



Univerzitet Singidunum
Tehnički fakultet

Arduino sistem igre ponavljanja sekvence

- Projektni rad -

Predmet: Sistemi za rad u realnom vremenu

Profesor:
prof. dr Marko Tanasković

Student:
Marko Dojkić 2018/201682

Asistent:
Uroš Dragović

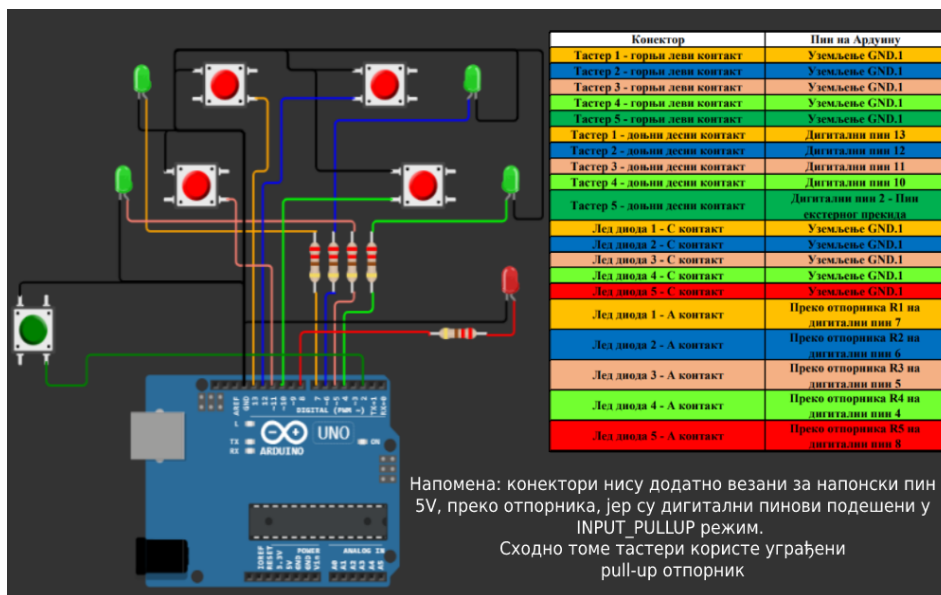
Beograd, 2022. godine

Sadržaj

1. Uvod i šema povezivanja sistema.....	3
2. Prikaz i objašnjenje koda	4
3. Prikaz i objašnjenje rada sistema	9
4. Literatura.....	13

1. Uvod i šema povezivanja sistema

Ovaj sistem predstavlja igru ponavljanja sekvence (eng. Simon game). Ceo projekat je realizovan pomoću wokwi simulatora i dostupan za testiranje na sledećem linku: <https://wokwi.com/projects/327318989616185938>.



Slika 1: Prikaz šeme povezivanja

Za potrebe projekta korišćen je **Arduino Uno** na koji su povezane sledeće komponente:

- 4 tastera čijim pritiskom omogućava treptaj diode pored svakog sa ciljem da se ponovi sekvencna od strane korisnika
- 1 zeleni taster čijim pritiskom se preko eksternog prekida započinje nova igra, odnosno sledeći nivo tokom igre
- 4 zelene diode koje prvo trepte po sekvenci, a onda ručno preko svojih tastera
- 1 crvena dioda koja signalizuje stanje greške u 3 slučaja:
 - 1) Treptanje* nakon pritiska tastera pogrešne diode koja nije sledeća u sekvenci
 - 2) Treptanje* nakon isteka vremena (3.5 sekunde) između pritiska dva tastera zaredom pri ponavljanju sekvence
 - 3) Trajno svetljenje na kraju igre

* Dioda će trepnuti onoliko puta koliko je pokušaja iskorišćeno, a maksimalno 3 puta.

