



ПЪРВИ СЪПКИ С АРДУИНО

ЗАНЯТИЕ №2 ЦИФРОВИ ВХОДОВЕ И ИЗХОДИ В АРДУИНО

ВЕНЦИСЛАВ НАЧЕВ



КАКВО ЩЕ ПРАВИМ ДНЕС?

- ❖ Електрически сигнали
- ❖ Цифров изход
- ❖ Цифров вход
- ❖ Pull-up и pull-down резистори
- ❖ Променливи
- ❖ Условен оператор
- ❖ Управление на светодиоди с бутон
- ❖ Реализиране на светофар със светодиоди

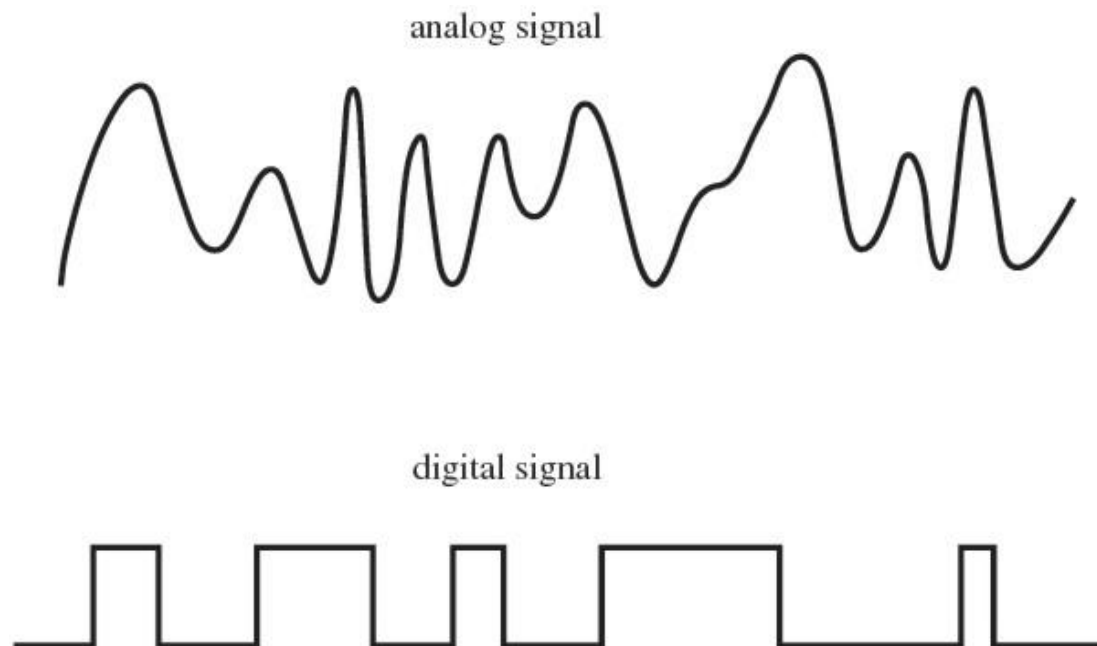
ЕЛЕКТРИЧЕСКИ СИГНАЛ

- ❖ Сигналят е физически процес който пренася информация или с енергията си е предназначен да управлява даден процес или устройство.

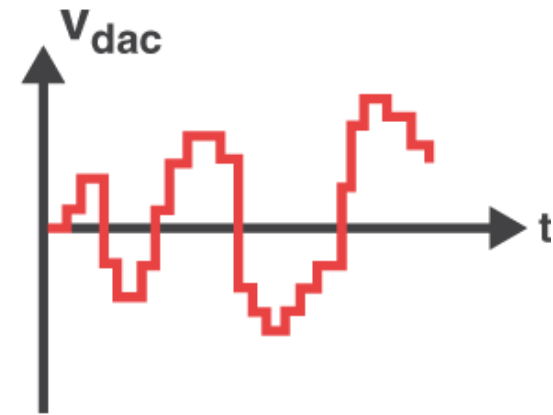
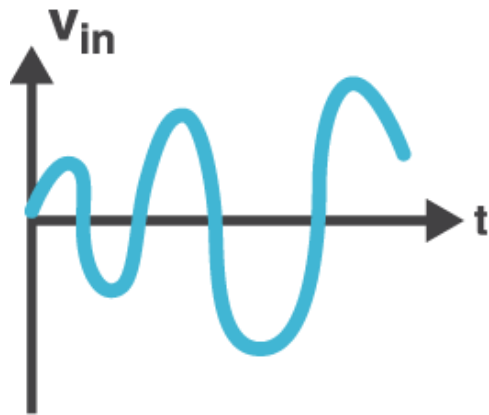
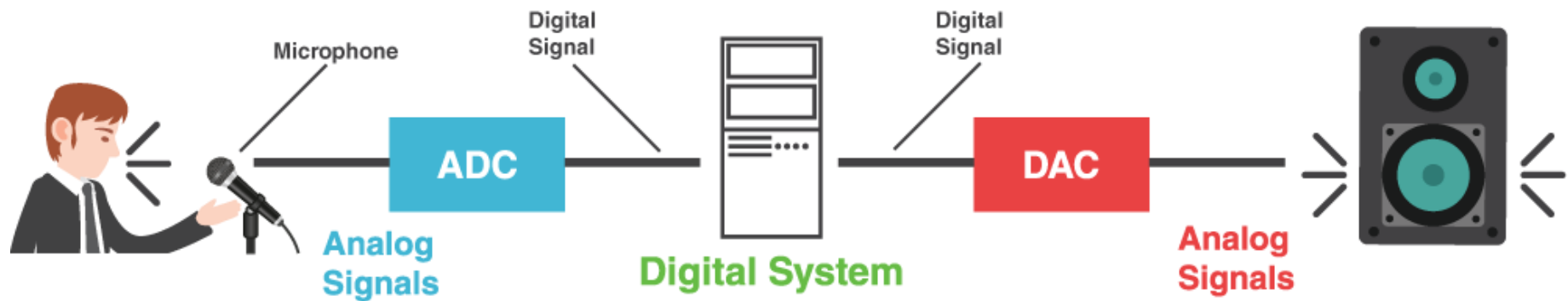


ВИДОВЕ ЕЛЕКТРИЧЕСКИ СИГНАЛИ

- ❖ **Аналогов сигнал** – има безкраен брой нива (състояния).
- ❖ **Цифров сигнал** – има краен брой състояния (най-често 2):
 - **0** и **1**;
 - **LOW** и **HIGH**;
 - **0V** и **5V (3.3V)**.



ПРИМЕР



ЦИФРОВИ ВХОДОВЕ И ИЗХОДИ

- ❖ Всеки микроконтролер има т.нар. **GPIO** – General-Purpose Input/Output – букв. пинове с общо предназначение.

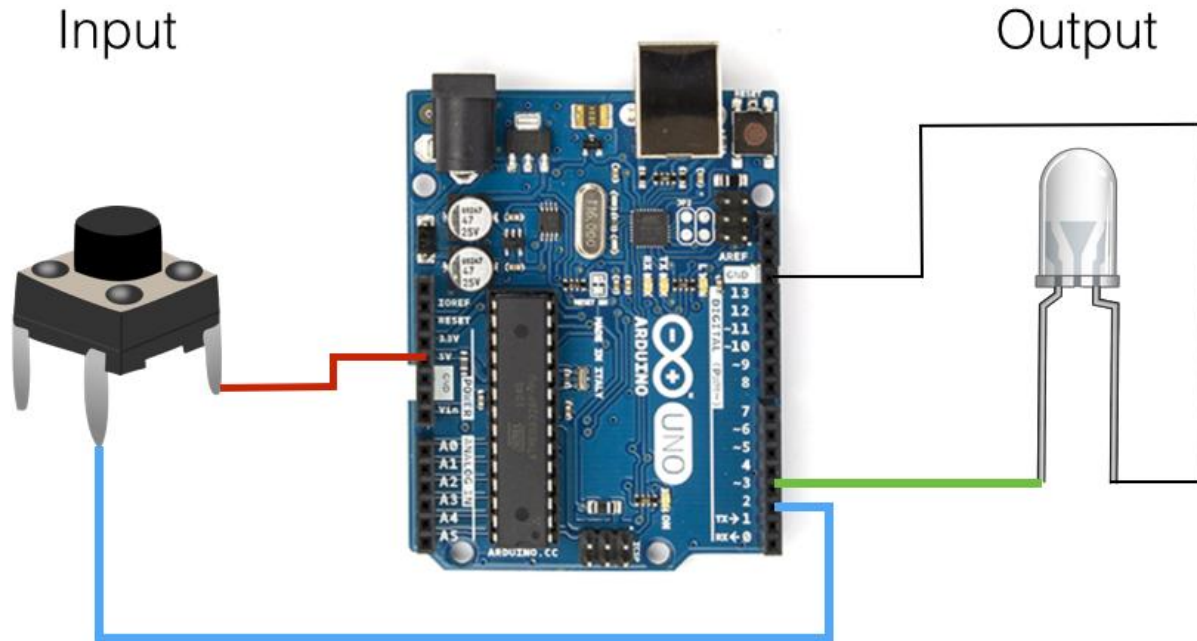
Те могат да бъдат конфигурирани като входове или изходи по желание на програмиста.

