

```

1  /*=====*/
/* Projekt      :   Olli - eine Holzkiste lernt laufen      */
/*                                                     */
/* Hardware      :   Infineon C509                        */
5 /*                                                     */
/* Dateiname     :   ba_teilauto.c                        */
/*                                                     */
/* Version       :   1.2 vom 17. März 2004                */
/*                                                     */
10 /* Autoren      :   L. Kulf, J. Roos                    */
/*                                                     */
/* Datei-        :   BA Teilautomatik, Ablauf              */
/* beschreibung  :   (nur wenn BA von LCD angefordert wurde, */
/*                                                     */
15 /*              sonst Ablauf Teilautomatik über Datei ba_auto.c) */
/*                                                     */
/*=====*/

20 // -----
// Dateien einbinden
#include "ba_teilauto.h"
#include "mot_outp.h"
#include "umdrehungszaehler.h"
25 #include "main.h"
#include "seriell_olli_seruell.h"

// -----
30 // Variablendeklaration, globale Variablen
_bit pos_erreicht_p1=0; // Position 1 ist erreicht
_bit pos_erreicht_p2=0; // Position 2 ist erreicht
_bit pos_erreicht_p3=0; // Position 3 ist erreicht
_bit pos_erreicht_p4=0; // Position 4 ist erreicht
35 _bit pos_erreicht_p5=0; // Position 5 ist erreicht
_bit pos_erreicht_p6=0; // Position 6 ist erreicht
_bit ta_p1_fahrt_aktiv=0; // BA TA Fahrt auf Pos.1 ist aktiv
_bit ta_p2_fahrt_aktiv=0; // BA TA Fahrt auf Pos.2 ist aktiv
_bit ta_p3_fahrt_aktiv=0; // BA TA Fahrt auf Pos.3 ist aktiv
40 _bit ta_p4_fahrt_aktiv=0; // BA TA Fahrt auf Pos.4 ist aktiv
_bit ta_p5_fahrt_aktiv=0; // BA TA Fahrt auf Pos.5 ist aktiv
_bit ta_p6_fahrt_aktiv=0; // TA-Grundstellungsfahrt ist aktiv
_bit ba_ta_p6_fahrt_hm=0; // Hilfsflag, P6-Fahrt in TA angef.

45 extern _bit m_mot_a1_plus; // Motor Achse 1 R+ ist angesteuert
extern _bit m_mot_a1_minus; // Motor Achse 1 R- ist angesteuert
extern _bit m_mot_a2_plus; // Motor Achse 2 R+ ist angesteuert
extern _bit m_mot_a2_minus; // Motor Achse 2 R- ist angesteuert
extern _bit m_mot_a3_plus; // Motor Achse 3 R+ ist angesteuert
50 extern _bit m_mot_a3_minus; // Motor Achse 3 R- ist angesteuert
extern _bit m_mot_a4_plus; // Motor Achse 4 R+ ist angesteuert
extern _bit m_mot_a4_minus; // Motor Achse 4 R- ist angesteuert
extern unsigned char refp[4]; // Referenzpunktschalter A1-4 angef.
