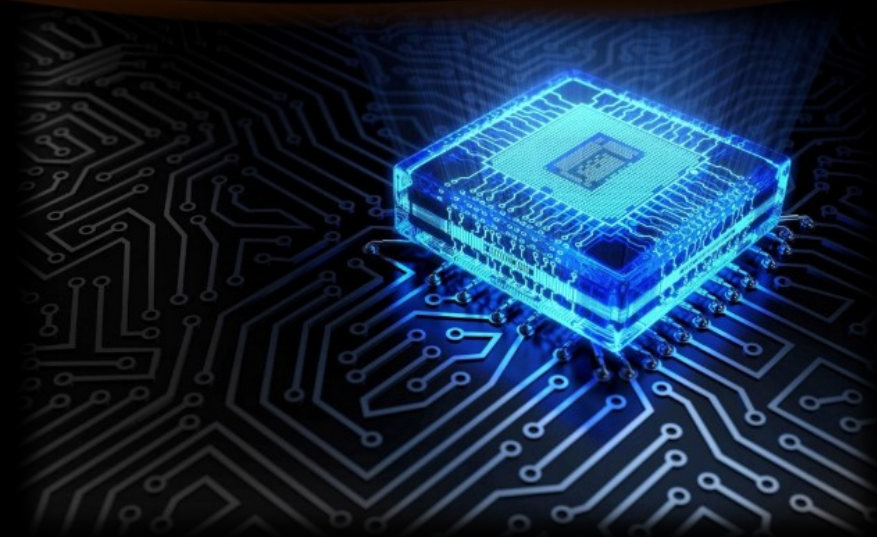
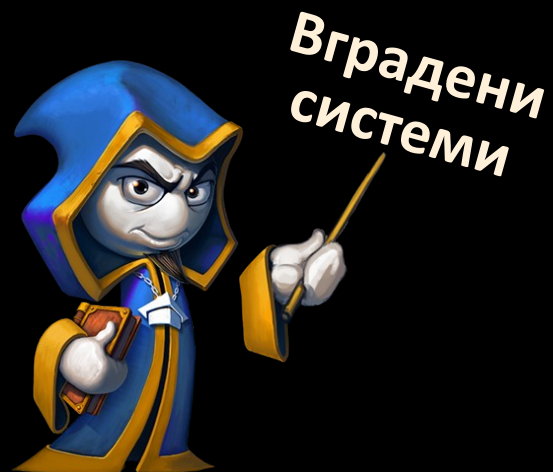


# Светване на LED



Инж. Венцеслав Кочанов

## Електрически сигнали

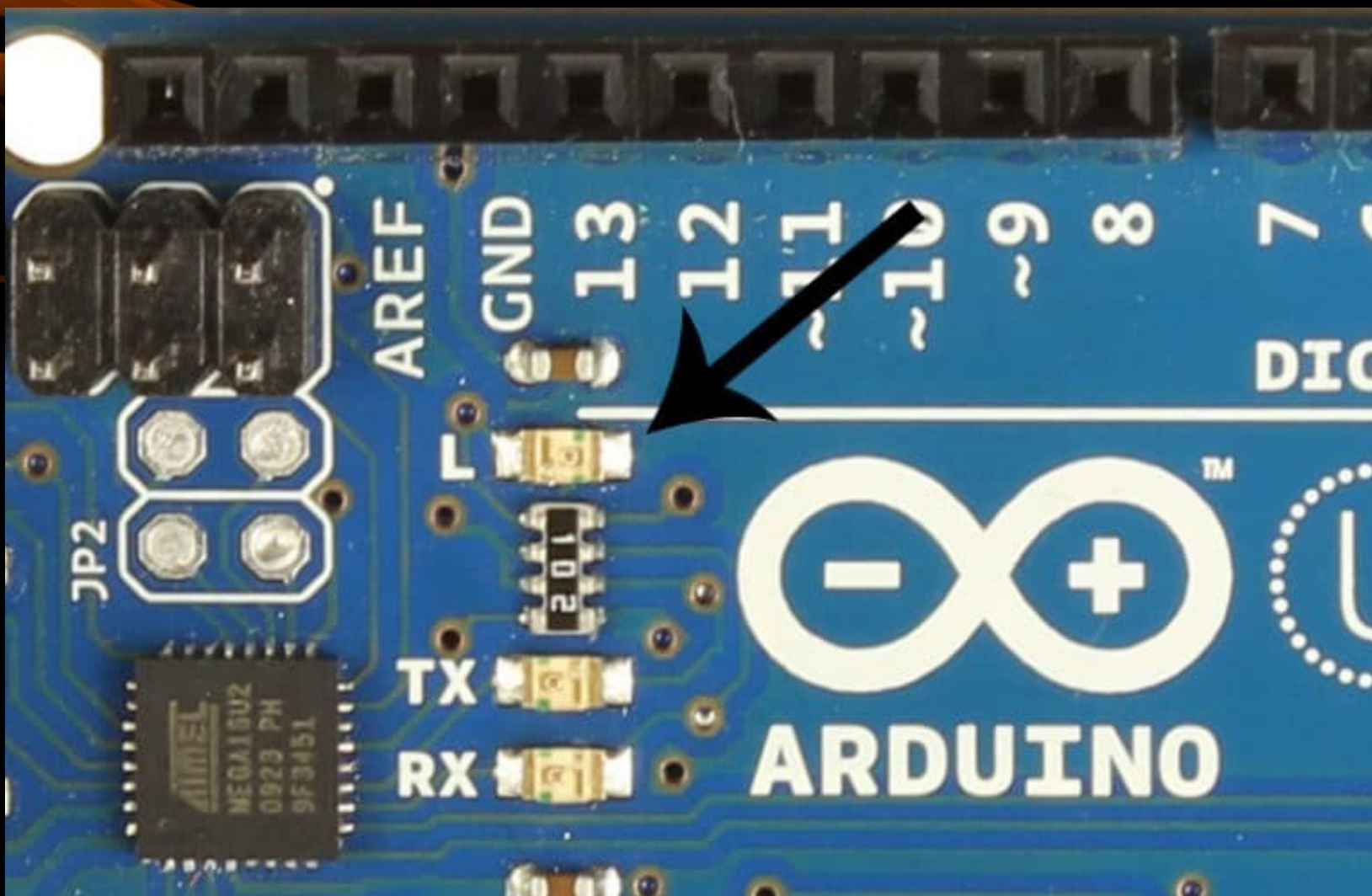
Arduino комуникира с модули и сензори чрез включване и изключване на електрически ток. Това е много подобно на единицата и нулата в двоичния код. Когато токът е включен, той е известен като „ВИСОК сигнал“. Това е сравнимо с „единицата“ в двоичния код. Когато токът е изключен, това е "НИСЪК сигнал", който е подобен на нулата в двоичния код. Продължителността на времето, през което токът остава включен или изключен, може да се промени от микросекунда до много минути.

