

LCD-Modul

Ansteuerung eines LCD-Moduls mit dem kitCON-167 (Infineon C167CR-Starterkit)

Walter Waldner

AUFGABENSTELLUNG

Die Ansteuerung einer **LCD-Punktmatrix-Anzeige** gehört zu den Standard-Anwendungen für Mikrocontroller. Der vorliegende Artikel beschreibt, wie gängige LCD-Module mit dem weitverbreiteten **HD44780**-LCD-Controller vom **Phytec kitCON-167-Board** (Inhalt des **Infineon Starterkit** für den **C167CR**-Mikrocontroller) angesteuert werden können. Die Anzeigeeinheit wird über den C167-Datenbus als **Memory-Mapped-I/O**-Komponente angeschlossen. Die Software wurde in der Sprache C für die KEIL-Toolkette entwickelt. Der Programmcode ist sehr klein, sodass sich das Ansteuerprogramm auch mit der Demoversion des KEIL-Systems (liegt dem Starterkit ebenfalls bei) problemlos übersetzen läßt.

Die folgende Beschreibung bezieht sich auf ein mir vorliegendes LCD-Punktmatrix-Modul mit dem Hitachi-Controller HD44780 (4 Zeilen zu je 20 Zeichen) mit Hintergrundbeleuchtung. Vergleiche mit einigen anderen Punktmatrix-Anzeigen, die vom selben Controller angesteuert werden, haben gezeigt, dass die Ausführungen auch für diese Module (1- und 2-zeilig, 16/20/40 Zeichen) gelten können. Überprüfen Sie bitte aber in jedem Fall anhand des Datenblatts Ihres Moduls, ob die vorgestellte Schaltung und insbesondere die Pinbelegungen übernommen werden können.

Für die folgenden Ausführungen setze ich voraus, dass Sie das Datenblatt des LCD-Moduls (siehe Anhang) und das Kapitel "External Bus Controller" des C167-User Manuals gelesen haben.

ANSCHLÜSSE DES LCD-MODULS

Die LCD-Module mit dem Hitachi HD44780-Controller haben folgende Anschlüsse (Pin-Belegungen gelten für den mir vorliegenden Typ mit 4 Zeilen / 20 Zeichen):

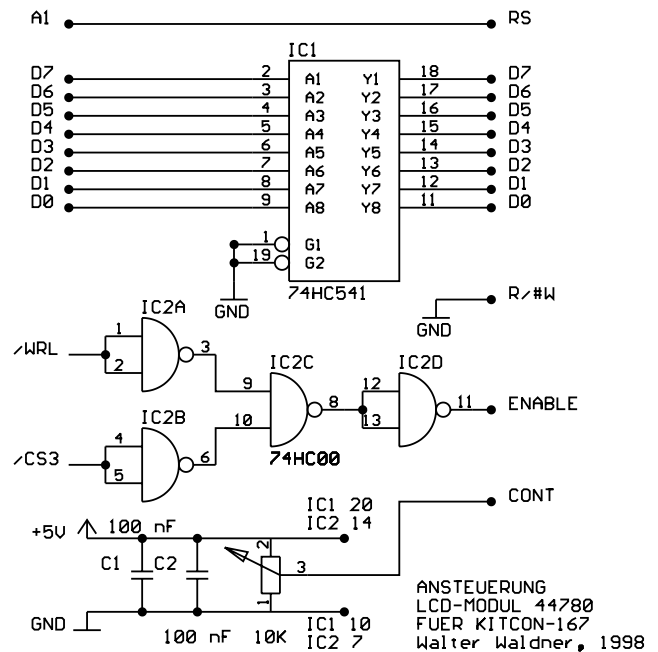
Pin	Symbol	Funktion
1	VSS	Versorgungsspannung: GND = 0V
2	VDD	Versorgungsspannung: 5 Volt
3	CONT	Spannungseingang für Kontrastwahl
4	RS	Register Select HIGH: Datenregister LOW: Befehlsregister
5	R/#W	Read / Write - Signal HIGH = READ LOW = WRITE
6	ENABLE	ENABLE SIGNAL
7	D0	Datenbus-Leitungen (D0=LSB)
8	D1	Datenbus-Leitung (D1)
9	D2	Datenbus-Leitung (D2)
10	D3	Datenbus-Leitung (D3)
11	D4	Datenbus-Leitung (D4)
12	D3	Datenbus-Leitung (D5)
13	D3	Datenbus-Leitung (D6)
14	D3	Datenbus-Leitung (D7)

Manche Module verfügen auch über eine Hintergrund-Beleuchtung. Der mir vorliegende Typ kann mit bis zu 150mA Strom für die Hintergrundbeleuchtung betrieben werden

(beim Anschluss an 5V entsprechenden dimensionierten Vorwiderstand verwenden!).

