Department: CSE

Program: B.Sc. in CSE

Course No: CSE3215

Course Title: Microcontroller Based System Design

Examination: Semester Final

Semester (Session): Fall2020

Student No: 18.01.04.072

Signature and Date: Rakib, 27 october, 2021

Student ID: 18.01.04.072

Course No: CSE 3215 Rakib

Ans. to the que. No: 2

(a)

Inter Integrated circuit Protocol (I_2C) is a serial communication protocol allow to connect multiple slaves to a single master and multiple masters controlling single or multiple slaves. Only two wires are required for communication between up to almost 128 devices when using 7 bits addressing and upto almost 1024 devices when using 10 bits addressing. with I_2C, data is transferred in messages. Messages are broken up into frames of data. Each message has an address frame that contains the binary address of the slave, and one or more data frames that contain the data being transmitted. The message also includes start and stop conditions, read/write bits and ACK/NACK bits between each data frame.

Start	7 or 10 Bits	Read/write bit	ACK/NACK bit	8 bits	ACK/NACK bit	8 bits	ACK/NACK bit	Stop
	Address Frame			Data Frame1		Data Frame2		

Fig: Message sent by I_2C

Student ID: 18.01.04.072

Course No: CSE3215 Rakib

Start condition: To initiate the address frame, the master device leaves SCL high and pulls SDA Low. This puts all slave devices on notice that a transmission is about to start.

Stop condition: Once all the data frames have been sent, the master will generate a stop condition.

Address Frame: A 7 or 10 bit sequence unique to each slave that identifies the slave when the master wants to talk to it.

Data Frame: After the master detects the ACK bit from the slave, the first data frame is ready to be sent.

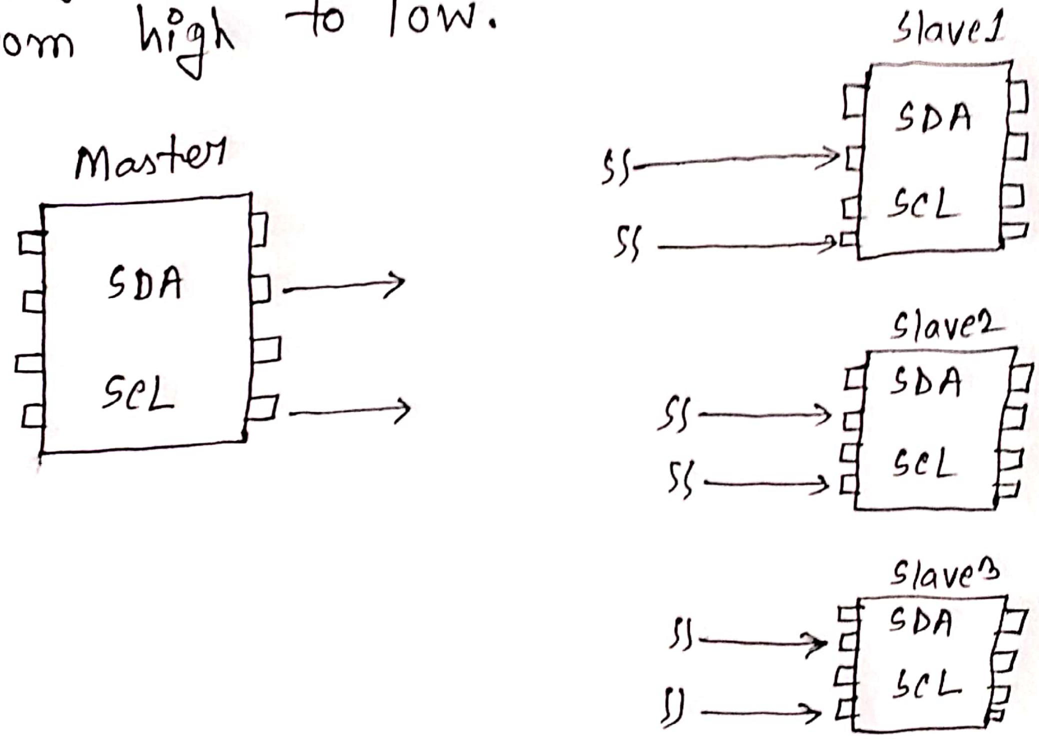
Read/write Bit: The address frame includes a single bit at the end that informs the slave whether the master wants to write data to it or receive data from it.