- ❖ **Вход** – когато външно устройство(схема) създава ел. сигнал, а микроконтролерът(Ардуино) се опитва да прочете неговото състояние.
- ❖ **Изход** – когато микроконтролерът задава състоянието на съответния пин.

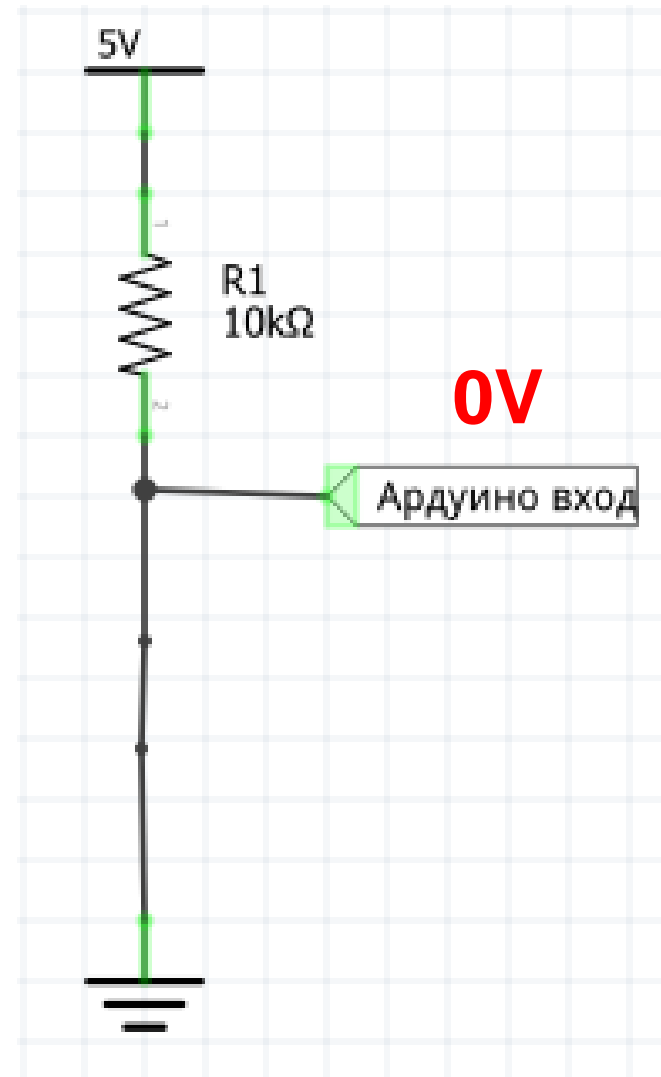
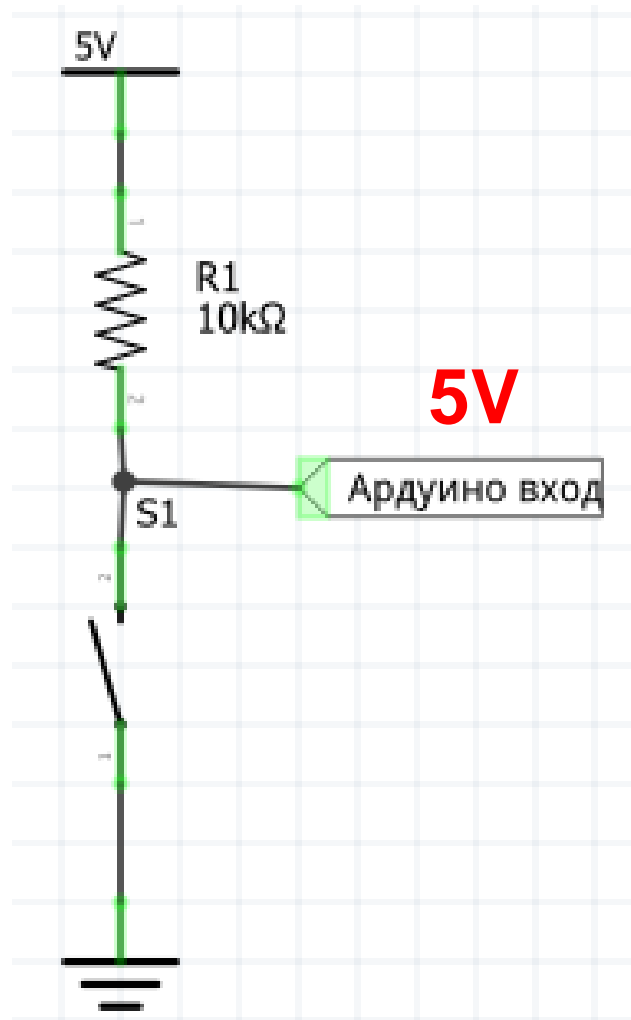


ВХОДОВЕ И ИЗХОДИ

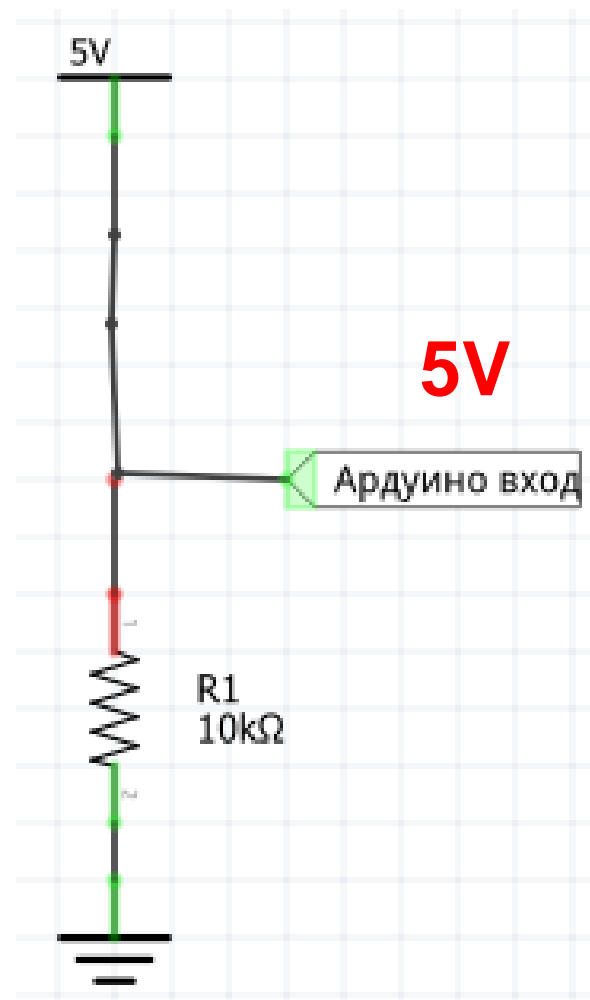
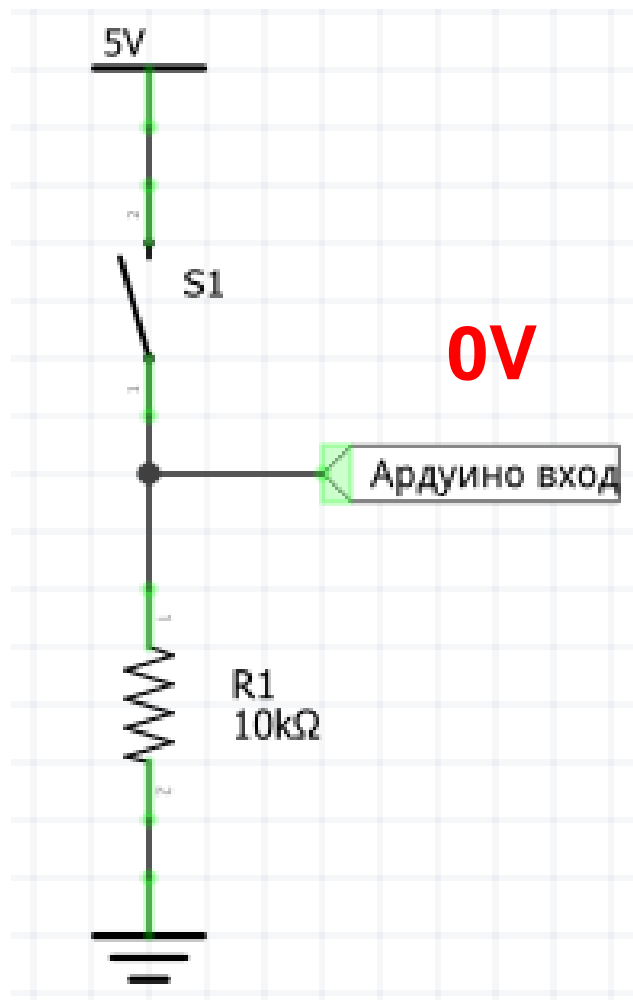
So!



