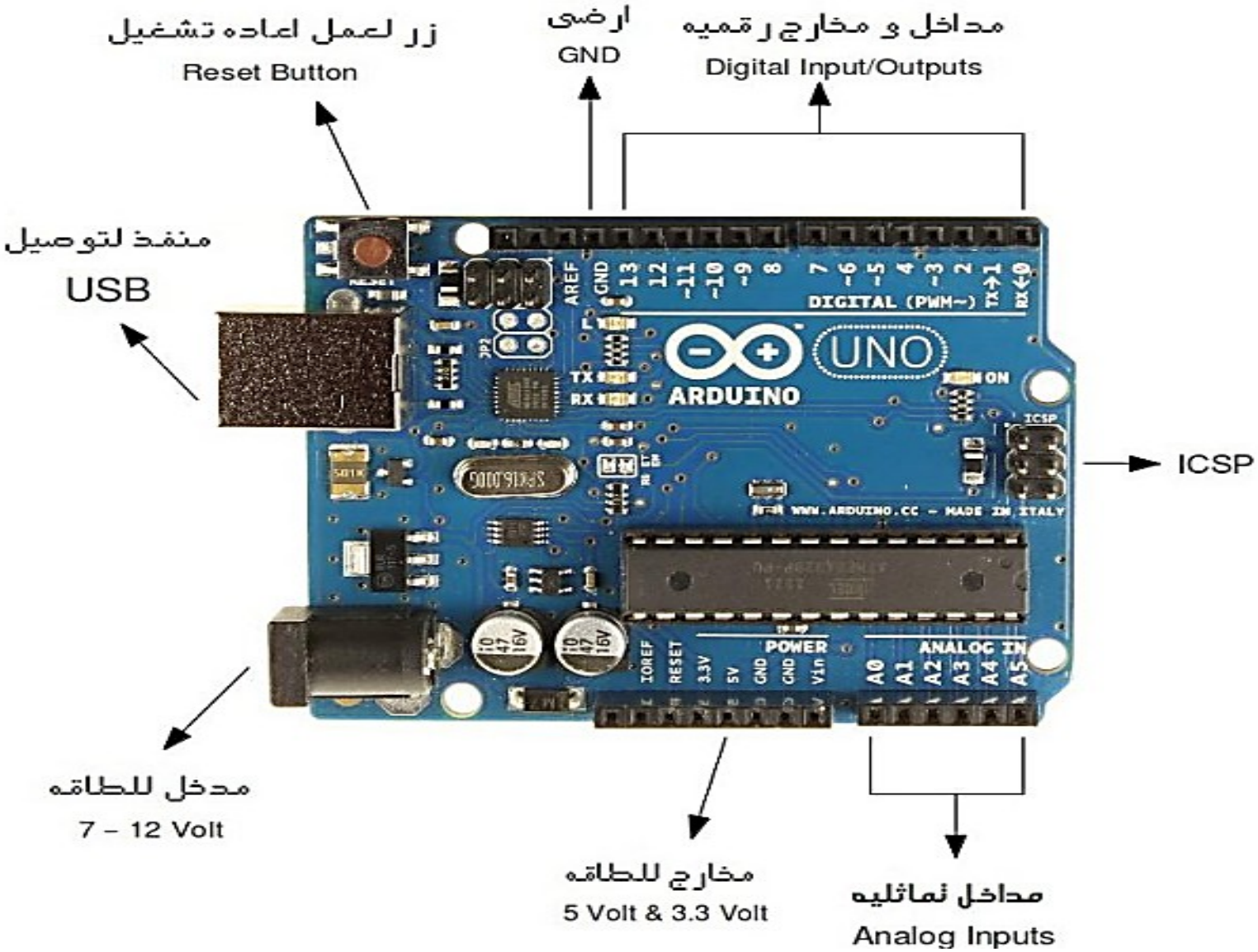
Embedded Systems and IOT

Proteus



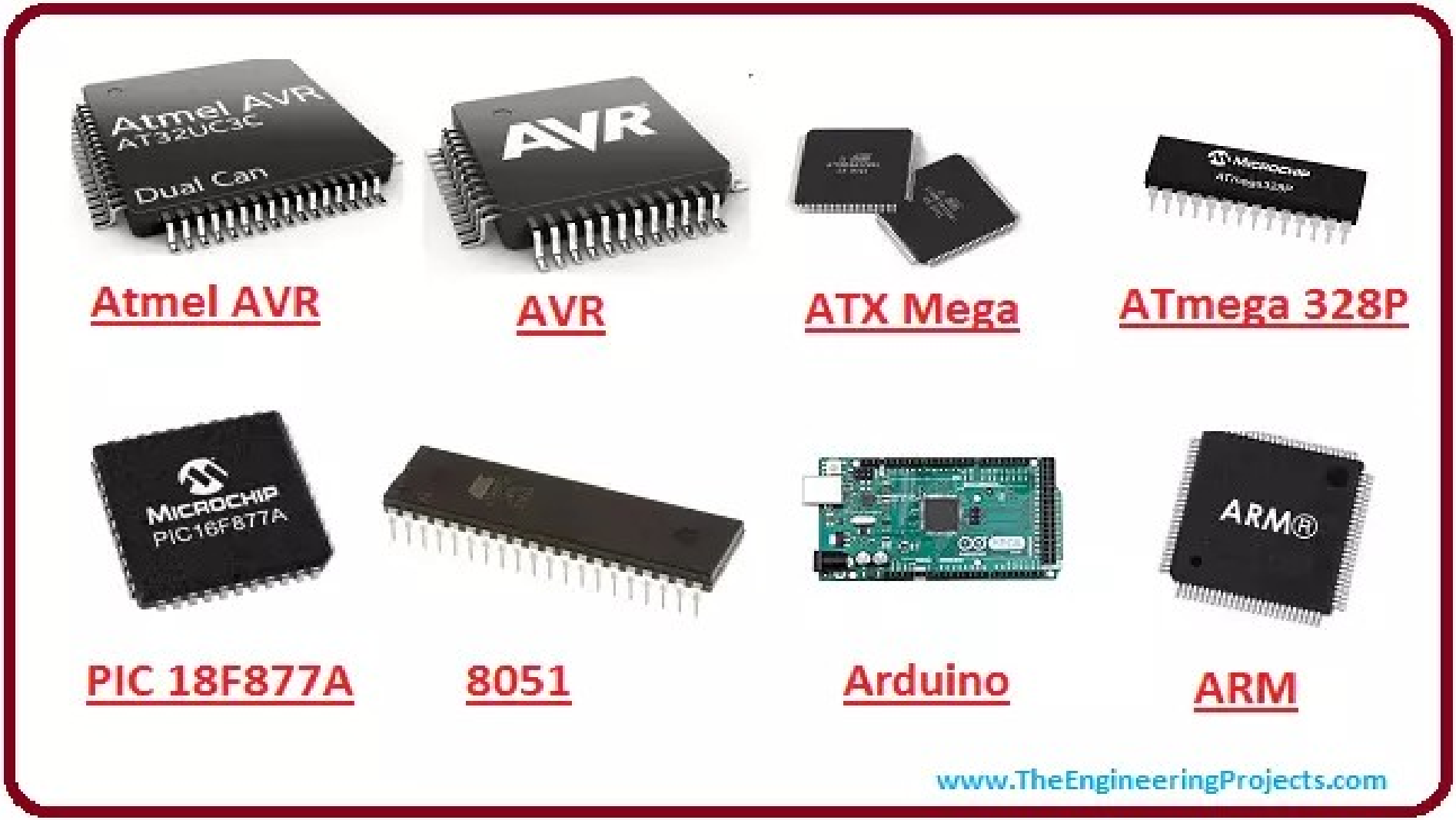
Embedded Systems and IOT

Arduino UNO



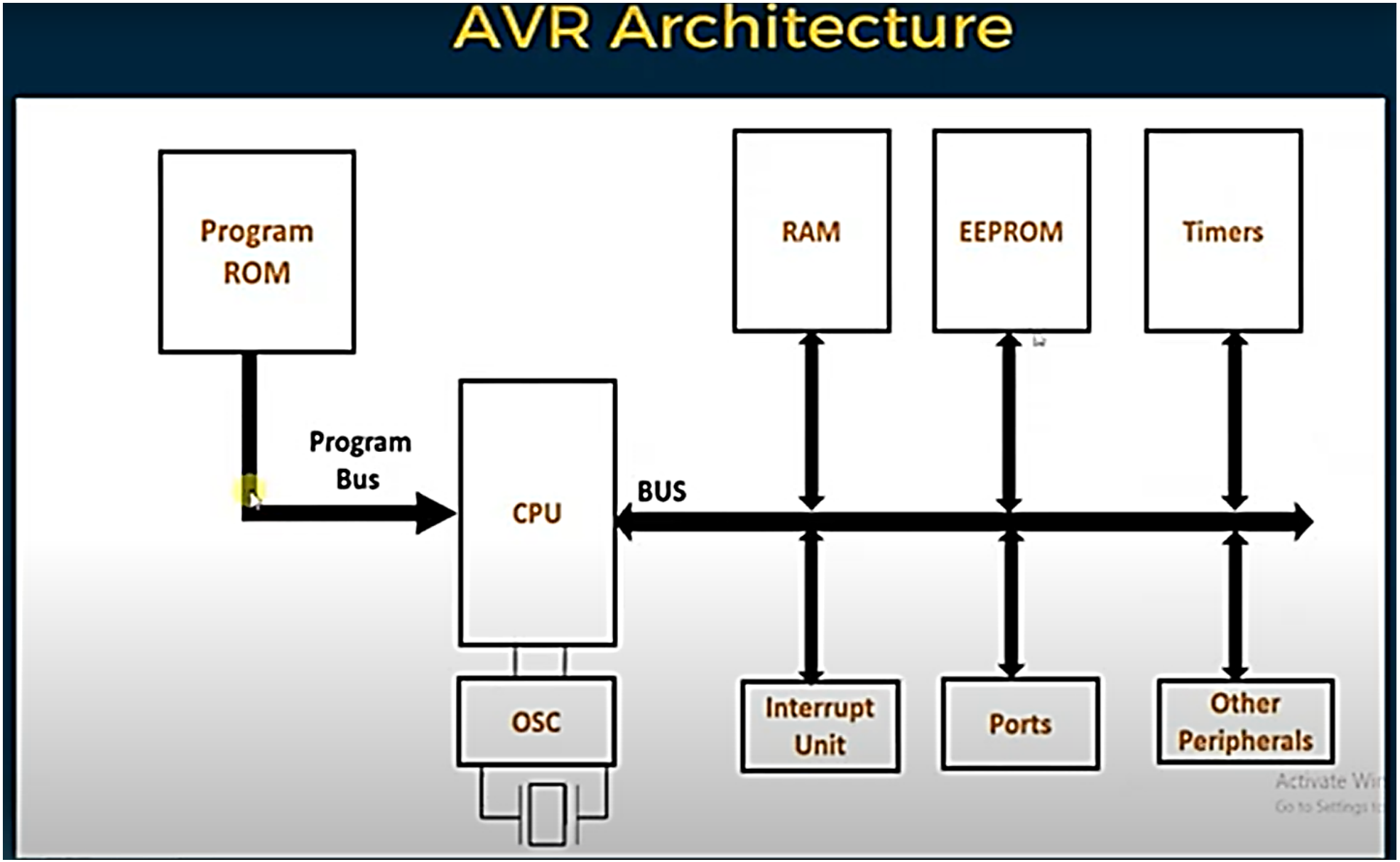
