CHANDIGARH UNIVERSITY

Discover. Learn. Empower.

UNIVERSITY INSTITUTE OF ENGINEERING

Department of Computer Science & Engineering

Subject Name: INTERNET OF THINGS LAB

Subject Code: 20CSP-358

Submitted to:

Faculty name: Er. Jyoti Bala

Submitted by:

Name: Sahil Kaundal

UID: 21BCS8197

Section: 616

Group: A

INDEX

Ex. No	List of Experiments	Conduct (MM: 12)	Viva (MM: 10)	Record (MM: 8)	Total (MM: 30)	Remarks/Signature
1						
2						
3						
4						
5						
6						
7						
8						
9						
10						

Experiment 1

Student Name: Sahil Kaundal
Branch: CSE (Lateral Entry)
Semester: 6th
Subject Name: Internet of Things Lab

UID: 21BCS8197
Section/Group: 616/A
Date of Performance: 20/02/2023
Subject Code: 20CSP-358

1. Aim:

Familiarization with Arduino/Raspberry Pi hardware and perform necessary software installation.

2. Objective:

- To study hardware and software related to IoT
- To understand the function of Node MCU, Arduino Uno and Raspberry Pi.

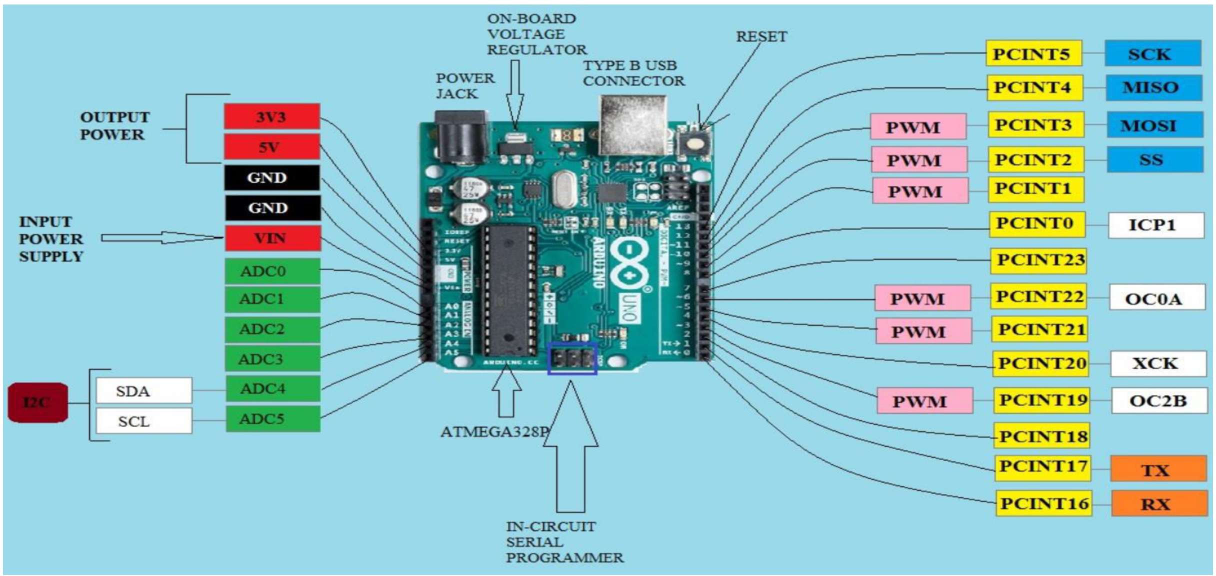
3. System Requirements:

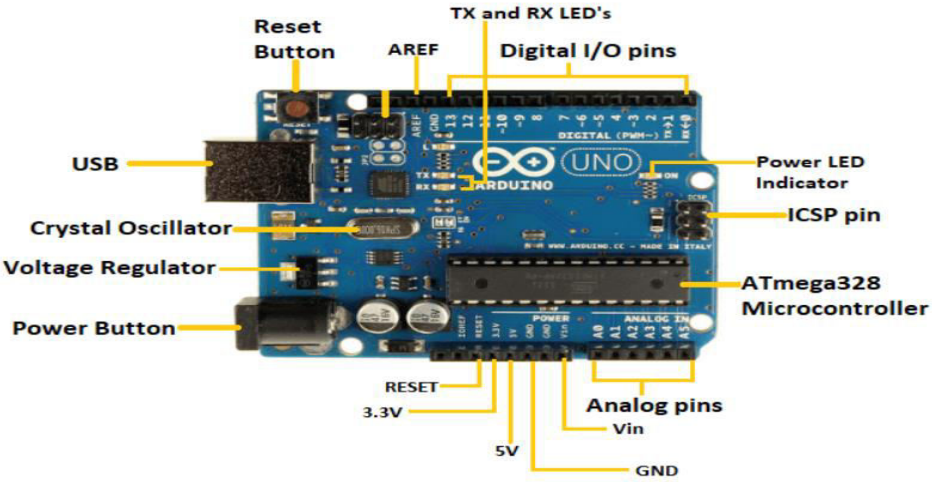
- Windows 7 or above.
- Google Chrome
- Arduino IDE