PULL-UP РЕЗИСТОР



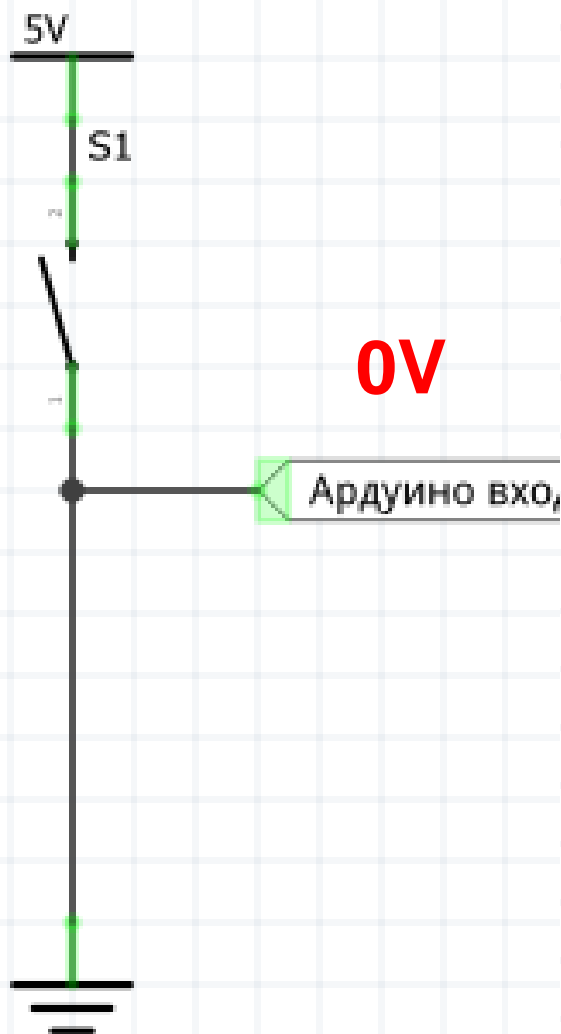
PULL-DOWN РЕЗИСТОР



БЕЗ РЕЗИСТОР – НЕ !!!

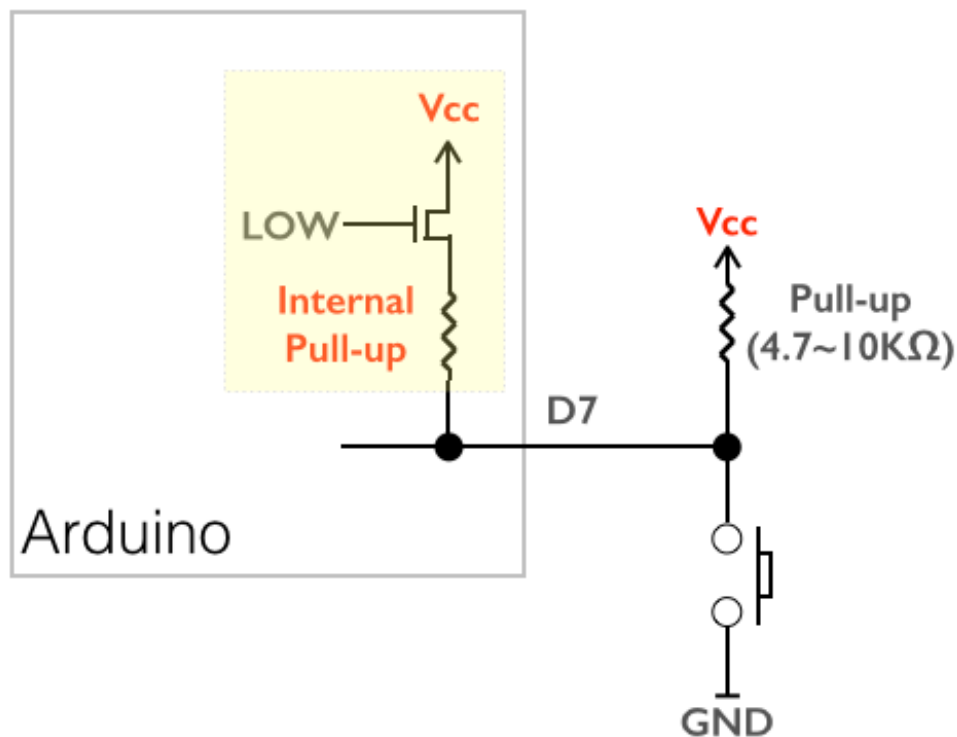


КЪСО СЪЕДИНЕНИЕ КОГАТО НАТИСНЕМ БУТОНА !!!

