

# EASY-SIM 51

## 8051/52-Processor-Simulator

**Neben der Simulation eines 8051/52-Mikroprozessors auf einem IBM-kompatiblen PC ermöglicht die in diesem Artikel vorgestellte Software, den vorhandenen Quellcode zu editieren, zu assemblieren und zu disassemblieren.**

### Allgemeines

Die Erfahrung zeigt, daß nach der Erstellung eines Steuerprogramms für einen Mikrocontroller im allgemeinen die volle Funktion nicht auf Anhieb gegeben ist. Je nach Fehlerart kann die dann einsetzende Fehlersuche recht zeitaufwendig ausfallen.

Hier bietet der Prozessor-Simulator EASY-SIM51 seine Dienste an, um auf komfortable Weise Fehler schnell einzukreisen und zu beheben.

EASY-SIM51 ist ein integriertes Paket zur Softwareentwicklung und Simulation für die Controller der 8051-Serie, bestehend aus einem Editor, Assembler, Simulator und Disassembler, die alle in einer komfortablen SAA-Oberfläche vereint sind. EASY-SIM51 erlaubt eine sehr schnelle Erstellung von Applikationen für diese Mikrocontroller, deren Lauffähigkeit sich ohne lange Zwischenphasen austesten läßt.

Die Hardware-Voraussetzungen der unter DOS lauffähigen Software sind denkbar gering:

- IBM-kompatibler PC-AT
- mindestens 640kByte-RAM

- MS-DOS ab Version 3.3
- beliebige Grafikkarte
- Mouse wird empfohlen

Grundsätzlich ist die Software auch über die Tastatur voll bedienbar, wobei eine Mouse besonders auch für den Einsteiger zu empfehlen ist.

### Editor

Der integrierte Wordstar-kompatible Texteditor erlaubt die direkte Bearbeitung des Source-Codes. Neben den bei Wordstar üblichen Steuerbefehlen lassen sich zusätzlich markierte Blöcke mit Hilfe der integrierten Zwischenablage exportieren bzw. importieren. Zusätzlich erlaubt der Texteditor das gezielte Setzen von Haltepunkten (Breakpoints), was die Programmsimulation an der gewünschten Stelle abbricht, um somit eine gezielte Fehlersuche vornehmen zu können.

Als Besonderheit ist hervorzuheben, daß zu jedem Prozessorbefehl eine Online-Hilfe zur Verfügung steht, so daß die Programmierung auch völlig ohne Handbuch möglich ist (Bild 1).

Für den Ausdruck läßt sich über ein spezielles Menü die Formatierung bestimmen. Dafür stehen Druckertreiber für den

Epson-FX- und LQ-Nadeldrucker sowie für den HP-LaserJet-Laserdrucker zur Verfügung. Die physikalische Ausgabe erfolgt auf einer der Parallel-Schnittstellen LPT 1 bis LPT 4, einer der seriellen Schnittstellen COM 1 bis COM 4 oder auf einer frei wählbaren Ausgabedatei. Für die serielle Schnittstelle COM 1 bis COM 4 lassen sich zusätzlich die Übertragungsgeschwindigkeit und die Parameter einstellen.

### Assembler

Der in EASY-SIM51 integrierte Assembler unterstützt den erweiterten Befehlsatz des MASM51-Assemblers von Intel. Er übersetzt den mnemonischen Programmtext (Quelltext) in die vom Prozessor verarbeitbaren Bytefolgen. Es lassen sich symbolische Variablen und Konstanten anlegen. Darüber hinaus sind ein echtes Makroprocessing und eine bedingte Übersetzung möglich. Natürlich ist der erzeugte Code auch für alle kompatiblen Mikrocontroller verschiedener Hersteller tauglich.