ANSCHALTUNG DER LCD-ANZEIGE AN DAS PHYTEC KITCON-167-BOARD

Die LCD-Anzeigeeinheit wird mit dem Phytec-kitCON-167-Board über eine kleine Zusatzplatine verbunden. Die Schaltung und die erforderlichen Signale sind nachfolgend beschrieben.



Auf das LCD-Modul kann schreibend und lesend zugegriffen werden. Gelesen werden kann unter anderem das Busy-Flag, das anzeigt, ob die letzte interne Operation abgeschlossen ist und die Inhalte des Daten-RAMs (DDRAM = display data RAM) und des Character-Generator-RAMs (CGRAM).

Der Einfachheit halber werden wir auf das LCD-Modul nur **schreibend** zugreifen. Lesende Zugriffe (insbesondere die Abfrage des Busy-Flags) sind für eine volle Funktion nicht unbedingt erforderlich, da die LCD-Operationen dokumentierte Ausführungszeiten haben. Durch entsprechende Warteschleifen (die Wartezeiten dürfen dabei beliebig über den Minimalangaben liegen) kann sichergestellt werden, dass die Befehle in akzeptabler Geschwindigkeit übergeben werden. Der R/#W-Eingang des LCD-Moduls wird also fest auf GND-Pegel (LOW) gelegt (WRITE ist damit ständig aktiviert).

Das LCD-Modul wird gesteuert, indem Instruktionen und Daten übertragen werden. Zwei Register des HD44780 stehen dafür zur Verfügung:

Befehlsregister: in dieses Befehlsregister werden 8-Bit-Instruktionen geschrieben. Die Instruktionen haben unterschiedliche Ausführungszeiten (siehe Datenblatt)

Datenregister: in dieses Datenregister werden 8-Bit-Daten übertragen. Ist das MSB=0 (most significant bit), stellen die Daten den 7-bit-ASCII-Code des Zeichens dar, das angezeigt wer-

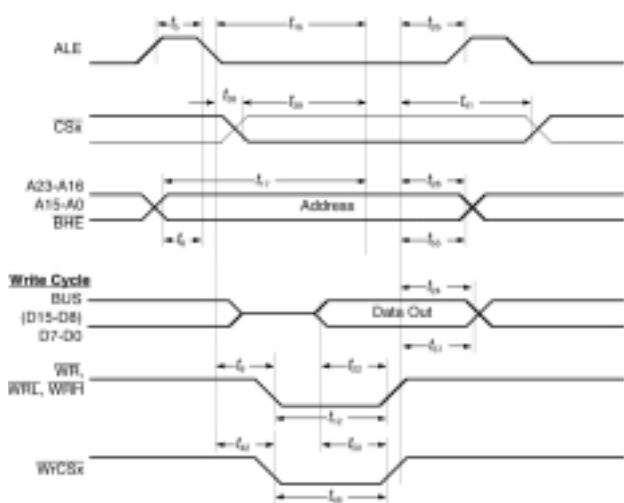
den soll. Manche LCD-Module haben zusätzliche Sonderzeichen (z.B. Katakana - japanische Schriftzeichen). Außerdem können benutzerspezifische Zeichen definiert werden (siehe HD44780-Datenblatt).

Zugriffe auf das Befehls- bzw. Datenregister werden durch den Pegel der Leitung RS (register select) unterschieden. RS kann somit als Adressleitung aufgefasst werden. In der vorliegenden Schaltung wird die C167-Adressleitung A1 auf den "Register Select"-Eingang (RS) des LCD-Moduls geführt. Es gilt:

- RS = 0: Controll-Register wird adressiert
- RS = 1: Datenregister wird adressiert

Die Übernahme der Daten vom Datenbus erfolgt durch die **fal-lende Flanke** des **ENABLE**-Signals. Leider stellt der C167 kein Bussignal zur Verfügung, das direkt als ENABLE verwendet werden kann. Wir können aber ein passendes ENABLE-Signal aus den C167-Signale #WRL und #CS3 erzeugen (siehe nachfolgende Bustiming-Diagramme). Dazu bilden wir die UND-Verknüpfung des invertierten #WRL- und des invertierten #CS3-Signals. ENABLE ist somit HIGH, wenn #CS3 und #WRL LOW sind und fällt auf LOW-Pegel, wenn #WRL wieder HIGH wird. In der hier abgebildeten Schaltung wird dazu der 4-fach-NAND-IC 74HC00 verwendet.

