

Montage und Programmierung
eines Roboters für
den Hessen SolarCup
Disziplin: SolaRobot
Teil 2.2: Interrupt

Von Charlotte und Andreas

INTERRUPT

„Paralleler“ Programmablauf

Timer 2 Interrupt

(intern)

Timer 2 Register 8 Bit

1	0	1	0	1	1	0	0
---	---	---	---	---	---	---	---

Timer 2 Interrupt

(intern)

Timer 2

Register

8 Bit

1	0	1	0	1	1	0	0
---	---	---	---	---	---	---	---

0 - 255

Timer 2 Interrupt

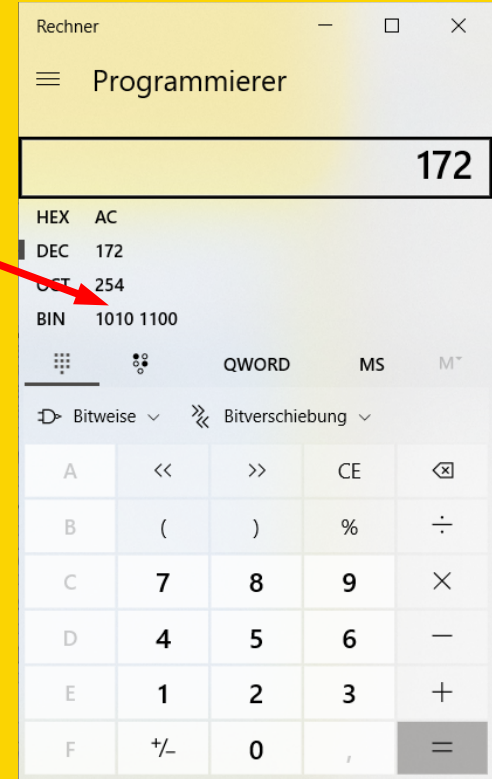
Timer 2 Register 8 Bit

1	0	1	0	1	1	0	0
---	---	---	---	---	---	---	---

= 172

0 - 255

Timer 2 Interrupt



Timer 2

Register

8 Bit

1	0	1	0	1	1	0	0
---	---	---	---	---	---	---	---

= 172

0 - 255

Timer 2 Interrupt

16MHz = 16.000.000 Schwingungen

Timer 2

Register

8 Bit

1	0	1	0	1	1	0	0
---	---	---	---	---	---	---	---

= 172

0 - 255

Timer 2 Interrupt

16MHz = 16.000.000 Schwingungen
16.000.000 Arbeitstakte pro Sekunde

Timer 2 Register 8 Bit

1	0	1	0	1	1	0	0
---	---	---	---	---	---	---	---

= 172

0 - 255

Timer 2 Interrupt

16MHz = 16.000.000 Schwingungen
16.000.000 Arbeitstakte pro Sekunde
Prescaler 1:8 ... 1:256... 1:1024

```
74 //Initialize the Microcontroller
75 void Init (void) {
76     //Port settings (in brackets: pin of the arduino uno board, see above too)
77     //0 = Input, 1 = Output
78     DDRB = 0b00100011; //PortB Pin 5 to Output, LED on Arduino Uno board (13)
79     //Timer2 Time divider
80     TCCR2A = 0b00000000; //0000Normal port operation,00unimpl.,00WGM2[1:0]
81     TCCR2B = 0b00000010; //0WGM22,010Prescaler 1:8
82     TIMSK2 = 0b00000001; //Interrupt at TMR2-OVFL, 128us, 1s = 1000000us, 16MHz
83 }
84
```