4. Steps/Program:

An Arduino is a micro controller-based kit. It is basically used in communications and in controlling or operating many devices. Arduino UNO board is the most popular board in the Arduino board family. In addition, it is the best board to get started with electronics and coding. Some boards look a bit different from the one given below, but most Arduino's have majority of these components in common. It consists of two memories- Program memory and the data memory. The code is stored in the flash program memory, whereas the data is stored in the data memory.

Arduino Uno consists of 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button





1. Power USB Arduino board can be powered by using the USB cable from your computer. All you need to do is connect the USB cable to the USB connection (1).
2. Power (Barrel Jack) Arduino boards can be powered directly from the AC mains power supply by connecting it to the Barrel Jack (2).
3. Voltage Regulator The function of the voltage regulator is to control the voltage given to the Arduino board and stabilize the DC voltages used by the processor and other elements.
4. Crystal Oscillator The crystal oscillator helps Arduino in dealing with time issues. How does Arduino calculate time? The answer is, by using the crystal oscillator. The number printed on top of the Arduino crystal is 16.000H9H. It tells us that the frequency is 16,000,000 Hertz or 16 MHz.
5. 17.Arduino Reset You can reset your Arduino board, i.e., start your program from the beginning. You can reset the UNO board in two ways. First, by using the reset button (17) on the board. Second, you can connect an external reset button to the Arduino pin labelled RESET (5).

6,7,8,9.Pins (3.3, 5, GND, Vin)

- 3.3V (6) − Supply 3.3 output volt
- 5V (7) − Supply 5 output volt
- Most of the components used with Arduino board works fine with 3.3 volt and 5 volt.
- GND (8)(Ground) − There are several GND pins on the Arduino, any of which can be used to ground your circuit.
- Vin (9) − This pin also can be used to power the Arduino board from an external power source, like AC mains power supply.

10. Analog pins The Arduino UNO board has six analog input pins A0 through A5. These pins can read the signal from an analog sensor like the humidity sensor or temperature sensor and convert it into a digital value that can be read by the microprocessor.
11. Main micro controller Each Arduino board has its own micro controller (11). You can assume it as the brain of your board. The main IC (integrated circuit) on the Arduino is slightly different from board to board. The micro controllers are usually of the ATMEL Company. You must know what IC your board has before loading up a new program from the Arduino IDE. This information is available on the top of the IC. For more details about the IC construction and functions, you can refer to the data sheet.
12. ICSP pin Mostly, ICSP (12) is an AVR, a tiny programming header for the Arduino consisting of MOSI, MISO, SCK, RESET, VCC, and GND. It is often referred to as an SPI (Serial Peripheral Interface), which could be considered as an "expansion" of the output. Actually, you are slaving the output device to the master of the SPI bus.
13. Power LED indicator This LED should light up when you plug your Arduino into a power source to indicate that your board is powered up correctly. If this light does not turn on, then there is something wrong with the connection.
14. TX and RX LEDs On your board, you will find two labels: TX (transmit) and RX (receive). They appear in two places on the Arduino UNO board. First, at the digital pins 0 and 1, to indicate the pins

responsible for serial communication. Second, the TX and RX led (13). The TX led flashes with different speed while sending the serial data. The speed of flashing depends on the baud rate used by the board. RX flashes during the receiving process.

15. Digital I/O

- The Arduino UNO board has 14 digital I/O pins (15) (of which 6 provide PWM (Pulse Width Modulation) output. These pins can be configured to work as input digital pins to read logic values (0 or 1) or as digital output pins to drive different modules like LEDs, relays, etc. The pins labeled "~" can be used to generate PWM.