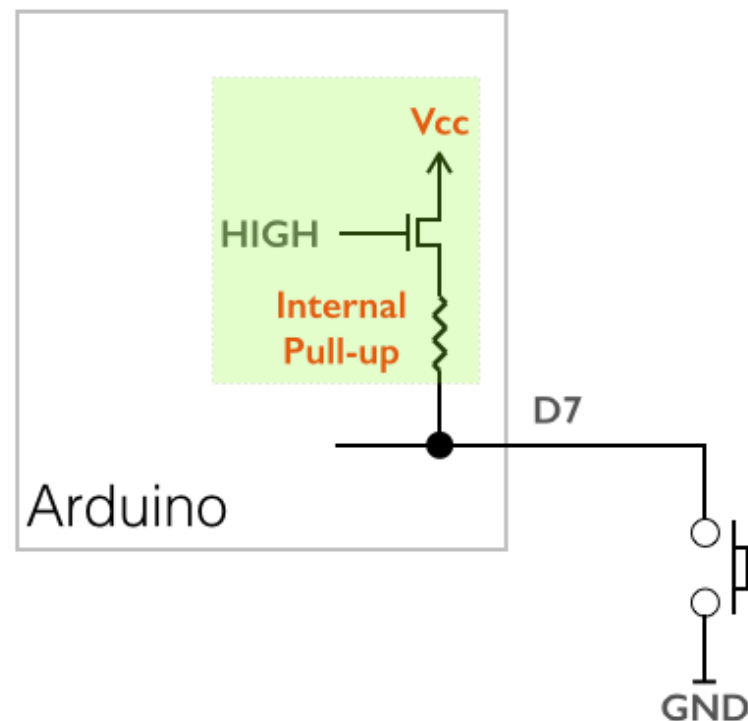


ВГРАДЕНИ PULL-UP РЕЗИСТОРИ

С изключен вътрешен pull-up резистор



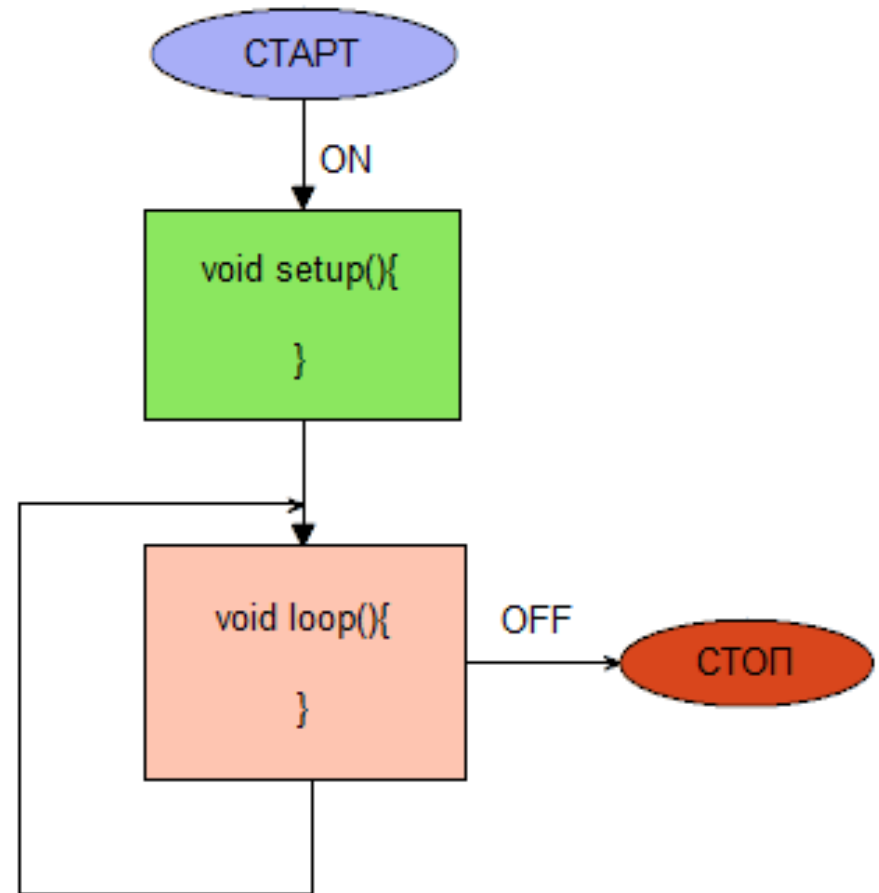
С включен вътрешен pull-up резистор



СТРУКТУРА НА ПРОГРАМАТА

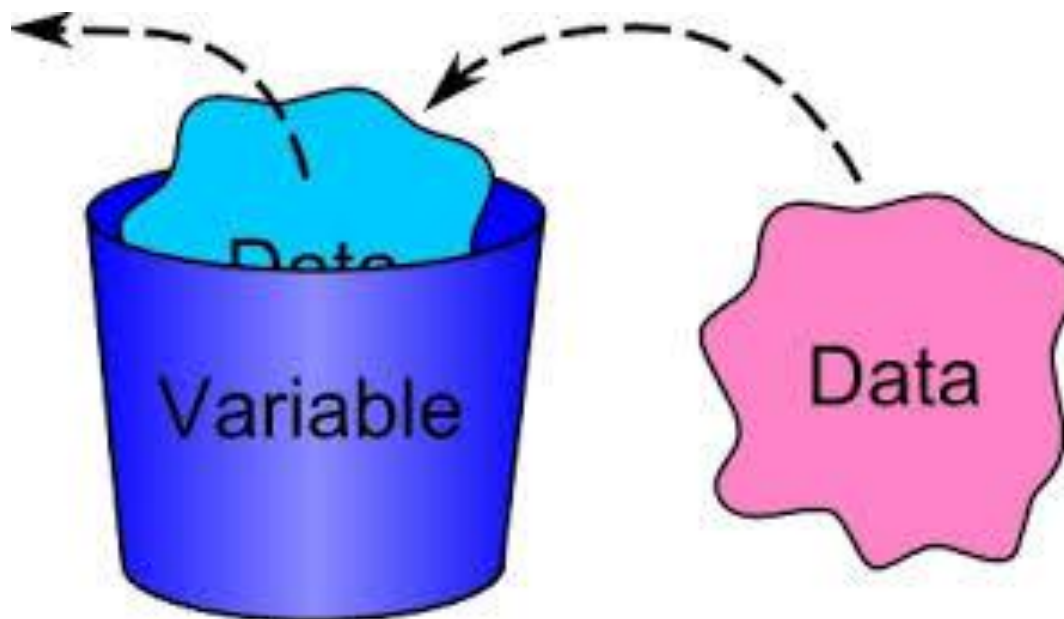
```
void setup() {  
    // изпълнява се веднъж  
}
```

```
void loop() {  
    // безкраен цикъл  
}
```



ПРОМЕНЛИВИ И ТИПОВЕ ДАННИ

- ❖ **Променливата (variable)** представлява място заделено в оперативната памет (RAM) на микроконтролера, която служи за съхранение на определен тип данни (data): цели числа, десетични числа, символи, текст, логическо състояние...



ОСНОВНИ ТИПОВЕ ДАННИ В АРДУИНО

ТИП ДАННИ	РАЗМЕР (БИТОВЕ)	ОПИСАНИЕ
boolean	8	лог. състояние true / false
byte	8	цели числа от 0-255
char	8	символ, цели числа от -127 – 128
Unsigned char	8	като byte
word	16	цели числа от 0-65535
<u>int</u>	16	цели числа от -32768 to 32767
unsigned int	16	като word
long	32	цели числа от -2,147,483,648 - 2,147,483,647
unsigned long	32	цели числа от 0 - 4,294,967,295
float	32	десетични числа от -3.4028235E38 to 3.4028235E38
String	-	текстова информация

ДЕКЛАРИРАНЕ НА ПРОМЕНЛИВИ

СТРУКТУРА:

<тип данни> <идентификатор> [= <инициализация>]

ПРИМЕР:

int a = 10;

int b;

.....

b = 10;

boolean isOpen;

isOpen = true;

ОБЛАСТ НА ВИДИМОСТ

❖ Локални и глобални променливи

```
int LEDpin = 13;
int ButtonPin = 2;

void setup() {
    pinMode(LEDpin, OUTPUT);
    pinMode(ButtonPin, INPUT);
}

void loop() {
    int buttonValue = digitalRead(ButtonPin);
    digitalWrite(LEDpin, buttonValue);
}
```


КОНСТАНТЫ В АРДУИНО

СТРУКТУРА:

const <тип данни> <идентификатор> = <инициализация>

ПРИМЕР:

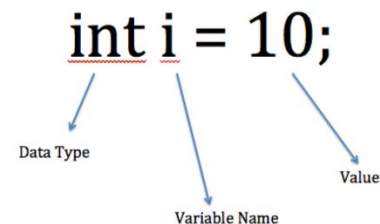
const float PI = 3.14;

const int redLed = 7;

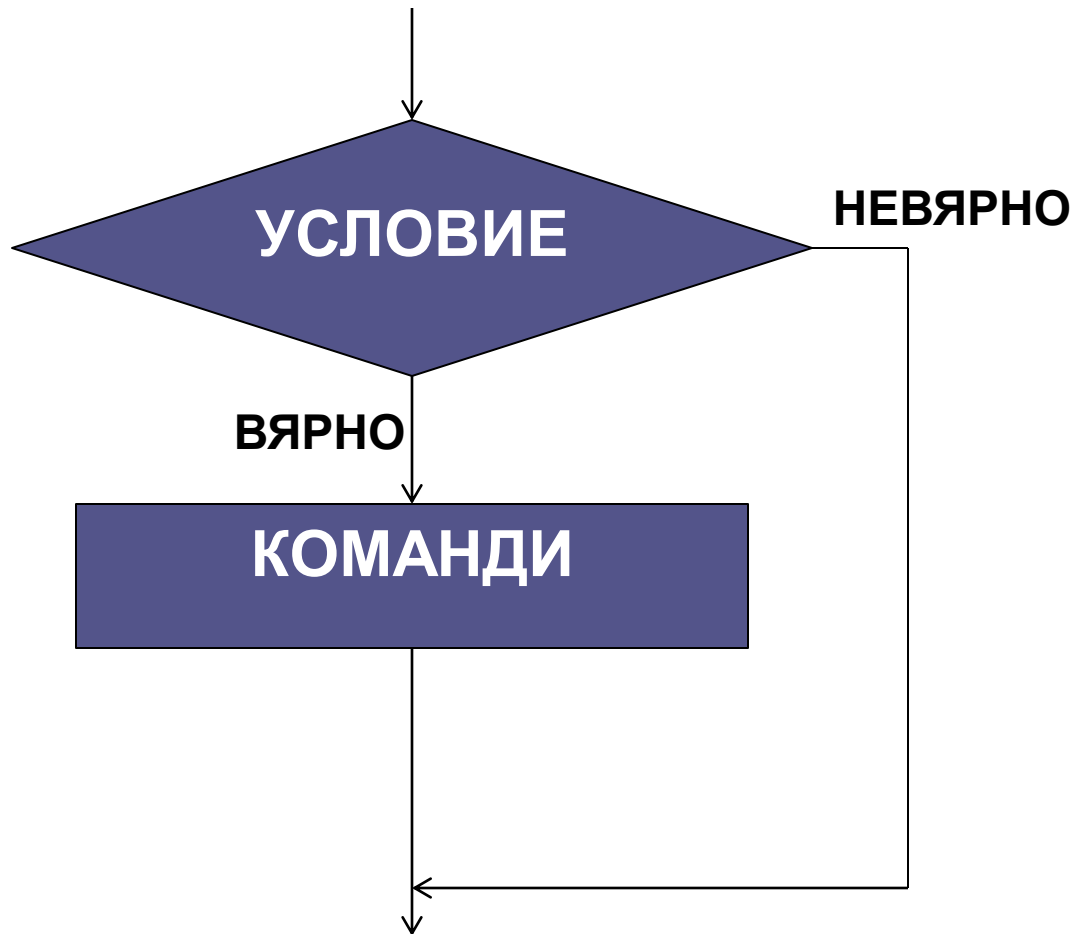
const int buttonPin = 4;

