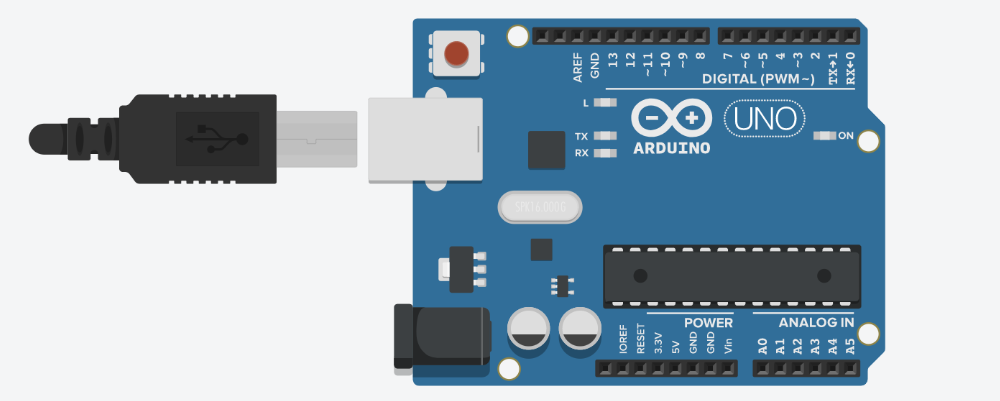
***Practical 7***

Aryaman Gautam

J001

1. EEPROM separate programs using read(), write(), get(), put(), update()



#include<EEPROM.h>

void setup()

{

pinMode(13, OUTPUT);

Serial.begin(9600);

Serial.println("Using write operation in EEPROM");

for (int i=0; i<EEPROM.length(); i++){

EEPROM.write(i, 100);

}

Serial.println("Using read operation in EEPROM");

for (int i=0; i<EEPROM.length(); i++){

Serial.print(i);

Serial.print("\t");

Serial.print(EEPROM.read(i));

Serial.println();

}

digitalWrite(13, HIGH);

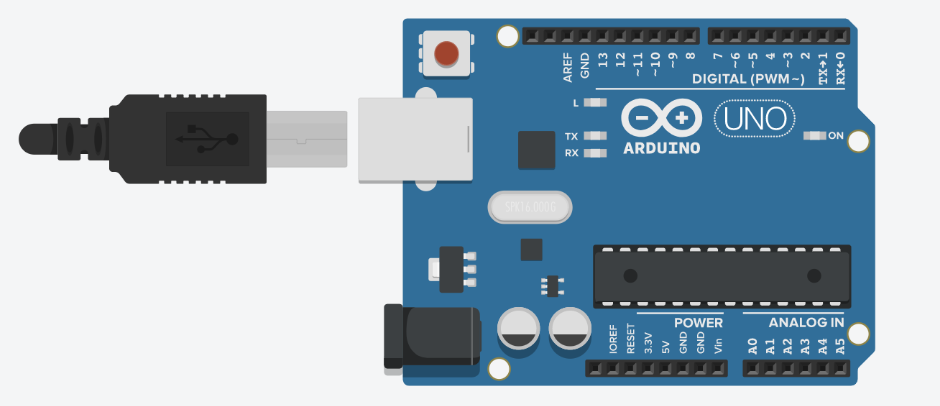
Serial.println("the Read operation is successful");

}

void loop()

{

}



#include<EEPROM.h>

void setup()

{

pinMode(13, OUTPUT);

Serial.begin(9600);

Serial.println("Using write operation in EEPROM");

for (int i=0; i<EEPROM.length(); i++){

EEPROM.write(i, 1);

}

digitalWrite(13, HIGH);

