



# *uC-Board (8051)*

Hardware

Spezifikation

Programmbeschreibung

Document Revision: 2.00

Document Status: Released

Date last changed: 28.07.03

Written by: Sven Freitag

Comments: Software Rev. 1.01



### Distribution:

Name	Internet	Tel.	Fax	E-Mail	OK
Sven Freitag	www.FrydaySoft.de			<a href="mailto:Svenfreitag@gmx.de">Svenfreitag@gmx.de</a> <a href="mailto:Info@frydaysoft.de">Info@frydaysoft.de</a>	

### Document Revision History

Revision	Update	Author	Change / Chapter
2.00		Sven Freitag	Dokumentation zum uC-Board (8051)
1.01		Sven Freitag	Up/DownLoader zum uC-Board (8051)

### Related Documents

Document	Revision	Author/Company/Departm.
	1.50	S. Freitag





*Inhaltsverzeichnis:*

**1. EINLEITUNG** ..... 4

**2. ALLGEMEINES ZUM uC-BOARD** ..... 5

**3. WAS BENÖTIGT MAN ZUSÄTZLICH** ..... 6

**4. BESTÜCKUNG** ..... 6

**5. JUMPEREINSTELLUNGEN** ..... 8

**6. ANSCHLUßMÖGLICHKEITEN** ..... 9

**7. ENTWICKLUNGSSOFTWARE** ..... 10

## 1. Einleitung

Einplatinencomputer sind kompakte, aber vollwertige Computer. Bild 1 zeigt schematisch die wichtigsten Komponenten von Computern und deren Zusammenspiel.