Der 2-Paß-Assembler durchläuft den Source-Code zweimal. Beim ersten Durchlauf werden alle angelegten Symbole und Makros gesammelt und in eine sogenannte Symboltabelle eingetragen, während im

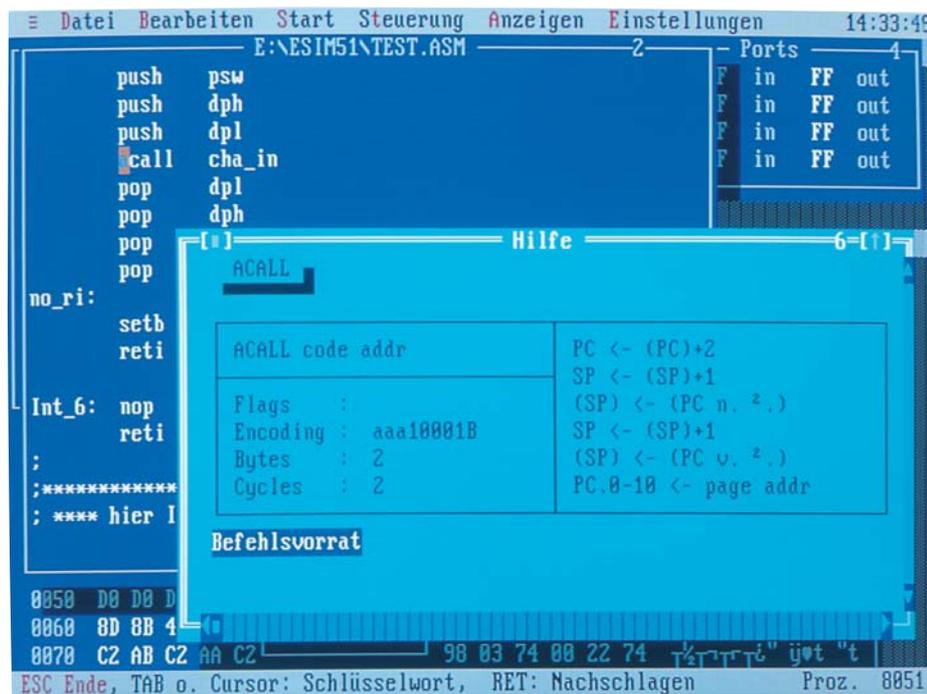


Bild 1: Zu jedem Prozessorbefehl steht eine Online-Hilfe zur Verfügung.

zweiten Durchlauf die ermittelten Werte für die Symbole in den generierten Objekt-Code eingesetzt werden.

Neben den Makro-Funktionen erlaubt der Assembler den Zugriff auf die unterschiedlichen Segment-Bereiche des 8051-Prozessorkerns (CODE-, DATA-, IDATA-, XDATA- und BIT-Segment). Die Speicherung der generierten Daten kann wahlweise im Binär- oder Intel-Hex-Format erfolgen.

## Simulator

Hauptbestandteil von EASY-SIM51 ist der 8031/32/51/52-Prozessor-Simulator, der über umfassende symbolische Debug-Möglichkeiten bei schneller Abarbeitung des 805X-Codes verfügt.

EASY-SIM51 benutzt für die Simulation einen virtuellen Speicher im System, der sich sowohl vom Prozessor als auch vom Benutzer selbst verändern läßt. Eine Veränderung durch den Benutzer wirkt sich sofort auf die momentane Arbeit des Prozessors aus. Für die Simulation des 8031/51-Prozessors stehen folgende Funktionen zur Verfügung:

- Bit- und Bytebefehle
- 32 bidirektional und individuell adressierbare I/O-Lines
- 128Byte-RAM auf dem Chip
- zwei 16-Bit-Zähler/Timer
- ein Full-Duplex-UART
- 4kByte-Programmspeicher auf dem Chip
- 64 kByte adressierbarer Programmspeicher maximal
- 64kByte adressierbarer Datenspeicher.

Für die 8032/52-Prozessoren bietet die Software folgende Verbesserungen:

- 256Byte-RAM auf dem Chip
- 3 Zähler/Timer
- 8kByte-Programmspeicher auf dem Chip

Die Anzeige der Simulationsergebnisse kann für jeden Bereich (Data, Register, Flags, Ports, ...) erfolgen, wobei jeweils ein Fenster, welches sich je nach Bedarf öffnen läßt, zur Anzeige vorgesehen ist.