Ack/NACK bit: Each frame in a message is followed by an acknowledge/no-acknowledge bit.

Now we will discuss the steps of I_2C Data Transmission:

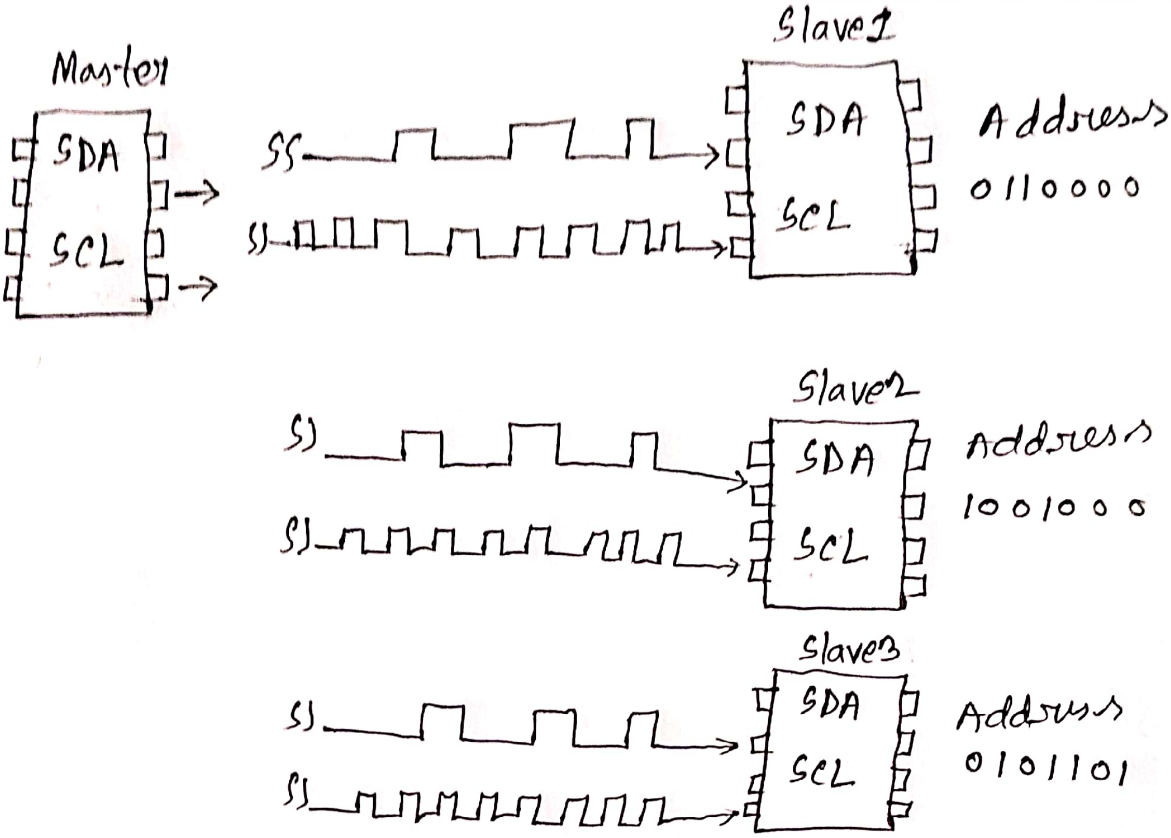
Step-1:

The master sends the start condition to every connected slave by switching the SDA line from a high voltage level to a low voltage level before switching the SCL line from high to low.



Step-2:

The master sends each slave the 7 or 10 bit address of the slave it wants to communicate with, along with the read/write bit.

