```
//##############################################################################
//  Timer Routinen
//  Eisenbahnsteuerung made by: Andreas Lutters / Achim Walder
//##############################################################################
// Revision:       01.11.2013
// portions by Ulrich Radig
// Dieses Programm ist freie Software. Sie können es unter den Bedingungen der
// GNU General Public License, wie von der Free Software Foundation veröffentlicht,
// weitergeben und/oder modifizieren, entweder gemäß Version 2 der Lizenz oder
// (nach Ihrer Option) jeder späteren Version.
// Die Veröffentlichung dieses Programms erfolgt in der Hoffnung,
// daß es Ihnen von Nutzen sein wird, aber OHNE IRGENDEINE GARANTIE,
// sogar ohne die implizite Garantie der MARKTREIFE oder der VERWENDBARKEIT
// FÜR EINEN BESTIMMTEN ZWECK. Details finden Sie in der GNU General Public License.
// Sie sollten eine Kopie der GNU General Public License zusammen mit diesem
// Programm erhalten haben.
// Falls nicht, schreiben Sie an die Free Software Foundation,
// Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307, USA.
//  Program uses parts of the above mentioned Software of Ulrich Radig.
// All other code is solely public domain software contributed to the public under the
// above mentioned GNU licence
//##############################################################################

#include "config.h"
#include <avr/io.h>
#include <avr/interrupt.h>
#include "usart.h"
#include "cmd.h"
#include "timer.h"
//#include "stack.h"


volatile unsigned long time;
volatile unsigned long pin = 0;
volatile unsigned long time_watchdog = 0;

	unsigned char Block_Karten_Adresse;		// Block-Karten-Adresse
	unsigned char Strecken_Karten_Adresse;		// Strecken-Karten-Adresse
	unsigned char Pause;                                    // Pause für Ausgabe

//-----------------------------------------------------------------------------
//Diese Routine startet und inizialisiert den Timer
void timer_init (void)
	{
	#if EXTCLOCK==1
		#if defined (__AVR_ATmega644__)
			//Asynchroner Modus ein, Oszillator an TOSC1 und TOSC2 aktiv
			ASSR |= (1 << AS2);
			TCCR2B = 0x05;
			while(ASSR & 0x11);
			//Capture/Compare-Interrupt aktiv
			TIMSK2 |= (1 << OCIE2A);
		#else
			//Asynchroner Modus ein, Oszillator an TOSC1 und TOSC2 aktiv
			ASSR  = (1 << AS2);
			//CTC-Modus an (Clear Timer on Compare Match)
			TCCR2 = (1 << WGM21);
			//dieser Wert ergibt eine Sekunde Periodendauer
			OCR2  = 31;
			//lösche Prescaler 2
			SFIE  = (1 << PSR2);
			//Starte Timer 2 mit Prescaler gleich 1/1024
			TCCR2 |= (1 << CS22)|(1 << CS21)|(1 << CS20);
			while(ASSR & 0x07);
			//Capture/Compare-Interrupt aktiv
			TIMSK = (1 << OCIE2);
		#endif
	#else
		TCCR1B |= (1 << WGM12) | (1 << CS10 | 0 << CS11 | 1 << CS12);
		TCNT1 = 0;
```

```
71      ⟶⟶OCR1A = timer_speed;
72      //⟶⟶OCR1A = (F_CPU / 1024) - 1;      //Org. Radig => F_CPU 11059200 -- 10800 = 1
        Sekunde
73      ⟶⟶TIMSK |= (1 << OCIE1A);
74      ⟶#endif
75      ⟶return;
76      ⟶};
77
78      ⟶//-----------------------------------------------------------------------
79      ⟶//Timer Interrupt
80      ⟶#if EXTCLOCK==1
81      ⟶⟶#if defined (__AVR_ATmega644__)
82      ⟶⟶⟶ISR (TIMER2_COMPA_vect)
83      ⟶⟶#else
84      ⟶⟶⟶ISR (TIMER2_COMP_vect)
85      ⟶⟶#endif
86      ⟶#else
87      ⟶⟶ISR (TIMER1_COMPA_vect)
88      ⟶#endif
89      ⟶
90      ⟶
91      ⟶
        //#######################################################################
        ##########################################
92      ⟶// Timer Routine
93      ⟶
        //#######################################################################
