

UNTERPROGRAMM

- Parameter
- Rückgabewert



Ich bin Mik, Dein
Mikrocontroller

```
int berechneFlaeche(int a, int b)
{
    return a*b;
}
```

UNTERPROGRAMM

```
int berechneFlaeche(int a, int b)
{
    return a*b;
}
```

Unterprogramme
haben Parameter



UNTERPROGRAMM



```
int berechneFlaeche(int a, int b)
{
    return a*b;
}
```

und
Rückgabewerte

UNTERPROGRAMM

```
int berechneFlaeche(int a, int b)
{
    return a*b;
}
```

=

```
berechneFlaeche:
    mul R0,R0,R1
    bx lr
```



in Assembler

UNTERPROGRAMM

```
int berechneFlaeche(int a, int b)
{
    return a*b;
}
```

=

```
berechneFlaeche:
    mul R0,R0,R1
    bx lr
```



welche Parameter
hat das
Unterprogramm?

UNTERPROGRAMM

```
int berechneFlaeche(int a, int b)
{
    return a*b;
}
```

=

```
berechneFlaeche:
    mul R0,R0,R1
    bx lr
```

Vor dem
Unterprogramm-
aufruf werden die
Werte in R0 und
R1 geladen



```
...
mov R0,#25
mov R1,#100
bl  berechneFlaeche
...
```

UNTERPROGRAMM

```
int berechneFlaeche(int a, int b)
{
    return a*b;
}
```

=

```
berechneFlaeche:
    mul R0,R0,R1
    bx lr
```

Unser C-Compiler
verwendet:
R0 als 1. Parameter und
R1 als 2. Parameter



```
...
mov R0,#25
mov R1,#100
bl  berechneFlaeche
...
```

UNTERPROGRAMM

Die
C-Methode
hat einen
Rückgabe-
wert



```
int berechneFlaeche(int a, int b)
{
    return a*b;
}
```

=

```
berechneFlaeche:
    mul R0,R0,R1
    bx lr
```

```
...
int x = berechneFlaeche(25,100)
...
```

=

```
...
mov R0,#25
mov R1,#100
bl  berechneFlaeche
...
```

UNTERPROGRAMM

ebenso das Assemblerunterprogramm:
Das Rechenergebnis findet sich in R0
denn: `mul R0,R0,R1 // R0 = R0 *R1`

```
int berechneFlaeche(int a, int b)
{
    return a*b;
}
```



```
berechneFlaeche:
    mul R0,R0,R1
    bx lr
```

```
...
int x = berechneFlaeche(25,100)
...
```



```
...
mov R0,#25
mov R1,#100
bl  berechneFlaeche
...
```

UNTERPROGRAMM

Der C-Compiler macht es übrigens genauso.
Auch er nutzt R0 als Return-Wert.

```
int berechneFlaeche(int a, int b)
{
    return a*b;
}
```



```
berechneFlaeche:
    mul R0,R0,R1
    bx lr
```

```
...
int x = berechneFlaeche(25,100)
...
```



```
...
mov R0,#25
mov R1,#100
bl  berechneFlaeche
...
```