Montage und Programmierung eines Roboters für ROBOCUP JUNIOR RESCUE mit Elegoo Car Kit Teil 2.2: Interrupt

Von Charlotte und Andreas

INTERRUPT

„Paralleler“ Programmablauf

Timer 2 Interrupt

(intern)

Timer 2 Register 8 Bit



Timer 2 Interrupt

(intern)

Timer 2 Register 8 Bit



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



= 172

0 - 255

Timer 2 Interrupt

16MHz = 16.000.000 Schwingungen

Timer 2 Interrupt

Timer 2 Register 8 Bit



= 172

0 - 255

16MHz = 16.000.000 Schwingungen
16.000.000 Arbeitstakte pro Sekunde

Timer 2 Interrupt

Timer 2 Register 8 Bit



= 172

0 - 255

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, 16MH
83  }
84
```

Timer 2 Interrupt

Timer 2 Register 8 Bit



= 172

0 - 255

16MHz = 16.000.000 Schwingungen
16.000.000 Arbeitstakte pro Sekunde
Prescaler 1:8 … 1:256... 1:1024
62,5ns * 8 = 0,5us(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, 16MH
83  }
84
```

Timer 2 Interrupt

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



16MHz = 16.000.000 Schwingungen
16.000.000 Arbeitstakte pro Sekunde
Prescaler 1:8 … 1:256... 1:1024
62,6ns * 8 = 0,5us(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, 16M
83  }
84
```

Timer 2 Interrupt

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



16MHz = 16.000.000 Schwingungen
16.000.000 Arbeitstakte pro Sekunde
Prescaler 1:8 … 1:256... 1:1024
62,6ns * 8 = 0,5us(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, 16M
83  }
84
```

Timer 2 Interrupt

Timer 2 Register 8 Bit

	1	0	1	0	1	1	0	0

= 172

0 - 255

	1	1	1	1	1	1	1	1

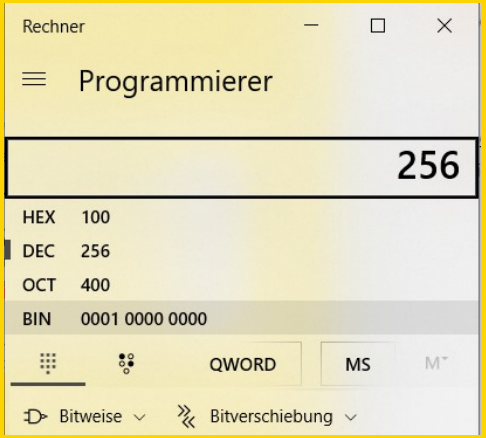
= 255

+	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
62,6ns * 8 = 0,5us(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, 16M
83  }
84
```

Timer 2 Interrupt

Timer 2 Register 8 Bit

	1	0	1	0	1	1	0	0

= 172

0 - 255

	1	1	1	1	1	1	1	1

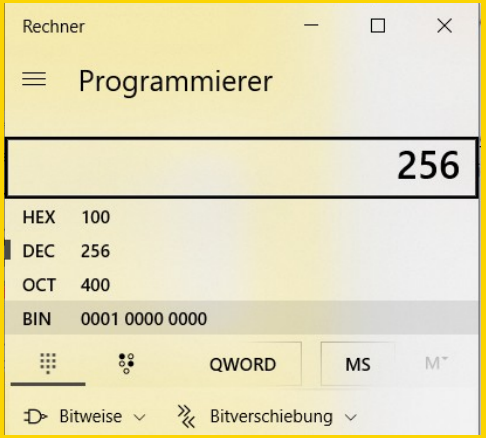
= 255

+	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
62,5ns * 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, 16M
83  }
84
```