        ##########################################
94      ⟶{   // tick 1 second
95      ⟶time++;
96
97      ⟶// MC_Betrieb_Modus = 3 - Weichensteuerung
        ---------------------------------------------------------------------------
        -----
98      ⟶if ((MC_Betrieb_Modus == 3) && (Weichen_NEU = 1))
99      ⟶⟶{
100     ⟶⟶Weichen_MC_TimerDelay --;
101     ⟶⟶if (Weichen_MC_TimerDelay == 0)
102     ⟶⟶⟶{
103     ⟶⟶⟶DDRC = 0xFF;⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶//
                MC-Port-C output
104     ⟶⟶⟶asm("nop");⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶//
                !!!! mindestens 4 * NOP
105     ⟶⟶⟶asm("nop");
106     ⟶⟶⟶asm("nop");
107     ⟶⟶⟶asm("nop");
108     ⟶⟶⟶asm("nop");
109     ⟶⟶⟶asm("nop");
110     ⟶⟶⟶PORTC = 0x00; ⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶//
                MC-Port-C = 00
111     ⟶⟶⟶usart_write("Timer-Rückstellung !\r\n");
112     ⟶⟶⟶Weichen_NEU = 0;
113     ⟶⟶⟶}
114     ⟶⟶}
115
116
117     ⟶// MC_Betrieb_Modus = 4 - Impuls-Weiten-Steuerung
        ---------------------------------------------------------------------------
118     ⟶if (MC_Betrieb_Modus == 4)
119     ⟶⟶{
120     ⟶⟶// Test Timer
121     ⟶⟶PORTD |= _BV(6);
122
123     ⟶⟶
            //#######################################################################
            ##########################################
124     ⟶⟶// Impulsstufe für die einzelnen Aufgaben
125     ⟶⟶// Peek 1 bis 80 ==> Fahrstufen ausgeben
126     ⟶⟶// Peek 81 ==> alle Blocks und Streckenteile eingeschaltet = 12 Volt
127     ⟶⟶// Peek 82 ==> alle Blocks und Streckenteile Kurschluss eingesen
128     ⟶⟶// Peek 83 ==> alle Blocks und Streckenteile auf NULL
```

```
129     ->->// Peek 84 ==> alle Blocks und Streckenteile Beleg eingesen
130     ->->// Peek 85 ==> alle Blocks und Streckenteile wieder mit Stellung (rot/grün) laden
131     ->->
            //#################################################################################################>
132     //->->if (Peek  > (Fahrstufen_MAX + 3))->->->->->->// Zähler für die Impulsstufen
133     //->->if (Peek  > (Fahrstufen_MAX + 10))->->->->->->// Zähler für die Impulsstufen    # vom 22-12-2018 #
134     //->->->{Peek = 0;}
135
136     ->->Peek++;  // Zähler für die Ausgabe
137
138     ->->// Test Timer - Auf dem Port D wir eine Bit angechaltet zum Messen der Zeit
139     ->->if (Peek  == 1)
140     ->->->{
141     ->->->PORTD |= _BV(5);
142     ->->->PORTD &= 255-_BV(5);
143     ->->->}
144
145     ->->
            //#################################################################################################
146     ->->// Geschwindigkeitsausgabe auf Blockkarten
147     ->->
            //#################################################################################################
148     ->->if ((Peek > 0) && (Peek < (Fahrstufen_MAX + 1)))
149     ->->->{
150     ->->->// MC und Interface Output ---------------------------------------------------------------------------------------------
151     ->->->PORTD |= _BV(7);->->->->->->->->->->->->->->->->->->->->->->->->// MC-Port-D7 auf 1 => Encoder-74LS154 nicht aktiv, kein CS
152     ->->->PORTD |= _BV(3);->->->->->->->->->->->->->->->->->->->->->->->->// MC-Port-D3 auf 1 => Interface-74LS245 Daten output
153     ->->->PORTD &= 255-_BV(4);->->->->->->->->->->->->->->->->->->->->->->->->// MC-Port-D4 auf 0 => Interface-74LS245 Daten aktiv
154     ->->->DDRC = 0xFF;->->->->->->->->->->->->->->->->->->->->->->->->->->->->// MC-Port-C output
155     ->->->// Spalte = Blockkarte; Ziel = Fahrstufe ---------------------------------------------------------------------------------
