

I²C-EEPROM-Board

Die dauerhafte Speicherung von Daten, die auch nach dem Abschalten der Betriebsspannung erhalten bleiben, ermöglicht diese mit einem I²C-EEPROM bestückte Schaltung. Zum Datenaustausch mit diesem I²C-EEPROM-Board ist z. B. das I²C-PC-Interface aus dem „ELVjournal“ 4/96 geeignet.

Allgemeines

Zahlreiche Anwendungen der Elektronik erfordern die Datenspeicherung auch nach Ausschalten des Gerätes.

So muß z. B. ein Fernsehgerät oder ein Videorecorder nach dem Ausschalten die Sendereinstellungen speichern, und in einigen Meßgeräten müssen Abgleichparameter dauerhaft verfügbar sein.

Oft wird dazu ein RAM eingesetzt, das durch einen Akku oder neuerdings einen Gold-Cap gepuffert ist. Diese Version hat den Vorteil, daß der Akku auch zur Versorgung anderer Systemkomponenten verwendet werden kann, um so z. B. eine interne Uhr zu versorgen.

Wenn jedoch nur Daten gespeichert werden sollen, so kann diese Aufgabe am einfachsten durch ein EEPROM realisiert werden, das eine Reihe von Vorteilen gegenüber einem RAM mit Akku bietet. Man denke nur einmal an die Folgen eines leeren Akkus...

Ein EEPROM bietet dabei entscheidende Vorteile, da ein Gerät mit Akkupufferung zwischenzeitlich eingeschaltet wer-

den muß, um den Akku wieder aufzuladen. Dazu ist die Lebensdauer eines Akkus begrenzt, so daß dieser meist nach einigen Jahren ausgetauscht werden muß.

Bei den am häufigsten verwendeten EEPROMs handelt es sich um Bausteine, die seriell beschrieben und ausgelesen werden können, da diese nur über wenige Anschlußpins verfügen und in sehr kleinen Gehäusen untergebracht sind.

Tabelle 1: Technische Daten

Speicher:	
24C02	1 x 256 Bytes
24C04	2 x 256 Bytes
24C08	4 x 256 Bytes
24C16	8 x 256 Bytes
Spannungsversorgung:	2,5 V bis 6,0 V
Stromaufnahme:	
10 µA	im Ruhezustand
400 µA	im normalen Betrieb
1 mA	während der Programmierung
Programmierzeit:	10 ms
Datensicherheit:	min. 10 Jahre
Programmierzyklen:	
	min. 500.000 pro Speicherstelle

Zur Datenübertragung zwischen einem steuernden Mikrocontroller und dem EEPROM bietet sich das I²C-Protokoll an, bei dem nur zwei bidirektionale Signalleitungen des Controllers verwendet werden müssen. Ebenfalls gibt es eine Vielzahl von I²C-Bausteinen mit anderen Funktionen, die dann einfach mit an den I²C-Bus angeschlossen werden können.

I²C-EEPROM

Die Standard-I²C-EEPROMs sind in einem 8poligen DIP-Gehäuse untergebracht und werden über eine 5V-Betriebsspannung versorgt. Die Erzeugung der Programmierspannung erfolgt intern und erfordert somit keine zusätzliche Spannungsversorgung.

Einige I²C-EEPROMs müssen extern mit einer RC-Kombination beschaltet werden, die zur Erzeugung des Programmierimpulses erforderlich ist.

Die hier vorgestellten Typen erzeugen hingegen den Programmierimpuls intern und kommen ohne externe Bauteile aus. Das prinzipielle Anschlußschema ist im Schaltbild (Abbildung 4) dargestellt. Die

technischen Daten sind in der Tabelle 1 aufgeführt.

Der Speicher der EEPROMs ist in 8Bit-Worten organisiert, die in Bänken zu jeweils 256 Bytes angeordnet sind. Der Typ 24C02 verfügt dabei über einen Speicherinhalt von 1 x 256 x 1Byte, der 24C04 über 2 x 256 x 1Byte, der 24C08 über 4 x 256 x 1 Byte und der 24C16 verfügt über 8 x 256 x 1 Byte Speicher.

Wie alle I²C-Bus-Bausteine wird auch das EEPROM über eine Adresse auf dem Bus angesprochen, die in Abbildung 1 dargestellt ist. Die Adresse besteht dabei aus 4 fest vorgegebenen Bits und 3 Bits, die über die Adreßleitungen A 1 bis A 3 des EEPROMs definiert werden.