16. AREF

    AREF stands for Analog Reference. It is sometimes, used to set an external reference voltage (between 0 and 5 Volts) as the upper limit for the analog input pins

Program an Arduino:

- The most important advantage with Arduino is the programs can be directly loaded to the device without requiring any hardware programmer to burn the program.
- This is done because of the presence of the 0.5KB of Boot-loader which allows the program to be burned into the circuit.
- All we have to do is to download the Arduino software and writing the code.
- The Arduino tool window consists of the toolbar with the buttons like verify, upload, new, open, save, serial monitor.
- It also consists of a text editor to write the code, a message area which displays the feedback like showing the errors, the text console which displays the output and a series of menus like the File, Edit, Tools menu.

Steps to program an Arduino

1. Programs written in Arduino are known as sketches. A basic sketch consists of 3 parts

- Declaration of Variables
- Initialisation: It is written in the setup () function.
- Control code: It is written in the loop () function.

2. The sketch is saved with .ino extension. Any operations like verifying, opening a sketch, saving a sketch can be done using the buttons on the toolbar or using the tool menu.
3. The sketch should be stored in the sketchbook directory.
4. Chose the proper board from the tools menu and the serial port numbers.
5. Click on the upload button or chose upload from the tools menu. Thus the code is uploaded by the boot loader onto the micro controller.

Basic Adruino functions are:

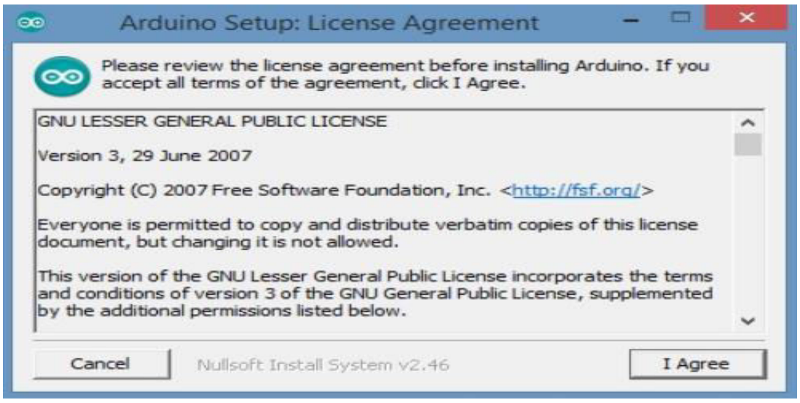
1. digitalRead(pin): Reads the digital value at the given pin.
2. digitalWrite(pin, value): Writes the digital value to the given pin.
3. pinMode(pin, mode): Sets the pin to input or output mode.
4. analogRead(pin): Reads and returns the value.
5. analogWrite(pin, value): Writes the value to that pin.
6. serial.begin(baud rate): Sets the beginning of serial communication by setting the bit rate.

Advantages of Arduino Board

- It is inexpensive.
- It comes with an open source hardware feature which enables users to develop their own kit using already available one as a reference source.
- The Arduino software is compatible with all types of operating systems like Windows, Linux, and Macintosh etc.
- It also comes with open source software feature which enables experienced software developers to use the Arduino code to merge with the existing programming language libraries and can be extended and modified.
- It is easy to use for beginners.
- We can develop an Arduino based project which can be completely stand alone or projects which involve direct communication with the software loaded in the computer.
- It comes with an easy provision of connecting with the CPU of the computer using serial communication over USB as it contains built in power and reset circuitry.

INSTALLING THE ARDUINO IDE

1. Visit http://www.arduino.cc/en/main/software to download the latest Arduino IDE version for your computer's operating system. There are versions for Windows, Mac, and Linux systems. At the download page, click on the "Windows Installer" option for the easiest installation.
2. Save the .exe file to your hard drive.
3. Open the .exe file.
4. Click the button to agree to the licensing agreement:



Arduino Setup: License Agreement

Please review the license agreement before installing Arduino. If you accept all terms of the agreement, click I Agree.

