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

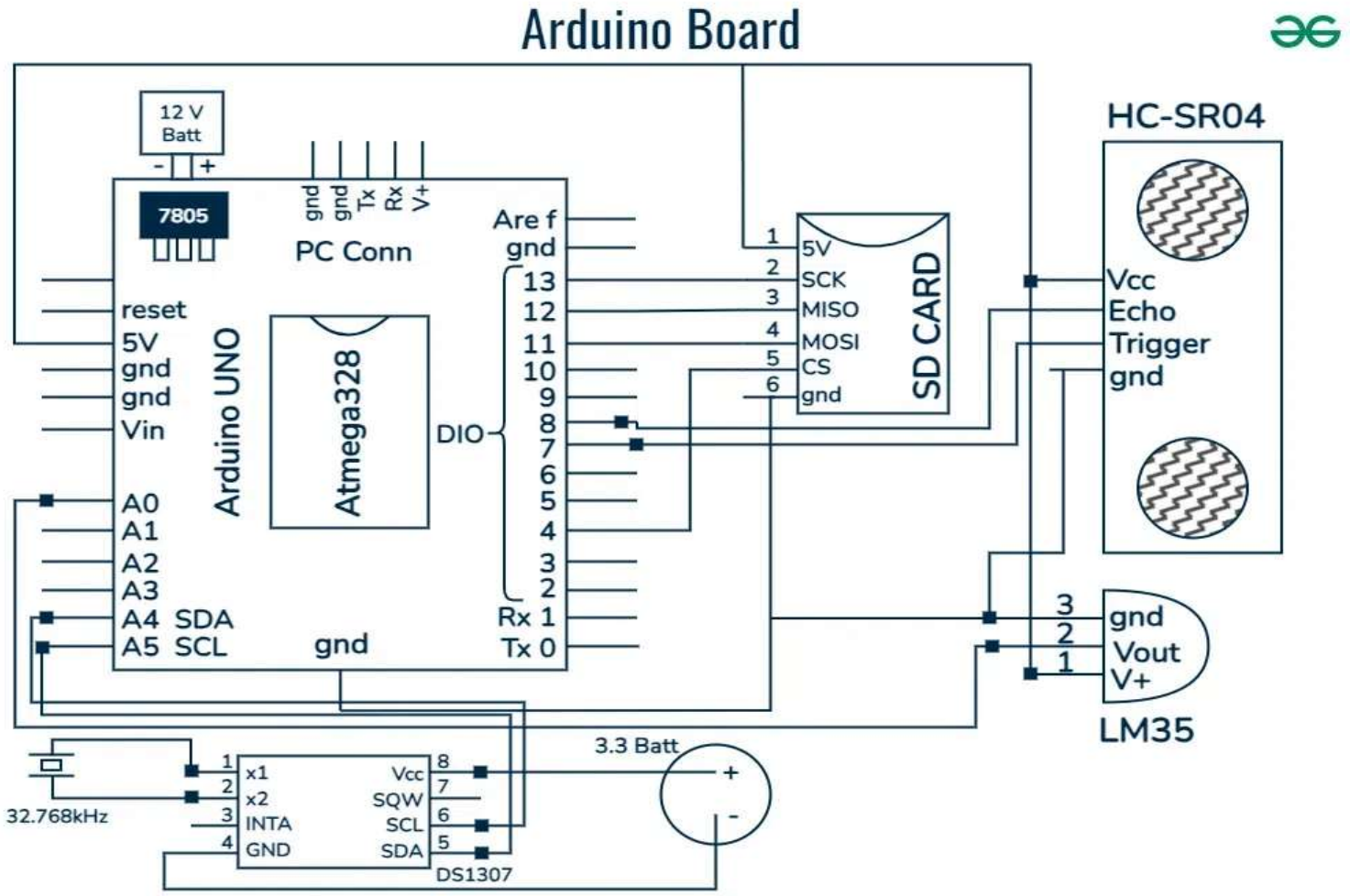
Unit V – **ARDUINO**

Topic: Overview of hardware

Overview of hardware

overview of the whole Arduino board. Anybody who has worked on Arduino will know that is a small board consisting of multiple components like ICs, and USB which are interconnected to form a whole connection. Here is a list of all the components

Overview of hardware



Overview of hardware

Analog Reference pin
Digital Ground
Digital Pins 2-13
Digital Pins 0-1/Serial In/Out - TX/RX
Reset Button - S1
In-circuit Serial Programmer
ICSP pin
Analog In Pins 0-5
Power and Ground Pins
External Power Supply In (9-12VDC) - X1
Toggles External Power and USB Power- SV1
USB (universal serial bus)
Crystal Oscillator

Overview of hardware

The Arduino needs to communicate with external devices like computers, sensors, and LEDs.

