

AN460

An RDS Decoder using the MC68HC05E0

Peter Topping,
MCU Applications Group,
Motorola Ltd., East Kilbride

INTRODUCTION

The Radio Data System (RDS) adds a digital data capability to the FM VHF transmissions on band II (87.5 to 108 MHz). This capability is in use in the UK and several other European countries, the intention being that most of western Europe will eventually adopt it. The specification is defined in EBU Technical Document 3244 (see reference 1).

To transmit the data, a sub carrier is added at 57 KHz. This sub carrier is amplitude modulated with a shaped bi-phase coded signal. The sub carrier itself is suppressed to avoid data modulated cross-talk in phase-locked loop stereo decoders and to maintain compatibility with the German ARI system which uses the same sub carrier frequency. Information is sent in groups of four 26-bit blocks. Each group of 104 bits is one of several types containing different information. It is up to the broadcaster which features are transmitted. The only constraints are that the specified format must be adhered to and that PI, PTY and TP should always be included. Each group contains a different sub-set of the RDS features; table1 lists all currently defined RDS features.

Table 1. RDS features

Feature	Information
PI	Program identification
PTY	Program type
PS	Program service name
RT	Radiotext
CT	Clock time and date
AF	Alternative frequencies
TA	Traffic announcement
TP	Traffic program
MS	Music/speech switch
DI	Decoder identification
PIN	Programme item number
EON	Enhanced other networks
TDC	Transparent data channel
INH	In-house data

The retrieval of data is carried out by a demodulator circuit which generates clock and data signals that can be used by a microprocessor. Suitable demodulators which can perform this function include SAA7579T, TDA7330, LA2231 and RDS hybrids. The block diagram of a typical application is shown in [Figure 1](#). The microprocessor, in this case an MC68HC05E0, decodes the RDS data using the clock and data signals from one of these demodulators and sends selected data to dot-matrix display modules.



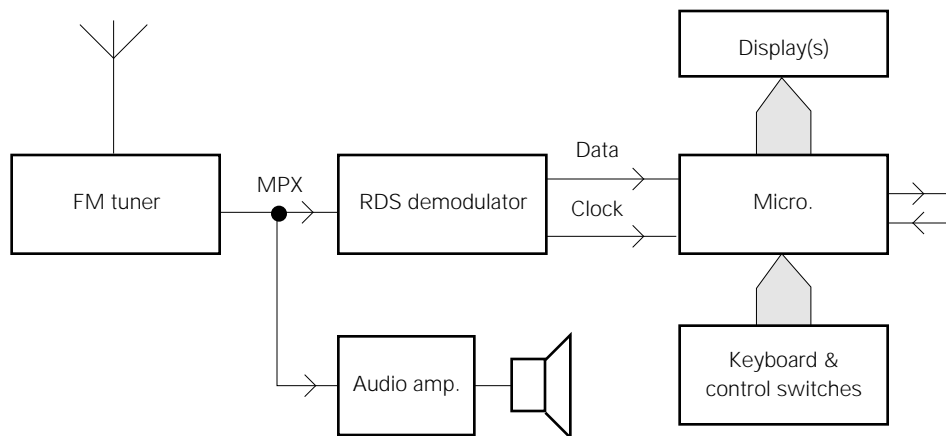


Figure 1. Typical application

This application incorporates an alarm clock which, if permanently powered, can be used to switch on the radio supplying the RDS data, at the required alarm time. There is a second alarm output intended to sound an alarm. This output is cancelled when any key is pressed, leaving the control output active. The control output could be used to switch the power supply of the radio or the audio stage. If an audio mute is used, RDS information can be updated even when the radio is "off". Alternatively the decoder can be used simply to display RDS data with its power being supplied from the radio and manually switched on and off.

RDS FEATURES

This application supports PI, PTY, PS, RT, CT, TP, TA, MS, DI, PIN and EON (see [Table 1](#)). These features facilitate permanent display of the 8-digit station name (PS) and time (CT) and, on request, can display program type (PTY), radiotext data (RT) and the status of the other RDS features. EON data can be displayed, but the retuning features associated with AF and EON are not supported as there is no capability to control the tuned frequency. In a car radio EON data would be used to switch the radio to a station which is broadcasting local traffic information and AF data to tune the radio to the strongest signal carrying the selected service.

PI is a two byte number which identifies the country, coverage area and service. It can be used by the control microprocessor but is not normally intended for display. A change in PI code causes the initialisation of all RDS data as it indicates that the radio has been retuned. This application also facilitates the display of the current PI code.

PTY is a 5-bit number which indicates the current program type. At present 16 of these types are defined. Examples include "no programme type", "Current affairs" and "Pop music", although the actual syntax which is displayed is determined by the software of the controlling microprocessor. In this example PTY can be displayed on request. [Table 2](#) shows the display used for each PTY code.

PS is the eight character name of the station and is permanently displayed (except in the standby mode).

RT is radiotext and constitutes a string of up to 64 characters which give additional information regarding the service or programme currently being transmitted. In this application, RT is displayed on request on the 16-digit dot-matrix displays using scrolling.

Table 2. PTY Types

PTY	Display
0	No program type
1	News
2	Current affairs
3	Information
4	Sport
5	Education
6	Drama
7	Culture
8	Science
9	Varied
10	Pop music
11	Rock music
12	Easy listening
13	Light classics
14	Serious classics
15	Other music
16-31	No program type

The data often contains extra spaces to centre the text on a 2x32 character display. As this is not suitable for a 16-character scrolling display the software reduces all sequences of two or more spaces to a single space.

CT data is transmitted every minute on the minute and provides a very accurate clock, traceable to national standards. The (Modified Julian) date and local time variation are also transmitted. Time is permanently displayed. In standby mode (see below) the date is displayed instead of the PS name. The MJD number, which is the form in which the date is received, can also be displayed. The microprocessor converts this number into day-of-week, day-of-month, month and year.

AF would be used by a car radio to retune to the strongest signal carrying the selected service. AF data, along with TDC and INH, is not used in this application.

TA and TP are flags. TP is set if the transmitter normally carries traffic information and TA is set if a traffic announcement is in progress. The combination, TA=1 and TP=0, is used to indicate that EON data is being used to supply information on other networks including traffic announcements. The status of these flags can be displayed and the combination, TA=TP=1, is brought out to a pin and can be used to control a LED or external hardware. An example of this could be to demute the radio or switch from cassette when a traffic announcement is taking place.

MS is a single bit indicating either music or speech and is intended to be used to make a tone or volume adjustment to a radio's audio stage. The MS bit is displayed on request.

Decoder information (DI) constitutes four bits indicating the type of transmission (mono, stereo, binaural, etc.). It is not currently in use in the UK but can be displayed as a number between 0 and 15.

Programme item number or PIN is used to identify the programme currently being broadcast. The format is a 2-byte number which includes the scheduled time and date (day-of-month) of the start of the programme. It can be displayed as four hexadecimal digits or fully decoded to day-of-month and time.

EON (Enhanced Other Networks) replaces the older ON format. If type 14 groups are used to provide EON data then type 3 groups (ON) will not be used (Table 6 shows the currently defined group types). Type 14A groups are used to send information about other networks. The PS name and principal frequency of up to 11 other networks can be displayed. Type 14B groups are intended to be used to switch to traffic announcements in a radio in which the microprocessor can control the tuned frequency.

DECODING

Each 26-bit block contains 16 bits of data and 10 extra bits which are used for synchronisation and error detection. There are no gaps between blocks or groups, the synchronisation being done by looking for specific checkwords in the incoming data. In order to look for a checkword a stream of 26 consecutive data bits has to be multiplied by the fixed 10x26 matrix shown in Figure 2.

The result of this multiplication is a 10-bit word which is compared with allowed values. There are 5 of these 10-bit "syndromes", one for each of the blocks 1, 2 and 4 and two for block 3 (see Table 3). The alternative syndrome for block 3 is used in the B version of a group. In this version the PI code is sent in block 3, replacing what would be sent in the A version of the same group type. This is done to increase the frequency of sending the PI code so that it can be acquired more quickly.

10 0000 0000	(\$02,\$00)
01 0000 0000	(\$01,\$00)
00 1000 0000	(\$00,\$80)
00 0100 0000	(\$00,\$40)
00 0010 0000	(\$00,\$20)
00 0001 0000	(\$00,\$10)
00 0000 1000	(\$00,\$08)
00 0000 0100	(\$00,\$04)
00 0000 0010	(\$00,\$02)
00 0000 0001	(\$00,\$01)
10 1101 1100	(\$02,\$DC)
01 0110 1110	(\$01,\$6E)
00 1011 0111	(\$00,\$B7)
10 1000 0111	(\$02,\$87)
11 1001 1111	(\$03,\$9F)
11 0001 0011	(\$03,\$13)
11 0101 0101	(\$03,\$55)
11 0111 0110	(\$03,\$76)
01 1011 1011	(\$01,\$BB)
10 0000 0001	(\$02,\$01)
11 1101 1100	(\$03,\$DC)
01 1110 1110	(\$01,\$EE)
00 1111 0111	(\$00,\$F7)
10 1010 0111	(\$02,\$A7)
11 1000 1111	(\$03,\$8F)
11 0001 1011	(\$03,\$1B)

Figure 2. 10x26 decoding matrix

Table 3. Syndromes

Block	Syndrome	Binary	Hex
1	A	11 1101 1000	\$03, \$D8
2	B	11 1101 0100	\$03, \$D4
3	C	10 0101 1100	\$02, \$5C
	C'	11 1100 1100	\$03, \$CC
4	D	01 0101 1000	\$01, \$58

This syndrome test has to take place after each bit is received. The test inspects the last 26 bits received, until a valid syndrome is found. In this application, only syndrome A is accepted during the bit-by-bit syndrome check and the data is used only after four valid syndromes have been acquired. A more complex algorithm could allow all syndromes to be accepted during initial synchronisation and require less than four valid syndromes before the data is used. This can reduce the time taken to acquire the PI code, which is also included in block 3 of type B groups, but increases the likelihood that random data, giving a valid syndrome, will be used in error. The bit rate is 1187.5 Hz so the control microprocessor has a lot to do during this initial synchronisation. Once the first valid syndrome has been found, subsequent syndrome checks need be done only after the next 26 bits have been received, as this is when the next valid syndrome would be expected. If it is not found, then the bit-by-bit synchronisation check is re-started. Once consecutive A, B, C (or C') and D syndromes have been detected, a complete group has been acquired and the data can be used.

Four bits in block 2 determine the group type. Block 2 also contains TP and PTY data. The use of the other bits in blocks 2, 3 and 4 depends upon the group type while block 1 always contains the PI code. [Table 7](#) shows the structures of the group types which are handled in this application.

CIRCUIT

[Figure 3.](#) shows the circuit diagram. As different demodulator devices can be used, the circuitry for the demodulator is not shown. The clock from the demodulator interrupts the microprocessor on each positive edge. At this time a data bit is available and is read on bit 2 of port B. Both an LCD and a VFD module are shown but normally only one will be used. If the LCD module is not connected, a pull-down resistor should be connected to bit 7 of port C, as the microprocessor uses this bit to check that the controller in the module is ready to receive a command. If this bit is left open circuit, it may cause the software to hang up. Alternatively the LCD drive software could be removed, allowing the use of port C for other purposes.

With more I/O available, additions to the software would allow access to the other control bits intended for controlling external hardware. These include the MS bit, DI data (4 bits) and PIN (match with current time and date). They could be brought out to port pins in a manner similar to that used for the TA=TP=1 signal. The unused port A and D pins could also be used for this purpose but in this application they were used during debug by the E0BUG monitor (reference 2). The application could make use of the port A and D pins, if debugging was done on a development system which did not have this limitation.

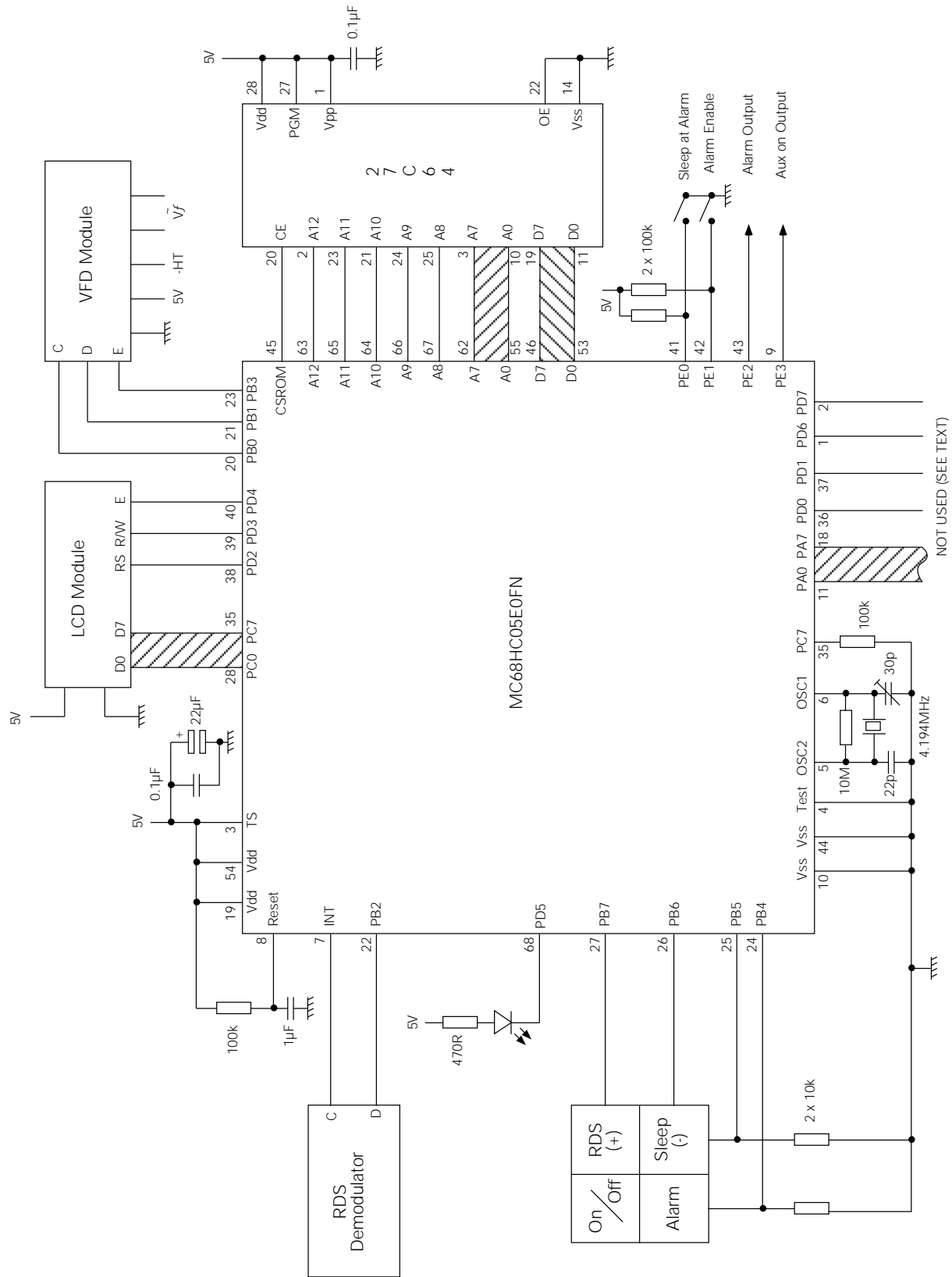


Figure 3. Circuit diagram

SOFTWARE

The complete software is listed. The reset routine (START) sets up the I/O ports including the enabling of some of the special functions available on port D. These signals (A15, A14, R/W and the P02 clock) were used during debug. The pins are not used in the final application. This also applies to all the port A pins which are configured as outputs. External interrupts are enabled on positive edges so that the RDS clock can interrupt the microprocessor when each data bit is available. Timer B runs as a real-time clock with interrupts every 125 ms. Correct operation of this clock in the absence of an RDS signal requires that a 4.194 MHz crystal be used (the trimmer on pin 6 should be adjusted for accurate timekeeping). Timer A's pre-scaler is set up to divide by 64; this causes the idle loop to cycle at 64Hz. The reset routine also initialises the LCD module (the display shows Mon 0 inv 00:00 until a valid group 4A is received), clears the RAM and calls a subroutine (INITD) to initialise the RAM locations used for displaying data.

Lines 114-118 and 193-208 are commented out as they are only relevant when de-bugging using the E0BUG monitor (reference 2).

The idle loop (IDLE) regularly checks the local keyboard for a keypress, compares the current time with the alarm time and performs other time-dependent functions related to the display modules and the sleep timer.

The keyboard software (KBD) scans the 4-key matrix for a keypress every 16ms. If the same key is held pressed for 3 successive scans, it acts on this key function by calling the relevant subroutine (ALARM, ONOFF, SLEEP or RDS). This software also controls the repeat rate of the SLEEP and RDS keys. This rate is set at 6Hz (after an initial 750ms delay) when the keys are used to change the alarm time and 1Hz for their normal function. The other keys do not repeat if held down. [Table 4](#) shows the functions available in each mode.

Table 4. Key functions

MODE	KEY					
	On/Off	Sleep	Alarm	RDS		
Standby (Off)	mode normal (On)	mode sleep (On)	mode alarm	—		
Normal (On)	mode stndby (Off)			mode alarm ON	RT PTY PI TA/TP PIN(h) PIN(d) MJD MS/DI EON 1 : : EON 11	
Alarm OFF			mode alarm set-up		mode alarm OFF	
Alarm ON						
Alarm SET UP	toggle hr/min		dec. hr/min	inc. hr/min		

The On/Off key uses the subroutine ONOFF to toggle between ON and standby. A port pin (3,PORTE) can be used to control the power to the VHF radio and/or other external hardware. In standby mode, with the alarm disabled, the time and date are displayed. If the alarm is enabled, the alarm time is displayed. In the ON mode the time is displayed along with the current RDS PS-name. Table 5 shows these display formats.

Table 5. Display formats

Display mode		Format
Standby (Off)	Alarm off	Thu 30 Apr 18:05
	Alarm off, no CT	Mon 0 inv 0:00
	Alarm on	0659 ALARM 18:05
Normal (On)	With RDS PS name	BBC R4 18:05
	Without RDS	----- 18:05
Alarm	Alarm off	Alarm - OFF
	Alarm on	Alarm - 6:59
Sleep		Sleep 60 min.
RDS	RT	BBC Radio 4
	PTY	News
	PI	PI code - C204
	TA & TP	TP - 0 TA - 1
	PIN(hex)	PIN no. - F480
	PIN(decod)	30th at 18:00
	MJD	MJ day - 48742
	MS & DI	M/S M DI 15
	EON 1	BBC R3 92.10
	2	BBC R.Sc 103.60
	3	BBC Nwcl 96.00
	4	BBC Scot 94.30
5	BBC Mtme 92.50	
6	BBC Twed 93.50	
7	BBC R5 909kHz	
8	BBC Eng. 100.00	
9	BBC R1 99.50	
10	BBC R2 89.90	
11	-----	

The Alarm key calls the subroutine ALARM which displays the current alarm status. A second press changes the alarm armed status. When the alarm is armed, the alarm time is displayed. In this mode the On/Off key can be used to select either hours or minutes (indicated by flashing) and the Sleep and RDS keys used to increment and decrement the settings. If the alarm has triggered then the first press of any key cancels it. The alarm display has one of the two alarm formats shown in Table 5 according to whether or not the alarm is armed. As all the keys have a special function in the alarm mode the only way to exit this mode is to wait for a timeout. If no keys are pressed for 5 seconds, the mode returns to normal.

The Sleep key controls the sleep timer. If the decoder is in the standby mode the first press of Sleep switches it on and initialises the sleep time to 60 minutes. When the sleep timer is running, this is indicated by a flashing decimal point in the right-most character of the display modules. Subsequent presses of the Sleep key decrement the time remaining by 5 minutes. When the sleep time has elapsed, the decoder returns to standby. In the alarm set-up mode this key decrements the alarm time.

The RDS key uses subroutine RDS to step through the various RDS data which can be displayed. Holding down this key steps through the displays at 1Hz. The displays are RT (scrolling), PTY, PI, TA/TP, PIN (hex), PIN (decoded), MJD, MS/DI and EON (11 networks) as shown in [Table 5](#). In the alarm set-up mode this key increments the alarm time.

The timer interrupt routine (TINTB) updates the RT scrolling pointers (DISP1 and DISP2). These pointers are incremented regularly whether or not an RT display is active. In this way, the software can be easily converted to using a 2-line LCD module in which the top line is the normal display of PS-name and time and the lower line a permanent display of scrolling RT. The timer interrupt also decrements the sleep timer and updates the RAM locations used to store hours, minutes, seconds and eighth-seconds. All RDS data (except date and time) is cleared by this routine if no valid RDS data is detected for a period of 10 seconds.

