

Use of the MC68HC68T1 RTC with M6805 Microprocessors

Prepared by
 Peter TOPPING
 Microprocessor Applications
 Motorola, East Kilbride

INTRODUCTION

The MC68HC68T1 is a real time clock (RTC) which also contains 20 bytes of RAM. Communication is via a serial port making the T1 suitable for use with single-chip microprocessors. The features on the MC68HC68T1 are similar to those on the MC146818 which, with its multiplexed-bus, is not so appropriate for use with MCUs. The MC68HC68T1 has, in addition, watchdog and power fail capabilities.

The MC68HC68T1 can be controlled using a clocked serial port, typically an SPI, but can also use port lines. This would be of use with a micro with no serial capability or in an application where the serial port was dedicated to another task. The example software was developed for use on an MC68HC05C4 and includes routines to use either the SPI or port lines.

Using one or other of these methods the code could be used in any 6805 microprocessor. The code would need to be modified to exclude the STOP instruction in an HMOS processor.

The example software constitutes an alarm clock using an 8-digit triplexed LCD display with ICM7231B driver, an MC68HC805C4 microprocessor and the RTC as shown in Figure 1. The software leaves the micro in stop mode until it is interrupted by the MC68HC68T1 in order to update the display. This is selected to be 1Hz by the software. The watchdog requirement cannot be met while the micro is in stop mode and is consequently not enabled in the software. As the software is for a battery supplied clock the power fail capability is also not used.

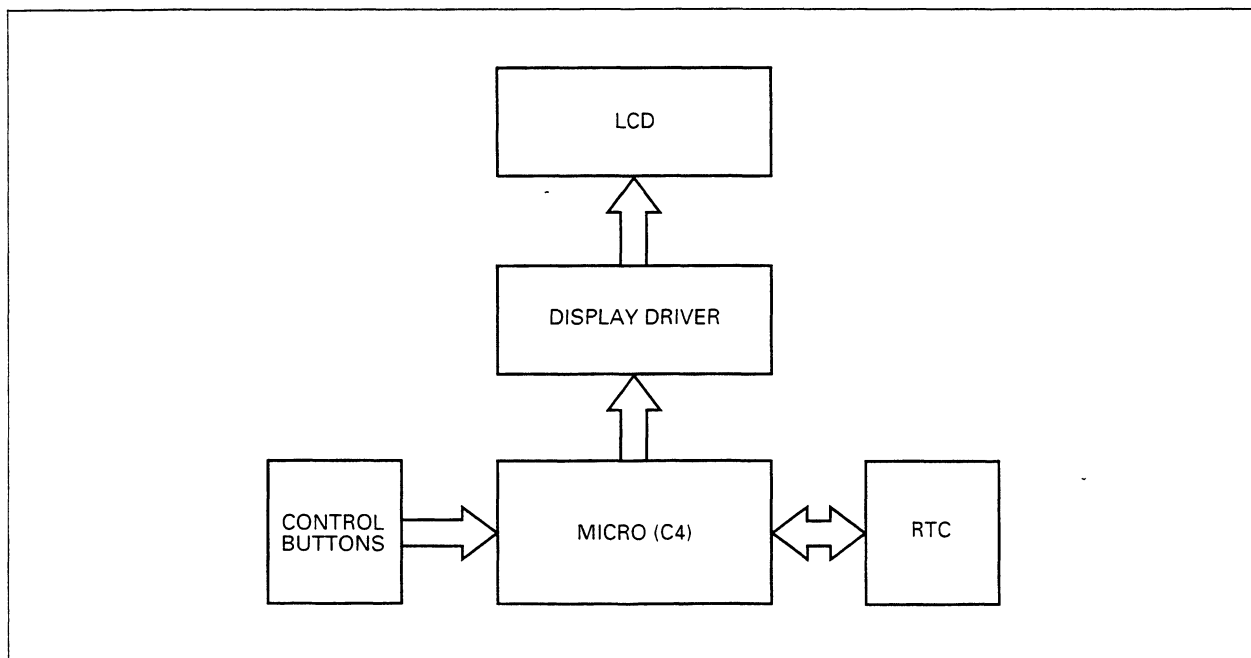


Figure 1.



CIRCUIT

Figure 2 shows the circuit used. The MC68HC68T1 can be connected either to port lines or to the SPI. With the two jumpers on MOSI and SCK connected between 1 & 2 the port lines are used. The third jumper on PD7 should also be connected between 1 & 2 to indicate to the software that I/O port communication is required. Jumper connection between 2 & 3 selects SPI communication. As MISO is an output from the RTC it can be connected in parallel to both the MISO and PC7 inputs on the C4.

The RTC uses a 32KHz clock for low power dissipation. The RTC and display driver consume about 65 microamps at 5.0v while the 2 milliamps taken by the C4 at a duty cycle of less than 1% contributes only another 15 microamps. The crystal frequency used on the C4 does not significantly affect overall consumption, 2MHz was used.