Embedded Systems and IOT

Introduction cont'd



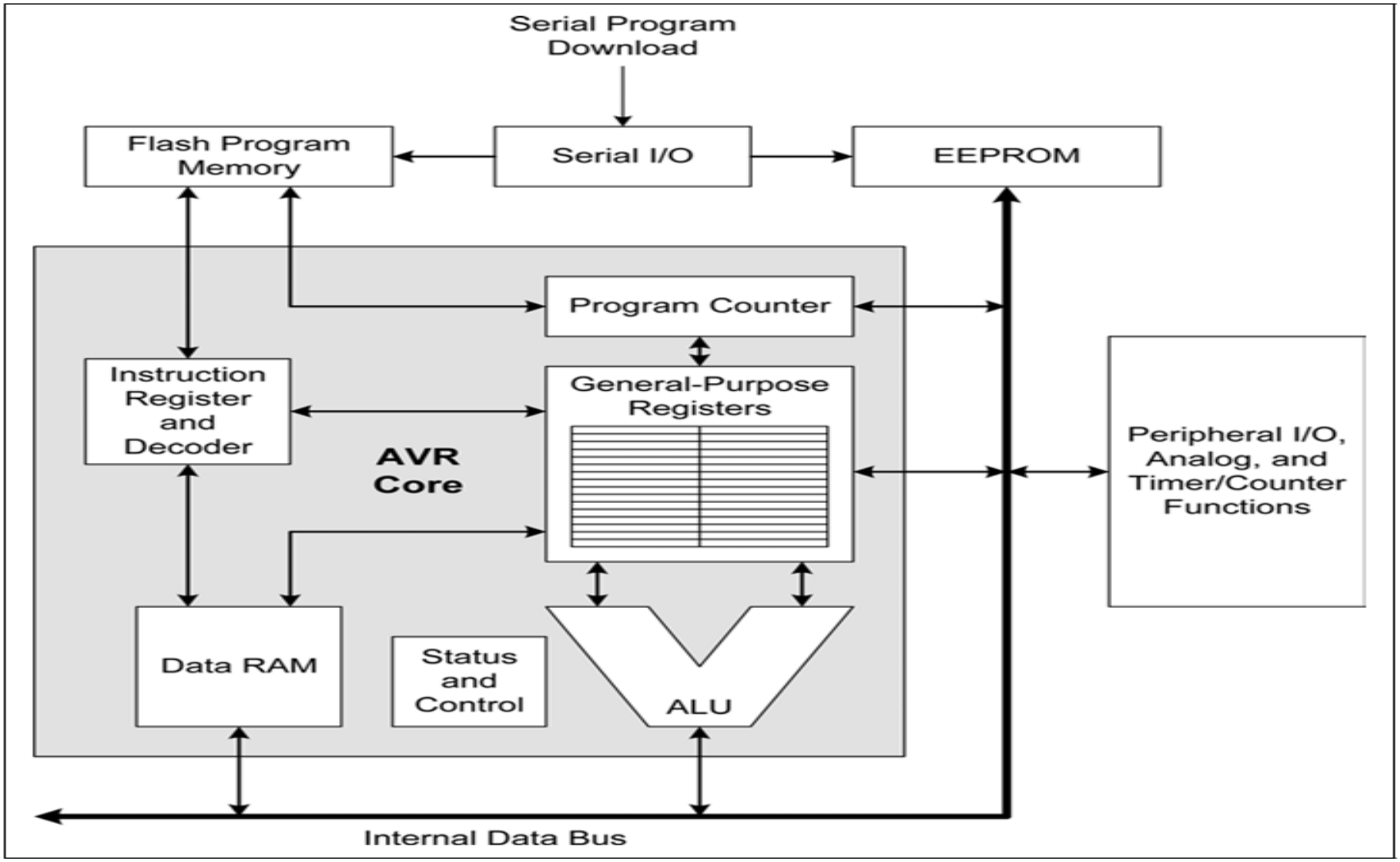
Embedded Systems and IOT

CH1: AVR microcontroller block diagram