SYNDROME AND CONFIDENCE

Hardware interrupts are vectored to jump to SDATA where serial data is received from the RDS demodulator. The clock edge causes an interrupt and the first instruction reads the data into the carry bit of the condition code register. The bit is shifted into a 4-byte RAM register and the matrix multiplication performed. The state of flag 0,STAT2, determines if the multiplication is to take place after every bit or only after all 26 bits have arrived. The multiplication is performed using two EOR instructions for every bit (two are required as the 10-bit syndrome requires two bytes). As the top of the matrix (see [Figure 2](#).) is the unity matrix, the first 10 bits are transferred directly into the syndrome RAM locations (SYN). This, the omission of any EOR #00 instructions, the reordering of the bits and the use of the index register for temporary storage help to reduce the length of inline code in this routine. The routine could be shortened by using a loop but this would incur an unacceptable penalty in execution time. Microprocessors with two accumulators would find this task a lot simpler and quicker but an MC68HC05E0, at half its maximum speed, can easily perform the calculation in the required time.

After the multiplication has been performed the resultant 10-bit number is compared with the allowed syndromes (see [Table 3](#)). The variable LEV records the current block level. It is initially zero but incremented each time a valid syndrome is found. When it is zero only syndrome A is accepted, if this is found then syndrome B is expected 26 bits later so when LEV is one only syndrome B is accepted. If an invalid syndrome is found LEV is cleared, the syndrome confidence level CONF is decremented and the interrupt ended.

When a valid syndrome is found, CONF is increased by 4 and the 16 data bits saved in the relevant bytes of TMPGRP. If the valid syndrome is type D then a complete group has been received and all 8 bytes are transferred to the 8 RAM locations at GROUP. This double buffer means that the data in GROUP can be used while interrupts are overwriting TMPGRP with new data.

The confidence level CONF is used to decide what should be done if the data becomes unreliable due to a poor RF input to the receiver. When the first valid syndrome is found it is initialised to 42. Subsequent valid syndromes increment it by four and invalid ones decrement it by 1. If CONF falls below 41, then it is assumed that synchronisation has been lost and a bit-by-bit re-synchronisation is carried out. If it falls below 10, the signal is deemed unacceptable and the displays are re-initialised. The confidence level is not incremented by the detection of a valid syndrome if it is higher than 56.

GROUPS HANDLED

If a complete group has been received the data can be processed. The buffering used would allow this to be done outside the interrupt but in this case there is sufficient time to do it within the interrupt. The PI code is checked to see if it has changed. If it has changed the displays are initialised. In an application using the AF capability of RDS, more use would be made of the PI code.

Next PTY and TP are updated and the group type identified. Group types 0A, 0B, 1A, 1B, 2A, 4A, 14A and 15B are handled. [Table 6](#) shows the type of information contained in each group and [Table 7](#) shows the detailed structure of the groups actually used.

Table 6. RDS Groups

Group	Features
All	PI, PTY, TP
0	TA, DI, MS, PS, AF
1	PIN
2	RT
3	ON (replaced by EON)
4A	CT
5	TDC
6	INH
14	EON
15B	TA, DI, MS

Group 0 & 15B

As AF data is not handled, there is no difference in the treatment of groups 0A and 0B. PS data is extracted and placed in RAM according to the address bits in block 2 (see [Table 7](#)). TA, DI and MS data are then read, DI is sent a single bit at a time and uses the same address bits as the PS name to determine which of the four bits is being updated. Groups of type 15B also contains all this switching information. They are used to increase the repetition rate of this data but contain no PS or AF information.

Group 1

Group types 1A and 1B contain the same data except for the repetition of the PI code in type 1B. The PIN data is recovered and saved in RAM. This is intended for future use to control external hardware, for example a tape recorder. This would facilitate the unattended recording of a pre-selected program. At present this application simply allows the display of PIN data both in its raw hexadecimal form and fully decoded to day-of-month and time. Full use of PIN data would require continuously comparing the PIN day-of-month and time with the current day-of-month and time enabling an I/O pin to be switched when there is a match.

Group 2A

RT data from blocks 3 and 4 is written to RAM according to the address included in block 2. There are four address bits and four ASCII encoded bytes giving the possibility of 64 characters. If the Text A/B flag changes state, the RT area in RAM is cleared, indicating that the message has changed. Group 2B is not handled as it is rarely, if ever, used.

Group 4A

Two of the more complex tasks to be performed are required by the CT calculations for group 4A. These are for the local time difference and the conversion of the MJD number into a recognisable date.

The broadcast time is Universal Coordinated Time (UTC), effectively the same as GMT. Time differences from UTC, including summer (daylight saving) time, are sent as an offset of up to +/- 12 hours in half-hour increments.

The software includes 4-function, 9-digit integral BCD arithmetic which is used to decode the date from the MJD number using the formulae:

$$\begin{aligned} Y' &= \text{int}[(\text{MJD}-15078.2)/365.25] \\ M' &= \text{int}[(\text{MJD}-14956.1-\text{int}\{Y' \times 365.25\})/30.6001] \\ \text{Day} &= \text{MJD}-14956-\text{int}(Y' \times 365.25)-\text{int}(M' \times 30.6001) \\ \text{If } M' = 14 \text{ or } M' = 15, \\ &\quad \text{then } K = 1; \\ &\quad \text{else } K = 0 \\ \text{Year} &= Y' + K \\ \text{Month} &= M' - 1 - 12K \end{aligned}$$

Group 14A

This group contains EON data. A large amount of information can be sent using this group, and it can take up to two minutes for all the data to arrive after the radio has been retuned. This application saves the PI code, PS name and principal frequency of up to 11 networks although more networks, each with many frequencies, and other data (e.g. PTY(ON), PIN(ON), TA(ON) etc.) may be sent. [Table 5](#) shows the format of the EON display. All the information shown is real data from the Black Hill transmitter in central Scotland.

Displays

The software drives both a parallel LCD module (based on an HD44780 driver with or without an HD44100) and a serial VFD module (based on an MSC7128 driver) to give a choice of display types. The displays show the same data (within the limitations of their character ROMs).

The display routine (MOD) is executed in the idle loop if flag 3, STAT2 is set. It is set every 125ms by timer B interrupts. If flag 4, STAT2 is set, the display is initialised, indicating no valid RDS data. The LCD module is then updated with new data. Each time anything is written to the module, the subroutine WAIT is used before the write is executed; this checks that the controller in the module is not busy. This is indicated by a low on bit 7, so bit 7 on port C should have a pull-down resistor to satisfy this condition if an LCD module is not being used.

Table 7.

	Block 1		Block 2		Block 3		Block 4	
Group 0 and 15B	PI code	chck A	bit(s) use 15-12 : group no. 11 : group type 10 : TP flag 9-5 : PTY code 4 : TA flag 3 : M/S bit 2 : DI bit 1-0 : PS/DI address	chck B	AF (PI code in type 0B and 15B)	chck C or C'	PS name (as block 2 for 15B)	chck D
Group 1	PI code	chck A	15-12 : 0001 11 : group type 10 : TP flag 9-5 : PTY code 4-0 : not used	chck B	not used (PI code in type 1B)	chck C or C'	PIN data 15-11 : day-of-month 10-6 : hour 5-0 : minute	chck D
Group 2A	PI code	chck A	15-12 : 0010 11 : 0 10 : TP flag 9-5 : PTY code 4 : text A/B flag 3-0 : text address	chck B	RT 2 ASCII characters	chck C	RT 2 ASCII characters	chck D
Group 4A	PI code	chck A	15-12 : 0100 11 : 0 10 : TP flag 9-5 : PTY code 4-2 : not used 1-0 : MJD (16-15)	chck B	CT 15-1 : MJD (14-0) 0 : hour (4)	chck C	CT 15-12 : hour (3-0) 11-6 : minute (5-0) 5 : offset sense 4-0 : offset (4-0)	chck D
Group 14A	PI code	chck A	15-12 : 1110 11 : 0 10 : TP flag 9-5 : PTY code 4 : TP (On) flag 3-0 : usage code	chck B	EON information code: 0-3 : PS 4 : AF 5-9 : AF (map) 10-11 : not used 12-15 : not imp.	chck C	PI (On)	chck D

The listing is shown for use with a divide by 8 multiplexing LCD module. This module will normally contain an HD44780 and an HD44100.

If a divide by 16 module (HD44780 only) is to be used then line 1294 should be replaced by line 1293 and line 1371 commented out to include the execution of the code on lines 1379 to 1392.

The different display formats are selected by checking the various flags and the relevant routine executed. The normal display permanently shows PS name and time. As the locations in RAM used for hours and minutes contain binary numbers they are converted to BCD before being written to the relevant bytes in DISP. Once all 16 bytes in DISP have been loaded, a loop is used to send the data to the LCD module.

The VFD routine sends the same data as is shown on the LCD module to the serial VFD module. The display driver used has a different character set from the standard ASCII set used by the LCD module. The table VTAB is used to convert ASCII data into the required character in the VFD module. The small table INITF is used to send the required initialisation bytes to the VFD module. This module does not require a busy check but does require a delay between successive bytes. This is satisfied by the wait loop within the serial output loop VFDF.

Alarm functions

The alarm time can be entered as described above. If the alarm is enabled (alarm time displayed on first press of the ALARM key, and permanently displayed in standby mode) then, at the alarm time, the auxiliary control line will go high. This can be used to control external hardware, for example to switch on the VHF radio supplying the RDS data. If the auxiliary line is already high (decoder fully on or on via the sleep timer), then it simply stays high. The operation of the sleep timer is not affected if bit 0 of port E is high. If this I/O line is low at the alarm time, then the sleep timer is activated for an hour. This takes place whether the decoder was previously on, off, or running the sleep timer, and has the effect of switching the auxiliary line low an hour after the alarm time, regardless of its condition prior to the alarm.

At the alarm time the alarm output will also be activated (active low) as long as it is enabled by bit 1 of port E being held low. This is intended to drive an alarm sounder. When this output is active, a press of any key cancels it until the next alarm. This cancellation does not affect the auxiliary output.

REFERENCES

- 1 EBU Technical Document 3244, Specifications of the Radio Data System RDS for VHF/FM Sound Broadcasting.
- 2 AN459, A Monitor for the MC68HC05E0.

APPENDIX (listing) follows

```

0001 *****
0002 *
0003 * HC05E0 RDS Decoder. *
0004 *
0005 * P. Topping 29th February '92 *
0006 *
0007 *****
0008
0009 0000 PORTA EQU $00 PORT A ADDRESS
0010 0001 PORTB EQU $01 " B "
0011 0002 PORTC EQU $02 " C "
0012 0003 PORTD EQU $03 " D "
0013 0004 PORTE EQU $04 " E "
0014 0005 PORTAD EQU $05 PORT A DATA DIRECTION REG.
0015 0006 PORTBD EQU $06 " B " " " "
0016 0007 PORTCD EQU $07 " C " " " "
0017 0008 PORTDD EQU $08 " D " " " "
0018 0009 PORTED EQU $09 " E " " " "
0019 000a TAP EQU $0A TIMER A PRE-SCALLER
0020 000b TBS EQU $0B TIMER B SCALLER
0021 000c TCR EQU $0C TIMER CONTROL REGISTER
0022 000e ICR EQU $0E INTERRUPT CONTROL REGISTER
0023 0012 PORTDSF EQU $12 PORTD SPECIAL FUNCTIONS
0024
0025 0009 ND EQU 9 No. BCD DIGITS
0026
0027 0030 ORG $0030
0028
0029 0030 Q RMB 9 BCD WORKING NUMBERS
0030 0039 TMQ RMB 9 SCRATCH
0031 0042 P RMB 9 WORKING NUMBER 2
0032 004b TMP RMB 9 MULT. OVER. OR DIV. REMAINDER
0033 0054 R RMB 9 WORKING NUMBER 3
0034 005d MJD RMB 9 MODIFIED JULIAN DAY NUMBER
0035 0066 YR RMB 9 YEAR
0036 006f MINTH RMB 2 MONTH
0037 0071 DOM RMB 2 DATE
0038 0073 DOW RMB 1 DAY OF WEEK
0039 0074 BMJD RMB 3 BINARY MJD
0040 0077 DIST RMB 1 DISPLAY TRANSIENT TIMEOUT COUNTER
0041 0078 SLEPT RMB 1 SLEEP TIMER MINUTES COUNTER
0042 0079 RDSTO RMB 1 RDS TIMEOUT COUNTER
0043 007a DAT RMB 4 SERIAL DATA BUFFER
0044 007e TMPGRP RMB 8 TEMPORARY GROUP DATA
0045 0086 GROUP RMB 8 COMPLETE GROUP DATA
0046 008e PTY RMB 1 PROGRAM-TYPE CODE (CURRENT)
0047 008f PI RMB 2 PROGRAM IDENTIFICATION CODE
0048 0091 PIN RMB 2 PROGRAM ITEM NUMBER
0049 0093 LEV RMB 1 VALID BLOCK LEVEL
0050 0094 BIT RMB 1 BIT LEVEL
0051 0095 ITMP1 RMB 1 TEMP BYTE FOR USE IN IRQ
0052 0096 SYN RMB 2 SYNDROME
0053 0098 CONF RMB 1 SYNDROME CONFIDENCE
0054 0099 TH8 RMB 1 TICS (EIGHTHS OF SECONDS)
0055 009a SEC RMB 1 SECONDS
0056 009b MIN RMB 1 MINUTES
0057 009c OUR RMB 1 HOURS
0058 009d AMIN RMB 1 ALARM MINUTES
0059 009e AOUR RMB 1 ALARM HOURS
0060 009f DISP1 RMB 1 RT DISPLAY POINTER #1
0061 00a0 DISP2 RMB 1 RT DISPLAY POINTER #2
0062 00a1 W1 RMB 1 W
0063 00a2 W2 RMB 1 O
0064 00a3 W3 RMB 1 R
0065 00a4 W4 RMB 1 K
0066 00a5 W5 RMB 1 I
0067 00a6 W6 RMB 1 N
0068 00a7 W7 RMB 1 G
0069 00a8 W8 RMB 1
0070 00a9 KEY RMB 1 CODE OF PRESSED KEY
0071 00aa KOUNT RMB 1 KEYBOARD COUNTER
0072 00ab CARRY RMB 1 BCD CARRY
0073 00ac COUNT RMB 1 LOOP COUNTER
0074 00ad NUM1 RMB 1 1ST No. POINTER (ADD & SUBTRACT)
0075 00ae NUM2 RMB 1 2ND No. POINTER (ADD & SUBTRACT)
0076 00af RTDIS RMB 1 RDS DISPLAY TYPE
0077 00b0 DI RMB 1 DECODER IDENTIFICATION
0078 00b1 DISP RMB 16 LCD MODULE BUFFER
0079 00c1 PSN RMB 8 PS NAME
0080
0081 00c9 STAT2 RMB 1 0: VALID SYNDROME
0082 * 1: VALID GROUP
0083 * 2: RT DISPLAY
0084 * 3: UPDATE DISPLAY
0085 * 4: CLEAR DISPLAY
0086 * 5: SPACE FLAG
0087 00ca STAT3 RMB 1 0: M/S, 0: M, 1: S
0088 * 1: TEXTA/TEXTB BIT (RT)
0089 * 2: TA FLAG
0090 * 3: TP FLAG
0091 * 4: KEY REPEATING
0092 * 5: KEY FUNCTION PERFORMED
0093 * 6: UPDATE DATE
0094 00cb STAT4 RMB 1 0: DISPLAY TRANSIENT
0095 * 1: SLEEP TIMER RUNNING
0096 * 2: SLEEP DISPLAY
0097 * 3: ALARM DISPLAY
0098 * 4: ALARM ARMED
0099 * 5: ALARM SET-UP
0100 * 6: ALARM HOURS (SET-UP)
0101 * 7: RDS DISPLAYS
0102
0103 00cc RMB 33 not used
0104 00ed STACK RMB 18 19 BYTES USED (1 INTERRUPT
0105 00ff SP RMB 1 AND 7 NESTED SUBROUTINES)
0106
0107 0100 ORG $0100
0108
0109 0100 RT RMB 69 RADIOTEXT
0110 0145 EON RMB 176 EON DATA (MAX: 11 NETWORKS)

```

```

0112 e000          ORG      $E000
0113
0114 *STRST  JMP      START      RESET VECTOR          ($0400 DURING DE-BUG)
0115 *IRQ    JMP      SDATA      IRQ          ($0403 DURING DE-BUG)
0116 *TIMERB JMP      START      TIMER A INTERRUPT (NOT USED, $0406 DURING DE-BUG)
0117 *TIMERB JMP      TINTB      TIMER B INTERRUPT ($0409 DURING DE-BUG)
0118 *SERINT JMP      START      SERIAL INTERRUPT (NOT USED, $040C DURING DE-BUG)
0119
0120 *****
0121 *
0122 *          Reset routine - setup ports.
0123 *
0124 *****
0125
0126 e000 a6 c3      START  LDA      #$C3          ENABLE PORTD SPECIAL FUNCTIONS
0127 e002 b7 12      STA      PORTDSF        P02, R/W, A14 & A15 (0,1,6,7)
0128 e004 a6 45      LDA      #$45          ENABLE POSITIVE EDGE/LEVEL
0129 e006 b7 0e      STA      ICR           INTERRUPTS
0130 e008 a6 01      LDA      #1           TIMER B SCALER: /2
0131 e00a b7 0b      STA      TBS          125 ms INTERRUPTS (4.194 MHz XTAL)
0132 e00c a6 3f      LDA      #63          TIMER A PRE-SCALER: /64
0133 e00e b7 0a      STA      TAP          64Hz IDLE LOOP
0134
0135 e010 3f 00      CLR      PORTA
0136 e012 a6 ff      LDA      #$FF          E0BUG DISPLAY/KEYBOARD I/O
0137 e014 b7 05      STA      PORTAD        NOT USED IN RDS APPLICATION
0138 e016 3f 01      CLR      PORTB          0, 1: SERIAL CLOCK AND DATA
0139 e018 a6 cb      LDA      #$CB          2: RDS DATA IN, 3: VFD SELECT
0140 e01a b7 06      STA      PORTBD        4, 5: KEYBOARD IN, 6, 7: KEYBOARD OUT
0141 e01c 3f 02      CLR      PORTC
0142 e01e a6 ff      LDA      #$FF          ALL OUT, LCD DATA BUS
0143 e020 b7 07      STA      PORTCD
0144 e022 a6 3c      LDA      #$3C          BITS 2, 3 & 4 OUT, LCD
0145 e024 3f 03      CLR      PORTD          2: RS, 3: R/W, 4: CLOCK, 5: LED (TA=TP=1)
0146 e026 b7 08      STA      PORTDD        0, 1, 6 & 7 USED DURING DE-BUG
0147 e028 a6 0c      LDA      #$0C          BIT0: INPUT, ENABLE SLEEP TIMER AT ALARM TIME
0148 e02a b7 04      STA      PORTE          BIT1: INPUT, ENABLE ALARM OUTPUT
0149 *          LDA      #$0C          BIT2: ALARM OUTPUT (ACTIVE LOW)
0150 e02c b7 09      STA      PORTED        BIT3: RADIO ON OUTPUT (ACTIVE HIGH)
0151
0152 *****
0153 *
0154 *          Initialise LCD.
0155 *
0156 *****
0157
0158 e02e a6 30          LDA      #$30
0159 e030 cd eb 65      JSR      CLOCK          INITIALISE LCD
0160 e033 cd eb e6      JSR      CLREON         CLEAR EON DATA
0161 e036 cd eb e6      JSR      CLREON
0162 e039 cd eb e6      JSR      CLREON         4 TIMES TO PROVIDE A 5ms DELAY
0163 e03c cd eb e6      JSR      CLREON         FOR LCD MODULE INITIALISATION
0164 e03f a6 30          LDA      #$30
0165 e041 cd eb 65      JSR      CLOCK          INITIALISE LCD
0166
0167 e044 ae 30          LDX      #Q            INITIALISE RAM
0168 e046 7f          CLR      0,X
0169 e047 5c          INCX
0170 e048 a3 ed          CPX      #STACK
0171 e04a 26 fa          BNE      CLOOP         PROVIDES A 1ms DELAY FOR LCD
0172
0173 e04c a6 30          LDA      #$30
0174 e04e cd eb 65      JSR      CLOCK          INITIALISE LCD
0175
0176 e051 cd eb 6c          JSR      WAIT
0177 e054 a6 30          LDA      #$30          1-LINE DISPLAY
0178 e056 cd eb 65      JSR      CLOCK          LATCH IT
0179 e059 cd eb 6c          JSR      WAIT
0180 e05c a6 08          LDA      #$08          SWITCH DISPLAY OFF
0181 e05e cd eb 65      JSR      CLOCK          LATCH IT
0182 e061 cd eb 6c          JSR      WAIT
0183 e064 a6 01          LDA      #$01          CLEAR DISPLAY
0184 e066 cd eb 65      JSR      CLOCK          LATCH IT
0185 e069 cd eb aa          JSR      INITD
0186
0187 *****
0188 *
0189 *          Vectors for de-bug using E0BUG monitor.
0190 *
0191 *****
0192
0193 *          LDA      #$0C          ENABLE EXTERNAL RAM WRITE
0194 *          STA      TCR
0195
0196 *          LDA      #$04          VECTORS FOR E0 MONITOR
0197 *          STA      $0201
0198 *          STA      $0204          USING JUMP TABLE AT $0400
0199 *          STA      $0207
0200 *          STA      $020A          (LINES 126-130)
0201 *          LDA      #$03
0202 *          STA      $0202          IRQ ($0403)
0203 *          LDA      #$06
0204 *          STA      $0205          TIMER A ($0406)
0205 *          LDA      #$09
0206 *          STA      $0208          TIMER B ($0409)
0207 *          LDA      #$0C
0208 *          STA      $020B          SERIAL ($040C)
0209
0210 *****
0211 *
0212 *          Enable interrupts.
0213 *
0214 *****
0215
0216 e06c a6 0b          LDA      #$0B          EDGE SENSITIVE IRQ, TIMERS A & B ENABLED
0217 e06e b7 0c          STA      TCR          SUB-SYS CLK = 262144 Hz (4.194 MHz XTAL)
0218 *
0219 e070 9a          CLI          DISSABLE EXTERNAL RAM WRITE