Timer 2 Register 8 Bit

1	0	1	0	1	1	0	0
---	---	---	---	---	---	---	---

= 172

0 - 255

Timer 2 Interrupt

16MHz = 16.000.000 Schwingungen
16.000.000 Arbeitstakte pro Sekunde
Prescaler 1:8 ... 1:256... 1:1024
 $62,5\text{ns} * 8 = 0,5\mu\text{s}$ (Mikrosekunden)

```
74 //Initialize the Microcontroller
75 void Init (void) {
76     //Port settings (in brackets: pin of the arduino uno board, see above too)
77     //0 = Input, 1 = Output
78     DDRB = 0b00100011; //PortB Pin 5 to Output, LED on Arduino Uno board (13)
79     //Timer2 Time divider
80     TCCR2A = 0b00000000; //0000Normal port operation,00unimpl.,00WGM2[1:0]
81     TCCR2B = 0b00000010; //0WGM22,010Prescaler 1:8
82     TIMSK2 = 0b00000001; //Interrupt at TMR2-OVFL, 128us, 1s = 1000000us, 16MHz
83 }
84
```

Timer 2 Register 8 Bit

1	0	1	0	1	1	0	0
---	---	---	---	---	---	---	---

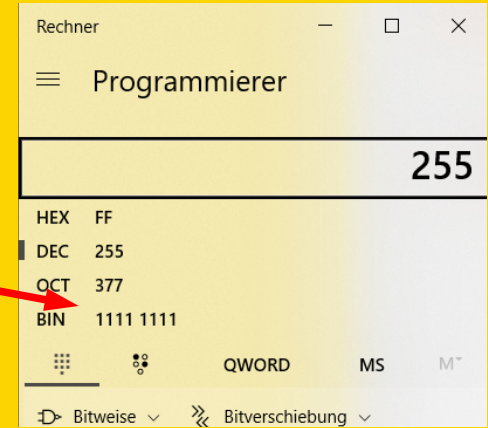
 = 172

0 - 255

1	1	1	1	1	1	1	1
---	---	---	---	---	---	---	---

 = 255

Timer 2 Interrupt



16MHz = 16.000.000 Schwingungen
16.000.000 Arbeitstakte pro Sekunde
Prescaler 1:8 ... 1:256... 1:1024
 $62,6\text{ns} * 8 = 0,5\mu\text{s}$ (Mikrosekunden)

```
74 //Initialize the Microcontroller
75 void Init (void) {
76     //Port settings (in brackets: pin of the arduino uno board, see above too)
77     //0 = Input, 1 = Output
78     DDRB = 0b00100011; //PortB Pin 5 to Output, LED on Arduino Uno board (13)
79     //Timer2 Time divider
80     TCCR2A = 0b00000000; //0000Normal port operation,00unimpl.,00WGM2[1:0]
81     TCCR2B = 0b00000010; //0WGM22,010Prescaler 1:8
82     TIMSK2 = 0b00000001; //Interrupt at TMR2-OVFL, 128us, 1s = 1000000us, 16MHz
83 }
84
```

Timer 2 Register 8 Bit

1	0	1	0	1	1	0	0
---	---	---	---	---	---	---	---

 = 172

0 - 255

1	1	1	1	1	1	1	1
---	---	---	---	---	---	---	---

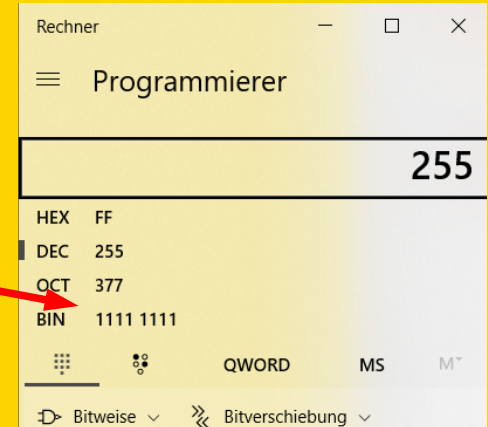
 = 255

+

0	0	0	0	0	0	0	1
---	---	---	---	---	---	---	---

 = 1

Timer 2 Interrupt



16MHz = 16.000.000 Schwingungen
16.000.000 Arbeitstakte pro Sekunde
Prescaler 1:8 ... 1:256... 1:1024
 $62,6\text{ns} * 8 = 0,5\mu\text{s}$ (Mikrosekunden)

```
74 //Initialize the Microcontroller
75 void Init (void) {
76     //Port settings (in brackets: pin of the arduino uno board, see above too)
77     //0 = Input, 1 = Output
78     DDRB = 0b00100011; //PortB Pin 5 to Output, LED on Arduino Uno board (13)
79     //Timer2 Time divider
80     TCCR2A = 0b00000000; //0000Normal port operation,00unimpl.,00WGM2[1:0]
81     TCCR2B = 0b00000010; //0WGM22,010Prescaler 1:8
82     TIMSK2 = 0b00000001; //Interrupt at TMR2-OVFL, 128us, 1s = 1000000us, 16MHz
83 }
84
```

