ARDUINO

Task program

PROGRAM:-

LED BLINK:-

```
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(5, OUTPUT); //set the pin 5 as output
  pinMode(6, OUTPUT); //set the pin 5 as output
  pinMode(7, OUTPUT); //set the pin 5 as output
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(5, HIGH); //set the 5 pin as 1
  delay(1000);
  digitalWrite(5, LOW); //set the 5 pin as 0
  delay(1000);
  digitalWrite(6, HIGH); //set the 6 pin as 1
  delay(1000);
  digitalWrite(6, LOW); //set the 5 pin as 0
  delay(1000);
  digitalWrite(7, HIGH); //set the 7 pin as 1
  delay(1000);
  digitalWrite(7, LOW); //set the 5 pin as 0
  delay(1000);
}
```

LED PATTERN USING SWITCH:-

```
const int led1Pin = 3; // Pin for the first LED
const int led2Pin = 4; // Pin for the second LED
const int led3Pin = 5; // Pin for the third LED
const int led4Pin = 6; // Pin for the fourth LED

void setup() {
  pinMode(1, INPUT_PULLUP); // Set pin 1 as input with internal pull-up resistor
  pinMode(2, INPUT_PULLUP); // Set pin 2 as input with internal pull-up resistor
  pinMode(led1Pin, OUTPUT); // Set led1Pin as output
  pinMode(led2Pin, OUTPUT); // Set led2Pin as output
  pinMode(led3Pin, OUTPUT); // Set led3Pin as output
  pinMode(led4Pin, OUTPUT); // Set led4Pin as output
}

void loop() {
  int inp1 = digitalRead(1); // Read state of pin 1
  int inp2 = digitalRead(2); // Read state of pin 2

  if (inp1 == HIGH & inp2==HIGH) // Check if both pins are HIGH
  {
    digitalWrite(led1Pin, HIGH); // Turn on all LEDs
    digitalWrite(led2Pin, HIGH);
    digitalWrite(led3Pin, HIGH);
    digitalWrite(led4Pin, HIGH);
    delay(100); // Pause
    digitalWrite(led1Pin, LOW); // Turn off all LEDs
    digitalWrite(led2Pin, LOW);
    digitalWrite(led3Pin, LOW);
    digitalWrite(led4Pin, LOW);
    delay(100); // Pause
  }
  else if(inp1 == LOW & inp2==HIGH) // Check if pin 1 is LOW and pin 2 is HIGH
  {
```

ARDUINO

Task program

```
 digitalWrite(led1Pin, HIGH); // Turn on first LED
 delay(100); // Pause
 digitalWrite(led1Pin, LOW); // Turn off first LED
 digitalWrite(led2Pin, HIGH); // Turn on second LED
 delay(100); // Pause
 digitalWrite(led2Pin, LOW); // Turn off second LED
 digitalWrite(led3Pin, HIGH); // Turn on third LED
 delay(100); // Pause
 digitalWrite(led3Pin, LOW); // Turn off third LED
 digitalWrite(led4Pin, HIGH); // Turn on fourth LED
 delay(100); // Pause
 digitalWrite(led4Pin, LOW); // Turn off fourth LED

lse if(inp1 == HIGH & inp2==LOW) // Check if pin 1 is HIGH and pin 2 is LOW

 digitalWrite(led4Pin, HIGH); // Turn on fourth LED
 delay(100); // Pause
 digitalWrite(led4Pin, LOW); // Turn off fourth LED
 digitalWrite(led3Pin, HIGH); // Turn on third LED
 delay(100); // Pause
 digitalWrite(led3Pin, LOW); // Turn off third LED
 digitalWrite(led2Pin, HIGH); // Turn on second LED
 delay(100); // Pause
 digitalWrite(led2Pin, LOW); // Turn off second LED
 digitalWrite(led1Pin, HIGH); // Turn on first LED
 delay(100); // Pause
 digitalWrite(led1Pin, LOW); // Turn off first LED

lse if(inp1 == LOW & inp2==LOW) // Check if both pins are LOW

 digitalWrite(led1Pin, HIGH); // Turn on first LED
 digitalWrite(led2Pin, HIGH); // Turn on second LED
 digitalWrite(led3Pin, HIGH); // Turn on third LED
```

```
 digitalWrite(led4Pin, HIGH); // Turn on fourth LED
 digitalWrite(led1Pin, LOW); // Turn off first LED
 delay(100); // Pause
 digitalWrite(led1Pin, HIGH); // Turn on first LED
 digitalWrite(led2Pin, LOW); // Turn off second LED
 delay(100); // Pause
 digitalWrite(led2Pin, HIGH); // Turn on second LED
 digitalWrite(led3Pin, LOW); // Turn off third LED
 delay(100); // Pause
 digitalWrite(led3Pin, HIGH); // Turn on third LED
 digitalWrite(led4Pin, LOW); // Turn off fourth LED
 delay(100); // Pause
 digitalWrite(led4Pin, HIGH); // Turn on fourth LED
```