ИДЕНТИФИКАТОРИ

- ❖ Опр: Имена на променливи, масиви, структури, обекти...
- ❖ ! Могат да съдържат само: букви **a-z**, **A-Z**, цифри **0-9** и **_**;
- ❖ ! Не може да започват с цифра;
- ❖ ! Не може да съдържат интервали;
- ❖ ! Не трябва да съвпадат с ключови думи от Ардуино езика: напр. **int**, **setup**...
- ❖ ! Добре е името на променливата да отговаря на това, което съхранява: напр. **age**
- ❖ ! Да бъде кратко;
- ❖ ! Да бъде на английски език;

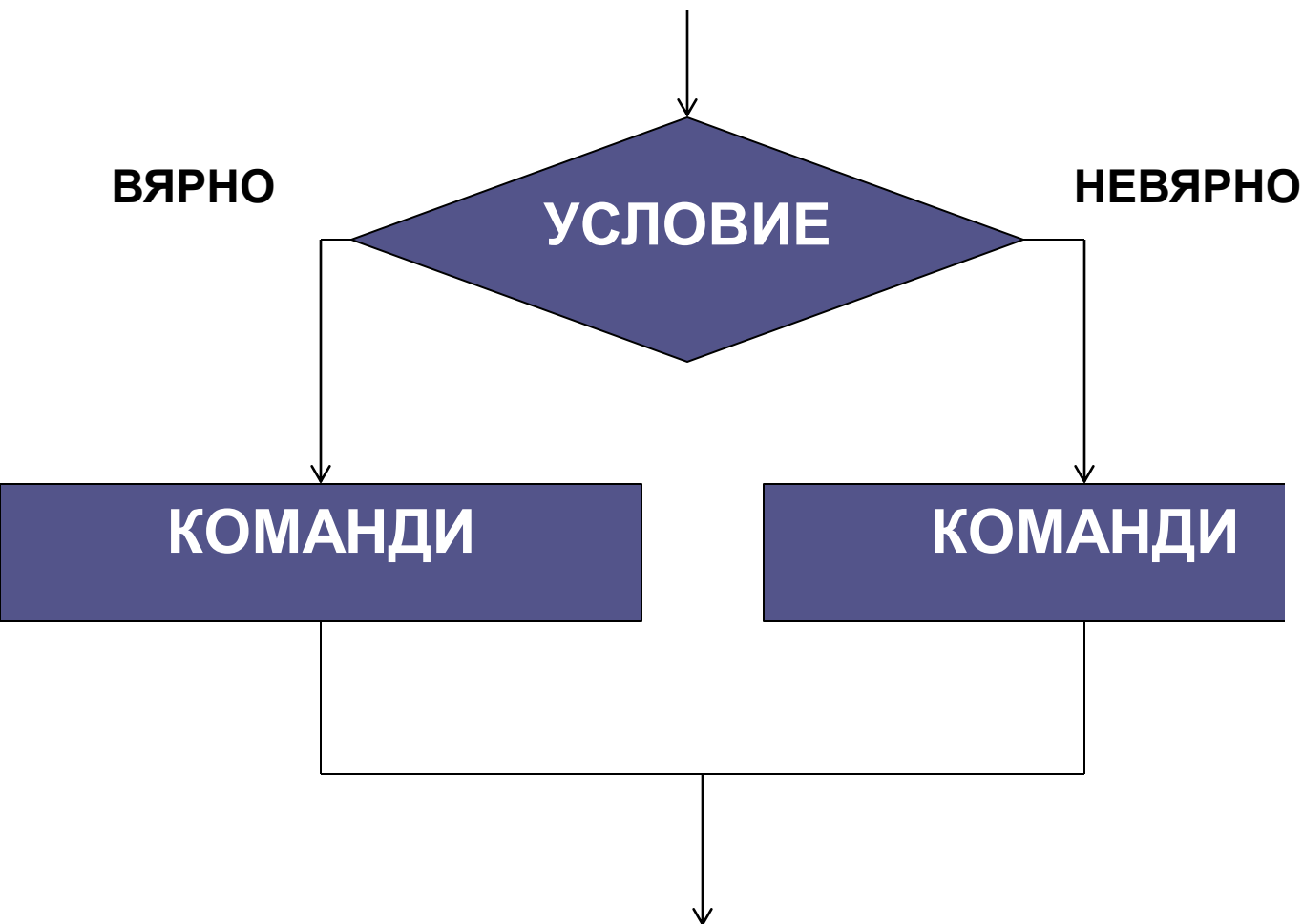


УСЛОВЕН ОПЕРАТОР IF



```
if( "условие" ){  
    "команди 1"  
    "команда 2"  
    "команда 3"  
}
```

УСЛОВЕН ОПЕРАТОР IF-ELSE



```
if( "условие" ){  
    "команди 1"  
    "команда 2"  
    "команда 3"  
}  
else {  
    "команди 4"  
    "команда 5"  
    "команда 6"  
}
```

ОСНОВНИ ФУНКЦИИ (1)

❖ **pinMode**("номер пин", "режим");

Конфигурира посочения пин като:

- цифров изход (**OUTPUT**);
- цифров вход (**INPUT**);
- цифров вход с включени pull-up резистори (**INPUT_PULLUP**);

❖ **delay**("милисекунди");

Микроконтролерът изчаква "милисекунди" време, преди да продължи със следващите команди.

ОСНОВНИ ФУНКЦИИ (2)

❖ **digitalWrite**("номер пин", LOW / HIGH);

Подава на посочения пин 0V (LOW) или 5V (HIGH).

❖ **digitalRead**("милисекунди");