The ICM7231B LCD driver requires each character to be addressed through pins A0, A1 & A2 and the appropriate data written to pins D0, D1, D2, & D3. As the display is written to after the RTC registers have been read the port lines PC0 & PC1 can be used for both interfaces. The annunciators and decimal points (the latter not used in the example software) are controlled by their own pins. The ICM7231B driver used in the clock, displays -, E, H, L, P and blank to represent A, B, C, D, E and F respectively. The "A" version of this driver displays A through F normally but does not display a "-" or a blank and is therefore not suitable. Contrast can be adjusted using the potentiometer on pin 2 of the ICM7231B. With the clock supplied by a battery of four zinc-carbon cells a fixed 100 kohms from pin 2 to ground was found to be satisfactory.

SOFTWARE

The first page of the software listing constitutes hardware address definitions and RAM allocation. As well as working space the RAM contains an image of the registers within the 68HC68T1. These consist of 3 time registers, 4 date registers, 3 alarm registers and 3 status/control registers. The alarm registers are different from the time/date registers in that they are write only and consequently have been treated differently in software.

Interrupts are vectored to address \$100 and the code at the start of the second page decides where the interrupt has come from. If it is not the result of a button press it must be from the MC68HC68T1 and its registers are read. Firstly the status register is read. This serves to return the interrupt line to a high and also allows the alarm bit (bit 1) to be checked to see if an alarm has occurred. If it has then bit 7 on port B is set high and the T1's clock output is enabled at 64 Hz. Either signal can be used to provide an alarm.

All the time registers are then read and displayed according to the selected display format (see below). The second page handles time displays and the third page those of date and alarm. As the alarm registers in the MC68HC68T1 are write only, alarm data is derived from the RAM of the micro.

Button 1 has two functions according to whether the clock is in the normal or the set-up mode. The mode is contained in status bit 0 in RAM location STAT. In set-up mode it increments the display format in RAM location DSPLY. In the set-up mode it increments the register currently "opened" by button 2. In this mode it sets STAT bit 1 to indicate to the main set-up routine that a change has been made. This part of the software also keeps track of the maximum permissible value for each register.

Button 2 is only used for set-up. When pressed it selects the set-up mode and displays the first register (time seconds). Subsequent presses move to the next register. A press from the last register (alarm hours) returns the clock to normal mode. Entry to the set-up routine stops further interrupts from the MC68HC68T1. The time and date registers are read from the MC68HC68T1 and so will be correct if it's power has been maintained even if the micro's RAM is not valid (alarm values are read from RAM). If a value is changed, by pressing button 1, then the new value is written to the appropriate MC68HC68T1 register when the next register is selected. Alarm values are written immediately to the MC68HC68T1 so that the displayed value is the same as that in the corresponding MC68HC68T1 alarm register.

The serial routine on page 12 communicates with the MC68HC68T1 using I/O lines and the next page shows how this can be done with less code using the SPI. The software is written for the level 1 SPI contained in the MC68HC05C4 & L6.

ALARM CLOCK OPERATION

The clock is controlled by two buttons. Button 1 selects the format of the display by sequencing through the formats shown in Table 1.

Format 5 shows the day of the week using the annunciator on the display. A "1" in register 6 (\$24) of the MC68HC68T1 which corresponds to Sunday displays at position 2 and so on through Saturday at position 8. This only appears when the display is set to date. Position 1 is reserved to indicate that the alarm is enabled and shows on all display formats if the alarm SPST switch is closed. Button 1 is also used to cancel the alarm.

Button 2 allows the various registers in the MC68HC68T1 to be set up. When this button is pressed the number in position 5 of the display indicates the register whose contents are being displayed in position 7 and 8 (see table 2). Pressing button 2 again moves to the next register without changing anything, pressing button 1 while in this mode increments the contents of the displayed register, the new contents are written into the register when button 2 is used to move to the next register.

Pressing button 2 when the last register (9) is being displayed returns the clock to the normal run mode in which the required format can be selected using button 1.

No.	Display Format
1	Time - hours (24 hr format), minutes and seconds.
2	Time - hours (24 hr format), minutes and date.
3	Time - hours (12 hr format), minutes and seconds.
4	Time - hours (12 hr format), minutes and date.
5	Date - day, month and year including day of week.
6	Alarm time in 24 hr format.

Table 1 Display Formats

Display	Address	Function	Range
0	\$20	seconds	0 - 59
1	\$21	minutes	0 - 59
2	\$22	hours	0 - 23
3	\$23	day of week	1 - 7 (Sun. - Sat.)
4	\$24	day of month	1 - 31
5	\$25	month	1 - 12
6	\$26	year	0 - 99
7	\$28	alarm seconds	0 - 59
8	\$29	alarm minutes	0 - 59
9	\$2A	alarm hours	0 - 23

