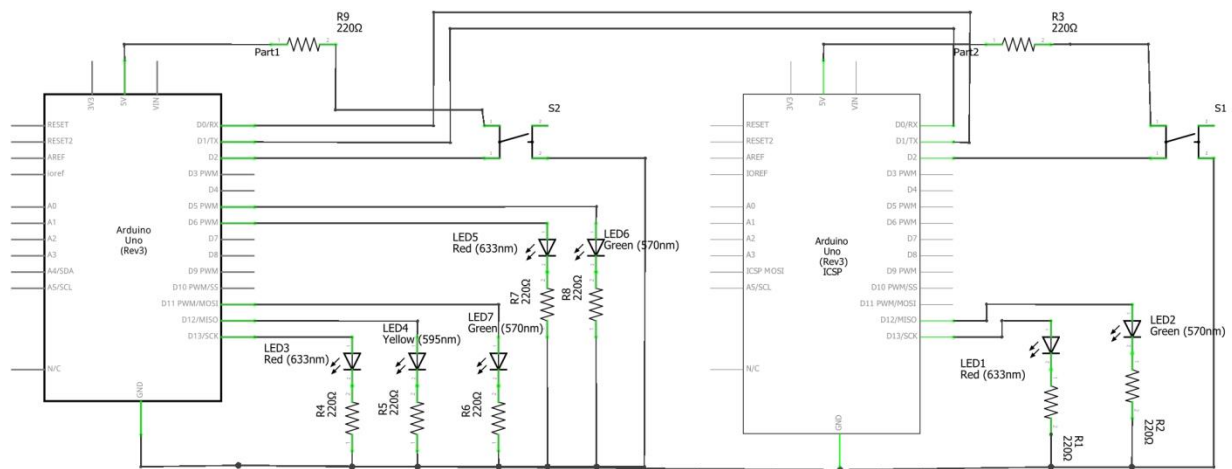


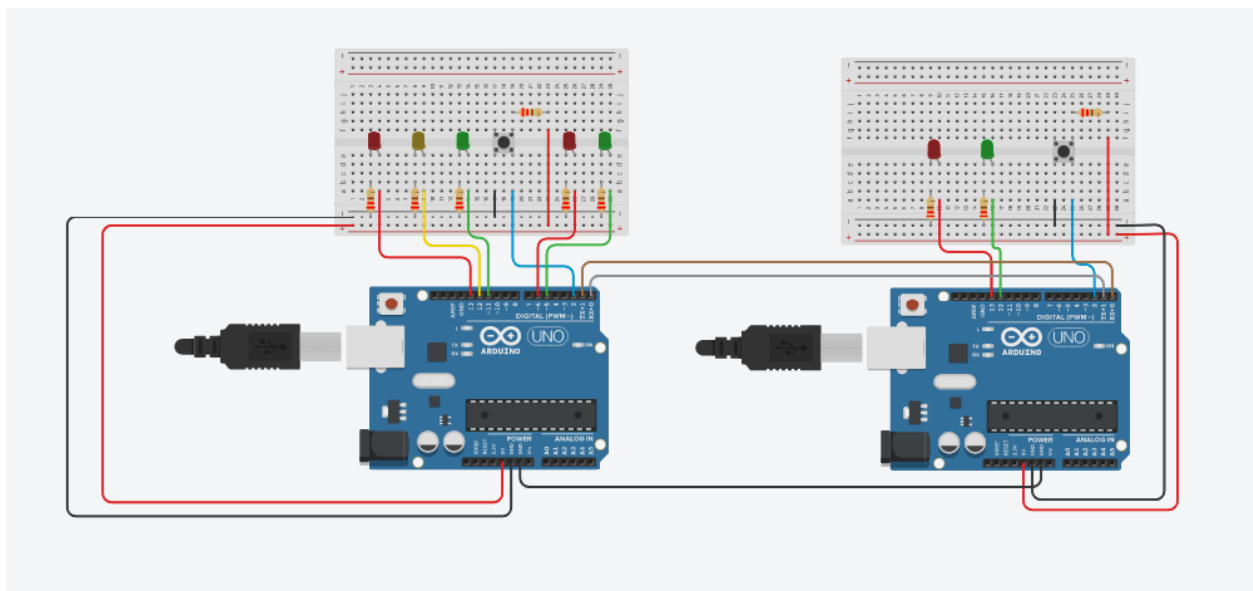
Izveštaj projektnog zadatka

Student: Milan Savić

Broj indeksa: 2016203834



Na slici je prikazana šema povezivanja kola za realizaciju projekta



Na slici je prikazan fizički izgled povezanih komponenti

U nastavku je dat kod implementacije Arduino uređaja koji kontroliše rad drugog Arduino uređaja na kojem je prikazana simulacija rada semafora:

```
const int CrvenaPinNo = 13;
const int ZelenaPinNo = 12;
const int DugmePinNo = 2;

volatile int stanjeCrvena = LOW;
volatile int stanjeZelena = HIGH;

void promeni() {
    stanjeCrvena = !stanjeCrvena;
    stanjeZelena = !stanjeZelena;
    digitalWrite(CrvenaPinNo, stanjeCrvena);
    digitalWrite(ZelenaPinNo, stanjeZelena);
    if(stanjeZelena){
        Serial.print(1,DEC); //salje poruku da aktivira semafor
    }else{
        Serial.print(0,DEC); //salje poruku da deaktivira semafor
    }
}

void setup()
{
    Serial.begin(9600);

    pinMode(CrvenaPinNo, OUTPUT);
    pinMode(ZelenaPinNo, OUTPUT);
    pinMode(DugmePinNo, INPUT);

    digitalWrite(CrvenaPinNo, stanjeCrvena);
    digitalWrite(ZelenaPinNo, stanjeZelena);

    Serial.print(1,DEC); //salje poruku da aktivira semafor
    attachInterrupt(digitalPinToInterrupt(DugmePinNo), promeni, FALLING);
}

void loop()
{
}
```

Na početku se definišu konstante koje određuju na kom pinu su povezane diode i taster kao i promenljive koje definišu stanje dioda. U metodi setup() definiše se brzina komunikacije između dva Arduino uređaja, definiše se pinMode svake diode i tastera, upisuju se stanja na diodama (početno stanje je zelena upaljena, crvena ugašena), šalje se poruka drugom Arduino uređaju da aktivira semafor. Pritiskom na dugme poziva se metoda promeni() koja menja stanje na diodama i u zavisnosti od stanja zelene diode šalje poruku drugom Arduino uređaju.

U nastavku je dat kod implementacije Arduino uređaja na kojem se simulira rad semafora:

```
const int CrvenaPinNo = 13;
const int ZutaPinNo = 12;
const int ZelenaPinNo = 11;
const int CrvenaPinNo2 = 6;
const int ZelenaPinNo2 = 5;
const int DugmePinNo = 2;

const int VremeZeleno = 10000;
const int VremeZuto = 3000;
const int VremeZelenoPesak = 10000;
const int VremeZaPromenu = 2000;
const int VremeZaPromenu2 = 1000;
const int VremeZutaTreperi = 500;

