

Arduino-kompendium — MEKMEK01

Innehåll

1 Teori	2
1.1 Vad är en pin	2
1.2 Utgång eller ingång?	2
1.3 Digital eller analog?	2
2 Programmering	2
2.1 <code>pinMode</code>	2
2.2 <code>digitalWrite</code>	3
2.3 <code>digitalRead</code>	3
2.4 <code>delay</code>	3
2.5 Seriell kommunikation	3
2.6 <code>millis</code>	3

1 Teori

Här förklaras lite grundläggande koncept som behövs för att förstå Arduino.

1.1 Vad är en pin

En pin är en fysisk kontakt på en mikrokontroller som kan användas för att koppla in och ut signaler. En pin kan vara en ingång eller en utgång (se 1.2). De pins som heter D0-D13 kan användas för digitala signaler, medan A0-A5 kan användas för analoga signaler (se 1.3).

1.2 Utgång eller ingång?

Du som programmerare väljer om en pin ska vara en ingång eller utgång genom `pinMode` (se 2.1). En ingång kan användas för att läsa av en signal, till exempel från en knapp eller en sensor. En utgång kan användas för att skicka en signal, till exempel till en lampa eller en motor.

1.3 Digital eller analog?

En digital signal kan bara ha två tillstånd: hög eller låg. En analog signal kan ha ett kontinuerligt värde mellan hög och låg. En analog signal kan till exempel användas för att styra en motor så att den snurrar med olika hastighet, eller en lampa så att den lyser med olika styrka.

2 Programmering

2.1 `pinMode`

Används för att bestämma om en `pin` ska vara en utgång eller ingång (1.2)

Exempelanvändning:

```
pinMode(4, OUTPUT);
```

Kommer att välja pin D4 som en utgång.

2.2 digitalWrite

2.3 digitalRead

2.4 delay

2.5 Seriell kommunikation

2.6 millis

Används för att mäta tid. Vid anrop returneras antal millisekunder sedan Arduino-enheten startades. En begränsning är att det inte går att mäta längre tid än vad datatypen `unsigned long` kan rymma, vilket är ungefär 50 dagar.

Exempelanvändning:

För att blinka en lampa varje sekund, utan att använda `delay`:

```
void loop() {  
    unsigned long currentTime = millis(); // Hämta aktuell tid i millisekunder  
    // Kontrollera om tillräcklig tid har passerat sedan senaste blinkningen  
    if (currentTime - lastTime >= interval) {  
        digitalWrite(ledPin, !digitalRead(ledPin)); // Byt tillstånd på lampan  
        lastTime = currentTime; // Uppdatera senaste tiden  
    }  
}
```