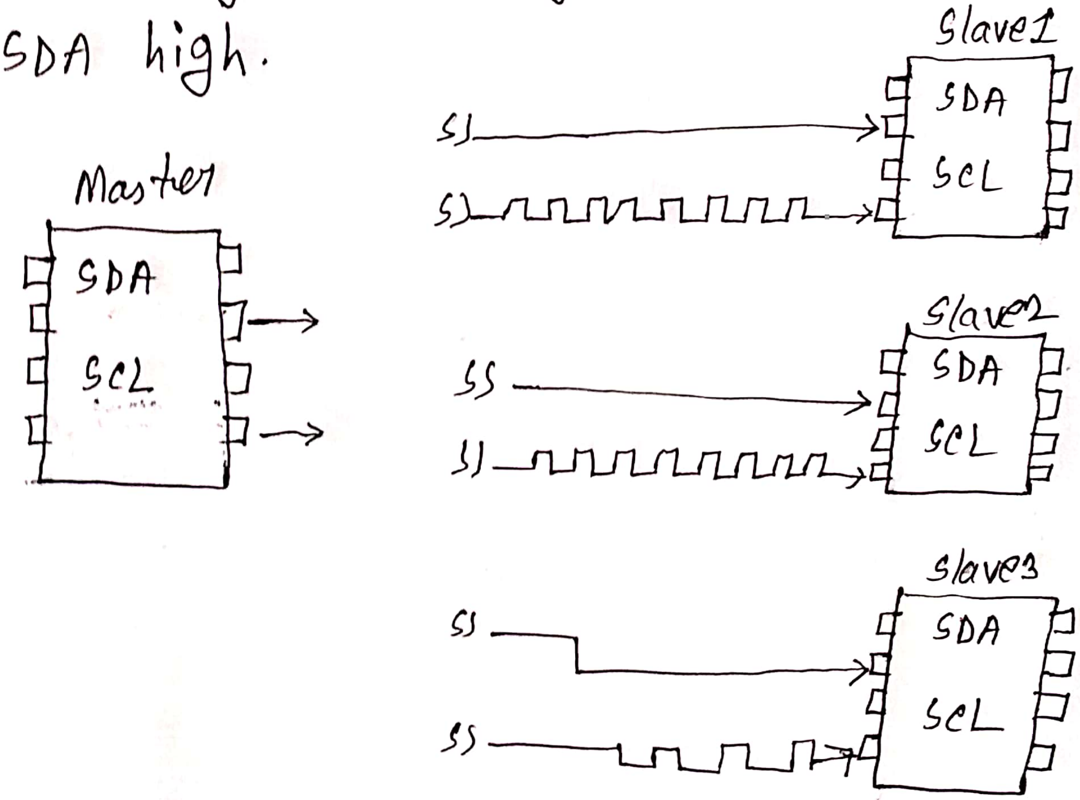


<u>Step-3:</u> Each slave compares the address sent from the master to its own address. If the address matches, the slave returns an ACK bit by pulling the SDA line low for one bit. If the address from the master does not match the slaves own address, the slave leaves the SDA line high.

<u>Step-4:</u> The master sends or receives the data frame.

Step-5: After each data frame has been transferred, the receiving device returns another ACK bit to the sender to acknowledge successful receipt of the frame.

Step-6: To stop the data transmission, the master sends a stop condition to the slave by switching SCL high before switching SDA high.



(b)

A watchdog timer is a hardware timer that automatically generates a system reset it the main program neglects to periodically service it. It is often used to automatically reset an embedded device that hangs because of a software or hardware fault.