volatile int stanjeZuta = LOW; // u neaktivnom stanju zuta dioda treperi

unsigned long pocetno;
unsigned long trenutno;
unsigned long vreme;

String poruka = "";

volatile bool dugmeAktivno = false;
bool pocetnoStanje = false;
bool neaktivnoStanje = false;

void promeni(){
    if(!neaktivnoStanje){//ukoliko je deaktiviran semafor
        dugmeAktivno = true;//pritisak na dugme nema funkciju
    }
}

void setup()
{
    Serial.begin(9600);
    pinMode(CrvenaPinNo, OUTPUT);
    pinMode(ZutaPinNo, OUTPUT);
    pinMode(ZelenaPinNo, OUTPUT);
    pinMode(CrvenaPinNo2, OUTPUT);
    pinMode(ZelenaPinNo2, OUTPUT);
    pinMode(DugmePinNo, INPUT);

    digitalWrite(CrvenaPinNo, LOW);
    digitalWrite(ZutaPinNo, LOW);
    digitalWrite(ZelenaPinNo, LOW);
    digitalWrite(CrvenaPinNo2, LOW);
    digitalWrite(ZelenaPinNo2, LOW);

    attachInterrupt(digitalPinToInterrupt(DugmePinNo), promeni, FALLING);
}
```

```

void aktivan(){
    trenutno = millis();
    vreme = trenutno - pocetno;
    while(vreme<VremeZeleno){//provera da li je proslo 10s
        trenutno = millis();
        vreme = trenutno - pocetno;
    }

    digitalWrite(ZelenaPinNo, LOW);
    digitalWrite(ZutaPinNo, HIGH);
    delay(VremeZuto);

    digitalWrite(ZutaPinNo, LOW);
    digitalWrite(CrvenaPinNo, HIGH);
    delay(VremeZaPromenu);

    digitalWrite(CrvenaPinNo2, LOW);
    digitalWrite(ZelenaPinNo2, HIGH);
    delay(VremeZelenoPesak);

    digitalWrite(ZelenaPinNo2, LOW);
    digitalWrite(CrvenaPinNo2, HIGH);
    delay(VremeZaPromenu2);

    digitalWrite(CrvenaPinNo, LOW);
    digitalWrite(ZutaPinNo, HIGH);
    delay(VremeZuto);

    digitalWrite(ZutaPinNo, LOW);
    digitalWrite(ZelenaPinNo, HIGH);

    pocetno = millis();

    dugmeAktivno = false;
}

void neaktivan(){
    //gase se sve lampice
    digitalWrite(CrvenaPinNo, LOW);
    digitalWrite(ZutaPinNo, LOW);
    digitalWrite(ZelenaPinNo, LOW);
    digitalWrite(CrvenaPinNo2, LOW);
    digitalWrite(ZelenaPinNo2, LOW);

    //promenljiva koja onemogucava pešaka da promeni stanje na semaforu
    neaktivnoStanje = true;

    while(poruka=="0"){
        stanjeZuta = !stanjeZuta;
        digitalWrite(ZutaPinNo, stanjeZuta);
        delay(VremeZutaTreperi);
        if(Serial.available()>0){
            poruka = Serial.readString();
        }
    }
}

