

Rüdiger Bäcker, Willi Sicking

Uhrenbaustein RTC-58321

Eine Uhrenkarte für den NDR-Klein-Computer

Mit dem RTC-58321 steht ein preiswerter und flexibler Uhrenbaustein zur Verfügung. Der Anschluß an einen Rechner über eine Ein-/Ausgabe-Schnittstelle soll am Beispiel des NDR-Klein-Computers demonstriert werden.

Nachdem der ursprünglich auf der Uhrenkarte des NDR-Klein-Computers eingesetzte Baustein E05-16 nicht mehr hergestellt wird, stellt sich die Frage nach einer Alternative. Mit dem RTC-58321 ist eine Software-Emulation der Originalkarte möglich; die E05-Karte kann daher problemlos ersetzt werden. Die Übertragung auf andere Rechner ist ebenfalls leicht möglich.

Die Schaltung beinhaltet eine Überraschung im positiven Sinn: Man braucht keine zusätzliche Karte! Der RTC-58321 wird einfach mit auf die IOE-Karte gesetzt, auf der sich schon die Centronics-Schnittstelle befindet. Dazu braucht man nur zwei vierfach Tristate-Treiber, zwei Widerstände, zwei Dioden, einen Taster und eine Lithium-Batterie. Die in

der Schaltung verwendeten Inverter sind auf der IOE-Karte noch frei.

Alles an Bord

Aus dieser Aufzählung ist ersichtlich, daß in dem verwendeten Uhrenbaustein schon alles mögliche „drin“ sein muß. Daher zunächst eine kurze Beschreibung des Bausteins:

Der RTC ist ein Echtzeit-Modul mit integriertem Quarzoszillator, Backup-Schaltung für Akku-Betrieb, Zeit- und Datumfunktion sowie einem Taktausgang für Interrupt-Betrieb. Verpackt ist das Ganze in in einem 16poligen DIP-Gehäuse. Durch den eingebauten Quarz ist kein Abgleich erforderlich, was einfachen Nachbau der Schaltung sichert. Der

Baustein besitzt einen eingebauten Zeit- (Stunden, Minuten, Sekunden) und Datumzähler (Jahr, Monat, Tag und Wochentag). Man kann zwischen 12- und 24-Stunden-Anzeige umschalten, auch Schaltjahre werden automatisch berücksichtigt. Auf den Datenleitungen können Impulse mit einer Periodendauer von 1/1024 Sekunde, 1 Sekunde, 1 Minute oder 1 Stunde ausgegeben werden. Des weiteren ist eine Start-Stop-Reset-Funktion vorhanden.

Zum Anschluß des RTC wird nur ein 4-Bit-Datenbus verwendet. Da der Baustein in CMOS-Technologie aufgebaut ist, verbraucht er sehr wenig Strom und läßt sich leicht mit einem Akku puffern. Einbrüche der Versorgungsspannung werden automatisch erkannt und mit einer Sperrung der Ein- und Ausgänge beantwortet. Der RTC ist mit einer Genauigkeit von 10 ppm erhältlich, also sehr präzise. Alle Funktionen sind in der Tabelle aufgeführt.

Die Funktionen 0...C dienen zum Lesen und Setzen von Tag, Datum und Uhrzeit. Bei der Angabe der Stunden-Zehner kann zwischen 12- und 24-Stunden-Ausgabe unterschieden werden. Mit Bit 2 und 3 der Tag-Zehner-Eingabe kann zwischen gregorianischem und dem japanischen Showa-Kalender unterschieden werden. Da für uns sicher nur der gregorianische Kalender wichtig ist, werden beide Bits auf Null gesetzt (das ist auch für die automatische Schaltjahr-Erkennung wichtig). Mit der Funktion D werden die Zeitstufen zurückgesetzt und die Funktionen E und F schalten die internen Zeitsignale auf die Datenleitungen D0...D3. Dann erscheint auf D0 ein Takt mit einer Frequenz von 1024 Hz, auf D1 wird jede Sekunde, auf D2 jede Minute und auf D3 jede Stunde ein Impuls ausgegeben. Damit kann man bei geeigneter Beschaltung eine Interrupt-Steuerung aufbauen.

Die Funktionen des RTC-58321

Funktion	Adr	Eingang A3 A2 A1 A0	Ausgang D3...D0	Wert	Bemerkung
Sek. Einer	0	L L L L	aktueller Wert	0...9	
Sek. Zehner	1	L L L H	aktueller Wert	0...5	
Min. Einer	2	L L H L	aktueller Wert	0...9	
Min. Zehner	3	L L H H	aktueller Wert	0...5	
Std. Einer	4	L H L L	aktueller Wert	0...9	
Std. Zehner	5	L H L H	aktueller Wert	0...1	D3 = H für 24 Std.-Anzeige D3 = L für 12 Std.-Anzeige D2 = H für P.M. D2 = L für A.M.
Wochentag	6	L H H L	aktueller Wert	0...6	hier 0 = Mittwoch
Tag Einer	7	L H H H	aktueller Wert	0...9	
Tag Zehner	8	H L L L	aktueller Wert	0...3	D3/D2 für Schaltjahr
Monat Einer	9	H L L H	aktueller Wert	0...9	
Monat Zehner	A	H L H L	aktueller Wert	0...1	
Jahr Einer	B	H L H H	aktueller Wert	0...9	
Jahr Zehner	C	H H L L	aktueller Wert	0...9	
Reset	D	H H L H			alle Zähler löschen
Test	E	H H H L	Referenz-Sign.	0...F	Signale auf D0...D3
Test	F	H H H H	Referenz-Sign.	0...F	Signale auf D0...D3

Die Schaltung

Wie bereits erwähnt, wird der RTC mit auf die Centronics-Schnittstelle gesetzt. Um hier zwischen Drucker- und RTC-Betrieb unterscheiden zu können, wird noch ein Tristate-Treiber benötigt. Er sorgt für die Entkopplung des RTC bei Druckerbetrieb. Die Freigabe des RTC erfolgt dann über Bit 6 des Ausgabe-Ports \$41; ist dieses Bit Null, so kann auf den RTC zugegriffen werden. Die Daten werden über Bit 0...3 der Ein-/Ausgabe-Adresse \$40 übertragen. Vier Widerstände sorgen für die TTL-Pegel auf diesen Leitungen. Zusätzlich ist eine Steuerlogik zur hardwaremäßigen Freigabe des

BUSY-Signals bzw. des 1MIN-Impulses als Interrupt-Quelle vorhanden. Die Schaltung kann mit einer Batterie (1,5 V) gepuffert werden, die Dioden D1 und D2 sind für die Entkopplung vorgesehen. Die Steuerung des RTC erfolgt über die Ausgabe-Adresse \$41 (ADRESSWRITE, WRITE, READ und CS2) sowie über die Eingabe-Adresse \$41 (BUSY). Da zur Verbindung zwischen RTC und Bus vier Datenleitungen vorhanden sind, werden diese im Multiplex-Verfahren benutzt. Die dabei verwendeten Steuerleitungen haben folgende Bedeutung:

Adr.-Write

Diese Leitung dient zur Adressierung der 16 internen Register des RTC. High-Pegel auf dieser Leitung teilt dem Baustein mit, daß sich auf dem Datenbus ein gültiges Adreßwort befindet. Der interne Adreßdecoder selektiert nun das entsprechende Register, das beim nächsten Zugriff als Ziel oder Quelle angesprochen werden kann.

Write

Die Steuerleitung für den Datentransfer zum RTC: Steht hier High-Pegel, so werden die Daten vom Bus in das vorher mit Adr.-Write selektierte Register übernommen.

Read

Die Steuerleitung für den Datentransfer vom RTC: High auf dieser Leitung bewirkt, daß der Inhalt des mit Adr.-Write ausgewählten Registers auf dem Datenbus ausgegeben wird.

