



I²C-LED-Board

Die vom steuernden PC räumlich abgesetzte Konfiguration von I²C-Applikationen legt eine ebenso abgesetzte Anzeige von Daten, z. B. Meßwerten und Uhrzeiten, nahe. Unser Artikel zeigt eine Lösung für eine vierstellige 7-Segment-LED-Anzeige, die beliebig am I²C-Bus betrieben werden kann. Bis zu vier dieser Anzeige-Boards sind über das ELV-I²C-PC-Interface ansteuerbar.

Allgemeines

Mit den bisher vorgestellten ELV-I²C-Projekten haben wir bereits interessante Ein- und Ausgabeapplikationen wie eine Echtzeituhr und ein AD-DA-Wandlerboard kennengelernt.

Durch eine der wesentlichen Funktionen des I²C-Buskonzeptes, über Zweidrahtleitungen weit abgesetzte Einzelkomponenten betreiben zu können, ergibt sich als sinnvolle Ergänzung des bisher vorgestellten Bausteinsystems ein vom zentral steuernden Mikrocontroller ebenfalls abgesetztes Display für Meßwerte, Daten und Zeitanzeigen.

Eine komplette Lösung für eine solche Baugruppe stellt die Anwendung des I²C-LED-Anzeigetreibers SAA1064 dar. Er ermöglicht, getreu unserem Credo vom möglichst geringen Bauelementeaufwand der I²C-Bausteine, mit sehr sparsamer

Schaltungsumgebung und als kompakte Lösung die Ausgabe von Daten über den Bus auf bis zu vier 7-Segment-LED-Anzeigen.

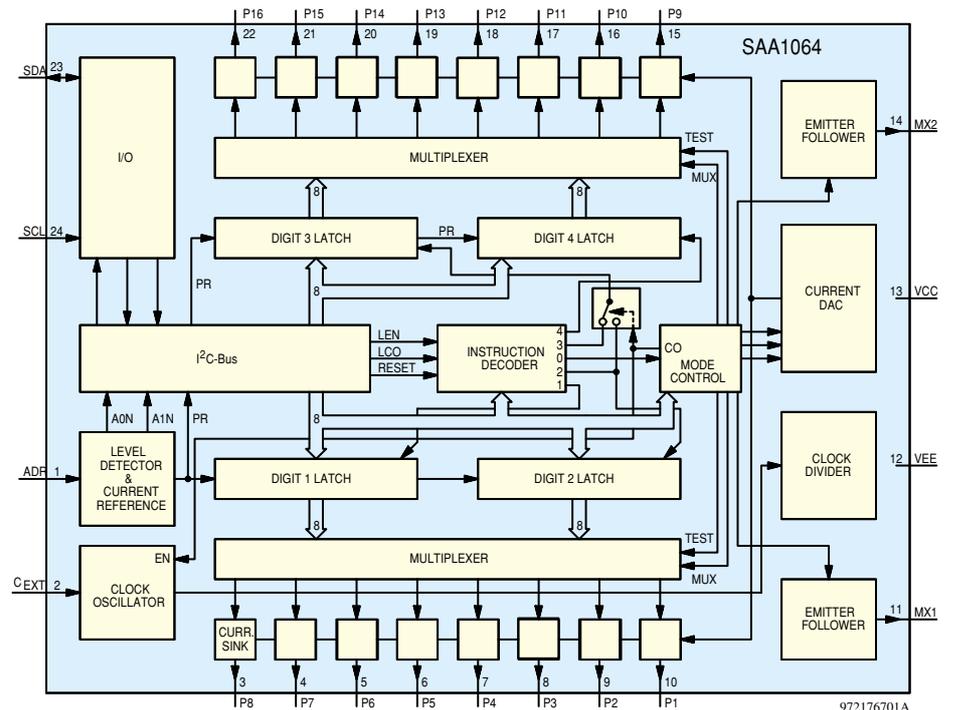


Bild 1: Blockschaltbild des SAA1064

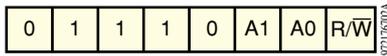


Bild 2: SLAVE-Adresse des SAA1064

Dabei sind alle LED-Segmente einzeln ansteuerbar, so daß die Anzeigen im Rahmen der Möglichkeiten einer 7-Segment-Anzeige beliebige Daten ausgeben können.

