

Programmieren von Mikrocontrollersystemen

Dr. Stefan Enderle

5. Fallunterscheidung

Fallunterscheidung - if

- Eine Fallunterscheidung führt eine Gruppe von Anweisungen nur dann aus, wenn eine Bedingung erfüllt ist.
- Allgemein:

```
if (<Bedingung>) {  
    <Anweisung>;  
    <Anweisung>;  
    ...  
    <Anweisung>;  
}
```

Aufgabe: Taster abfragen

Der Taster 1 soll andauernd abgefragt werden. Wird er gedrückt, soll die LED 1 leuchten.

Lösung: Taster abfragen

```
#include "qfixMiniBoard.h"

MiniBoard board;

int main()
{
    while(true) {
        if (board.button(1)) {
            board.ledOn(1);
        }
    }
}
```

Fallunterscheidung – if / else

- Eine Fallunterscheidung kann einen ELSE-Zweig enthalten. Dieser wird dann ausgeführt, wenn die Bedingung NICHT erfüllt ist.
- Allgemein:

```
if (<Bedingung>) {  
    ...  
}  
else {  
    ...  
}
```

Aufgabe: Taster abfragen 2

Der Taster 1 soll andauernd abgefragt werden.

Taster gedrückt → LED 1 an

Taster nicht gedrückt → LED 1 aus

Lösung: Taster abfragen 2

```
#include "qfixMiniBoard.h"
```

```
MiniBoard board;
```

```
int main()
```

```
{
```

```
    while(true) {
```

```
        if (board.button(1)) {
```

```
            board.ledOn(1);
```

```
        }
```

```
        else {
```

```
            board.ledOff(1);
```

```
        }
```

```
    }
```

```
}
```

Aufgabe: Externer Taster mit Motor

Ein externer Taster soll andauernd abgefragt werden.

Taster gedrückt → Motor 0 an
Taster nicht gedrückt → Motor 0 aus

Lösung: Externer Taster mit Motor

```
#include "qfixMiniBoard.h"

MiniBoard board;

int main()
{
    while(true) {
        if (board.digital(0)) {
            board.motor(0,255);
        }
        else {
            board.motor(0,0);
        }
    }
}
```

Fallunterscheidung – Bedingungen

- In der Bedingung einer Fallunterscheidung sind beliebige logische (boolsche) Ausdrücke möglich.
- Beispiele:

```
if (a>b) {  
    ...  
}
```

```
if (board.button(0)==true) {  
    ...  
}
```

Aufgabe: Analogwerte einlesen

Ein Potentiometer soll andauernd abgefragt werden.

Weit aufgedreht	→ LED 1 an
Wenig aufgedreht	→ LED 1 aus

Lösung: Analogwerte einlesen

```
#include "qfixMiniBoard.h"

MiniBoard board;

int main()
{
    while(true) {
        if (board.analog(0)>128) {
            board.ledOn(1);
        }
        else {
            board.ledOff(1);
        }
    }
}
```