Timer 2 Interrupt

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

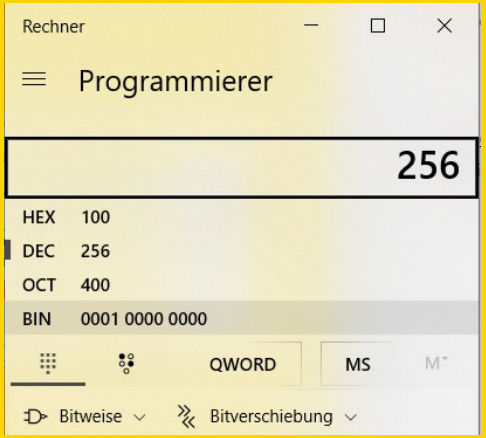
= 1

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

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
62,5ns * 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, 16M
83  }
84
```

Timer 2 Interrupt

Timer 2 Register 8 Bit

1	0	1	0	1	1	0	0	= 172

0 - 255

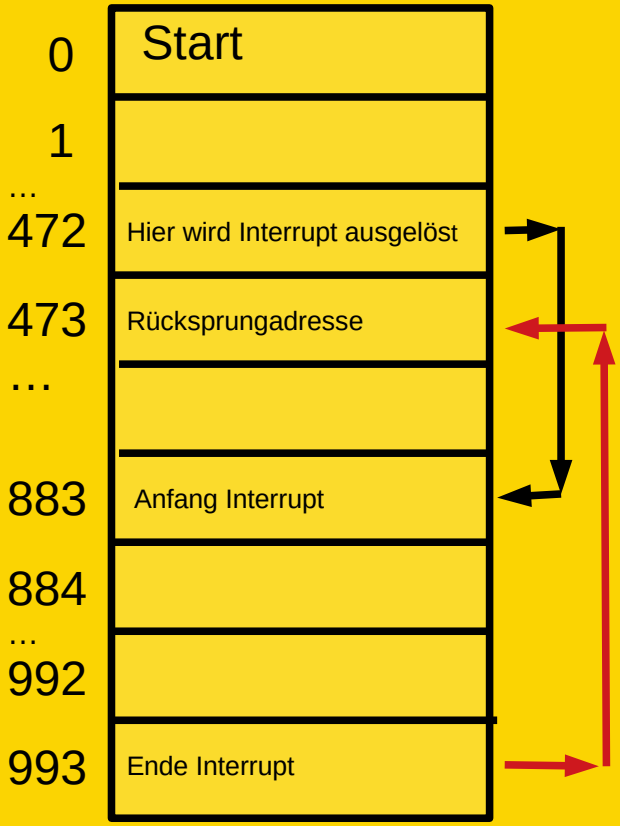
1	1	1	1	1	1	1	1	= 255
+ 0	0	0	0	0	0	0	1	= 1
1 0	0	0	0	0	0	0	0	= 0

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

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

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

Programmspeicher



Global Interrupt enable:

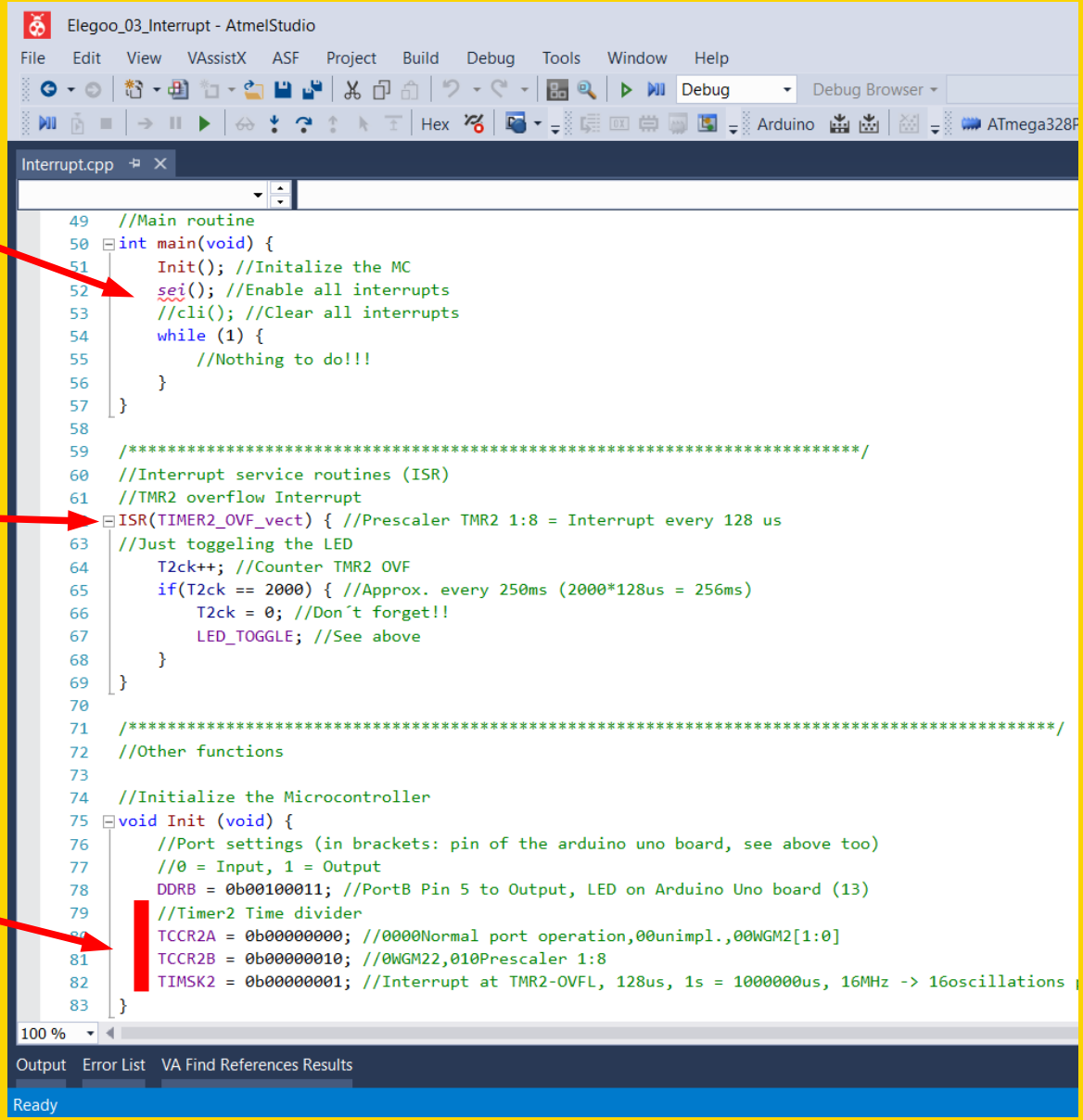
Mit der Funktion sei() stellen wir alle Interrupts an, die wir benutzen wollen.
Mit cli() verhindern wir alle Interrupts.

Timer 2 OVF (Overflow)

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



Montage und Programmierung eines Roboters für ROBOCUP JUNIOR RESCUE mit Elegoo Car Kit Teil 2.3: Data Visualizer

Von Charlotte und Andreas