2. Prikaz i objašnjenje koda

```
#include <TimerOne.h> //Укључивање библиотеке која нам омогућава рад са Timer1
#define ERROR_LED_PIN 8 //Дефинисање пина диоде која означава стање грешке
#define EXTERNAL_INTERRUPT_BUTTON_PIN 2 //Дефинисање пина тастера
екстерног прекида
#define MAX_GAME_DIODE_COUNT 100 //Дефинисање максималног нивоа (100
диода у секвенци)
#define NUMBER_OF_DIODES 4 //Дефинисање броја диода

const byte ledPins[NUMBER_OF_DIODES] = {7, 6, 5, 4}; //Дефинисање низа пинова
диода
const byte buttonPins[NUMBER_OF_DIODES] = {13, 12, 11, 10}; //Дефинисање низа
пинова тастера (редослед одговара редоследу диода)
byte gameSequence[MAX_GAME_DIODE_COUNT]; //Дефинисање низа који чува
секвенцу диода (максималне дужине 100)
byte lastCorrectDiodeInSequenceIndex, currentSequenceDiodeCount, triesCount;
/* Дефинисање променљивих који чувају индекс последње погођење диоде у секвенци,
тренутног броја диода у секвенци и број покушаја погађања */
bool isUserTimeoutAllowed; //Дефинисање променљиве која проверава да ли је
дозвољено активирати временски прекид

enum {
    IDLE_STATE,
    BEFORE_GAME_START,
    NEW_LEVEL_STARTED,
    WAITING_FOR_USER_SEQUENCE,
    BEFORE_NEXT_LEVEL,
    ON_GAME_END,
} state; //Дефинисање променљиве enum типа која означава могућа стања током игре

void blinkDiode(byte diodePin){ //Функција која омогућава трептај једне диоде
    digitalWrite(diodePin, HIGH); //Паљење диоде
    delay(500); //Пауза
    digitalWrite(diodePin, LOW); //Гашење диоде
    delay(500); //Пауза
}

void playCurrentSequence() { //Функција која пролази кроз секвенцу и пали појединачне
диоде
    for (int i = 0; i < currentSequenceDiodeCount; i++) { //Петља за пролазак кроз секвенцу
        blinkDiode(ledPins[gameSequence[i]]);
    }
    state = WAITING_FOR_USER_SEQUENCE; //Промена стања у стање чекања да
корисник понови секвенцу
}
```

```

void startGame() { //Функција екстерног за враћање првобитних вредности приликом
покретања нове игре
    triesCount = 0;
    lastCorrectDiodeInSequenceIndex = 0;
    memset(gameSequence, 0, sizeof(gameSequence));
    gameSequence[0] = random(0, NUMBER_OF_DIODES); //Генерисање прве вредности у
низу секвенци...
    currentSequenceDiodeCount = 1; //...и померање индекса у низу секвенци за један јер се
почетна секвенца састоји од 2 диоде
    digitalWrite(ERROR_LED_PIN, LOW);
    digitalWrite(EXTERNAL_INTERRUPT_BUTTON_PIN, LOW);
    detachInterrupt(digitalPinToInterrupt(EXTERNAL_INTERRUPT_BUTTON_PIN));
//Уклањамо екстерни прекид јер нам више није потребан
    startNextLevel();
}

void startNextLevel(){ //Функција екстреног прекида за прелазак на следећи ниво
    delay(5*1000*1000);
    state = NEW_LEVEL_STARTED; //Прелазак у стање почетка новог нивоа
}

void wrongButtonPressed() { //Функција која се позива услед притиска погрешног
тастера или путем временског прекида
    lastCorrectDiodeInSequenceIndex = 0; //Ресетовање бројача тренутне диоде коју је
корисник последње погодио
    Serial.print(", почните секвенцу испочетка - Имате још "); //Наредне три линије
исписују поруку о преосталом броју покушаја
    Serial.print(3 - ++triesCount);
    Serial.println(" покушај(a)! -");
    for (int i = 0; i < triesCount; i++) { //Диода која представља грешку трепће онолико пута
колико је покушаја искоришћено
        blinkDiode(ERROR_LED_PIN);
    }
    isUserTimeoutAllowed = false; //Забрана активирања временског прекида (активираће
се поново када корисник притисне први исправан тастер поново)
    if (triesCount == 3) state = ON_GAME_END; //Уколико је истекао број покушаја
систем прелази у стање завршетка
}

void checkUserSequenceTimeout() { //Прекидна рутина која настаје након истека
времена тајмера
    if (isUserTimeoutAllowed) { //Провера да ли је дозвољено активирати временски
прекид
        Serial.print("- Сувише сте дуго чекали да притиснете следећи тастер");
        wrongButtonPressed(); //Позив функције грешке
    } //Овај временски прекид се активира када корисник чека сувише дуго (3.5 секунди)
између понављања две узастопне диоде у секвенци
}