CS2

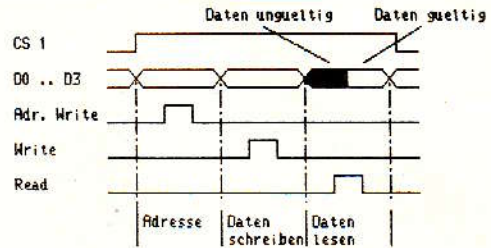
Dies ist eine der beiden Baustein-Auswahl-Leitungen. Sind beide Leitungen aktiv (high), so ist der Baustein adressiert, andernfalls werden alle Ein-/Ausgabeleitungen des RTC inaktiv. In der Schaltung ist CS1 mit der Versorgungsspannung verbunden, womit bei fehlender Spannung jeder Zugriff gesperrt ist.

BUSY

Mit diesem Ausgang signalisiert der Baustein, daß er nicht zur Datenübertragung bereit ist. Das ist immer dann der Fall, wenn der interne Zeitähler um 1 Sekunde erhöht wird. Der Ausgang liefert also jede Sekunde einen negativen Impuls.

Der RTC hat noch zwei weitere Steuerleitungen, nämlich Test und Stop. Wie der Name schon sagt, dient der eine zu Testzwecken; eine 1 am Stop-Eingang hält den internen Zeitähler an. In der vorliegenden Schaltung werden die beiden Eingänge nicht benutzt und sind daher an Masse gelegt.

Bild 1. Zeitabläufe beim RTC 58321



Über die Ausgabe-Adresse \$41 wird außerdem der Drucker (Bit 0) und die Interrupt-Freigabe (Bit 6, 7) gesteuert. Der Schreibschutz-Schalter verhindert ein unbeabsichtigtes Verstellen der Uhr, nur mit geöffnetem Schalter kann in den RTC geschrieben werden. Dies ist jedoch keine Einschränkung, da man die Zeit ja nur beim Wechsel zwischen Sommer- und Normalzeit sowie nach einem Batteriewechsel stellen muß.

Datentransfer

Ein Zugriff auf den Uhrenbaustein hat also den folgenden Ablauf:

- CS2 auf Eins legen
- Register-Adresse anlegen
- Adr.-Write auf Eins legen

- ADR.-Write wieder auf Null
- Schreibzugriff:
 - Daten anlegen
 - Write auf Eins legen
 - mindestens 2 µs warten
 - Write wieder auf Null
- Lesezugriff:
 - Read auf Eins legen
 - mindestens 2 µs warten
 - Daten auslesen
 - Read wieder auf Null
- CS2 wieder auf Null legen.

Ist der Baustein nicht in der Lage, einen Datentransfer durchzuführen, so wird dies durch eine Null am BUSY-Ausgang angezeigt. Da dies immer dann auftritt, wenn der RTC seinen internen Zeitähler um eine Sekunde erhöht, erscheint

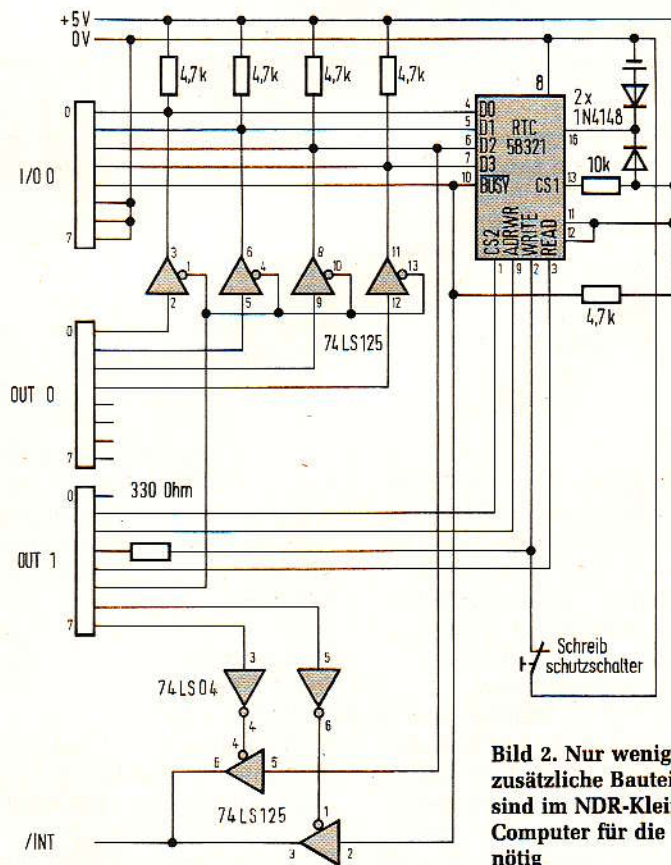


Bild 2. Nur wenige zusätzliche Bauteile sind im NDR-Klein-Computer für die Uhr nötig

1. OUT 0 (\$40) :

```

+-----+
| 0 | 1 | 1 | 1 | 2 | 1 | 3 | 1 | 4 | 1 | 5 | 1 | 6 | 1 | 7 | 1 |
+-----+
| I | I | I | I | I | I | I | I |
| I | I | I | I | I | I | I | I | +----> D7 zum Drucker
| I | I | I | I | I | I | I | I | +----> D6 zum Drucker
| I | I | I | I | I | I | I | I | +----> D5 zum Drucker
| I | I | I | I | I | I | I | I | +----> D4 zum Drucker
| I | I | I | I | I | I | I | I | +----> D3 zum Drucker oder zum RTC *
| I | I | I | I | I | I | I | I | +----> D2 zum Drucker oder zum RTC *
| I | I | I | I | I | I | I | I | +----> D1 zum Drucker oder zum RTC *
| I | I | I | I | I | I | I | I | +----> D0 zum Drucker oder zum RTC *
+-----+
(* ist abhängig von Bit 5, Port OUT 1 )

```

2. OUT 1 (\$40) :

```

+-----+
| 0 | 1 | 1 | 1 | 2 | 1 | 3 | 1 | 4 | 1 | 5 | 1 | 6 | 1 | 7 | 1 |
+-----+
| I | I | I | I | I | I | I | I |
| I | I | I | I | I | I | I | I | +----> 1 = Freigabe des Minuten-Interrupts
| I | I | I | I | I | I | I | I | +----> 1 = Freigabe des Sekunden-Interrupts
| I | I | I | I | I | I | I | I | +----> 1 = Tristate
| I | I | I | I | I | I | I | I | +----> 1 = READ
| I | I | I | I | I | I | I | I | +----> 1 = WRITE
| I | I | I | I | I | I | I | I | +----> 1 = ADDRESSWRITE
| I | I | I | I | I | I | I | I | +----> 1 = CS
+-----+
+-----+ /STROBE zum Drucker

```

3. I/O 0 (\$40) :

```

+-----+
| 0 | 1 | 1 | 1 | 2 | 1 | 3 | 1 | 4 | 1 | 5 | 1 | 6 | 1 | 7 | 1 |
+-----+
| I | I | I | I | I | I | I | I |
| I | I | I | I | I | I | I | I | +----> NC
| I | I | I | I | I | I | I | I | +----> NC
| I | I | I | I | I | I | I | I | +----> NC
| I | I | I | I | I | I | I | I | +----> /BUSY vom RTC (immer 1 Sek. Takt)
| I | I | I | I | I | I | I | I | +----> D3 vom RTC oder 1 Std. Takt *
| I | I | I | I | I | I | I | I | +----> D2 vom RTC oder 1 Min. Takt *
| I | I | I | I | I | I | I | I | +----> D1 vom RTC oder 1 Sek. Takt *
| I | I | I | I | I | I | I | I | +----> D0 vom RTC oder 1024 Hz Takt *
+-----+
(* ist abhängig von der Programmierung - siehe REFON )

```

Bild 3. Beachten Sie bitte, daß bei der Schnittstellen-Belegung die Reihenfolge der Bits anders herum zählt

auch der BUSY-Impuls genau im 1-Sekunden-Takt. Man kann damit einen Interrupt auslösen und die Uhr somit im Hintergrund laufen lassen. Ein Zeitdiagramm dazu zeigt Bild 1. Die Schaltung (Bild 2) läßt sich in Fädeltchnik realisieren. Die dadurch geschaffene Belegung der Ein- und Ausgabe-Adressen zeigt Bild 3.