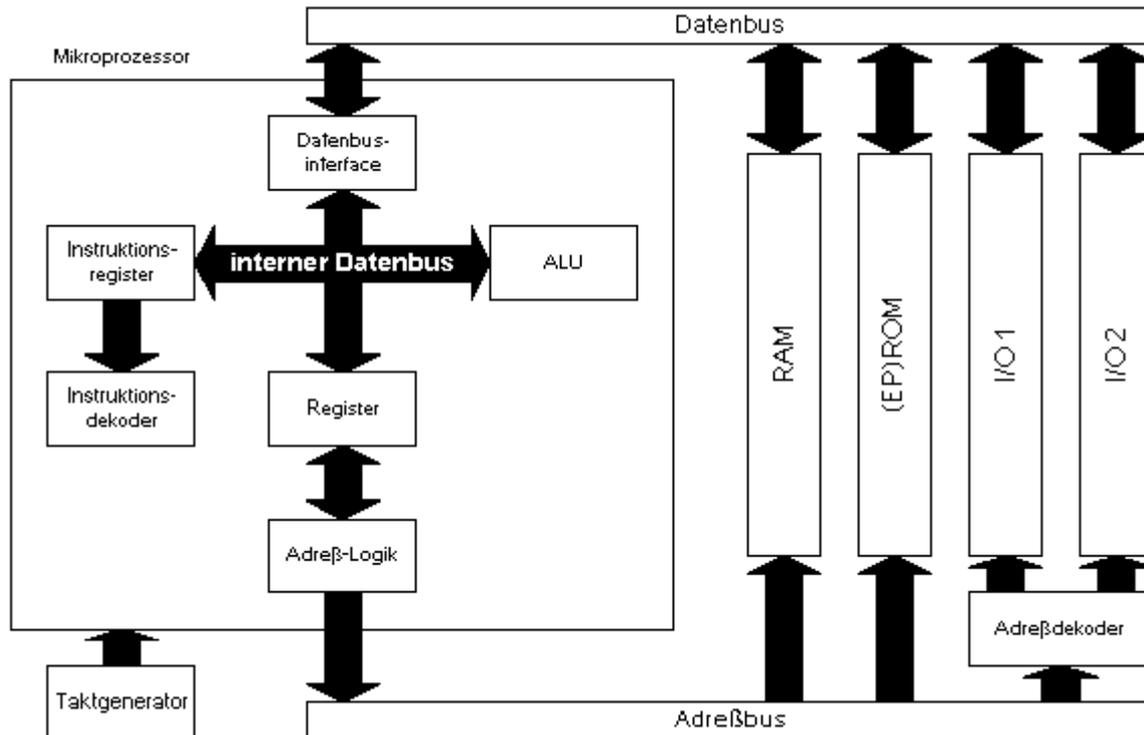


Bild 1

Externe Speicher und Peripherie sind mit dem Mikroprozessor über zwei verschiedene Bussysteme verbunden, dem Datenbus und dem Adreßbus. Festwertspeicher (ROM) und Schreib/Lese-Speicher (RAM) sind mit beiden Bussystemen gekoppelt. Sie können daher direkt vom Mikroprozessor angesprochen werden. Die Peripherie läßt sich in der Regel nicht direkt adressieren und wird daher über einen Adreßdekodeur angesprochen. Damit der Mikroprozessor arbeiten kann, benötigt er noch einen Takt. Dieser wird von dem Taktgenerator erzeugt.

Ein Mikrocontroller, wie z.B. der 8051 von Intel enthält die in Bild 1 gezeigten Komponenten auf einem Chip integriert. Er enthält neben diesen Komponenten eine Serielle Schnittstelle und zwei Timer/Zähler. Da der RAM und ROM auf dem Chip nur sehr klein ist, können über die I/O-Ports externe Speicher angesprochen werden. Beim 8031/32 ist dies sogar erforderlich, da diese Typen keinen internen ROM besitzen.

Wer kennt dieses Problem nicht. Man hat eine Idee, die mit ein paar Bauteilen ausprobiert werden könnte, wenn da nicht die vielen Daten- und Adreßleitungen zu verdrahten wären, damit der Mikroprozessor seine Dienste verrichten kann. Das uC-Board (8051) wurde aus diesem Grund entwickelt. Es bietet ein Basissystem, das sich leicht erweitern läßt. Ein weiterer großer Vorteil ist der integrierte Epromemulator. Das Programm, welches von dem Mikroprozessor ausgeführt werden soll, kann bequem über die parallele Schnittstelle von dem PC aus in das Onboard-RAM geladen werden. Ist der Ladevorgang abgeschlossen wird das Programm sofort ausgeführt. Das uC-Board beinhaltet ebenfalls einige Peripheriekomponenten wie z.B.: serielle Schnittstelle, 8Bit Ein/Ausgabe, Adreßdekodeur, LED's & Taster sowie ein LCD zur Ausgabe von Text. Diese Komponenten werden oft benötigt und erleichtern die Softwareentwicklung, da somit sehr einfach Ein- und Ausgaben möglich sind.

## 2. Allgemeines zum uC-Board (8051)

In Bild 2 werden schematisch die Komponenten des Boards gezeigt.