Table 2 MC68HC68T1 Registers

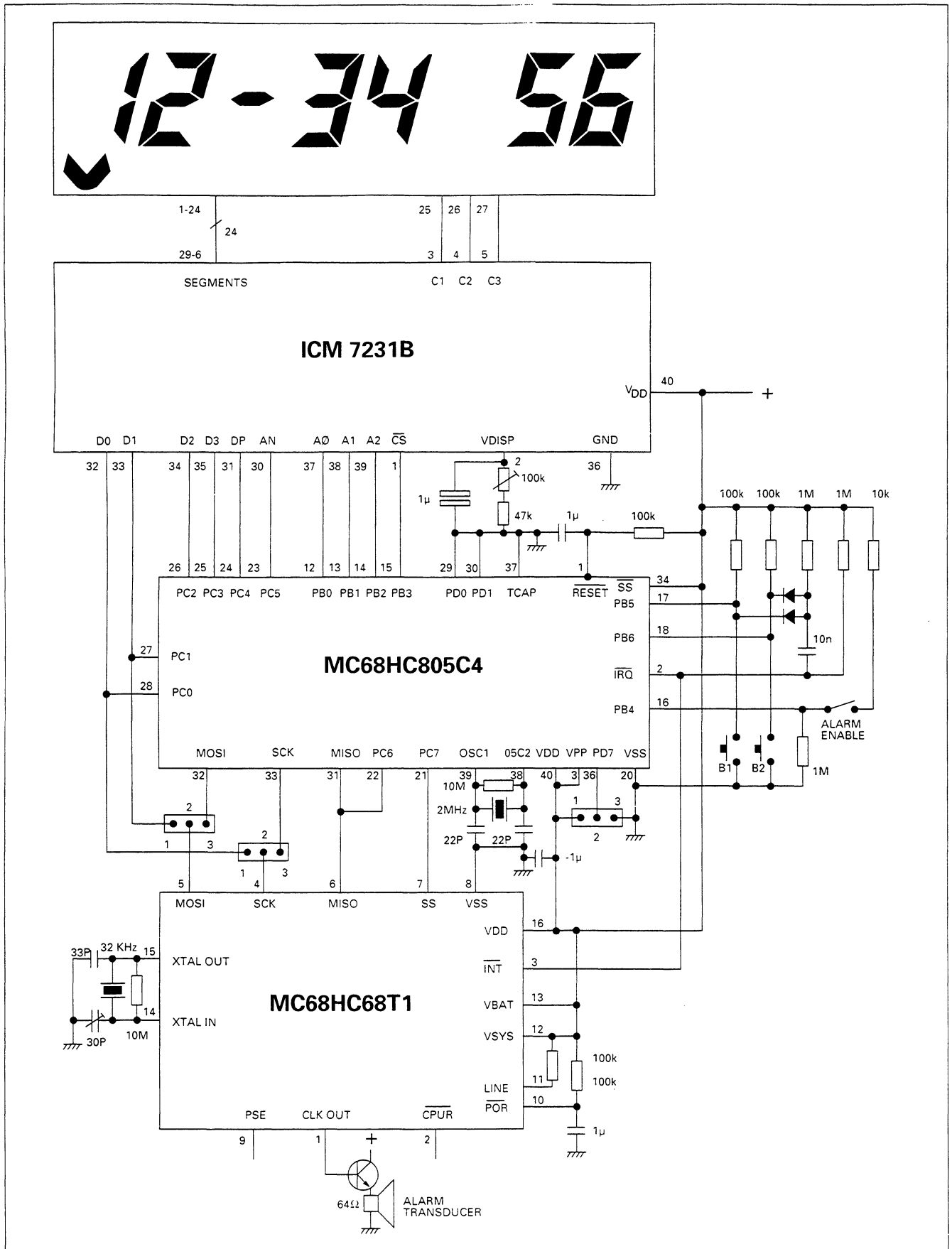


Figure 2.

```

1
2
3
4
5
6
7
8
9 00000001          PORTB  EQU    $0001          PORT B DATA
10 00000002         PORTC  EQU    $0002          PORT C DATA
11 00000003         PORTD  EQU    $0003          PORT D DATA
12 00000005         DDRB   EQU    $0005          PORT B DDR
13 00000006         DDRC   EQU    $0006          PORT C DDR
14
15 0000000a         SPCR   EQU    $000A          SPI CONTROL REGISTER
16 0000000b         SPSR   EQU    $000B          SPI STATUS REGISTER
17 0000000c         SPD    EQU    $000C          SPI DATA REGISTER
18
19 00000000         SCK    EQU    0          SERIAL CLOCK
20 00000001         MOSI   EQU    1          " OUT (MCU)
21 00000006         MISO   EQU    6          " IN (MCU)
22 00000007         SS     EQU    7          " ENABLE
23
24                 ORG     $0050
25
26
27 00000050         STAT   RMB    1          STATUS FLAGS
28 *
29 *
30 *
31
32 00000051         Q      RMB    8          DISPLAY REGISTER
33 00000059         DSPLY  RMB    1          DISPLAY MODE
34 0000005a         W1     RMB    1          TEMP. STORAGE
35 0000005b         W2     RMB    1          "
36 0000005c         W3     RMB    1          "
37 0000005d         W4     RMB    1          "
38 0000005e         W5     RMB    1          "
39 0000005f         W6     RMB    1          "
40 00000060         ADDR   RMB    1          SERIAL ADDRESS
41 00000061         DPNT  RMB    1          RAM POINTER
42
43 00000062         SEC    RMB    1          68HC68T1 TIME SECONDS ($20)
44 00000063         MIN    RMB    1          " " MINUTES ($21)
45 00000064         HOUR   RMB    1          " " HOURS ($22)
46 00000065         DAY    RMB    1          " " DAY ($23)
47 00000066         DATE   RMB    1          " " DATE ($24)
48 00000067         MONTH  RMB    1          " " MONTH ($25)
49 00000068         YEAR   RMB    1          " " YEAR ($26)
50 00000069         ASEC   RMB    1          " ALARM SECONDS ($28)
51 0000006a         AMIN   RMB    1          " " MINUTES ($29)
52 0000006b         AHOURL RMB    1          " " HOURS ($2A)
53 0000006c         STREG  RMB    1          STATUS REGISTER ($30)
54 0000006d         CCREG  RMB    1          CLOCK CONTROL REGISTER ($31)
55 0000006e         ICREG  RMB    1          INT. CONTROL REGISTER ($32)
56
57                 ORG     $0100