```

```

void setup() { //Функција која се увек позива приликом покретања ардуина
  Serial.begin(9600); //Иницијализација серијског приказа са брзином преноса података
од 9600 bps.
  for(int i = 0; i < NUMBER_OF_DIODES; i++){
    pinMode(ledPins[i], OUTPUT); //Постављање дигиталних пинова диода као излазне
    pinMode(buttonPins[i], INPUT_PULLUP); //https://create.arduino.cc/projecthub/Hack-
star-Arduino/push-buttons-and-arduino-a-simple-guide-wokwi-simulator-
c2281f?ref=user&ref_id=1743724&offset=0
    //digitalWrite(buttonPins[i], HIGH); //Уколико би изнад био моде INPUT потребна је и
ова линија }
    pinMode(ERROR_LED_PIN, OUTPUT);
    pinMode(EXTERNAL_INTERRUPT_BUTTON_PIN, INPUT_PULLUP); //Користи се
интерни pull-up отпорник
    //digitalWrite(EXTERNAL_INTERRUPT_BUTTON_PIN, HIGH); //Уколико би изнад био
моде INPUT потребна је и ова линија
    attachInterrupt(digitalPinToInterrupt(EXTERNAL_INTERRUPT_BUTTON_PIN),
startGame, FALLING); //Додељује се функција покретања нове игре екстерног прекида
зеленом тастеру која се активира детектовањем силазне ивице напона (1 --> 0)
    randomSeed(analogRead(A0)); //Иницијализација генератора псеудо сличајних бројева
коришћењем вредности прочитане на аналогном пину A0 као seed
    Timer1.initialize(3500000); //Иницијализација тајмера за временски прекид приликом
дугог чекања на 3.5 секунди
    Timer1.attachInterrupt(checkUserSequenceTimeout); //Везивање функције као прекидну
рутину тајмера
    Timer1.stop(); //Прекид тајмера до почетка игре
    Serial.println("*** Добродошли у игру ***"); //Приказ поруке добродошлице
    state = BEFORE_GAME_START; //Промена стања на стање почетка нове игре
  }
void loop() {
  switch (state) { //Провера стања у коме се тренутно налази систем
    case BEFORE_GAME_START: //Случај када је систем у стању пре почетка нове игре
      Serial.println("*** Можете почети нову игру притиском зеленог тастер. Када игра
почне будите спремни да испратите прву секвенцу која се састоји од две упаљене
диоде. Срећно! ***");
      state = IDLE_STATE; //Прелазак у стање мировања
      break;
    case NEW_LEVEL_STARTED: //Случај када је систем у стању почетка новог нивоа
      Serial.print("* НИВО "); //Наредне 4 линије исписују поруку која показује који је
тренутни ниво
      Serial.print(currentSequenceDiodeCount);
      Serial.print(" ЈЕ ДОСТИГНУТ - ");
      Serial.println("ГЛЕДАЈТЕ ПАЖЉИВО ДИОДЕ! *");
      gameSequence[currentSequenceDiodeCount++] = random(0, NUMBER_OF_DIODES);
//Додавање нове вредности индекса диода у секвенцу и померање показивача на
највећи индекс за један
      state = IDLE_STATE; //Прелазак у стање мировања
      playCurrentSequence(); //Позив функције за трептање диода
      break;