•**Serial Communication (UART):** The UART is a protocol used by Arduino for serial communication with other devices. UART stands for Universal Asynchronous Receiver/Transmitter and is used for bit data transfer. The built-in hardware in Arduino aids it in this communication with other sensors, actuators, Rasperry pies, and other boards.

•**Inter-Integrated Circuit (I2C):** This is another communication protocol that comes into the picture when we want multiple connections but with minimal wiring. The way it allows communication between multiple channels is using two wires known as the SDA - Serial Data Line and SCL - Serial Clock Line. Arduino is designed with pins that help the Arduino to connect with sensors and displays without any inconvenience.

Overview of hardware

•**Serial Peripheral Interface (SPI):** The last serial communication protocol that is used when we need high speed for data transfer. The multiple lines used in this protocol help to connect the microcontroller to other devices. Unlike I2C, it uses different wires to coordinate different tasks like communication, clock controls, etc. This protocol is suitable for connecting Arduino with SD cards, display modules, and digital-to-analog converters (DACs).

Overview of hardware

Digital Pins

In general, digital pins are used for general purposes like taking input or generating output. The commands that are used for setting the modes of the pins are **pinMode(), digitalRead(), and digitalWrite() commands**.

digitalWrite() is used to turn the resistors in each pin ON or OFF which will assign a HIGH or LOW value to the pins. The maximum current that can flow in each pin is 40 mA. Here are some digital pins.

•**Serial:** These pins are categorized into two types namely **receive (RX) and transmit (TX)** serial data. On the Arduino Diecimila, the two pins are usually numbered '0' and '1' when they perform the task of communication. They are also present at pin 12 where TX flashes the LED while data is sent and RX flashes when data is being received. Sometimes, they are used with an external TTL serial module (e.g. the Mini-USB Adapter).

Overview of hardware

•**External Interrupts:** As the name suggests, external interrupts are used to trigger an interrupt when required. This interruption can be due to a rising or falling edge, or a change in value. Once an interrupt is called, the Arduino will come to a halt and begin working only when told. These pins are PIN '2' and '3' which are controlled using the **attachInterrupt()** function.

•**PWM**: PWM stands for pulse width modulation. The pin numbers 3, 5, 6, 9, 10, and 11 are PWM pins. The **analogWrite()** function is used for generating an 8-bit output. So when a large output is to be received or transmitted, the 8-bit output is generated. On certain boards like ATmega8, these pins are limited and present at 9, 10, and 11.

Overview of hardware

•**SPI(serial peripheral interface):** This is a synchronous serial data protocol generally used by microcontrollers. This is present at pin number 10 (SS), 11 (MOSI), 12 (MISO), and 13 (SCK) which are used by microcontrollers for communicating with different devices. The relationship can be understood as the output device acting as a slave to the master of the SPI bus.

•**LED:** Present at pin number 13 in some Arduino, LED is often used for testing purposes. The LED glows when the pin is HIGH, and turns off when the pin is LOW. Sometimes it is also possible to connect some external LEDS by using breadboard and jumper wires.

Overview of hardware

Analog Pins

In general, the analog pins are used for general purposes like supporting 10-bit analog-to-digital conversion (ADC) which is performed using **analog the Read()** function. These analog inputs can also be used as digital pins: analog input 0 as digital pin 14 through analog input 5 as digital pin 19. Analog pins are particularly helpful since they can store 0-255 bits which is not possible using digital pins. This feature is not available on every Arduino board.

•**I2C(Inter-Integrated Circuit):** These pins are present at numbers 4 (SDA) and 5 (SCL) and are used to perform I2C (TWI) communication. Note that we need to import the Wire library to use this protocol

Overview of hardware

Crystal Oscillator

The crystal oscillator is a device on Arduino that deals with issues involving time. The Arduino calculates time using this oscillator only. If you observe, you will see the number '16.000H9H' printed on top of the Arduino crystal. This indicates that the Arduino operates at a frequency of 16,000,000 Hertz pf. Crystal oscillators are very precise and accurate devices. For example, a crystal oscillator is also present on he Arduino to provide clock pulses to the microcontroller Atmega 328 and help it control all commands and order of execution.