```

```

0221
0222
0223
0224
0225
0226
0227 e071 09 0e fd
0228 e074 19 0e
0229
0230 e076 01 cb 07
0231 e079 b6 77
0232 e07b 26 03
0233 e07d cd e8 0a
0234
0235 e080 07 c9 05
0236 e083 cd e6 b6
0237 e086 17 c9
0238
0239 e088 09 cb 1d
0240 e08b b6 9e
0241 e08d b1 9c
0242 e08f 26 17
0243 e091 b6 9d
0244 e093 b1 9b
0245 e095 26 11
0246 e097 b6 9a
0247 e099 26 0d
0248 e09b 16 04
0249
0250 e09d 02 04 02
0251 e0a0 15 04
0252 e0a2 00 04 03
0253 e0a5 cd e2 09
0254
0255 e0a8 03 cb 08
0256 e0ab b6 78
0257 e0ad 26 04
0258 e0af 13 cb
0259 e0b1 17 04
0260
0261 e0b3 cd e1 11
0262 e0b6 cd e1 6f
0263
0264 e0b9 b6 ca
0265 e0bb a4 0c
0266 e0bd a1 0c
0267 e0bf 27 07
0268 e0c1 0a 03 09
0269 e0c4 1a 03
0270 e0c6 20 05
0271 e0c8 0b 03 02
0272 e0cb 1b 03
0273
0274 e0cd 0d ca 02
0275 e0d0 ad 02
0276 e0d2 20 9d
0277
0278
0279
0280
0281
0282
0283
0284 e0d4 b6 76
0285 e0d6 b7 68
0286 e0d8 b6 75
0287 e0da b7 67
0288 e0dc b6 74
0289 e0de b7 66
0290 e0e0 ae 54
0291 e0e2 bf ad
0292 e0e4 cd ef 86
0293 e0e7 3c 5c
0294 e0e9 ae 5d
0295 e0eb cd ef 86
0296 e0ee a6 11
0297 e0f0 b7 a6
0298 e0f2 34 66
0299 e0f4 36 67
0300 e0f6 36 68
0301 e0f8 24 07
0302 e0fa ae 5d
0303 e0fc bf ae
0304 e0fe cd ee 33
0305 e101 ae 54
0306 e103 bf ae
0307 e105 cd ee 33
0308 e108 3a a6
0309 e10a 26 e6
0310 e10c ld ca
0311 e10e cc ef 95

*****
*
* Idle loop.
*
*****

IDLE BRCLR 4,ICR,* 64 Hz
      BCLR 4,ICR

NO2D BRCLR 0,STAT4,NOPS DISPLAY TRANSIENT ?
      LDA DIST
      BNE NOPS YES, TIMED OUT ?
      JSR CLTR YES, CLEAR TRANSIENT DISPLAYS

NOPS BRCLR 3,STAT2,SCAN DISPLAY UPDATE REQUIRED ?
      JSR MOD YES, DO IT
      BCLR 3,STAT2 AND CLEAR FLAG

SCAN BRCLR 4,STAT4,CHSLP ALARM ARMED ?
      LDA AGUR YES, COMPARE ALARM HOURS
      CMP OUR WITH TIME
      BNE CHSLP SAME ?
      LDA AMIN YES, COMPARE ALARM MINUTES
      CMP MIN WITH TIME
      BNE CHSLP SAME ?
      LDA SEC ONLY ALLOW WAKE-UP IN FIRST SECOND
      BNE CHSLP TO PREVENT SWITCH-OFF LOCKOUT
      BSET 3,PORTE YES, SWITCH ON

FULON2 BRSET 1,PORTE,FULON2 ALARM ENABLED (SWITCH) ?
        BCLR 2,PORTE YES, SOUND ALARM
        BRSET 0,PORTE,CHSLP SLEEP TIMER AT ALARM TIME ?
        JSR INSLP YES, START SLEEP TIMER

CHSLP BRCLR 1,STAT4,FLN SLEEP TIMER RUNNING ?
        LDA SLEPT YES
        BNE FLN TIME TO FINISH ?
        BCLR 1,STAT4 YES, CLEAR FLAG
        BCLR 3,PORTE AND SWITCH OFF

FLN JSR KBD READ KEYBOARD
     JSR KEYP EXECUTE KEY

      LDA STAT3
      AND #S0C
      CMP #S0C TA AND TP BOTH HIGH ?
      BEQ TATP
      BRSET 5,PORTD,IOOK NO, I/O LINE ALREADY HIGH ?
      BSET 5,PORTD NO, MAKE IT HIGH
      BRA IOOK
      BRCLR 5,PORTD,IOOK TA=TP=1, I/O LINE ALREADY LOW ?
      BCLR 5,PORTD NO, MAKE IT LOW

IOOK BRCLR 6,STAT3,IDLEJ UPDATE DATE ?
      BSR MJDAT YES, CONVERT FROM MJD
      BRA IDLE

*****
*
* Extract MJD and convert to decimal.
*
*****

MJDAT LDA BMJD+2
      STA YR+2
      LDA BMJD+1
      STA YR+1
      LDA BMJD
      STA YR
      LDX #R CLEAR
      STX NUM1
      JSR CLRAS R
      INC R+ND-1 R <- 1
      LDX #MJD
      JSR CLRAS CLEAR MJD
      LDA #17 17 BITS TO CONVERT
      STA W6
      LOOPJ LSR YR MOVE OUT
           ROR YR+1
           ROR YR+2 FIRST (LS) BIT
           EOC NXTJ ZERO ?
           LDX #MJD ONE, ADD
           STX NUM2 CURRENT VALUE
           JSR ADD OF R
           LDX #R ADD R
           STX NUM2 TO
           JSR ADD ITSELF
           DEC W6 ALL
           BNE LOOPJ DONE ?
           BCLR 6,STAT3 MJD UPDATED
           JMP MJDC CONVERT MJD TO DAY, DATE, MONTH & YEAR

```

```

0313
0314
0315
0316
0317
0318
0319 e111 a6 20
0320 e113 ae 02
0321 e115 48
0322 e116 a4 c0
0323 e118 aa 08
0324 e11a b7 01
0325 e11c b6 01
0326 e11e a5 30
0327 e120 26 07
0328 e122 5a
0329 e123 26 f0
0330 e125 3f a9
0331 e127 20 0c
0332
0333 e129 b6 01
0334 e12b a4 f0
0335 e12d b1 a9
0336 e12f 27 04
0337 e131 b7 a9
0338 e133 3f aa
0339 e135 3c aa
0340 e137 b6 aa
0341 e139 09 ca 04
0342 e13c a1 0a
0343 e13e 20 08
0344 e140 a1 03
0345 e142 25 29
0346 e144 27 1b
0347 e146 a1 30
0348 e148 22 06
0349 e14a b6 a9
0350 e14c 27 19
0351 e14e 98
0352 e14f 81
0353
0354 e150 b6 a9
0355 e152 a1 50
0356 e154 27 04
0357 e156 a1 90
0358 e158 26 0f
0359 e15a 0b cb 0c
0360 e15d 18 ca
0361 e15f 3f aa
0362 e161 b6 a9
0363 e163 27 02
0364 e165 99
0365 e166 81
0366 e167 1b ca
0367 e169 19 ca
0368 e16b 3f aa
0369 e16d 98
0370 e16e 81
0371
0372
0373
0374
0375
0376
0377
0378 e16f 24 26
0379 e171 b6 a9
0380 e173 a1 50
0381 e175 27 07
0382 e177 a1 90
0383 e179 27 03
0384 e17b 0a ca 19
0385
0386 e17e 5f
0387 e17f d6 e1 98
0388 e182 b1 a9
0389 e184 27 0b
0390 e186 c1 e1 a4
0391 e189 27 0c
0392 e18b 5c
0393 e18c 5c
0394 e18d 5c
0395 e18e 5c
0396 e18f 20 ee
0397 e191 la ca
0398 e193 5c
0399 e194 dd e1 98
0400 e197 81
0401
0402
0403
0404
0405
0406
0407
0408 e198 60
0409 e199 cc e1 a8
0410 e19c a0
0411 e19d cc e1 c7
0412 e1a0 50
0413 e1a1 cc e1 fa
0414 e1a4 90
0415 e1a5 cc e2 26

*****
*
* Keyboard routine.
*
*****
KBD LDA #20
LDX #2
KEY1 LSLA SELECT ROW
AND #20 BITS 6 & 7 ONLY
ORA #08 VFD ENABLE HIGH
STA PORTB
ROW LDA PORTB READ KEYBOARD
BIT #30 ANY INPUT LINE HIGH ?
BNE L1
DECCX NO, TRY NEXT COLUMN
BNE KEY1 LAST COLUMN ?
CLR KEY YES, NO KEY PRESSED
BRA EXIT
L1 LDA PORTB READ KEYBOARD
AND #F0
CMP KEY SAME AS LAST TIME ?
BEQ EXIT
STA KEY NO, SAVE THIS KEY
CLR KOUNT
EXIT INC KOUNT YES, THE SAME
LDA KOUNT
BRCLR 4,STAT3,NRML REPEATING ?
CMP #10 YES, REPEAT AT 6 Hz
BRA GON2
NRML CMP #3 NO, 3 THE SAME ?
BLO KCLC IF NOT DO NOTHING
BEQ GOON IF 3 THEN PERFORM KEY FUNCTION
CMP #48 MORE THAN 3, MORE THAN 48 (750ms) ?
BHI GOON2 TIME TO DO SOMETHING ?
LDA KEY NO
BEQ RKEY KEY PRESSED ?
CLC
RTS YES BUT DO NOTHING
GOON2 LDA KEY
CMP #50 SLEEP (DEC.)
BEQ GOON3
CMP #90
BNE DNT2 RDS (INC.)
IF NOT A REPEAT KEY, DO NOTHING
GOON3 BRCLR 5,STAT4,DNT2 REPEAT KEY, BUT IS MODE ALARM SET-UP ?
BSET 4,STAT3 YES, SET REPEAT FLAG
CLR KOUNT
GOON LDA KEY
BEQ RKEY SOMETHING TO DO ?
SEC YES, SET C
RTS
RKEY BCLR 5,STAT3 NO, CLEAR DONE FLAG
DNT2 BCLR 4,STAT3 CLEAR REPEAT FLAG
CLR KOUNT CLEAR COUNTER
KCLC CLC
RTS
*****
*
* Execute key function.
*
*****
KEYP BCC DNT ANYTHING TO DO ?
KEYP2 LDA KEY YES, GET KEY
CMP #50 SLEEP (DEC.)
BEQ RPT
CMP #90 RDS (INC.)
BEQ RPT
BRSET 5,STAT3,DNT NOT A REPEAT KEY, DONE FLAG SET ?
RPT CLRX
RJ LDA CTAB,X FETCH KEYCODE
CMP KEY THIS ONE ?
BEQ PJ YES
CMP LAST NO, LAST CHANCE ?
BEQ DNT YES, ABORT
INCX NO
INCX TRY
INCX THE
INCX NEXT
BRA RJ KEY
PJ BSET 5,STAT3 KEY FUNCTION DONE
INCX
JSR CTAB,X
DNT RTS
*****
*
* Keyboard jump table.
*
*****
CTAB FCB $60 ALARM
JMP ALARM
FCB $A0 ON/OFF
JMP ONOFF
FCB $50 SLEEP TIMER START
JMP SLEEP
LAST FCB $90 RDS DISPLAYS
JMP RDS

```

```

0417
0418
0419
0420
0421
0422
0423 e1a8 05 04 4c
0424 e1ab 07 cb 0b
0425 e1ae 09 cb 04
0426 e1b1 19 cb
0427 e1b3 20 09
0428 e1b5 18 cb
0429 e1b7 20 05
0430 e1b9 cd e8 0a
0431 e1bc 16 cb
0432 e1be 1b cb
0433 e1c0 a6 19
0434 e1c2 b7 77
0435 e1c4 10 cb
0436 e1c6 81
0437
0438
0439
0440
0441
0442
0443
0444 e1c7 05 04 2d
0445 e1ca 07 cb 1c
0446 e1cd 09 cb 19
0447 e1d0 0a cb 0b
0448 e1d3 1a cb
0449 e1d5 1c cb
0450 e1d7 a6 50
0451 e1d9 b7 77
0452 e1db 10 cb
0453 e1dd 81
0454
0455 e1de 0c cb 04
0456 e1e1 1b cb
0457 e1e3 20 f2
0458 e1e5 1d cb
0459 e1e7 20 ee
0460
0461
0462
0463
0464
0465
0466
0467 e1e9 cd e8 0a
0468 e1ec 13 cb
0469 e1ee 06 04 03
0470 e1f1 16 04
0471 e1f3 81
0472 e1f4 17 04
0473 e1f6 81
0474 e1f7 14 04
0475 e1f9 81
0476
0477
0478
0479
0480
0481
0482
0483 e1fa 05 04 fa
0484 e1fd 0b cb 03
0485 e200 cc e2 79
0486 e203 04 cb 10
0487 e206 02 cb 06
0488 e209 a6 3c
0489 e20b b7 78
0490 e20d 12 cb
0491 e20f cd e8 0a
0492 e212 14 cb
0493 e214 20 08
0494 e216 b6 78
0495 e218 a0 05
0496 e21a b7 78
0497 e21c 2b eb
0498 e21e a6 19
0499 e220 b7 77
0500 e222 10 cb
0501 e224 20 cb
0502
0503
0504
0505
0506
0507
0508
0509 e226 05 04 ce
0510 e229 0a cb 29
0511 e22c 07 04 17
0512 e22f 0e cb 03
0513 e232 05 c9 12
0514 e235 1e cb
0515 e237 b6 af
0516 e239 4c
0517 e23a a1 13
0518 e23c 27 09
0519 e23e b7 af
0520 e240 a6 64
0521 e242 b7 77
0522 e244 10 cb
0523 e246 81
0524
0525 e247 cd e8 0a
0526 e24a 14 c9
0527 e24c a6 09
0528 e24e b7 9f
0529 e250 a6 01
0530 e252 b7 a0
0531 e254 81
*****
*
* Alarm key.
*
*****
ALARM BRCLR 2,PORTE,ALRG ALARM RINGING ?
BRCLR 3,STAT4,ADON NO, ALARM DISPLAY ON ?
BRCLR 4,STAT4,ALOF YES, ALARM ON ?
BCLR 4,STAT4 YES, SWITCH OFF
BRA UDCNT
ALOF BSET 4,STAT4 NO, SWITCH ON
BRA UDCNT
ADON JSR CLTR
BSET 3,STAT4 ALARM DISPLAY FLAG
UDCNT BCLR 5,STAT4 CANCEL SET-UP
LDA #25 3 SECOND TIMEOUT
STA DIST
ABOA BSET 0,STAT4 SET DISPLAY TRANSIENT FLAG
RTS
*****
*
* On/off key (alarm set-up).
*
*****
ONOFF BRCLR 2,PORTE,ALRG ALARM RINGING ?
BRCLR 3,STAT4,NOTALR NO, ALARM DISPLAY ?
BRCLR 4,STAT4,NOTALR YES, ALARM ARMED ?
BRSET 5,STAT4,AISM YES, ALREADY SET-UP MODE ?
BSET 5,STAT4 NO, ENTER SET-UP MODE
BSET 6,STAT4 WITH HOURS
A5SD LDA #80
STA DIST
BSET 0,STAT4 SET DISPLAY TRANSIENT FLAG
NTB2 RTS
AISM BRSET 6,STAT4,MSM SET-UP HOURS ?
BCLR 5,STAT4 NO, CANCELL SET-UP
BRA A5SD
MSM BCLR 6,STAT4 YES, MAKE IT MINUTES
BRA A5SD
*****
*
* On/off key (normal function).
*
*****
NOTALR JSR CLTR CLEAR DISPLAY TRANSIENTS
BCLR 1,STAT4 CANCEL SLEEP TIMER
BRSET 3,PORTE,ALRON ON ?
SODM BSET 3,PORTE NO, SWITCH ON
RTS
ALRON BCLR 3,PORTE YES, SWITCH OFF
RTS
ALRG BSET 2,PORTE CANCEL ALARM
RTS
*****
*
* Sleep key.
*
*****
SLEEP BRCLR 2,PORTE,ALRG ALARM RINGING ?
BRCLR 5,STAT4,NOTAL NO, ALARM SET-UP ?
JMP PDEC YES
NOTAL BRSET 2,STAT4,DECS NO, ALREADY SLEEP DISPLAY ?
BRSET 1,STAT4,STR2 NO, SLEEP TIMER ALREADY RUNNING ?
INSLP LDA #60 NO, INITIALISE SLEEP TIMER
STA SLEPT
STR2 BSET 1,STAT4 START SLEEP TIMER
JSR CLTR YES, CLEAR DISPLAY TRANSIENTS
BSET 2,STAT4 SLEEP DISPLAY
BRA SLPTOK NO DECREMENT IF FIRST TIME
DECS LDA SLEPT DECREMENT SLEEP TIMER
SUB #5
STA SLEPT
SLPTOK BMI INSLP IF UNDERFLOW WRAP ROUND TO 60
LDA #25
STA DIST
BSET 0,STAT4 START DISPLAY TRANSIENT
BRA SODM
*****
*
* RDS display key.
*
*****
RDS BRCLR 2,PORTE,ALRG ALARM RINGING ?
BRSET 5,STAT4,PINC NO, ALARM SET-UP ?
BRCLR 3,PORTE,SRT3 NO, STANDBY ?
BRSET 7,STAT4,NOTRT ALREADY RDS ?
BRCLR 2,STAT2,NORT ALREADY RT DISPLAY ?
BSET 7,STAT4 SET RDS DISPLAY FLAG
LDA RTDIS MOVE ON
INCA
CMP #19
BEQ NORT
STA RTDIS
LDA #100 12 SECOND TIMEOUT
STA DIST
SRT3 BSET 0,STAT4 RE-START TRANSIENT TIMEOUT
RTS
NORT JSR CLTR CLEAR DISPLAY TRANSIENTS
BSET 5,STAT4 SET RT DISPLAY FLAG
LDA #9
STA DISP1
LDA #1
STA DISP2
RTS