Die Software

Das Treiberprogramm (Bild 4), mit dem der 68008 die Uhrenkarte bedienen kann, besteht im Wesentlichen aus zwei Teilen, auf denen alle anderen Routinen aufbauen: Dem Block mit den Unterprogrammen zur Programmierung der Handshake-Leitungen (AWRITE usw.), sozusagen die unterste Software-Schnittstelle zum RTC, und dem Teil, in dem diese Grundroutinen zum kompletten Lese- und Schreibzugriff auf den RTC zusammengefügt werden. Dabei wird jeweils in D0 die Adresse übergeben, während in D1 die Daten vom bzw. für den RTC stehen.

Aufbauend auf diese Routinen sind folgende Programme und Funktionen vorhanden:

RTCSHOW

An einer festgelegten Position auf dem Bildschirm werden Wochentag, Datum und Uhrzeit ausgegeben.

LESEN

Wochentag, Datum und Uhrzeit werden entweder mit wählbarer Größe und Position auf dem Bildschirm oder in einen Puffer ausgegeben. Das aufrufende Programm muß dazu folgende Parameter übergeben:

- D0.B Schriftgröße
- D1.W X-Position
- D2.W Y-Position
- D3.B Ausgabe (0 = Puffer, 1 = Bild)
- D6.B Interrupt-Maske
- A0.L Puffer-Adresse (23 Byte)

Bei Puffer-Ausgabe werden die Werte in D0 und D1 nicht beachtet.

SETUHR

Ein Menüprogramm zum Stellen der Uhr (Bild 5). Die Eingabe von Wochentag, Datum und Zeit erfolgt im ASCII-Format, Fehler bei der Eingabe werden gemeldet. Dabei ist zu beachten, daß zum Zeitpunkt der Übernahme (also beim Drücken der Return-Taste) der Schreibschutz-Schalter geöffnet sein muß.

HEXREAD

Wochentag, Datum und Uhrzeit werden hexadezimal in einen Puffer geschrieben. Beim Aufruf wird in A0 die Puffer-Adresse übergeben.

TRAPHANDLER

Mit diesem Programm können alle bisher genannten Funktionen über den Trap-#7-Befehl aufgerufen werden. Im RDK-Grundprogramm-RAM ist nämlich in einer Tabelle ein Sprung eingetragen, der auf ein eigenes Programm umgelenkt werden kann. Damit kann das Uhr-Programm von jedem anderen Programm benutzt werden. Wie bei allen Grundprogramm-Traps wird einfach im Register D7.W die Nummer der gewünschten Funktion eingetragen und anschließend mit Trap #7 die RTC-Routine aufgerufen. Die einzelnen Funktions-Codes und die Aufruf-Parameter sind im Bild 4 zu finden. Natürlich muß vor dem ersten Aufruf der Trap-Handler mit TRAPINIT installiert werden.

Damit auch andere Programme TRAPINIT finden und ausführen können, ist ein HEADER vorgesehen, nach dem man suchen und dann TRAPINIT ausführen kann. Der Trap-Vektor wird ebenfalls auf die Vektor-Adresse der Prozessors 68008 geschrieben. Liegt das Grundprogramm nicht ab Adresse \$0, so wird damit der Umweg über die RAM-Tabelle gespart.

RTCINIT

Durch Aufrufen dieses Programmes wird der interrupt-gesteuerte Hintergrundbetrieb mit Ausgabe auf dem Bildschirm initialisiert. Zum Auslösen des Interrupts wird hier der BUSY-Ausgang benutzt, wodurch die Zeitausgabe jede Sekunde erfolgt. Die Schaltung ist so ausgelegt, daß der Interrupt hard- und softwaremäßig freigegeben werden muß. Auch für diese Initialisierung ist wieder ein Programm mit Header und eine Demo-Routine vorhanden. Nach Ausführen von INTINIT erscheint die Uhrzeit in der linken unteren Ecke, bis ein Reset ausgelöst oder die Interrupt-Priorität von einem anderen Programm erhöht wird.

0C7000	086 \$17000	* ADRESSE DER ROUTINE IM EPROM	0C7040	13C5 FFFFFFF4		0C7098		
0C7000			0C7046	6100 FFE0	MOVE.B D5, CONTR	0C7098		
0C7000			0C704A	4E75	BSR RTCWAIT	0C7098		
0C7000	OFFSET -\$20000	* ABLAGEADRESSE RAM !!	0C704C		RTS	0C7098		
0C7000	KOPF:		0C704C		ANRITES:	0C7098		
0C7000	DC.L \$55AA0180		0C704C	1A3C 00C7	MOVE.B #WRITES,D5	0C7098		
0C7004	DC.B \$254432049AE49		0C7050	CA06	AND.B D6,D5	0C7098		
0C7008	54		0C7052	13C5 FFFFFFF4	MOVE.B D5, CONTR	0C7098		
0C700C	0000046C	* USERTRAP WIRD INITIALISIERT	0C7058	6100 FFE0	BSR RTCWAIT	0C7098		
0C7010	0001D034		0C705C	4E75	RTS	0C7098		
0C7014	01 00 00 00 00	DC.L ENDEA-KOPF	0C705E			0C7098		
0C7019	00	DC.B 1,0,0,0	0C705E		WRITE:	0C7098		
0C701A		DS 0	0C705E		MOVE.B #WRITES,D5	0C7098		
0C701A	*****		0C705E		AND.B D6,D5	0C7098		
0C701A	***** ALLGEMEINE UNTERROUTINEN ZUM BETRIEB DES RTC (RTC-TOOL)		0C7062	CA06	MOVE.B D5, CONTR	0C7098		
0C701A	*****		0C7064	13C5 FFFFFFF4	BSR RTCWAIT	0C7098		
0C701A	*** VEREINBARUNGEN		0C706E	4E75	RTS	0C7098		
0C701A			0C7070			0C7098		
= FFFFFFF4	DIN	EDU \$FFFFFF40	0C7070		*** DATEN VOM RTC LESEN	0C7098		
= FFFFFFF40	DOUT	EDU \$FFFFFF40	0C7070			0C7098		
= FFFFFFF41	CONTR	EDU \$FFFFFF41	0C7070		UPREAD:	0C7098		
= FFFFFFF68	TASTE	EDU \$FFFFFF68	0C7070		* HIER WIRD EIN WERT AUS DEM UHRENBANSTEIN GELESEN	0C7098		
= 000000C7	WRITES	EDU \$11000011	0C7070	13C0 FFFFFFF40	* DO = GEW. ADRESSE, D1 = ERGEBNIS IN HEX	0C7098		
= 000000C9	WRITES	EDU \$11001011	0C7076	6100 FFD4	* ADRESSE ANLEGEN	0C7098		
= 000000F3	READS	EDU \$11110011	0C707A	6100 FFBE	* UND IN RTC SCHREIBEN	0C7098		
= 000000E1	TRISTS	EDU \$11110001	0C707E	4E75	* DANN DATEN AUSLESEN, ERGEBNISS IN D1	0C7098		
= 0000003F	INTDIS	EDU \$00111111	0C7080		RTS	0C7098		
= 0000008F	INTMIN	EDU \$10111111	0C7080		*** DATEN IN DEN RTC SCHREIBEN	0C7098		
= 0000007F	INTSEK	EDU \$01111111	0C7080			0C7098		
0C701A			0C7080		UPWRITE:	0C7098		
0C701A	* BIT: AAAAAAAAA	BELEGUNG DES KONTROLLPORTS :	0C7080	1401	* HIER WIRD EIN WERT IN DEN UHRENBANSTEIN BESCHREIBEN	0C7098		
0C701A	* 0 AAAAAAAAA	STROBE ZUM DRUCKER	0C7082	13C0 FFFFFFF40	* DO = GEW. ADRESSE, D1 = ZU SCHREIBENDER WERT	0C7098		
0C701A	* 1 AAAAAAAAA	CS	0C7088	6100 FFC2	* WERT RETTEN	0C7098		
0C701A	* 2 AAAAAAAAA	ADRESSWRITE	0C708C	13C2 FFFFFFF40	* ADRESSE ANLEGEN	0C7098		
0C701A	* 3 AAAAA	WRITE	0C7092	6100 FFCA	* UND IN RTC SCHREIBEN	0C7098		
0C701A	* 4 A	READ	0C7096	4E75	* DATEN ANLEGEN	0C7098		
0C701A	* 5 A	TRI-STATE	0C7098		* UND IN RTC SCHREIBEN	0C7098		
0C701A	* 6 A	FREIGABE SEKUNDEN INTERRUPT	0C7098		RTS	0C7098		
0C701A	* 7 A	FREIGABE MINUTEN INTERRUPT	0C7098		***** NACHFOLGEND PROGRAMME ZUM BETRIEB DES RTC ALS UHR	0C7098		
0C701A	*** DIVERSE UNTERPROGRAMME		0C7098		*****	0C7098		
0C701A			0C7098		*** EIN-/AUSGABEBUFFER - WICHTIG, MUSS RAM SEIN ! (V 1.4 NUM IM SPRG.-BUFFER)	0C7098		
0C701A	BUSYTEST:		0C7098		DOUT	0C7098		
0C701A	MOVE.B D1N,D1	* TEST OB RTC BEREIT	0C7098		EDU \$1000-27*2	0C7098		
0C7020	0801 0004	* DATEN STEHEN AUCH FUER ANDERE UPRG. ZUR VERF.	0C7098		DUT1	0C7098		
0C7024	67F4	* BUSY ?	0C7098		EDU OUT+26	0C7098		
0C7026	4E75	* JA, DANN WARTEN	0C7098		*** UNTERPROGRAMM ZUR TEXTAUSGABE	0C7098		
0C7028			0C7098		WRITECT:	0C7098		
0C7028	6100 FFF0	* BUSYTEST REICHT	0C7098		MOVE #WRITE,D7	0C7098		
0C703C	1A3C 90E1	* ALS WAIT	0C7098		TRAP #1	0C7098		
0C7030	CA06	* IN D6 STEHT CODE FUER INTERRUPTBETRIEB	0C709C	3E3C 000A	RTS	0C7098		
0C7032	13C5 FFFFFFF41	* TRI-STATE-SELECT	0C709E	4E75	CLR:	0C7098		
0C7038	4E75		0C70A0		MOVE.B #CLR,D7	0C7098		
0C703A			0C70A0		TRAP #1	0C7098		
0C703A	1A3C 00F3	* DATEN VOM RTC LESEN	0C70A4	4E41	RTS	0C7098		
0C703E	CA06	* READ-SELECT	0C70A6	4E75	*** UHR STELLEN	0C7098		
		* IN D6 STEHT CODE FUER INTERRUPTBETRIEB	0C70A8			0C7098		