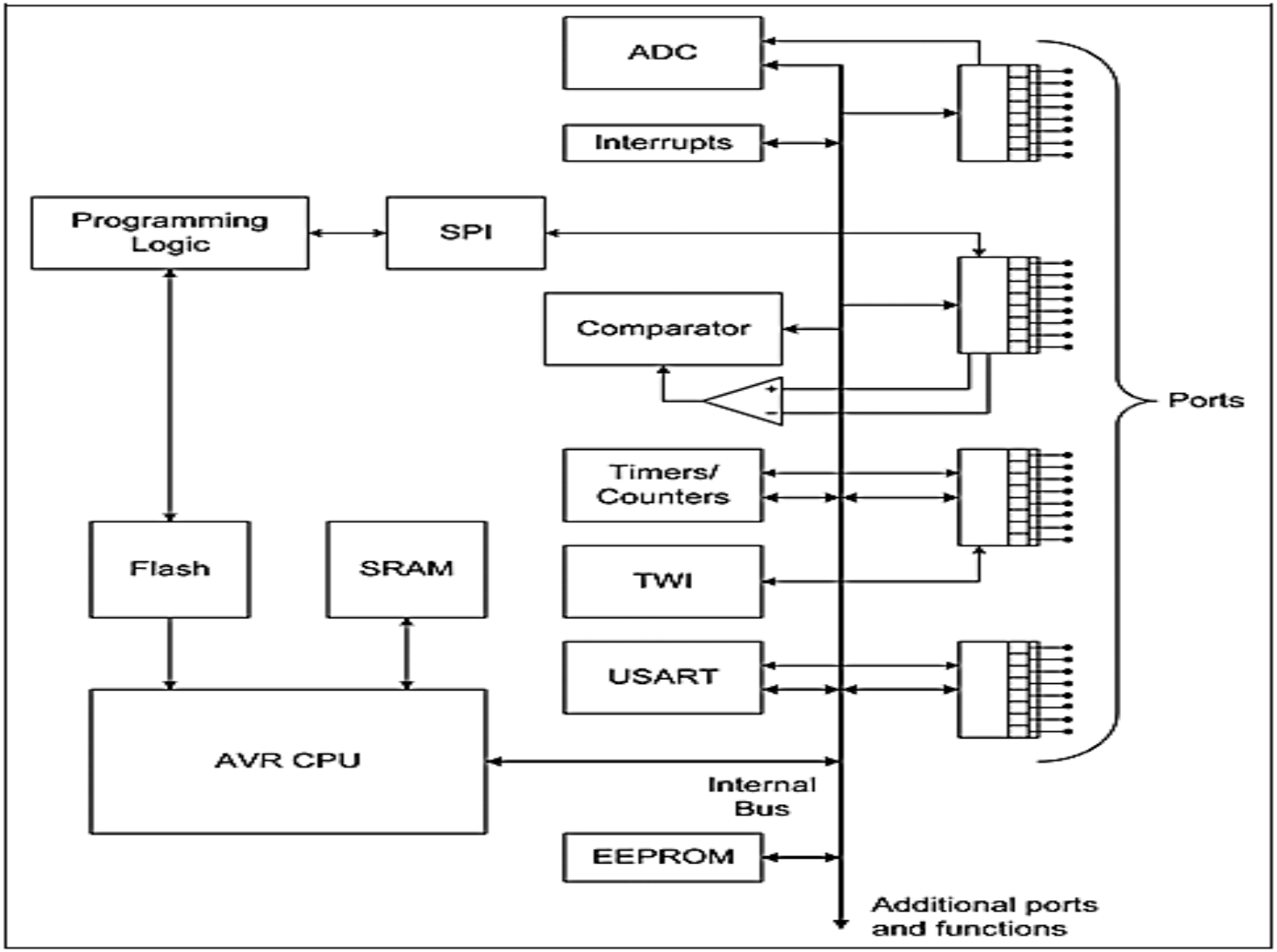
Embedded Systems and IOT

AVR CPU block diagram



Embedded Systems and IOT

Generic AVR microcontroller block diagram



Embedded Systems and IOT

Introduction: ARM microcontroller block diagram

The flash memory is used to store program code, the SRAM is used to hold transient data such as program variables and the stack.

EEPROM can hold data that needs to persist between software changes and power cycles.

The flash and EEPROM can be loaded externally, and both will retain their contents when the AVR is powered off.