Die folgenden Diagramme zeigen das Timing eines Schreibzyklus für den C167 (demultiplexed 16-Bit-Datenbus) und den HD44780:



Timing für den "demultiplex" Bus - aus dem C167-Datenblatt

Der HD44780 kann über einen 4- oder 8-Bit-Datenbus angesteuert werden. Wir wählen die 8-Bit-Variante. Die C167-Datenleitungen **D0 bis D7** (Port 0L) werden über den Treiber-IC **74HC541** an die Dateneingänge des LCD-Moduls geführt. Eine direkte Verbindung der C167-Datenpins mit dem LCD-Datenbus ist beim Phytec-Board nicht möglich. Folgende Punkte sind zu berücksichtigen:

- Für die C167-Datenbusleitungen (Port 0) gelten (wie für alle Pins) laut Datenblatt bestimmte Grenzwerte für Sink-/Source-Ströme (siehe [1] und [2])
- Der Port 0 des C167 wird beim RESET-Vorgang zur Einstellung bestimmter Buskonfigurationen gelesen. Durch externe Pull-Down-Widerstände (siehe Schaltplan im kitCON-167-Handbuch) werden bestimmte Pins während des Resets auf

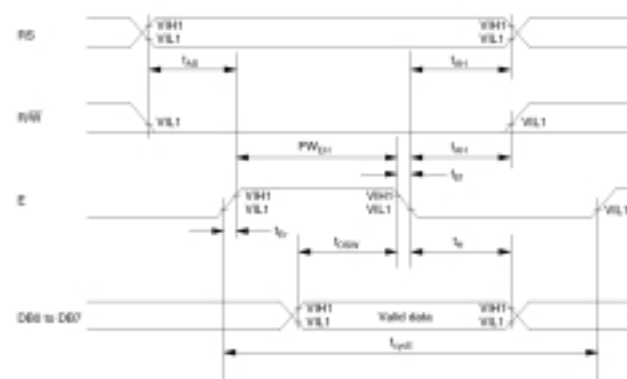
LOW-Pegel gezogen. Die LOW-Pegel werden nur dann korrekt interpretiert, wenn bestimmte Grenzwerte für die Ströme nicht überschritten werden (siehe [2])

Beim Anlegen eines LOW-Signals an eine der Datenleitungen des HD44780 fließen ca. 125 μA über den internen Pull-Up-MOS-Transistor (siehe Abbildung der Eingangsschaltung im HD44780-Datenblatt), was zu viel ist. Insbesondere ist zu berücksichtigen, dass ja auch die Onboard-SRAMs und FLASH-EEPROMs den Datenbus belasten und insbesondere die Pull-Down-Widerstände des Phytec-Boards so gewählt sind, dass das zusätzliche Anschließen des LCD-Controllers die erforderlichen RESET-Pegel für die korrekte Initialisierung des Bus-Interface auf PORT0 verändert.

Aus den eben erwähnten Gründen schalten wir zwischen die unteren 8 Bits des C167-Datenbus (Port 0L) und den Datenleitungen des LCD-Controllers einen Treiber-IC vom Typ 74HC541 (Datenblatt siehe [3]). Die Eingänge des 74HC541 liefern bzw. ziehen bei LOW- und HIGH-Pegel des Eingangssignals nur 0.1 μA (typisch, 1.0 μA maximal).

Beachten Sie bitte außerdem folgende Punkte:

- Als ICs müssen unbedingt CMOS-Bausteine der HC-Serie verwendet werden
- Die 100nF-Kondensatoren sind möglichst nahe an die Versorgungspins der ICs 1 und 2 zu bringen (diese buffern die Spitzenbelastungen der Versorgungsspannung bei Umschaltvorgängen in den CMOS-Ausgangsschaltungen)



Timing für den HD44780-Schreibzyklus - HD44780-Datenblatt

Das **Trim-Poti** (Wert ca. 10 kOhm) dient zur Kontrast-Einstellung der Anzeige. Der Schleiferausgang des Poti wird mit dem **CONTRAST**-Eingang (Pin 3) des Moduls verbunden.