```

```

    case WAITING_FOR_USER_SEQUENCE: //Случај када је систем у стању чекања на
    понављање секвенце од стране корисника
        Timer1.start(); //Покретање временског бројача
        do {
            for (byte i = 0; i < NUMBER_OF_DIODES; i++) { //Петља која пролази кроз све
            тастере и проверава да ли је неки од њих притиснут
                if (digitalRead(buttonPins[i]) == LOW) { //Уколико је тастер притиснут дешава се
                провера
                    blinkDiode(ledPins[i]);
                    if (i != gameSequence[lastCorrectDiodeInSequenceIndex]) { //Уколико та диода
                    није тренутна у секвенци...
                        Serial.print("- Порешна диода"); //Исписујемо поруку и...
                        wrongButtonPressed(); //...позивамо функцију грешке
                    }
                    else {
                        Timer1.restart(); //Рестартовање бројача тајмера
                        isUserTimeoutAllowed = true; //Давање дозволе та временски прекид
                        lastCorrectDiodeInSequenceIndex++; //Уколико је диода тачна померамо
показивач на следећу
                    } } }
                    delay(1); //Неопходна пауза приликом провере
                } while (lastCorrectDiodeInSequenceIndex < currentSequenceDiodeCount &&
                triesCount < 3); //Петља се понавља докле год се не понови цела секвенца или истекне
                број покушаја
                Timer1.stop(); //Прекидање временског бројача
                if (state == WAITING_FOR_USER_SEQUENCE) state = BEFORE_NEXT_LEVEL;
                //Прелазак у стање пре почетка следећег нивоа (уколико се идаље налазимо у овом
                стању*)
                break; /* Могуће је да се нађемо и у стању завршетка игре уколико је искоришћен
                максимални број покушаја, а није погођена секвенца
                case BEFORE_NEXT_LEVEL: //Случај када је систем у стању пре почетка следећег
                нивоа
                    Serial.println("+ ЧЕСТИТКЕ СЕКВЕНЦА ЈЕ ТАЧНА * Када будете спремни
                    притисните зелени тастер за наставак +"); //Испис поруке честитке
                    triesCount = 0; //Ресетовање броја покушаја
                    lastCorrectDiodeInSequenceIndex = 0; //Ресетовање индекса тренутно погођене диоде
                    у секвенци
                    if (currentSequenceDiodeCount >= MAX_GAME_DIODE_COUNT) { //Провера да ли
                    смо прешли послени ниво
                        isUserTimeoutAllowed = false;
                        state = ON_GAME_END; //Прелазак у стање завршетка игре
                    } else {
                        attachInterrupt(digitalPinToInterrupt(EXTERNAL_INTERRUPT_BUTTON_PIN),
                        startNextLevel, RISING); //Додељује се функција промене нивоа екстерног прекида
                        зеленом тастеру која се активира детектовањем силазне ивице напона (1 --> 0)
                        state = IDLE_STATE; //Прелазак у стање мировања
                    }
                    break;

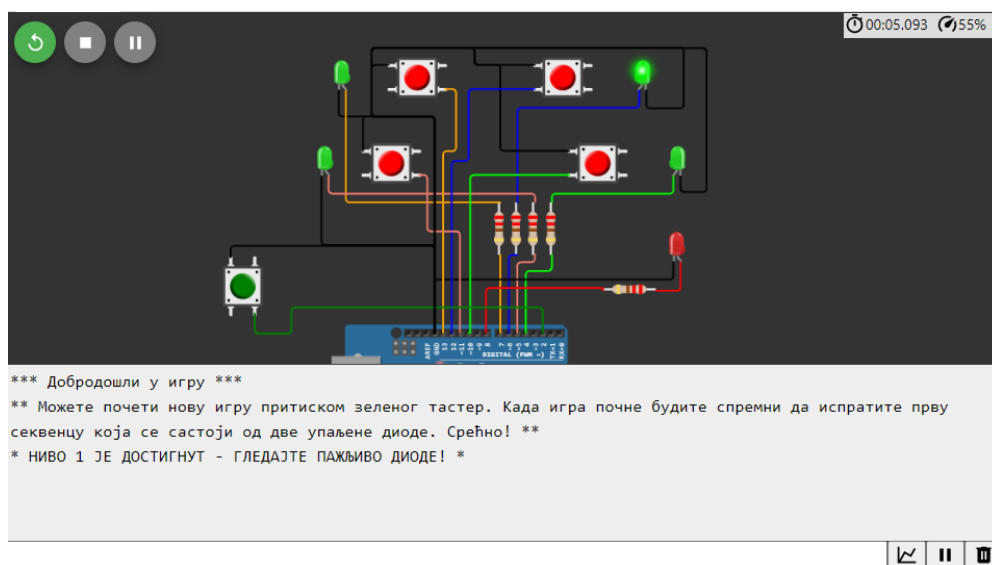
```

```

case ON_GAME_END: //Случај када је систем у стању завршетка игре
    digitalWrite(ERROR_LED_PIN, HIGH); //Диода грешке остаје заувек упаљена
    if (triesCount == 3) { //Провера да ли смо у ово стање дошли услед истека броја
покушаја
        Serial.print("*** Игра је завршена услед истека броја покушаја! Достигли сте ниво: ");
        Serial.print(currentSequenceDiodeCount - 2); //Последњи достигнути ниво одговара
тренутном броју диода у секвенци умањен за 2, јер не рачунамо последњи ниво и због
чињенице да низ креће од нуле
        Serial.println(" ***");
    } //Или смо успешно прешли све нивое
    else Serial.println("*** Игра је успешно завршена! Успешно сте поновили
целокупну секвенцу. Све честитке! ***");
    attachInterrupt(digitalPinToInterrupt(EXTERNAL_INTERRUPT_BUTTON_PIN),
startGame, FALLING); //Додељује се функција покретања нове игре екстерног прекида
зеленом тастеру која се активира детектовањем силазне ивице напона (1 --> 0)
    state = IDLE_STATE; //Прелазак у стање мировања (у овом случају ће то бити
крајње стање)
    break;
case IDLE_STATE: break; //Случај када је систем у стању мировања (ништа се не
дешава већ се понавља празна петља)
default: //Непознато стање (највероватније систем никада неће доћи у ово стање)
    for(int i = 0; i < NUMBER_OF_DIODES; i++){
        digitalWrite(ledPins[i], HIGH);
    }
    Serial.println("**** Грешка: Систем се налази у непознатом стању! Потребно је
поновно покретање система ****");
}
}

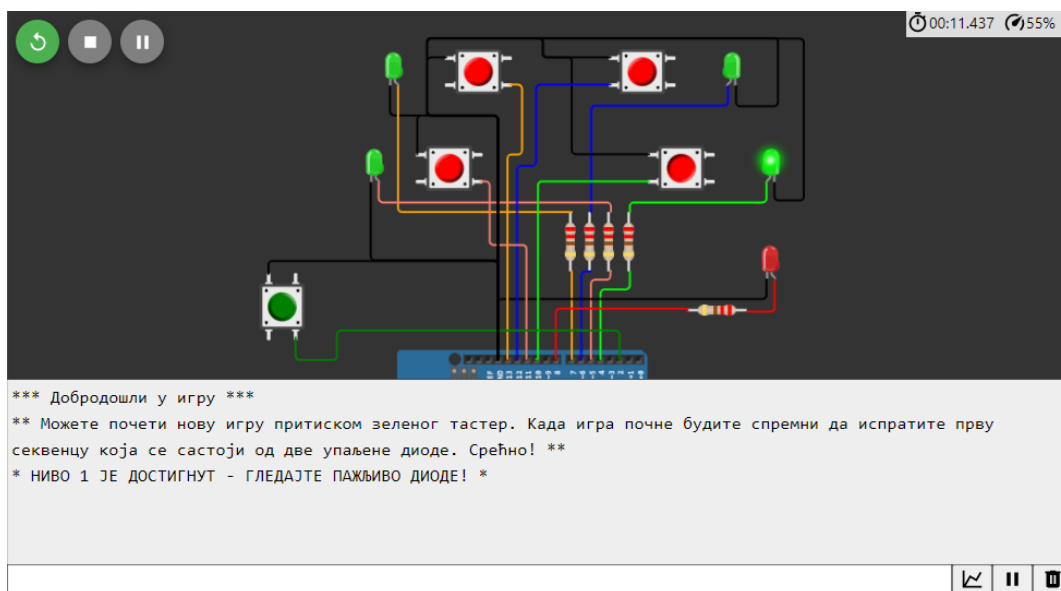
```