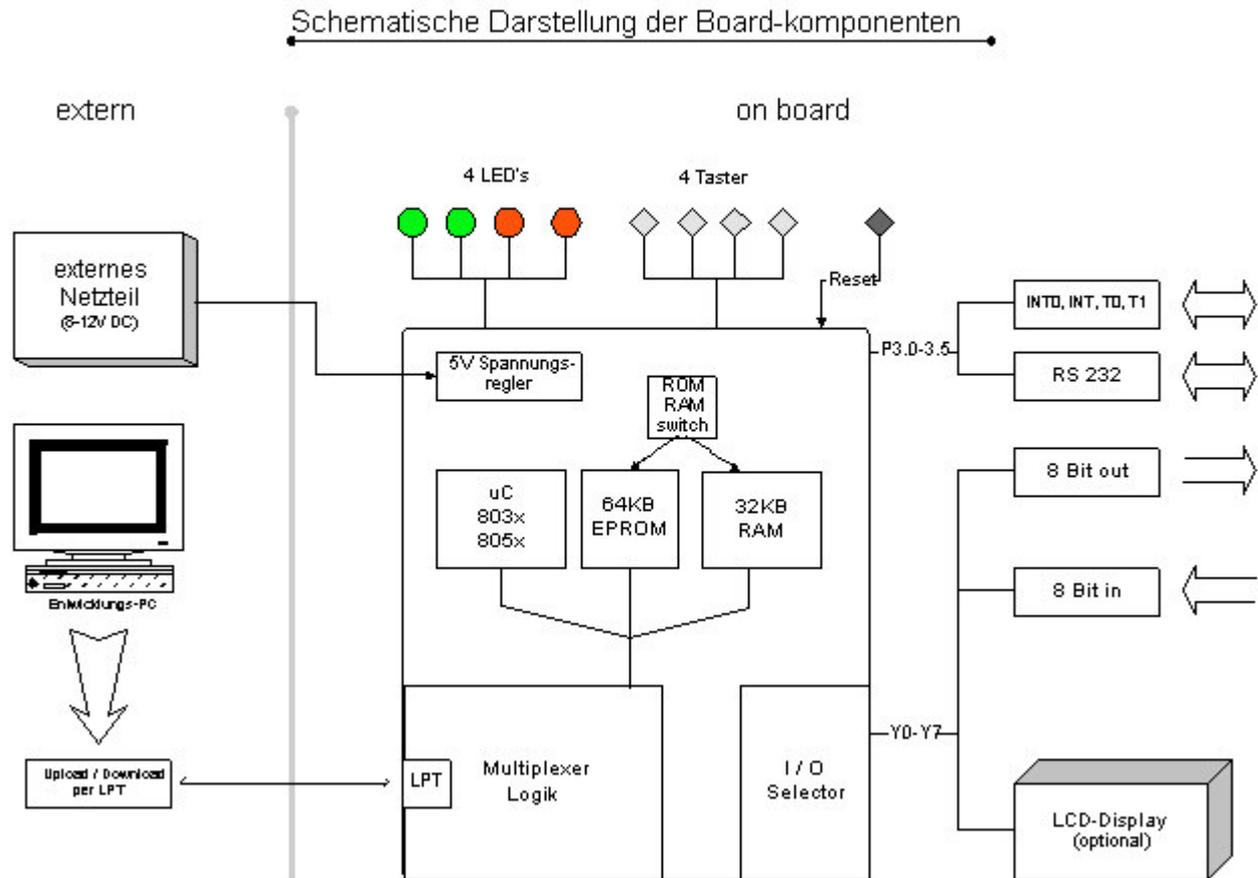


Bild 2

Das Board kann mit einem einfachen Steckernetzteil betrieben werden oder die Versorgungsspannung (5V DC) über den Steckverbinder (BUS) erhalten. Es unterstützt zwei Betriebsmodi. Während der Softwareentwicklung kann der Code im Speicher (RAM) ausgeführt werden. Das Programm wird über die parallele Schnittstelle geladen. Da der RAM ohne Spannung seinen Inhalt verliert, können Programme dauerhaft in einen Festwertspeicher wie dem optionalen Flashmodul gesichert werden. Damit Programme nach einem Reset aus dem Festwertspeicher ausgeführt werden, ist das Board für diesen Betriebsmode zu jumpern. Da das Board hauptsächlich zum Experimentieren eingesetzt werden dürfte, sind einige Schaltkreise steckbar, so daß ein defektes IC leicht gewechselt werden kann. An den Adreßdekoder können an die freien Dekoderausgänge weitere Peripheriebausteine angeschlossen werden. Von den 8 Dekoderausgängen sind 3 für die Peripherie auf dem Board genutzt. Zur Anzeige von Text ist bzw. kann ein handelsübliches LCD-Display angeschlossen werden.

### 3 Was benötigt man zusätzlich

Um das Board nutzen zu können, benötigt man zusätzlich ein handelsübliches Steckernetzteil, einen PC mit einem Cross-Assembler sowie einem 1:1-Kabel für die parallele Schnittstelle. Die Ausgangsspannung des Netzteiles sollte im Bereich von 8-12V Gleichspannung liegen und es sollte mindestens 2 Watt Leistung haben. Will man externe Hardware über den internen 5V Regler mitversorgen, so sollte die Leistung des Netzteiles dementsprechend höher sein. Die Verlustleistung des 5V-Reglers ist dabei zu beachten!

Der Cross-Assembler der Metalink Cororation kann aus dem Web frei bezogen werden oder alternativ kann unter Windows die Entwicklungsumgebung IDE51 genutzt werden.

### 4 Bestückung

Die Platine ist im Euroformat, verzinkt und mit Lötstopplack.  
Die Bestückung sollte deshalb keine Probleme bereiten.

In Bild 3 wird die Unterseite der Platine gezeigt.