```

```

0533
0534
0535
0536
0537
0538
*****
*
* Increment alarm time.
*
*****
0539 e255 0c cb 0e PINC BRSET 6,STAT4,IHR SET-UP HOURS ?
0540 e258 b6 9d LDA AMIN NO, MINUTES
0541 e25a a1 3b CMP #59
0542 e25c 24 04 BHS TOO H
0543 e25e 3c 9d INC AMIN
0544 e260 20 0c BRA T5S
0545 e262 3f 9d TOO H CLR AMIN
0546 e264 20 08 BRA T5S
0547 e266 b6 9e IHR LDA AOUR
0548 e268 a1 17 CMP #23
0549 e26a 24 09 BHS HTO H
0550 e26c 3c 9e INC AOUR
0551 e26e a6 50 T5S LDA #80 10 SECOND TIMEOUT
0552 e270 b7 77 STA DIST
0553 e272 10 cb BSET 0,STAT4 SET DISPLAY TRANSIENT FLAG
0554 e274 81 RTS
0555 e275 3f 9e HTO H CLR AOUR
0556 e277 20 f5 BRA T5S
0557
0558
*****
*
* Decrement alarm time.
*
*****
0559
0560
0561
0562
0563
0564 e279 0c cb 0e PDEC BRSET 6,STAT4,IHRD SET-UP HOURS ?
0565 e27c 3d 9d TST AMIN NO, MINUTES
0566 e27e 27 04 BEQ MZ
0567 e280 3a 9d DEC AMIN
0568 e282 20 0c BRA T5SD
0569 e284 a6 3b MZ LDA #59
0570 e286 b7 9d STA AMIN
0571 e288 20 06 BRA T5SD
0572 e28a 3d 9e IHRD TST AOUR
0573 e28c 27 09 BEQ HZ
0574 e28e 3a 9e DEC AOUR
0575 e290 a6 50 T5SD LDA #80 10 SECOND TIMEOUT
0576 e292 b7 77 STA DIST
0577 e294 10 cb BSET 0,STAT4 SET DISPLAY TRANSIENT FLAG
0578 e296 81 RTS
0579 e297 a6 17 HZ LDA #23
0580 e299 b7 9e STA AOUR
0581 e29b 20 f3 BRA T5SD
0582
0583
*****
*
* Timer interrupt routine.
*
*****
0584
0585
0586
0587
0588
0589 e29d 3c 9f TINTB INC DISP1 DISP1 DISP2 DISPLAY
0590 e29f b6 9f LDA DISP1 0 - 8 0 PTY
0591 e2a1 a1 08 CMP #8 9 - 78 1 - 70 MOVING RT
0592 e2a3 23 06 BLS NWR 78 - 88 70 END OF RT
0593 e2a5 a1 4e CMP #78
0594 e2a7 22 02 BHI NWR END OF RADIOTEXT ?
0595 e2a9 3c a0 INC DISP2 NO, MOVE RADIOTEXT ONE CHARACTER
0596 e2ab a1 58 CMP #88 2 SECONDS AT END OF RADIOTEXT
0597 e2ad 25 02 BLO NWR 2
0598 e2af 15 c9 BCLR 2,STAT2 RETURN TO NORMAL DISPLAY
0599 e2b1 1b 0e NWR2 BCLR 5,ICR CLEAR TIMER B INTERRUPT FLAG
0600 e2b3 16 c9 BSET 3,STAT2 UPDATE DISPLAY
0601 e2b5 3c 99 CLCK INC TH8 UPDATE EIGHTHS OF SECONDS
0602 e2b7 3a 77 DEC DIST DECREMENT TRANSIENT DISPLAY TIMER
0603 e2b9 3c 79 INC RDSTO
0604 e2bb b6 79 LDA RDSTO
0605 e2bd a1 50 CMP #80 10S WITHOUT A GROUP 0 OR 15 ?
0606 e2bf 25 10 BLO RDSOK
0607 e2c1 15 ca N14B BCLR 2,STAT3 YES, CLEAR TA FLAG
0608 e2c3 3f 8e CLR PTY PROGRAM TYPE
0609 e2c5 3f 8f CLR PI AND
0610 e2c7 3f 90 CLR PI+1 PI CODE
0611 e2c9 3f 91 CLR PIN AND
0612 e2cb 3f 92 CLR PIN+1 PIN
0613 e2cd 3f b0 CLR DI AND DI
0614 e2cf 11 ca N14B BCLR 0,STAT3 AND M/S
0615 e2d1 b6 99 RDSOK LDA TH8 EIGHTHS OF SECONDS
0616 e2d3 a1 08 CMP #8
0617 e2d5 26 32 BNE NOTC PAST 7 ?
0618 e2d7 3f 99 CLR TH8 YES, CLEAR
0619 e2d9 3c 9a INC SEC UPDATE SECONDS
0620 e2db b6 9a LDA SEC
0621 e2dd a1 38 CMP #56
0622 e2df 26 02 BNE NOT5
0623 e2e1 3a 78 DEC SLEPT DECREMENT SLEEP TIMER MINUTES
0624 e2e3 a1 3c NOT5 CMP #60
0625 e2e5 26 22 BNE NOTC PAST 59 ?
0626 e2e7 3f 9a CLR SEC YES, CLEAR
0627 e2e9 3c 9b INC MIN UPDATE MINUTES
0628 e2eb b6 9b LDA MIN
0629 e2ed a1 3c CMP #60
0630 e2ef 26 18 BNE NOTC PAST 59 ?
0631 e2f1 3f 9b CLR MIN YES, CLEAR
0632 e2f3 3c 9c INC OUR UPDATE HOURS
0633 e2f5 b6 9c LDA OUR
0634 e2f7 a1 18 CMP #24
0635 e2f9 26 0e BNE NOTC PAST 23 ?
0636 e2fb 3f 9c CLR OUR YES CLEAR
0637 e2fd 3c 7e INC BMJD+2 AND ADD A DAY
0638 e2ff 26 06 BNE NOTD
0639 e301 3c 75 INC BMJD+1
0640 e303 26 02 BNE NOTD INC BMJD only ever executes once, at midnight
0641 e305 3c 74 INC BMJD on the night of Thu/Fri 22/23 April 2038.
0642 e307 1c ca BSET 6,STAT3 UPDATE DATE
0643 e309 80 NOTD RTI

```

```

0645
0646
0647
0648
0649
0650
0651
0652 e30a 04 01 00 SDATA BRSET 2,PORTB,*+3
0653 e30d 39 7d ROL DAT+3
0654 e30f 39 7c ROL DAT+2
0655 e311 39 7b ROL DAT+1
0656 e313 39 7a ROL DAT
0657 e315 01 c9 0b BRCLR 0,STAT2,TRY2 BIT BY BIT CHECK ?
0658 e318 3a 94 DEC BIT NO, WAIT FOR BIT 26
0659 e31a 27 03 BEQ TRY1 THIS TIME ?
0660 e31c 17 0e BCLR 3,ICR CLEAR IRQ INTERRUPT FLAG
0661 e31e 80 RTI
0662
0663 e31f a6 1a TRY1 LDA #26
0664 e321 b7 94 STA BIT
0665 e323 b6 7a TRY2 LDA DAT MSB (2 BITS)
0666 e325 a4 03 AND #3
0667 e327 97 TAX
0668 e328 b6 7b LDA DAT+1
0669 e32a b7 97 STA SYN+1 LSB
0670 e32c 01 7d 0a S03 BRCLR 0,DAT+3,S13
0671 e32f b6 97 LDA SYN+1
0672 e331 a8 1b EOR #S1B
0673 e333 b7 97 STA SYN+1
0674 e335 9f TXA
0675 e336 a8 03 EOR #S03
0676 e338 97 TAX
0677 e339 03 7d 0a S13 BRCLR 1,DAT+3,S23
0678 e33c b6 97 LDA SYN+1
0679 e33e a8 8f EOR #S8F
0680 e340 b7 97 STA SYN+1
0681 e342 9f TXA
0682 e343 a8 03 EOR #S03
0683 e345 97 TAX
0684 e346 05 7d 0a S23 BRCLR 2,DAT+3,S43
0685 e349 b6 97 LDA SYN+1
0686 e34b a8 a7 EOR #SA7
0687 e34d b7 97 STA SYN+1
0688 e34f 9f TXA
0689 e350 a8 02 EOR #S02
0690 e352 97 TAX
0691 e353 09 7d 0a S43 BRCLR 4,DAT+3,S53
0692 e356 b6 97 LDA SYN+1
0693 e358 a8 ee EOR #SEE
0694 e35a b7 97 STA SYN+1
0695 e35c 9f TXA
0696 e35d a8 01 EOR #S01
0697 e35f 97 TAX
0698 e360 0b 7d 0a S53 BRCLR 5,DAT+3,S63
0699 e363 b6 97 LDA SYN+1
0700 e365 a8 dc EOR #SDC
0701 e367 b7 97 STA SYN+1
0702 e369 9f TXA
0703 e36a a8 03 EOR #S03
0704 e36c 97 TAX
0705 e36d 0d 7d 0a S63 BRCLR 6,DAT+3,S73
0706 e370 b6 97 LDA SYN+1
0707 e372 a8 01 EOR #S01
0708 e374 b7 97 STA SYN+1
0709 e376 9f TXA
0710 e377 a8 02 EOR #S02
0711 e379 97 TAX
0712 e37a 0f 7d 0a S73 BRCLR 7,DAT+3,S02
0713 e37d b6 97 LDA SYN+1
0714 e37f a8 bb EOR #SBB
0715 e381 b7 97 STA SYN+1
0716 e383 9f TXA
0717 e384 a8 01 EOR #S01
0718 e386 97 TAX
0719 e387 01 7c 0a S02 BRCLR 0,DAT+2,S12
0720 e38a b6 97 LDA SYN+1
0721 e38c a8 76 EOR #S76
0722 e38e b7 97 STA SYN+1
0723 e390 9f TXA
0724 e391 a8 03 EOR #S03
0725 e393 97 TAX
0726 e394 03 7c 0a S12 BRCLR 1,DAT+2,S22
0727 e397 b6 97 LDA SYN+1
0728 e399 a8 55 EOR #S55
0729 e39b b7 97 STA SYN+1
0730 e39d 9f TXA
0731 e39e a8 03 EOR #S03
0732 e3a0 97 TAX
0733 e3a1 05 7c 0a S22 BRCLR 2,DAT+2,S32
0734 e3a4 b6 97 LDA SYN+1
0735 e3a6 a8 13 EOR #S13
0736 e3a8 b7 97 STA SYN+1
0737 e3aa 9f TXA
0738 e3ab a8 03 EOR #S03
0739 e3ad 97 TAX
0740 e3ae 07 7c 0a S32 BRCLR 3,DAT+2,S42
0741 e3b1 b6 97 LDA SYN+1
0742 e3b3 a8 9f EOR #S9F
0743 e3b5 b7 97 STA SYN+1
0744 e3b7 9f TXA
0745 e3b8 a8 03 EOR #S03
0746 e3ba 97 TAX
0747 e3bb 09 7c 0a S42 BRCLR 4,DAT+2,S62
0748 e3be b6 97 LDA SYN+1
0749 e3c0 a8 87 EOR #S87
0750 e3c2 b7 97 STA SYN+1
0751 e3c4 9f TXA
0752 e3c5 a8 02 EOR #S02
0753 e3c7 97 TAX
0754 e3c8 0d 7c 0a S62 BRCLR 6,DAT+2,S72
0755 e3cb b6 97 LDA SYN+1
0756 e3cd a8 6e EOR #S6E
0757 e3cf b7 97 STA SYN+1
0758 e3d1 9f TXA
0759 e3d2 a8 01 EOR #S01
0760 e3d4 97 TAX
0761 e3d5 0f 7c 09 S72 BRCLR 7,DAT+2,S33
0762 e3d8 b6 97 LDA SYN+1
0763 e3da a8 dc EOR #SDC
0764 e3dc b7 97 STA SYN+1
0765 e3de 9f TXA
0766 e3df a8 02 EOR #S02
0767 e3e1 b7 96 STA SYN
0768 e3e3 b6 97 LDA SYN+1
0769 e3e5 07 7d 02 BRCLR 3,DAT+3,S52
0770 e3e8 a8 f7 EOR #SF7
0771 e3ea 0b 7c 02 S52 BRCLR 5,DAT+2,FIN
0772 e3ed a8 b7 EOR #SB7
0773 e3ef b7 97 STA SYN+1

```

```

0775
0776
0777
0778
0779
0780
0781 e3f1 17 0e
0782
0783 e3f3 b6 93
0784 e3f5 a1 03
0785 e3f7 27 5d
0786 e3f9 a1 02
0787 e3fb 27 22
0788 e3fd a1 01
0789 e3ff 27 10
0790 e401 3f 93
0791
0792 e403 b6 97
0793 e405 a1 d8
0794 e407 26 31
0795 e409 b6 96
0796 e40b a1 03
0797 e40d 26 2b
0798 e40f 20 53
0799
0800 e411 b6 97
0801 e413 a1 d4
0802 e415 26 23
0803 e417 b6 96
0804 e419 a1 03
0805 e41b 26 1d
0806 e41d 20 45
0807
0808 e41f 06 80 0c
0809 e422 b6 97
0810 e424 a1 5c
0811 e426 26 12
0812 e428 b6 96
0813 e42a a1 02
0814 e42c 20 0a
0815
0816 e42e b6 97
0817 e430 a1 cc
0818 e432 26 06
0819 e434 b6 96
0820 e436 a1 03
0821 e438 27 2a
0822
0823
0824
0825
0826
0827
0828
0829
0830 e43a 3f 93
0831 e43c b6 98
0832 e43e a1 29
0833 e440 24 0e
0834 e442 11 c9
0835 e444 a1 0a
0836 e446 23 0b
0837 e448 3a 94
0838 e44a 26 06
0839 e44c a6 1a
0840 e44e b7 94
0841 e450 3a 98
0842 e452 80
0843 e453 18 c9
0844 e455 80
0845
0846 e456 b6 97
0847 e458 a1 58
0848 e45a 26 de
0849 e45c b6 96
0850 e45e a1 02
0851 e460 26 d8
0852 e462 12 c9
0853
0854 e464 00 c9 06
0855 e467 a6 26
0856 e469 b7 98
0857 e46b 10 c9
0858 e46d b6 98
0859 e46f a1 38
0860 e471 22 04
0861 e473 ab 04
0862 e475 b7 98
0863 e477 be 93
0864 e479 59
0865 e47a 3c 93
0866 e47c a6 1a
0867 e47e b7 94
0868 e480 36 7a
0869 e482 36 7b
0870 e484 36 7c
0871 e486 36 7a
0872 e488 36 7b
0873 e48a 36 7c
0874 e48c b6 7c
0875 e48e e7 7f
0876 e490 b6 7b
0877 e492 e7 7e
0878 e494 03 c9 be
0879 e497 ae 08
0880 e499 e6 7d
0881 e49b e7 85
0882 e49d 5a
0883 e49e 26 f9
*****
*
* Check for syndromes A, B, C & C'.
*
*****
BCLR 3,ICR CLEAR IRQ INTERRUPT FLAG
LDA LEV
CMP #3
BEQ TRYD
CMP #2
BEQ TRYC
CMP #1
BEQ TRYB
CLR LEV
TRYA LDA SYN+1 BLOCK 1
CMP #SD8
BNE NOTV
LDA SYN
CMP #S03
BNE NOTV
BRA VALID
TRYB LDA SYN+1 BLOCK 2
CMP #SD4
BNE NOTV
LDA SYN
CMP #S03
BNE NOTV
BRA VALID
TRYC BRSET 3,TMPGRP+2,TRYCD BLOCK 3 TYPE A
LDA SYN+1
CMP #S5C
BNE NOTV
LDA SYN
CMP #S02
BRA VC
TRYCD LDA SYN+1 BLOCK 3 TYPE B
CMP #SCC
BNE NOTV
LDA SYN
CMP #S03
VC BEQ VALID
*****
*
* Invalid syndrome handling, check for
* block 4 and save group data if valid.
*
*****
NOTV CLR LEV RESTART AT BLOCK 1
LDA CONF
CMP #41 CONFIDENCE 41 OR GREATER ?
BHS DECC
BCLR 0,STAT2 BIT BY BIT SYNDROME CHECK
CMP #10
BLS SKPDC CONFIDENCE 10 OR LESS ?
DEC BIT
BNE NNOW USE BIT COUNTER TO SLOW CONFIDENCE
LDA #26 DROP DURING BIT BY BIT ATTEMPT TO
STA BIT RE-SYNCRONISE
DEC CONF
NNOW RTI
SKPDC BSET 4,STAT2 10 OR LESS, INITIALISE DISPLAY
NOT4 RTI
TRYD LDA SYN+1
CMP #S58
BNE NOTV
LDA SYN
CMP #S02
BNE NOTV
BSET 1,STAT2 GROUP COMPLETE
VALID BRSET 0,STAT2,VLD VALID SYNDROME FLAG ALREADY SET ?
LDA #38 NO,
STA CONF INITIALISE CONFIDENCE (38+4=42)
BSET 0,STAT2 AND SET FLAG
VLD LDA CONF
CMP #56
BHI NMR
ADD #4
STA CONF
NMR LDX LEV
ROLX
INC LEV
LDA #26
STA BIT
ROR DAT
ROR DAT+1
ROR DAT+2
ROR DAT
ROR DAT+1
ROR DAT+2
LDA DAT+2
STA TMPGRP+1,X
LDA DAT+1
STA TMPGRP,X
BRCLR 1,STAT2,NOT4 GROUP COMPLETE ?
XFER LDX #8
TXLP LDA TMPGRP-1,X
STA GROUP-1,X
DECC
BNE TXLP

```

```

0885
0886
0887
0888
0889
0890
0891
0892 e4a0 b6 86
0893 e4a2 b1 8f
0894 e4a4 26 06
0895 e4a6 b6 87
0896 e4a8 b1 90
0897 e4aa 27 10
0898 e4ac b6 86
0899 e4ae b7 8f
0900 e4b0 b6 87
0901 e4b2 b7 90
0902 e4b4 cd eb e6
0903 e4b7 cd e8 0a
0904 e4ba 18 c9
0905
0906
0907
0908
0909
0910
0911
0912
0913 e4bc b6 88
0914 e4be b7 95
0915 e4c0 05 95 04
0916 e4c3 16 ca
0917 e4c5 20 02
0918 e4c7 17 ca
0919 e4c9 b6 89
0920 e4cb 36 95
0921 e4cd 46
0922 e4ce 44
0923 e4cf 44
0924 e4d0 44
0925 e4d1 44
0926 e4d2 b7 8e
0927
0928
0929
0930
0931
0932
0933
0934
0935
0936
0937
0938
0939
0940
0941
0942
0943
0944
0945
0946
0947
0948 e4d4 b6 88
0949 e4d6 a4 f8
0950 e4d8 27 0a
0951 e4da a1 08
0952 e4dc 27 06
0953
0954 e4de a1 f8
0955 e4e0 27 10
0956 e4e2 20 57
0957
0958 e4e4 b6 89
0959 e4e6 a4 03
0960 e4e8 48
0961 e4e9 97
0962 e4ea b6 8c
0963 e4ec e7 c1
0964 e4ee b6 8d
0965 e4f0 e7 c2
0966
0967 e4f2 3f 79
0968 e4f4 08 89 04
0969 e4f7 15 ca
0970 e4f9 20 02
0971 e4fb 14 ca

```

```

*****
*
* Update PI code, initialise if changed. *
* All block 1s used, block 3s not used. *
*
*****
PROC LDA GROUP COMPARE PI WITH PREVIOUS
CMP PI
BNE DNDX
LDA GROUP+1
CMP PI+1
BEQ PTYL
DNDX LDA GROUP DIFFERENT, SAVE NEW PI
STA PI
LDA GROUP+1
STA PI+1
JSR CLREON CLEAR EON,
JSR CLTR TRANSIENTS
BSET 4,STAT2 AND INITIALISE DISPLAY DATA
*****
*
* Update PTY and TP. *
* All block 2s used, not block 4 (grp 15B). *
*
*****
PTYL LDA GROUP+2
STA ITMP1
BRCLR 2,ITMP1,TPL1 TP HIGH ?
BSET 3,STAT3 YES, FLAG HIGH
BRA TPL
TPL1 BCLR 3,STAT3 NO, FLAG LOW
TPL LDA GROUP+3
ROR ITMP1
RORA
LSRA
LSRA
LSRA
LSRA
STA PTY
*****
*
* Groups handled. *
*
* All PI, PTY & TP *
* 0 A & B TA, PS, DI & M/S *
* 1 A & B FIN *
* 2 A RT *
* 4 A CT *
* 14 A EON *
* 15 B TA, DI & M/S *
*
*****
*
* Process groups 0 & 15B (PS & TA). *
*
*****
LDA GROUP+2
AND #$F8
BEQ GRP0 GROUP 0A
CMP #$08 GROUP 0B
BEQ GRP0
TGRP15 CMP #$F8 GROUP 15B
BEQ TACK
BRA PROCL
GRP0 LDA GROUP+3 GROUP 0 - PS & TA
AND #$03
TAX
LDA GROUP+6
STA PSN,X
LDA GROUP+7
STA PSN+1,X
TACK CLR RDSTO RDS OK, RESET TIME-OUT
BRSET 4,GRP0+3,TAH TA HIGH ?
BCLR 2,STAT3 NO, TA FLAG LOW
BRA NTD
TAH BSET 2,STAT3 YES, TA FLAG HIGH