extern unsigned char neu_anzufahrende_pos; // Teilautomatik : aktiver Fahrbefehl
55 extern signed char s_umdr_a1; // Umdrehungszähler Spindel Achse 1
extern signed char s_umdr_a2; // Umdrehungszähler Spindel Achse 2
extern signed char s_umdr_a3; // Umdrehungszähler Spindel Achse 3
extern signed char s_umdr_a4; // Umdrehungszähler Spindel Achse 4

60 // -----
// weitere Informationen :

// Spindelumdrehungen Sollwerte sind als define hinterlegt

65 // Achse 1: Olli links heben -> nach rechts neigen (45 Spindelumdr)
// Achse 2: Olli hinten heben -> nach vorne neigen (45 Spindelumdr)
// Achse 3: Olli rechts heben -> nach links neigen (45 Spindelumdr)
// Achse 4: Hals ausfahren -> unten : refp[0] == 1
70 //               Mitte : 40 Spindelumdr in R+
//               oben : 80 Spindelumdr in R+

75 // -----
// Ablauf Betriebsart Teilautomatik
void ba_teilauto (void)
{
// -----
80 // Grundstellung
if ((neu_anzufahrende_pos>0)
&&(neu_anzufahrende_pos<=6)
&&(!ta_p1_fahrt_aktiv)
&&(!ta_p2_fahrt_aktiv)
85 &&(!ta_p3_fahrt_aktiv)
&&(!ta_p4_fahrt_aktiv)
&&(!ta_p5_fahrt_aktiv))
{
if(refp[0] == 0)

```

```

90      {
          if(m_mot_a1_minus == 0)
          {
              mot_a1_minus();
              ta_p6_fahrt_aktiv=1;
95      }
      }

      if(refp[1] == 0)
      {
100         if(m_mot_a2_minus == 0)
            {
                mot_a2_minus();
                ta_p6_fahrt_aktiv=1;
            }
105     }

      if(refp[2] == 0)
      {
110         if(m_mot_a3_minus == 0)
            {
                mot_a3_minus();
                ta_p6_fahrt_aktiv=1;
            }
115     }

      if(refp[3] == 0)
      {
120         if(m_mot_a4_minus == 0)
            {
                mot_a4_minus();
                ta_p6_fahrt_aktiv=1;
            }
            }

125     if(refp[0] == 1)
        {
            mot_a1_stp();
            umdr_zae_hler_init_a1();
130        }

        if(refp[1] == 1)
        {
            mot_a2_stp();
            umdr_zae_hler_init_a2();
135        }

        if(refp[2] == 1)
        {
140            mot_a3_stp();
            umdr_zae_hler_init_a3();
            pos_erreicht_p4=0;
        }

145     if(refp[3] == 1)
        {
            mot_a4_stp();
            umdr_zae_hler_init_a4();
            pos_erreicht_p1=0;
150            pos_erreicht_p2=0;
            pos_erreicht_p3=0;
            pos_erreicht_p5=0;
        }

155     if ((refp[0] == 1)
        &&(refp[1] == 1)
        &&(refp[2] == 1)
        &&(refp[3] == 1))
        {
160         ta_p6_fahrt_aktiv=0;
        }

    // Hilfsflag bilden, P6-Fahrt in Teilauto angefordert
165     if ((neu_anzufahrende_pos == 6) && (ta_p6_fahrt_aktiv == 1))
        {
            ba_ta_p6_fahrt_hm=1;
        }

170     // Bestätigung an Olli-Control, wenn P6-Fahrt zu Ende ist (BA Teilauto)
    if((ba_ta_p6_fahrt_hm == 1) && (ta_p6_fahrt_aktiv == 0))
    {
        transmit_S0_vorbereitung(6);
        neu_anzufahrende_pos=0;
175        ba_ta_p6_fahrt_hm=0;
    }
}

```

```

180 // Position 1: Olli Mitte, Achse 4 Mitte
    if((neu_anzufahrende_pos == 1) && (!ta_p6_fahrt_aktiv))
    {
185         if(pos_erreicht_p1 == 0)
        {
            if ((!m_mot_a4_plus)
                &&(!m_mot_a4_minus)
                &&(!m_mot_a1_minus)
                &&(!m_mot_a2_minus)
190                &&(!m_mot_a3_minus))
            {
                ta_p1_fahrt_aktiv=1;

                // Motor A4 Richtung+ ein
195                mot_a4_plus();

                // Meldung LCD ausgeben
                ba_ta_msg_p1_aktiv();
            }

200            // Wenn Spindelumdrehungen = Sollwert, dann Motor stop
            if (s_umdr_a4 == S_U_SOLL_A4_M)