Bis zu vier dieser Display-Boards sind gleichzeitig am I²C-Bus einsetzbar. Die Platine ist für den Anschluß des ELV-I²C-PC-Interface ausgelegt, kann aber universell an jedem Mikrocontroller mit I²C-Bus-Interface betrieben werden.

SAA 1064 - intern

Hauptbestandteil der Schaltung ist das speziell zur Ansteuerung von LED-Anzeigen konzipierte IC vom Typ SAA 1064, dessen Blockschaltbild in Abbildung 1 dargestellt ist. Die Anschlußbelegung des Schaltkreises ist dem Schaltbild (Abbildung 9) zu entnehmen.

Das in einem 24poligen Gehäuse untergebrachte IC weist folgende Merkmale auf:

- I²C-Bus-Interface
- Ansteuerung von bis zu vier 7-Segment-Anzeigen oder 32 einzelnen LEDs
- Softwaremäßige Einstellung des Segmentstroms (0 bis 21mA in 3 mA-Schritten)

Tabelle 1: Adreßauswahl des SAA1064

Adreßbits		Jumper		
A1	A2	J2	J3	J4
0	0	0	0	1
0	1	0	0	0
1	0	0	1	0
1	1	1	0	0

0 = Jumper offen
1 = Jumper geschlossen

Jeder der 16 Ausgänge verfügt über eine Stromsenke. Alle Stromsenken sind gemeinsam über einen Softwarebefehl programmierbar (LED-Strom von 0 bis 21 mA, in 3 mA-Schritten einzustellen). Hierdurch entfallen die sonst üblichen Vorwiderstände für die LEDs, und es kann jederzeit eine Anpassung der LED-Helligkeit an die Um-

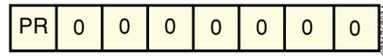


Bild 3: STATUS-Byte des SAA1064

gebungshelligkeit vorgenommen werden.

Anstelle der 7-Segment-Anzeigen können aber auch einzelne LEDs zum Einsatz kommen, da der SAA 1064 die Programmierung jedes Ausgangs und somit jedes einzelnen Segments zuläßt. Damit ist eine große Vielfalt an Anzeigemöglichkeiten eröffnet.

Der SAA 1064 kennt zwei Betriebsarten:

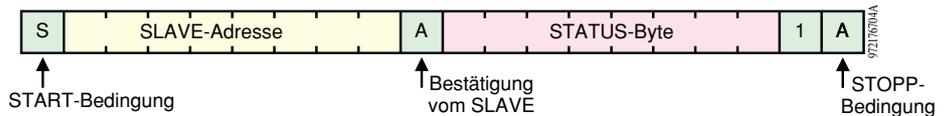


Bild 4: Lesen des STATUS-Byte

- 1. Static Mode:** In dieser Betriebsart stehen 16 statische Ausgänge zur Verfügung, somit sind zwei 7-Segment-Anzeigen mit Dezimalpunkt ansteuerbar.
- 2. Dynamic Mode:** Dieser Modus realisiert den Multiplexbetrieb, wodurch sich die Anzahl der Ausgänge von 16 auf 32 verdoppelt. Somit können vier 7-Segment-Anzeigen angesteuert werden. Lediglich ein externer Kondensator an Pin 2 (C_{ext}) bestimmt den Wert der Multiplexfrequenz.

Die Kommunikation des SAA 1064 mit dem I²C-Bus erfolgt über das I²C-Bus-Interface des ICs (Pin 23 und Pin 24), wobei der Baustein durch seine SLAVE-Adresse (Abbildung 2) über den Bus eindeutig anzusprechen ist.

Um mehrere Bausteine gleichen Typs am I²C-Bus betreiben zu können, sind die unteren zwei Bits (A 0 und A 1) der SLAVE-Adresse programmierbar.

Damit lassen sich vier verschiedene SLAVE-Adressen über einen spannungsempfindlichen Eingang (Pin 1) einstellen. Der interne Spannungskomparator des SAA 1064 kann dazu vier verschiedene Spannungspegel (0V, 3/8 V, 5/8 V und V_{CC}) auswerten. Jedem Pegel ist eine Adresse zugeordnet. Mit Hilfe der Jumper J 2 bis

J 4 kann der externe Spannungsteiler auf die entsprechende Spannung und damit Adresse eingestellt werden. Die Zuordnung der Jumper J 2 bis J 4 zu den Adress-Bits A 0 und A 1 ist in Tabelle 1 dargestellt.