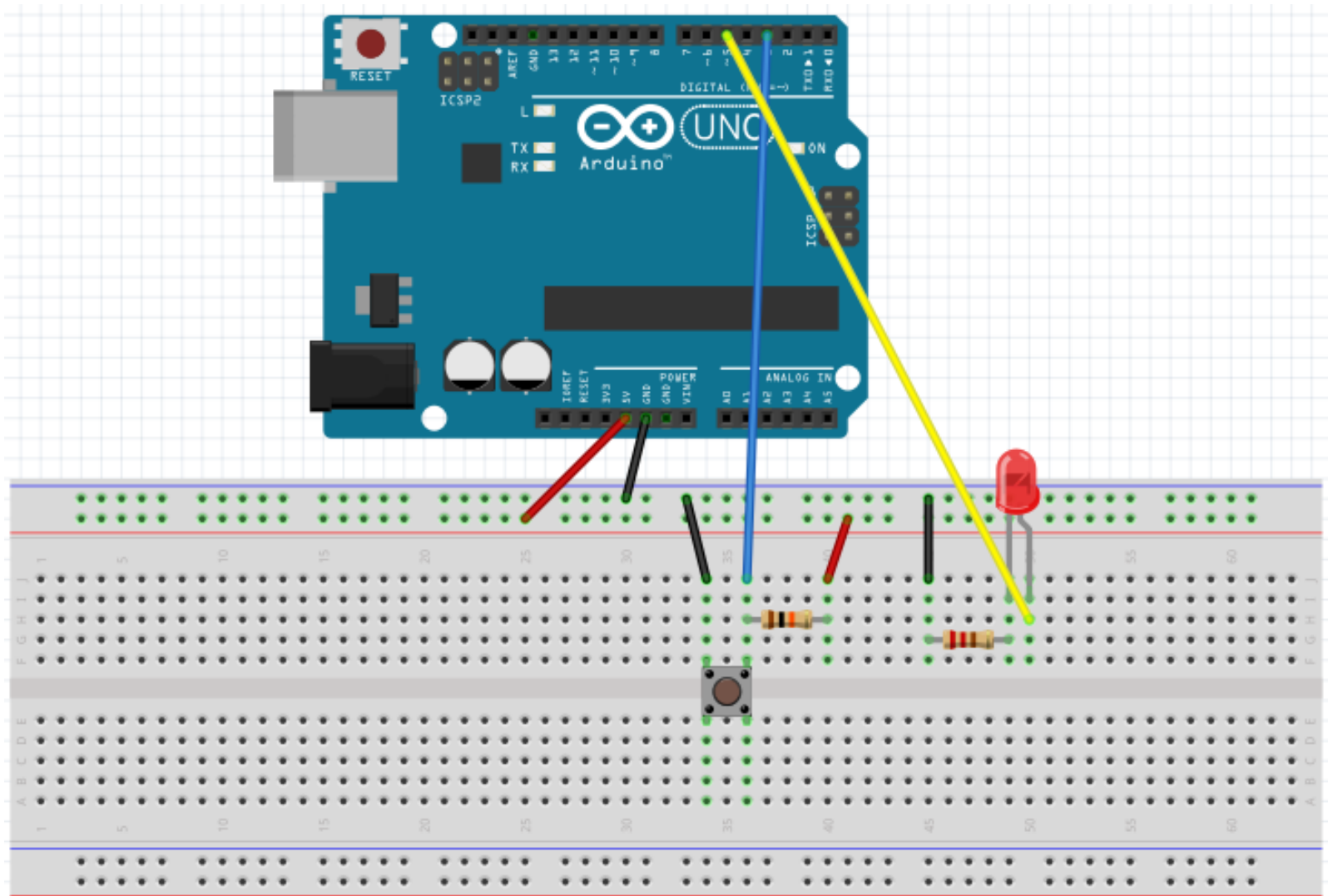
- Връща **0** ако напрежението на цифровия вход е 0V;
- Връща **1** ако напрежението на цифровия вход е 5V;

ЗАДАЧА 2.1

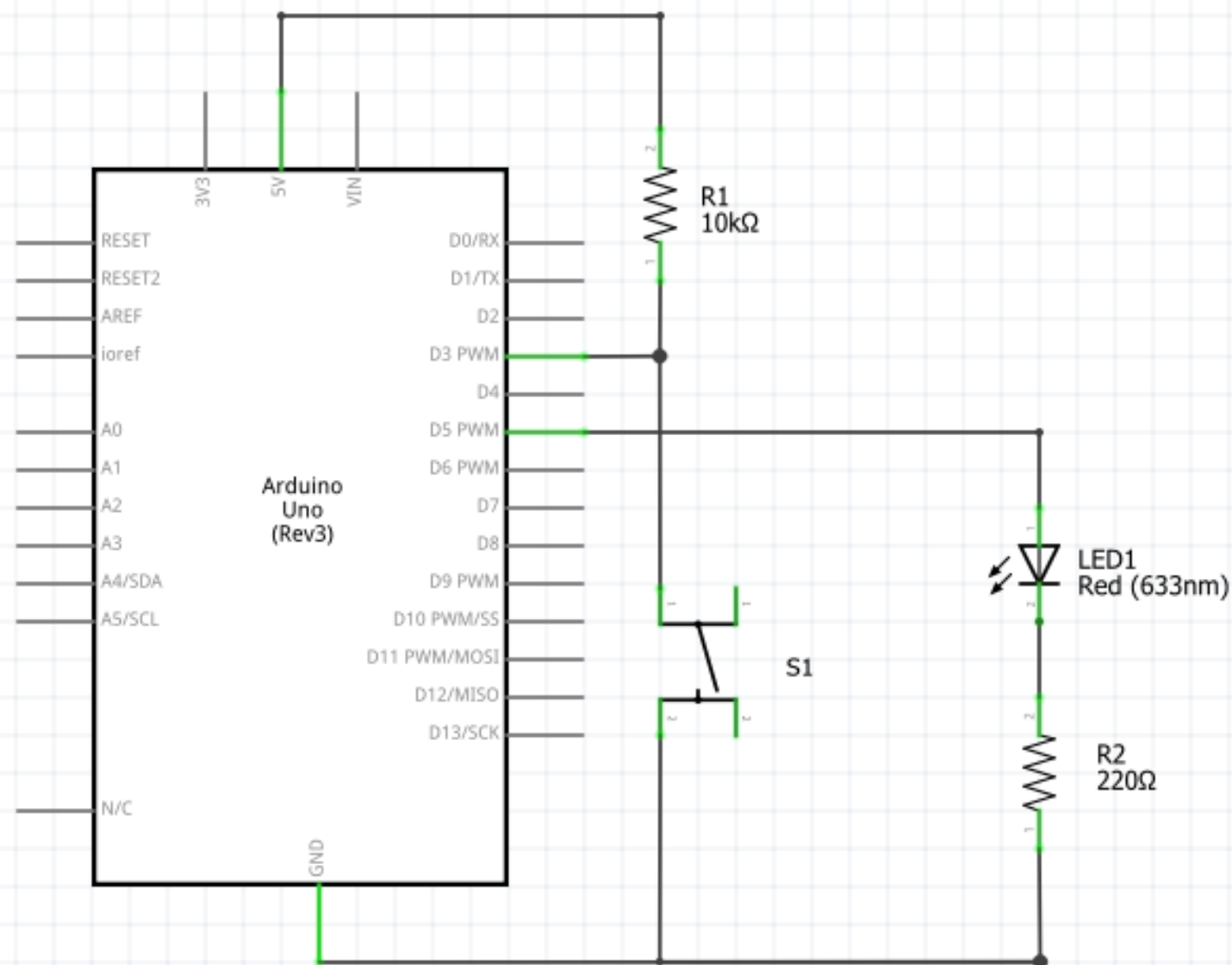
Прочитане на бутон.