3. Prikaz i objašnjenje rada sistema



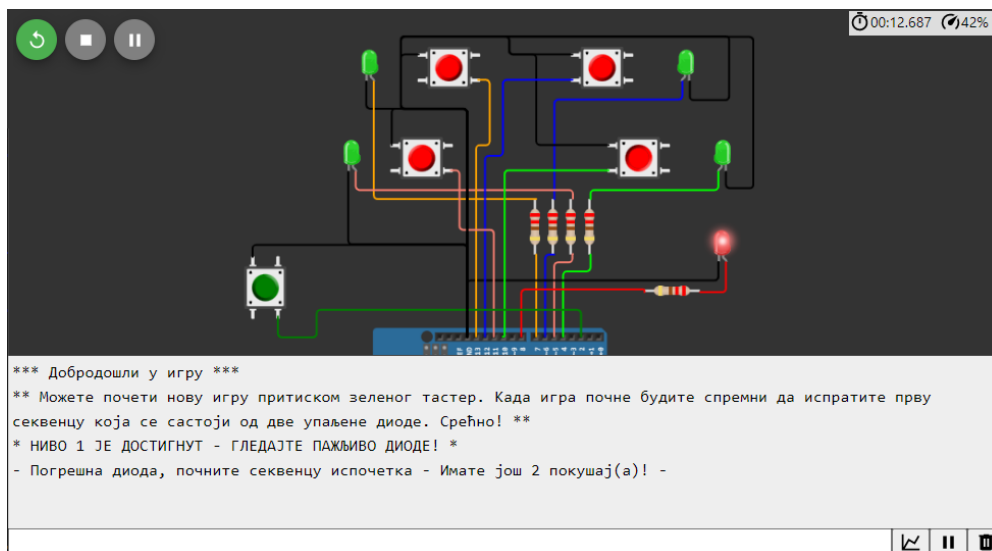
Slika 2:Prikaz stanja sistema tokom prikaza sekvence

Na slici 2 vidimo da se na serijskoj vezi ispisala poruka dobrodošlice, koja se ispisuje jedanput prilikom pokretanja sistema, potom poruka da nova igra počinje pritiskom na zeleni taster. Pre pritiska sistem je u stanju mirovanja, a nakon pritiska poziva se funkcija eksternog prekida, a onda se ispisuje poruka da je dostignut prvi nivo. Poslednja poruka i činjenica da gornja leva dioda svetli bez pritiska tastera nam ukazuju da se sistem trenutno nalazi u stanju početka novog nivoa u kome se prikazuje sekvenca koju korisnik treba da ponovi, a kako je tek prvi nivo ova sekvenca se sastoji od 2 diode.