156     ->->->for(Block_Karten_Adresse = 0; Block_Karten_Adresse <= 3; Block_Karten_Adresse++)->->->->// Block-Karten-Adresse [0..3]
157     ->->->->{
158     ->->->->PORTA = Block_Karten_Adresse; ->->->->->->->->->->->->->->->->->->->->// MC-Port-A => Adresse der Karte ausgeben
159     ->->->->PORTC = FahrStufe[Peek][Block_Karten_Adresse];->->->->->->->->->->->->// MC-Port-C => Daten ausgeben
160     ->->->->PORTD &= 255-_BV(7);->->->->->->->->->->->->->->->->->->->->->->->->// MC-Port-D7 = 0 => Encoder-74LS154 CS für 74LS377
161     ->->->->
                asm("nop");->->->->->->->->->->->->->->->->->->->->->->->->->->->->// !!!! mindestens 4 * NOP
162     ->->->->asm("nop");
163     ->->->->asm("nop");
164     ->->->->asm("nop");
165     ->->->->asm("nop");
166     ->->->->asm("nop");
167     ->->->->PORTD |= _BV(7);->->->->->->->->->->->->->->->->->->->->->->->->
```

```
                    // MC-Port-D7 = [1___.____] => Encoder-74LS154 kein CS für 74LS377 ==> Wert im 74LS377 speichern
            }
        // MC und Interface Grundstellung ----------------------------------------------
        PORTD |= _BV(7);                                                    // MC-Port-D7 = 1 => Encoder-74LS154 nicht aktiv, kein CS
        PORTD |= _BV(3);                                                    // MC-Port-D3 = 1 => Interface-74LS245 Interface-Daten output
        PORTD |= _BV(4);                                                    // MC-Port-D4 = 1 => Interface-74LS245 Daten inaktiv
        PORTA = 0x00;                                                       // MC-Port-D Adresse = [00]
        PORTC = 0x00;                                                       // MC-Port-C Daten = [00]
        }
    // Geschwindigkeitsausgabe ENDE ------------------------------------------------------------------------------------------------
/*

    //#################################################################################################
    // Kurzschlussmessung auf Strecken-Karten
    //#################################################################################################
    if (Peek == (Fahrstufen_MAX + 1))
    {
    // Test Timer
    }

    //#################################################################################################
    // Kurzschlussmessung auf Strecken-Karten
    //#################################################################################################
    if (Peek == (Fahrstufen_MAX + 2))
    {

        //@-----------------------------------------------------------------------------------------------------
        //@ MC und Inteface Output
        PORTD |= _BV(7);                                                    // MC-SteuerPort-D7 auf 1 => Encoder-74LS154 nicht aktiv, kein CS
        PORTD |= _BV(3);                                                    // MC-SteuerPort-D3 auf 1 => Interface-74LS245 Daten output
        PORTD &= 255-_BV(4);                                                // MC-SteuerPort-D4 auf 0 => Interface-74LS245 Daten aktiv
        DDRC = 0xFF;                                                        // MC-Port-C output
        //@ alle Daten = Geschwindigkeit = 0
        PORTC = 0xFF;                                                       // MC-DatenPort-C = [00]

        //@-----------------------------------------------------------------------------------------------------
        //@ Adressen-Port-A = [0..34] alle Block- und Strecken-Daten auf 0
        for (Strecken_Karten_Adresse = 0; Strecken_Karten_Adresse < 35; Strecken_Karten_Adresse++)    // Karte - Adressen-Port-A = [0+35..34+35]
        {
        PORTA = Strecken_Karten_Adresse;                                    //