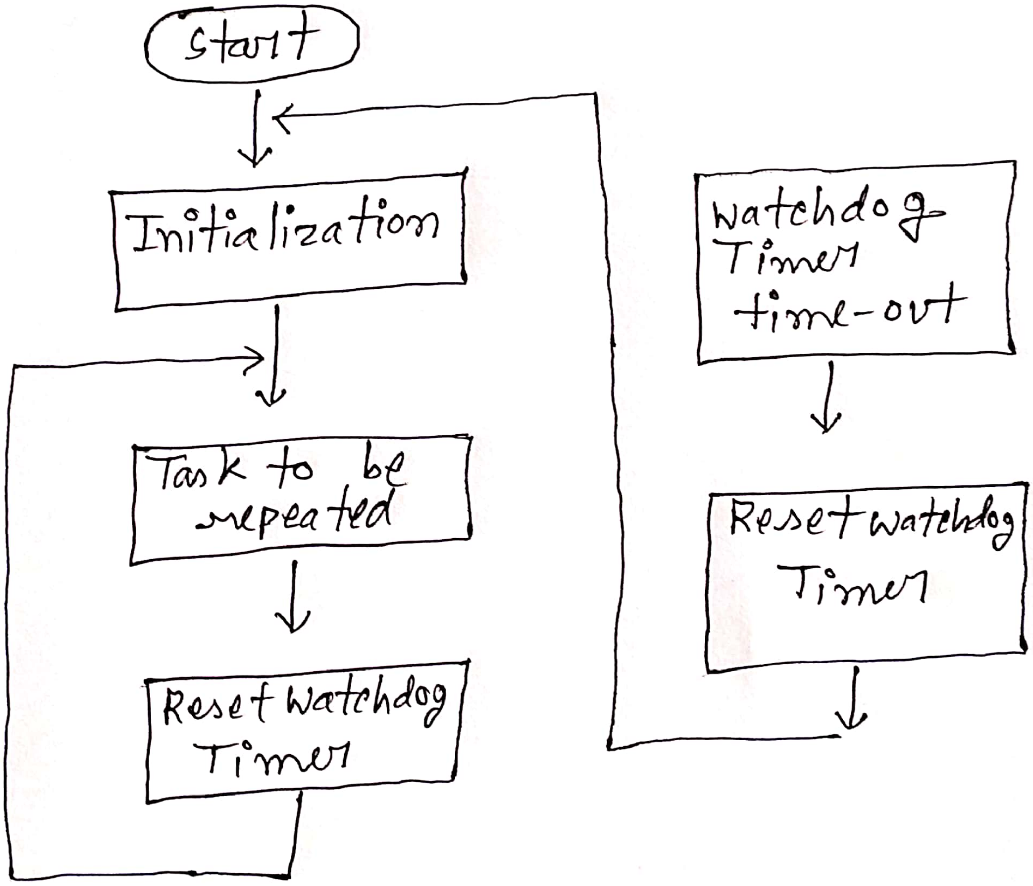


Fig: Flowchart of Watchdog Timer

After the initialization, the microcontroller remains in an infinite loop. within the loop, a few instructions are embedded as part of the main program, which resets a timer, known as watchdog timer, which is initialized with a time-out parameter longer than the worst-case time to go around the loop. The timer acts as a resetable timer - each time the program goes around the loop, the timer is reset before it reaches time-out. when the system goes astray, that is the micro-controller is no longer executing the loop, the watchdog timer is not reseted and it proceeds towards time-out. when the timer reaches the end of the time-out interval, the timer output resets the micro-controller bringing it back to the predefined initial state. So, a watchdog timer plays a very important role in the case of embedded systems where there is no human access.

(c)

we know, $V_{out} = V_{ref} * \sum_{i=1}^{n} \frac{b_{n-i}}{2^i}$

$$\Rightarrow 4.5V = V_{ref} * \left(\frac{1}{1} + \frac{1}{4} + \frac{1}{16} + \frac{1}{64}\right)$$