EASY-SIM51 kennt mehrere Arten der Programmausführung.

Unter dem Menüpunkt „Ausführen“ arbeitet der Simulator mit der maximalen Geschwindigkeit. Dabei werden die Inhalte der Fenster nicht sofort aktualisiert, sondern erst nach Ende des Simulationslaufes, der durch die „Esc“- oder Leertaste oder einen Haltepunkt erzwungen wird. Dieser Betriebsmode ist vorwiegend im Zusammenhang mit Breakpoints interessant.

Eine weitere Möglichkeit ist die „Ausführung bis Cursor“-Simulation, die bis zum aktuellen Cursorstand im Assembler-Fenster ausgeführt wird. Sollen während der Simulation die Fensterinhalte sofort nach Veränderung aktualisiert werden, so ist die Programmsimulation über „langsam ausführen“ zu starten. Dieser Mode ist bedingt durch die ständige Fensteraktualisierung wesentlich langsamer in seiner Ausführung als die vorgenannten Simulationsmodi.

Bei Ausführung der Simulation in Einzelschritten wird jeder Befehl einzeln abgearbeitet, womit eine genaue Kontrolle der ausgeführten Aktivität möglich ist. Ist die Anweisung ein Sprung zu einer Unter-

routine, wird zu dieser verzweigt und auch hier die Abarbeitung der Befehle einzeln vorgenommen.

Damit nun bereits getestete Unterprogramme nicht jedesmal im Einzelschritt-Mode durchgegangen werden müssen, stellt EASY-SIM51 die Möglichkeit zur Verfügung, während der Einzelschritt-Simulationen jeweils die Unterprogramme ohne Unterbrechung ausführen zu lassen.

Die einzelnen Simulations-Ausgabefenster wollen wir nachfolgend näher beschreiben.

### DATA

Im DATA-Fenster sind die Hauptregister des Prozessors untergebracht. Dazu gehören der Programmzähler InstP, der Datenzeiger DPTR, der Akkumulator ACC, das B-Register B und das Prozessor-Statuswort PSW.

### Timer 1

Im Timer-Fenster sind die Interrupt- und Timer-Kontroll-Register IE, IP, TMODE und TCON sowie die beiden 8-Bit-Zähler der beiden Timer 0 und 1 (TH 0, TL 0, TH 1, TL 1) enthalten.

### Timer 2

Für die Simulation des 8032/52-Prozessors ist ein weiteres Fenster vorgesehen, welches die Inhalte der für Timer 2 enthaltenen zusätzlichen Register darstellt (ET 2, PT 2, T2CON, TL2, TH2, RCAP2L und RCAP2H).

### Serial

Das Serial-Fenster zeigt den aktuellen Zustand des Send- und Empfangsbuffers SBUF sowie die seriellen Steuerregister SCON und PCON an, während ein weiteres Fenster die über die serielle Schnittstelle übertragenen Daten anzeigt.

### IDATA

Das IDATA-Fenster stellt den Inhalt der ersten 128 bzw. 256 Byte des internen RAMs des 8051-Mikrocontrollers dar. Dieser ist mit Hilfe der Tastatur über die ASCII- oder HEX-Eingabe direkt veränderbar, wodurch sich beispielsweise Schleifenzähler oder ähnliches direkt beeinflussen lassen. Die sich daraus ergebenden Änderungen werden in allen betroffenen Fenstern sofort aktualisiert.

### XDATA

Ebenso wie der IDATA-Bereich läßt sich der bis zu 65536 Byte umfassende XDATA-Bereich anzeigen und frei editieren, wodurch die Überwachung dieses Bereiches, in dem häufig ein externes RAM untergebracht ist, sehr einfach möglich ist.

### CODE

In diesem Fenster ist der abzuarbeitende Programmcode dargestellt. Üblicherweise wird der Inhalt durch den Assembler oder die geladene Binär-Datei überschrieben. Auch hier lassen sich nachträglich beliebige Änderungen vornehmen.