```

```
                    MC-Adressen-Port-A => Adresse des 74LS244 [35..69] ausgeben
205 ————————————PORTD &= 255-_BV(7);————————————————————————————————————//
                    MC-SteuerPort-D7 = 0 => Encoder-74LS154 CS für 74LS377
206 ————————————asm("nop");————————————————————————————————————————————//
                    !!!! mindestens 4 * NOP
207 ————————————asm("nop");
208 ————————————asm("nop");
209 ————————————asm("nop");
210 ————————————asm("nop");
211 ————————————asm("nop");
212 ————————————PORTD |= _BV(7);————————————————————————————————————//
                    MC-SteuerPort-D7 = [1___._____] => Encoder-74LS154 kein CS für 74LS377
213 ————————————————————————————————————————————————————————————————//
Wert im 74LS377 gespeichert
214 ————————————}
215 ————————
                //@-----------------------------------------------------------------------
                ---------------------------------------------------
216 ————————//@ MC und Inteface Input
217 ————————PORTD |= _BV(7);————————————————————————————————————————//
                MC-D7 auf 1 => 74LS154 Encoder nicht aktiv, kein CS
218 ————————DDRC = 0x00; ————————————————————————————————————————//
                MC-Port-C (Daten) als Eingang
219 ————————PORTC = 0xFF; ————————————————————————————————————————//
                MC-Port-C (Daten) als Eingang - with pull-ups enabled
220 ————————PORTD &= 255-_BV(3);————————————————————————————————————//
                MC-D3 auf 0 => 74LS245 Interface-Daten input
221 ————————PORTD &= 255-_BV(4);————————————————————————————————————//
                MC-D4 auf 0 => 74LS245 Interface-Daten aktiv
222 ————————asm("nop");————————————————————————————————————————————//
                !!!! mindestens 4 * NOP
223 ————————asm("nop");
224 ————————asm("nop");
225 ————————asm("nop");
226 ————————asm("nop");
227 ————————asm("nop");
228 ————————
                //@-----------------------------------------------------------------------
                ---------------------------------------------------
229 ————————//@ Adressen-Port-A = [35..70] ==> Belegdaten von den Eisenbahnkarte 35
                bis 70 einlesen
230 ————————for (Strecken_Karten_Adresse = 0; Strecken_Karten_Adresse < 35;
                Strecken_Karten_Adresse++)——
231 ————————————{
232 ————————————PORTA = Strecken_Karten_Adresse + 35;————————————————//
                    MC-AdressenPort-A für 74LS244 [35..69] ausgeben
233 ————————————//@ Beleg 74LS244 aktiv
234 ————————————PORTD &= 255-_BV(7);————————————————————————————————//
                    MC-SteuerPort-D7 = 0 => Karten-CS 74LS244 = aktiv
235 ————————————asm("nop");————————————————————————————————————————//
                    !!!! mindestens 4 * NOP
236 ————————————asm("nop");
237 ————————————asm("nop");
238 ————————————asm("nop");
239 ————————————asm("nop");
240 ————————————asm("nop");
241 ————————————//@ Beleg von Eisenbahn einlesen
242 ————————————Kurzschluss_Register[Strecken_Karten_Adresse + 1] = PINC;————//
                    Einlesen von Karten-Adresse [35..69] * 8 Bit => BS-Array
243 ————————————PORTD |= _BV(7);————————————————————————————————————//
                    MC-SteuerPort-D7 = 1 => 74LS154 Encoder nicht aktiv
244 ————————————}
245 ————————// Test Timer
246 ————————PORTD &= 255-_BV(2);
247 ————————}
248 ————————// Kurzschlussmessung ENDE
            ---------------------------------------------------------------------------
            ----------------------
249
250 */
251 ————————
```

```
            //##########################################################################
            ##################################################
