

6.6 UART in C

Es empfiehlt sich, wie auch in Assembler, das Programm in zweckmäßige Unterprogramme zu gliedern. Das folgende Beispiel zeigt eine Lösung in C zum Senden von Daten per UART.

```
//-----
// Titel      : Beispiel Zeichen senden mit UART myAVR Board
// Prozessor  : ATmega8 mit 3,6864 MHz
// Funktion   : Sendet den Text „Hallo Welt!“ per UART
// Schaltung  : Nullmodemkabel an PC und AVR Board anschließen
// Sprache    : C
// Datum     : 28.10.2003
// Version    : 1.4
// Autor     : Dipl. Ing. Päd. Alexander Huwaldt
//-----
#include <avr\io.h>
#include <avr\wdt.h>
//-----
void initUART ()
{
    UBRRL = 23;          // 9600 Baud (siehe Baudratentabelle)
    UCSRB = 8;          // Sender ein UCR = UCSRB
}
//-----
void putChar (char data)
{
    while (!(UCSRA & 32)); // Warte bis UDR leer UDR = UCSRA
    UDR=data;             // sende
}
//-----
void print (char buffer [])
{
    for (int i=0; buffer [i] != 0; i++)
        putChar (buffer[i]);
}
//-----
main ()
{
    initUART ();
    while (true) // Mainloop
    {
        print ("Hallo Welt!");
    }
}
//-----
```

6.7 Ausgaben in C

Eine und Ausgabefunktionen gestalten sich in C recht einfach. Dazu werden die Wertausgaben geschrieben:

```
printf = 0x0000; // wird der PORT 0 als Ausgang konfiguriert
printf = 0x0001; // wird der PORT 1 auf 1kHz getriggert
```

Zu beachten ist, auch dass die „Nachsprache“ C dem Entwickler nicht die Kontrolle der Register, der Status des konkreten Contollers und dessen Funktionsweise abnimmt.