Das 8. Bit bestimmt, wie im I²C-Protokoll üblich, ob Daten in das EEPROM geschrieben oder gelesen werden sollen. Um mehrere EEPROMs an einen Bus anzuschließen, sind die Adreßleitungen unterschiedlich einzustellen, um die Bausteine einzeln ansprechen zu können.

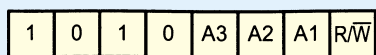


Bild 1: SLAVE-Adresse des EEPROMs

Die Adressierung des Speicherinhaltes erfolgt über ein 8Bit-Wort, zum Ansprechen von 256 Speicherstellen (eine Bank).

Beim Einsatz eines 24C04 ist zu beachten, daß dieser über zwei Speicherbänke verfügt, die über das Bit A 1 bei der Adres-

sierung auszuwählen sind. Ebenso werden die 8 Speicherbänke des 24C16 über die Bits A 1 bis A 3 bestimmt. Zu beachten ist hierbei, daß diese Adreßleitungen des Chips unbedingt stets auf Low-Potential liegen müssen.

Das Protokoll zum Schreiben in das EEPROM ist in Abbildung 2 dargestellt. Die Übertragung beginnt, indem der MASTER die Start-Bedingung generiert und anschließend die Bausteinadresse überträgt, wobei der SLAVE jedes übertragene Byte bestätigt. Danach folgt die Adresse der anzusprechenden Speicherstelle im EEPROM und die zu schreibenden Daten. Dabei wird der Adreßzähler automatisch weitersetzt und die Bytes der Reihe nach im Speicher abgelegt. Der MASTER beendet die Übertragung, indem er eine Stopp-Bedingung generiert. Als Reaktion darauf programmiert das EEPROM die Daten und ist für ca. 10 ms nicht mehr ansprechbar.

Eine Besonderheit ist zu beachten, wenn mehr als ein Byte in das EEPROM zu schreiben ist. Bei dem automatischen Weitersetzen des Adreßzählers werden nur die unteren 3 Bits weitergezählt, und die oberen 5 Bit bleiben unverändert. Somit sind immer nur maximal 8 Bytes auf einmal übertragbar, wobei die Anfangsadresse dann bei 0, 8, 16, ... liegen muß.

Will man zum Beispiel ab Adresse 5 schreiben, so dürfen nur 3 Bytes übertragen werden, da es beim 4. Byte zu einem Überlauf des Adreßzählers kommt und das Byte an Adresse 0 geschrieben würde.