```

```

0973
0974
0975
0976
0977
0978
0979 e4fd b6 89
0980 e4ff a4 03
0981 e501 97
0982 e502 b6 89
0983 e504 a4 40
0984 e506 5d
0985 e507 26 07
0986 e509 11 b0
0987 e50b 4d
0988 e50c 27 02
0989 e50e 10 b0
0990 e510 a3 01
0991 e512 26 07
0992 e514 13 b0
0993 e516 4d
0994 e517 27 02
0995 e519 12 b0
0996 e51b a3 02
0997 e51d 26 07
0998 e51f 15 b0
0999 e521 4d
1000 e522 27 02
1001 e524 14 b0
1002 e526 a3 03
1003 e528 26 07
1004 e52a 17 b0
1005 e52c 4d
1006 e52d 27 02
1007 e52f 16 b0
1008
1009 e531 11 ca
1010 e533 07 89 02
1011 e536 10 ca
1012 e538 cc e6 18
1013
1014
1015
1016
1017
1018
1019
1020 e53b a1 10
1021 e53d 27 04
1022 e53f a1 18
1023 e541 26 0b
1024
1025 e543 b6 8c
1026 e545 b7 91
1027 e547 b6 8d
1028 e549 b7 92
1029
1030 e54b cc e6 18
1031
1032
1033
1034
1035
1036
1037
1038
1039 e54e a1 20
1040 e550 26 30
1041
1042 e552 08 89 07
1043 e555 02 ca 0c
1044 e558 12 ca
1045 e55a 20 05
1046 e55c 03 ca 05
1047 e55f 13 ca
1048 e561 cd eb aa
1049
1050 e564 b6 89
1051 e566 a4 0f
1052 e568 48
1053 e569 48
1054 e56a 97
1055 e56b b6 8a
1056 e56d d7 01 05
1057 e570 b6 8b
1058 e572 d7 01 06
1059 e575 b6 8c
1060 e577 d7 01 07
1061 e57a b6 8d
1062 e57c d7 01 08
1063 e57f cc e6 18

```

```

*****
*
* Process group 0 & 15B (DI & M/S).
*
*****
NTD LDA GROUP+3 DI
AND #3
TAX
LDA GROUP+3
AND #$40
TSTX
BNE NOT0
BCLR 0,DI
TSTA
BEQ NOT0
BSET 0,DI
NOT0 CPX #1
BNE NOT1
BCLR 1,DI
TSTA
BEQ NOT1
BSET 1,DI
NOT1 CPX #2
BNE NOT2
BCLR 2,DI
TSTA
BEQ NOT2
BSET 2,DI
NOT2 CPX #3
BNE NOT3
BCLR 3,DI
TSTA
BEQ NOT3
BSET 3,DI
NOT3 BCLR 0,STAT3 M/S
BRCLR 3,GROUP+3,MSZ
BSET 0,STAT3
MSZ JMP OUT1
*****
*
* Process group 1 (PIN).
*
*****
PROC1 CMP #$10 GROUP 1A
BEQ GRP1
CMP #$18 GROUP 1B
BNE PROC2
GRP1 LDA GROUP+6
STA PIN
LDA GROUP+7
STA PIN+1
JMP OUT1
*****
*
* Process group 2A (RT).
* Group 2B not handled.
*
*****
PROC2 CMP #$20 GROUP 2A
BNE PROC4
GRP2 BRSET 4,GROUP+3,TEXTB
TEXTA BRSET 1,STAT3,NCH
BSET 1,STAT3
BRA LCDINI
TEXTB BRCLR 1,STAT3,NCH
BCLR 1,STAT3
LCDINI JSR INITD
NCH LDA GROUP+3 GROUP 2A - RT
AND #$0F
LSLA
LSLA
TAX
LDA GROUP+4
STA RT+5,X
LDA GROUP+5
STA RT+6,X
LDA GROUP+6
STA RT+7,X
LDA GROUP+7
STA RT+8,X
JMP OUT1

```

```

1065
1066
1067
1068
1069
1070
1071 e582 a1 40
1072 e584 27 03
1073 e586 cc e6 1b
1074
1075 e589 b6 89
1076 e58b 46
1077 e58c a4 01
1078 e58e b7 74
1079 e590 b6 8a
1080 e592 46
1081 e593 b7 75
1082
1083 e595 b6 8c
1084 e597 36 8b
1085 e599 46
1086 e59a 44
1087 e59b 44
1088 e59c 44
1089 e59d b7 9c
1090
1091 e59f b6 8b
1092 e5a1 b7 76
1093
1094 e5a3 b6 8c
1095 e5a5 38 8d
1096 e5a7 49
1097 e5a8 38 8d
1098 e5aa 49
1099 e5ab a4 3f
1100 e5ad b7 9b
1101 e5af 3f 9a
1102 e5b1 3f 99
1103 e5b3 1c ca
1104
1105
1106
1107
1108
1109
1110
1111 e5b5 b6 8d
1112 e5b7 48
1113 e5b8 27 5e
1114 e5ba 24 32
1115
1116 e5bc 44
1117 e5bd 44
1118 e5be 44
1119 e5bf 44
1120 e5c0 97
1121 e5c1 24 0c
1122 e5c3 b6 9b
1123 e5c5 a0 1e
1124 e5c7 2a 04
1125 e5c9 ab 3c
1126 e5cb 3a 9c
1127 e5cd b7 9b
1128
1129 e5cf 9f
1130 e5d0 b0 9c
1131 e5d2 43
1132 e5d3 4c
1133 e5d4 2a 14
1134 e5d6 ab 18
1135 e5d8 b7 9c
1136
1137 e5da 3d 76
1138 e5dc 26 08
1139 e5de 3d 75
1140 e5e0 26 02
1141 e5e2 3a 74
1142 e5e4 3a 75
1143 e5e6 3a 76
1144 e5e8 20 2e
1145
1146 e5ea b7 9c
1147 e5ec 20 2a
1148
1149 e5ee 44
1150 e5ef 44
1151 e5f0 44
1152 e5f1 44
1153 e5f2 97
1154 e5f3 24 0e
1155 e5f5 a6 1e
1156 e5f7 bb 9b
1157 e5f9 a1 3b
1158 e5fb 23 04
1159 e5fd a0 3c
1160 e5ff 3c 9c
1161 e601 b7 9b
1162
1163 e603 9f
1164 e604 bb 9c
1165 e606 a1 17
1166 e608 23 0c
1167 e60a a0 18
1168 e60c 3c 76
1169 e60e 26 06
1170 e610 3c 75
1171 e612 26 02
1172 e614 3c 74
1173 e616 b7 9c
1174 e618 13 c9
1175 e61a 80

*****
*
* Process group 4A (CT).
*
*****

PROC4  CMP  #540      GROUP 4A - CT
        BEQ  GRP4
        JMP  PROC14

GRP4   LDA  GROUP+3
        RORA
        AND  #501
        STA  BMJD      MJD MS BIT
        LDA  GROUP+4
        RORA
        STA  BMJD+1    MJD MSD

        LDA  GROUP+6      GROUP 4
        ROR  GROUP+5      3210xxxx 4
        RORA              43210xxx  x
        LSR  LSR          -43210xx  x
        LSR  LSR          --43210x  x
        LSR  LSR          ---43210  x

        STA  OUR

        LDA  GROUP+5
        STA  BMJD+2      MJD LSD

        LDA  GROUP+6      xxxx5432  x
        LSL  GROUP+7      xxxxx5432  1
        ROLA              xxx54321  x
        LSL  GROUP+7      xxx54321  0
        ROLA              xx543210  x
        AND  #53F         --543210  x
        STA  MIN
        CLR  SEC
        CLR  TH8
        BSET 6,STAT3      UPDATE MJD

*****
*
* Local time difference adjustment.
*
*****

LOCAL  LDA  GROUP+7
        LSLA
        BEQ  OUT1      ADJUSTMENT ?
        BCC  POS       YES, POSITIVE ?

NEG    LSR  LSR          NO, NEGATIVE
        LSR  LSR
        LSR  LSR
        TAX
        BCC  NOTHN     HOURS IN X
        LDA  MIN        1/2 HOUR ?
        SUB  #30        YES
        BPL  LT60      SUBTRACT 30 MINUTES
        ADD  #60        UNDERFLOW ?
        DEC  OUR        YES, ADD 60 MINUTES
        STA  MIN        AND SUBTRACT 1 HOUR

LT60   STA  MIN

NOTHN  TXA
        SUB  OUR        NEGATIVE HOUR OFFSET
        COMA           MINUS UTC HOURS
        INCA          WRONG WAY ROUND SO COMPLEMENT
        BPL  ZOM       UNDERFLOW ?
        ADD  #24        YES, ADD 24 HOURS
        STA  OUR

        TST  BMJD+2    AND SUBTRACT A DAY
        BNE  TT2       LSB WILL UNDERFLOW ?
        TST  BMJD+1    YES
        BNE  TT1       MSB WILL UNDERFLOW ?
        DEC  BMJD      YES DECREMENT MS BIT
        DEC  BMJD+1    DECREMENT MSB
        BRA  OUT1      DECREMENT LSB

ZOM    STA  OUR
        BRA  OUT1

POS    LSR  LSR          POSITIVE ADJUSTMENT
        LSR  LSR
        LSR  LSR
        TAX
        BCC  NOTHP     HOURS IN X
        LDA  #30        HALF HOUR ?
        ADD  MIN        YES, ADD 30 MINUTES
        CMP  #59
        BLS  HDON      OVERFLOW ?
        SUB  #60        YES, SUBTRACT 60 MINUTES
        INC  OUR        AND ADD AN HOUR
        STA  MIN

HDON   STA  MIN

NOTHP  TXA
        ADD  OUR        HOUR OFFSET
        CMP  #23        ADD UTC HOURS
        BLS  ADDON     OVERFLOW ?
        SUB  #24        YES, SUBTRACT 24 HOURS
        INC  BMJD+2    AND ADD A DAY
        BNE  ADDON
        INC  BMJD+1
        BNE  ADDON
        INC  BMJD
        ADDON          ADDON

        STA  OUR
        OUT1          1,STAT2      GROUP HANDLED, CLEAR FLAG
        BCLR RTI

```

```

1177
1178
1179
1180
1181
1182
1183 e61b a1 e0
1184 e61d 27 03
1185 e61f cc e6 b3
1186
1187 e622 3f 95
1188 e624 be 95
1189 e626 d6 01 45
1190 e629 b1 8c
1191 e62b 26 69
1192 e62d d6 01 46
1193 e630 b1 8d
1194 e632 26 62
1195
1196
1197
1198
1199
1200
1201 e634 b6 89
1202 e636 a4 0f
1203 e638 a1 04
1204 e63a 24 10
1205 e63c 48
1206 e63d bb 95
1207 e63f 97
1208 e640 b6 8a
1209 e642 d7 01 47
1210 e645 b6 8b
1211 e647 d7 01 48
1212 e64a 20 cc
1213
1214 e64c a1 04
1215 e64e 26 34
1216
1217 e650 b6 8a
1218
1219 e652 a1 fa
1220 e654 26 1a
1221 e656 d6 01 51
1222 e659 a1 ff
1223 e65b 27 56
1224 e65d d6 01 53
1225 e660 a1 ff
1226 e662 26 4f
1227 e664 a6 fa
1228 e666 d7 01 53
1229 e669 b6 8b
1230 e66b d7 01 54
1231 e66e 20 43
1232
1233 e670 a1 e0
1234 e672 25 0e
1235 e674 a1 f9
1236 e676 22 0a
1237 e678 be 95
1238 e67a d7 01 51
1239 e67d b6 8b
1240 e67f d7 01 52
1241 e682 20 2f
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253 e684 a1 0e
1254 e686 26 2b
1255 e688 be 95
1256 e68a b6 8a
1257 e68c d7 01 4f
1258 e68f b6 8b
1259 e691 d7 01 50
1260 e694 20 1d
1261
1262 e696 a1 ff
1263 e698 26 0c
1264 e69a b6 8c
1265 e69c d7 01 45
1266 e69f b6 8d
1267 e6a1 d7 01 46
1268 e6a4 20 0d
1269
1270 e6a6 b6 95
1271 e6a8 ab 10
1272 e6aa b7 95
1273 e6ac a1 b0
1274 e6ae 27 03
1275 e6b0 cc e6 24
1276
1277 e6b3 13 c9
1278 e6b5 80

```

```

*****
*
* Process group 14 (EON).
*
*****
PROC14  CMP  #$E0
        BEQ  GRP14A
        JMP  OUT2
GRP14A  CLR  ITMP1          LOOK FOR PI CODE IN TABLE
LPIL    LDX  ITMP1
        LDA  EON,X
        CMP  GROUP+6
        BNE  NOTH
        LDA  EON+1,X
        CMP  GROUP+7
        BNE  NOTH
*
* LDA  GROUP+3          TP (ON), NOT USED
* AND  #$10
* LDX  ITMP1
* STA  EON+11,X
*
* LDA  GROUP+3          PI CODE FOUND
* AND  #$0F
* CMP  #4
* BHS  NPS
* LSLA
* ADD  ITMP1
* TAX
* LDA  GROUP+4
* STA  EON+2,X          SAVE 2 PS-NAME CHARACTERS
* LDA  GROUP+5
* STA  EON+3,X
* BRA  OUT1
*
* NPS  CMP  #4
* BNE  TRYPIN          AF ?
*
* LDA  GROUP+4          YES, METHOD A
*
* CMP  #250
* BNE  NMLW          MEDIUM OR LONG WAVE ?
* LDA  EON+12,X
* CMP  #$FF          YES
* BEQ  OUT2          FIRST 2 BYTES ALREADY IN ?
* LDA  EON+14,X          IF NOT, DO NOTHING
* CMP  #$FF          YES
* BNE  OUT2          M/L FREQUENCY ALREADY IN ?
* LDA  #250          IF SO, DO NOTHING
* STA  EON+14,X          NO, STORE FIRST FREQUENCY AFTER
* LDA  GROUP+5          ARRIVAL OF INITIAL BYTES
* STA  EON+15,X
* BRA  OUT2
*
* NMLW  CMP  #224          FM
* BLO  TOOLS          LEGAL ? (No. OF FREQUENCIES)
* CMP  #249
* BHI  TOOLS
* LDX  ITMP1
* STA  EON+12,X          YES, SAVE No. OF FREQUENCIES
* LDA  GROUP+5
* STA  EON+13,X
* BRA  OUT2          AND FIRST FREQUENCY
*
* TRYPIN  CMP  #$0D
* BNE  TRYPIN
* LDA  GROUP+4          PTY (EON), NOT USED
*
* LSRA
* LSRA
* LDX  ITMP1
* STA  EON+10,X
* BRA  OUT2
*
* TRYPIN  CMP  #$0E
* BNE  OUT2          PIN
* LDX  ITMP1
* LDA  GROUP+4
* STA  EON+10,X
* LDA  GROUP+5
* STA  EON+11,X
* BRA  OUT2
*
* NOTH  CMP  #$FF          END OF PI LIST ?
* BNE  NOTH1
* LDA  GROUP+6          YES, ADD THIS PI CODE
* STA  EON,X
* LDA  GROUP+7          TO EON TABLE
* STA  EON+1,X
* BRA  OUT2
*
* NOTH1  LDA  ITMP1          NOT END, TRY NEXT ENTRY
* ADD  #16
* STA  ITMP1
* CMP  #SB0
* BEQ  OUT2          END OF TABLE (11 ENTRIES) ?
* JMP  LPIL
*
* OUT2  BCLR 1,STAT2          GROUP HANDLED, CLEAR FLAG
        RTI

```

```

1280
1281
1282
1283
1284
1285
1286 e6b6 09 c9 05 MOD BRCLR 4,STAT2,NOCL SHOULD DISPALY BE INITIALISED ?
1287 e6b9 cd eb aa JSR INITD YES, DO IT
1288 e6bc 19 c9 BCLR 4,STAT2 AND CLEAR FLAG
1289 e6be cd eb 6c NOCL JSR WAIT
1290 e6c1 a6 0c LDA #80C SWITCH DISPLAY ON
1291 e6c3 cd eb 65 JSR CLOCK LATCH IT
1292 e6c6 cd eb 6c JSR WAIT
1293 * LDA #838 /16 DISPLAY
1294 e6c9 a6 30 LDA #830 /8 DISPLAY
1295 e6cb cd eb 65 JSR CLOCK LATCH IT
1296 e6ce cd eb 6c JSR WAIT
1297 e6d1 a6 80 LDA #880 ADDRESS DISPLAY RAM
1298 e6d3 cd eb 65 JSR CLOCK LATCH IT
1299
1300 e6d6 06 04 0b BRSET 3,PORTE,TRYRT STANDBY ?
1301 e6d9 04 cb 59 BRSET 2,STAT4,SLPD YES, SLEEP DISPLAY ?
1302 e6dc 06 cb 66 BRSET 3,STAT4,ALRMJ NO, ALARM DISPLAY ?
1303 e6df cd eb 69 JSR STBYD NO, NORMAL STANDBY DISPLAY
1304 e6e2 20 64 BRA ROW1
1305
1306 e6e4 0f cb 46 TRYRT BRCLR 7,STAT4,RTITS RDS DISPLAYS ?
1307 e6e7 b6 af LDA RTDIS
1308 e6e9 a1 01 CMP #1
1309 e6eb 26 05 BNE NPTY
1310 e6ed cd eb 19 JSR PTYD PTY
1311 e6f0 20 56 BRA ROW1
1312
1313 e6f2 a1 02 NPTY CMP #2
1314 e6f4 26 05 BNE NPI
1315 e6f6 cd eb fa JSR DIPI PI
1316 e6f9 20 4d BRA ROW1
1317
1318 e6fb a1 03 NPI CMP #3
1319 e6fd 26 05 BNE NTAP
1320 e6ff cd eb 5a JSR DITAP TA & TP
1321 e702 20 44 BRA ROW1
1322
1323 e704 a1 04 NTAP CMP #4
1324 e706 26 05 BNE NPIN1
1325 e708 cd eb 72 JSR DPIN1 PIN - HEX
1326 e70b 20 3b BRA ROW1
1327
1328 e70d a1 05 NPIN1 CMP #5
1329 e70f 26 05 BNE NPIN2
1330 e711 cd eb 92 JSR DFIN2 PIN - DAY AND TIME
1331 e714 20 32 BRA ROW1
1332
1333 e716 a1 06 NPIN2 CMP #6
1334 e718 26 05 BNE NMJD
1335 e71a cd eb f6 JSR DMJD MJD
1336 e71d 20 29 BRA ROW1
1337
1338 e71f a1 07 NMJD CMP #7
1339 e721 26 05 BNE NMSD
1340 e723 cd eb 03 JSR DMSD M/S & DI
1341 e726 20 20 BRA ROW1
1342
1343 e728 cd ea 25 NMSD JSR DEON
1344 e72b 20 1b BRA ROW1
1345
1346 e72d 05 c9 05 RTITS BRCLR 2,STAT2,SLPD RT DISPLAY ?
1347 e730 cd eb 4a JSR RTDS
1348 e733 20 13 BRA ROW1
1349
1350 e735 05 cb 05 SLPD BRCLR 2,STAT4,NRMD SLEEP TIMER DISPLAY ?
1351 e738 cd ea ee JSR SLEEPD
1352 e73b 20 0b BRA ROW1
1353
1354 e73d 06 cb 05 NRMD BRSET 3,STAT4,ALRMJ ALARM DISPLAY ?
1355 e740 cd eb 7c JSR NORMD
1356 e743 20 03 BRA ROW1
1357
1358 e745 cd eb 1a ALRMJ JSR ALRMD
1359
1360 e748 5f ROW1 CLRX
1361 e749 cd eb 6c LCD JSR WAIT
1362 e74c 14 03 BSET 2,PORTD WRITE DATA
1363 e74e e6 b1 LDA DISP,X GET A BYTE
1364 e750 a1 ff CMP #80C
1365 e752 26 02 BNE COK
1366 e754 a6 2d LDA #82D
1367 e756 cd eb 65 COK JSR CLOCK SEND IT TO MODULE
1368 e759 5c INCX
1369 e75a a3 10 CPX #16 DONE ?
1370 e75c 26 eb BNE LCD
1371 e75e 20 1e BRA VFD REMOVE FOR /16 LCDs
1372
1373
1374
1375
1376
1377
1378
1379 e760 cd eb 6c LCD401 JSR WAIT
1380 e763 a6 a8 LDA #8A8 TO 40
1381 e765 cd eb 65 JSR CLOCK SEND IT TO MODULE
1382 e768 5f LCD41 CLRX
1383 e769 cd eb 6c JSR WAIT
1384 e76c 14 03 BSET 2,PORTD WRITE DATA
1385 e76e e6 b9 LDA DISP+8,X GET A BYTE
1386 e770 a1 ff CMP #80C
1387 e772 26 02 BNE COK2
1388 e774 a6 2d LDA #82D
1389 e776 cd eb 65 COK2 JSR CLOCK SEND IT TO MODULE
1390 e779 5c INCX
1391 e77a a3 08 CPX #8 DONE ?
1392 e77c 26 eb BNE LCD41