ЕЛЕКТРИЧЕСКА ВЕРИГА



ЕЛЕКТРИЧЕСКА СХЕМА



ПРОГРАМА

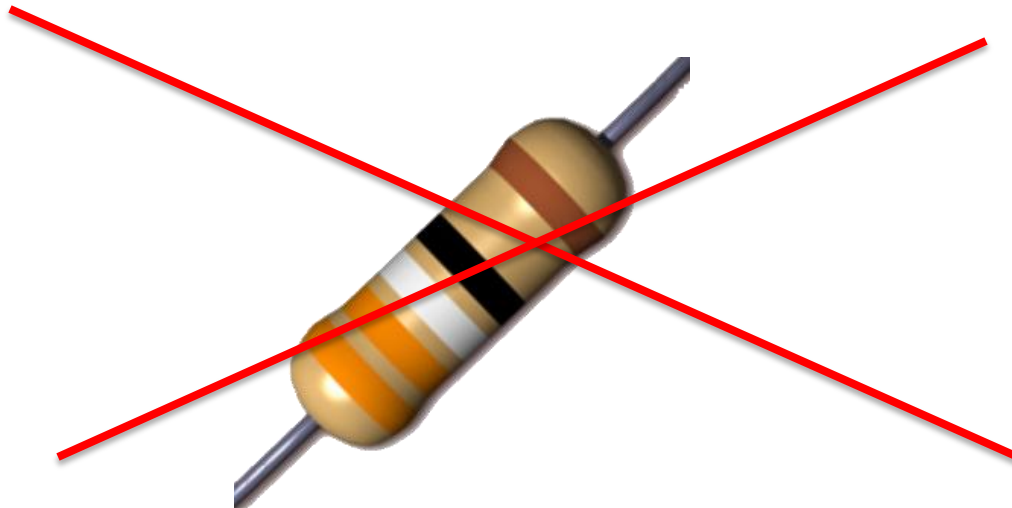
```
const int LED = 5;
const int BUTTON = 3;
int check;

void setup() {
    pinMode(LED, OUTPUT);
    pinMode(BUTTON, INPUT);
}

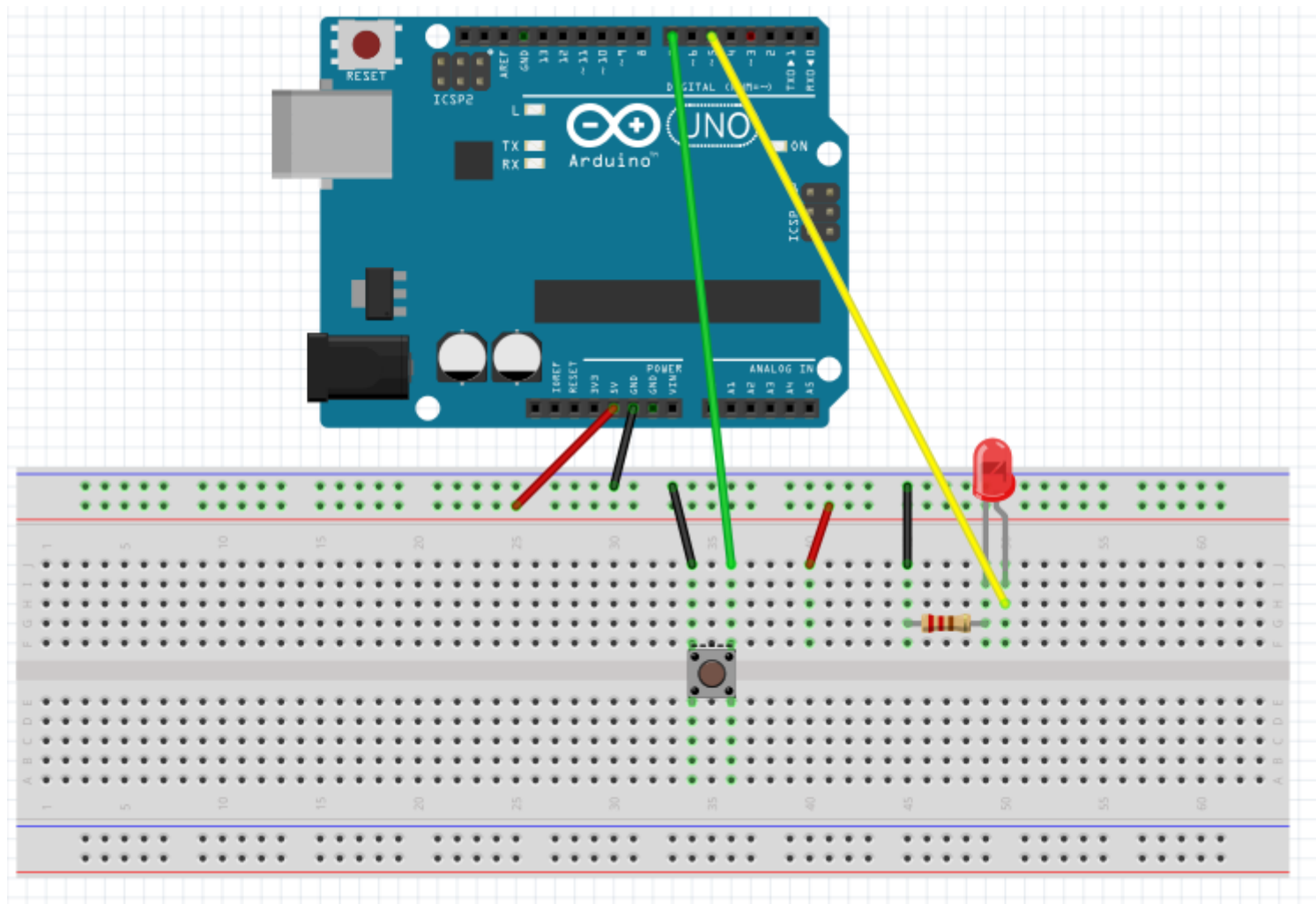
void loop() {
    check = digitalRead(BUTTON);
    if(check == 1) {
        digitalWrite(LED, HIGH);
    } else {
        digitalWrite(LED, LOW);
    }
}
```

ЗАДАЧА 2.2

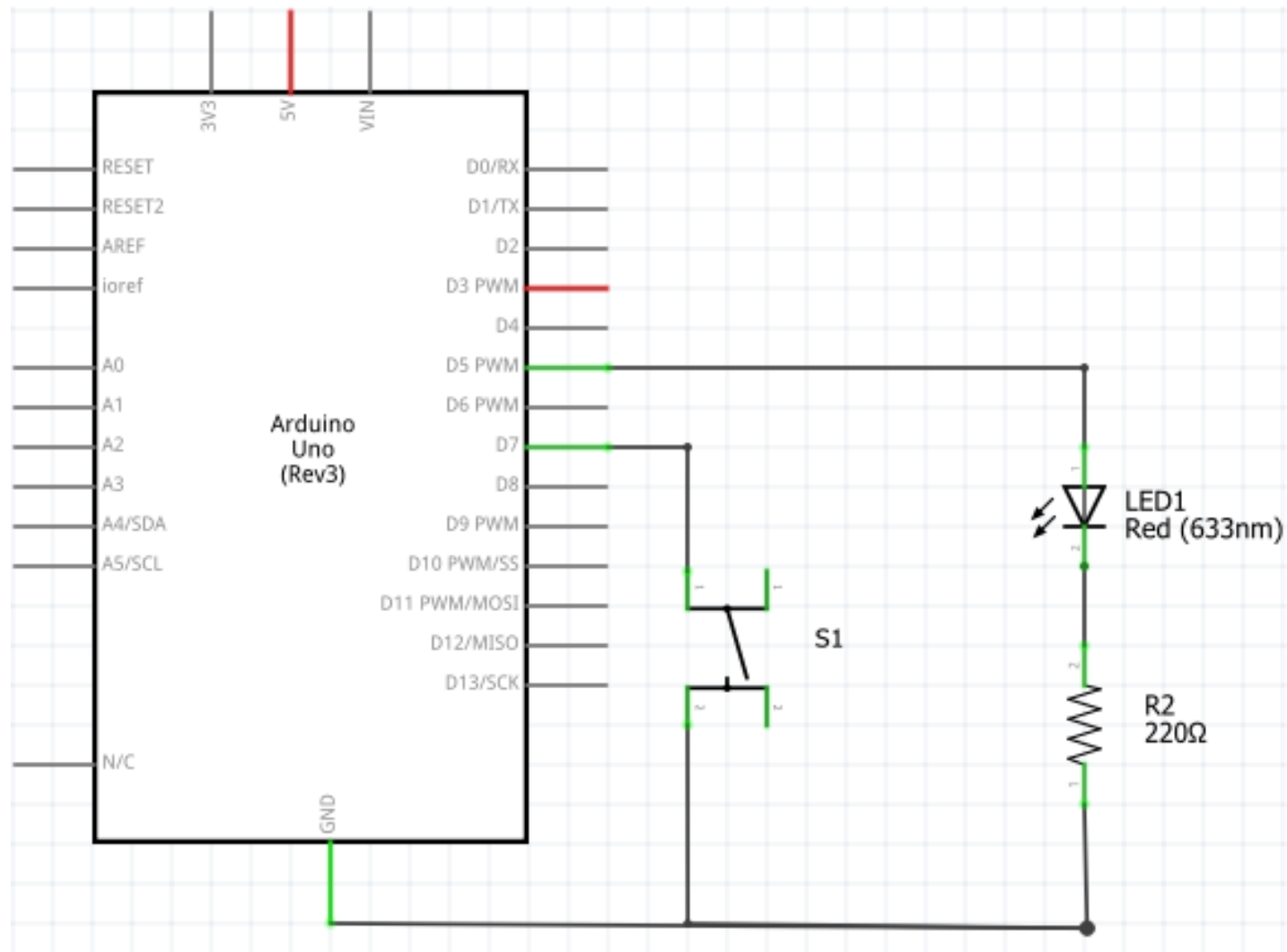
Прочитане на бутон
с вграден pull-up резистор