```

```

59
60
61
62
63
64
65
66 00000100 0a0103      IRQ1   BRSET   5,PORTB,IRQ2
67 00000103 cc01ba      JMP     BUT1           1 PRESSED
68 00000106 0c0103      IRQ2   BRSET   6,PORTB,IRQ3
69 00000109 cc0212      JMP     BUT2           2 PRESSED
70
71 0000010c a601      IRQ3   LDA     #1           READ STATUS REGISTER
72 0000010e b75a      STA     W1           TO RESTORE IRQ TO HIGH
73 00000110 a630      LDA     #$30
74 00000112 b760      STA     ADDR
75 00000114 a66c      LDA     #STREG
76 00000116 b761      STA     DPNT
77 00000118 cd02f6      JSR     READ
78
79 0000011b 09010e      BRCLR  4,PORTB,NOTAL  ALARMS ARMED ?
80 0000011e 036c0b      BRCLR  1,STREG,NOTAL  YES, HAS ONE OCCURED ?
81 00000121 1e01      BSET   7,PORTB       YES, DO IT
82 00000123 a601      LDA     #1
83 00000125 b75a      STA     W1
84 00000127 a6bf      LDA     #$BF         64Hz CLOCK OUT
85 00000129 cd02d4      JSR     CCONLY
86
87 0000012c 3f5e      NOTAL  CLR     W5         READ ALL TIME REGISTERS
88 0000012e a607      LDA     #7           FROM 68HC68T1
89 00000130 cd028d      JSR     RWSU
90 00000133 cd02f6      JSR     READ
91 00000136 ae08      LDX    #8           PREPARE TO WRITE DISPLAY
92 00000138 045932      BRSET  2,DSPLY,DISP2  NOT TIME ?
93 0000013b a60a      DISP1  LDA     #$0A         TIME DISPLAY
94 0000013d b753      STA     Q+2
95 0000013f b662      LDA     SEC         SECONDS
96 00000141 01590a      BRCLR  0,DSPLY,SC
97 00000144 a401      AND    #$01         DISPLAY DASH ONLY
98 00000146 2704      BEQ    DT           ON EVEN SECONDS
99 00000148 a60f      LDA     #$0F         IF DATE DISPLAY
100 0000014a b753      STA     Q+2         IS SELECTED
101 0000014c b666      DT     LDA     DATE   DATE INSTEAD OF SECONDS
102 0000014e ad56      SC     BSR     SPLIT
103 00000150 b657      LDA     Q+6         LEADING DIGIT
104 00000152 2604      BNE    CONT        A ZERO ?
105 00000154 a60f      LDA     #$0F         YES, REPLACE WITH BLANK
106 00000156 b757      STA     Q+6
107 00000158 b663      CONT  LDA     MIN     MINUTES
108 0000015a ad4a      BSR     SPLIT
109 0000015c b664      LDA     HOUR        HOURS
110 0000015e cd02e0      JSR     FORMAT
111 00000161 ad43      BSR     SPLIT
112 00000163 b651      LDA     Q           ZERO ?
113 00000165 2637      BNE    OK           NO
114 00000167 a60f      LDA     #$0F         YES REPLACE
115 00000169 b751      STA     Q           WITH BLANK
116 0000016b 2031      BRA    OK

