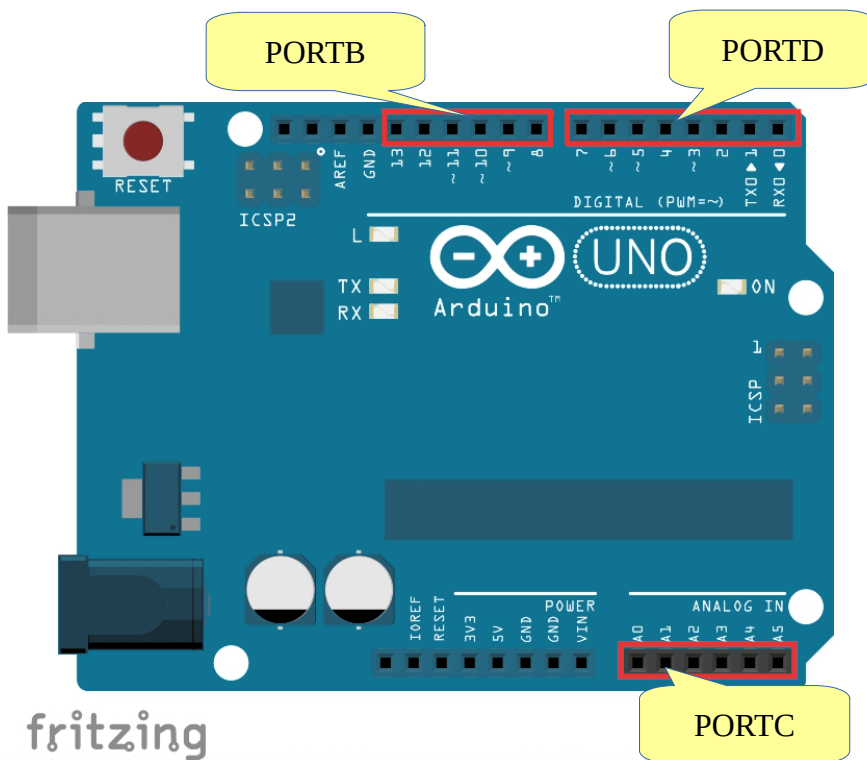


## LEDs schalten mit PORT/DDR

Die LEDs sollen als Lauflicht und in verschiedenen Kombination leuchten:

- ➔ Lauflicht hin und zurück
- ➔ alle LEDs
- ➔ die LEDs 7 5 3
- ➔ die beiden inneren LEDs
- ➔ die beiden äußeren LEDs
- ➔ alle außer den äußeren LEDs
- ➔ nur die mittlere

Die digitalen und die analogen Pins des Arduinos sind in drei Gruppen aufgeteilt:



Port D	D7	D6	D5	D4	D3	D2	D1	D0
Port B	0	0	D13	D12	D11	D10	D9	D8
Port C	0	0	A5	A4	A3	A2	A1	A0

Funktionen wie `digitalWrite()` und `digitalRead()` sind relativ langsam in der Ausführung und manchmal werden Programme mit vielen Ein- oder Ausgängen komplex und schwer überschaubar. Deshalb ist es oft übersichtlicher die Pins mithilfe der Ports zu definieren und zu schalten.

Es existieren drei sogenannte Register:

- DDR → Pins als INPUT = 0 oder als OUTPUT = 1 definieren
- PORT → Pins als HIGH = 1 oder als LOW = 0 festlegen
- PIN → gibt den Zustand der Pins an, die im DDR-Register auf Input gesetzt wurden

Das Register wird jeweils mit dem Namen des Ports (D, B oder C) ergänzt.  
Beim Zugriff auf die Ports wird jeweils ein Bit gesetzt. Sein Wert ist entweder 1 = an, oder 0 = aus.