GNU LESSER GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

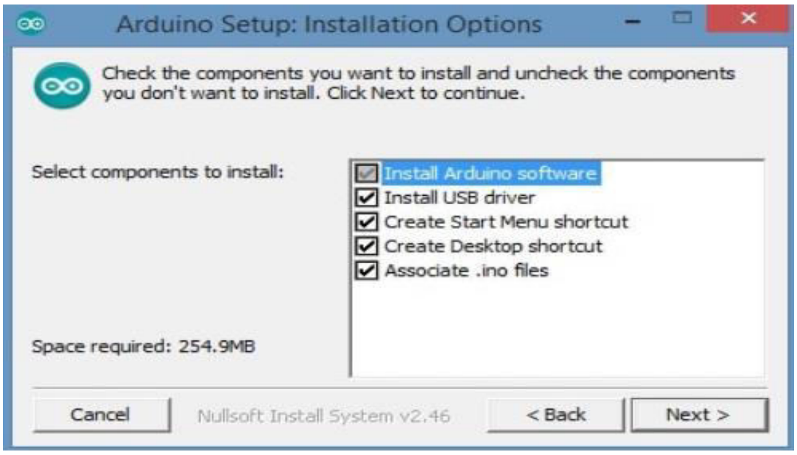
Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

Nullsoft Install System v2.46

5. Decide which components to install, then click "Next":



Arduino Setup: Installation Options

Check the components you want to install and uncheck the components you don't want to install. Click Next to continue.

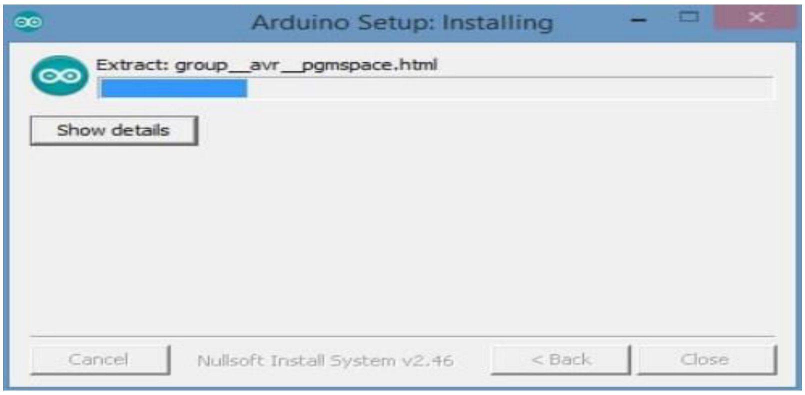
Select components to install:

☑ Install Arduino software
☑ Install USB driver
☑ Create Start Menu shortcut
☑ Create Desktop shortcut
☑ Associate .ino files

Space required: 254.9MB

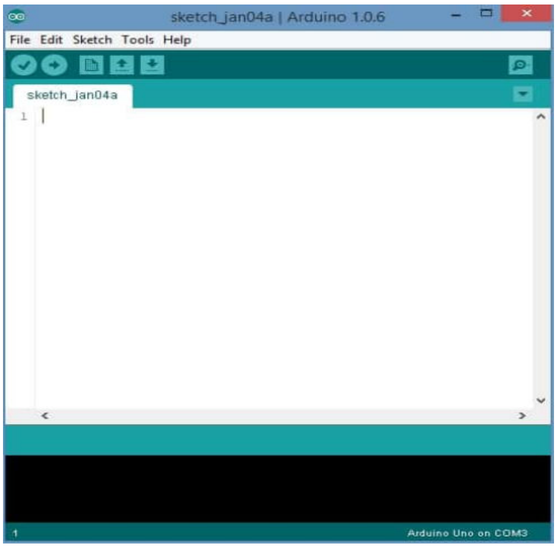
Nullsoft Install System v2.46

6. Select which folder to install the program to, then click "Install":
7. Wait for the program to finish installing, then click "Close":



