



Dokument NR.: AB-001-U-02

IN/OUT - SBUF REGISTER



Einfache Ein- und Ausgabe über
serielle Schnittstelle mittels Polling



Bitte denken Sie an die Umwelt,
bevor Sie diese Datei ausdrucken

INHALTSVERZEICHNIS

- 1. Aufgabe 1 3
 - 1.1 Verwendete Module: 3
 - 1.2 „Target Mission“ 3
- 2. Lösung Aufgabe 1 4

Modification History:

Version	Date	Comments
1.0	12.2012	first release

Dok. NR.: AB-001-U-02

1. AUFGABE 1

Schreiben Sie ein Programm, das einen Benutzer auffordert einen Zeichen mittels Tastatur-Taste einzugeben. Nach dem dieses Zeichen vom Mikrocontroller empfangen wurde, muss der Mikrocontroller das erkannte Zeichen auf dem HypertTerminal wiedergeben.

- Meiden Sie vorgefertigte Funktionen wie printf()
- Realisieren Sie Empfang- und Sende-Funktionen mittels „Polling“

1.1 VERWENDETE MODULE:

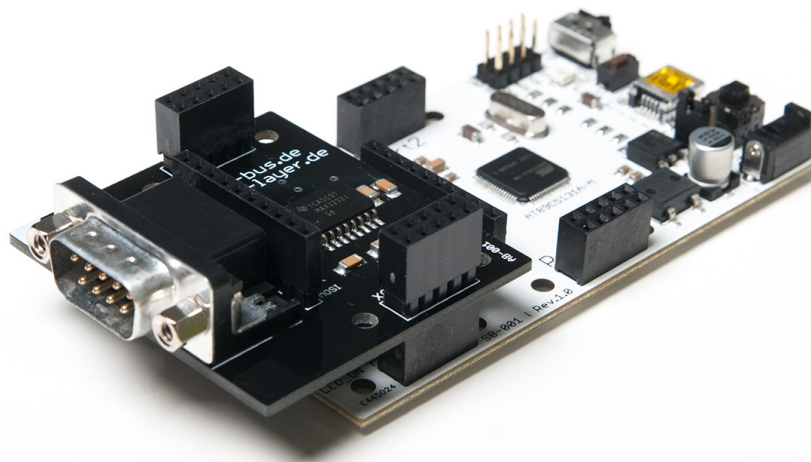
Mikrocontroller Board (SB-001/SB-002) und RS232 TTL Umsetzer (AB-001).

1.2 „TARGET MISSION“

Ein Video sagt mehr, als tausend geschriebene Worte.

In diesem Video wird gezeigt was genau in der Aufgabenstellung 1 (AB-001-U-02) gemeint ist.

[„Target Mission“ auf YouTube](#)



2. LÖSUNG AUFGABE 1

```

/*****
Datei:      Aufgabel_AB-001-U-02.c
Produkt:   AB-001-U-02
Projekt:   Aufgabe 1 IN/OUT with SBUF

Datum:     12.2012 - first release
Version   1.0
Kompiler:  Keil V9.05

Autor:     Viktor Schabelski info@i2c-bus.de.de
Lizenz:    Creative Commons Lizenz
           Namensnennung - Keine kommerzielle Nutzung - Keine Bearbeitungen
           www.i2c-bus.de/cc

```

Schreiben Sie ein Programm, das einen Benutzer auffordert einen Zeichen mittels Tastatur-Taste einzugeben.

Nach dem dieses Zeichen vom Mikrocontroller empfangen wurde, muss der Mikrocontroller das erkannte Zeichen auf dem HypertTerminal wiedergeben.

- Meiden Sie vorgefertigte Funktionen wie printf()
- Realisieren Sie Empfang- und Sendefunktionen mittels „Polling“

```

*****/

/*****
Includes
*****/
#include <at89c5131.h>
#include <stdio.h>

#define QUARZ_FRQ 24 /* Quarz Frequenz */

void init_UART(unsigned char uchQuarz);
void sendString_UART(unsigned char* pString);
void sendChar_UART(unsigned char uchCharacter);
unsigned char receiveChar_UART (void);

/*****
Funktionen
*****/

/*=====
* Name : main
* Input: none
* Output : none
* Descr. : Main Funktion
-----*/
void main (void) {
    init_UART(QUARZ_FRQ);

    sendString_UART („Druecken Sie eine Taste: „);

    while (1) {
        sendChar_UART(receiveChar_UART());
    }
}

```

```

/*=====
* Name : init_UART
* Input: uchQuarz - Quarzfrequenz - erlaubte Werte 12 fuer 12MHz und 24 fuer 24MHz
* Output : none
* Descr. : Initialisierung RS232/V24 Schnittstelle.
*         Initialisierung der seriellen Schnittstelle in Mode 1
*         Schnittstellenparameter: 9600Baud, 8 Datenbit, 1 Stopp-Bit asynchroner Betrieb
*         mit Empfang
*         Quarz Frequenz 12/24 MHz
-----*/
void init_UART(unsigned char uchQuarz){

// Schnittstelle Initialisierung
  SCON |= 0x50;          // SM1 (2^6) = 1; REN (2^4) = 1;

//-----
// Timer2 Initialisierung
  T2CON |= 0x30;        // TCLK (2^4) = 1; RCLK (2^5) = 1;
  TR2 = 1;              // Timmer2 run T2CON 2^2
  RCAP2H = 0xFF;

  if(uchQuarz == 12) RCAP2L = 0xD9;    // Reloadwert bei 12MHz Quarz
  else                RCAP2L = 0xB2;    // Reloadwert bei 24MHz Quarz
}

/*=====
* Name : sendString_UART
* Input: pString - Pointer auf String
* Output : none
* Descr. : Sende String ueber RS232/V24 Schnittstelle.
-----*/
void sendString_UART(unsigned char* pString){

  while(*pString != 0){
    SBUF = *pString; /* sende Zeichen */
    while(TI == 0); /* warte bis Zeichen gesendet ist */
    TI = 0;
    pString++;      /* naechstes Zeichen */
  }
}

/*=====
* Name : sendChar_UART
* Input: uchCharacter - Zeichen zum Senden
* Output : none
* Descr. : Sende Zeichen ueber RS232/V24 Schnittstelle.
-----*/
void sendChar_UART(unsigned char uchCharacter){
  SBUF = uchCharacter; /* sende Zeichen */
  while(TI == 0);     /* warte bis Zeichen gesendet ist */
  TI = 0;
}

```

```
/*=====
 * Name : receiveChar_UART
 * Input : none
 * Output : empfangener Zeichen
 * Descr. : Sende Zeichen ueber RS232/V24 Schnittstelle.
-----*/
unsigned char receiveChar_UART (void) {
    unsigned char uchZeichen = '\0';

    while (RI == 0);      /* Warten bis ganzer Byte ,bertragen ist */
    uchZeichen = SBUF;
    SCON &= 0xFE;        /* RI Flag (2^0) loeschen */
    return uchZeichen;
}
```

Haben Sie einen Fehler entdeckt?

Wir sind dankbar für Ihren Hinweis.
Schicken Sie uns bitte diesen Hinweis
einfach per E-Mail: info@i2c-bus.de.

Vielen Dank!