Die Versorgungsspannung (GND und 5V) wird von der kitCON-Steckerleiste auf die LCD-Zusatzplatine geleitet. **ACHTUNG:** Sollte Ihr LCD-Modul über eine Hintergrundbeleuchtung verfügen, so darf die Spannungsversorgung dafür **NICHT** vom kitCON-Board geholt werden, da der Spannungsregler den dafür notwendigen Strom nicht liefern kann. Die Hintergrundbeleuchtung erfordert ca. 150 mA Strom (Vorwiderstand oder Konstantstromschaltung verwenden).

Das C-Source-Programm

```

/*****
Ansteuerung eines LCD-Modul mit Baustein HD44780
ueber C167-Datenbus (Memory-Mapped-I/O)
(c) 1999 by Walter Waldner

```

Anschlüsse:

```

DO..D7: Datenleitungen
A1   : register select (0=instruction, 1=data)
WRL  : write enable (low byte)
CS3  : chip select
*****/

```

```

#include <reg167.h>
#include <absacc.h>
#include <string.h>

```

```
void write_string(char * s, unsigned char line);
```

```

#define LCD_CNTL MVAR (unsigned char,0x800000)
#define LCD_DATA MVAR (unsigned char,0x800002)

```

```
void wait(unsigned int w);
```

```
void main(void)
```

```

{

    unsigned char x;

    /* BUSCON3 / ADDRSEL3 */
    BUSCON3 = 0x0430;
    /* Adressbereich = 80:0000 .. 80:0FFF */
    ADDRSEL3 = 0x8000;

    /* Initialisierung Timer T3 - 1.6 us */
    T3CON = 0x0002;

    /* Software-Initialisierung */
    x = 0x38;
    wait(10000); /* 16 ms */
    LCD_CNTL = x;
    wait(3000); /* 4.8 ms */
    LCD_CNTL = x;
    wait(3000); /* 4.8 ms */
    LCD_CNTL = x;
    wait(3000); /* 4.8 ms */

    /* init display */
    LCD_CNTL = x;
    wait(3000); /* 4.8 ms */
}

```

```

/* set display off */
LCD_CNTL = 0x08;
wait(3000); /* 4.8 ms */

```

```

/* clear display */
LCD_CNTL = 0x01;
wait(3000); /* 4.8 ms */

```

```

/* display on */
LCD_CNTL = 0x0C;
wait(3000); /* 4.8 ms */

```

```

write_string("LCD-MODUL 44780 ",1);
write_string("POWERED BY INFINEON",2);
write_string("C167 MICROCONTROLLER",3);
write_string("Walter Waldner 1999",4);
while (1);
}

```

```
void wait(unsigned int w)
```

```

{
    /* Aufloesung T3: 1.6 microseconds */
    T3 = 0;
    T3R = 1;
    while (T3 < w);
    T3R = 0;
}

```

```
void write_string(char * s, unsigned char line)
```

```

{
    unsigned int i;

    /* DDRAM-Adresse setzen */

    if (line == 1)
        LCD_CNTL = 0x80;
    if (line == 2)
        LCD_CNTL = 0xC0;
    if (line == 3)
        LCD_CNTL = 0x94;
    if (line == 4)
        LCD_CNTL = 0xD4;
    wait(3000); /* 4.8 ms */

    for (i=0; i<strlen(s); i++)
    {
        LCD_DATA = s[i];
        wait(30); /* 48 us */
    }
}

```

Wie bereits eingangs erwähnt, wird das LCD-Modul mittels der oben beschriebenen Schaltung als I/O-Komponente an den C167 angeschlossen. **Memory-Mapped-I/O** bedeutet, dass ein Ein-/Ausgabe-Baustein quasi wie Speicher mit den Controller verbunden wird und die Register über Variable angesprochen werden können. Dem I/O-Baustein wird ein Adressbereich im Adressraum zugeordnet. Wenn eine Adresse innerhalb dieses Adressraums generiert wird, muss die I/O-Komponente selektiert sein ("**Chip select**"). Das Erzeugen eines Chip-Select-Signals erledigen oft externe Adressdekoder-Schaltungen. Der Infineon C167-Mikrocontroller stellt bereits 5 Chip-Select-Leitungen zur Verfügung, sodass auf externe Zusatzlogik verzichtet werden kann. Das Generieren der Chip-Select-Signale wird über die On-chip-Einheit "**External Bus-Controller**" des C167 gesteuert.