```

```

1394
1395
1396
1397
1398
1399
1400 e77e 13 01 VFD BCLR 1,PORTB DATA LOW ?
1401 e780 10 01 BSET 0,PORTB CLOCK HIGH ?
1402 e782 17 01 BCLR 3,PORTB ENABLE LOW
1403
1404 e784 5f CLRX SEND VFD SET-UP BYTES
1405 e785 d6 e7 c5 DIS5 LDA INITF,X
1406 e788 bf a7 STX W7 SAVE INDEX
1407 e78a ad 20 BSR VFDD
1408 e78c a3 07 CPX #7
1409 e78e 26 f5 BNE DIS5 LAST BYTE ?
1410
1411 e790 5f CLRX SEND 16 CHARACTER BYTES
1412 e791 bf a7 VFD3 STX W7 SAVE INDEX
1413 e793 e6 b1 LDA DISP,X ASCII
1414 e795 a1 ff CMP #5FH
1415 e797 26 02 BNE NOTFF
1416 e799 a6 2d LDA #52D REPLACE $FF WITH "-"
1417 e79b a4 7f NOTFF AND #57F IGNORE BIT 7
1418 e79d 97 TAX
1419 e79e d6 ed ae LDA VTAB,X CONVERT TO VFD CHARACTER SET
1420 e7a1 ad 09 BSR VFDD
1421 e7a3 a3 10 CPX #16
1422 e7a5 26 ea BNE VFD3 LAST BYTE ?
1423
1424 e7a7 16 01 BSET 3,PORTB ENABLE HIGH
1425 e7a9 11 01 BCLR 0,PORTB CLOCK LOW ?
1426 e7ab 81 RTS
1427
1428 e7ac ae 08 VFDL LDX #8
1429 e7ae 44 02 DIS3 LSRA
1430 e7af 24 02 BCC DIS4 GET A BIT
1431 e7b1 12 01 BSET 1,PORTB DATA HIGH
1432 e7b3 11 01 DIS4 BCLR 0,PORTB CLOCK
1433 e7b5 10 01 BSET 0,PORTB IT
1434 e7b7 13 01 BCLR 1,PORTB CLEAR DATA
1435 e7b9 5a DECX COMPLETE ?
1436 e7ba 26 f2 BNE DIS3 NO
1437 e7bc ae 40 LDX #64
1438 e7be 5a DEL DECX WAIT 200uS
1439 e7bf 26 fd BNE DEL
1440 e7c1 be a7 LDX W7 RESTORE INDEX
1441 e7c3 5c INCX
1442 e7c4 81 RTS
1443
1444 e7c5 a0 of b0 00 80 00 INITF FCB $A0,$0F,$B0,$00,$80,$00,$90
1445
1446
1447
1448
1449
1450
1451
1452 e7cc a6 20 NORMD LDA #520
1453 e7ce b7 b1 STA DISP
1454 e7d0 b7 ba STA DISP+9
1455 e7d2 b7 c0 STA DISP+15
1456 e7d4 a6 2e LDA #52E
1457 e7d6 03 cb 05 BRCLR 1,STAT4,TYP1 DP TO INDICATE SLEEP TIMER RUNNING
1458 e7d9 05 99 02 BRCLR 2,TH8,TYP1 FLASH IT
1459 e7dc b7 c0 STA DISP+15
1460
1461 e7de 5f TYP1 CLRX
1462 e7df e6 c1 MPS LDA PSN,X GET PS NAME
1463 e7e1 e7 b2 STA DISP+1,X
1464 e7e3 5c SCNG INCX
1465 e7e4 a3 07 CPX #7
1466 e7e6 23 f7 BLS MPS
1467
1468 e7e8 b6 9c CJ LDA OUR GET TIME
1469 e7ea cd eb 84 JSR CBCD
1470 e7ed a3 30 CPX #530 LEADING ZERO ?
1471 e7ef 26 02 BNE TNZ
1472 e7f1 ae 20 LDX #520 YES, MAKE IT A SPACE
1473 e7f3 bf bb STX DISP+10
1474 e7f5 b7 bc STA DISP+11
1475 e7f7 b6 9b CMIN LDA MIN
1476 e7f9 cd eb 84 JSR CBCD
1477 e7fc bf be STX DISP+13
1478 e7fe b7 bf STA DISP+14
1479 e800 a6 20 CSEC LDA #520
1480 e802 05 99 02 BRCLR 2,TH8,DDC
1481 e805 a6 3a LDA #53A 0.5 Hz FLASHING COLON
1482 e807 b7 bd DDC STA DISP+12
1483 e809 81 RTS
1484
1485
1486
1487
1488
1489
1490
1491 e80a 11 cb CLTR BCLR 0,STAT4 CLEAR DISPLAY TRANSIENT FLAG
1492 e80c 15 c9 BCLR 2,STAT2 NOT RT DISPLAY
1493 e80e 3f af CLR RTDIS CLEAR RDS DISPLAY INDEX
1494 e810 17 cb BCLR 3,STAT4 NOT ALARM DISPLAY
1495 e812 1b cb BCLR 5,STAT4 NOT ALARM SET-UP
1496 e814 1f cb BCLR 7,STAT4 NOT RDS DISPLAYS
1497 e816 15 cb BCLR 2,STAT4 NOT SLEEP TIMER DISPLAY
1498 e818 81 RTS

```

```

1500
1501
1502
1503
1504
1505
1506 e819 be 8e
1507 e81b a3 10
1508 e81d 25 01
1509 e81f 5f
1510 e820 a6 10
1511 e822 42
1512 e823 b7 a8
1513 e825 3f a7
1514 e827 be a8
1515 e829 d6 ec ae
1516 e82c be a7
1517 e82e e7 b1
1518 e830 3c a8
1519 e832 3c a7
1520 e834 b6 a7
1521 e836 a1 10
1522 e838 25 ed
1523 e83a 81
1524
1525
1526
1527
1528
1529
1530
1531 e83b be a0
1532 e83d e6 ff
1533 e83f a1 20
1534 e841 26 13
1535 e843 0b c9
1536 e846 3c 9f
1537 e848 3c a0
1538 e84a b6 a0
1539 e84c a1 45
1540 e84e 22 18
1541 e850 20 e9
1542
1543 e852 1a c9
1544 e854 20 02
1545 e856 1b c9
1546 e858 b7 a8
1547 e85a 5f
1548 e85b e6 b2
1549 e85d e7 b1
1550 e85f 5c
1551 e860 a3 0f
1552 e862 26 f7
1553 e864 b6 a8
1554 e866 b7 c0
1555 e868 81
1556
1557
1558
1559
1560
1561
1562
1563
1564 e869 08 cb 4f
1565 e86c b6 73
1566 e86e 48
1567 e86f bb 73
1568 e871 97
1569 e872 d6 ec 72
1570 e875 b7 b1
1571 e877 d6 ec 73
1572 e87a b7 b2
1573 e87c d6 ec 74
1574 e87f b7 b3
1575 e881 a6 20
1576 e883 b7 b4
1577 e885 b7 b7
1578 e887 b7 bb
1579 e889 b6 72
1580 e88b ab 30
1581 e88d b7 b6
1582 e88f b6 71
1583 e891 27 02
1584 e893 ab 10
1585 e895 ab 20
1586 e897 b7 b5
1587 e899 be 70
1588 e89b b6 6f
1589 e89d 27 04
1590 e89f 9f
1591 e8a0 ab 0a
1592 e8a2 97
1593 e8a3 bf a8
1594 e8a5 9f
1595 e8a6 48
1596 e8a7 bb a8
1597 e8a9 97
1598 e8aa d6 ec 87
1599 e8ad b7 b8
1600 e8af d6 ec 88
1601 e8b2 b7 b9
1602 e8b4 d6 ec 89
1603 e8b7 b7 ba
1604 e8b9 20 1d

```

```

*****
*
*   PTY display.
*
*****
PTYD  LDX   PTY
      CPX   #16
      BLO   XOK2
      CLRX
XOK2  LDA   #16
      MUL
      STA   W8
      CLR   W7
LCD3  LDX   W8
      LDA   PTYT,X
      LDX   W7
      STA   DISP,X      WAS MOD2
      INC   W8
      INC   W7
      LDA   W7
      CMP   #16
      BLO   LCD3
      RTS
*****
*
*   RDS display.
*
*****
NXTC  LDX   DISP2
      LDA   RT-1,X      RT
      CMP   #S20
      BNE   NOTSP      SPACE ?
      BRCLR 5,STAT2,FSP YES, FIRST ONE ?
      INC   DISP1      NO, SKIP THIS ONE
      INC   DISP2
RTDS  LDA   DISP2
SKP1  CMP   #69
      BHI   LCD4      END OF RT BUFFER
      BRA   NXTC      NO, GET NEXT CHARACTER
*****
FSP   BSET  5,STAT2    FIRST SPACE, SET FLAG
      BRA   CONT
NOTSP BCLR  5,STAT2    NOT A SPACE, CLEAR FLAG
CONT  STA   W8         SAVE NEW CHARACTER
      CLRX
ILP1  LDA   DISP+1,X   MOVE
      STA   DISP,X     REST
      INCX
      CPX   #15       LEFT
      BNE   ILP1      ONE
      LDA   W8         PLACE
      STA   DISP+15   ADD NEW CHAR. (WAS MOD2)
LCD4  RTS
*****
*
*   Standby display.
*
*****
STBYD BRSET  4,STAT4,ALRMA ALARM ARMED ?
      LDA   DOW        NO, GET DAY OF WEEK
      LSLA
      ADD   DOW
      TAX
      LDA   DNAME,X
      STA   DISP
      LDA   DNAME+1,X
      STA   DISP+1
      LDA   DNAME+2,X
      STA   DISP+2
      LDA   #S20
      STA   DISP+3
      STA   DISP+6
      STA   DISP+10
      LDA   DOM+1      DATE
      ADD   #S30
      STA   DISP+5
      LDA   DOM
      BEQ   ADD20      IF ZERO USE A SPACE
      ADD   #S10      IF NOT MAKE ASCII
      ADD   #S20
      STA   DISP+4
      LDX   MNTH+1    MONTH, LSD
      LDA   MNTH      MONTH, MSD
      BEQ   MTHZ
      TXA
      ADD   #10
      TAX
MTHZ  STX   W8
      TXA
      LSLA
      ADD   W8
      TAX
      LDA   MNAME-3,X
      STA   DISP+7
      LDA   MNAME-2,X
      STA   DISP+8
      LDA   MNAME-1,X
      STA   DISP+9
      BRA   STIME

```

```

1606
1607
1608
1609
1610
1611
1612 e8bb b6 9e
1613 e8bd cd eb 84
1614 e8c0 bf b1
1615 e8c2 b7 b2
1616 e8c4 b6 9d
1617 e8c6 cd eb 84
1618 e8c9 bf b3
1619 e8cb b7 b4
1620 e8cd 5f
1621 e8ce d6 eb f3
1622 e8d1 e7 b5
1623 e8d3 5c
1624 e8d4 a3 06
1625 e8d6 23 f6
1626 e8d8 b6 9c
1627 e8da cd eb 84
1628 e8dd a3 30
1629 e8df 26 02
1630 e8e1 ae 20
1631 e8e3 bf bc
1632 e8e5 b7 bd
1633 e8e7 b6 9b
1634 e8e9 cd eb 84
1635 e8ec bf bf
1636 e8ee b7 c0
1637 e8f0 a6 20
1638 e8f2 05 99 02
1639 e8f5 a6 3a
1640 e8f7 b7 be
1641 e8f9 81
1642
1643
1644
1645
1646
1647
1648
1649 e8fa 5f
1650 e8fb d6 ec 02
1651 e8fe e7 b1
1652 e900 5c
1653 e901 a3 0f
1654 e903 23 f6
1655 e905 b6 8f
1656 e907 27 10
1657 e909 cd eb 48
1658 e90c bf bc
1659 e90e b7 bd
1660 e910 b6 90
1661 e912 cd eb 48
1662 e915 bf be
1663 e917 b7 bf
1664 e919 81
1665
1666
1667
1668
1669
1670
1671
1672 e91a 5f
1673 e91b d6 eb f2
1674 e91e e7 b1
1675 e920 5c
1676 e921 a3 0f
1677 e923 23 f6
1678 e925 09 cb 31
1679 e928 a6 3a
1680 e92a b7 bd
1681 e92c b6 9e
1682 e92e cd eb 84
1683 e931 a3 30
1684 e933 26 02
1685 e935 ae 20
1686 e937 bf bb
1687 e939 b7 bc
1688 e93b b6 9d
1689 e93d cd eb 84
1690 e940 bf be
1691 e942 b7 bf
1692 e944 0b cb 12
1693 e947 05 99 0f
1694 e94a a6 20
1695 e94c 0c cb 06
1696 e94f b7 be
1697 e951 b7 bf
1698 e953 20 04
1699 e955 b7 bb
1700 e957 b7 bc
1701 e959 81
1702
1703
1704
1705
1706
1707
1708
1709 e95a 5f
1710 e95b d6 ec 12
1711 e95e e7 b1
1712 e960 5c
1713 e961 a3 0f
1714 e963 23 f6
1715 e965 a6 31
1716 e967 07 ca 02
1717 e96a b7 b7
1718 e96c 05 ca 02
1719 e96f b7 bf
1720 e971 81

```

```

*****
*
* Standby (alarm armed) display.
*
*****
ALRMA LDA AOUR GET ALARM HOURS
JSR CBCD
STX DISP
STA DISP+1
LDA AMIN
JSR CBCD
STX DISP+2
STA DISP+3
CLRXL
ALOP2 LDA ALARMS+1,X
STA DISP+4,X
INCLX
CPX #6
BLS ALOP2
LDA OUR GET TIME
JSR CBCD
CPX #S30 LEADING ZERO ?
BNE TMZ
LDX #S20 YES, MAKE IT A SPACE
TMZ STX DISP+11
STA DISP+12
LDA MIN
JSR CBCD
STX DISP+14
STA DISP+15
LDA #S20
BRCLRL 2,TH8,DTF FLASH ?
LDA #S3A 0.5 Hz FLASHING COLON
DTF STA DISP+13
RTS
*****
*
* PI display.
*
*****
DIPI CLRXL
DLOP LDA PIST,X
STA DISP,X
INCLX
CPX #15
BLS DLOP
LDA PI
BEQ PINV
JSR SPLIT
STX DISP+11
STA DISP+12
LDA PI+1
JSR SPLIT
STX DISP+13
STA DISP+14
PINV RTS
*****
*
* Alarm display.
*
*****
ALRMD CLRXL YES
ALOP LDA ALARMS,X
STA DISP,X
INCLX
CPX #15
BLS ALOP
BRCLRL 4,STAT4,ALOP2 ALARM ARMED ?
LDA #S3A YES
STA DISP+12
LDA AOUR GET ALARM HOURS
JSR CBCD
CPX #S30 LEADING ZERO ?
BNE TN3
LDX #S20 YES, MAKE IT A SPACE
TN3 STX DISP+10
STA DISP+11
LDA AMIN
JSR CBCD
STX DISP+13
STA DISP+14
BRCLRL 5,STAT4,ALOP2 SET-UP ?
BRCLRL 2,TH8,ALOP2
LDA #S20
BRSET 6,STAT4,FH HOURS ?
STA DISP+13 NO, FLASH MINUTES
STA DISP+14
BRA ALOP2
FH STA DISP+10 YES, FLASH HOURS
STA DISP+11
ALOP2 RTS
*****
*
* TA & TP flags display.
*
*****
DITAP CLRXL
BLOP LDA TAPST,X
STA DISP,X
INCLX
CPX #15
BLS BLOP
LDA #S31
BRCLRL 3,STAT3,TPLOW TP FLAG HIGH ?
STA DISP+6 YES, DISPLAY A 1
BRCLRL 2,STAT3,TALOW TA FLAG HIGH ?
STA DISP+14 YES, DISPLAY A 1
TALOW RTS

```

```

1722
1723
1724
1725
1726
1727
1728 e972 5f
1729 e973 d6 ec 22
1730 e976 e7 bl
1731 e978 5c
1732 e979 a3 0f
1733 e97b 23 f6
1734 e97d b6 91
1735 e97f 27 10
1736 e981 cd eb 48
1737 e984 bf bc
1738 e986 b7 bd
1739 e988 b6 92
1740 e98a cd eb 48
1741 e98d bf be
1742 e98f b7 bf
1743 e991 81
1744
1745 e992 5f
1746 e993 d6 ec 32
1747 e996 e7 bl
1748 e998 5c
1749 e999 a3 0f
1750 e99b 23 f6
1751 e99d b6 91
1752 e99f 27 f0
1753 e9a1 44
1754 e9a2 44
1755 e9a3 44
1756 e9a4 cd eb 84
1757 e9a7 a3 30
1758 e9a9 26 02
1759 e9ab ae 20
1760 e9ad bf b3
1761 e9af b7 b4
1762 e9b1 a3 31
1763 e9b3 27 24
1764 e9b5 a1 31
1765 e9b7 26 08
1766 e9b9 a6 73
1767 e9bb b7 b5
1768 e9bd a6 74
1769 e9bf b7 b6
1770 e9c1 a1 32
1771 e9c3 26 08
1772 e9c5 a6 6e
1773 e9c7 b7 b5
1774 e9c9 a6 64
1775 e9cb b7 b6
1776 e9cd a1 33
1777 e9cf 26 08
1778 e9d1 a6 72
1779 e9d3 b7 b5
1780 e9d5 a6 64
1781 e9d7 b7 b6
1782 e9d9 b6 91
1783 e9db a4 07
1784 e9dd be 92
1785 e9df 58
1786 e9e0 49
1787 e9e1 58
1788 e9e2 49
1789 e9e3 cd eb 84
1790 e9e6 bf bb
1791 e9e8 b7 bc
1792 e9ea b6 92
1793 e9ec a4 3f
1794 e9ee cd eb 84
1795 e9f1 bf be
1796 e9f3 b7 bf
1797 e9f5 81
1798
1799
1800
1801
1802
1803
1804
1805 e9f6 ad 21
1806 e9f8 b6 5d
1807 e9fa 27 1c
1808 e9fc ab 30
1809 e9fe b7 bb
1810 ea00 b6 5e
1811 ea02 ab 30
1812 ea04 b7 bc
1813 ea06 b6 5f
1814 ea08 ab 30
1815 ea0a b7 bd
1816 ea0c b6 60
1817 ea0e ab 30
1818 ea10 b7 be
1819 ea12 b6 61
1820 ea14 ab 30
1821 ea16 b7 bf
1822 ea18 81
1823
1824 ea19 5f
1825 ea1a d6 ec 42
1826 ea1d e7 bl
1827 ea1f 5c
1828 ea20 a3 0f
1829 ea22 23 f6
1830 ea24 81

*****
*
* PIN displays.
*
*****

DPIN1 CLRX
PLOP LDA PINST1,X
STA DISP,X
INCX
CPX #15
BLS PLOP
LDA PIN
BEQ PINNV
JSR SPLIT
STX DISP+11
STA DISP+12
LDA PIN+1
JSR SPLIT
STX DISP+13
STA DISP+14
PINNV RTS

DPIN2 CLRX
PLOP2 LDA PINST2,X
STA DISP,X
INCX
CPX #15
BLS PLOP2
LDA PIN
BEQ PINNV
LSRA
LSRA
LSRA
JSR CBCD
CPX #30
BNE DTNO
LDX #320
DTNO STX DISP+2
STA DISP+3
CPX #331
BEQ NOTRD
CMP #331
BNE NOTST
LDA #'s'
STA DISP+4
LDA #'t'
STA DISP+5
NOTST CMP #332
BNE NOTND
LDA #'n'
STA DISP+4
LDA #'d'
STA DISP+5
NOTND CMP #333
BNE NOTRD
LDA #'r'
STA DISP+4
LDA #'d'
STA DISP+5
NOTRD LDA PIN
AND #7
LDX PIN+1
ASLX
ROLA
ASLX
ROLA
JSR CBCD
STX DISP+10
STA DISP+11
LDA PIN+1
AND #33F
MINUTES JSR CBCD
STX DISP+13
STA DISP+14
RTS

*****
*
* MJD display.
*
*****

DMJD BSR SMJD
LDA MJD
BEQ MJDNV
ADD #30
STA DISP+10
LDA MJD+1
ADD #30
STA DISP+11
LDA MJD+2
ADD #30
STA DISP+12
LDA MJD+3
ADD #30
STA DISP+13
LDA MJD+4
ADD #30
STA DISP+14
MJDNV RTS

SMJD CLRX
MLOP LDA MJDST,X
STA DISP,X
INCX
CPX #15
BLS MLOP
RTS