```

```

118
119
120
121
122
123
124 0000016d 00591e      DISP2  BRSET   0,DSPLY,DISP3  NOT DATE ?
125 00000170 b668      LDA     YEAR             DATE DISPLAY
126 00000172 ad32      BSR     SPLIT
127 00000174 b667      LDA     MONTH           MONTH
128 00000176 ad2e      BSR     SPLIT
129 00000178 b666      LDA     DATE            DATE
130 0000017a ad2a      BSR     SPLIT
131 0000017c a60a      LDA     #$0A           INSERT DASHES
132 0000017e b753      STA     Q+2
133 00000180 b756      STA     Q+5
134 00000182 be65      LDX     DAY             DAY OF WEEK
135 00000184 271c      BEQ     DISPJ          ZERO ?
136 00000186 e651      LDA     Q,X            NO, GET DIGIT
137 00000188 aa20      ORA     #$20           SET BIT FOR ANNUNCIATOR
138 0000018a e751      STA     Q,X            PUT BACK
139 0000018c 2014      BRA     DISPJ
140
141 0000018e b669      DISP3  LDA     ASEC         ALARM SECONDS
142 00000190 ad14      BSR     SPLIT
143 00000192 b66a      LDA     AMIN           ALARM MINUTES
144 00000194 ad10      BSR     SPLIT
145 00000196 b66b      LDA     AHOOR          ALARM HOURS
146 00000198 ad0c      BSR     SPLIT
147 0000019a a60a      LDA     #$0A
148 0000019c b753      STA     Q+2
149 0000019e a60f      OK     LDA     #$0F     BLANK SIXTH DIGIT
150 000001a0 b756      STA     Q+5
151 000001a2 cd02a1  DISPJ  JSR     DISP
152 000001a5 80      RTI
153
154 000001a6 b75b      SPLIT  STA     W2             EXTRACT 2 PACKED BCD
155 000001a8 a40f      AND     #$0F           DIGITS AND PUT INTO Q
156 000001aa 5a      DECX
157 000001ab e751      STA     Q,X           LS BYTE
158 000001ad b65b      LDA     W2
159 000001af 44      LSRA
160 000001b0 44      LSRA
161 000001b1 44      LSRA
162 000001b2 44      LSRA
163 000001b3 a40f      AND     #$0F           MS BYTE
164 000001b5 5a      DECX
165 000001b6 e751      STA     Q,X
166 000001b8 5a      DECX
167 000001b9 81      RTS

```

```

169
170
171
172
173
174
175 000001ba cd0285
176 000001bd 0b01fd
177 000001c0 015003
178 000001c3 cc01e9
179 000001c6 0f010c
180 000001c9 1f01
181 000001cb a601
182 000001cd b75a
183 000001cf a6bc
184 000001d1 cd02d4
185 000001d4 80
186
187 000001d5 3c59
188 000001d7 b659
189 000001d9 a106
190 000001db 2602
191 000001dd 3f59
192 000001df cc010c
193
194
195
196
197
198
199
200 000001e2 59592307311299
201
202 000001e9 1250
203 000001eb 3c5c
204 000001ed be5e
205
206 000001ef d601e2
207 000001f2 055003
208 000001f5 d601db
209
210 000001f8 b15c
211 000001fa 2402
212 000001fc 3f5c
213
214 000001fe b65c
215 00000200 a40f
216 00000202 a10a
217 00000204 2508
218 00000206 a610
219 00000208 bb5c
220 0000020a a4f0
221 0000020c b75c
222 0000020e b65c
223 00000210 204b

*****
*
*      Button 1 routines.
*
*****

BUT1   JSR      DELAY      DEBOUNCE
        BRCLR   5,PORTB,*   STAY UNTIL KEY RELEASED
        BRCLR   0,STAT,SK2  SETUP MODE
        JMP     SET        YES
SK2    BRCLR   7,PORTB,NOA  ALARM ?
        BCLR   7,PORTB     YES, CANCEL IT
        LDA    #1
        STA    W1
        LDA    #$BC
        JSR    CCONLY      CLOCK OUT LOW
        RTI

NOA    INC     DSPLY      NO, MOVE TO NEXT
        LDA    DSPLY      DISPLAY OPTION
        CMP    #$06
        BNE   NOT6      TOO FAR ?
        CLR   DSPLY      YES, GO BACK TO ZERO
NOT6   JMP     IRQ3

*****
*
*      Button 1 setup function.
*
*****

TABL1  FCB     $59,$59,$23,$07,$31,$12,$99
SET    BSET   1,STAT      CHANGE MADE
        INC    W3         INCREMENT BYTE
        LDX   W5         POINTER
        LDA   TABL1,X     MAXIMUM FOR REGISTER ?
        BRCLR 2,STAT,NAL  ALARM ?
        LDA   TABL1-7,X   YES, OFFSET TABLE
NAL    CMP    W3         COMPARE MAX WITH ACTUAL
        BHS   NTB         MAX HIGHER OR SAME ?
        CLR   W3         NO, SO SET TO ZERO
NTB    LDA    W3         DECIMAL ADJUST
        AND   #$0F       LOOK AT LS NIBBLE
        CMP   #$0A       MORE THAN 9 ?
        BLO  NOADJ      NO, DO NOTHING
        LDA   #$10       YES, INC. MS NIBBLE
        ADD   W3
        AND   #$F0       CLEAR LS NIBBLE
        STA   W3
NOADJ  LDA    W3
        BRA   DW5         DISPLAY NEW CONTENTS