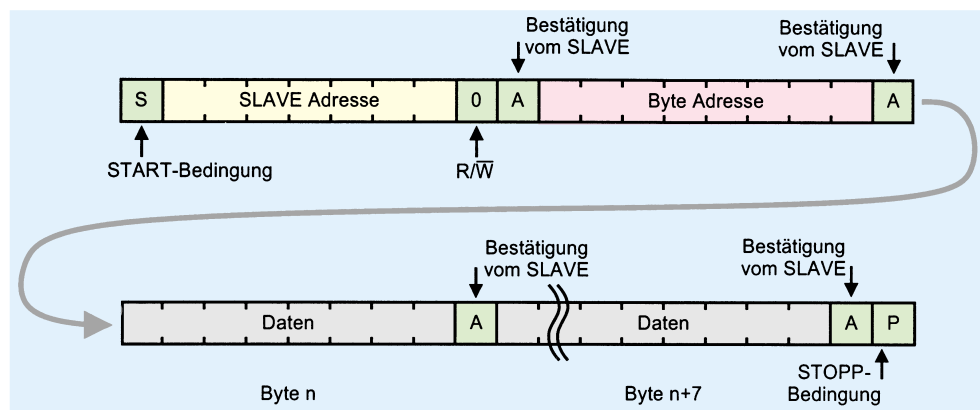


Bild 2 (links): I²C-Protokoll zum Schreiben in das EEPROM

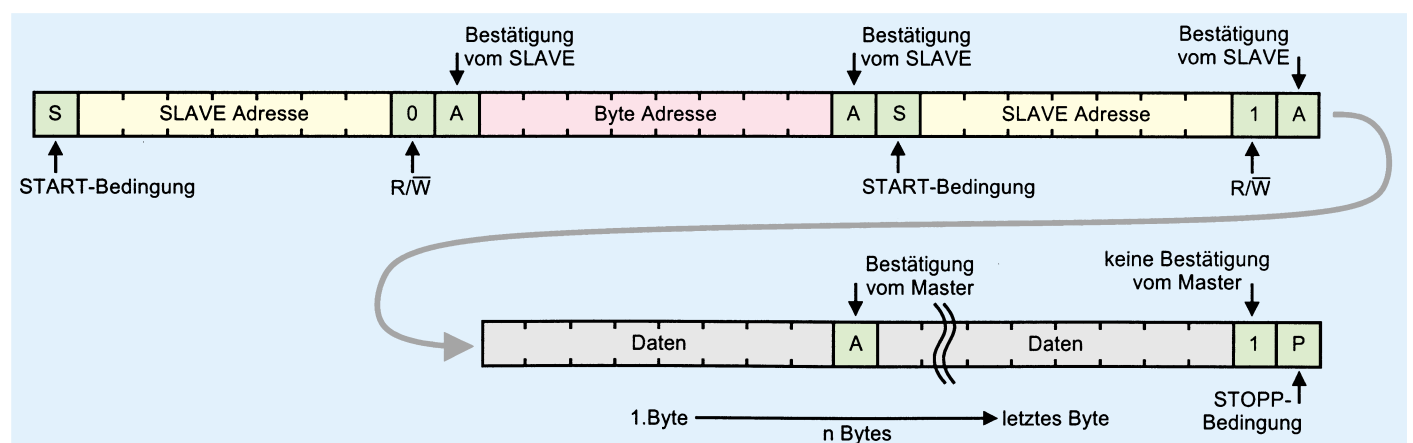


Bild 3 (unten): I²C-Protokoll zum Lesen aus dem EEPROM

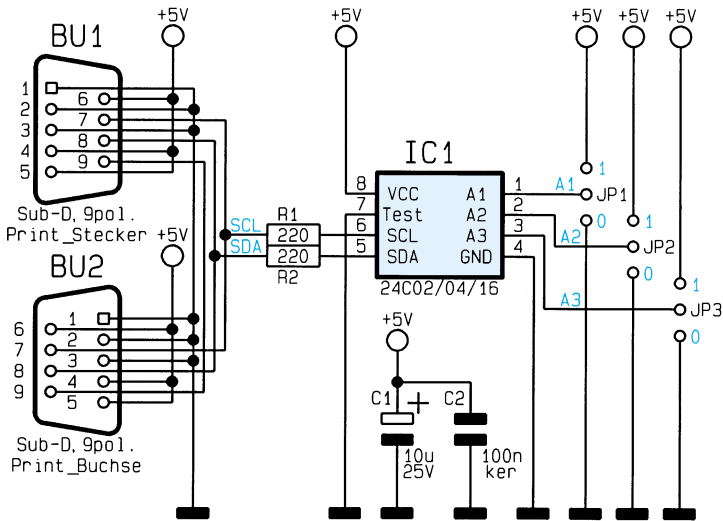


Bild 4: Schaltbild des I²C-EEPROM-Boards

stens 100 Millionen Schreibzyklen pro Speicherstelle garantieren, ist der Baustein auch als Quasi-RAM einsetzbar, um z. B. Daten zwischenspeichern.

Schaltung

Die Schaltung des I²C-EEPROM-Boards ist in Abbildung 4 dargestellt. Die Verbindung zum I²C-PC-Interface erfolgt über die Buchse BU 2 und den Stecker BU 1 (beide parallelgeschaltet), die das Board mit der 5V-Betriebsspannung versorgen und die SCL- und SDA-Leitung mit sich führen. Die Betriebsspannung wird über die Kondensatoren C 1 und C 2 gepuffert und die I²C-Leitungen sind über die Widerstände R 1 und R 2 auf das IC 1 geschaltet.

Zur Auswahl der Bausteinadresse dienen die Jumper JP 1 bis JP 3, die entsprechend des eingesetzten Bausteins auf die Stellung „1“ oder „0“ zu stecken sind und die Adreßleitungen mit „high“ oder „low“ beschalten.