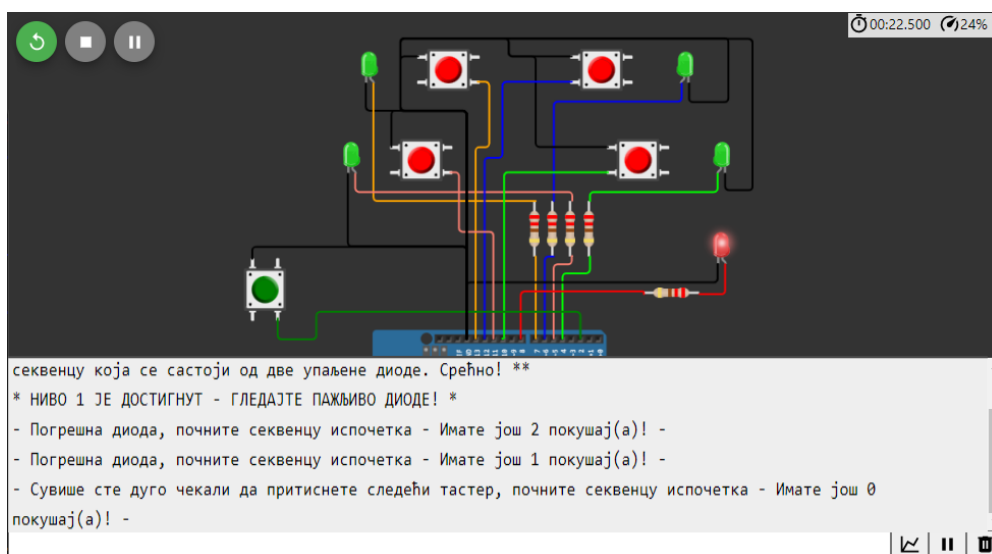
Slika 3:Prikaz stanja sistema tokom pritiska na taster za treptanje 4-te diode

Nakon što se sekvenca prikazala sistem dolazi u stanje čekanja na ponavljanje iste od strane korisnika. Na slici 3 vidimo da je korisnik odlučio da pritisne 4-ti taster i aktivira donju desnu diodu. Na osnovu slike 2 možemo zaključiti koje će sledeće stanje sistema biti.



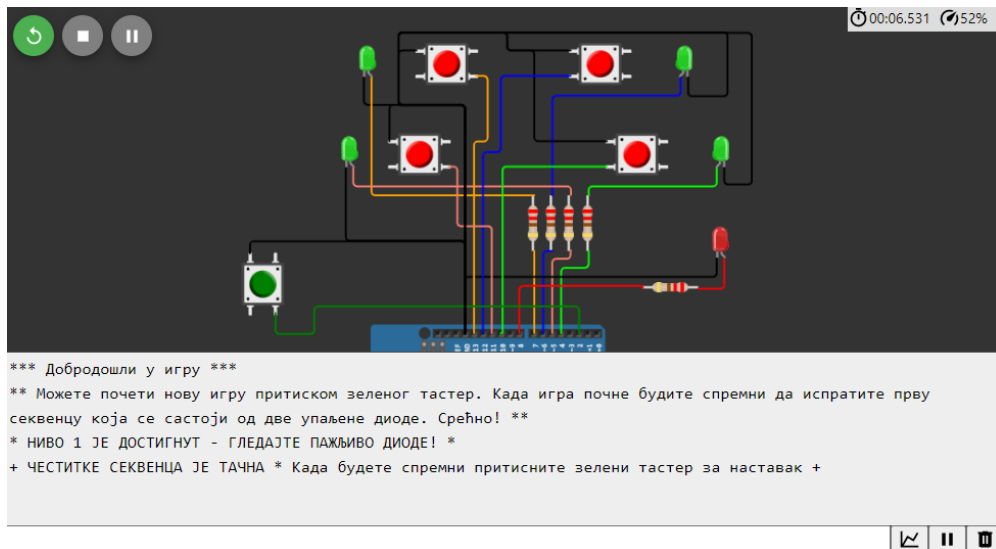
Slika 4: Detektovana je greška koja je nastala usled odabira pogrešne diode u sekvenci

Kao što smo mogli da pretpostavimo odabir je pogrešan, jer sekvenca počinje 2-om, odnosno gornjom desnom diodom. Shodno time dogodio se prekid i crvena dioda greške je trepnula jedanput, jer je korisniku iskorišćen tek prvi pokušaj, a potom je ispisana i adekvatna poruka na serijskoj vezi. Nakon ovoga sistem se vraća u stanje čekaња na ponavljanje sekvence od strane korisnika, s tim što se svi brojači resetuju pa korisnik mora da ponovi **celu** sekвенцу ispočetka.