            {
                mot_a4_stp();
                ta_p1_fahrt_aktiv=0;
                neu_anzufahrende_pos=0;

                // Meldung auf LCD : Position 1 erreicht
205                msg_p1();
            }

            if(m_mot_a4_plus == 0)
            {
210                pos_erreicht_p1=1;
            }
        }
    }

215 }

220 // Position 2: Achse 1 in R+, Achse 4 Mitte
    if((neu_anzufahrende_pos == 2) && (!ta_p6_fahrt_aktiv))
    {
        if(pos_erreicht_p2 == 0)
        {
225            if (((!m_mot_a4_plus)
                &&(!m_mot_a4_minus))
                &&(!m_mot_a1_plus)
                &&(!m_mot_a1_minus)))
            {
                ta_p2_fahrt_aktiv=1;

                // Motor A4 Richtung+ ein
                mot_a4_plus();
                mot_a1_plus();
230                mot_al_plus();

                // Meldung LCD ausgeben
                b
235                if((m_mot_a4_plus == 0) && (m_mot_a1_plus == 0))
                {
                    pos_erreicht_p2=1;
                }
            }
        }
    }

240 }

245 // Position 3: Achse 3 in R+, Achse 4 Mitte
    if((neu_anzufahrende_pos == 3) && (!ta_p6_fahrt_aktiv))
    {
        if(pos_erreicht_p3 == 0)
        {
250            if (((!m_mot_a4_plus)
                &&(!m_mot_a4_minus))
                &&(!m_mot_a3_plus)
                &&(!m_mot_a3_minus)))
            {
                ta_p3_fahrt_aktiv=1;

                // Motor A4 Richtung+ ein
                mot_a4_plus();
                mot_a3_plus();
255                mot_a3_plus();

                // Meldung LCD ausgeben
                ba_ta_msg_p3_aktiv();
            }
        }
    }

260 }

265 // Wenn Spindelumdrehungen = Sollwert, dann Motor stop
    if (s_umdr_a4 == S_U_SOLL_A4_M)
    {

```

```

        mot_a4_stp();
    }

    if (s_umdr_a3 == S_U_SOLL_A3)
    {
        mot_a3_stp();
        ta_p3_fahrt_aktiv=0;
        neu_anzufahrende_pos=0;

        // Meldung auf LCD : Position 3 erreicht
        msg_p3();
    }

    if((m_mot_a4_plus == 0) && (m_mot_a3_plus == 0))
    {
        pos_erreicht_p3=1;
    }
}

// Position 4: Achse 2 in R+
if((neu_anzufahrende_pos == 4) && (!ta_p6_fahrt_aktiv))
{
    if(pos_erreicht_p4 == 0)
    {
        if((!m_mot_a2_plus) && (!m_mot_a2_minus))
        {
            ta_p4_fahrt_aktiv=1;

            // Motor A2 Richtung+ ein
            mot_a2_plus();

            // Meldung LCD ausgeben
            ba_ta_msg_p4_aktiv();
        }

        // Wenn Spindelumdrehungen = Sollwert, dann Motor stop
        if (s_umdr_a2 == S_U_SOLL_A2)
        {
            mot_a2_stp();
            ta_p4_fahrt_aktiv=0;
            neu_anzufahrende_pos=0;

            // Meldung auf LCD : Position 4 erreicht
            msg_p4();
        }

        if(m_mot_a2_plus == 0)
        {
            pos_erreicht_p4=1;
        }
    }
}

// Position 5: Achse 4 in R+ bis oben
if((neu_anzufahrende_pos == 5) && (!ta_p6_fahrt_aktiv))
{
    if(pos_erreicht_p5 == 0)
    {
        if((!m_mot_a4_plus) && (!m_mot_a4_minus))
        {
            ta_p5_fahrt_aktiv=1;

            // Motor A4 Richtung+ ein
            mot_a4_plus();

            // Meldung LCD ausgeben
            ba_ta_msg_p5_aktiv();
        }

        // Wenn Spindelumdrehungen = Sollwert, dann Motor stop
        if (s_umdr_a4 == S_U_SOLL_A4_O)
        {
            mot_a4_stp();
            ta_p5_fahrt_aktiv=0;
            neu_anzufahrende_pos=0;

            // Meldung auf LCD : Position 5 erreicht
            msg_p5();
        }

        if(m_mot_a4_plus == 0)
        {
            pos_erreicht_p5=1;
        }
    }
}
}

```

```
// -----  
// Dateiende
```