$$\Rightarrow 4.5V = V_{ref} * 1.33$$

$$\therefore V_{ref} = \frac{4.5}{1.33} = 3.383V.$$

Student ID: 18.01.04.072

Course No: CSE 3215 Rakib

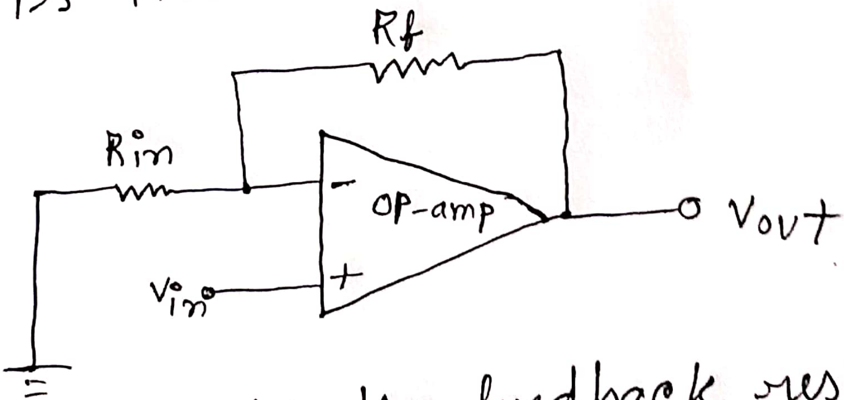
Total bits = 7 [As 1010101]

$\therefore LSB = \frac{12\,V}{2^7} = 0.094\,V$

Ans. to the que. No: 4

(b)

A voltage follower is an op-amp circuit which has a voltage gain of 1. In a voltage follower, the op-amp does not provide any amplification to the signal. In other words, it does not amplify the input signal. A voltage follower is called voltage follower because the output directly follows the input voltage, which means the output voltage is the same as the input voltage.



Here, if we make the feedback resistance equal to zero ($R_f = 0$) and input resistance equal to infinity ($R_{in} = \infty$), then the circuit will have a fixed gain of "1" as all the output voltage must be present on the

Student ID: 18.01.04.072

Course No: CSE3215 Rakib

inverting input terminal. This will produce a voltage follower circuit or also called a unity gain buffer. As an example, if we give 10 v input, we will get exactly 10 v as output.

The purposes of using voltage follower are:

i) A voltage follower can be used as a buffer because it draws very little current due to the high input impedance of amplifier thus eliminating loading effect while still maintaining the same voltage at the output.

ii) voltage followers are also important in voltage divider circuits. A large voltage drop across the op-amp due to its high impedance. So, if we use voltage follower in voltage divider circuits, it will let adequate voltage to be supplied across the load.

(c)

An ultrasonic sensor is a device that can measure the distance to an object by using sound waves. It measures distance by sending out a sound wave at a specific frequency and listening for that sound wave to come back. By recording the elapsed time between the sound have being generated and the sound wave bouncing back, it is possible to calculate the distance of the object by following formula,

$$distance = \frac{speed\ of\ sound \times time\ taken}{2}$$

An Infrared Sensor or IR sensor detects obstacles in front and can also differentiate between colours depending on the configuration of the sensor. It emits IR light and gives a signal when it detects the reflected light. It can differentiate between dark and bright surface because the IR light will only reflect on the dark surface.

i) water level measurement in tank: we can use the ultrasonic sensor to detect water level measurement. It will send sound into the water take and get back the reflection of sound and calculate the distance. By sensing distance it will let us know the water level inside the tank.

Student ID: 18.01.04.072

Course No: CSE 3215 Rakib

ii) <u>Recognizing obstacles in distance:</u> we can recognize obstacles in distance by using IR sensor. IR sensor can differentiate between objects. So it will be useful to recognize obstacles in distance.

