

Aus 8085 wird Z80

Da der Befehlssatz des Mikroprozessors 8085 nur eine Teilmenge (ausgenommen SIM und RIM) des viel umfangreicheren Z80-Befehlssatzes bildet, können zwar 8085-Programme auf Z80-Systemen ablaufen, die Übernahme von Z80-Software auf 8085-Systeme ist jedoch nicht bzw. nicht direkt möglich. Im folgenden wird ein einfacher Weg aufgezeigt, der die Verarbeitung von Z80-Programmen in 8085-Mikrocomputern ermöglicht.

Der Schlüssel zur Lösung dieser Aufgabe ist der CMOS-Mikroprozessor NSC800 von National Semiconductor. Dieser Prozessor weist ein (fast) Z80-gleiches Innenleben auf und ist nach außen hin dem 8085 sehr ähnlich. Die internen Unterschiede gegenüber dem Z80 bestehen in dem auf 8 Bit erweiterten Refresh-Register des NSC800 und in der fixen internen Verwendung der I/O-Adresse BB_H als Interrupt Control Register (ICR). (Durch Setzen entsprechender ICR-Bits wird die Annahme der vier maskierbaren Interrupts \overline{RSTA} , \overline{RSTB} , \overline{RSTC} und \overline{INTR} ermöglicht.) Nach außen hin differieren 8085 und NSC800 in folgenden Punkten:

- Anschlußschema
- logische Aktivierungspegel diverser Leitungen
- serielle Ein-/Ausgänge.

Wie aus den Anschlußbildern in den entsprechenden Datenblättern hervorgeht, weisen NSC800 und 8085 achtunddreißig funktionell gleiche Anschlüsse auf, nur sind sie unterschiedlich plaziert. Neben Anordnungsunterschieden bestehen aber auch noch Differenzen in den

logischen Aktivierungspegeln diverser Leitungen, einer Interruptadresse und der Funktion des Interrupteingangs niedrigster Priorität (Tabelle).

Außer den achtunddreißig funktionsgleichen Anschlüssen von 8085 und NSC800 gibt es lediglich zwei, die funktionell völlig verschieden sind. Es sind dies \overline{PS} (power save input) und \overline{RFSH} (refresh output) beim NSC800 und \overline{SID} (serial input data) und \overline{SOF}

Tabelle der funktionellen Unterschiede zwischen 8085 und NSC800

8085		NSC800		Funktion
Anschluß	Restart-Adresse	Anschluß	Restart-Adresse	
TRAP	24H	\overline{NMI}	66H	Interrupt höchster Priorität. Nicht maskierbar.
RST 7,5	3CH	\overline{RSTA}	3CH	Interr. zweithöchst. Priorität. Maskierbar.
RST 6,5	34H	\overline{RSTB}	34H	Interr. dritthöchster Priorität. Maskierbar.
RST 5,5	2CH	\overline{RSTC}	2CH	Interr. vierthöchster Priorität. Maskierbar.
\overline{INTR}		\overline{INT}		Interr. niedrigster Priorität. NSC800: drei Betriebsarten wie Z80.
\overline{HOLD}		\overline{BREQ}		Busfreigabeaufforderung
HLDA		BACK		Busfreigabesignal

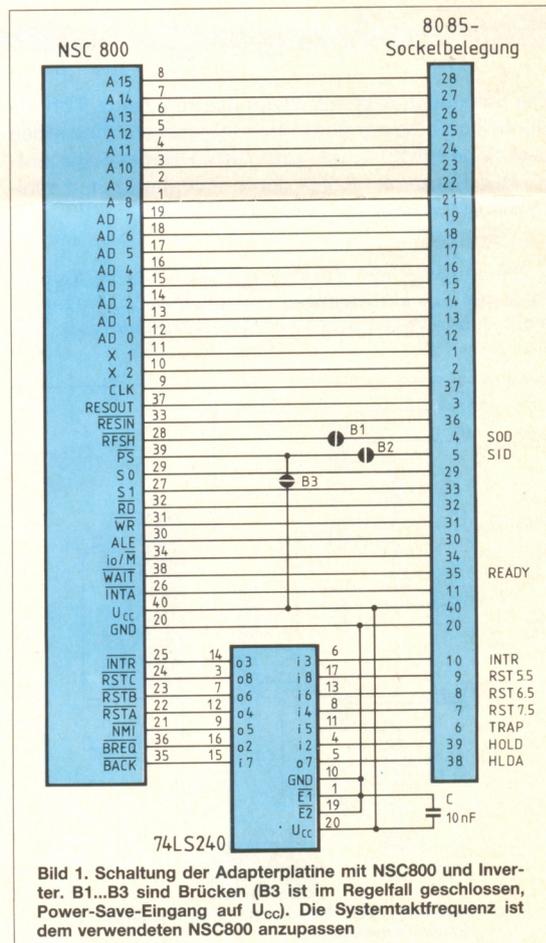


Bild 1. Schaltung der Adapterplatine mit NSC800 und Inverter. B1...B3 sind Brücken (B3 ist im Regelfall geschlossen, Power-Save-Eingang auf U_{CC}). Die Systemtaktfrequenz ist dem verwendeten NSC800 anzupassen