```

```

1832
1833
1834
1835
1836
1837
1838 ea25 cd ea 19
1839 ea28 b6 af
1840 ea2a a0 08
1841 ea2c ae 10
1842 ea2e 42
1843 ea2f 97
1844 ea30 a6 20
1845 ea32 b7 b9
1846 ea34 b7 ba
1847 ea36 d6 01 47
1848 ea39 b7 b1
1849 ea3b d6 01 48
1850 ea3e b7 b2
1851 ea40 d6 01 49
1852 ea43 b7 b3
1853 ea45 d6 01 4a
1854 ea48 b7 b4
1855 ea4a d6 01 4b
1856 ea4d b7 b5
1857 ea4f d6 01 4c
1858 ea52 b7 b6
1859 ea54 d6 01 4d
1860 ea57 b7 b7
1861 ea59 d6 01 4e
1862 ea5c b7 b8
1863
1864 ea5e d6 01 52
1865 ea61 a1 cd
1866 ea63 26 04
1867 ea65 5c
1868 ea66 d6 01 52
1869 ea69 a1 fa
1870 ea6b 27 43
1871 ea6d a1 cc
1872 ea6f 22 3e
1873 ea71 ae 0a
1874 ea73 42
1875 ea74 ab 2e
1876 ea76 b7 a1
1877 ea78 9f
1878 ea79 a9 22
1879 ea7b b7 a2
1880 ea7d cd eb 1f
1881
1882 ea80 b6 34
1883 ea82 26 02
1884 ea84 a6 f0
1885 ea86 ab 30
1886 ea88 b7 bb
1887 ea8a 97
1888 ea8b b6 35
1889 ea8d 26 06
1890 ea8f a3 20
1891 ea91 26 02
1892 ea93 a6 f0
1893 ea95 ab 30
1894 ea97 b7 bc
1895 ea99 b6 36
1896 ea9b ab 30
1897 ea9d b7 bd
1898 ea9f a6 2e
1899 eaa1 b7 be
1900 eaa3 b6 37
1901 eaa5 ab 30
1902 eaa7 b7 bf
1903 eaa9 b6 38
1904 eaab ab 30
1905 eaad b7 c0
1906 eaaf 81
1907
1908 eab0 5c
1909 eab1 d6 01 52
1910 eab4 a1 0f
1911 eab6 23 02
1912 eab8 ab 1b
1913 eaba ab 10
1914 eabc ae 09
1915 eabe 42
1916 eabf bf a2
1917 eac1 b7 a1
1918 eac3 ad 5a
1919 eac5 b6 35
1920 eac7 26 02
1921 eac9 a6 f0
1922 eacb ab 30
1923 eacd b7 ba
1924 eacf b6 36
1925 ead1 ab 30
1926 ead3 b7 bb
1927 ead5 b6 37
1928 ead7 ab 30
1929 ead9 b7 bc
1930 eadb b6 38
1931 eadd ab 30
1932 eadf b7 bd
1933 eae1 a6 6b
1934 eae3 b7 be
1935 eae5 a6 48
1936 eae7 b7 bf
1937 eae9 a6 7a
1938 eaeb b7 c0
1939 eaed 81

*****
*                                     *
*      EON display.                   *
*                                     *
*****

DEON   JSR   SMJD   CLEAR FREQUENCY CHARACTERS
        LDA   RTDIS
        SUB   #8
        LDX   #16
        MUL
        TAX
        LDA   #S20
        STA   DISP+8
        STA   DISP+9
        LDA   EON+2,X   DISPLAY PS (EON)
        STA   DISP
        LDA   EON+3,X
        STA   DISP+1
        LDA   EON+4,X
        STA   DISP+2
        LDA   EON+5,X
        STA   DISP+3
        LDA   EON+6,X
        STA   DISP+4
        LDA   EON+7,X
        STA   DISP+5
        LDA   EON+8,X
        STA   DISP+6
        LDA   EON+9,X
        STA   DISP+7
        LDA   EON+13,X
        CMP   #205   FILLER ?
        BNE   NFIL
        INCX
        LDA   EON+13,X
        NFIL  CMP   #250   YES, TRY AGAIN
        BEQ   MLWF     MEDIUM/LONG ?
        CMP   #204     NO, FREQUENCY OK ?
        BHI   FNOK2
        FOK2  LDX   #10   VHF
        MUL
        ADD   #S2E     CALCULATE FREQUENCY (BINARY)
        STA   W1
        TXA
        ADC   #S22
        STA   W2
        JSR   DCON2    CONVERT TO DECIMAL
        LDA   Q+4     DISPLAY VHF EON FREQUENCY
        BNE   NZ1
        LDA   #SF0
        NZ1  ADD   #S30
        STA   DISP+10
        TAX
        LDA   Q+5
        BNE   NZ2
        CPX   #S20
        BNE   NZ2
        LDA   #SF0
        NZ2  ADD   #S30
        STA   DISP+11
        LDA   Q+6
        ADD   #S30
        STA   DISP+12
        LDA   #S2E
        STA   DISP+13
        LDA   Q+7
        ADD   #S30
        STA   DISP+14
        LDA   Q+8
        ADD   #S30
        STA   DISP+15
        FNOK2 RTS
        MLWF  INCX     DISPLAY M/L EON FREQUENCY
        LDA   EON+13,X
        CMP   #15
        BLS   LONG
        ADD   #27     MW OFFSET
        LONG  ADD   #16     M/L OFFSET
        LDX   #9
        MUL
        STX   W2
        STA   W1
        BSR   DCON2    CONVERT TO BCD IN Q
        LDA   Q+5
        BNE   NZ3
        LDA   #ZF0
        NZ3  ADD   #S30
        STA   DISP+9
        LDA   Q+6
        ADD   #S30
        STA   DISP+10
        LDA   Q+7
        ADD   #S30
        STA   DISP+11
        LDA   Q+8
        ADD   #S30
        STA   DISP+12
        LDA   #'k'
        STA   DISP+13
        LDA   #'H'
        STA   DISP+14
        LDA   #'z'
        STA   DISP+15
        RTS

```

```

1941
1942
1943
1944
1945
1946
1947 eaaa 5f
1948 eaef d6 ec 52
1949 eaf2 e7 bl
1950 eaf4 5c
1951 eaf5 a3 0f
1952 eaf7 23 f6
1953 eaf9 b6 78
1954 eafb cd eb 84
1955 eafe bf b9
1956 eb00 b7 ba
1957 eb02 81
1958
1959
1960
1961
1962
1963
1964
1965 eb03 5f
1966 eb04 d6 ec 62
1967 eb07 e7 bl
1968 eb09 5c
1969 eb0a a3 0f
1970 eb0c 23 f6
1971 eb0e 01 ca 04
1972 eb11 a6 4d
1973 eb13 b7 b7
1974 eb15 b6 b0
1975 eb17 cd eb 84
1976 eb1a bf be
1977 eb1c b7 bf
1978 eb1e 81
1979
1980
1981
1982
1983
1984
1985
1986 eb1f ae 54
1987 eb21 bf ad
1988 eb23 cd ef 86
1989 eb26 3c 5c
1990 eb28 cd ef 84
1991 eb2b a6 0e
1992 eb2d b7 a6
1993 eb2f 34 a2
1994 eb31 36 a1
1995 eb33 24 07
1996 eb35 ae 30
1997 eb37 bf ae
1998 eb39 cd ee 33
1999 eb3c ae 54
2000 eb3e bf ae
2001 eb40 cd ee 33
2002 eb43 3a a6
2003 eb45 26 e8
2004 eb47 81
2005
2006
2007
2008
2009
2010
2011
2012
2013 eb48 97
2014 eb49 99
2015 eb4a 56
2016 eb4b 99
2017 eb4c 56
2018 eb4d 54
2019 eb4e 54
2020 eb4f a3 39
2021 eb51 23 07
2022 eb53 5c
2023 eb54 5c
2024 eb55 5c
2025 eb56 5c
2026 eb57 5c
2027 eb58 5c
2028 eb59 5c
2029 eb5a a4 0f
2030 eb5c ab 30
2031 eb5e a1 39
2032 eb60 23 02
2033 eb62 ab 07
2034 eb64 81

```

```

*****
*
* Sleep display.
*
*****
SLEPPD CLRX
SLOP LDA SLPST,X
STA DISP,X
INCX
CPX #15
BLS SLOP
LDA SLEPT
JSR CBCD
STX DISP+8
STA DISP+9
RTS
*****
*
* M/S & DI display.
*
*****
DMSD CLRX
ILOP LDA MSDST,X
STA DISP,X
INCX
CPX #15
BLS ILOP
BRCLR 0,STAT3,MSM2 M/S FLAG SET
LDA #'M' YES, MUSIC
STA DISP+6
MSM2 LDA DI
JSR CBCD
STX DISP+13
STA DISP+14
RTS
*****
*
* Convert binary to unpacked BCD in Q.
*
*****
DCON2 LDX #R CLEAR
STX NUM1
JSR CLRAS RR
INC R+8 R <- 1
JSR CLQ CLEAR RQ
LDA #14 14 BITS TO CONVERT
STA W6
LOOP2 LSR W2 MOVE OUT
ROR W1 FIRST (LS) BIT
BCC NXT ZERO
LDX #Q ONE, ADD
STX NUM2 CURRENT VALUE
JSR ADD OF R
NXT LDX #R ADD R
STX NUM2 TO
JSR ADD ITSELF
DEC W6 ALL
BNE LOOP2 DONE ?
RTS
*****
*
* Split A nibbles into A (LS) and X (MS)
* and convert to ASCII.
*
*****
SPLIT TAX MSD INTO X, LSD INTO A
SEC
RORX
SEC
RORX
LSRX
LSRX
CPX #39 $30-$39 <- 0-9
BLS XOK
INCX
INCX
INCX
INCX
INCX
INCX
XOK AND #30F $41-$46 <- A-F
ADD #30
CMP #39
BLS AOK
ADD #7
AOK RTS

```

```

2036
2037
2038
2039
2040
2041
2042
2043
2044 eb65 b7 02
2045 eb67 18 03
2046 eb69 19 03
2047 eb6b 81
2048
2049 eb6c 15 03
2050 eb6e 16 03
2051 eb70 19 03
2052 eb72 3f 07
2053 eb74 18 03
2054 eb76 b6 02
2055 eb78 19 03
2056 eb7a b7 a7
2057 eb7c 0e a7 f5
2058 eb7f 33 07
2059 eb81 17 03
2060 eb83 81
2061
2062
2063
2064
2065
2066
2067
2068 eb84 ad 1c
2069 eb86 ad 13
2070 eb88 b7 a7
2071 eb8a ab 16
2072 eb8c ad 08
2073 eb8e 5a
2074 eb8f 2a f7
2075 eb91 b6 a7
2076 eb93 cc eb 48
2077
2078 eb96 28 03
2079 eb98 ab 06
2080 eb9a 81
2081
2082 eb9b ab 06
2083 eb9d 29 02
2084 eb9f a0 06
2085 eba1 81
2086
2087 eba2 97
2088 eba3 54
2089 eba4 54
2090 eba5 54
2091 eba6 54
2092 eba7 a4 0f
2093 eba9 81
2094
2095
2096
2097
2098
2099
2100
2101 ebaa a6 a0
2102 ebac c7 01 00
2103 ebae c7 01 01
2104 ebb2 c7 01 03
2105 ebb5 c7 01 04
2106 ebb8 a6 2d
2107 ebba c7 01 02
2108 ebbd a6 20
2109 ebbf ae 05
2110 ebc1 d7 01 00
2111 ebc4 5c
2112 ebc5 a3 45
2113 ebc7 26 f8
2114 ebc9 3f 9f
2115 ebc9 3f 9f
2116 ebc9 3f 9f
2117 ebc9 3f 9f
2118 ebd1 3f 92
2119 ebd3 3f b0
2120 ebd5 11 ca
2121 ebd7 17 ca
2122 ebd9 15 c9
2123
2124 ebdb 5f
2125 ebdc a6 2d
2126 ebde e7 c1
2127 ebe0 5c
2128 ebe1 a3 08
2129 ebe3 26 f9
2130 ebe5 81
2131
2132 ebe6 5f
2133 ebe7 a6 ff
2134 ebe9 d7 01 45
2135 ebec 5c
2136 ebed a3 b0
2137 ebef 26 f8
2138 ebf1 81

*****
*
*   Send and clock data to LCD module.
*
*
*   Check to see if LCD module is busy.
*
*****

CLOCK  STA   PORTC
        BSET  4,PORTD
        BCLR  4,PORTD   CLOCK IT
        RTS

WAIT   BCLR  2,PORTD
        BSET  3,PORTD   READ LCD MODULE BUSY FLAG
        BCLR  4,PORTD

WLOOP  CLR   PORTCD     INPUT ON PORTC
        BSET  4,PORTD   CLOCK HIGH
        LDA   PORTC     READ MODULE
        BCLR  4,PORTD   CLOCK LOW
        STA   W7
        BRSET 7,W7,WLOOP BUSHY ?
        COM  PORTCD     OUTPUT ON PORTC
        BCLR  3,PORTD
        RTS

*****
*
*   Hex->BCD conversion (& decimal adjust).
*
*****

CBCD   BSR   UPX
        BSR  ADJI     DECIMAL ADJUST
        STA  W7       SAVE
BCD    ADD  #16      ADD $16 (BCD 10)
        BSR  ADJU     ADJUST
        DECB
        BPL  BCD     TOO FAR ?
        LDA  W7       YES, RESTORE A
        JMP  SPLIT

ADJU   BHCC  ADJI    OVERFLOW ?
        ADD  #6      YES
        RTS

ADJI   ADD  #6      NO, BUT IS LS DIGIT
        BHCS  ARTS   BIGGER THAN 9 ?
        SUB  #6      NO, RESTORE

ARTS   RTS

UPX    TAX
        LSRX
        LSRX
        LSRX
        LSRX
        AND  #50F    MSB IN X
                        LSB IN A
        RTS

*****
*
*   LCD initialisation.
*
*****

INITD  LDA  #5A0
        STA  RT
        STA  RT+1
        STA  RT+3
        STA  RT+4
        LDA  #52D
        STA  RT+2
        LDA  #520
        LDX  #5
        STA  RT,X
        INCX
        CPX  #69
        BNE  CLOP
        CLR  DISP1
        CLR  DISP2
        CLR  PTY
        CLR  PIN
        CLR  PIN+1
        CLR  DI
        BCLR 0,STAT3
        BCLR 3,STAT3
        BCLR 2,STAT2
        AND  M/S
        CLEAR TP FLAG
        CANCEL RT DISPLAY

PLOP3  CLRX
        LDA  #52D
        STA  PSN,X
        CLEAR PS NAME
        INCX
        CPX  #8
        BNE  PLOP3
        RTS

CLREON CLRX
ELOP   LDA  #5FF
        STA  EON,X
        INCX
        CPX  #176
        BNE  ELOP
        RTS

```

```

2140
2141
2142
2143
2144
2145
2146 ebf2 20 20 41 6c 61 72 ALARMS FCC ' Alarm - OFF '
      6d 20 2d 20 20 4f
      46 46 20 20
2147 ec02 20 50 49 20 63 6f PIST FCC ' PI code - '
      64 65 20 2d 20 20
      20 20 20 20
2148 ec12 20 54 50 20 2d 20 TAPST FCC ' TP - 0 TA - 0 '
      30 20 20 54 41 20
      2d 20 30 20
2149 ec22 20 50 49 4e 20 6e PINST1 FCC ' PIN no. - '
      6f 2e 20 2d 20 20
      20 20 20 20
2150 ec32 20 20 20 20 74 68 PINST2 FCC ' th at --- '
      20 61 74 20 2d 2d
      2e 2d 2d 20
2151 ec42 20 4d 4a 20 64 61 MJDST FCC ' MJ day - '
      79 20 2d 20 20 20
      20 20 20 20
2152 ec52 20 53 6c 65 65 70 SLPST FCC ' Sleep 0 min. '
      20 20 20 30 20 6d
      69 6e 2e 2f
2153 ec62 20 4d 2f 53 20 20 MSDST FCC ' M/S S DI 0 '
      53 20 20 20 44 49
      20 20 30 20
2154
2155
2156
2157
2158
2159
2160
2161 ec72 4d 6f 6e 54 75 65 DNAME FCC 'MonTueWedThuFriSatSun'
      57 65 64 54 68 75
      46 72 69 53 61 74
      53 75 6e
2162
2163 ec87 69 6e 76 FCC 'inv'
2164
2165 ec8a 4a 61 6e 46 65 62 MNAME FCC 'JanFebMarAprMayJunJulAugSepOctNovDec'
      4d 61 72 41 70 72
      4d 61 79 4a 75 6e
      4a 75 6c 41 75 67
      53 65 70 4f 63 74
      4e 6f 76 44 65 63
2166
2167
2168
2169
2170
2171
2172
2173
2174 ecae 6e 6f 20 70 72 6f PTYT FCC 'no program type ' 0
      67 72 61 6d 20 74
      79 70 65 20
2175 ecbe 20 20 20 20 20 20 FCC ' News ' 1
      4e 65 77 73 20 20
      20 20 20 20
2176 ecce 43 75 72 72 65 6e FCC 'Current affairs ' 2
      74 20 61 66 66 61
      69 72 73 20
2177 ecde 20 20 49 6e 66 6f FCC ' Information ' 3
      72 6d 61 74 69 6f
      6e 20 20 20
2178 ecee 20 20 20 20 20 53 FCC ' Sport ' 4
      70 6f 72 74 20 20
      20 20 20 20
2179 ecfe 20 20 20 45 64 75 FCC ' Education ' 5
      63 61 74 69 6f 6e
      20 20 20 20
2180 ed0e 20 20 20 20 20 44 FCC ' Drama ' 6
      72 61 6d 61 20 20
      20 20 20 20
2181 ed1e 20 20 20 20 43 75 FCC ' Culture ' 7
      6c 74 75 72 65 20
      20 20 20 20
2182 ed2e 20 20 20 20 53 63 FCC ' Science ' 8
      69 65 6e 63 65 20
      20 20 20 20
2183 ed3e 20 20 20 20 20 56 FCC ' Varied ' 9
      61 72 69 65 64 20
      20 20 20 20
2184 ed4e 20 20 20 50 6f 70 FCC ' Pop music ' 10
      20 6d 75 73 69 63
      20 20 20 20
2185 ed5e 20 20 20 52 6f 63 FCC ' Rock music ' 11
      6b 20 6d 75 73 69
      63 20 20 20
2186 ed6e 20 45 61 73 79 20 FCC ' Easy listening ' 12
      6c 69 73 74 65 6e
      69 6e 67 20
2187 ed7e 20 4c 69 67 68 74 FCC ' Light classics ' 13
      20 63 6c 61 73 73
      69 63 73 20
2188 ed8e 53 65 72 69 6f 75 FCC 'Serious classics' 14
      73 20 63 6c 61 73
      73 69 63 73
2189 ed9e 20 20 4f 74 68 65 FCC ' Other music ' 15
      72 20 6d 75 73 69
      63 20 20 20

```

```

2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206 edae 7e 7e 7e 7e V TAB FCB $7E,$7E,$7E,$7E all
2207 edb2 7e 7e 7e 7e FCB $7E,$7E,$7E,$7E all
2208 edb6 7e 7e 7e 7e FCB $7E,$7E,$7E,$7E all
2209 edba 7e 7e 7e 7e FCB $7E,$7E,$7E,$7E all
2210
2211 edbe 7e 7e 7e 7e FCB $7E,$7E,$7E,$7E all
2212 edc2 7e 7e 7e 7e FCB $7E,$7E,$7E,$7E all
2213 edc6 7e 7e 7e 7e FCB $7E,$7E,$7E,$7E all
2214 edca 7e 7e 7e 7e FCB $7E,$7E,$7E,$7E all
2215
2216 edce 7e 7b 7a 7e FCB $7E,$7B,$7A,$7E ! " # #
2217 edd2 7e 7e 7e 7a FCB $7E,$7E,$7E,$7A $ % & ' $%&
2218 edd6 7e 7e 7e 7e FCB $7E,$7E,$7E,$7E ( ) * + all
2219 edda 3f 7d 3e 7d FCB $3F,$7D,$3E,$7D , - . /
2220
2221 edde 00 01 02 03 FCB $00,$01,$02,$03 0 1 2 3
2222 ede2 04 05 06 07 FCB $04,$05,$06,$07 4 5 6 7
2223 ede6 08 09 7d 7e FCB $08,$09,$7D,$7E 8 9 : ; ;
2224 edea 7e 7e 7e 7c FCB $7E,$7E,$7E,$7C < = > ? <=>
2225
2226 edee 7e 0a 0b 0c FCB $7E,$0A,$0B,$0C @ A B C @
2227 edf2 0d 0e 0f 10 FCB $0D,$0E,$0F,$10 D E F G
2228 edf6 11 12 13 14 FCB $11,$12,$13,$14 H I J K
2229 edfa 15 16 17 18 FCB $15,$16,$17,$18 L M N O
2230
2231 edfe 19 1a 1b 1c FCB $19,$1A,$1B,$1C P Q R S
2232 ee02 1d 1e 1f 20 FCB $1D,$1E,$1F,$20 T U V W
2233 ee06 21 22 23 7e FCB $21,$22,$23,$7E X Y Z [ \
2234 ee0a 7e 7e 7e 7d FCB $7E,$7E,$7E,$7D \ ] ^ _ \ ] ^
2235
2236 ee0e 7a 2a 25 26 FCB $7A,$24,$25,$26 ' a b c
2237 ee12 27 28 29 2a FCB $27,$28,$29,$2A d e f g
2238 ee16 2b 2c 2d 2e FCB $2B,$2C,$2D,$2E h i j k
2239 ee1a 2f 30 31 32 FCB $2F,$30,$31,$32 l m n o
2240
2241 ee1e 33 34 35 36 FCB $33,$34,$35,$36 p q r s
2242 ee22 37 38 39 3a FCB $37,$38,$39,$3A t u v w
2243 ee26 3b 3c 3d 7e FCB $3B,$3C,$3D,$7E x y z { }
2244 ee2a 7e 7e 7e 7e FCB $7E,$7E,$7E,$7E | } ~ all
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261 ee2e bf ae TRA STX NUM2 CLEAR DESTINATION
2262 ee30 cd ef 86 JSR CLRAS AND ADD IT TO No. AT NUM1
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272 ee33 3f ab ADD CLR CARRY
2273 ee35 bf a7 STX W7
2274 ee37 bf a5 AD STX W5 ANSWER POINTER
2275 ee39 a6 09 LDA #ND
2276 ee3b b7 ac STA COUNT
2277 ee3d be ad LDX NUM1 1st No. POINTER
2278 ee3f bf a3 STX W3
2279 ee41 be ae LDX NUM2 2nd No. POINTER
2280 ee43 bf a4 STX W4
2281 ee45 be a3 LOOP LDX W3
2282 ee47 e6 08 LDA ND-1,X
2283 ee49 3a a3 DEC W3
2284 ee4b be a4 LDX W4
2285 ee4d eb 08 ADD ND-1,X
2286 ee4f 3a a4 DEC W4
2287 ee51 bb ab ADD CARRY SET ON ADDITION OVERFLOW
2288 ee53 3f ab CLR CARRY OR POS. RESULT SUBTRACTION
2289 ee55 ad 11 BSR ADJ DECIMAL ADJUST
2290 ee57 be a5 LDX W5
2291 ee59 e7 08 STA ND-1,X SAVE ANSWER
2292 ee5b 3a a5 DEC W5
2293 ee5d 3a ac DEC COUNT
2294 ee5f 26 e4 BNE LOOP DONE ?
2295 ee61 be a7 LDX W7
2296 ee63 81 RTS
2297
2298 ee64 a0 0a AJ SUB #10 YES, SUBTRACT 10
2299 ee66 3c ab INC CARRY AND RECORD CARRY
2300 ee68 a1 0a ADJ CMP #10
2301 ee6a 24 f8 BHS AJ 10 OR MORE ?
2302 ee6c 81 RTS NO