Timer 2 Register 8 Bit

1	0	1	0	1	1	0	0
---	---	---	---	---	---	---	---

= 172

0 - 255

1	1	1	1	1	1	1	1
---	---	---	---	---	---	---	---

= 255

+

0	0	0	0	0	0	0	1
---	---	---	---	---	---	---	---

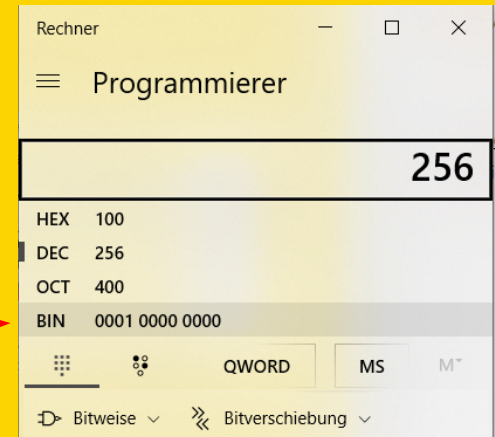
= 1

1

0	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---

= 0

Timer 2 Interrupt



16MHz = 16.000.000 Schwingungen
 16.000.000 Arbeitstakte pro Sekunde
 Prescaler 1:8 ... 1:256... 1:1024
 $62,6\text{ns} * 8 = 0,5\mu\text{s}$ (Mikrosekunden)

```

74 //Initialize the Microcontroller
75 void Init (void) {
76     //Port settings (in brackets: pin of the arduino uno board, see above too)
77     //0 = Input, 1 = Output
78     DDRB = 0b00100011; //PortB Pin 5 to Output, LED on Arduino Uno board (13)
79     //Timer2 Time divider
80     TCCR2A = 0b00000000; //0000Normal port operation,00unimpl.,00WGM2[1:0]
81     TCCR2B = 0b00000010; //0WGM22,010Prescaler 1:8
82     TIMSK2 = 0b00000001; //Interrupt at TMR2-OVFL, 128us, 1s = 1000000us, 16MHz
83 }
84

```

Timer 2 Register 8 Bit

1	0	1	0	1	1	0	0
---	---	---	---	---	---	---	---

= 172

0 - 255

1	1	1	1	1	1	1	1
---	---	---	---	---	---	---	---

= 255

+

0	0	0	0	0	0	0	1
---	---	---	---	---	---	---	---

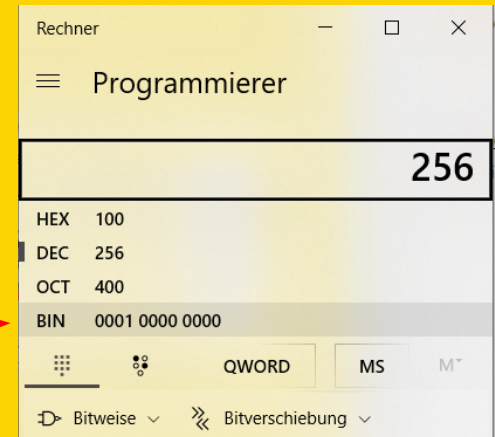
= 1

1

0	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---

= 0

Timer 2 Interrupt



16MHz = 16.000.000 Schwingungen
 16.000.000 Arbeitstakte pro Sekunde
 Prescaler 1:8 ... 1:256... 1:1024
 $62,5\text{ns} * 8 = 0,5\mu\text{s}$ (Mikrosekunden)
 $0,5\mu\text{s} * 256 = 128\mu\text{s}$

```

74 //Initialize the Microcontroller
75 void Init (void) {
76     //Port settings (in brackets: pin of the arduino uno board, see above too)
77     //0 = Input, 1 = Output
78     DDRB = 0b00100011; //PortB Pin 5 to Output, LED on Arduino Uno board (13)
79     //Timer2 Time divider
80     TCCR2A = 0b00000000; //0000Normal port operation,00unimpl.,00WGM2[1:0]
81     TCCR2B = 0b00000010; //0WGM22,010Prescaler 1:8
82     TIMSK2 = 0b00000001; //Interrupt at TMR2-OVFL, 128us, 1s = 1000000us, 16MHz
83 }
84