Bild 4. Nicht beliebig eingebunden, sondern voll kompatibel zur NDR-Uhrenkarte wird die Erweiterung durch dieses Programm

0C7206	102B 0012	0C7206	102B 0012	MOVE.B #12(A0),D0	0C7284	10FC 002E	0C7284	10FC 002E	MOVE.B #',(A0)+
0C720A	0440 0030	0C720A	0440 0030	SUB #430,D0	0C7288	103C 000A	0C7288	103C 000A	MOVE.B #10,D0
0C720E	1340 000C	0C720E	1340 000C	MOVE.B D0,#C(A1)	BSR GET	BSR GET	BSR GET	BSR GET	BSR GET
0C7212		0C7212			0C729C	6100 0064	0C729C	6100 0064	MOVE.B #9,D0
0C7216	1028 0013	0C7216	1028 0013	MOVE.B #13(A0),D0	0C72A4	6100 0064	0C72A4	6100 0064	BSR GET
0C721A	0440 0030	0C721A	0440 0030	SUB #430,D0	0C72F8	10FC 002E	0C72F8	10FC 002E	MOVE.B #',(A0)+
0C721E	1340 0008	0C721E	1340 0008	MOVE.B D0,#R(A1)	0C72C0	103C 000C	0C72C0	103C 000C	MOVE.B #12,D0
0C721E		0C721E			BSR GET	BSR GET	BSR GET	BSR GET	BSR GET
0C721E	41ED 0FEA	0C721E	41ED 0FEA	LEA OUT1(A5),A0	0C72D4	103C 0008	0C72D4	103C 0008	MOVE.B #11,D0
0C7222	4280	0C7222	4280	CLR.L D0	0C72E0	10FC 0020	0C72E0	10FC 0020	MOVE.B #',(A0)+
0C7226	4281	0C7226	4281	CLR.L D1	0C72E4	10FC 0020	0C72E4	10FC 0020	MOVE.B #',(A0)+
0C7226		0C7226			BSR GET	BSR GET	BSR GET	BSR GET	BSR GET
0C7226		0C7226			0C72F0	103C 0004	0C72F0	103C 0004	MOVE.B #4,D0
0C7226	121B	0C7226	121B	MOVE.B (A0)+,D1	BSR GET	BSR GET	BSR GET	BSR GET	BSR GET
0C7228	6100 FE56	0C7228	6100 FE56	BSR UPRNITE	0C72F4	6100 0034	0C72F4	6100 0034	MOVE.B #',(A0)+
0C722C	5200	0C722C	5200	ADDD.B #1,D0	0C72F8	10FC 003A	0C72F8	10FC 003A	MOVE.B #',(A0)+
0C722E	0C00 000C	0C722E	0C00 000C	CHP.B #4C,D0	0C72FC	103C 0003	0C72FC	103C 0003	MOVE.B #3,D0
0C7232	6E00 0006	0C7232	6E00 0006	BET SETEND	BSR GET	BSR GET	BSR GET	BSR GET	BSR GET
0C7236	6000 FFE6	0C7236	6000 FFE6	BRA SETLOOP	0C7304	103C 0002	0C7304	103C 0002	MOVE.B #2,D0
0C723A		0C723A			BSR GET	BSR GET	BSR GET	BSR GET	BSR GET
0C723A		0C723A			0C7308	6100 0020	0C7308	6100 0020	MOVE.B #',(A0)+
0C723A	4E75	0C723A	4E75	SETEND:	0C730C	10FC 003A	0C730C	10FC 003A	MOVE.B #',(A0)+
0C723C		0C723C		RTS	0C7310	103C 0001	0C7310	103C 0001	MOVE.B #1,D0
0C723C		0C723C			BSR GET	BSR GET	BSR GET	BSR GET	BSR GET
0C723C		0C723C			0C7314	6100 0014	0C7314	6100 0014	MOVE.B #0,D0
0C723C		0C723C			BSR GET	BSR GET	BSR GET	BSR GET	BSR GET
0C723C		0C723C			0C7318	103C 0000	0C7318	103C 0000	MOVE.B #0,D0
0C723C	363C 0001	0C723C	363C 0001	RTCSHOW:	0C731C	6100 000C	0C731C	6100 000C	BSR GET
0C7240	41ED 0FCA	0C7240	41ED 0FCA	MOVE #1,D3	0C7320	10FC 0000	0C7320	10FC 0000	MOVE.B #0,(A0)+
0C7244	203C 00000011	0C7244	203C 00000011	LEA OUT(A5),A0	0C7324	4CDF 0100	0C7324	4CDF 0100	MOVEM.L (A7)+,A0
0C724A	243C 000000F7	0C724A	243C 000000F7	MOVE.L #247,D2	0C7328	4E75	0C7328	4E75	RTS
0C7250	223C 0000017C	0C7250	223C 0000017C	MOVE.L #360,D1	GET:		GET:		
0C7256	363C 0001	0C7256	363C 0001	MOVE #1,D4	CLR.L D1		CLR.L D1		
0C725A	1E3C 003F	0C725A	1E3C 003F	MOVE.B #INTD15,D6	BSR UPREAD		BSR UPREAD		
0C725E		0C725E			AND1.B #4F,D1		AND1.B #4F,D1		
0C725E		0C725E			CHP1.B #5,D0		CHP1.B #5,D0		
0C725E	48E7 E000	0C725E	48E7 E000	LESEN:	BNE NH10		BNE NH10		
0C7262	6100 0014	0C7262	6100 0014	MOVEM.L D0-D2,-(A7)	AND1.B #43,D1		AND1.B #43,D1		
0C7266	6100 0014	0C7266	6100 0014	MOVEM.L B3,-(A7)	RTS		RTS		
0C726A	4CDF 0008	0C726A	4CDF 0008	BSR LESELP	SHOW:		SHOW:		
0C726E	4CDF 0007	0C726E	4CDF 0007	MOVEM.L (A7)+,B3	MOVEM.L D0-D1,-(A7)		MOVEM.L D0-D1,-(A7)		
0C7272	0C03 0001	0C7272	0C03 0001	MOVEM.L (A7)+,D0-D2	BSR WRITEBT		BSR WRITEBT		
0C7276	6700 0000	0C7276	6700 0000	CHP.B #1,D3	MOVE.B D4,D0		MOVE.B D4,D0		
0C727A	4E75	0C727A	4E75	BEG SHOW	NEG1.B D4		NEG1.B D4		
0C727C		0C727C		RTS	AND1.B #3,D4		AND1.B #3,D4		
0C727C		0C727C			MOVE.B #1,D1		MOVE.B #1,D1		
0C727C		0C727C			MOVE #1MENPAGE,D7		MOVE #1MENPAGE,D7		
0C727C	48E7 0080	0C727C	48E7 0080	LESEL:	TRAP #1		TRAP #1		
0C7280	4280	0C7280	4280	MOVEM.L A0,-(A7)	MOVEM.L (A7)+,D0-D1		MOVEM.L (A7)+,D0-D1		
0C7286	303C 0006	0C7286	303C 0006	CLR.L D0	LEA TABTE,A1		LEA TABTE,A1		
0C728A	4281	0C728A	4281	LEA TABTAB(IPC),A2	CHP1.B #403,(A1)		CHP1.B #403,(A1)		
0C728C	6100 FDE2	0C728C	6100 FDE2	MOVE #6,D0	RTS		RTS		
0C7290	0201 000F	0C7290	0201 000F	CLR.L D1	*** DATEN-TABELLEN		*** DATEN-TABELLEN		
0C7294	C2FC 0002	0C7294	C2FC 0002	BSR UPREAD					
0C7298	30C1	0C7298	30C1	AND1.B #8F,D1					
0C729E	4CB2 0002 1000	0C729E	4CB2 0002 1000	MULU #2,D1					
0C72A0	10FC 0020	0C72A0	10FC 0020	MOVEM.W #0(A2,D1),N1,D1					
0C72A4	103C 0008	0C72A4	103C 0008	MOVE.W D1,(A0)+					
0C72A8	6100 0080	0C72A8	6100 0080	MOVE.B #',(A0)+					
0C72AC	103C 0007	0C72AC	103C 0007	BSR GET					
0C72B0	6100 0078	0C72B0	6100 0078	MOVE.B #7,D0					
				BSR GET					

```

0C7374 406944F467253 TABTAB: * WOCHENABSTABELLE ( 0 = MITTWOCH )
0C7378 61536F406F4469 DC.B 'M', 'Do', 'Fr', 'Sa', 'So', 'Mo', 'Di'
0C7382 *** ABLAGEBEIET FUER TEXTE
0C7382 55687220736574 TEXT1: DC.B 'Uhr setzen', 0
0C7389 76656AE0I TEXT2: DC.B 'Datenformat = Tag/Std/Min/Sec/Do/Mo/Ja', 0
0C7394 72606174203020 TEXT3: DC.B 'Eingabeteil er', 0
0C73A2 692F5362E74461
0C73A9 2F406F2F406100
0C73B0 45696E67616265 TEXT3: DC.B 'Eingabeteil er', 0
0C73B7 66656660657220
0C73BE 2100
0C73C0 466F766617420 TEXT4: DC.B 'Format : B. Mo/12/30/00/13/00/86', 0
0C73C7 74E2042E2040
0C73CE 6F2F31322F3330
0C73D5 2F30302F31332F
0C73DC 30302F3036300
0C73E2 DS 0
0C73E2 *****
0C73E2 ***** PROGRAMM ZUM AUFRUF DER RTC-FUNKTIONEN USER=USERTRAP
0C73E2 *****
0C73E2 TRAPTRAP EDU #9C * TRAP 7 EXCEPTION-VEKTORTABELLE 68008
0C73E2 TRAPTRAP EDU #A16 * IN GRUNDPROGRAMM-SPEICHER
= 000000A16
0C73E2 *** USER-TRAP HANDLER
0C73E2
0C73E2 * IMPLEMENTIERTE FUNKTIONEN (CODE IN D7.M) :
0C73E2 * #700 = DATEN VON RTC LESEN (OO GEN, ADRESSE, D1 DATEN RETOUR)
0C73E2 * #701 = DATEN IN RTC SCHREIBEN (OO GEN, ADRESSE, D1 DATEN FUER RTC)
0C73E2 * #702 = UHR LESEN (OO GROSSE, D1, D2 KOORDINATEN, A0 FUERER, D4 CRT / PUFFER)
0C73E2 * #703 = UHRZEITANZEIGE AUF BILDSCHIRM
0C73E2 * #704 = UHR STELLEN (MENUES/EUER)
0C73E2 * #705 = RTC-DATEN IN HEX-FORMAT IN PUFFER LESEN (PUFFERADRESSE IN A0)
0C73E2 * #706 = REFERENZSIKNALE AUF DO-D3
0C73E2 * #707 = EMULATION BEUTUHR
0C73E2 * #708 = EMULATION BEUTUHR
0C73E2
0C73E2 USERTRAP:
0C73E2 CMP #700, D7 * NR. DER GEN. FUNKTION IN D7.M
0C73E2 BNE UT1 * DATEN VON RTC LESEN ?
0C73E2 BSR UPREAD
0C73E2 RTE
0C73E2
0C73E2 UT1:
0C73E2 CMP #701, D7 * DATEN IN RTC SCHREIBEN ?
0C73E2 BNE UT2
0C73E2 BSR UPWRITE
0C73E2 RTE
0C73E2
0C73E2 UT2:
0C73E2 CMP #702, D7 * UHR LESEN (CRT / PUFFER)
0C7402 6600 0008 BNE UT3
0C7406 6100 FE36 BSR LESEN
0C740A 4E73
0C740C