Das Phytec-kitCON-167-Board verwendet Chip-Select-1 (CS1) für die Ansteuerung des SRAMs und Chip-Select-0 (CS0) für die Ansteuerung des FLASH-EEPROMs. Die übrigen Chip-Select-Leitungen (CS2, CS3, CS4) sind verfügbar. Für unser Projekt verwenden wir **CS3**.

Für die Anbindung des LCD-Moduls über den Datenbus des C167 müssen die beiden Register **BUSCON3** und **ADDRSEL3** entsprechend konfiguriert werden.

BUSCON3 enthält die Einstellungen für die Bussignale und das Timing, **ADDRSEL3** spezifiziert den Adressbereich, für den CS3 aktiviert (LOW) werden soll.

In unserem Programm weisen wir dem LCD-Modul den Adressbereich **80000** bis **803FFF** zu (4 kB ist die kleinstmögliche Größe eines Adressfensters). Dazu muss in das ADDRSEL3-Register der Wert 0x8000 geschrieben werden. Die Wahl des Adressbereichs ist ziemlich willkürlich. Nicht verwendet werden sollten Adressbereiche, die von den SRAM- bzw. FLASH-Speichern belegt sind (siehe [4] und [5]). Der hier gewählte Bereich liegt fernab von anderwärtig belegten Adressen.

Das LCD-Modul erfordert, wie bereits erwähnt, lediglich zwei Adressen (zur Unterscheidung zwischen Zugriffen auf das Daten- bzw. Befehlsregister). Gemäß unserer Schaltung ist die C167-Adressleitung A1 mit RS (register select) verbunden. Wir definieren im Programm zwei **absolute Variable** (Variable mit genau festgelegter Adresse) im Adressbereich 80000 bis 803FFF. Über

diese Variablen können wir Daten bzw. Befehle an den LCD-Controller senden.

Für Zugriffe auf das HD44780-Befehlsregister muss RS=0 (A1=0) sein, für Zugriffe auf das HD44780-Datenregister muss RS=1 (A1=1) sein. Wir legen daher folgende Variablen fest:

```
#define LCD_CNTL MVAR (unsigned char,0x800000)
#define LCD_DATA MVAR (unsigned char,0x800002)
```

LCD_CNTL ist eine absolute char-Variable (8 bit) an der Adresse 0x800000 (A1=0), LCD_DATA eine absolute char-Variable an der Adresse 0x800002 (A1=1). Eine Zuweisung auf die Variable LCD_CNTL bewirkt ein Schreiben einer Instruktion in das HD44780-Befehlsregister, eine Zuweisung an die Variable LCD_DATA schreibt Daten in das Datenregister des HD44780.

Beachten Sie: für die beiden absoluten Variablen könnten auch andere Adressen im Adressbereich 800000 bis 803FFF gewählt werden. Voraussetzung ist nur, dass die Adressen stets gerade sind (da wir #WRL für die Generierung des ENABLE-Signals verwenden) und dass die Adressleitung A1 den entsprechenden Wert hat.

Bustiming

Über das BUSCON3-Register wird das Timing der Bussignale für Lese-/Schreiboperationen im durch ADDRSEL3 spezifizierten Adressbereich bestimmt.

Für unser Beispiel stellen wir ein (siehe **Kapitel 8** im C167-User Manual):

```
BUSCON3 = 0x0430;
```

Für die Ansteuerung des LCD-Moduls funktionieren auch verschiedene andere BUSCON3-Werte. Beachtet werden muss, dass ein "demultiplexed"-Bus (getrennter Adress- und Datenbus) verwendet wird. Außerdem sind Waitstates erforderlich. Das erklärt sich daraus, dass wir das ENABLE-Signal aus dem #WRL- und dem #CS3-Signal erzeugen. Laut HD44780-Datenblatt sollte ein ENABLE-Signal mindestens 500 ns HIGH sein. Wir bewegen uns auf der sicheren Seite und programmieren die maximale Anzahl von Waitstates (15) und erfüllen so die Mindestforderung, da durch die Waitstates auch die Signale #WRL und #CS3 entsprechend verlängert werden.

Programmablauf

Zunächst werden die Register BUSCON3 und ADDRSEL3 konfiguriert. Anschließend wird der Timer T3 initialisiert. Der Timer T3 wird (im Unterprogramm wait) für die Realisierung von Warteschleifen verwendet. Nach der Übergabe von Befehlen bzw. Daten sind gemäß HD44780-Datenblatt bestimmte Mindest-Wartezeiten einzuhalten, wenn der Abschluss einer internen Operation nicht über das Busy-Flag abgefragt wird. **T3CON** wird mit dem Wert **0x0002** beschrieben. Damit wird festgelegt, dass das Timer-Register T3 in Abständen von 1.6 µs inkrementiert wird, wenn der Timer über T3R=1 gestartet wurde.

