Bitweise Operatoren zur Bitmanipulation

Bitweise ODER -> zum gezielten Setzen einzelner Bits

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| int wert = 0x29; | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 |
| int maske = 0x03; | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| int ergebnis = wert | maske; | 0 | 0 | 1 | 0 | 1 | 0 | **1** | **1** |

Bitweise UND -> zum gezielten Löschen einzelner Bits

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| int wert = 0x29; | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 |
| int maske = 0xF4; | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 |
| int ergebnis = wert & maske; | 0 | 0 | 1 | 0 | **0** | 0 | **0** | **0** |

Bitweise EXOR -> zum gezielten Invertieren einzelner Bits

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| int wert = 0x29; | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 |
| int maske = 0x33; | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| int ergebnis = wert ^ maske; | 0 | 0 | **0** | **1** | 0 | 0 | **1** | **0** |

Alle Bits invertieren

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| int wert = 0x29; | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 |
| int ergebnis = ~wert | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 |
| int maske = ~0x03; | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 |
| int ergebnis = wert & maske; | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 |

Bitweise Schieben nach links -> entspricht Multiplikation mit 2

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| int wert = 0x29; | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 |
| int ergebnis = wert << 1; | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 |
| ergebnis = wert << 3; | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 |

Bitweise Schieben nach rechts -> entspricht Division durch 2

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| int wert = 0x29; | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 |
| int ergebnis = wert >> 1; | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| ergebnis = wert << 3; | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |

PortIn: Die Ports unseres Mikrocontrollers können gleichzeitig gelesen werden.

(PortA: PA\_15 .. PA\_0, PortB: PB\_15 .. PB\_0 PortC: PC\_15 .. PC\_0)

Legen Sie zunächst fest, welche PINS eines Ports beim gemeinsamen Einlesen beteiligt sind:  
1 -> ist beteiligt 0 -> ist nicht beteiligt. Die Schalterchen sind an PB\_7 .. PB\_0 angeschlossen

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| PB\_15 | PB\_14 | PB\_13 | PB\_12 | PB\_11 | PB\_10 | PB\_9 | PB\_8 | PB\_7 | PB\_6 | PB\_5 | PB\_4 | PB\_3 | PB\_2 | PB\_1 | PB\_0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

in binär: 0b0000000011111111 in hexadezimal 0x00FF

Programmauszug (Gleichzeitiges Einlesen von 8 Schaltern mit PortIn)

#include „mbed.h“

DigitalOut myled(PA\_5);

**PortIn schalter(PortB,0x00FF);**

int main()

{

int wert;

**Schalter.mode(PullDown); //schalter mit Pulldowns**

while (true) {

wert=schalter;

if (wert != 0) //mindestens ein Schalter eingeschaltet

{

myled = 1;

}

else

{

Myled = 0;

}

}

}

Übungen: Lesen Sie immer gleichzeitig die Eingangsinformationen der PortPins PB\_7 .. PB\_0

1. myled soll leichten, wenn PB\_4, PB\_3 und PB\_1 eingeschaltet sind.  
   Maskieren Sie nicht benötigte Eingänge mit bitweise UND
2. Die LED an PC\_0 soll leuchten, wenn einer der Schalter PB\_4, PB\_3 und PB\_1 ausgeschaltet sind. Maskieren Sie die nicht benötigte Eingänge aus.
3. Die LED an PC\_1 soll leuchten, wenn PB\_4 und PB\_3 eingeschaltet und PB\_1 ausgeschaltet sind. Maskieren Sie die nicht benötigte Eingänge aus.
4. Geben sie die Schalter PB\_7..PB\_0 invertiert an PC\_7..PC\_0 aus.
5. Geben Sie PB\_7 .. PB\_0 an PC\_7..PC\_0 aus. Setzen Sie zuvor die Bits mit der Wertigkeit 25 und 21 mit bitweisem ODER.
6. Geben Sie PB\_7 .. PB\_0 an PC\_7..PC\_0 aus. Löschen Sie zuvor die Bits mit der Wertigkeit 25 und 21 mit geeigneter Maskierung.
7. Geben Sie PB\_7 .. PB\_0 an PC\_7..PC\_0 aus. Inverieren Sie zuvor die Bits mit der Wertigkeit 25 und 21 mit bitweisem EXOR.
8. Alle LEDs an PortC sollen blinken