```

```

0C740C 0C740C 0C47 028F UT3:
0C7410 6600 0008 CMP #703, D7
0C7414 6100 FE26 BNE UT4
0C7418 4E73 BSR RTCSHOW
0C741A
0C741A UT4:
0C741A 0C47 0220 CMP #704, D7
0C741E 6600 0008 BNE UT5
0C7422 6100 FE94 BSR SETUHR
0C7426 4E73 RTE
0C7428
0C7428 UT5:
0C7428 0C47 02C1 CMP #705, D7
0C742C 6600 0008 BNE UT6
0C7430 6100 0064 BSR HEUREAD
0C7434 4E73 RTE
0C7436
0C7436 UT6:
0C7436 0C47 02C2 CMP #706, D7
0C743A 6600 0008 BNE UT7
0C743E 6100 003C BSR REFON
0C7442 4E73 RTE
0C7444
0C7444 UT7:
0C7444 0C47 02C3 CMP #707, D7
0C7448 6600 0008 BNE UT8
0C744C 6100 0166 BSR ESETUHR
0C7450 4E73 RTE
0C7452
0C7452 UT8:
0C7452 0C47 02C4 CMP #708, D7
0C7456 6600 0008 BNE UT9
0C745A 6100 0194 BSR ESETUHR
0C745E 4E73 RTE
0C7460
0C7460 UT9:
0C7460 3E3C FFFF * FEHLERCODE, WENN FUNKTION NICHT DA
0C7464 4E73 RTE
0C7466
0C7466 *** INITIALISIEREN DES USER-TRAP
0C7466
0C7466 OS 0
0C7466 DC.M $59A4 * HEADER ZUR KENNUNG, DAMIT KOENNE ANDERE
0C7466 DC.L $221AA655 * PROGRAMME DIESE ROUTINE SICHEN UND AUSFUEHREN
0C746C
0C746C TRAPTRAP1:
0C746C LEA USERTRAP, A0 * ADRESSE IN TRAP-VEKTOR EINTRAGEN
0C7470 2308 0000009C * DAZU ADRESSE IN A0 HOLEN UND VON DOBT
0C7476 2948 0A16 * IN ENTSPRECHENDE SPEICHER EINTRAGEN,
0C747A 4E73 RTS * DADURCH RELOKATIV !
0C747C * BENUTZT WIRD TRAP #7
0C747E
0C747E *****
0C747E ***** PROGRAMM FUER TIMER-OUTPUT AUF DO-D4
0C747E *****
0C747C
0C747C * DO = 1024 HZ, D1 = /1 SEK., D2 = /1 MIN., D3 = /1 STD.
0C747C * IMPULSE D0 = POSITIVE LOGIK - IMPULSLAENGE = 488,3 US ( 1 2U 1 )
0C747C * IMPULSE D1-D3 = NEGATIVE LOGIK - IMPULSLAENGE = 122,1 US
0C747C
0C747C * SCHLIESST REFERENZIMPULSE AUF DO-D3
0C747C

```

0C747C	103C 00FE	HEXRLP: MOVE.B #4FE, D0	= FFFFFFF0	GDP	EDU \$FFFFFF0	* ADRESSE GDP
0C7480	1300 FFFFFFF40	MOVE.B D0, DOUT	= 00000217	VIERPAG EDU \$217		* HIER STEHT DIE AKT. LESESEITE (REL. -> AS)
0C7486	6100 FBFA	BSR AMRTE	= 00000218	WRTPAGE EDU \$218		* HIER STEHT DIE AKT. WRITPAGE (REL. -> AS)
0C748A	103C 0033	MOVE.B #100110011, D0	0C74FB	RTICINI:		* HIER WIRD TAG, DATUM UND UHRZEIT ANGEZEIGT
0C748E	1300 FFFFFFF41	MOVE.B D0, CONTR	0C74FB	MOVE.L D0-D7/A0-A7, -A7)		* ALLE REGISTER RETTEN
0C7494	4E75	RTS	0C74FC	MOVE.B #1INTSEK, D6		* INT HARDWAREMÄSSIG FREITREIBEN
0C7496		*****	0C7500	MOVE #0, D3		* GEHT UEBER ROUTINE LESEN, ANZEIGE-CODE IN D3
0C7496		***** PROGRAMM ZUM AUSLESEN DER RTC-DATEN IN HEX-FORMAT IN PUFFER A0 -> PUFFER	0C7500	LEA OUT(A5), A0		* UND PUFFERADRESSE IN A0
0C7496		*****	0C7508	BSR LESEN		* DANN IN REGISTER RETTEN
0C7496		*****	0C750C	BSR GPPSAVE		* GPP - REGISTER RETTEN
0C7496		*****	0C7510	MOVE.L D5-D6, -A7)		* MERKEN BENDET107
0C7496	43FA 0030	HEXREAD: LEA READTAB(PC), A1	0C7514	MOVE.B VIERPAG(A5), D5		* UM SCHREIB- UND LESESEITE ZU RETTEN
0C749A	1300 003F	MOVE.B #INTDIS, D6	0C7518	MOVE.B #0, D0		* NUN AUSGABE AUF SEITE 0
0C749E		HEXRLP:	0C751C	MOVE #MEMPAGE, D7		
0C749E	1019	MOVE.B (A1), D0	0C7520	TRAP #1		
0C749E	0000 00FF	CHP.B #FFF, D0	0C7524	LEA OUT(A5), A0		* PUFFERADRESSE
0C74A4	6700 001C	BED HEXREND	0C752A	ADDA.L #3, A0		* WOCHENTAG NICHT ANZEIGEN (PASST NICHT !)
0C74AB	6100 FB66	BSR UPREAD	0C752E	MOVE.B #11, D0		* GROSSE
0C74AC	0201 000F	AND.L B #F, D1	0C7538	MOVE #395, D1		
0C74B0	0000 0005	CMPL.B #5, D0	0C753C	MOVE #1, D2		
0C74B4	6600 0006	BNE HEXRLP1	0C7540	BSR WRITECT		* AUSGABE AUF BILDSCHIRM
0C74B8	0201 0003	AND.L B #3, D1	0C7544	MOVE.B #0, D0		* DANN NOCH AUF SEITE 1
0C74BC		HEXRLP1:	0C7548	MOVE.B #0, D1		
0C74BC	10C1	MOVE.B D1, (A0)+	0C754C	TRAP #1		
0C74BE	6000 FFDE	BSR HEXRLP	0C7550	LEA OUT(A5), A0		* PUFFERADRESSE
0C74C2		HEXREND:	0C7556	ADDA.L #3, A0		* WOCHENTAG NICHT ANZEIGEN (PASST NICHT !)
0C74C2	10BC 00FF	MOVE.B #FF, (A0)	0C755C	MOVE.B #11, D0		* GROSSE
0C74C6	4E75	RTS	0C7560	MOVE #395, D1		
0C74C8		READTAB:	0C7564	MOVE #1, D2		
0C74CB		DC.B #6, #8, #7, #6, #5, #4, #3, #2, #1, #0, #FF	0C7568	BSR WRITECT		* AUSGABE AUF BILDSCHIRM
0C74CB	06 08 07 0A 09		0C756C	MOVE.B D5, VIERPAG(A5)		* DANN SCHREIB- UND LESESEITE RETOUR
0C74CD	0C 08 05 04 03		0C7570	MOVE.B D6, WRTPAGE(A5)		
0C74D2	02 01 00 FF		0C7574	MOVE.L (A7), D5-D6		* REGISTER DTD.
0C74D6		*****	0C7578	BSR GPPLOAD		* UND GDP - REGISTER RETOUR
0C74D6		***** PROGRAMM ZUM BETRIEB ES RTC UEBER INTERRUPT	0C757C	MOVE.L (A7)+, D0-D7/A0-A7		* REGISTER RETOUR
0C74D6		*****	0C7580	RTS		
0C74D6		*****	0C7582	*** RETTEN DER GDP - REGISTER		
0C74D6		*****	0C7582	GDPSAVE:		
0C74D6	00000074	INTSJMP EDU #74	0C7582	LEA GDP+1, A0		* RETTET DIE REGISTER IN EINEM PUFFER
0C74D6	00000008	INTSJMP EDU #08	0C7582	LEA OUT1(A5), A1		* ADRESSE GDP, NUR CND NICHT RETTEN
0C74D6		*** INITIALISIEREN DES INTERRUPTS	0C7588	MOVE #15-4-1, D1		* HIER REGISTER ABLEGEN
0C74D6		***	0C7590	SAVEP:		* ANZAHL
0C74D6	55AA	DC.W #55AA	0C7590	MOVE.B (A0)+, (A1)+		
0C74D6	1122A55	DC.L #1122A55	0C7590	DBRA D1, SAVEP		
0C74D6		INTININI:	0C7590	RTS		* WENN ALLES ABGESPEICHERT
0C74D6	41FA 001A	LEA RTICINI(PC), A0	0C7598	*** ZURUECKSCHREIBEN DER GDP-REGISTER		
0C74E0	230B 00000074	MOVE.L A0, INTSJMP	0C7598	GPPLOAD:		
0C74E6	2B4B 000B	MOVE.L A0, INTSJMP(A5)	0C7598	LEA GDP+1, A0		* GDP REGISTER ZURUECK
0C74EA	46FC 2000	MOVE #2000, SR	0C759E	LEA OUT1(A5), A1		* ADRESSE GDP, NUR CND NICHT RETTEN
0C74EE	13FC 007F	MOVE.B #INTSEK, CONTR	0C75A6	MOVE #15-4-1, D1		* HIER REGISTER ABLEGEN
0C74F2	FFFFFF41	RTS	0C75A6	LOADP:		* ANZAHL
0C74F6	4E75	*** BENUTZT WIRD INTLVL 5 = LTB. /INT				
0C74F8		*** INTERRUPTROUTINE				
0C74F8						