ЕЛЕКТРИЧЕСКА ВЕРИГА



ЕЛЕКТРИЧЕСКА СХЕМА



ПРОГРАМА

```
const int LED = 3;
const int BUTTON = 7;

void setup() {
    pinMode(LED, OUTPUT);
    pinMode(BUTTON, INPUT_PULLUP);
}

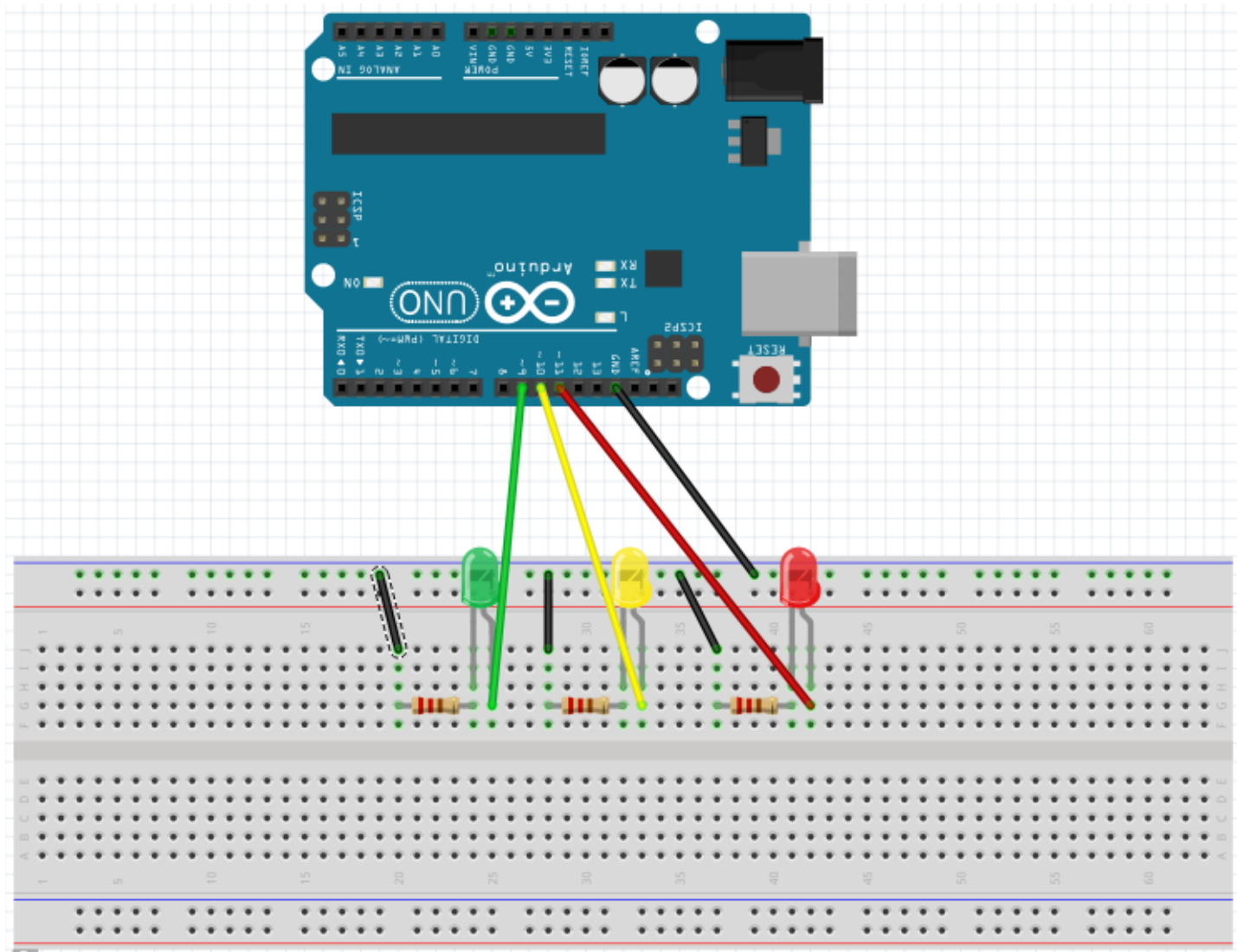
void loop() {
    if(digitalRead(BUTTON)) {
        digitalWrite(LED, HIGH);
    } else {
        digitalWrite(LED, LOW);
    }
}
```

ЗАДАЧА 2.3

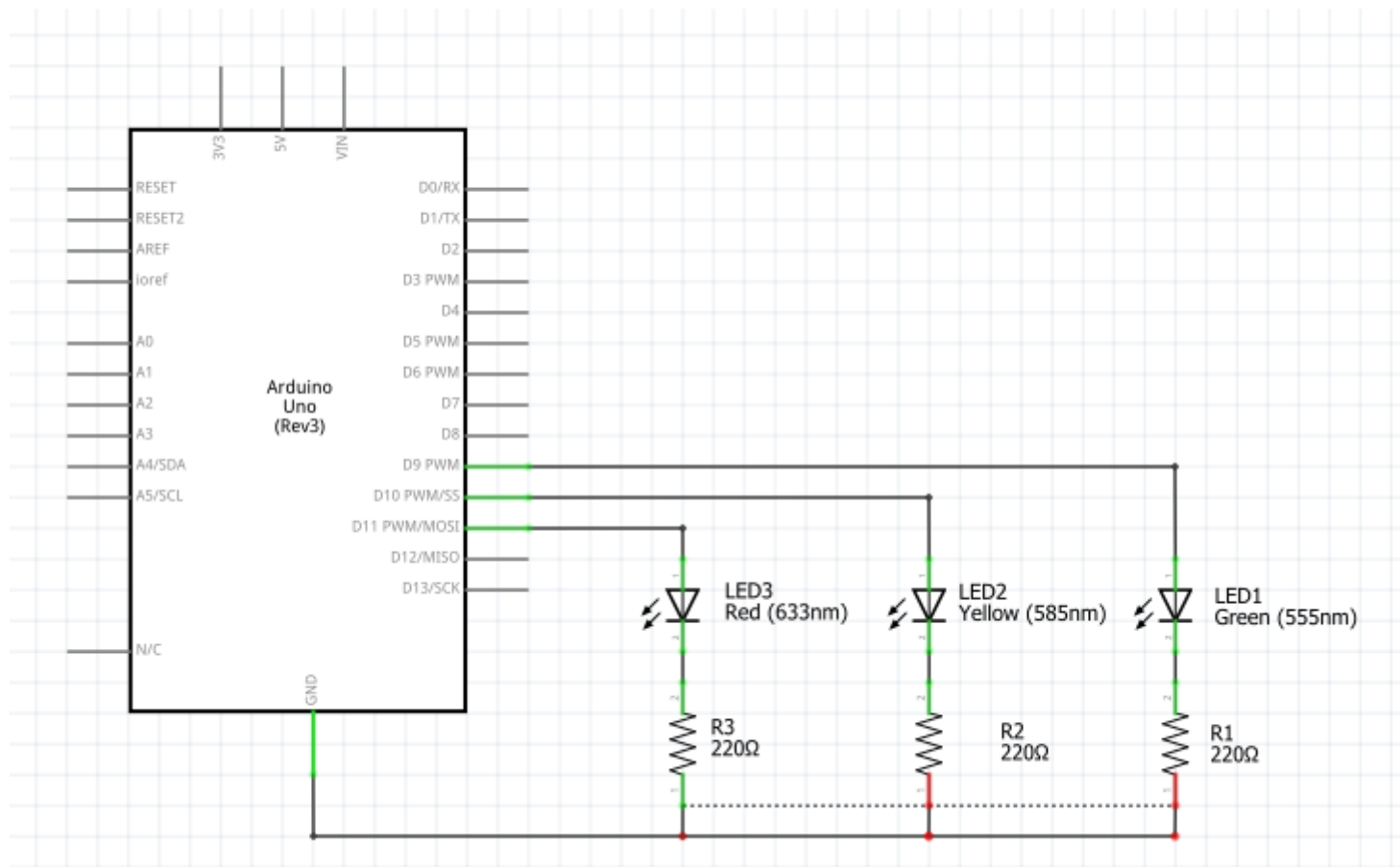
Проектиране на светофарна система



ЕЛЕКТРИЧЕСКА ВЕРИГА



ЕЛЕКТРИЧЕСКА СХЕМА



ПРОГРАМА

(1)

```
const int RED = 11;
const int YELLOW = 10;
const int GREEN = 9;

void setup() {
  pinMode(RED, OUTPUT);
  pinMode(YELLOW, OUTPUT);
  pinMode(GREEN, OUTPUT);
}
```

(2)

```
void loop() {
  digitalWrite(RED, HIGH);
  digitalWrite(YELLOW, LOW);
  digitalWrite(GREEN, LOW);
  delay(3000);
  digitalWrite(RED, HIGH);
  digitalWrite(YELLOW, HIGH);
  digitalWrite(GREEN, LOW);
  delay(1000);
  digitalWrite(RED, LOW);
  digitalWrite(YELLOW, LOW);
  digitalWrite(GREEN, HIGH);
  delay(3000);
  digitalWrite(RED, LOW);
  digitalWrite(YELLOW, HIGH);
  digitalWrite(GREEN, LOW);
  delay(1000);
}
```



БЛАГОДАРЯ ВИ ЗА ВНИМАНИЕТО!