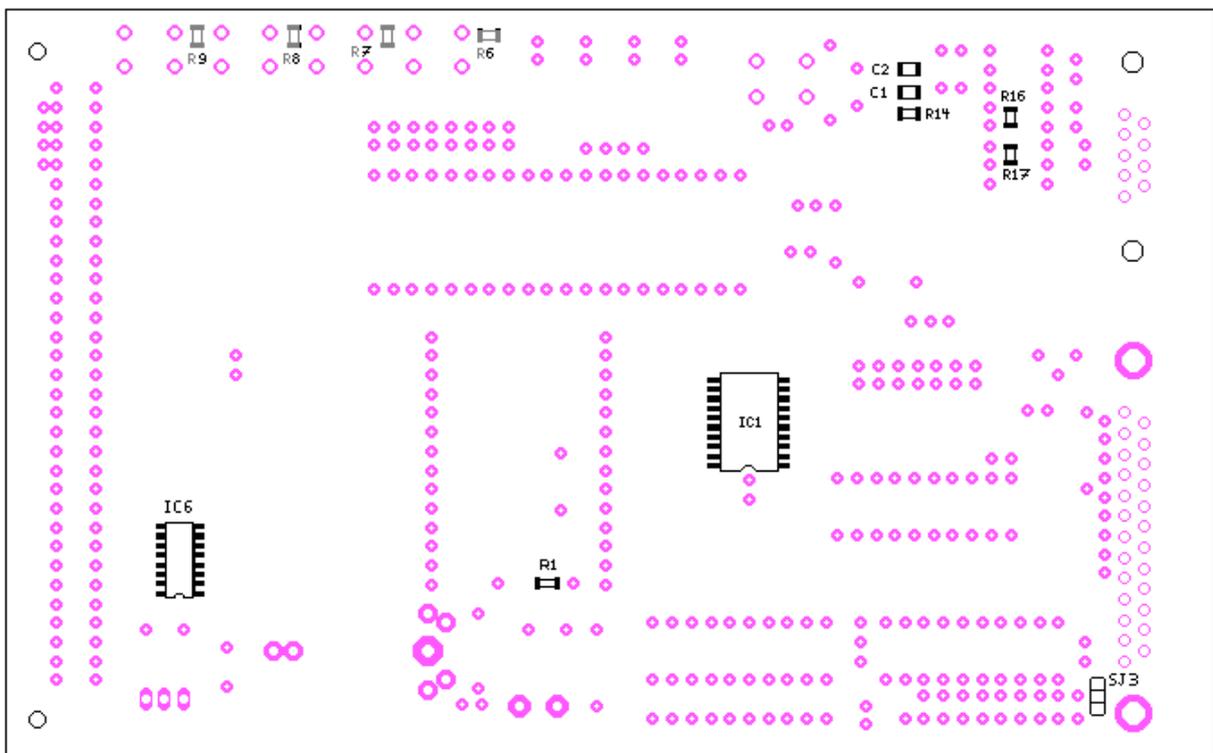


Bild 3

Positionen:

IC1	SMD	74573
IC6	SMD	74138
R1 (future use)	SMD	10K
R6,7,8,9,16,17	SMD	10K
R14 (future use)	SMD	47K
C1,2	SMD	33pF