**Register**

Das Register-Fenster zeigt den Inhalt der Register R 0 bis R 7 und Register B. Eine Änderung dieser Registerinhalte wird sofort auch in allen anderen betroffenen Fenstern aktualisiert.

**Stack**

Wie die Register liegt auch der Stack-Bereich in den unteren 128 Byte des prozessorinternen RAMs. Das Stack-Fenster zeigt den aktuellen Inhalt des Stack-Poin- ters an sowie die letzten 8 Byte des Stack- Inhaltes.

**Stimulus-Funktion**

Das Steuerprogramm eines Mikroprozessors kommuniziert immer über die Port-Pins mit seiner Umwelt. Je nach Peripherie können sich die Zustände der Prozessors in bestimmten Zeitabständen verändern, worauf der Prozessor (Programm) dann zu reagieren hat.

Für die Simulation der Port-Veränderungen stellt EASY-SIM51 die Stimulus-Funktion zur Verfügung. Die Stimulus-

Dateien enthalten Port-Eingangssignale, die während der Simulation die Port-Aktivitäten steuern und damit die Hardwareumgebung des Mikrocontrollers nachbilden. Mittels des integrierten Editors kann die Stimulus-Datei erstellt bzw. editiert werden.

Abbildung 2 zeigt ein Beispiel für den Aufbau dieser Datei, die zeilenweise synchron zu den Mikrocontroller-Befehlen abgearbeitet wird. Pro Befehl wird dabei eine Zeile in der Stimulus-Datei, die auch mehrere Befehle enthalten kann, abgearbeitet.

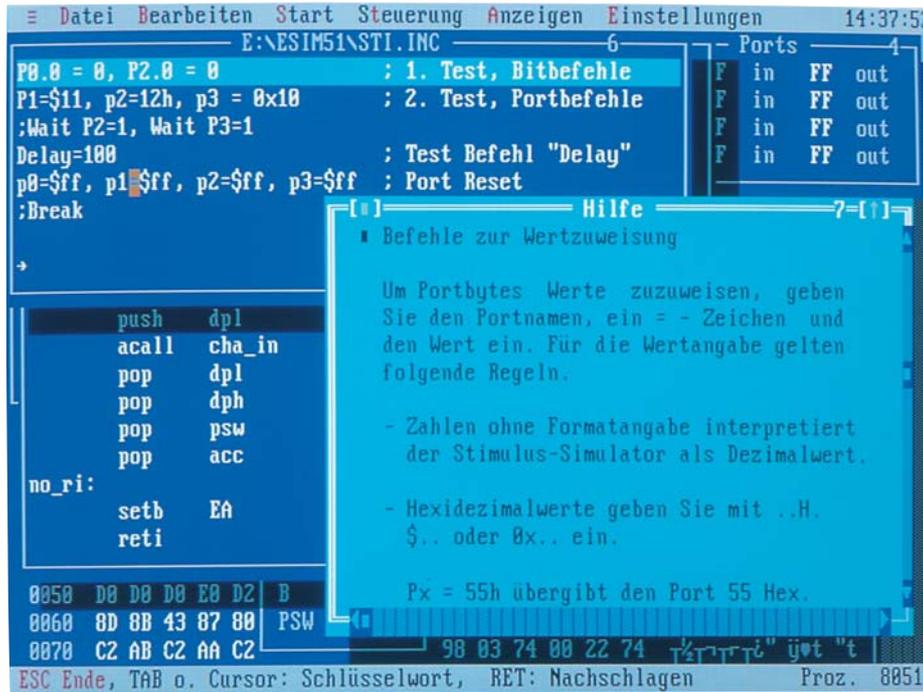
Die Stimulus-Funktion erlaubt die Beeinflussung einzelner Port-Bits aber auch kompletter 8-Bit-Ports. Ferner kann mit Hilfe des Wait-Befehls auf einen bestimmten Port-Zustand gewartet werden. Der Delay-Befehl gibt die Anzahl der Prozessorzyklen an, die der Simulator abzuwarten hat.