The SRAM is volatile, and its contents will be lost when the AVR loses power. The heart of an AVR microcontroller is the 8-bit

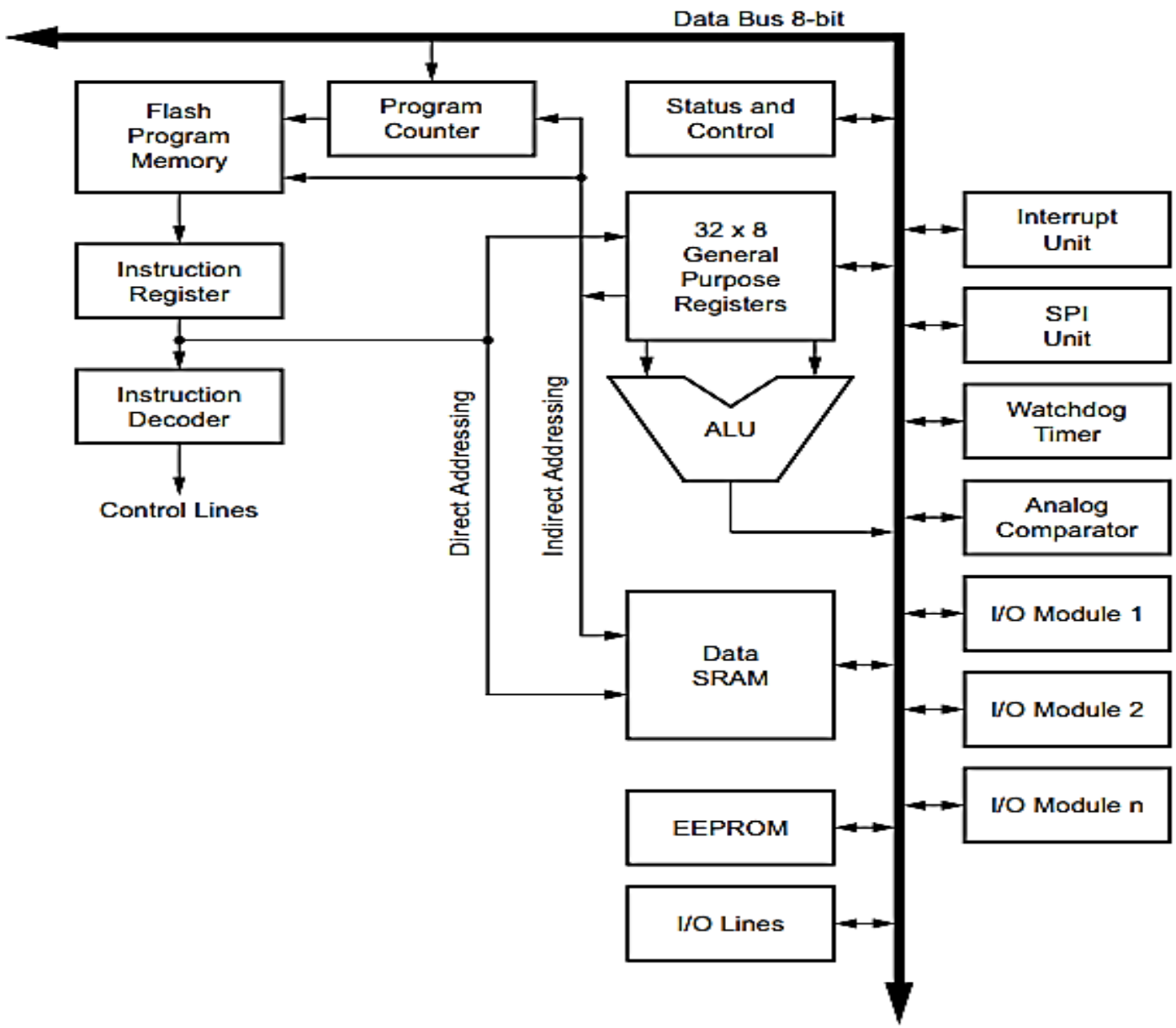
CPU, but what makes it a truly useful microcontroller is the built-in peripheral functions integrated into the IC with the CPU logic.

The peripheral functions of an AVR device vary from one type to another. Some have one timer; some have two or more (up to six for some types). Other parts may have a 10-bit A/D converter (ADC), whereas others feature a 12-bit converter. All AVR parts provide bidirectional I/O pins for discrete digital signals. Some versions also support a touchscreen and other types of interactive interfaces.