```



```

225
226
227
228
229
230
231 00000212 cd0285
232 00000215 0d01fd
233 00000218 00500b
234 0000021b cd02c6
235 0000021e 1050
236 00000220 3f5e
237 00000222 3a5e
238 00000224 1350
239 00000226 03500b
240
241 00000229 be5e
242 0000022b b65c
243 0000022d e762
244 0000022f ad5a
245 00000231 cd02f2
246
247 00000234 1350
248 00000236 3c5e
249 00000238 be5e
250 0000023a a30a
251 0000023c 2741
252
253 0000023e a307
254 00000240 2504
255 00000242 1450
256 00000244 2007
257
258 00000246 1550
259 00000248 ad41
260 0000024a cd02f6
261
262 0000024d be5e
263 0000024f e662
264 00000251 b75c
265 00000253 055007
266 00000256 ad33
267 00000258 cd02f2
268 0000025b b65c

*****
*
*      Button 2 routine (set-up).
*
*****

BUT2   JSR      DELAY          DEBOUNCE
        BRCLR   6,PORTB,*      STAY UNTIL KEY RELEASED
        BRSET   0,STAT,REG     SETUP MODE ?
        JSR     INTOFF         NO, STOP INTERRUPTS
        BSET    0,STAT         AND GO TO SET-UP MODE
        CLR     W5             INITIALISE POINTER
        DEC     W5             FIRST INC GIVES ZERO
        BCLR   1,STAT         CHANGE FLAG RESET
REG     BRCLR   1,STAT,NOSTOR  STORING

        LDX     W5             YES
        LDA     W3
        STA     SEC,X         STORE NEW VALUE IN RAM
        BSR     RWSU1
        JSR     WRITE        AND ALSO IN HC68T1

NOSTOR BCLR   1,STAT         CHANGE FLAG RESET
        INC     W5             NEXT REGISTER
        LDX     W5
        CPX     #$0A         TOO FAR ?
        BEQ     EXSET        YES, EXIT FROM SETUP

        CPX     #$07
        BLO     U7           TIME REGISTER ?
        BSET   2,STAT         NO, ALARM REGISTER
        BRA     NORD        WRITE ONLY

U7      BCLR   2,STAT         YES, TIME
        BSR     RWSU1        SO READ BYTE FROM RTC
        JSR     READ

NORD    LDX     W5
        LDA     SEC,X         FETCH RAM DATA
        STA     W3
        BRCLR  2,STAT,DW5    IF ALARM THEN
        BSR     RWSU1        WRITE RAM DATA
        JSR     WRITE        INTO 68HC68T1
        LDA     W3

```

```

270
271
272
273
274
275
276 0000025d b75b
277 0000025f 44
278 00000260 44
279 00000261 44
280 00000262 44
281 00000263 b757
282 00000265 b65b
283 00000267 a40f
284 00000269 b758
285 0000026b a60f
286 0000026d b751
287 0000026f b752
288 00000271 b753
289 00000273 b754
290 00000275 b756
291 00000277 be5e
292 00000279 bf55
293 0000027b cd02a1
294 0000027e 80
295
296 0000027f 1150
297 00000281 cd02ca
298 00000284 80
299
300 00000285 ae5f
301 00000287 5a
302 00000288 26fd
303 0000028a 81
304
305
306
307
308
309
310
311
312 0000028b a601
313 0000028d b75a
314 0000028f a620
315 00000291 bb5e
316 00000293 a126
317 00000295 2301
318 00000297 4c
319 00000298 b760
320 0000029a a662
321 0000029c bb5e
322 0000029e b761
323 000002a0 81

*****
*
*          Set-up mode display.
*
*****

DW5      STA      W2
          LSRA
          LSRA
          LSRA
          LSRA
          STA      Q+6          MS DIGIT
          LDA      W2
          AND      #$0F
          STA      Q+7          LS DIGIT
          LDA      #$0F
          STA      Q
          STA      Q+1         CLEAR
          STA      Q+2         REST
          STA      Q+3         OF
          STA      Q+5         DISPLAY
          LDX      W5          REGISTER
          STX      Q+4         ADDRESS
          JSR      DISP
          RTI

EXSET    BCLR    0,STAT      RETURN
          JSR    INTON      TO NORMAL MODE

ALEN     RTI

DELAY    LDX     #$FF        DEBOUNCE DELAY
AG       DECC
          BNE     AG
          RTS

*****
*
*          Read/write a byte to/from RAM at DPNT
*          from/to serial address ADDR.
*
*****

RWSU1   LDA     #1
RWSU     STA     W1
          LDA     #32
          ADD     W5
          CMP     #$26        IF GREATER THAN $26
          BLS     NO6         INCREMENT TO SKIP
          INCA                    UN-USED BYTE AT $27
NO6      STA     ADDR
          LDA     #SEC
          ADD     W5
          STA     DPNT
          RTS

```

```

325
326
327
328
329
330
331 000002a1 b601
332 000002a3 a4f0
333 000002a5 b701
334 000002a7 ae08
335 000002a9 090102
336 000002ac 1a51
337 000002ae e650
338 000002b0 b702
339 000002b2 1701
340 000002b4 1601
341 000002b6 5a
342 000002b7 2704
343 000002b9 3c01
344 000002bb 20f1
345 000002bd b601
346 000002bf a480
347 000002c1 b701
348 000002c3 3f02
349 000002c5 81
350
351
352
353
354
355
356
357 000002c6 a600
358 000002c8 2002
359 000002ca a61c
360 000002cc b76e
361 000002ce a602
362 000002d0 b75a
363 000002d2 a6bc
364 000002d4 b76d
365 000002d6 a631
366 000002d8 b760
367 000002da ae6d
368 000002dc bf61
369 000002de 2012
370
371 000002e0 03590e
372 000002e3 2602
373 000002e5 a612
374 000002e7 a112
375 000002e9 2306
376 000002eb abee
377 000002ed 2902
378 000002ef a006
379 000002f1 81