## 5. Jumpereinstellungen

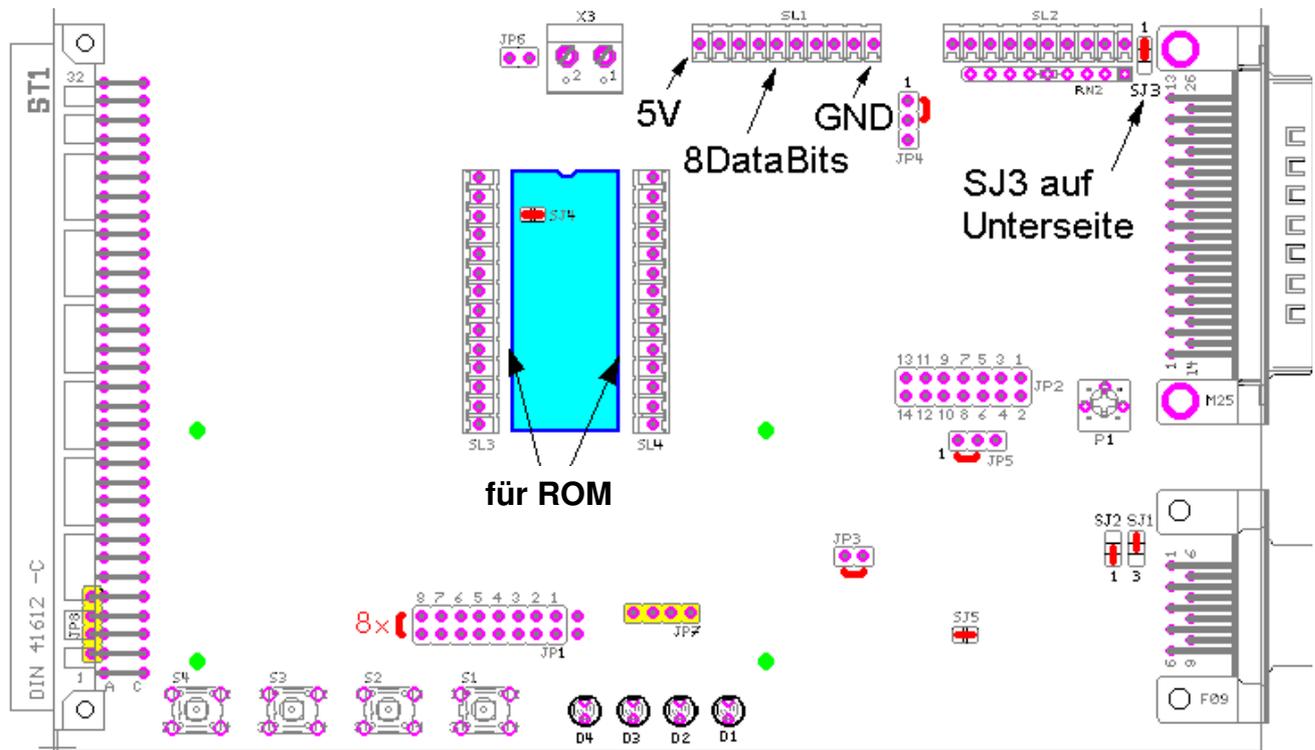


Bild 5

Die roten Markierungen zeigen die Defaulteinstellung der Jumper an.  
SJx sind Lötjumper, welche nur mit Hilfe eines LötKolben geändert werden können.

Kurze Erklärung der Jumper/Anschlüsse:

- JP1: Die 8 Jumper verbinden die LED's/Taster mit dem I/O-Port des Controllers
- JP2: Hier wird das LCD-Modul angeschlossen
- JP3: Leitet das Schreibsignal /WR an den RAM weiter
- JP4: Setzt die Datenrichtung des IC 74245 (1-2 als Input / 2-3 als I/O aber ungepufferter Ausgang)
- JP5: Arbeitsmodus des RAM (1-2 virtueller EPROM / 2-3 normaler RAM-betrieb)
- JP6: future use (Backupstrom G1 zu RAM)
- JP7: Anschluß der Signale INT0, INT1, T0, T1
- JP8: 4 freie Pins an der VG-Leiste ST1 (sinnvoll JP7 mit JP8 elektrisch zu verbinden)
- SJ1: RxD oder TxD (1-2 zu Pin2 / 2-3 zu Pin3 der Sub-D-Buchse F09)
- SJ2: TxD oder RxD (1-2 zu Pin3 / 2-3 zu Pin2 der Sub-D-Buchse F09)
- SJ3: Widerstandsnetzwerk als Pull-Up/Down für Input-default-potential (1-2 Pull-Down /2-3 Pull-Up)
- SJ4: Geschlossen: EPROM-Pin (27) mit A14 verbunden , Offen: EPROM-Pin (27) an Pull-Up (R1)
- SJ5: Leitet externes TxD-signal an Mikrocontroller weiter, wenn geschlossen
- SL1: Gepufferte 8-Bit-Ausgabe mit Betriebsspannung zur Versorgung externer Komponenten
- SL2: Ungepufferte 8-Bit-Eingabe mit Betriebsspannung zur Versorgung externer Komponenten
- SL3,4: Anschluß für einen ROM-Speicher (EPROM z.B. über Adapterplatine)
- X3: Anschluß der Versorgungsspannung / Netzteil (Verpolungsschutz durch Eingangsdiode)

Hinweise:

P1 (Trimpotentiometer) dient zum Einstellen des Kontrastes vom LCD-Mudul.  
Die Jumper SJ1 und SJ2 ermöglichen den Anschluß eines 1:1 oder eines Null-Modem-Kabels je nach Jumperstellung. Sie müssen immer kreuzweise gesetzt sein für den normalen Betrieb.  
Für Testzwecke können die Lötjumper auf gleicher Position auf der Platine gesetzt werden, um einen internen LOOP zu erzeugen. Es darf aber in diesem Modus kein Serielles Kabel angeschlossen werden. Bei einem LOOP werden die gesendeten Daten direkt wieder selbst empfangen.