0C75A6	3E3C 0012	MOVE #1,017,07	* WARTEN BIS 50P READY	0C761B	101B	ESETLP:	* WERT AUS PUFFER IN D0
0C75A8	4E41	TRAP #1		0C761A	0C11 0006	MOVE.B (A0)+,D0	* TESTEN OB WOCHENTAG
0C75AC	10D9	MOVE.B (A1)+,(A0)+		0C761E	6700 0032	CMP.B #6,(A1)	* DANN ANDERE FUNKTION
0C75AE	51C9 FFF6	DBRA D1+,LOADLP		0C7622	6100 003A	BSR BCDHEX	* NUM BCD-ZAHL IN 2 HEX-ZAHLEN WANDELN
0C75B2	4E75	RTS	* WENN ALLES ZURUECKGESCHRIEBEN	0C7626	1400	MOVE.B D0,D2	* IN D0 SIND EINER, IN D1 ZEHNER
0C75B4		*****		0C7628	1019	MOVE.B (A1)+,D0	* ADRESSE AUS TABELLE IN D0
0C75B8		***** HIER PROGRAMME ZUR NACHBILDUNG DER ORIGINAL-UHR MIT E05-16		0C762A	0C00 0005	CMP.B #5,D0	
0C75BA		***** HIER PROGRAMME ZUR NACHBILDUNG DER ORIGINAL-UHR MIT E05-16		0C762E	6600 0006	BNE ESETLP1	
0C75BC		*****		0C7632	08C1 0003	BSET #3,D1	* FUER 24 STD. UHR
0C75BD		*** ROUTINE LEGT DIE DATEN WIE DIE GRUNDPROGRAMMRoutine @GETUHR IN BCD-FORMAT		0C7636		ESETLP1:	
0C75BE		* IN DURCH A0 ADRESSIERTEN PUFFER AB		0C7638	2F02	MOVE.L D2,-(A7)	* DZ RETTEN, DA IN UPRARTE BENUTZT
0C75B8	48E7 FFC0	EBETUHR:		0C763C	6100 FA46	BSR UPRARTE	
0C75BA	43FA 008B	MOVEM.L A0-A1/D0-D7,-(A7)	* REGISTER RETTEN	0C763E	241F	MOVE.L (A7)+,D2	* NAECHSTE ADRESSE
0C75BC	42B0	LEA ENUTAR(PC),A1	* LESETABELLE IN A0	0C7640	1202	MOVE.B D2,D1	* WERT IN D1
0C75BE	1C3C 003F	CLR.L D0		0C7642	6100 F83C	BSR UPRARTE	* SCHREIBEN
0C75C2		MOVE.B #INTDIS,D6		0C7646	0C10 00FF	CMP.B #FFF,(A0)	* \$FF IST ENDEZEICHEN
0C75C4	4281	CLR.L D1		0C764A	6700 FBFE	BEQ ESETEND	
0C75C6	6100 FAAB	MOVE.B (A1)+,D0	* ADRESSE IN D0	0C764E	6000 FFCB	BRA ESETLP	
0C75CA	0201 800F	BSR UPREAD	* DANN WERT LESEN	0C7652		ESETWT:	
0C75CE	0C00 0005	AND.B #8F,D1		0C7652	1200	MOVE.B D0,D1	* WOCHENTAG WIRD SO GEBSETZT, DA NUR EINSTELLIG
0C75D0	6600 0006	CMP.B #5,D0		0C7654	1019	MOVE.B (A1)+,D0	
0C75D2	0201 0003	BNE EBETLP1	* WENN STD. ZEHNER, BIT 3 RUECKSETZEN WEGEN ZAH	0C7656	6100 FA2B	BSR UPRARTE	
0C75D4		AND.I.B #*3,D1		0C765A	6000 FFCB	BRA ESETLP	
0C75D8		EBETLP1:		0C765E		BCDHEX:	
0C75DA	3401	MOVE.D1,D2	* UND RETTEN	0C765E	3200	MOVE.D0,D1	* WANDELT BCD-ZAHL AUS D0 IN HEXZAHL
0C75DC	0C00 0006	CMP.B #6,D0	* TESTEN OB WOCHENTAG	0C7660	0240 000F	AND #*0F,D0	* WERT AUCH IN D1
0C75DE	6700 0022	BEQ EBETWT		0C7664	0241 00F0	AND #*F0,D1	* EINER IN D0
0C75E0	1019	MOVE.B (A1)+,D0	* NAECHTE ADRESSE HOLEN	0C7668	EB59	ROR #4,D1	* ZEHNEN IN D1
0C75E4	6100 FB8B	BSR UPREAD	* UND HOLEN	0C766A	4E75	RTS	* UND NOCH ROTIEREN
0C75E8		AND.I.B #*F,D1		0C766C		HEXBCD:	
0C75EA	0201 000F	MOVE.D1,D0	* EINER IN D0	0C766C	E919	ROL.B #4,D1	* WANDELT HEXZAHL AUS D0 + D1 IN BCD-ZAHL IN D0
0C75EC	3001	MOVE.D2,D1		0C766E	B041	OR D1,D0	
0C75F0	3202			0C7670	4E75	RTS	
0C75F2	6100 007B	BSR HEXBCD	* UND IN BCD-ZAHL WANDELN	0C7672		ENUTAR:	
0C75F6	10C0	MOVE.B D0,(A0)+	* DANN DEN BCD-WERT IN PUFFER	0C7672	05 04 03 02 0B	DC.B #5,#4,#3,#2,#1,#0,\$F	
0C75F8	0C11 00FF	CMP.B #*FF,(A1)	* ENDE	0C7677	07 0A 09 0C 0B	DC.B #5,#4,#3,#2,#1,\$F	
0C75FC	6700 000C	BEQ EBETEND		0C767C	06 01 00 FF		
0C7600	6000 FFC0	BRA EBETLP		0C7680		DS 0	
0C7604				0C7680		*****	
0C7608		EBETWT:	* WOCHENTAG BEDARF KEINER WANDELUNG	0C7680		***** HIER TESTPROGRAMME UND BEISPIELE	
0C760C	10C1	MOVE.B D1,(A0)+	* DANN WEITER	0C7680		*****	
0C760E	6000 FFB9	BRA EBETLP		0C7680		*** TESTROUTINEN FUER AUFRUF UEBER TRAP	
0C7610		EBETEND:		0C7680		TEST:	