### Beispiele:

Digitale Pins 7, 6, 5, 4 und 3 als OUTPUT setzen

```
DDRD = B11111000;
```

Digitale Pins 7, 6, 5, 4 und 3 auf HIGH setzen:

```
PORTD = B11111000;
```

Taster an Pin 13 als INPUT setzen

```
DDRB = B00100000;
```

PULL\_UP-Widerstand an Pin 13 einschalten

```
PORTB = B00100000;
```

### Bits nach rechts schieben:

1	1	1	1	1	0	0	0	<code>PORTD = PORTD &gt;&gt; 1;</code>
0	1	1	1	1	1	0	0	<code>PORTD = PORTD &gt;&gt; 1;</code>
0	0	1	1	1	1	1	0	<code>PORTD = PORTD &gt;&gt; 1;</code>
0	0	0	1	1	1	1	1	<code>PORTD = PORTD &gt;&gt; 1;</code>
0	0	0	0	1	1	1	1	<code>PORTD = PORTD &gt;&gt; 1;</code>

### Bits nach links schieben:

1	1	1	1	1	1	1	1	<code>PORTD = PORTD &gt;&gt; 1;</code>
1	1	1	1	1	1	1	0	<code>PORTD = PORTD &gt;&gt; 1;</code>
1	1	1	1	1	1	0	0	<code>PORTD = PORTD &gt;&gt; 1;</code>
1	1	1	1	1	0	0	0	<code>PORTD = PORTD &gt;&gt; 1;</code>
1	1	1	1	0	0	0	0	<code>PORTD = PORTD &gt;&gt; 1;</code>

Der Parameter ~ kehrt den Wert eines Bits um:

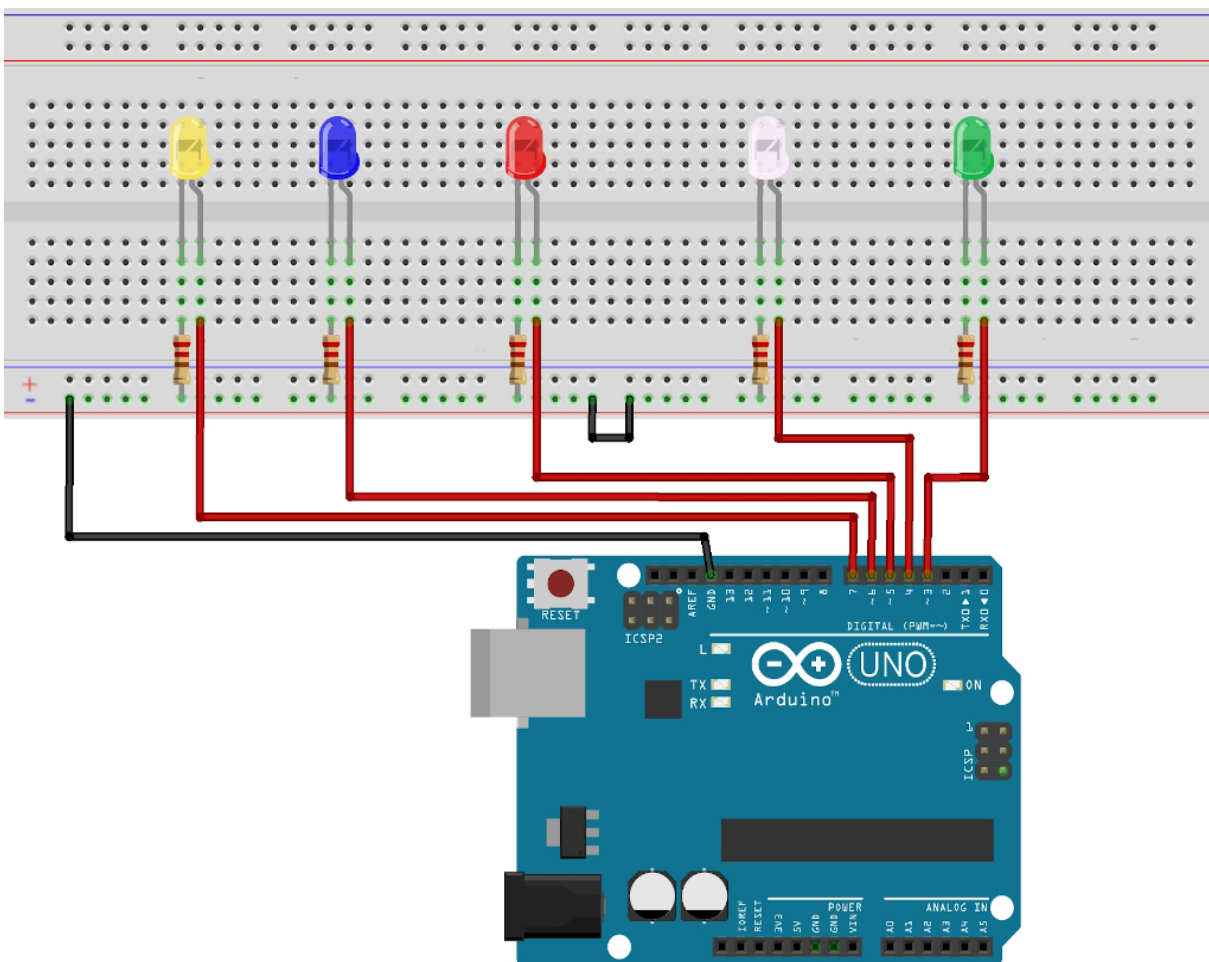
```
// LEDs an den Pins 7 5 3 leuchten
PORTD = B10101000;

// Werte umkehren, aus 0 wird 1 und aus 1 wird 0
PORTD = ~PORTD;
// PORTD hat jetzt den Wert B01010111
// -> Die LEDs an den Pins 6 4 leuchten (Pins 2 bis 0 sind nicht belegt)
```