## Управление на светодиода на Arduino

За да включи светодиод, Arduino трябва да изпрати сигнал HIGH към един от своите щифтове. За да изключите светодиода, той трябва да изпрати LOW сигнал към щифта. Можете да накарате светодиода да мига, като промените продължителността на състоянията HIGH и LOW.

Arduino има вграден светодиод за повърхностен монтаж, който е твърдо свързан към цифров щифт 13. Това е този с „L“ до него:





За да накараме този светодиод да мига, трябва да качиме програмата „Blink“ на вашия Arduino:

```
void setup() {  
  pinMode(13, OUTPUT);  
}  
void loop() {  
  digitalWrite(13, HIGH);  
  delay(1000);  
  digitalWrite(13, LOW);  
  delay(1000);  
}
```

Сега светодиодът трябва да мига и да мига с честота от 1000 милисекунди (1000 милисекунди = 1 секунда).

Функцията `delay()` на линия 6 казва на Arduino да задържи сигнала HIGH на пин 13 за 1000 ms. Функцията `delay()` на линия 8 й казва да задържи LOW сигнала на пин 13 за 1000 ms. Можете да промените скоростта на мигане, като промените числото в скобите на функциите `delay()`.

## Как да управляваме външен светодиод

Външен светодиод или друг захранващ модул може да се управлява по подобен начин.

Светодиодите трябва да имат резистор, поставен последователно (в линия) с него. В противен случай неограниченият ток бързо ще изгори светодиода. Резисторът може да бъде всяка стойност между 100 ома и около 10К ома. Резисторите с по-ниска стойност ще позволят протичането на повече ток, което прави светодиода по-ярък. Резисторите с по-висока стойност ще ограничат текущия поток, което прави светодиода по-слаб.



Тъй като повечето светодиоди имат полярност, означава, че трябва да бъдат свързани по правилния начин. Обикновено най-късият проводник на светодиода се свързва със заземената страна или минуса.

Ако свържем светодиода към щифт 13, както е показано на изображението по-долу, можем да използваме същия код, който използвахме по-горе, за да включиме и изключиме светодиода.

