SNS COLLEGE OF TECHNOLOGY

(An Autonomous Institution)

COIMBATORE-35

Accredited by NBA-AICTE and Accredited by NAAC – UGC with A++ Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

COURSE NAME: 19EEB303 / Microcontroller and its Applications

III YEAR / VI SEMESTER

Unit V – **ARDUINO**

Topic: Arduino IDE and sketch overview

Arduino IDE and sketch overview

In the getting started guide (Windows, MacOS, Linux), you uploaded a sketch that blinks an LED. A sketch is the name that Arduino uses for a program. It's the unit of code that is uploaded to and run on an Arduino board.

Variables

A *variable* is a place for storing a piece of data. It has a name, a type, and a value. For example, the line from the Blink sketch above declares a variable with the name ledPin, the type **int**, and an initial value of 13. It's being used to indicate which Arduino pin the LED is connected to. Every time the name ledPin appears in the code, its value will be retrieved. In this case, the person writing the program could have chosen not to bother creating the ledPin variable and instead have simply written 13 everywhere they needed to specify a pin number. The advantage of using a variable is that it's easier to move the LED to a different pin: you only need to edit the one line that assigns the initial value to the variable.

Often, however, the value of a variable will change while the sketch runs. For example, you could store the value read from an input into a variable.

Arduino IDE and sketch overview

Functions

A *function* (otherwise known as a *procedure* or *sub-routine*) is a named piece of code that can be used from elsewhere in a sketch. For example, here's the definition of the `setup()`.

The first line provides information about the function, like its name, "setup". The text before and after the name specify its return type and parameters: these will be explained later. The code between the `{` and `}` is called the *body* of the function: what the function does.

The *call* a function that's already been defined (either in your sketch or as part of the Arduino language). For example, the line `pinMode(ledPin, OUTPUT);` calls the `pinMode()` function, passing it two *parameters*: `ledPin` and `OUTPUT`. These parameters are used by the `pinMode()` function to decide which pin and mode to set.

Arduino IDE and sketch overview

pinMode(), digitalWrite(), and delay()

The pinMode() function configures a pin as either an input or an output. To use it, you pass it the number of the pin to configure and the constant INPUT or OUTPUT. When configured as an input, a pin can detect the state of a sensor like a pushbutton; this is discussed in the **Digital Read Serial tutorial**. As an output, it can drive an actuator like an LED.

The digitalWrite() functions outputs a value on a pin.

Arduino IDE and sketch overview

setup() and loop()

There are two special functions that are a part of every Arduino sketch: setup() and loop(). The setup() is called once, when the sketch starts. It's a good place to do setup tasks like setting pin modes or initializing libraries. The loop() function is called over and over and is heart of most sketches. You need to include both functions in your sketch, even if you don't need them for anything.

Arduino IDE and sketch overview

Arduino IDE and sketch overview

Arduino IDE and sketch overview

Arduino IDE and sketch overview