0C7614	4EDF 03FF	MOVEM.L (A7)+,A0-A1/D0-D7	* PROGRAMM UEBERTRAEGT DIE BCD-DATEN AUS DURCH A0 ADRESSIERTEN PUFFER WIE DIE	0C7680	6100 0052	BSR TRAPPEND	* USERTRAP INITIALISIEREN
0C7618	4E75	RTS	* GRUNDPROGRAMMRoutine @GETUHR	0C7684	3E3C 02BE	MOVE #702,D7	* CODE IN D7
0C761C				0C7688	363C 0001	MOVE #1,D3	* AUSGABE, DAZU CODE IN D3
0C7620				0C768C	41EB 0FCA	LEA OUT(A5),A0	* UND PUFFERADRESSE IN A0
0C7624	48E7 FFC0	EBETUHR:		0C7690	203C 0000003B	MOVE.L #*3B,D0	* KOORDINATEN BELEGEN
0C7628	43FA 805C	MOVEM.L A0-A1/D0-D7,-(A7)		0C769C	243C 0000000A	MOVE.L #10,D2	
0C762C		LEA ENUTAR(PC),A1		0C769C	223C 0000000A	MOVE.L #10,D1	



```

0C76A2 3B3C 0001 MOVE #1,D4 * SEITENZAHLER VORBELEGEN
0C76A6 1C3C 003F MOVE.B #INTDIS,D6
0C76AA 4E47 TRAP #7 * AUFRUF NUN UEBER USERTRAP
0C76AC
0C76AC 3E3C 0010 MOVE #!CLR,D7
0C76B0 4E41 TRAP #1
0C76B2 3E3C 02D0 MOVE #704,D7
0C76B6 4E47 TRAP #7
0C76B8
0C76B8 3E3C 0010 MOVE #!CLR,D7
0C76BC 4E41 TRAP #1
0C76BE 3E3C 02BF MOVE #703,D7
0C76C2 42B4 CLR.L D4
0C76C4 4E47 TRAP #7
0C76C6 4E75 RTS
0C76C8
0C76C8 *** TESTROUTINE FUER HEXREAD
0C76C8
0C76C8 TEST1:
0C76C8 41F9 00005000 LEA $5000,A0
0C76CE 6100 FDC6 BSR HEXREAD
0C76D2 4E75 RTS
0C76D4
0C76D4 *** PROGRAMM ZUM FINDEN UND STARTEN DER ROUTINE TRAPINIT
0C76D4
0C76D4 TRAPFIND: * PROGRAMM SUCHT HEADER DER INITIALISIERUNGS-
0C76D4 41F9 00000400 LEA $400,A0 * ROUTINE
0C76DA
0C76DA TRAPFLP:
0C76DA B1FC FFFFFFFF CMPA.L #FFFFFFF,A0 * WENN SPEICHER-ENDE ERREICHT
0C76E0 6700 001C BEQ FLPERRR * DANN FEHLER
0C76E4 0C5B 55AA CMP.W #55AA,(A0)+ * SOLANGE BIS HEADER BEFUNDEN
0C76E8 6600 FFF0 BNE TRAPFLP
0C76EC D1FC 00000002 ADDA.L #2,A0 * NUN PRUEFEN, OB REST AUCH OK
0C76F2 0C5B AA55 CMP.W #221AA55,(A0)+
0C76F6 6600 FFE2 BNE TRAPFLP * WEITER, WENN NICHT OK,
0C76FA 4E90 JSR (A0) * SONST AUSFUEHREN
0C76FC 4E75 RTS
0C76FE
0C76FE FLPERRR: * ALS FEHLER-CODE STEHT NUN #FFFFFFF IN A0
0C76FE 4E75 RTS
0C7700
0C7700 *** PROGRAMM ZUM FINDEN UND STARTEN DER ROUTINE INTINIT
0C7700
0C7700 INTFIND: * PROGRAMM SUCHT HEADER DER INITIALISIERUNGS-
0C7700 41F9 00000400 LEA $400,A0 * ROUTINE
0C7706
0C7706 INTFLP:
0C7706 B1FC FFFFFFFF CMPA.L #FFFFFFF,A0 * WENN SPEICHER-ENDE ERREICHT
0C770C 6700 FFF0 BEQ FLPERRR * DANN FEHLER
0C7710 0C5B 55AA CMP.W #55AA,(A0)+ * SOLANGE BIS HEADER BEFUNDEN
0C7714 6600 FFF0 BNE INTFLP
0C7718 0C9B 1122AA55 CMP.L #1122AA55,(A0)+
0C771E 6600 FFE6 BNE INTFLP
0C7722 4E90 JSR (A0)
0C7724 4E75 RTS
0C7726
0C7726 *****
0C7726 *** ROUTINE ZUM IMPLEMENTIEREN IN GRUNDPROGRAMM
0C7726 *****
0C7726
0C7726 * DIE ROUTINEN KOENNEN AUF FOLGENDE WEISE IN DAS GRUNDPROGRAMM IMPLEMENTIERT
0C7726 * WERDEN :
0C7726 * 1. GRUNDPROGRAMM INS RAM VERSCHIEBEN (Z.B. $20000
0C7726 * 2. GRUNDPROGRAMM NEU STARTEN (MENUE = 2 PROGRAMM STARTEN ADR. #233F4)
0C7726 * 3. RTC ROUTINEN LADEN UND ASSEMBLIEREN
0C7726 * 4. GRUNDPROGRAMMEPRDM 3 NEU BRENNEN (VON $24000 - $25FFF)
0C7726 * 5. RTC-ROUTINEN IN EPROM BRENNEN ( $A7000 - $A7FFF NACH $1000 !!)
0C7726 * 6. EPROM AUF ADRESSE #C6000 STECKEN (4.SOCKET BANK C)
0C7726 * 7. GRUNDPROGRAMM-EPROM TAUSCHEN
0C7726 * DIE ADRESSEN KOENNEN NATUERLICH ANGEPAESST WERDEN
0C7726
0E4CF0 ORG @SETUHR
0E4CF0 4EF9 000C75B4 JMP EGETUHR
0E4CF6
0E4D2E ORG @SETUHR
0E4D2E 4EF9 000C7610 JMP ESETUHR
0E4D34
0E4D34 ENDEA:
0E4D34
0E4D34 END.
0E9132 Ende-Symboltabelle