```

```

2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314 ee6d bf a6      SUB   STX   W6           ANSWER POINTER
2315 ee6f ad 06      BSR   COM2          9S COMP. SECOND NUMBER
2316 ee71 3f ab      CLR   CARRY         SET CARRY TO ONE
2317 ee73 3c ab      INC   CARRY         BEFORE ADDING
2318 ee75 ad c0      BSR   AD            ADD FIRST NUMBER
2319
2320 ee77 be ae      COM2  LDX   NUM2          9S COMPLIMENT
2321 ee79 ad 03      BSR   COMP          SECOND NUMBER
2322 ee7b be a6      LDX   W6            RESTORE ANSWER POINTER
2323 ee7d 81          RTS
2324
2325 ee7e a6 09      COMP  LDA   #ND        9S COMPLIMENT
2326 ee80 b7 ac      STA   COUNT
2327 ee82 a6 09      LOOP3 LDA   #S09
2328 ee84 e0 08      SUB   ND-1,X
2329 ee86 e7 08      STA   ND-1,X
2330 ee88 5a          DECX
2331 ee89 3a ac      DEC   COUNT
2332 ee8b 26 f5      BNE   LOOP3
2333 ee8d 81          RTS
2334
2335 ee8e ad ee      COM10 BSR   COMP          NINES COMPLIMENT THEN
2336 ee90 a6 09      ADD1  LDA   #ND        ADD 1 FOR TENS COMPLIMENT
2337 ee92 b7 ac      STA   COUNT          ENTER WITH X = REG-ND
2338 ee94 6c 11      ADD2  INC   2*ND-1,X
2339 ee96 e6 11      LDA   2*ND-1,X
2340 ee98 a1 0a      CMP   #S0A
2341 ee9a 25 09      BLO   RETURN
2342 ee9c a0 0a      SUB   #10
2343 ee9e e7 11      STA   2*ND-1,X
2344 eea0 5a          DECX
2345 eea1 3a ac      DEC   COUNT
2346 eea3 26 ef      BNE   ADD2
2347 eea5 81          RETURN RTS
2348
2349
2350
2351
2352
2353
2354
2355 eea6 ae 54      MULT  LDX   #R
2356 eea8 cd ef 86  JSR   CLRAS
2357 eeab ae 4b      LDX   #TMP
2358 eead cd ef 86  JSR   CLRAS          CLEAR RESULT
2359 eeb0 ae 12      LDX   #2*ND
2360 eeb2 bf a6      STX   W6            INIT. R POINTER
2361 eeb4 ae 09      LDX   #ND
2362 eeb6 e6 41      STR   LDA   P-1,X
2363 eeb8 bf a1      STX   W1            SAVE P POINTER
2364 eeba b7 ab      STA   CARRY         SAVE P
2365 eebc ae 09      LDX   #ND
2366 eebe e6 2f      XTT  LDA   Q-1,X
2367 eec0 b7 a4      STA   W4            SAVE Q
2368 eec2 27 30      BEQ   TZ0           IF ZERO GOTO NEXT Q
2369 eec4 b6 ab      LDA   CARRY
2370 eec6 b7 a3      STA   W3            RECALL P
2371 eec8 4f          CLRA
2372 eec9 34 ab      PLY  LSR   CARRY     RIGHT SHIFT INTO C
2373 eecb 24 02      BCC   SHF           C = ZERO ?
2374 eecd bb a4      ADD   W4            NO, A=A+Q
2375 eecf 3d ab      SHF  TST   CARRY     ZERO ?
2376 eed1 27 04      BEQ   C4            YES, FINISHED WITH THIS Q
2377 eed3 38 a4      ASL   W4            NO, LEFT SHIFT Q
2378 eed5 20 f2      BRA   PLY
2379 eed7 5a          C4   DECX
2380 eed8 bf a2      STX   W2            Q = Q + 1
2381 eeda be a6      LDX   W6            SAVE Q POINTER
2382 eedc eb 4a      ADD   R-ND-1,X     R POINTER
2383 eede cd ee 68  JSR   ADJ           ADD R TO A
2384 eeef e7 4a      STA   R-ND-1,X     ADJUST
2385 eef3 b6 ab      LDA   CARRY         R = R + A
2386 eef5 eb 49      ADD   R-ND-2,X     ADD R-(ND+2) TO CARRY
2387 eef7 e7 49      STA   R-ND-2,X     R-(ND+2) = R-(ND+2) + CARRY
2388 eef9 b6 a3      LDA   W3            RECALL P
2389 efeb b7 ab      STA   CARRY         SAVE IN CARRY
2390 eeed 5a          DECX
2391 eeee bf a6      STX   W6            SAVE R POINTER
2392 eef0 be a2      LDX   W2            Q POINTER
2393 eef2 20 03      BRA   C3
2394 eef4 3a a6      TZ0  DEC   W6
2395 eef6 5a          DECX
2396 eef7 26 c5      C3   BNE   XTT
2397 eef9 b6 a6      LDA   W6            R POINTER
2398 eefb ab 08      ADD   #ND-1
2399 eefd b7 a6      STA   W6
2400 eeff be a1      LDX   W1            R = R + ND-1
2401 ef01 5a          DECX
2402 ef02 26 b2      BNE   STR           P = P + 1
2403 ef04 ae 54      LDX   #R            IF NOT ZERO GOTO NEXT P
2404 ef06 81          RTS

```

```

2406
2407
2408
2409
2410
2411
2412
2413
2414
2415 ef07 ae 54
2416 ef09 cd ef 86
2417 ef0c ae 42
2418 ef0e bf ad
2419 ef10 ae 4b
2420 ef12 cd ee 2e
2421 ef15 ae 30
2422 ef17 bf ad
2423 ef19 ae 39
2424 ef1b cd ee 2e
2425
2426 ef1e a6 09
2427 ef20 b7 ac
2428 ef22 ae 39
2429 ef24 f6
2430 ef25 26 07
2431 ef27 cd ef 64
2432 ef2a 26 f6
2433 ef2c 20 33
2434 ef2e b6 ac
2435 ef30 b7 al
2436
2437 ef32 ae 4b
2438 ef34 bf ad
2439 ef36 cd ee 6d
2440 ef39 b6 ab
2441 ef3b 27 06
2442 ef3d be al
2443 ef3f 6c 53
2444 ef41 20 ef
2445 ef43 ae 4b
2446 ef45 cd ee 33
2447 ef48 ae 39
2448 ef4a a6 08
2449 ef4c b7 ac
2450 ef4e e6 07
2451 ef50 e7 08
2452 ef52 5a
2453 ef53 3a ac
2454 ef55 26 f7
2455 ef57 6f 08
2456 ef59 3c al
2457 ef5b b6 al
2458 ef5d al 0a
2459 ef5f 26 dl
2460 ef61 ae 54
2461 ef63 81
2462
2463
2464
2465
2466
2467
2468
2469 ef64 b7 a3
2470 ef66 cd ef 79
2471 ef69 be al
2472 ef6b e6 01
2473 ef6d f7
2474 ef6e 5c
2475 ef6f b3 a2
2476 ef71 26 f8
2477 ef73 b6 a3
2478 ef75 f7
2479 ef76 3a ac
2480 ef78 81
2481
2482 ef79 bf al
2483 ef7b a6 08
2484 ef7d 5c
2485 ef7e 4a
2486 ef7f 26 fc
2487 ef81 bf a2
2488 ef83 81
2489
2490
2491
2492
2493
2494
2495
2496 ef84 ae 30
2497 ef86 bf a5
2498 ef88 a6 09
2499 ef8a b7 ac
2500 ef8c 7f
2501 ef8d 5c
2502 ef8e 3a ac
2503 ef90 26 fa
2504 ef92 be a5
2505 ef94 81

*****
*
* Division of BCD numbers.
*
* R <- P / Q, remainder in TMP.
* on exit X = #R, TMQ used.
*
*****

DIV LDX #R CLEAR
JSR CLRAS RESULT
LDX #P TRANSFER
STX NUM1 P TO
LDX #TMP WORKING
JSR TRA P (TMP)
LDX #Q TRANSFER
STX NUM1 Q TO
LDX #TMQ WORKING
JSR TRA Q (TMQ)

POSS LDA #ND NUMBER
STA COUNT DIGITS

LOOP6 LDX #TMQ FIND LEAST SIGNIFICANT
LDA 0,X NON-ZERO DIGIT
BNE NOSH ZERO ?
JSR SHIFT YES, SHIFT Q
BNE LOOP6 UP ONE PLACE
ZQ BRA RTRN Q WAS ZERO
NOSH LDA COUNT SAVE
STA W1 NO. DIDITS - No. SHIFTS

SUBB LDX #TMP SUBTRACT Q
STX NUM1 FROM
JSR SUB P
LDA CARRY TOO FAR ?
BEQ NEXTD IF YES, GO TO NEXT DIGIT
LDX W1 INCREMENT RELEVANT
INC R-1,X DIGIT IN RESULT
BRA SUBB ONCE AGAIN
NEXTD LDX #TMP TOO FAR, ADD
JSR ADD Q BACK ON
ROR LDX #TMQ SET UP TO
LDA #ND-1 SHIFT BACK
STA COUNT WORKING Q
RRR LDA ND-2,X MOVE ALL
STA ND-1,X DIGITS
DECX DOWN
DEC COUNT ONE PLACE
BNE RRR DONE ?
CLR ND-1,X CLEAR MS DIGIT
INC W1 INCREMENT POINTER
LDA W1
CMP #ND+1 FINISHED ?
BNE SUBB NO, NEXT DIGIT
RTRN LDX #R
RTS

*****
*
* Shift.
*
*****

SHIFT STA W3
JSR DR1 W1: MSD, W2: LSD
LDX W1
AGS LDA 1,X MOVE ALL DIGITS
STA 0,X UP ONE PLACE
INCX
CPX W2
BNE AGS DONE ?
LDA W3 YES, RECOVER NEW DIGIT
STA 0,X AND PUT IT IN LSD
DEC COUNT
RTS

DR1 STX W1 STORE POINTERS
LDA #ND-1 (USED IN DIGIT AND DQ)
AXL INCX
DECA
BNE AXL
STX W2
RTS

*****
*
* Clear.
*
*****

CLQ LDX #Q CLEAR Q
CLRAS STX W5
LDA #ND CLEAR No. DIGITS
STA COUNT STARTING AT X
CR CLR 0,X
INCX
DEC COUNT
BNE CR DONE ?
LDX W5
RTS

```

```

2507 *****
2508 *
2509 * MJD - day of week and year. *
2510 *
2511 * DOW = (MJD+2)MOD7 (= WD-1) (DOW) *
2512 * Y' = INT((MJD-15078.2)/3652500) (YR) *
2513 *
2514 *****
2515
2516 ef95 ae 5d MJDC LDX #MJD
2517 ef97 bf ad STX NUM1
2518 ef99 ae 42 LDX #P
2519 ef9b cd ee 2e JSR TRA P <- MTD
2520 ef9e ae 5d LDX #MJD
2521 efa0 cd f0 83 JSR T10K MJD <- MJD TIMES 10,000
2522
2523 efa3 ae 39 DOFFW LDX #P-ND
2524 efa5 cd ee 90 JSR ADD1 P <- MJD + 1
2525 efa8 ae 39 LDX #P-ND
2526 efaa cd ee 90 JSR ADD1 P <- MJD + 2
2527 efad ae 30 LDX #Q
2528 efaf cd ef 86 JSR CLRAS
2529 efb2 a6 07 LDA #7
2530 efb4 b7 38 STA Q+ND-1 Q <- 7
2531 efb6 cd ef 07 JSR DIV R <- (MJD+2)/7
2532 efb9 b6 53 LDA TMP+ND-1 REMAINDER (WD-1) IN TMP
2533 efbf b7 73 STA DOW
2534
2535 efbf ae 5d YEAR LDX #MJD
2536 efbf bf ad STX NUM1
2537 efc1 ae 30 LDX #Q
2538 efc3 bf ae STX NUM2
2539 efc5 cd f0 98 JSR TRCY Q <- CY (150782000)
2540 efc8 ae 42 LDX #P
2541 efca cd ee 6d JSR SUB P <- 10K(MJD-15078.2)
2542 efcd cd f0 a3 JSR TRDY Q <- 3652500
2543 efd0 cd ef 07 JSR DIV R <- Y' ((MJD-15078.2)/365.25)
2544 efd3 bf ad STX NUM1
2545 efd5 ae 66 LDX #YR
2546 efd7 cd ee 2e JSR TRA YR <- Y'
2547
2548 *****
2549 *
2550 * MJD - month and day. *
2551 *
2552 * M' = INT((MJD-14956.1-INT(Y'*365.25))/306001) (P) *
2553 * D = MJD-14956-INT(Y'*365.25)-INT(M'*30.6001) (Q(x10K)) *
2554 *
2555 *****
2556
2557 efd8 cd f0 b9 MONTH JSR INT R <- 10K(INT(Y'*365.25))
2558 efd9 ae 5d LDX #MJD
2559 efd9 bf ad STX NUM1
2560 efd9 ae 42 LDX #P
2561 efd9 bf ae STX NUM2
2562 efd9 cd f0 b9 JSR TRD01 P <- 149561000
2563 efd9 ae 30 LDX #Q
2564 efd9 cd ee 6d JSR SUB Q <- 10K(MJD-14956.1)
2565 efd9 bf ad STX NUM1
2566 efd9 ae 54 LDX #R
2567 efd9 bf ae STX NUM2
2568 efd9 ae 42 LDX #P
2569 efd9 cd ee 6d JSR SUB P <- 10K(MJD-14956.1-INT(Y'*365.25))
2570 efd9 cd f0 ae JSR TRDM Q <- 306001
2571 efd9 cd ef 07 JSR DIV R <- M' ( MJD-14956.1-INT(Y'*365.25) )
2572 efd9 bf ad STX NUM1 INT ( ----- )
2573 efd9 ae 42 LDX #P ( 306001 )
2574 efd9 cd ee 2e JSR TRA P <- M'
2575 efd9 b6 49 LDA P+ND-2 SAVE M'
2576 efd9 b7 6f STA MNTH
2577 efd9 b6 4a LDA P+ND-1
2578 efd9 b7 70 STA MNTH+1
2579
2580 efd9 cd f0 ae DAY JSR TRDM Q <- 306001
2581 efd9 cd f0 77 JSR MULTI R <- 10K(INT(M'*30.6001))
2582 efd9 bf ad STX NUM1
2583 efd9 ae 39 LDX #TMQ
2584 efd9 cd ee 2e JSR TRA TMQ <- 10K(INT(M'*30.6001))
2585 efd9 cd f0 6b JSR INT R <- 10K(INT(Y'*365.25))
2586 efd9 bf ae STX NUM2
2587 efd9 ae 39 LDX #TMQ
2588 efd9 bf ad STX NUM1
2589 efd9 cd ee 33 JSR ADD TMQ <- 10K(INT(Y'*365.25)+INT(M'*30.6001))
2590 efd9 bf ad STX NUM1
2591 efd9 ae 42 LDX #P
2592 efd9 bf ae STX NUM2
2593 efd9 cd f0 b9 JSR TRD01 P <- 149561000
2594 efd9 3f 47 CLR P+ND-4 P <- 149560000
2595 efd9 ae 54 LDX #R
2596 efd9 cd ee 33 JSR ADD R <- 10K(14956+INT(Y'*365.25)+INT(M'*30.6001))
2597 efd9 bf ae STX NUM2
2598 efd9 ae 5d LDX #MJD
2599 efd9 bf ad STX NUM1
2600 efd9 ae 30 LDX #Q
2601 efd9 cd ee 6d JSR SUB Q <- MJD-R (10K*DOM)
2602 efd9 e6 04 LDA ND-5,X
2603 efd9 b7 72 STA DOM+1
2604 efd9 e6 03 LDA ND-6,X
2605 efd9 b7 71 STA DOM


```

```

2607
2608
2609
2610
2611
2612
2613
2614
2615
2616
2617 f049 b6 6f
2618 f04b 27 1b
2619 f04d b6 70
2620 f04f 27 11
2621 f051 a1 04
2622 f053 25 13
2623 f055 ae 5d
2624 f057 cd ee 90
2625 f05a 3f 6f
2626 f05c 3a 70
2627 f05e 3a 70
2628 f060 20 06
2629 f062 a6 0a
2630 f064 b7 70
2631 f066 3f 6f
2632 f068 3a 70
2633 f06a 81
2634
2635 f06b ae 66
2636 f06d bf ad
2637 f06f ae 42
2638 f071 cd ee 2e
2639 f074 cd f0 a3
2640 f077 cd ee a6
2641 f07a 3f 59
2642 f07c 3f 5a
2643 f07e 3f 5b
2644 f080 3f 5c
2645 f082 81
2646
2647 f083 9f
2648 f08a ab 05
2649 f086 b7 a1
2650 f088 e6 04
2651 f08a f7
2652 f08b 5c
2653 f08c b3 a1
2654 f08e 26 f8
2655 f090 7f
2656 f091 6f 01
2657 f093 6f 02
2658 f095 6f 03
2659 f097 81
2660
2661
2662
2663
2664
2665
2666
2667 f098 ae 09
2668 f09a d6 f0 c3
2669 f09d e7 2f
2670 f09f 5a
2671 f0a0 26 f8
2672 f0a2 81
2673
2674 f0a3 ae 09
2675 f0a5 d6 f0 cc
2676 f0a8 e7 2f
2677 f0aa 5a
2678 f0ab 26 f8
2679 f0ad 81
2680
2681 f0ae ae 09
2682 f0b0 d6 f0 de
2683 f0b3 e7 2f
2684 f0b5 5a
2685 f0b6 26 f8
2686 f0b8 81
2687
2688 f0b9 ae 09
2689 f0bb d6 f0 d5
2690 f0be e7 41
2691 f0c0 5a
2692 f0c1 26 f8
2693 f0c3 81
2694
2695 f0c4 01 05 00 07 08 02 CY FCB 1,5,0,7,8,2,0,0,0
00 00 00
2696 f0cd 00 00 03 06 05 02 DY FCB 0,0,3,6,5,2,5,0,0
05 00 00
2697 f0d6 01 04 09 05 06 01 D01 FCB 1,4,9,5,6,1,0,0,0
00 00 00
2698 f0df 00 00 00 03 00 06 DM FCB 0,0,0,3,0,6,0,0,1
00 00 01
2699
2700
2701
2702
2703
2704
2705
2706 fff4 ORG $FFF4
2707
2708 fff4 e0 00 FDB START SERIAL
2709 fff6 e2 9d FDB TINTB TIMER B
2710 fff8 e0 00 FDB START TIMER A
2711 fffa e3 0a FDB SDATA EXTERNAL INTERRUPT & RTI
2712 fffc e0 00 FDB START SWI
2713 fffe e0 00 FDB START RESET
2714
2715

```

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 Centres:

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

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.

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

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



MOTOROLA