ARDUINO

Task program

LCD NAME PRINT:-

```
#include <LiquidCrystal.h> // Include the LiquidCrystal library

LiquidCrystal lcd(13,12,7,6,5,4,3,2,1,0); // Create an instance of the LiquidCrystal class

void setup() {
  lcd.begin(16, 2); // Initialize the LCD with 16 columns and 2 rows
}

void loop() {
  char array = "MAGESH"; // Define a character array with the name "array"
  unsigned char cmd1[6] = {0, 1, 2, 3, 4, 5}; // Define an array of unsigned characters for the first row commands
  unsigned char cmd2[6] = {15, 14, 13, 12, 11, 10}; // Define an array of unsigned characters for the second row commands

  for (int x = 0; x < 6; x++) {
    lcd.setCursor(cmd1[x], 0); // Set the cursor position for the first row
    lcd.write(array[x]); // Write the character from the array to the LCD
    delay(100); // Pause for 100 milliseconds
  }

  for (int j = 0; j < 6; j++) {
    lcd.setCursor(cmd2[j], 1); // Set the cursor position for the second row
    lcd.write(array[j]); // Write the character from the array to the LCD
    delay(100); // Pause for 100 milliseconds
  }
}
```

PWM:-

```
void setup() {
  pinMode(3, OUTPUT);  // sets pin 3 as an output
}

void loop() {
  int i;
  for(i=0;i<255;i++)
  {
    analogWrite(3,i);  // gradually increase the PWM signal on pin 3 from 0 to 255
  }
  for(i=255;i>0;i--)
  {
    analogWrite(3,i);  // gradually decrease the PWM signal on pin 3 from 255 to 0
  }
}
```

ARDUINO

Task program

ADC:-

```
#include <LiquidCrystal.h> // Include the LiquidCrystal library

LiquidCrystal lcd(13, 12, 7, 6, 5, 4, 3, 2, 1, 0); // Create an instance of the LiquidCrystal class

void setup() {
  lcd.begin(16, 2); // Initialize the LCD with 16 columns and 2 rows
  const char* array = "ORIG VAL:"; // Define a character array for the label "ORIG VAL:"
  const char* array1 = "CALIB VAL:"; // Define a character array for the label "CALIB VAL:"
  lcd.setCursor(0, 0); // Set the cursor position to the beginning of the first row
  for (int x = 0; x < 9; x++) {
    lcd.write(array[x]); // Write each character of "ORIG VAL:" to the LCD with a delay
    delay(50);
  }
  lcd.setCursor(0, 1); // Set the cursor position to the beginning of the second row
  for (int x = 0; x < 10; x++) {
    lcd.write(array1[x]); // Write each character of "CALIB VAL:" to the LCD with a delay
    delay(100);
  }
}

void loop() {
  int adc = analogRead(A0); // Read the analog input from pin A0
  lcd.setCursor(9, 0); // Set the cursor position for displaying the ADC value
  if (adc < 1000)
    lcd.write(' '); // Add space for formatting if ADC value is less than 1000
  if (adc < 100)
    lcd.write(' '); // Add space for formatting if ADC value is less than 100
  if (adc < 10)
    lcd.write(' '); // Add space for formatting if ADC value is less than 10
  lcd.print(adc); // Display the ADC value on the LCD

  lcd.setCursor(10, 1); // Set the cursor position for displaying the voltage value
  int voltage = map(adc, 0, 1023, 0, 100); // Map the ADC value to a percentage
```

```
    if (voltage < 10)
      lcd.write(' '); // Add space for formatting if voltage is less than 10
    if (voltage < 100)
      lcd.write(' '); // Add space for formatting if voltage is less than 100
    lcd.print(voltage); // Display the voltage percentage on the LCD

    delay(500); // Delay before reading the analog input again
  }
```