Das R/W-Bit der SLAVE-Adresse gibt an, ob Daten geschrieben oder gelesen werden sollen. Eine logische „1“ bedeutet Lesen, und eine „0“ steht für Schreiben.

Der SAA1064 bietet die Möglichkeit, ein STATUS-Byte (siehe Abbildung 3) auszulesen. Dieses STATUS-Byte enthält das „PR“- (Power-Reset)-Flag, welches bei einem korrekt durchgeführten Reset ge-

setzt wird. Tritt ein Fehler im Einschaltmoment auf, bleibt dieses Bit auf „0“ stehen. Durch Auslesen des STATUS-Bytes erfolgt das Rücksetzen des PR-Bit.

Die Steuerung über den I²C-Bus

Das I²C-Protokoll für einen Lesezugriff ist in Abbildung 4 dargestellt.

Die Übertragung wird vom MASTER gestartet, indem dieser eine START-Bedingung generiert und anschließend die SLAVE-Adresse sendet, die der SLAVE bestätigt.

Für einen Lesezugriff muß das R/W-Bit gesetzt sein, damit der SLAVE (SAA 1064) den Lesezugriff erkennt.

Anschließend kann das Lesen des STATUS-Bytes (enthält das „PR“-Bit) erfolgen. Das gelesene Byte wird nicht vom MASTER bestätigt.

Die Generierung einer STOPP-Bedingung durch den MASTER beendet dann die Übertragung.

In Abbildung 5 ist ein Protokoll zum Schreiben von Daten in den SAA 1064 zu sehen. Zuerst wird wie gewohnt die SLAVE-Adresse vom MASTER gesendet und bei korrekter Übertragung vom SLAVE bestätigt.

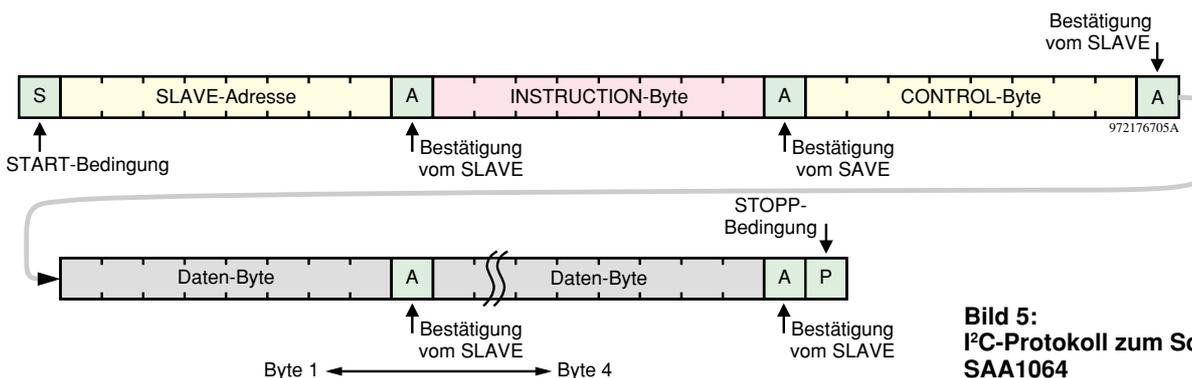


Bild 5: I²C-Protokoll zum Schreiben in das SAA1064

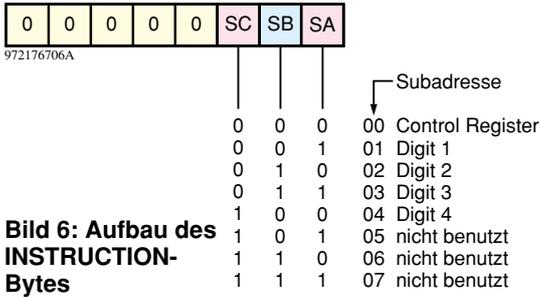


Bild 6: Aufbau des INSTRUCTION-Bytes

Wird ein Bit auf „1“ gesetzt, leuchtet das entsprechende Segment der Anzeige auf (siehe Abbildung 8). Auf diese Weise kann man neben den Ziffern auch selbstdefinierte Zeichen zur Anzeige bringen.