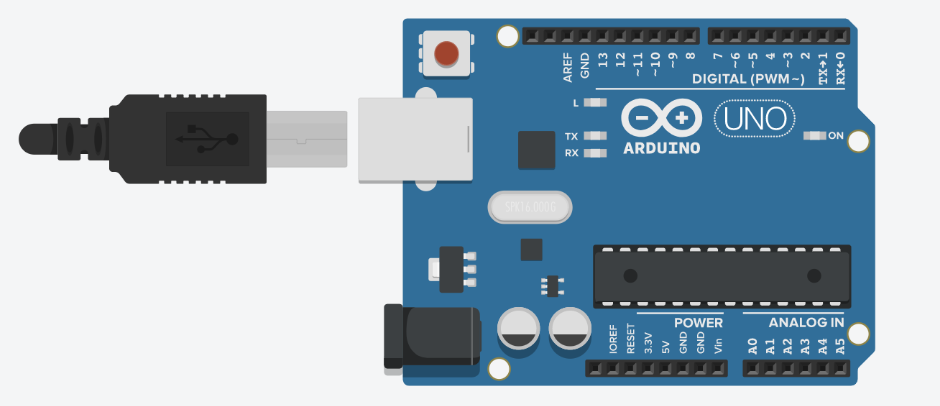
Serial.println("the write operation is successful");

}

void loop()

{

}



#include<EEPROM.h>

void setup()

{

pinMode(13, OUTPUT);

Serial.begin(9600);

Serial.println("Using Write operation in EEPROM");

for (int i=0; i<EEPROM.length(); i++){

EEPROM.write(i, 0);}

digitalWrite(13, HIGH);

int myint = 1;

EEPROM.put(0,myint);

Serial.println("The Put operation is successful");

Serial.println("Getting the value which we had put");

int get = EEPROM.get(0,myint);

Serial.println(get);

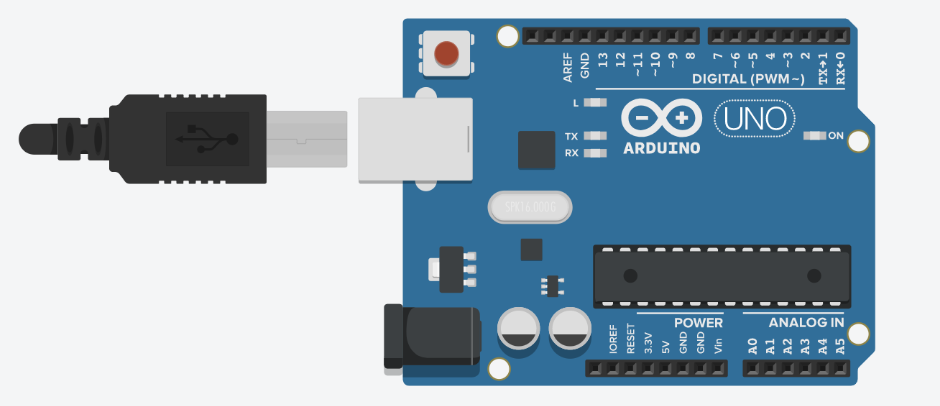
Serial.println("The Get operation is successful");

}

void loop()

{

}



#include<EEPROM.h>

void setup()

{

pinMode(13, OUTPUT);

Serial.begin(9600);

Serial.println("Using Write operation in EEPROM");

for (int i=0; i<EEPROM.length(); i++){

EEPROM.write(i, 0);}

digitalWrite(13, HIGH);

int myint = 1;

EEPROM.put(0,myint);

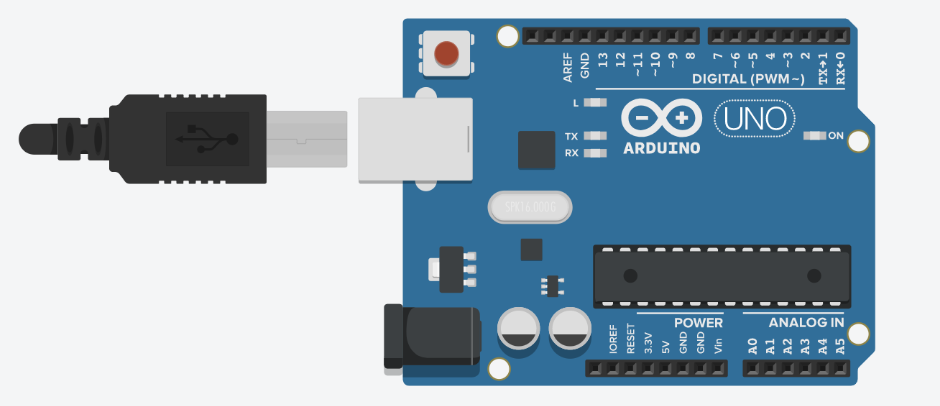
Serial.println("The Put operation is successful");