```

Uhr setzen
Datenformat = Tg/St/Mi/Se/Da/Mo/Ja

Sa/12/35/00/20/09/86

Bild 5. Ein Beispiel für die Bedienung:
Das Stellen der Uhr

REFON

Diese Routine schaltet die Referenzsignale auf die Datenleitungen. Damit läßt sich ebenfalls jede Minute ein Interrupt auslösen oder, durch Abfrage der Schnittstelle, eine Timerfunktion realisieren. Bild 6 zeigt die genauen Zeiten.

ESETUHR

Mit diesem Programm wird die Routine \$SETUHR des Grundprogramms emuliert. So kann die Uhr auf die gleiche Weise gestellt werden wie bei der Originalkarte mit dem E05-16. Die Daten werden im BCD-Format in einen durch A0 adressierten Puffer gelegt und von der Routine in den Baustein geschrieben, wobei sie in die für den RTC verständliche Form übertragen werden.

EGETUHR

Tag, Datum und Uhrzeit werden im BCD-Format in einen Puffer geschrieben; die Pufferadresse steht in A0. Da sich dieses Programm exakt wie die Grundprogramm-Routine \$GETUHR verhält, können alle Programme, die für die Original-Uhrenkarte geschrieben wurden, auch mit dem RTC-58321 verwendet werden. Man muß dazu lediglich im EPROM die Sprünge auf die Originalroutinen im Grundprogramm durch solche auf die neuen Unterprogramme ersetzen.

Da im Listing mit Kommentaren nicht gespart wurde, dürfte das Umschreiben für andere Rechner kein Problem sein. Man sollte auch hier mit den Elementarfunktionen beginnen und die übrigen Programmteile darauf aufbauen.

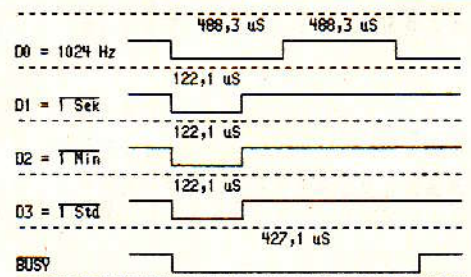


Bild 6.
Diese Zeiten
sind für den
RTC-58321
spezifiziert