ARDUINO

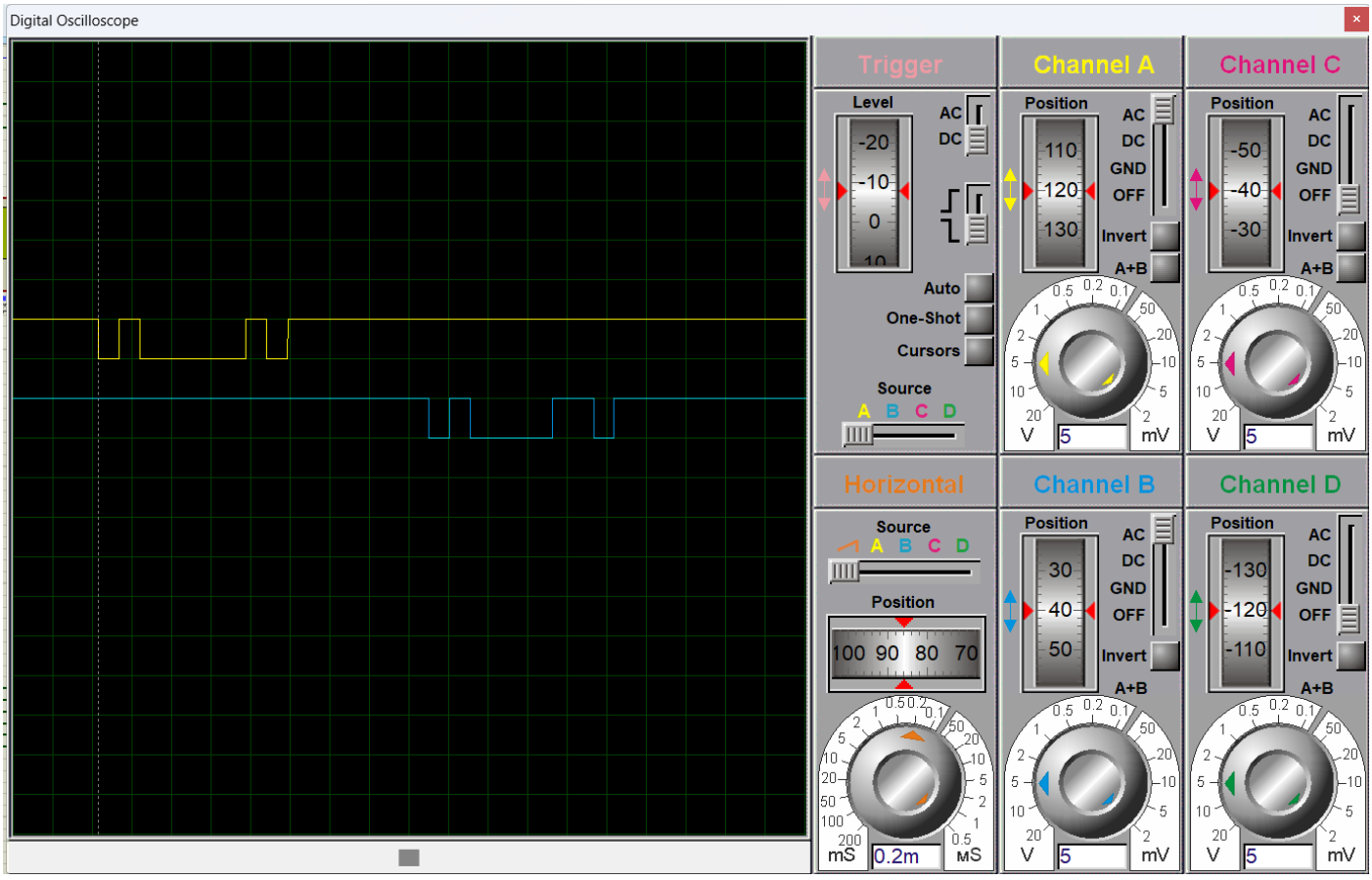
Task program

UART:-

```
#include <LiquidCrystal.h> // Include the LiquidCrystal library
// Initialize the LiquidCrystal display with the specified pin numbers
LiquidCrystal lcd(13, 12, 5, 4, 3, 2);
void setup() {
  lcd.begin(16, 2); // Initialize the LCD with 16 columns and 2 rows
  Serial.begin(9600); // Initialize serial communication at 9600 baud
}
void loop() {
  char pr[15]={"RX DATA:"}; // Define a character array for the label "RX DATA:"
  char prt[15]={"TX DATA:"}; // Define a character array for the label "TX DATA:"
  lcd.setCursor(0,0); // Set the cursor position to the beginning of the first row
  for(int i=0;i<8;i++) // Loop to write the characters of the "RX DATA:" label
    lcd.write(pr[i]); // Write each character to the LCD
  lcd.setCursor(0,1); // Set the cursor position to the beginning of the second row
  for(int i=0;i<8;i++) // Loop to write the characters of the "TX DATA:" label
    lcd.write(prt[i]); // Write each character to the LCD
  // Check if there is serial data available
  while (Serial.available() >= 0) {
    char receivedData = Serial.read(); // Read the incoming serial data
    // Check the received data and update the LCD and send back modified data via serial
    if (receivedData == 'A') {
      lcd.setCursor(8, 0); // Set the cursor position for displaying received data
      lcd.write(receivedData); // Write the received data to the LCD
      receivedData = 'a'; // Modify the received data
      Serial.write(receivedData); // Send back modified data via serial
      lcd.setCursor(8, 1); // Set the cursor position for displaying modified data
      lcd.write('a'); // Write the modified data to the LCD
    } else if (receivedData == 'B') {
      // Repeat the above steps for other received characters ('B', 'C', 'D')
    } else if (receivedData == 'C') {
      // Repeat the above steps for other received characters ('B', 'C', 'D')
    } else if (receivedData == 'D') {
      // Repeat the above steps for other received characters ('B', 'C', 'D')
```