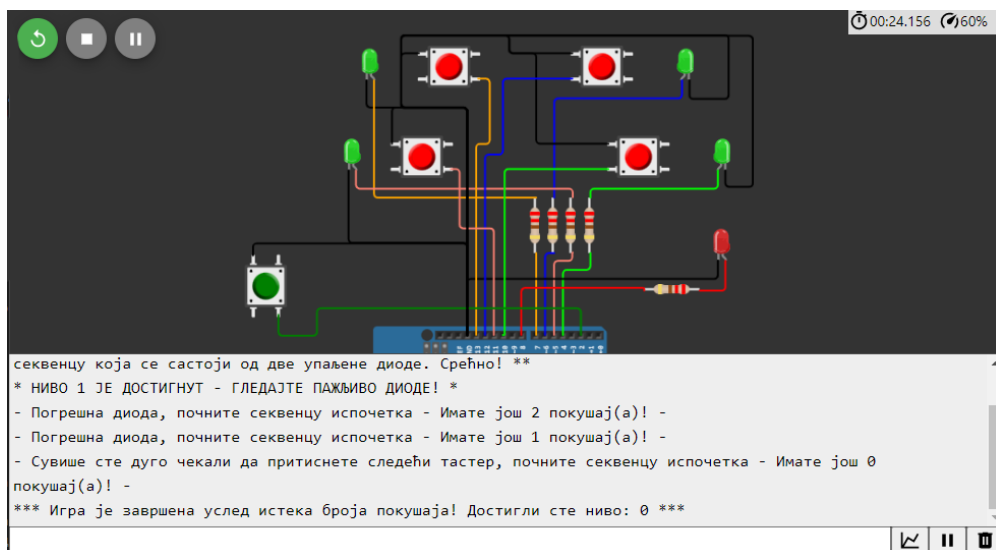
Slika 5: Detektovana je greška usled predugog čekanja između dva pritiska tastera – vremenski prekid

Na slici 5 se desila druga moguća vrsta greške koja nastaje kada korisnik, nakon pritiska jednog tastera, čeka suviše dugo da pritisne sledeći (čeka 3,5 sekundi). Ovaj tip greške je implementiran pomoću tajmera Timer1 kao vremenski prekid.



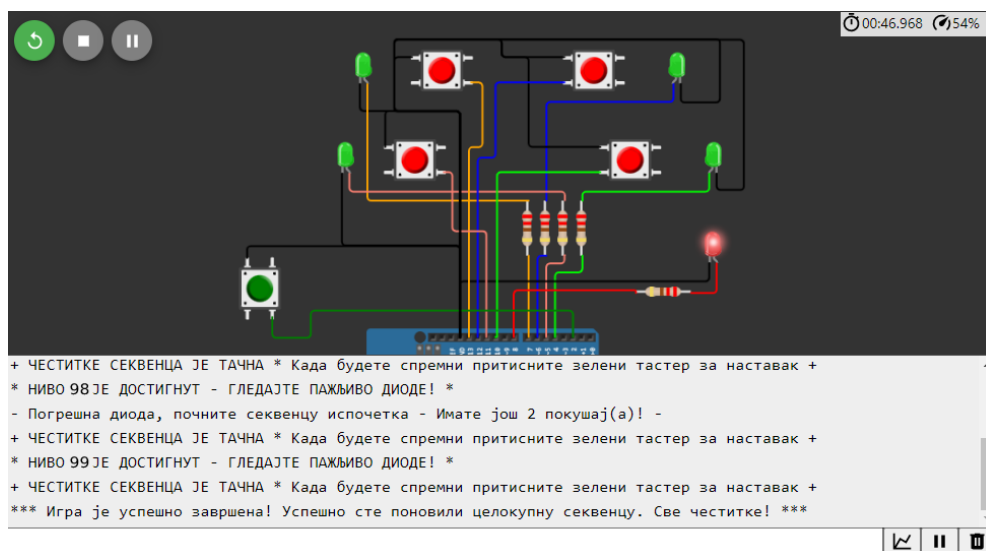
Slika 6: Prikaz stanja sistema pre prelaska na sledeći nivo

Nakon uspešnog ponavljanja sekvence od strane korisnika sistem prelazi u stanje pre prelaska na sledeći nivo. U tom stanju se samo ispisuje poruka čestitke i dodeljuje nova funkcija eksternog prekida zelenom tasteru. Pritiskom na njega, ta funkcija će se i aktivirati, a ona će resetovati brojače koji označavaju broj pokušaja i poslednje pogođenu diodu u sekvenci i promeniti stanje sistema u stanje početka novog nivoa.



