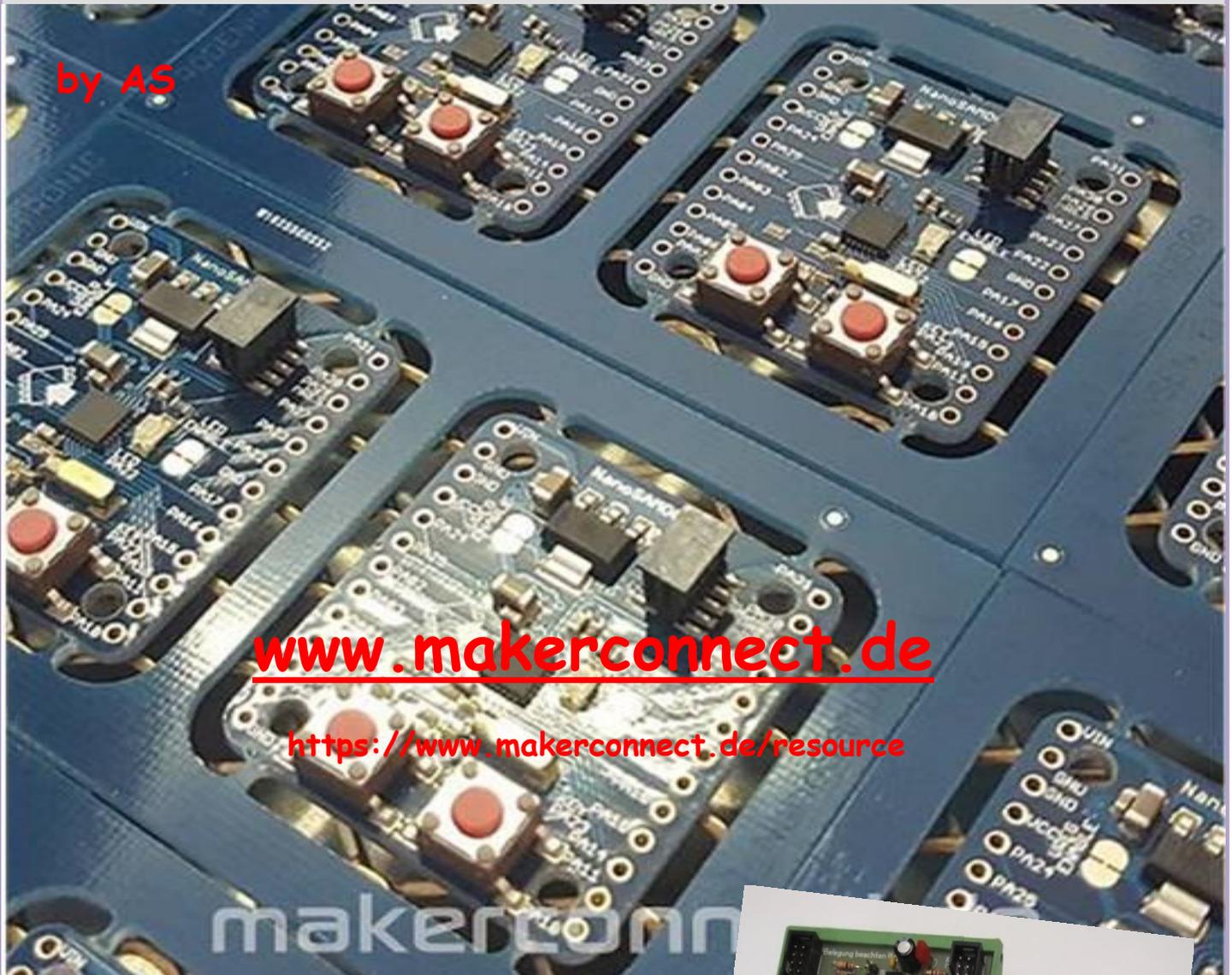


MIKROKONTROLLER & I²C BUS



Eine Spielkonsole mit dem I²C Bus
Teil 3 - Software 2

Der I²C Bus als Spielkonsole



Copyright

Sofern nicht anders angegeben, stehen die Inhalte dieser Dokumentation unter einer „Creative Commons - Namensnennung-NichtKommerziell-Weitergabe unter gleichen Bedingungen 3.0 DE Lizenz“



Sicherheitshinweise

Lesen Sie diese *Gebrauchsanleitung*, bevor Sie diesen Bausatz in Betrieb nehmen und bewahren Sie diese an einem für alle Benutzer jederzeit zugänglichen Platz auf. Bei Schäden, die durch Nichtbeachtung dieser Bedienungsanleitung verursacht werden, erlischt die *Gewährleistung / Garantie*. Für Folgeschäden übernehmen wir keine Haftung! Bei allen Geräten, die zu ihrem Betrieb eine elektrische Spannung benötigen, müssen die gültigen VDE-Vorschriften beachtet werden. Besonders relevant sind für diesen Bausatz die VDE-Richtlinien VDE 0100, VDE 0550/0551, VDE 0700, VDE 0711 und VDE 0860. Bitte beachten Sie auch nachfolgende Sicherheitshinweise:

- Nehmen Sie diesen Bausatz nur dann in Betrieb, wenn er zuvor berührungssicher in ein Gehäuse eingebaut wurde. Erst danach darf dieser an eine Spannungsversorgung angeschlossen werden.
- Lassen Sie Geräte, die mit einer Versorgungsspannung größer als 24 V- betrieben werden, nur durch eine fachkundige Person anschließen.
- In Schulen, Ausbildungseinrichtungen, Hobby- und Selbsthilfwerkstätten ist das Betreiben dieser Baugruppe durch geschultes Personal verantwortlich zu überwachen.
- In einer Umgebung in der brennbare Gase, Dämpfe oder Stäube vorhanden sind oder vorhanden sein können, darf diese Baugruppe nicht betrieben werden.
- Im Falle einer Reparatur dieser Baugruppe, dürfen nur Original-Ersatzteile verwendet werden! Die Verwendung abweichender Ersatzteile kann zu ernsthaften Sach- und Personenschäden führen. Eine Reparatur des Gerätes darf nur von fachkundigen Personen durchgeführt werden.
- Spannungsführende Teile an dieser Baugruppe dürfen nur dann berührt werden (gilt auch für Werkzeuge, Messinstrumente o.ä.), wenn sichergestellt ist, dass die Baugruppe von der Versorgungsspannung getrennt wurde und elektrische Ladungen, die in den in der Baugruppe befindlichen Bauteilen gespeichert sind, vorher entladen wurden.
- Sind Messungen bei geöffnetem Gehäuse unumgänglich, muss ein Trenntrafo zur Spannungsversorgung verwendet werden
- Spannungsführende Kabel oder Leitungen, mit denen die Baugruppe verbunden ist, müssen immer auf Isolationsfehler oder Bruchstellen kontrolliert werden. Bei einem Fehler muss das Gerät unverzüglich ausser Betrieb genommen werden, bis die defekte Leitung ausgewechselt worden ist.
- Es ist auf die genaue Einhaltung der genannten Kenndaten der Baugruppe und der in der Baugruppe verwendeten Bauteile zu achten. Gehen diese aus der beiliegenden Beschreibung nicht hervor, so ist eine fachkundige Person hinzuzuziehen

Bestimmungsgemäße Verwendung

- Auf keinen Fall darf 230 V~ Netzspannung angeschlossen werden. Es besteht dann Lebensgefahr!
- Dieser Bausatz ist nur zum Einsatz unter Lern- und Laborbedingungen konzipiert worden. Er ist nicht geeignet, reale Steuerungsaufgaben jeglicher Art zu übernehmen. Ein anderer Einsatz als angegeben ist nicht zulässig!
- Der Bausatz ist nur für den Gebrauch in trockenen und sauberen Räumen bestimmt.
- Wird dieser Bausatz nicht bestimmungsgemäß eingesetzt kann er beschädigt werden, was mit Gefahren, wie z.B. Kurzschluss, Brand, elektrischer Schlag etc. verbunden ist. Der Bausatz darf nicht geändert bzw. umgebaut werden!
- Für alle Personen- und Sachschäden, die aus nicht bestimmungsgemäßer Verwendung entstehen, ist nicht der Hersteller, sondern der Betreiber verantwortlich. Bitte beachten Sie, dass Bedien- und /oder Anschlussfehler außerhalb unseres Einflussbereiches liegen. Verständlicherweise können wir für Schäden, die daraus entstehen, keinerlei Haftung übernehmen.
- Der Autor dieses Tutorials übernimmt keine Haftung für Schäden. Die Nutzung der Hard- und Software erfolgt auf eigenes Risiko.

Eine Spielkonsole mit dem I²C Bus - Software 2

In zweiten Teil der Software zur Spielkonsole möchte ich die Funktion der einzelnen Taster anwenden. Bei einem Druck auf die Taster auf dem GamePad leuchten die LEDs auf dem Gameboard auf. Im Kopf der Software habe ich die einzelnen Kontakte, die Zuordnung am PCF8574 und die einzelnen Bits beschrieben. Gleichzeitig wird auf dem Display der Name der Taste dargestellt. Damit kann man ohne Probleme die gewünschte Kombination einstellen und testen.