Zur Beendigung der Datenübertragung generiert der MASTER eine STOPP-Bedingung.

Schaltung

Das Schaltbild des I²C-Display-Boards ist in Abbildung 9 dargestellt. Die beiden parallel geschalteten Buchsen BU1 und BU2 stellen die Verbindung zum I²C-Bus her. Sie versorgen das Board mit Betriebsspannung und führen die SCL- und SDA-Leitungen mit sich.

Die Versorgungsspannung für die Schaltung ist wahlweise extern über die Buchse BU 3 (7V bis 15V) oder intern (über den I²C-Bus) über die Buchse BU 1 bzw. BU 2 (5V) zuführbar. Diese Auswahl ist durch entsprechendes Setzen des Jumpers J1 auf „Int.“ oder „Ext.“ möglich.

Im Fall der externen Spannungsversorgung erfolgt mittels IC 1 die Stabilisierung der Betriebsspannung auf 5V. Die Kondensatoren C1 bis C4 sowie C 6 und C 7 dienen zur Siebung bzw. Störunterdrückung.

Das IC 2 (SAA1064) steuert über die Ausgänge P 1 bis P 16 die Segmente und den Dezimalpunkt der vier 7-Segment-Anzeigen (DI 1 bis DI 4) an.

Die Steuerausgänge MX 1 (Pin 11) und MX 2 (Pin 14) sorgen über die Transistoren T 1 und T 2 dafür, daß jeweils nur zwei der vier 7-Segment-Anzeigen gleichzeitig aktiv sind (Multiplexbetrieb).

Die Widerstände R3 bis R5 bilden einen mittels Jumper J 2 bis J 4 einstellbaren Spannungsteiler zur Einstellung der gewünschten SLAVE-Adresse über Pin 1 des SAA 1064.

Mit dem nachfolgenden INSTRUCTION-Byte erfolgt das Laden eines Registerzählers, mit dem acht verschiedene Register auswählbar sind. Für die Adressierung der Subadresse sind die Bits SA bis SC zuständig. Die Zuordnung, welche Bitkombination welches Register bzw. welche Subadresse anspricht, ist in Abbildung 6 zu sehen. Eine Auto-Inkrement-Schaltung sorgt dafür, daß der Registerzähler nach jedem Daten-Byte um eins erhöht wird.

Erfolgt z. B. mit dem INSTRUCTION-Byte zuerst die Auswahl des Registers mit der Subadresse 0 (CONTROL-Register), wird das nachfolgende Daten-Byte in dieses Register geschrieben, das nächste Byte in Register 1 (Digit 1) usw.

Nach Erreichen des Registers 7 beginnt der Registerzähler wieder bei 0.

Die einzelnen Register haben folgende Funktionen:

- CONTROL-Register:

Das CONTROL-Register legt den Betriebsmodus, den Ausgangsstrom und die Anzahl der eingeschalteten Digits fest. Zusätzlich können mit Bit C3 alle Segmente aktiviert werden (Testmode). Eine Auflistung aller Funktionen ist in Abbildung 7 dargestellt.

- Digit 1 bis Digit 4:

Jedes dieser Register korrespondiert mit einer der vier möglichen 7-Segment-Anzeigen.

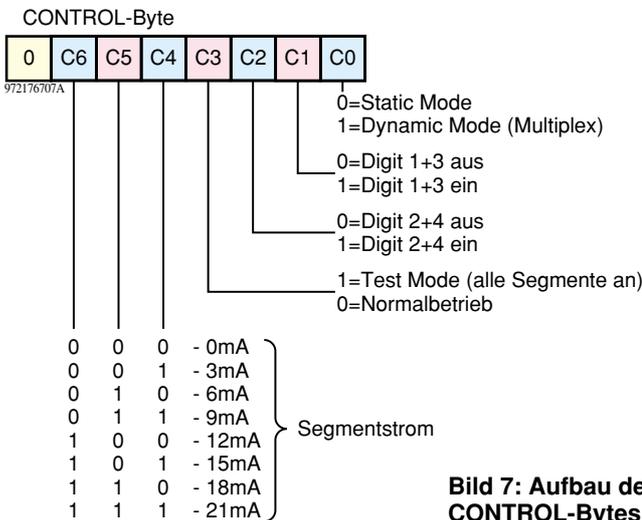


Bild 7: Aufbau des CONTROL-Bytes

Stückliste: I²C-LED-Board

Widerstände:

- 220Ω R1, R2
- 10kΩ R3
- 18kΩ R5
- 22kΩ R4

Kondensatoren:

- 2,7nF C5
- 100nF/ker C2, C4, C7
- 10µF/25V C3, C6
- 47µF/25V C1

Halbleiter:

- 7805 IC1
- SAA1064 IC2
- BC548 T1, T2
- HDSP5501 DI1-DI4

Sonstiges:

- SUB-D-Buchsenleiste, 9polig, print BU1
- SUB-D-Stiftleiste, 9polig, print BU2
- Klinkenbuchse, 3,5mm, mono, print BU3
- Stiftleiste, 1 x 3polig J1
- Stiftleiste, 1 x 2polig J2-J4
- 2 Jumper
- 1 Kühlkörper, SK 13
- 1 Zylinderkopfschraube, M3 x 8mm
- 1 Mutter, M3

Nachbau

Die wenigen Bauteile für das I²C-LED-Board finden auf einer doppelseitigen Platine mit den Abmessungen 66 x 84 mm Platz. Die Bestückung erfolgt in gewohnter Weise anhand der Stückliste und des Bestückungsplans, wobei die Bauteile von oben durch die entsprechenden Bohrungen gesteckt, auf der Platinenunterseite verlötet und überstehende Drahtenden mit einem Seitenschneider abzuschneiden sind, ohne dabei die Lötstellen zu beschädigen. Bei den Halbleitern und den Elkos ist unbedingt auf die richtige Einbaulage bzw. Polung zu achten.

IC 1 wird zur besseren Wärmeabfuhr mit einem Kühlkörper versehen, immerhin kann die Baugruppe aufgrund der LED-Anzeige bis zu 200 mA aufnehmen. Hierzu sind die

