

Arduino-Befehlsübersicht

Programmstruktur

```
//Voreinstellungen
void setup(){
  /*
   * Anweisungen, die beim Start
   * einmalig ausgeführt werden
   */
}
void loop(){
  /*
   * Anweisungen, die wiederholt
   * ausgeführt werden
   */
}
```

Die Programmierung des Arduino Boards

pinMode (int, <TYP>;) <TYP> kann hierbei INPUT oder OUTPUT sein.

digitalWrite (int, <ZUSTAND>;) <ZUSTAND> kann hierbei HIGH (5V) oder LOW (0V) sein.

x = **digitalRead** (int); Liest den Pegel (HIGH oder LOW) aus.

x = **analogRead** (int); Liest den Pegel (von 0 bis 1023) aus.

Schleifen

```
for (START; BEDINGUNG; UPDATE){...}

[Bsp.: for (int i=1; i<10; i++){...}]

while (BEDINGUNG){...}

[Bsp.: while (i<10){...}]
```

Weitere wichtige Befehle

s = **String** (int); Wandelt einen Integer in einen String um.

delay (int); Pausiert die Ausführung. Angabe der Zeit in Millisekunden.

Serial.begin (9600); Startet die Konsole und legt die Baudrate zur Kommunikation mit dem PC fest.

Serial.println (String); Sendet eine Nachricht an die Konsole.

y = **abs** (x); Gibt den absoluten Betrag von x zurück

y = **constraint** (x,min,max); Nimmt den Integer x und beschränkt ihn auf den Maximalwert max bzw. den Minimalwert min.

y = **map** (x, min_ein, max_ein min_aus, max_aus); Nimmt den Integer x aus dem Bereich von min_ein bis max_ein und bildet ihn auf den entsprechenden Wert aus dem Bereich von min_aus, max_aus ab.

Operatoren

Zuweisung	=
Arithmetische Operationen	+ - * /
Modulo	%
Vergleiche	== (gleich)
Vergleiche	!= (ungleich)
Vergleiche	< > (kleiner bzw. größer)
Vergleiche	<= >= (kleiner bzw. größer gleich)
Logische Operatoren	&& (und)
	(or)
	! (not)

Variablen

int zahl = 0;

boolean wahr = true;

string name = "Kurt";

char buchstabe = 's';

PI Entspricht der mathematischen Konstante π

Kontrollstrukturen

if (BEDINGUNG) {...}	Bsp.: if (x==5) {...}
else {...}	else {...}
switch (VARIABLE) {	switch (x) {
case 1: ...; break;	case 1: ...; break;
case 2: ...; break;	case 2: ...; break;
default : ...;	default : ...;
}	}