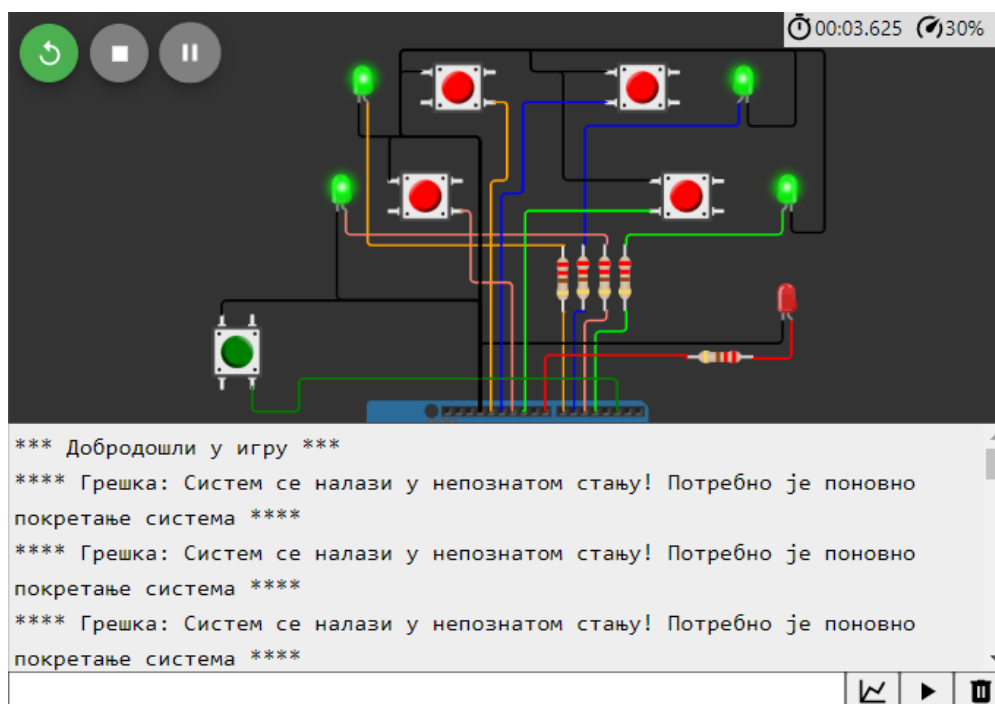
Slika 7: Prikaz stanja sistema nakon završetka igre usled isteka broja pokušaja – krajnje stanje sistema

Na slici 7 vidimo, na osnovu poruka serijske veze, da je korisnik stigao do drugog nivoa, a potom usled dovođenja sistema u stanje greške 3 puta (dva puta jer je pogrešio diodu, a treći put jer je čekaо suviše dugo da pritisne sledeći taster) doveo sistem u stanje završetka igre.



Slika 8: Prikaz stanja sistema nakon uspešnog završetka igre

Stanje na slici 8 je simulirano promenom maksimalnog broja dioda, a potom ručno ispisan nivo. U stvarnosti bi prikaz bio identičan, ali bi on zahtevao pogađanje sekvence koja se sastoji od 100 dioda, što je za prosečnog korisnika gotovo nemoguće.



Slika 9: Prikaz nepoznatog stanja sistema

Sistem nikada ne bi trebalo da se nađe u stanju prikazanom na slici 9. Za potrebe demonstracije ovoga stanja kod je namerno promenjen, tj. Izbrisan je prvi slučaj switch naredbe, a promenljivi state je dodeljena ta vrednost, a kako ona više nepostoji pozvan je podrazumevani slučaj.

4. Literatura

1. <https://docs.wokwi.com/> - Dokumentacija korišćenja simulatora - pregledano dana 28.03.2022.
2. https://create.arduino.cc/projecthub/Hack-star-Arduino/push-buttons-and-arduino-a-simple-guide-wokwi-simulator-c2281f?ref=user&ref_id=1743724&offset=0 – Pojašnjenje načina implementacije tastera pomoću Arduina - pregledano dana 28.03.2022.
3. <https://microcontrollerslab.com/arduino-timer-interrupts-tutorial> - Upustvo za implementaciju vremenskog prekida pomoću Arduina - pregledano dana 28.03.2022.
4. https://create.arduino.cc/projecthub/tolentinocotesta/let-s-learn-how-to-use-finite-state-machine-with-arduino-c524ac?ref=part&ref_id=8233&offset=13 - Upustvo za implementaciju konačne mašine stanja pomoću Arduina - pregledano dana 28.03.2022
5. <https://www.arduino.cc/reference/en/language/functions/external-interrupts/attachinterrupt> - Upustvo za implementaciju eksternog prekida pomoću Arduina - pregledano dana 28.03.2022