```

```

*****
*
*           Display contents of Q.
*
*****
DISP  LDA      PORTB      CLEAR
      AND      #$F0      LS BYTE OF PORTB
      STA      PORTB     IE DIGIT ADDRESS = 0
      LDX      #8
      BRCLR   4,PORTB,AGAIN  ALARM ?
      BSET    5,Q
AGAIN  LDA      Q-1,X
      STA      PORTC
      BCLR   3,PORTB      LATCH
      BSET    3,PORTB     DIGIT
      DECX
      BEQ     OUT         DONE ?
      INC    PORTB        NO, GOTO NEXT DIGIT
      BRA    AGAIN
OUT    LDA      PORTB
      AND      #$80      DON'T CANCEL ALARM
      STA      PORTB
      CLR     PORTC
      RTS
*****
*
*           Control reg. init. & 12hr. formatting.
*
*****
INTOFF LDA      #%00000000
      BRA
INTON  LDA      #%00011100      SET-UP CLOCK CONTROL
DOIT   STA      ICREG          AND INTERRUPT CONTROL
      LDA      #2
      STA      W1
      LDA      #%10111100      ENABLE COUNTERS
CCONLY STA      CCREG          SELECT 32kHz CRYSTAL
      LDA      #$31          CLOCK OUT DISABLED
      STA      ADDR
      LDX      #CCREG
      STX     DPNT
      BRA    WRITE
FORMAT BRCLR   1,DSPLY,EXIT    12 HOUR DISPLAY MOD.
      BNE    NOTMID          MIDNIGHT ?
      LDA    #$12          YES, MAKE IT 12
NOTMID CMP     #$12          NO, PM ?
      BLS   EXIT            NO, DO NOTHING
      ADD   #SEE           YES, SUBTRACT $12
      BHCS EXIT            USING HALF CARRY FOR
      SUB   #$06          DEC. ADJUST OF 8 & 9PM
EXIT   RTS

```

```

381
382
383
384
385
386
387 000002f2 1e60
388 000002f4 2002
389
390 000002f6 1f60
391
392 000002f8 1e02
393
394 000002fa 0f0347
395
396 000002fd b660
397 000002ff ad2c
398 00000301 0e601b
399
400 00000304 be61
401 00000306 a608
402 00000308 b75d
403 0000030a 1002
404 0000030c 0c0200
405 0000030f 1102
406 00000311 79
407 00000312 3a5d
408 00000314 26f4
409 00000316 3c61
410 00000318 3a5a
411 0000031a 26e8
412 0000031c 1f02
413 0000031e 81
414
415 0000031f be61
416 00000321 f6
417 00000322 ad09
418 00000324 3c61
419 00000326 3a5a
420 00000328 26f5
421 0000032a 1f02
422 0000032c 81
423
424 0000032d ae08
425 0000032f 49
426 00000330 2404
427 00000332 1202
428 00000334 2004
429 00000336 1302
430 00000338 2000
431 0000033a 1002
432 0000033c 1102
433 0000033e 1302
434 00000340 5a
435 00000341 26ec
436 00000343 81

```

```

*****
*
*      68HC68T1 serial routine (I/O PINS).
*
*****
WRITE  BSET    7,ADDR      BIT 7 HIGH
      BRA     SERT
READ   BCLR    7,ADDR      BIT 7 LOW
SERT   BSET    SS,PORTC    ENABLE HIGH (CPOL=0)
      BRCLR   7,PORTD,SPI  SELECT SERIAL LINK
PPSER  LDA     ADDR        SEND CHIP ADDRESS
      BSR     SHIFT       CHIP ADDRESS OUT
      BRSET   7,ADDR,WRBUS READ OR WRITE ?
RDBUS  LDX     DPNT
      LDA     #8
      STA     W4
LOOP2  BSET    SCK,PORTC   CLOCK HIGH
      BRSET   MISO,PORTC,*+3 DATA LINE (RESULT IN CARRY)
      BCLR    SCK,PORTC   CLOCK LOW
      ROL     0,X
      DEC     W4
      BNE     LOOP2
      INC     DPNT
      DEC     W1
      BNE     RDBUS
      BCLR    SS,PORTC    ENABLE LOW
      RTS
WRBUS  LDX     DPNT        DATA BUFFER POINTER
      LDA     0,X         DATA
      BSR     SHIFT
      INC     DPNT
      DEC     W1         No. BYTES
      BNE     WRBUS
      BCLR    SS,PORTC    ENABLE LOW
      RTS
SHIFT  LDX     #8
SHIFT1 ROLA
      BCC     SHIFT2     ZERO ?
      BSET    MOSI,PORTC NO, DATA = 1
      BRA     SHIFT3
SHIFT2 BCLR    MOSI,PORTC DATA = 0
      BRA     SHIFT3     DELAY
SHIFT3 BSET    SCK,PORTC  CLOCK HIGH
      BCLR    SCK,PORTC  CLOCK LOW
      BCLR    MOSI,PORTC DATA LOW
      DECX
      BNE     SHIFT1
      RTS