## 6. Anschlußmöglichkeiten

Über JP1 (siehe Bild5) kann recht einfach externe Hardware angesprochen werden, da der I/O-Port bidirektional und gepuffert ist. Die LED's und Taster sind in die externe Hardware integrierbar, da sie am Pfostenstecker direkt abgegriffen werden können. Die beiden nebenliegenden Durchkontaktierungen sind für die Spannungsversorgung vorgesehen. Das Signal GND ist fast an eine Durchkontaktierung herangeführt und kann leicht mit einer Zinn-Brücke verbunden werden.

Es ist dann nur noch VDD auf die andere Durchkontaktierung zu legen.

Mit einer 20-poligen Pfostenbuchse (18polig gibt es wohl nicht) kann nun die Hardware-Komponente angeschlossen werden und vom uC-Board mitversorgt werden.

In Bild 6 sind zwei Schaltplanbeispiele für den Output-Port (SL1) und den Input-Port (SL2) dargestellt, wobei nicht alle notwendigen Leitungen eingezeichnet sind, sondern nur die erste und letzte Verbindung, um den Sachverhalt übersichtlich darstellen zu können.

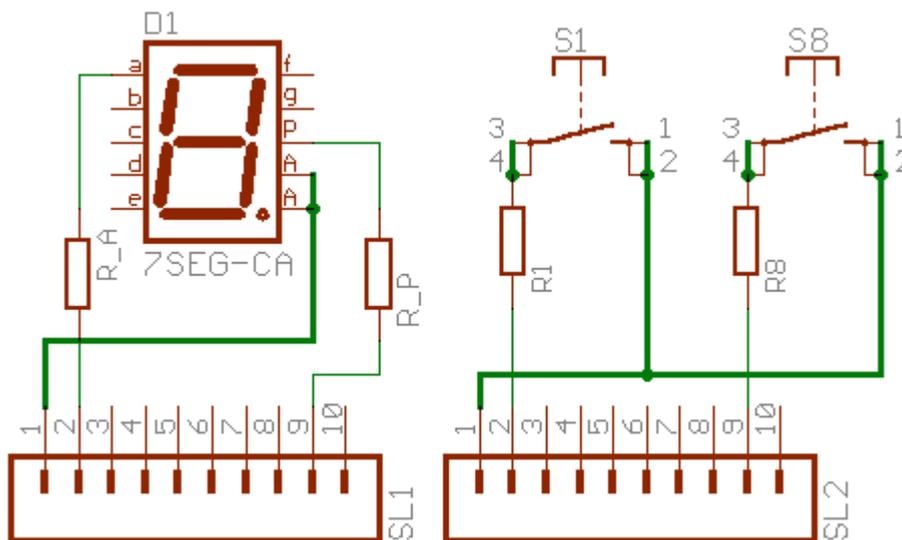


Bild 6

Wenn das Programm für die Hardware fertig programmiert und getestet wurde, so will man evtl. das Programm dauerhaft in einen ROM-Speicher sichern. Bei Einsatz eines EPROMs sind alle Pins des EPROM auf das Rastermaß der Leisten SL3,4 zu bringen. Für EPROM's vom Typ 27256, 27128 und 2764 ist /CE auf Low und ggf. Pin1 auf High zu legen. Weiterhin ist der RAM in den normalen RAM-Modus zu setzen, in dem der Jumper JP5 auf Position 2-3 zu stecken ist. Der Jumper JP3 sollte geschlossen sein, wenn der Schreib-Zugriff zum RAM genutzt wird.

Das LCD-Modul wird an JP2 angeschlossen. Die Pin-Nummern aus Bild 5 entsprechen den Pin-Nummern des LCD-Modules. Die grünen Punkte in Bild 5 stellen die Befestigungslöcher für das LCD-Modul dar. Es ist mit Abstandsbolzen (M2,5) zu befestigen.