```

Timer 2 Register 8 Bit

1	0	1	0	1	1	0	0
---	---	---	---	---	---	---	---

= 172

0 - 255

1	1	1	1	1	1	1	1
---	---	---	---	---	---	---	---

= 255

+

0	0	0	0	0	0	0	1
---	---	---	---	---	---	---	---

 = 1

1

0	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---

 = 0

→ Timer 2 Überlauf
Flag wird gesetzt
TOV2 = 1 (...Interrupt Flag)

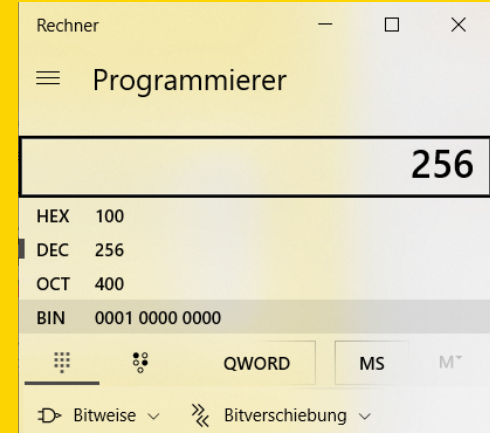
Timer 2 Interrupt

16MHz = 16.000.000 Schwingungen
16.000.000 Arbeitstakte pro Sekunde
Prescaler 1:8 ... 1:256... 1:1024
62,5ns * 8 = 0,5us(Mikrosekunden)
0,5us * 256 = 128us

```

74 //Initialize the Microcontroller
75 void Init (void) {
76     //Port settings (in brackets: pin of the arduino uno board, see above too)
77     //0 = Input, 1 = Output
78     DDRB = 0b00100011; //PortB Pin 5 to Output, LED on Arduino Uno board (13)
79     //Timer2 Time divider
80     TCCR2A = 0b00000000; //0000Normal port operation,00unimpl.,00WGM2[1:0]
81     TCCR2B = 0b00000010; //0WGM22,010Prescaler 1:8
82     TIMSK2 = 0b00000001; //Interrupt at TMR2-OVFL, 128us, 1s = 1000000us, 16MHz
83 }
84