(a)

Given,

$R_f = 250 k\Omega$

$V_{in} = 1V$

$R_{in} = 10 k\Omega$

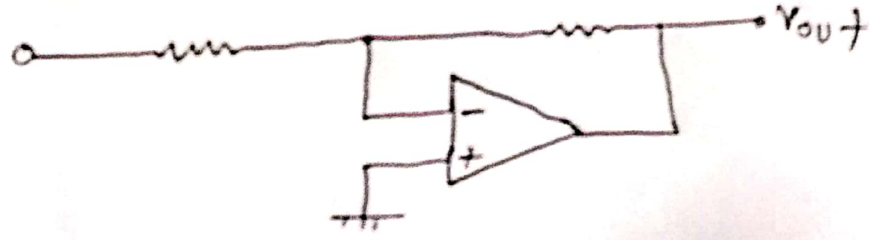
$\therefore I = \frac{V_{in}}{R_{in}} = \frac{1V}{10kHz} = 0.1 mA$

Now, $V_o = -\frac{R_f}{R_{in}} \times V_{in} = -\frac{2\cdot5k\Omega}{10k\Omega} \times 1V$

$= -25V$

Gain, $A = -\frac{R_f}{R_{in}} = -\frac{250k\Omega}{10k\Omega} = -25$

Now, we will redraw the circuit below.



Now,

$$I = \frac{V_{in} - V_{out}}{R_{in} + R_f} \quad \ldots (i)$$

If we take the point V_{in} and V_2, we can also write I as,

$$I = \frac{V_{in} - V_2}{R_{in}} \quad \ldots (ii)$$

Again, lets take the point V_2 and V_o, we can write I as,

$$I = \frac{V_2 - V_{out}}{R_f} \quad \ldots (iii)$$

From equation (i), (ii), (iii) we can write,

$$I = \frac{V_{in} - V_2}{R_{in}} = \frac{V_2 - V_{out}}{R_f} .$$ [Proved]

Ans. to the que. No: 1

(a)

Major difference between AVR and 8051 microcontrollers are following:

AVR microcontroller: The AVR stand for Advanced virtual RISC. It is one of the first micro-controller to use on chip flash memory for program storage.

Student ID: 18.01.04.072

Course No: CSE3215 Rakib

8051 Microcontroller: 8051 microcontroller is an 8 bit Microcontroller which is basically CISC based.

AVR microcontroller	8051 microcontroller
AVRs are all 8 bit microcontroller. AVR32 which is exception. Because it is 32 bit microcontroller.	8051 microcontrollers are all 8 bits.
It use RISC instruction set.	It use CISC instruction set.
The AVR has 64k bytes Data RAM memory.	8051 has 128 byte on chip data memory.
AVR has 256k on chip RAM.	8051 has 4k on chip Ram
stack pointer is 16 bit wide.	stack pointer is 8 bit wide.
SP starts from the last address of memory.	SP point to 07h initially.
SP decrements when push.	SP increments when push.
SP increments when pop.	SP decrements when pop.

Student ID: 18.01.04.072

Course No: CSE 3215

Rakib

Uses of LDI and LDS instruction:

AVR has 32 general purpose register (R_0-R_{31})

LDI can use only (R_{16} - R_{31}).

LDI R_{20}, 0x25

so, R_{20} = 0x25

LDI uses for loading Direct value in Register.

LDS R_{20}, 0xFE

0xFE is a ddress.

R_{20} = value 0xFE $\boxed{\text{value}}$

LDS can use R_0 to R_{31} Register.

(b)

$(7)_d = (00000111)_b$

1's complement = 11111000

2's " = +1

(-7) = 11111001

$(4)_d = (00000100)_b$

1's complement = 11111011

2's " = +1

(4) = 11111100

-7 = 11111001

(+) -4 = 11111100

11110101

V=0 [As D_6 transfer a carry, also D_7 transfer a carry]

N=1 [As D_7 is 1]

$S = V \oplus N = 0 \oplus 1 = 1$

P=0 [Even no. of one bits in 8 bit result]

```
LDI  R16, -7
LDI  R17, -4
ADD  R16, R17
```

(c)

(i) ATtiny25 with 2kB

$2^1 = 2$

on chip ROM = 2 kB

IN AVR microcontrollers each location is 2 byte wide.

ROM organization = 1k x 2 bytes

memory address Range = 00000H – 003FFH

(ii) ATmega64 with 64 kB

$2^6 = 64$

on chip ROM = 64 kB

In AVR microcontrollers each location is 2 bytes wide.