Die 64-polige Messerleiste ST1 kann stehend oder liegend eingelötet werden. Wird die stehende Bauform genutzt, so kann die Zusatzplatine mit dem uC-Board in Sandwich-form verbunden werden. Sinnvollerweise wird das uC-Board oben liegen und die Zusatzkarte (mit der Peripherie) wird darunter liegen. Es sind 3 Befestigungslöcher für die Sandwich- bzw. direkte Befestigung vorgesehen.

## 7. Entwicklungssoftware

Das Wichtigste zum Erzeugen des ausführbaren Programmes sind die Tools, die den Quellcode in den Maschinencode übersetzen. Für die 51er-Familie existieren unzählige freie Cross-Assembler. Eleganter und schneller ist die Programmentwicklung mit einem Crosscompiler, wie zB. von der Firma Keil, jedoch sind diese kommerziellen Compiler recht teuer. Die Assembler-Quellfiles für das uC-Board können mit dem Cross-Assembler der "MetaLink Corporation" übersetzt werden. Dieser Assembler ist free, welcher syntax-kompatibel zum Keil-Assembler ist. Der Metalink-Assembler ist in der IDE51 integriert und kann auch separat von der Homepage herunter geladen werden.

Beim Übersetzen des Quellcodes in den Maschinencode fallen in den Übersetzungsschritten je nach Assembler verschiedene Ausgabedatein an.

Die gängigsten sind:

- Objekt-file
- List-file
- Hex-file
- Bin-file bzw. Rom-file.

Für den Up/downloader werden die Bin/Rom-files benötigt, welche den reinen Maschinencode für den Microcontroller enthalten. Der Up/downloader ist für 3 Betriebssysteme verfügbar. Für Dos, Windows und Linux gibt es kommandozeilenorientierte Loader und für Windows/Linux sind auch grafische Varianten verfügbar.

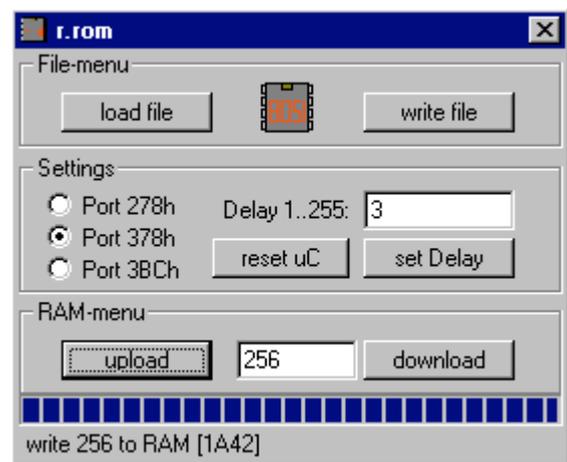
Der kommandozeilenorientierte Loader:

```
C:\asm51\ram51 ↵  
  
use: ram51 binaryfile 1/2 0..10  
or: ram51 r 1/2  
  
-binaryfile: file for upload  
-r          : call reset  
-1/2       : 1=LPT1(378h) / 2=LPT2(278h)  
-0..10     : upload-speed (0=quick..10=slow)
```

Mit den 3 Parametern kann die Datenübertragung gesteuert werden. Es ist gedacht für einen Aufruf aus einer Entwicklungsumgebung.

Der Loader kann auch so aufgerufen werden, daß ein Reset-impulse am Controller erzeugt wird.

Der grafisch orientierte Loader:



Hier wird alles über die Controls gesteuert.

Der Quellcode für den kommandozeilenorientierte Up/downloader (Win/NT) ist unter [www.frydaysoft.de](http://www.frydaysoft.de) abgelegt. Dieser ist unter Visual-C++ erstellt worden. Der Port-zugriff ist in eine DLL gekapselt.

Für das uC-Board wurde (für Win32) eine Entwicklungsumgebung programmiert. Diese Entwicklungsumgebung (IDE51) umfaßt einen Editor, eine Projektverwaltung sowie eingebundene Softwarekomponenten. Diese Komponenten werden durch entsprechend konfigurierte Parameter gesteuert. Diese sind z.B. der Metalink-Assembler sowie der Uploader. Der interne Editor unterstützt Syntaxcoloring sowie eine einfache Hilfe zu den Assemblerbefehlen.

Die aktuellsten Informationen und Programme sind auf der Homepage abgelegt.

[www.frydaysoft.de](http://www.frydaysoft.de)