```



Timer 2 Register 8 Bit

1	0	1	0	1	1	0	0
---	---	---	---	---	---	---	---

 = 172

0 - 255

1	1	1	1	1	1	1	1
---	---	---	---	---	---	---	---

 = 255

+

0	0	0	0	0	0	0	1
---	---	---	---	---	---	---	---

 = 1

1

0	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---

 = 0

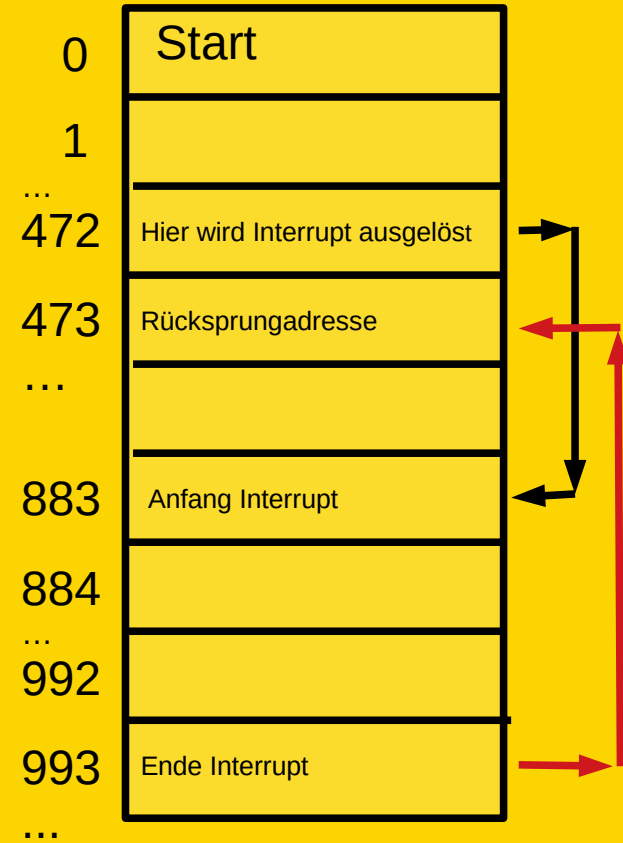
16MHz = 16.000.000 Schwingungen
16.000.000 Arbeitstakte pro Sekunde
Prescaler 1:8 ... 1:256... 1:1024
 $62,5\text{ns} * 8 = 0,5\mu\text{s}$ (Mikrosekunden)
 $0,5\mu\text{s} * 256 = 128\mu\text{s}$

Timer 2 Interrupt

Timer 2 Überlauf
Flag wird gesetzt
TOV2 = 1 (...Interrupt Flag)

↓
Programm wird an der
aktuellen Stelle
angehalten
ATmega springt zur
Interruptroutine →
Am Ende des Interrupts,
springt der MC wieder
an die Stelle zurück →

Programmspeicher



Global Interrupt enable:

Mit der Funktion sei() stellen wir alle Interrupts an, die wir benutzen wollen.

Mit cli() verhindern wir alle Interrupts.

Timer 2 OVF (Overflow)

Beim Überlauf des Timer 2 Registers von 255 auf 0, springt der MC in die Zeile, in der die entsprechende

Interrupt **S**ervice **R**outine definiert ist.

Einstellungen für den Timer 2:

Wir wollen ihn als Timer, nicht als Counter benutzen.

Wir teilen den Arbeitszyklus durch 8, um die Zeit zu messen.

Wir machen den Timer 2 Interrupt an.

```

03_Interrupt - AtmelStudio
File Edit View VAssistX ASF Project Build Debug Tools Window Help
Debug Debug Browser
Interrupt.cpp
49 //Main routine
50 int main(void) {
51     Init(); //Initialize the MC
52     sei(); //Enable all interrupts
53     cli(); //Clear all interrupts
54     while (1) {
55         //Nothing to do!!!
56     }
57 }
58
59 /*****
60 //Interrupt service routines (ISR)
61 //TMR2 overflow Interrupt
62 */
63 ISR(TIMER2_OVF_vect) { //Prescaler TMR2 1:8 = Interrupt every 128 us
64     //Just toggling the LED
65     T2ck++; //Counter TMR2 OVF
66     if(T2ck == 2000) { //Approx. every 250ms (2000*128us = 256ms)
67         T2ck = 0; //Don't forget!!
68         LED_TOGGLE; //See above
69     }
70 }
71 /*****
72 //Other functions
73 */
74 //Initialize the Microcontroller
75 void Init(void) {
76     //Port settings (in brackets: pin of the arduino uno board, see above too)
77     //0 = Input, 1 = Output
78     DDRB = 0b00100011; //PortB Pin 5 to Output, LED on Arduino Uno board (13)
79     //Timer2 Time divider
80     TCCR2A = 0b00000000; //0000Normal port operation,00unimpl.,00WGM2[1:0]
81     TCCR2B = 0b00000010; //00WGM22,010Prescaler 1:8
82     TIMSK2 = 0b00000001; //Interrupt at TMR2-OVFL, 128us, 1s = 1000000us, 16MHz -> 16oscillations
83 }
100 %
Output Error List VA Find References Results
Ready
```

Montage und Programmierung
eines Roboters für
den Hessen SolarCup
Disziplin: SolaRobot
Teil 2.3: Data Visualizer

Von Charlotte und Andreas