Da ich in dieser Software **else if** verwende, ist ein Multitasking Betrieb nicht möglich.

```
/* ATB Gameboard P43 PRG 2 Author: AS */  
/* Platine P43 Gameboard mit 2 x GamePad P61 am P30 mit OPLED Display  
Anzeige der Taster am Joystick mit Auswertung der PCF8574 */
```

// Kontakte	PCF	Wo?	Bit
// GamePad	8574		
// Pin 3 -	P0 -	L1 auf dem Pad	- fe
// Pin 4 -	P1 -	T0 vom Joystick	- 02
// Pin 5 -	P2 -	T3 auf dem Pad	- 04
// Pin 6 -	P3 -	T1 auf dem Pad	- 08
// Pin 7 -	P4 -	T4 auf dem Pad	- 10
// Pin 8 -	P5 -	T2 auf dem Pad	- 20
//	P6 -	L2 auf dem Gameboard	- BF
//	P7 -	L3 auf dem Gameboard	- 7F

```
// Nachteil: es ist kein Multitasking Betrieb möglich
```

```
#define F_CPU 16000000 // CPU clock in Hz  
#include <stdbool.h>  
#include <avr/pgmspace.h>  
#include "main.h"  
#include <util/delay.h>  
#include "i2clcd.h"  
#include "i2cmaster.h"  
#include "avr/io.h"  
#include "avr/interrupt.h"  
#include <stdlib.h>  
  
uint8_t e;  
uint8_t d;  
uint8_t f;  
#define ADDR2 0x44  
#define ADDR3 0x4c  
#define adr1_w 0x40  
  
void startanzeige() // Anzeige Titelbild  
{  
    lcd_command(LCD_CLEAR);  
    _delay_ms(2);  
    lcd_printlc(1,2,"Gameboard 2 - I2C");  
}
```

```
    lcd_printlc(2,1,"Modul P43-2xJoystick");
    lcd_printlc(3,2,"AT1284 und OLED");
    lcd_printlc(4,4,"bei achim S.");
    _delay_ms(5000);
}

void funktion() // Anzeige GamePad
{
    lcd_command(LCD_CLEAR);
    _delay_ms(2);
    lcd_printlc(1,2,"GamePad1 GamePad2 ");
    lcd_printlc(2,1,"=====");
}

int main(void)
{
    i2c_init(); // Starte I2C Bus
    lcd_init(); // Starte I2CLCD
    lcd_light(0); // 0=Licht an, 1=Licht aus
    // Display Befehle
    lcd_command(LCD_DISPLAYON | LCD_CURSOROFF | LCD_BLINKINGOFF);
    // Display ON/OFF / Cursor ON/OFF / Blinken ON/OFF
    startanzeige();
    _delay_ms(2); // warte 2ms
    funktion();
    _delay_ms(2);

    i2c_start(ADDR2); // alle LED aus an ADDR 2
    i2c_write(0xff); // Schreibe
    i2c_stop(); // Bus Stop

    i2c_start(ADDR3); // alle LED aus an ADDR 3
    i2c_write(0xff); // Schreibe
    i2c_stop(); // Bus Stop

    while(1)
    {
        i2c_start(ADDR2+1); // Starte Lesezugriff auf ADDR 2
        d=i2c_readNak(); // Schreibe Leseergebnis in d
        i2c_stop(); // Bus Stop

        i2c_start(ADDR3+1); // Starte Lesezugriff auf ADDR 3
        f=i2c_readNak(); // Schreibe Leseergebnis in f
        i2c_stop(); // Bus Stop

        if (~d & 0x02) // Abfrage T0 Adresse 2
        { // Wenn T0 gedrückt ist...
            e = 0x7f; // Angabe LED
            i2c_start(ADDR2); // schreibe an Adresse 2
            i2c_write(e); // schreibe e
            i2c_stop(); // Bus stop
        }
    }
}
```

```
    lcd_printlc(3,4," T0 ");           // Anzeige T0
}
else if (~d & 0x04)                   // Abfrage T3 Adresse 2
{                                       // Wenn T3 gedrückt ist...
    e = 0xbf;                          // Angabe LED
    i2c_start(ADDR2);                  // Schreibbefehl
    i2c_write(e);                      // Schreibe e
    i2c_stop();                        // Bus stop
    lcd_printlc(3,4," T3 ");           // Anzeige T3
}
else if (~f & 0x20)                   // Abfrage T2 Adresse 3
{                                       // Wenn T2 gedrückt ist...
    e = 0xfe;                          // Angabe LED
    i2c_start(ADDR3);                  // Schreibbefehl
    i2c_write(e);                      // Schreibe e
    i2c_stop();                        // Bus Stop
    lcd_printlc(3,14," T2 ");          // Anzeige T2
}
else
{
    e = 0xff;                          // alle LED aus
    i2c_start(ADDR2);                  // Angabe Adresse 2
    i2c_write(e);                      // Schreibe e
    i2c_stop();                        // Bus Stop
    lcd_printlc(3,4,"  ");             // Lösche Anzeige

    e = 0xff;                          // alle LED aus 0xff
    i2c_start(ADDR3);                  // Schreibbefehl
    i2c_write(e);                      // Schreibe e
    i2c_stop();                        // Bus Stop
    lcd_printlc(3,14,"  ");            // Lösche Anzeige
}
}
}
```

Das ist die gesamte Software für den zweiten Teil. Die Bibliotheken für den Bus und LCD habe ich hier nicht angegeben.

Falls sie jemand haben möchte kann ich sie gern schicken.

Einige Teile des Textes wurden zur besseren Übersicht farblich gestaltet.

Die Nutzung erfolgt auf eigenes Risiko.

Ich wünsche viel Spaß beim Bauen und programmieren

Achim

myroboter@web.de