**Stückliste:
I²C-EEPROM-Board**

Widerstände:

220Ω..... R1, R2

Kondensatoren:

100nF/ker C2
10µF/25V C1

Halbleiter:

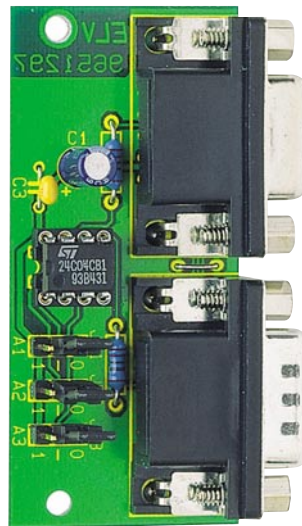
24C04 IC1

Sonstiges:

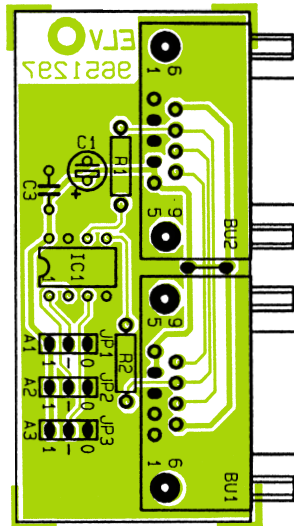
- SUB-D-Stiftleiste, 9polig, print BU1
- SUB-D-Buchsenleiste, 9polig, print BU2
- Stiftleisten, 1 x 3polig JP1-JP3
- 3 Jumper
- 1 IC-Fassung, 8polig
- 1 I²C-EEPROM-Treiberdiskette

Nachbau

Der Nachbau des I²C-EEPROM-Boards gestaltet sich sehr einfach, da die Schaltung nur aus wenigen Bauteilen besteht. Die Bestückung erfolgt in gewohnter Wei-



Ansicht der fertig bestückten Leiterplatte



Bestückungsplan des I²C-EEPROM-Board

se anhand der Stückliste und des Bestückungsdruckes, wobei zuerst die niedrigen, gefolgt von den höheren Bauteilen bestückt und verlötet werden.

Für das EEPROM IC 1 ist ein 8poliger IC-Sockel vorgesehen, damit das IC leicht austauschbar ist. Für das IC 1 ist im Bausatz ein 24C04 enthalten, das über einen 2x256 Byte-Speicher verfügt. Für diesen Baustein ist es unbedingt erforderlich, daß der Jumper JP 1 auf die Stellung „0“ gesteckt wird, da der Baustein bereits zwei Speicherbänke beinhaltet. Ebenso sind beim Einsatz eines 24C16 zusätzlich die Jumper JP 2 und JP 3 auf „0“ zu stecken, da dieser Chip insgesamt 8 Speicherbänke beinhaltet.

Software

Die Grundroutinen zur Ansteuerung von I²C-Bausteinen liegen als Quellcode in den Programmiersprachen C und PASCAL vor. Zusätzlich zeigt ein kleines Demoprogramm die Anwendung der Routinen im Zusammenhang mit dem I²C-PC-Interface.

Die Funktion „EEPROM_LESE_BYTE (Adresse, Bank, Daten)“ liest ein Byte aus dem EEPROM an der vorgegebenen Adresse und Speicherbank, das dann als Funktionswert zurückgegeben wird. Ebenso arbeitet die Funktion „EEPROM_LESE_WORD (Adresse, Bank, Daten)“, die ein Wort (2 Bytes) aus dem EEPROM liest, wobei hier zu beachten ist, daß zwei aufeinanderfolgende Adressen aus dem EEPROM gelesen werden.

Die weiteren vorhandenen Funktionen „EEPROM_SCHREIBE_BYTE (Adresse, Bank, Daten)“ sowie „EEPROM_SCHREIBE_WORD (Adresse, Bank, Daten)“ dienen zum Schreiben in das EEPROM, wobei die zu schreibenden Daten, die Adresse und die Speicherbank übergeben werden.

Da das EEPROM nach einer Schreiboperation zuerst die Daten programmieren muß, ist das EEPROM während dieser Zeit nicht neu ansprechbar. Es bestätigt dabei eine neue Adressierung nicht. Die Software-routinen warten kurz, versuchen dann erneut das EEPROM zu adressieren, dies solange, bis es den Programmiervorgang abgeschlossen hat und die Adressierung bestätigt.

Selbstverständlich ist es auch möglich, das EEPROM-Board an andere Komponenten anzuschließen und zu steuern. So kann es zum Beispiel auch an das ELV-Mikrocontroller-Testboard angeschlossen und vom Controller gesteuert werden. Dazu sei auf den Artikel „Mikrocontroller-Grundlagen“ verwiesen, der in dieser Ausgabe die Ansteuerung eines EEPROMs in Assembler erläutert.