Als nächstes folgt die Software-Initialisierung des LCD-Moduls. Dazu wird dreimal der Wert 0x0038 in das Befehlsregister geschrieben, wobei nach jedem Schreiben bestimmte Wartezeiten einzuhalten sind.

Danach ist das LCD-Modul bereit, verschiedene Befehle entgegenzunehmen. In unserem Programm werden folgende Befehle gesandt (siehe Datenblatt):

- Initialisierung (Busbreite, Anzeigeart)
- Display ausschalten
- Display löschen
- Display einschalten

Die entsprechenden Operationen werden durch Einschreiben bestimmter Werte in das Befehlsregister (über die absolute Variable LCD_CNTL) programmiert.

Nun können die Datenbytes übertragen werden. Das Beispielprogramm enthält dazu das Unterprogramm

```
void write_string(char * s, unsigned char line)
```

Der Parameter s ist ein Pointer auf den String, der angezeigt werden soll. line beschreibt, in welche Zeile des Displays der Text geschrieben werden soll. Für den Zeilenwechsel muss die Startadresse des Display Data RAMs gesetzt werden, was durch die vier if-Anweisungen geschieht. Nach dem Schreiben eines Zeichens wird der interne Adresszähler automatisch inkrementiert. Das Setzen der DDRAM-Adressen ist somit nur von Zeile zu Zeile erforderlich.

Bemerkung zum Timing

Die Ausführung der HD44780-Operationen dauert unterschiedlich lange. Ein nachfolgender Befehl darf erst übertragen werden, wenn der vorangegangene abgeschlossen ist. Da wir auf die Abfrage des Busy-Flags verzichten (dazu wäre anstelle des 74HC541 ein bidirektionaler Transceiver-IC erforderlich), muss nach jedem Befehl eine Warteschleife ablaufen. Die Ausführungszeiten der HD44780-Operationen hängen im wesentlichen von der Taktfrequenz ab, mit der dieser Controller betrieben wird. Diese Taktfrequenz ist nicht bei allen Modulen einheitlich. Im HD44780-Datenblatt können Sie die Ausführungszeiten der einzelnen Befehle bei gegebener Taktfrequenz nachlesen. Der Einfachheit halber wurde im obigen Beispielprogramm einheitlich eine Wartezeit von ca. 4.8 ms nach jedem Befehl und 48 µs nach dem Schreiben eines Wertes in das DDRAM programmiert. Diese Wartezeiten sind zum Teil stark überhöht (können beliebig lang gewählt werden) und sollten für die gängigen 44780-basierenden LCD-Anzeigen passen. Anhand des 44780-Datenblattes und entsprechend der Taktfrequenz des LCD-Controllers können Sie die Parameter in den wait-Aufrufen für Ihr Modul auf Wunsch herabsetzen.

Ergänzende und weiterführende Literatur und Web-Sites zum Thema des Artikels

- [1] Datenblatt zum Hitachi LCD-Controller HD44780
<http://semiconductor.hitachi.com/search/tree/>
- [2] Arbeiten mit C166-Controllern, Karl-Heinz Mattheis u. Steffen Strandt, Traunreut, Feger Hard- und Software-Verlag, 1995. ISBN: 3-928434-26-8
- [3] Datenblatt zum 74HC541
<http://www-eu.semiconductors.philips.com/logic/>
http://www.fairchildsemi.com/catalog/Digital_Buffers.html
- [4] Erfolgreich Starten mit dem Infineon C167CR-Starterkit und dem Software-Entwicklungssystem von KEIL, Walter Waldner 1999. Verfügbar auf <http://www.htblmo-klu.ac.at/lernen/>
- [5] Generieren des Target-Monitors für das Phytect-kitCON-167-Board und die Keil-Toolkette (Infineon C167-Starterkit), Walter Waldner, 1999. Verfügbar auf <http://www.htblmo-klu.ac.at/lernen/>
- [6] Umfassende Informationen über die Infineon-Mikrocontroller finden Sie auf der Web-Seite <http://www.infineon.com/microcontrollers/>
- [7] Das Internet-Angebot der Firma KEIL finden Sie unter den Adressen <http://www.keil.com/>
<http://www.keil.com/-market/>