ROM organization = 32k x 2 bytes

memory address range = (00000H – 07FFFH)

Student ID: 18.01.04.072

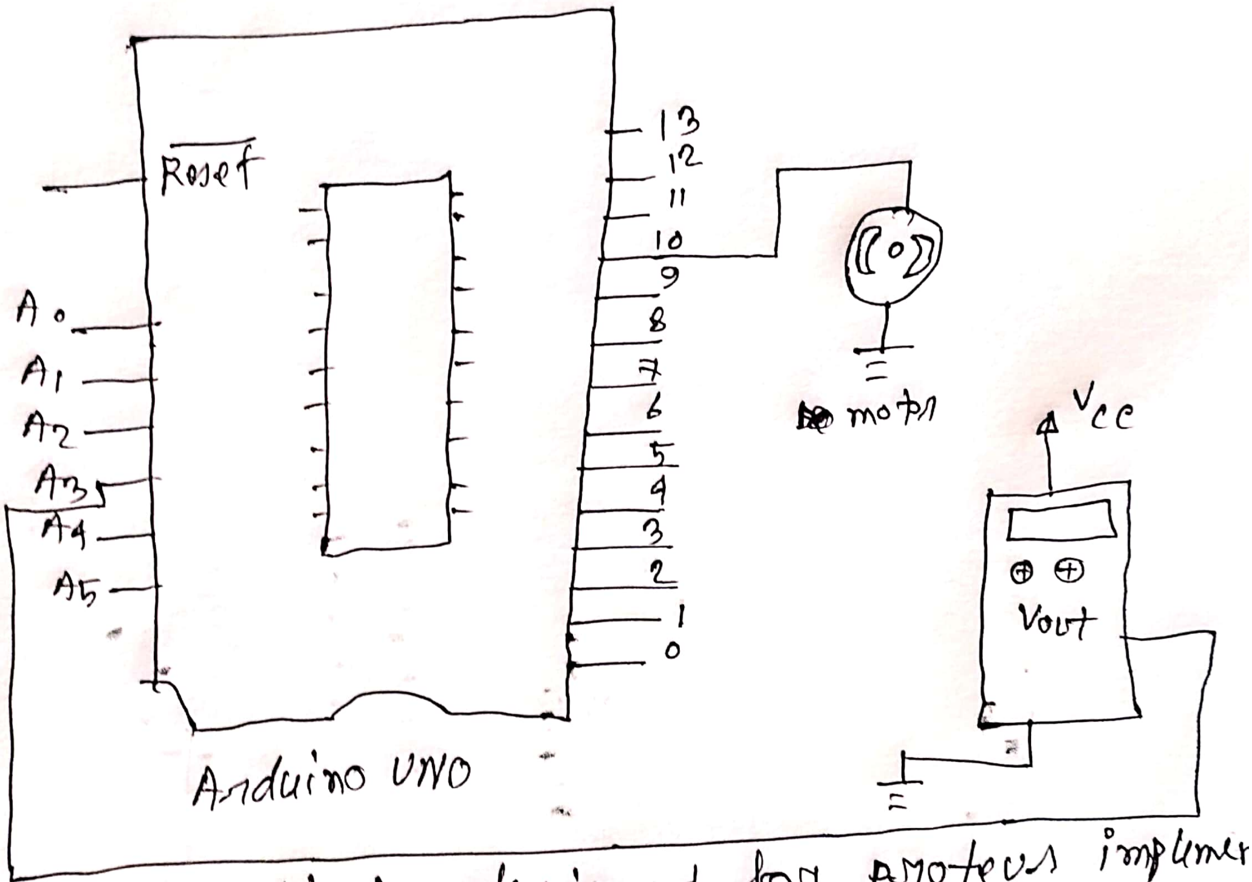
Course No: CSE 3215 Rakib

Ans. to the que. No: 5

(c)

A circuit is designed below with the temperature sensor and Motor for an Automatic Garden System.