**Besonderheiten**

Neben der Verwendung von Symbol-Bezeichnungen erlaubt der Simulator die individuelle Farbgestaltung sowie die Verwendung des 24- bzw. 43/50-Zeilen-Textmodus, der eine sehr übersichtliche Darstellung der gesamten Anzeigen erlaubt (Abbildung 3).

Alles in allem bietet EASY-SIM51 ein komplettes Entwicklungssystem (Editor, Assembler, Simulator) für die MSC51-Familie, das sowohl für den Anfänger als auch für den Profi nützliche Dienste leistet.

ELV



**Bild 2: Aufbau einer Stimulus-Datei, die zeilenweise synchron zu den Mikrocontroller-Befehlen abgearbeitet wird.**

**Ports**

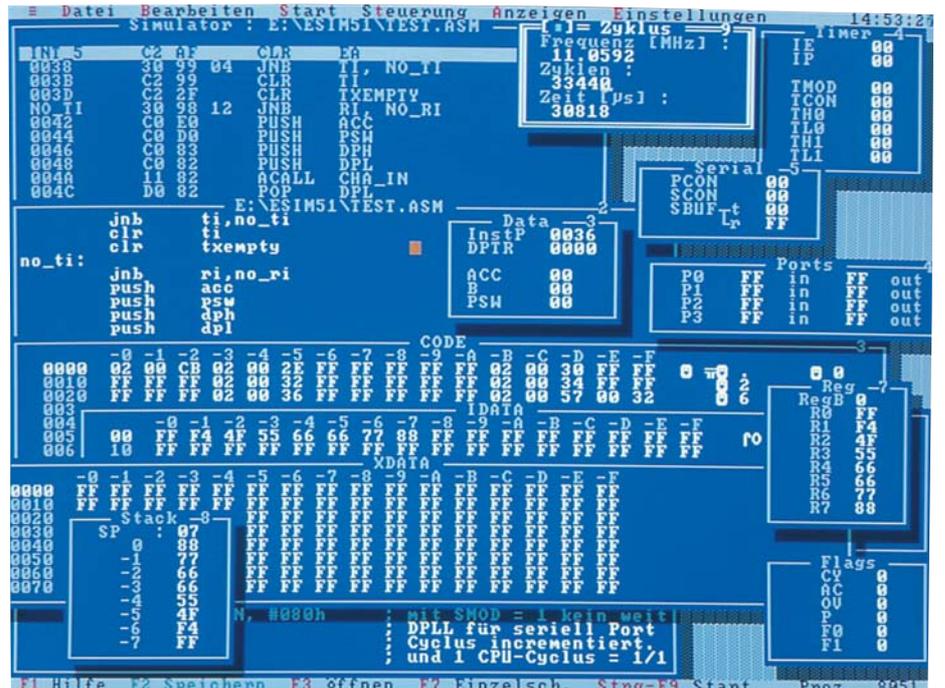
Im Port-Fenster sind die direkt adressierbaren Register der Ports P0 bis P3 zu sehen, wobei sich jeweils die getrennt dargestellten Input- und Output-Register editieren lassen.

**Flags**

In diesem Fenster lassen sich die aktuellen Zustände der Prozessor-Flags CY, AC, OV, P1, F0 und F1 darstellen und verändern. Besonders bei der Überprüfung von Arithmetik-Routinen ist die Auswertung der Prozessor-Flags unerlässlich.

**Zyklus**

Im Zyklus-Fenster wird zunächst die Oszillatorfrequenz eingestellt. Der Simulator inkrementiert mit fortlaufender Zeit den Zyklenzähler, woraus in diesem Fenster die Laufzeit-Anzeige entsprechend aktualisiert wird. Damit läßt sich sehr exakt das Laufzeitverhalten von Programmen bzw. Programmteilen ermitteln.



**Bild 3: Die Verwendung des 43/50-Zeilen-Textmodus erlaubt eine übersichtliche Darstellung der ausgewählten Fenster.**