Embedded Systems and IOT

CH1: AVR microcontroller block diagram

Figure 6-1. Block Diagram of the AVR Architecture



Embedded Systems and IOT

Introduction: ARM microcontroller block diagram

The flash memory is used to store program code, the SRAM is used to hold transient data such as program variables and the stack.

EEPROM can hold data that needs to persist between software changes and power cycles.

The flash and EEPROM can be loaded externally, and both will retain their contents when the AVR is powered off.

The SRAM is volatile, and its contents will be lost when the AVR loses power. The heart of an AVR microcontroller is the 8-bit

CPU, but what makes it a truly useful microcontroller is the built-in peripheral functions integrated into the IC with the CPU logic.

The peripheral functions of an AVR device vary from one type to another. Some have one timer; some have two or more (up to six for some types). Other parts may have a 10-bit A/D converter (ADC), whereas others feature a 12-bit converter. All AVR parts provide bidirectional I/O pins for discrete digital signals. Some versions also support a touchscreen and other types of interactive interfaces.

Learning Objectives

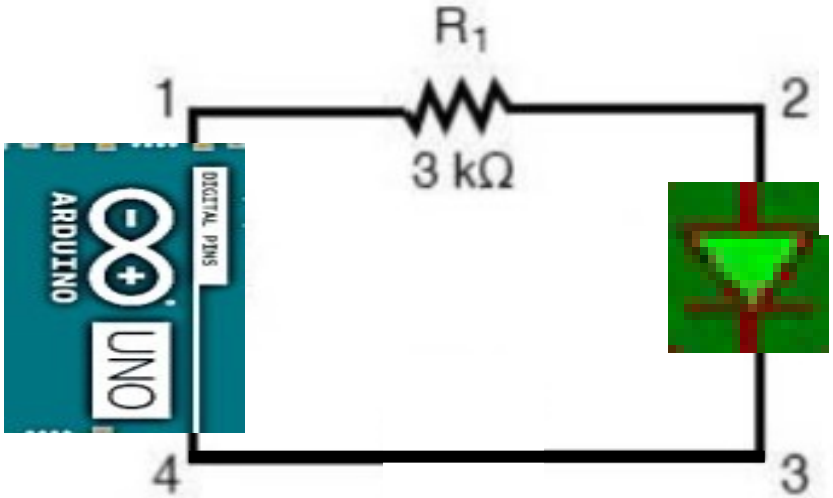
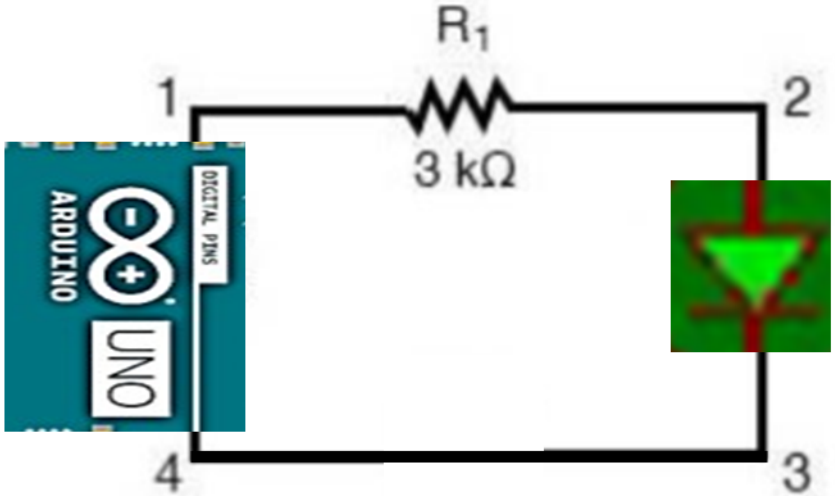
Upon completion of this lecture, you will be able to:

- **Design different applications of Embedded Systems**

Questions

THANK YOU

Embedded Systems and IOT

Embedded System Development Life cycle

1- System Design