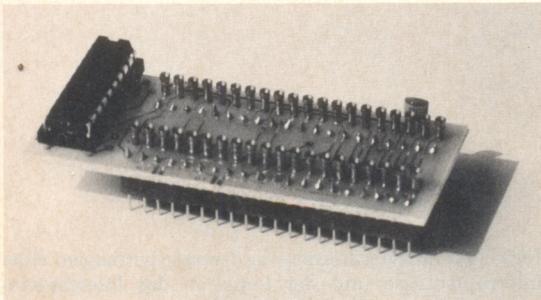


Bild 2. Die Adapterplatte ohne NSC800 aber mit eingesetztem Inverterbaustein; auf der Unterseite der 40polige Stecker als Verbindung zur Systemplatte (8085-Sockel)

(serial output data) beim 8085. Ihre Verwendung entscheidet, ob eine Adaptierung für Z80-Software möglich ist oder nicht.

Werden in einem 8085-System die Leitungen SID und SOD nicht verwendet, so bietet sich die Möglichkeit, durch Auskreuzen der Anschlüsse und Invertieren von sieben Leitungen einen NSC800 anstelle eines 8085 einzusetzen. Die entsprechende Schaltung ist in Bild 1 dargestellt. Bild 2 zeigt die praktische Realisierung der Schaltung in Form einer $70 \times 30 \text{ mm}^2$ großen Adapterplatte, die oben zwei Sockel zur Aufnahme von NSC800 und Inverter (74LS240) aufweist und unten mit einem 40poligen Stecker versehen ist, der ein Einstecken in einen 8085-Sockel ermöglicht.

Dipl.-Ing. Franz Braunschmid

Signalübergangs-Reihenfolgen ausgewertet

Die Schaltung zeigt mit ihrem Ausgangssignal an, ob die Reihenfolge der an ihren Eingängen stattfindenden Signalübergänge einer geforderten Reihenfolge entspricht. Sie zeichnet sich durch Modularität und minimalen Schaltungsaufwand aus.

Die Aufgabe, auf eine richtige oder falsche Reihenfolge von Signalübergängen entsprechend zu reagieren, stellt sich in vielen Anwendungen, insbesondere bei der

Prozeßüberwachung und -steuerung. Das anschaulichste Beispiel ist ein elektronisches Schloß, das sich nach Eingabe einer richtigen Kombination öffnet, während eine falsche Eingabe ignoriert wird oder einen Alarm auslöst. Die sukzessive Eingabe einer Kombination über ein angeschlossenes Schalterfeld führt zum Wechsel einzelner Eingangssignale (E_i) der im Bild dargestellten Schaltung vom Zustand „0“ in den Zustand „1“.

Wird dabei die Reihenfolge der Numerierung der Eingänge eingehalten (E_1, E_2, \dots, E_n), so wechselt mit dem Übergang des letzten Eingangssignals E_n bei gesetztem Flipflop FF auch das Ausgangssignal A von „0“ nach „1“. Sobald die Reihenfolge jedoch nicht mehr mit den Nummern der Eingänge übereinstimmt (z. B. bei FF wird über das entsprechende ODER-Gatter (OR_{i-1}) das Signal am Ausgang R des gemeinsamen UND-Gatters AN_1 von „1“ auf „0“ gesetzt und über das gesetzte Flipflop FF auch das Signal Q von „1“ nach „0“ geändert ($Q \cong$ „0“ könnte z. B. einen Alarm auslösen). Gleichzeitig sperrt FF, und das Ausgangssignal A bleibt auf „0“. Das Verzögerungsglied VG dient dazu, einen kurzfristigen Wechsel des Signals A von „0“ nach „1“ zu verhindern, wenn als erstes das Signal E_n von „0“ nach „1“ wechselt.

Um eine neue Reihenfolge auszuwerten, müssen nun erst wieder zumindest der falsch gewählte Eingang E_i auf „0“ und das Flipflop FF durch ein negatives Signal am Eingang S gesetzt werden.

Durch Kombination mehrerer Schaltungen als Module lassen sich auch komplexere Bedingungen an die Reihenfolge realisieren. Da die Schaltung nur eine geringe Signallaufzeit-Verzögerung verursacht, eignet sie sich insbesondere auch für zeitkritische Anwendungen.

Dipl.-Inform. Helmut Bährin,
Dr. rer. nat. Klaus Heidtmann