Circuit:



This circuit is designed for proteus implementation.

Following is the Arduino code for this system:

```
int value = 0;
int lm35 = A3;
int motor = 10;
```

Student ID: 18.01.04.072

Course No: CSE3215 Rakib

```
void setup()
{
  Serial.begin(9600);
  pinMode (motor, OUTPUT);
}
void loop()
{
   value = analogRead (Im35);
   float milivolts = (value/1024.0)*5*1000;
   float cel = milivots/10;
   if (cel > 35)
   {
      digitalwrite(motor, HIGH);
   }
   else
   {  digitalwrite(motor, LOW);
   }
}
```

Student ID: 18.01.04.072

Course No: CSE 3215 Rakib

(a)

```
SETB  PSW.3
MOV   R1, #50H
MOV   R2, #60H
MOV   R3, #70H

SETB  PSW.4
MOV   R0, #2H
MOV   R2, #4H
```

In RAM location 09, 0A, 0B, there exists #50H, #60H, #70H and in location 18, 1A, there exists #2H and #4H

RAM location	content
⋮	
1A	#4H
18	#2H
⋮	
0B	#70H
0A	#60H
09	#50H

Student ID: 18.01.04.072

Course No: CSE3215 Rakib

MOV SP, #4FH

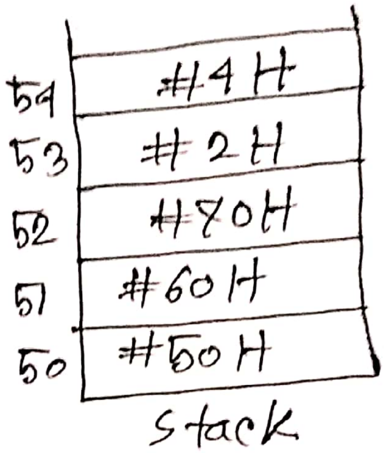
PUSH 9

PUSH A

PUSH B

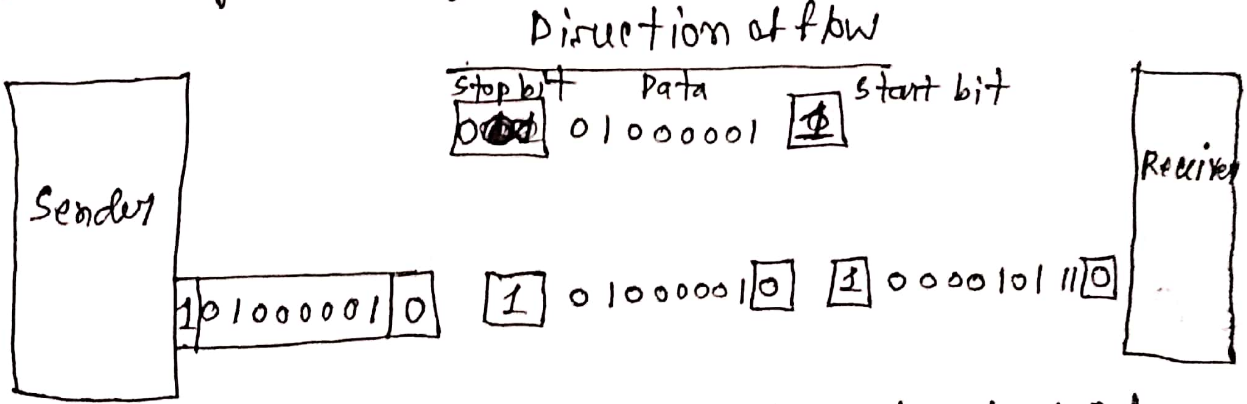
PUSH 18

PUSH 1A



(b)

Following is the frame format for the transmission of the ASCII character 'A' using the asynchronous serial mode:



The addition of stop and start bits and the insertion of gaps into the bit stream make asynchronous transmission slower than forms of transmission that can operate without the addition of control information. But it is cheap and effective.