252     ->->// Besetztmessung
253     ->->// alle Blocks und Streckenteile Daten auf NULL
254     ->->
            //##########################################################################
            ##################################################
255     ->->if (Peek == (Fahrstufen_MAX + 3))
256     ->->->{
257     ->->->
                //@-----------------------------------------------------------------------
                ---------------------------------------------------
258     ->->->//@ MC und Inteface Output
259     ->->->PORTD |= _BV(7);->->->->->->->->->->->->->->->->->->->->//
                MC-SteuerPort-D7 auf 1 => Encoder-74LS154 nicht aktiv, kein CS
260     ->->->PORTD |= _BV(3);->->->->->->->->->->->->->->->->->->->->//
                MC-SteuerPort-D3 auf 1 => Interface-74LS245 Daten output
261     ->->->PORTD &= 255-_BV(4);->->->->->->->->->->->->->->->->->->//
                MC-SteuerPort-D4 auf 0 => Interface-74LS245 Daten aktiv
262     ->->->DDRC = 0xFF;->->->->->->->->->->->->->->->->->->->->->->//
                MC-Port-C output
263     ->->->//@ alle Daten = Geschwindigkeit = 0
264     ->->->PORTC = 0x00;->->->->->->->->->->->->->->->->->->->->->//
                MC-DatenPort-C = [00]
265     ->->->
                //@-----------------------------------------------------------------------
                ---------------------------------------------------
266     ->->->//@ Adressen-Port-A = [0..34] alle Block- und Strecken-Daten auf 0
267     ->->->for (Strecken_Karten_Adresse = 0; Strecken_Karten_Adresse < 35;
                Strecken_Karten_Adresse++)->->->// Karte - Adressen-Port-A = [0+35..34+35]
268     ->->->->{
269     ->->->->PORTA = Strecken_Karten_Adresse;->->->->->->->->->->//
                    MC-Adressen-Port-A => Adresse des 74LS244 [35..69] ausgeben
270     ->->->->PORTD &= 255-_BV(7);->->->->->->->->->->->->->->->->//
                    MC-SteuerPort-D7 = 0 => Encoder-74LS154 CS für 74LS377
271     ->->->->asm("nop");->->->->->->->->->->->->->->->->->->->->//
                    !!!! mindestens 4 * NOP
272     ->->->->asm("nop");
273     ->->->->asm("nop");
274     ->->->->asm("nop");
275     ->->->->asm("nop");
276     ->->->->asm("nop");
277     ->->->->PORTD |= _BV(7);->->->->->->->->->->->->->->->->->->//
                    MC-SteuerPort-D7 = [1___._____] => Encoder-74LS154 kein CS für 74LS377
278     ->->->->->->->->->->->->->->->->->->->->->->->->->->->->->->//
Wert im 74LS377 gespeichert
279     ->->->->}
280     ->->->
                //@-----------------------------------------------------------------------
                ---------------------------------------------------
281     ->->->//@ MC und Inteface Input
282     ->->->PORTD |= _BV(7);->->->->->->->->->->->->->->->->->->->->//
                MC-D7 auf 1 => 74LS154 Encoder nicht aktiv, kein CS
283     ->->->DDRC = 0x00; ->->->->->->->->->->->->->->->->->->->->->//
                MC-Port-C (Daten) als Eingang
284     ->->->PORTC = 0xFF; ->->->->->->->->->->->->->->->->->->->->//
                MC-Port-C (Daten) als Eingang - with pull-ups enabled
285     ->->->PORTD &= 255-_BV(3);->->->->->->->->->->->->->->->->->->//
                MC-D3 auf 0 => 74LS245 Interface-Daten input
286     ->->->PORTD &= 255-_BV(4);->->->->->->->->->->->->->->->->->->//
                MC-D4 auf 0 => 74LS245 Interface-Daten aktiv
287     ->->->asm("nop");->->->->->->->->->->->->->->->->->->->->->//
                !!!! mindestens 4 * NOP
288     ->->->asm("nop");
289     ->->->asm("nop");
290     ->->->asm("nop");
291     ->->->asm("nop");
292     ->->->asm("nop");
293     ->->->