}

void loop()

{

}



#include<EEPROM.h>

void setup()

{

pinMode(13, OUTPUT);

Serial.begin(9600);

Serial.println("Using Write operation in EEPROM");

for (int i=0; i<EEPROM.length(); i++){

EEPROM.write(i, 0);}

Serial.println("Using Update operation in EEPROM");

for (int i=0; i<EEPROM.length(); i++){

EEPROM.update(i, 1);

}

digitalWrite(13, HIGH);

Serial.println("The Update operation is successful");

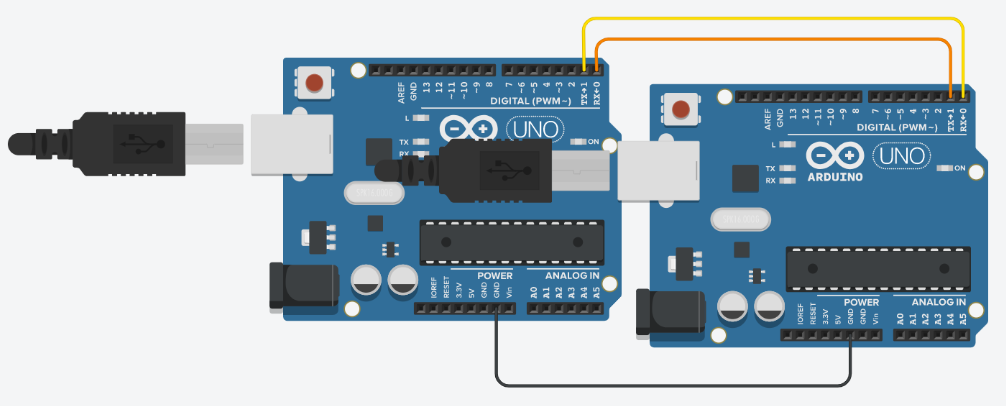
}

void loop()

{

}

1. UART - sending and receiving messages



#include<Wire.h>

#define ADDR 1

// Master arduino

void setup()

{

Wire.begin();

Serial.begin(9600);

}

void loop()

{

Wire.requestFrom(ADDR, 6);

while (Wire.available()) {

char c = Wire.read();

Serial.print(c);

}

Serial.println();

delay(500);

}

#include<Wire.h>

#define ADDR 1

//Slave

void setup(){

Wire.begin(ADDR);

Wire.onRequest(requestEvent);

}

void loop() {

delay(500);

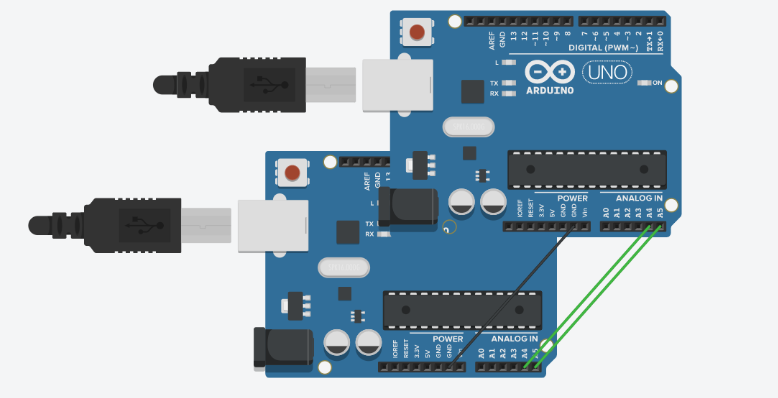
}

void requestEvent() {

Wire.write("hello ");

}

1. I2C - sending and receiving messages



#include<Wire.h>

#define ADDR 1

// Master arduino

void setup()

{

Wire.begin();

Serial.begin(9600);

}

void loop()

{

Wire.requestFrom(ADDR, 6);

while (Wire.available()) {

char c = Wire.read();

Serial.print(c);

}

Serial.println();

delay(500);

}

#include<Wire.h>

#define ADDR 1

//Slave

void setup(){

Wire.begin(ADDR);

Wire.onRequest(requestEvent);

}

void loop() {

delay(500);

}

void requestEvent() {

Wire.write("hello ");

}