### Benötigte Bauteile:

- ➔ 5 LEDs
- ➔ 3 Widerstände 220 Ω (gelb, rot und grüne LEDs)
- ➔ 2 Widerstände 100 Ω (blaue und weiße LEDs)
- ➔ Leitungsdrähte

Baue die Schaltung auf



Das Programm. Beachte die Kommentare.

```
int Leuchtdauer = 200;

void setup()
{
  // Pins 7 bis 3 als OUTPUT definieren
  DDRD = B11111000;
}

void loop()
{
  // Lauflicht hin Start mit LED Pin 7
  PORTD = B10000000;
  delay(Leuchtdauer);
  for (int i = 0; i < 4; i++)
  {
    /*
     1 Bit nach rechts schieben
     B01000000 -> Pin 6
     B00100000 -> Pin 5
     B00010000 -> Pin 4
     B00001000 -> Pin 3
    */
    PORTD = PORTD >> 1;
    delay(Leuchtdauer);
  }

  // Lauflicht zurück Start mit LED Pin 3
  PORTD = B00001000;
  delay(Leuchtdauer);
  for (int i = 0; i < 4; i++)
  {
    /*
     1 Bit nach links schieben
     B00010000 -> Pin 4
     B00100000 -> Pin 5
     B01000000 -> Pin 6
     B10000000 -> Pin 7
    */
    PORTD = PORTD << 1;
    delay(Leuchtdauer);
  }
  delay(Leuchtdauer);

  // alle LEDs
  PORTD = B11111000;
  delay(1000);

  // alle aus
  PORTD = B00000000;
```

```
delay(1000);

// LEDs an den Pins 7 5 3 leuchten
PORTD = B10101000;
delay(1000);

// umkehren die inneren LEDs leuchten
PORTD = ~PORTD;
delay(1000);

// die beiden äußeren
PORTD = B10001000;
delay(1000);

// die inneren
PORTD = ~PORTD;;
delay(1000);

// nur die mittlere
PORTD = B00100000;
delay(1000);

// alle aus
PORTD = B00000000;
delay(1000);
}
```