                //@-----------------------------------------------------------------------
                ---------------------------------------------------
```

```
294         //@ Adressen-Port-A = [35..70] ==> Belegdaten von den Eisenbahnkarte 35 bis 70 einlesen
295         for (Strecken_Karten_Adresse = 0; Strecken_Karten_Adresse < 35; Strecken_Karten_Adresse++)
296         {
297         PORTA = Strecken_Karten_Adresse + 35;                    // MC-AdressenPort-A für 74LS244 [35..69] ausgeben
298         //@ Beleg 74LS244 aktiv
299         PORTD &= 255-_BV(7);                                     // MC-SteuerPort-D7 = 0 => Karten-CS 74LS244 = aktiv
300         asm("nop");                                              // !!!! mindestens 4 * NOP
301         asm("nop");
302         asm("nop");
303         asm("nop");
304         asm("nop");
305         asm("nop");
306         //@ Beleg von Eisenbahn einlesen
307         Beleg_Register[Strecken_Karten_Adresse + 1] = PINC;      // Einlesen von Karten-Adresse [35..69] * 8 Bit => BS-Array
308         PORTD |= _BV(7);                                         // MC-SteuerPort-D7 = 1 => 74LS154 Encoder nicht aktiv
309         }
310         }
311
312
313         //#################################################################################################
314         // Besetztmessung jetzt weil alle Impulswerte = 0
315         // alle Werte von Block und Streckenteilen eingelesen
316         // aufgeteilt in zwei Peeks am 22-12-2018
317         //#################################################################################################
318         if (Peek == (Fahrstufen_MAX + 4))
319         {
320         // xxxxxxxxxxxx
321         }
322         // Besetztmessung ENDE ------------------------------------------------------------------------------------------------------
323
324         //#################################################################################################
325         // StreckenRegister an Platinen ausgeben
326         // es werden die einzelnen Streckenteil-Transistoren geschaltet
327         // vom 28.02.2014
328         //#################################################################################################
329         if (Peek == (Fahrstufen_MAX + 5))
330         {
331         // MC und Inteface Output ------------------------------------------------------------------------------------------------------
332         PORTD |= _BV(7);                                         // MC-D7 auf 1 => 74LS154 Encoder nicht aktiv
333         PORTD |= _BV(3);                                         // MC-D3 auf 1 => 74LS245 Interface-Daten output
334         PORTD &= 255-_BV(4);                                     // MC-D4 auf 0 => 74LS245 Interface-Daten aktiv
335         DDRC = 0xFF;                                             // MC-Port-C Daten Output
336         // Streckenteile aktualisieren ------------------------------------------------------------------------------------------------
```

```
                ------------
337 ⟶⟶⟶for (Strecken_Karten_Adresse = 1; Strecken_Karten_Adresse <= 32;
                Strecken_Karten_Adresse++)  ⟶// Karten-Nr [4..34] - Regelung ab
                01.02.2014
338 ⟶⟶⟶⟶{
339 ⟶⟶⟶⟶PORTA = Strecken_Karten_Adresse +
                3;⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶// Port-A
                => Adresse der Karte ausgeben
340 ⟶⟶⟶⟶asm("nop");
341 ⟶⟶⟶⟶PORTC =
                Strecken_Status_Register[Strecken_Karten_Adresse];⟶⟶⟶⟶⟶⟶
                ⟶⟶⟶⟶⟶⟶// Port-C => Daten ausgeben
342 ⟶⟶⟶⟶asm("nop");
343 ⟶⟶⟶⟶
344 ⟶⟶⟶⟶PORTD &=
                255-_BV(7);⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶⟶
```