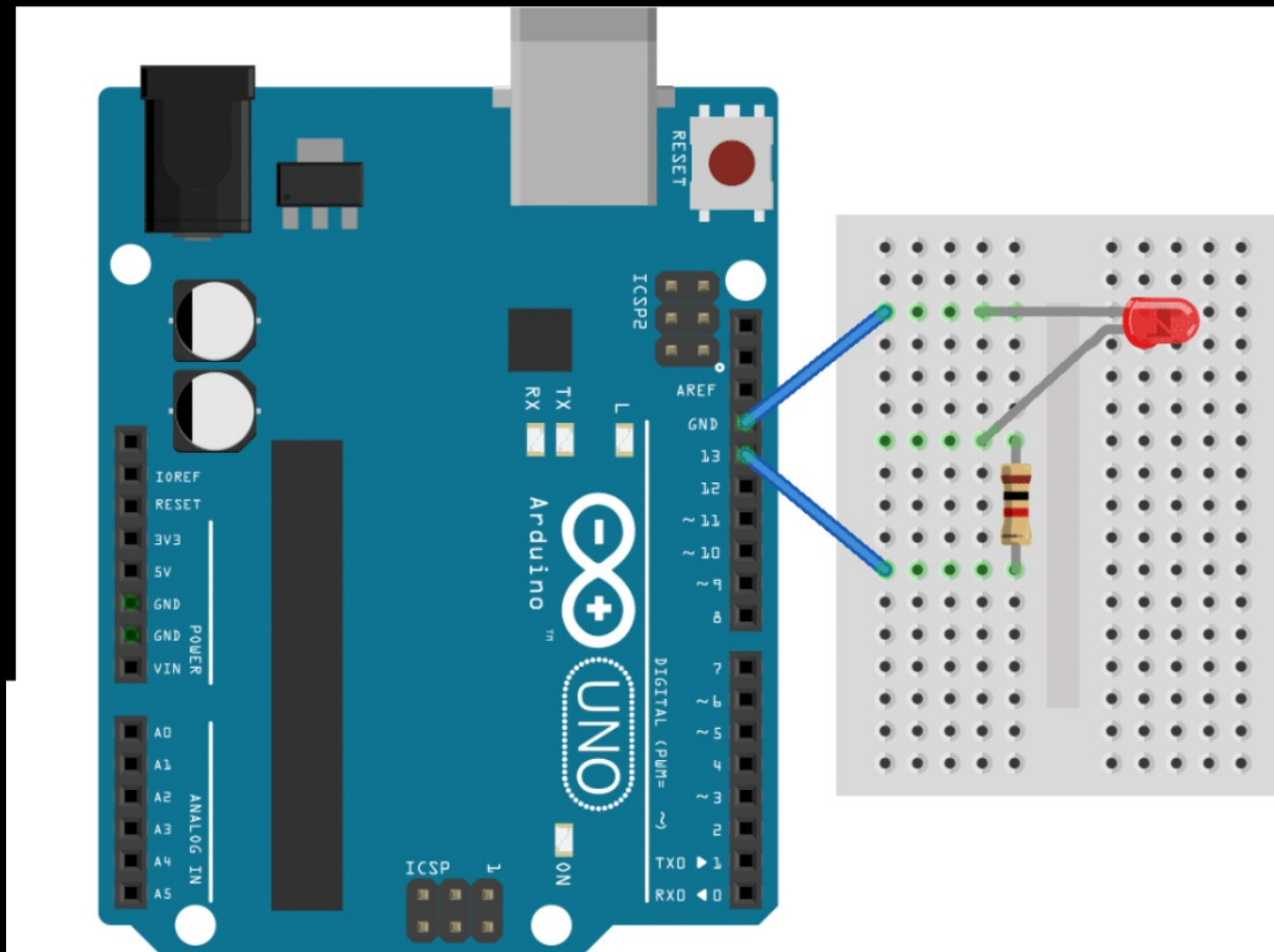
Ако искаме да използваме различен щифт за захранване на светодиода, лесно можем да го сменим. Например, ако искате да използвате пин 8 вместо пин 13, премесваме сигналния проводник от пин 13 към пин 8, като променим `pinMode(13, OUTPUT);` на `pinMode(8, OUTPUT);`

След това ще трябва да промените кода, който казва на Arduino кои щифтове ще получат HIGH и LOW изходни сигнали. Това се прави навсякъде, където има функция `digitalWrite()`:

```
digitalWrite(13, HIGH);  
digitalWrite(13, LOW);
```

се променя на

```
digitalWrite(8, HIGH);  
digitalWrite(8, LOW);
```





## Управление на множество светодиоди заедно

Можете да управлявате толкова светодиоди, колкото искате, стига да имате достатъчно налични щифтове. Нека накараме външния светодиод да мига заедно с вградения светодиод, за да демонстрираме. Всичко, което трябва да направим, е да дублираме кода за пин 8 и да променим номерата на пиновете на пин 13.

Ето един пример за това:

```
void setup() {  
  pinMode(8, OUTPUT);  
  pinMode(13, OUTPUT);  
}  
void loop() {  
  digitalWrite(8, HIGH);  
  delay(1000);  
  digitalWrite(13, HIGH);  
  delay(1000);  
  digitalWrite(8, LOW);  
  delay(1000);  
  digitalWrite(13, LOW);  
  delay(1000);  
}
```

Вградени системи



Въпроси?