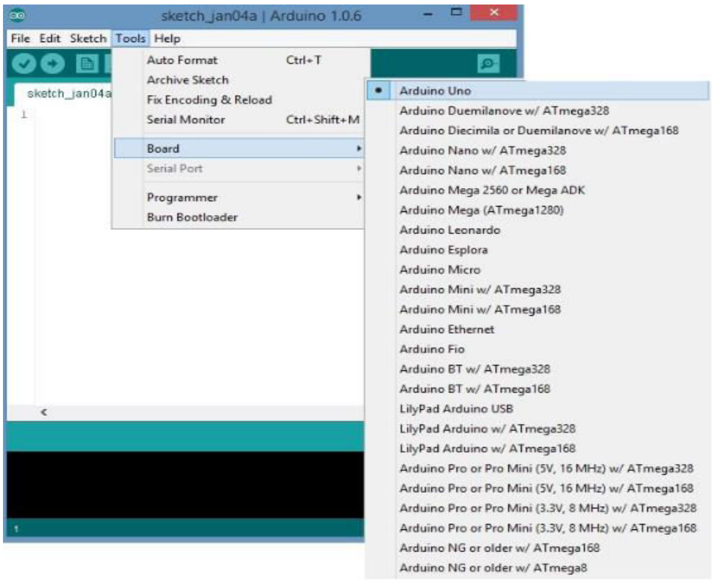
Extract: group__avr__pgmspace.html

8. Now find the Arduino shortcut on your Desktop and click on it. The IDE will open up and you'll see the code editor:



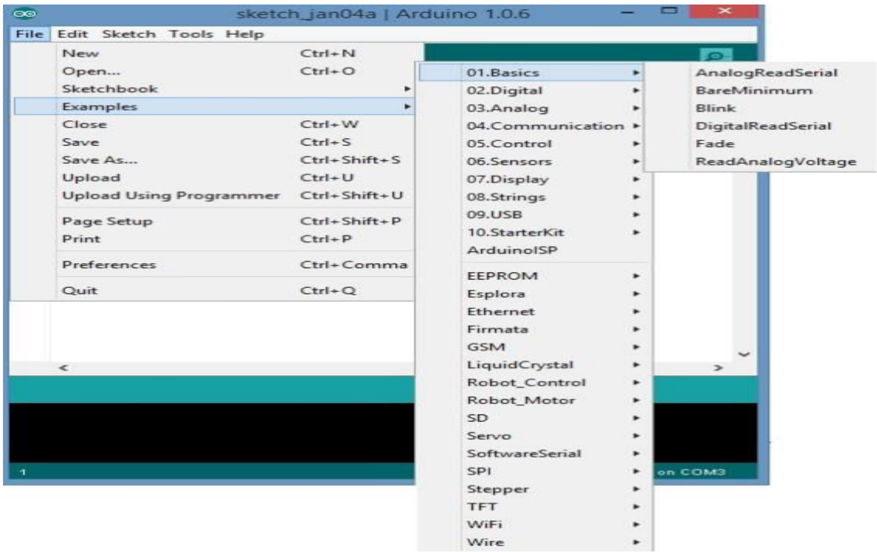
CONFIGURING THE ARDUINO IDE

The next thing to do is to make sure the software is set up for your particular Arduino board. Go to the "Tools" drop-down menu, and find "Board". Another menu will appear, where you can select from a list of Arduino models. I have the Arduino Uno R3, so I chose "Arduino Uno".


sketch_jan04a | Arduino 1.0.6
File Edit Sketch Tools Help
sketch_jan04a
Auto Format Ctrl+T
Archive Sketch
Fix Encoding & Reload
Serial Monitor Ctrl+Shift+M
Board
Serial Port
Programmer
Burn Bootloader
• Arduino Uno
Arduino Duemilanove w/ ATmega328
Arduino Diecimila or Duemilanove w/ ATmega168
Arduino Nano w/ ATmega328
Arduino Nano w/ ATmega168
Arduino Mega 2560 or Mega ADK
Arduino Mega (ATmega1280)
Arduino Leonardo
Arduino Esplora
Arduino Micro
Arduino Mini w/ ATmega328
Arduino Mini w/ ATmega168
Arduino Ethernet
Arduino Fio
Arduino BT w/ ATmega328
Arduino BT w/ ATmega168
LilyPad Arduino USB
LilyPad Arduino w/ ATmega328
LilyPad Arduino w/ ATmega168
Arduino Pro or Pro Mini (5V, 16 MHz) w/ ATmega328
Arduino Pro or Pro Mini (5V, 16 MHz) w/ ATmega168
Arduino Pro or Pro Mini (3.3V, 8 MHz) w/ ATmega328
Arduino Pro or Pro Mini (3.3V, 8 MHz) w/ ATmega168
Arduino NG or older w/ ATmega168
Arduino NG or older w/ ATmega8

EXPLORING THE ARDUINO IDE

If you want, take a minute to browse through the different menus in the IDE. There is a good variety of example programs that come with the IDE in the “Examples” menu. These will help you get started with your Arduino right away without having to do lots of research:



Learning outcomes (What I have learnt):

- Understand the function of Node MCU, Arduino Uno and Raspberry Pi.
- Learnt about Arduino IDE.

Evaluation Grid (To be created as per the SOP and Assessment guidelines by the faculty):

Sr. No.	Parameters	Marks Obtained	Maximum Marks
1.			
2.			
3.			