```

```

438
439
440
441
442
443
444 00000344 b660
445 00000346 b70c
446 00000348 0f0bfd
447 0000034b be61
448 0000034d f6
449 0000034e b70c
450 00000350 0f0bfd
451 00000353 0e6003
452 00000356 b60c
453 00000358 f7
454 00000359 5c
455 0000035a 3a5a
456 0000035c 26ef
457 0000035e 1f02
458 00000360 81
459
460
461
462
463
464
465
466 00000361 3f01
467 00000363 3f02
468 00000365 a68f
469 00000367 b705
470 00000369 a6bf
471 0000036b b706
472 0000036d a655
473 0000036f b70a
474 00000371 cd02ca
475 00000374 3f50
476 00000376 3f59
477 00000378 8e
478 00000379 20fd
479
480
481
482
483
484
485
486
487 00001ff4 0361
488 00001ff6 0361
489 00001ff8 0361
490 00001ffa 0100
491 00001ffc 0361
492 00001ffe 0361
493
494

*****
*
*           SPI serial routine.
*
*****
SPI      LDA      ADDR      GET ADDRESS
          STA      SPD       SEND IT
          BRCLR   7,SPSR,*   WAIT TILL FINISHED
          LDX      DPNT      LOCATION OF 1st BYTE
SPIILP   LDA      0,X       GET IT
          STA      SPD       SEND IT
          BRCLR   7,SPSR,*   WAIT TILL FINISHED
          BRSET   7,ADDR,SKIP ARE WE WRITING ?
          LDA      SPD       NO, SO READ BYTE
          STA      0,X       AND SAVE IT
SKIP     INCX
          DEC      W1        DECREMENT COUNT
          BNE     SPILP      FINISHED ?
SPIIP    BCLR    SS,PORTC   YES, ENABLE LOW
          RTS

*****
*
*           Reset routine.
*
*****
RESET    CLR      PORTB
          CLR      PORTC
          LDA      #$8F      PORTB
          STA      DDRB      4, 5, & 6 INPUTS
          LDA      #$BF      PORTC
          STA      DDRC      ALL OUTPUTS EXCEPT 6
          LDA      #%01010101 ENABLE SPI AS MASTER AT
          STA      SPCR      500kHz, NO INTERRUPTS
          JSR     INTON      INIT TO 1Hz INTERRUPTS
          CLR     STAT
          CLR     DSPLY
STOPP    STOP
          BRA     STOP

*****
*
*           MC68HC05C4 Vectors.
*
*****
          ORG     $1FF4
          FDB     RESET     SPI STATUS
          FDB     RESET     SCI STATUS
          FDB     RESET     TIMER
          FDB     IRQ1      EXTERNAL INTERRUPT
          FDB     RESET     SWI
          FDB     RESET     RESET

          END

```


This page intentionally left blank.



This page intentionally left blank.



All products are sold on Motorola's Terms & Conditions of Supply. In ordering a product covered by this document the Customer agrees to be bound by those Terms & Conditions and nothing contained in this document constitutes or forms part of a contract (with the exception of the contents of this Notice). A copy of Motorola's Terms & Conditions of Supply is available on request.

Motorola reserves the right to make changes without further notice to any products herein. Motorola makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Motorola assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters can and do vary in different applications. All operating parameters, including "Typicals" must be validated for each customer application by customer's technical experts. Motorola does not convey any license under its patent rights nor the rights of others. Motorola products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Motorola product could create a situation where personal injury or death may occur. Should Buyer purchase or use Motorola products for any such unintended or unauthorized application, Buyer shall indemnify and hold Motorola and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Motorola was negligent regarding the design or manufacture of the part. Motorola and  are registered trademarks of Motorola, Inc. Motorola, Inc. is an Equal Opportunity/Affirmative Action Employer.

The Customer should ensure that it has the most up to date version of the document by contacting its local Motorola office. This document supersedes any earlier documentation relating to the products referred to herein. The information contained in this document is current at the date of publication. It may subsequently be updated, revised or withdrawn.

Literature Distribution Centers:

USA: Motorola Literature Distribution; P.O. Box 20912; Phoenix, Arizona 85036.

EUROPE: Motorola Ltd.; European Literature Centre; 88 Tanners Drive, Blakelands, Milton Keynes, MK14 5BP, England.

JAPAN: Nippon Motorola Ltd.; 4-32-1, Nishi-Gotanda, Shinagawa-ku, Tokyo 141, Japan.

ASIA PACIFIC: Motorola Semiconductors H.K. Ltd.; Silicon Harbour Center, No. 2 Dai King Street, Tai Po Industrial Estate, Tai Po, N.T., Hong Kong.



MOTOROLA

JIT PRINTED IN THE USA 1993 MPS

ANE425/D