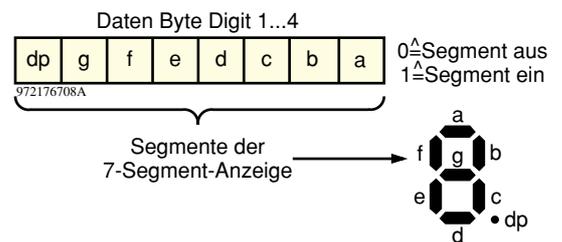


Bild 8: Aufbau des Datenbytes für Digit 1 bis 4

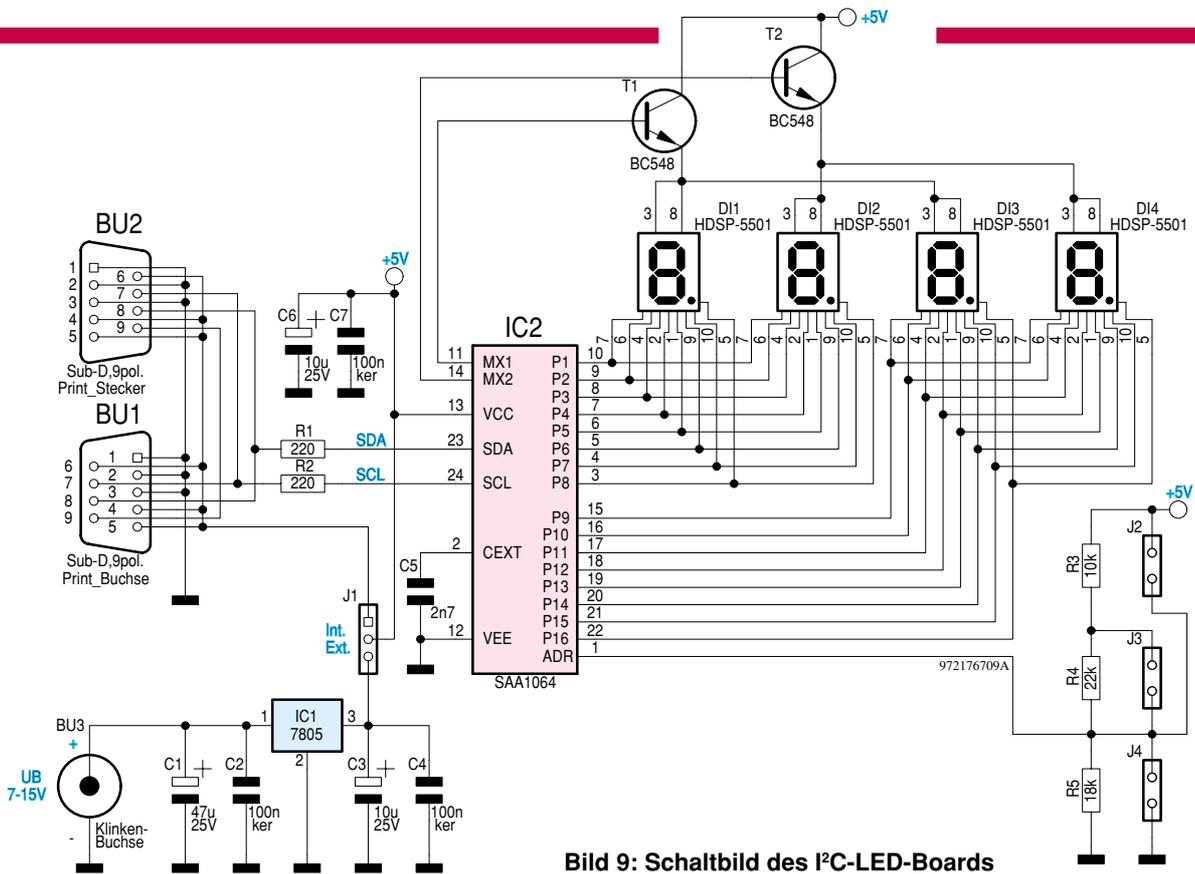


Bild 9: Schaltbild des I²C-LED-Boards

Anschlüsse des Spannungsreglers entsprechend den Bohrungen auf der Platine abzuwinkeln. Anschließend wird er zusammen mit dem Kühlkörper und einer M3x8mm-Schraube auf der Platine festgeschraubt und anschließend verlötet.

Damit ist der Aufbau bereits beendet, und die Schaltung ist nach nochmaliger Überprüfung der exakten Bestückung einsatzbereit.

Software

Die Grundroutinen zur Ansteuerung des

I²C-Display-Boards liegen als Quellcode in den Programmiersprachen C und PASCAL vor.

Ein kleines Beispielpogramm zeigt die Anwendung der Routinen im Zusammenhang mit dem I²C-PC-Interface.

Die Ansteuerung des I²C-Display-Boards ist sehr einfach, so daß dazu lediglich zwei Funktionen benötigt werden.

Mit der Funktion „LED_OUT (Adresse, Control_Byte, Display[]);“ erfolgt die Ausgabe auf das I²C-Display-Board.

Die Variable „Adresse“ entspricht der I²C-Adresse des Boards, so daß bis zu 4

verschiedene Boards über einen I²C-Bus ansteuerbar sind.

Die Variable „Control_Byte“ dient der Konfiguration des LED-Treibers. Diese gibt sowohl die Anzahl der 7-Segment-Anzeigen als auch deren Kontrast vor.

Der Inhalt der Anzeigen wird in das Datenfeld „Display[]“ übergeben. Es enthält für jede Anzeigenstelle einen Eintrag, bei dem jedes Bit einem Anzeigensegment entspricht.

Die Funktion „LED_CONTROL_OUT (Adresse, Control_Byte);“ sorgt lediglich für die Übertragung des Control-Bytes. Hierdurch kann zum

Beispiel der Kontrast der Anzeige verändert werden.

Der aktuelle Wert des Control-Bytes ist in der globalen Variable „CONTROL_BYTE“ abgelegt, so daß einzelne Bits mit Hilfe von UND- und ODER-Verknüpfungen einstellbar sind.

Mit dieser Funktion kann man zum Beispiel eine blinkende Anzeige realisieren, indem der Kontrast ständig zwischen 0 und dem Normalwert umgeschaltet wird.

Die Quellcodes zur Ansteuerung des I²C-Display-Boards liegen dem Bausatz auf einer Programmdiskette bei oder können über die ELV-Infobox per Modem kostenlos abgerufen werden, wobei nur die Telefongebühren anfallen.

ELV

Fertig aufgebaute Platine mit Bestückungsplan