OUTPUT WAVE FORM FOR MASTER

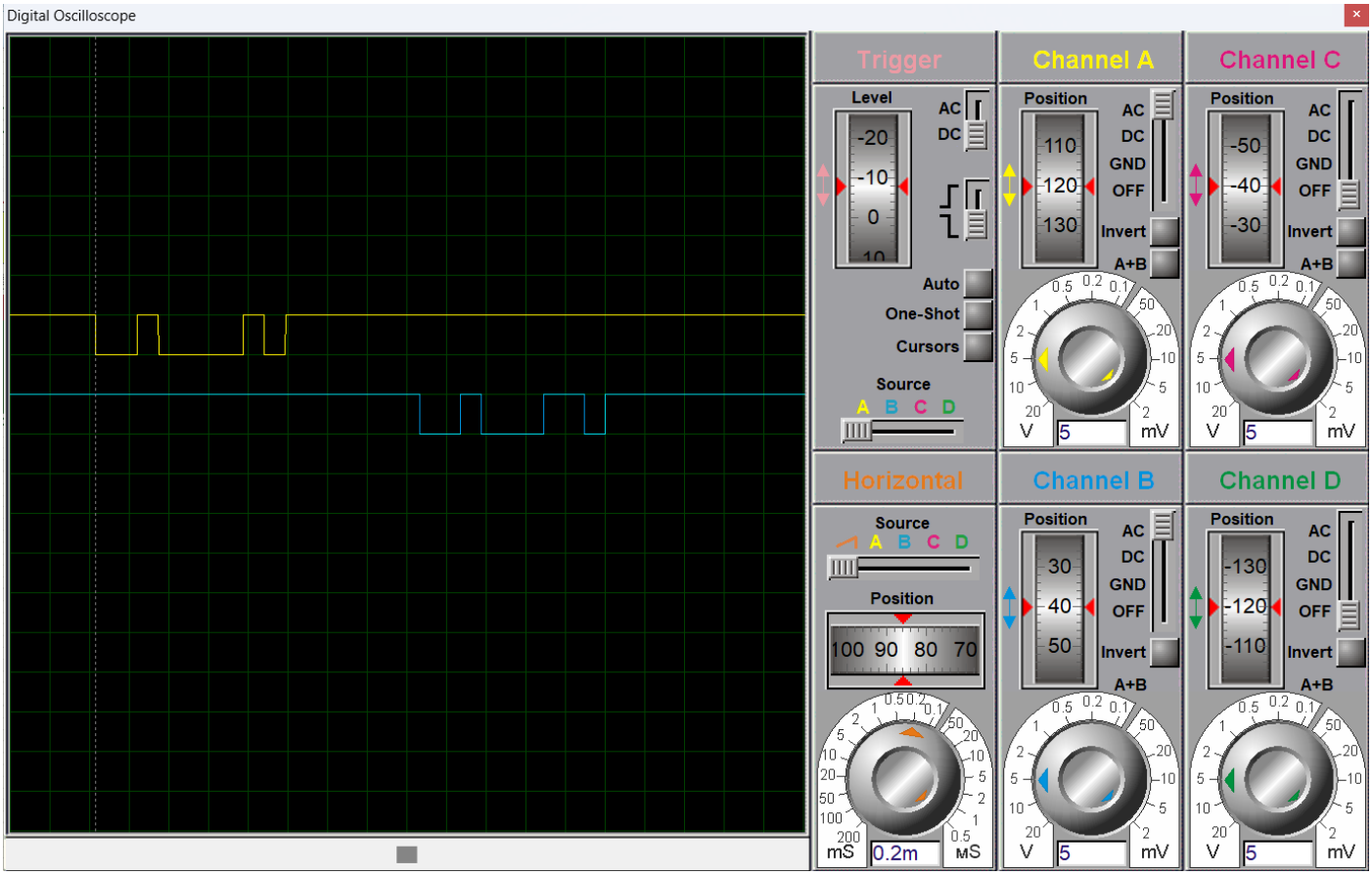
when transmitting-**A**



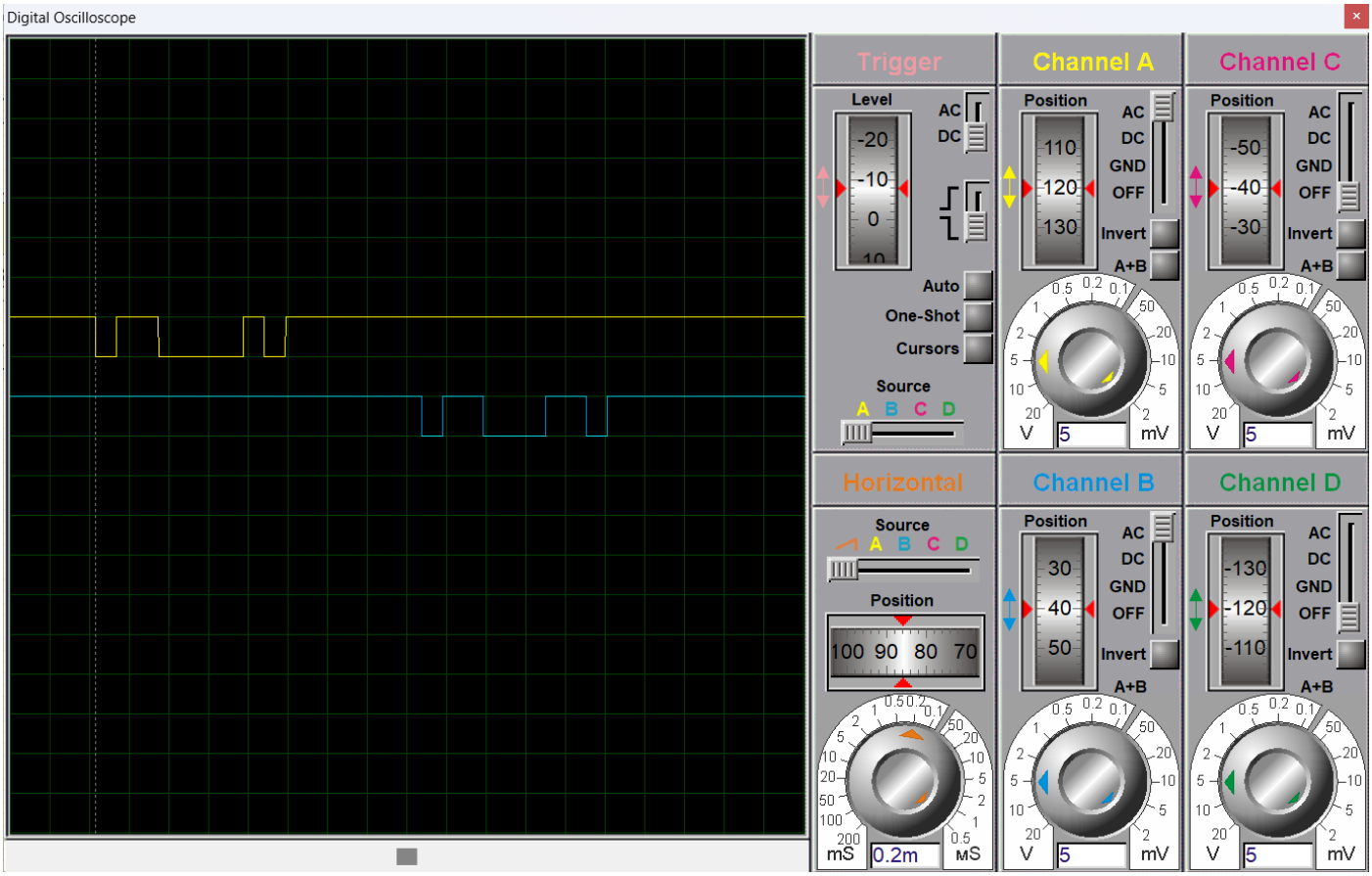
ARDUINO

Task program

when transmitting-**B**



when transmitting-**C**



ARDUINO

Task program

when transmitting-**D**