```

```

void loop()
{
    if(Serial.available()>0){
        poruka = Serial.readString();
    }
    if(poruka == "1"){
        if(!pocetnoStanje){
            digitalWrite(CrvenaPinNo, LOW);
            digitalWrite(ZutaPinNo, LOW);
            digitalWrite(ZelenaPinNo, HIGH);
            digitalWrite(CrvenaPinNo2, HIGH);
            digitalWrite(ZelenaPinNo2, LOW);
            pocetno = millis(); //pocinje da meri vreme za zeleno svetlo za vozila
            pocetnoStanje = true;
        }
        if(dugmeAktivno){
            aktivan();
        }
    }else if(poruka == "0"){
        neaktivn();
        pocetnoStanje = false;
        neaktivnoStanje = false;
    }
}
}

```

Na početku se definišu konstante koje predstavljaju pinove dioda i prekidača, vreme koliko diode svetle, promenljive koje prikazuju stanje žute diode, promenljive koje se koriste za merenje vremena minimalnog prikazivanja zelenog svetla za vozila, promenljiva u kojoj se smešta poruka drugog Arduino uređaja kao i nekoliko kontrolnih promenljivih.

U metodi setup() definiše se brzina komunikacije između dva Arduino uređaja, definiše se pinMode svake diode i tastera, upisuju se stanja na diodama. Dok se ne pročita poruka od kontrolnog Arduino uređaja sve diode su isključene. Definiše se prekid na pinu tastera koja poziva metodu promeni().

Metoda promeni() postavlja vrednost promenljive dugmeAktivno na true ukoliko je semafor u aktivnom režimu rada.

U metodi loop() proverava se da li postoji poruka od kontrolnog Arduino uređaja. Ukoliko postoji poruka se smešta u promenljivu poruka.

Ukoliko je vrednost poruke "1" proverava se da li je semafor u početnom stanju, ukoliko nije diode na semaforu se inicijalizuju tako da su zelena dioda za vozila i crvena za pešake upaljene, ostale su ugašene i počinje da se meri vreme trajanja zelenog svetla za vozila. U aktivnom režimu rada proverava se da li je registrovan poziv pešaka. Ukoliko jeste poziva se metoda aktivn() u kojoj se ceka da istekne predviđeno minimalno trajanje zelenog svetla za vozila, nakon toga diode menjaju svoja stanja po određenim pravilima koja su zadata u zadatku. Nakon vraćanja u početno stanje (zelena dioda za vozila i crvena za pešake su upaljene) počinje da se meri vreme trajanja zelenog svetla. Za vreme menjanja stanja na diodama u ovoj metodi

pritisak na taster nema funkciju, pošto se na kraju metode vrednost promenljive `dugmeAktivno` postavlja na `false`.

Ukoliko je vrednost poruke "0" poziva se metoda `neaktivan()` u kojoj je implementirano da žuta dioda treperi. To će se dešavati sve dok Arduino uređaj ne dobije poruku za promenu stanja. Nakon povratka u `loop()` metodu definiše se da ne postoji početno stanje za aktivan režim rada semafora.