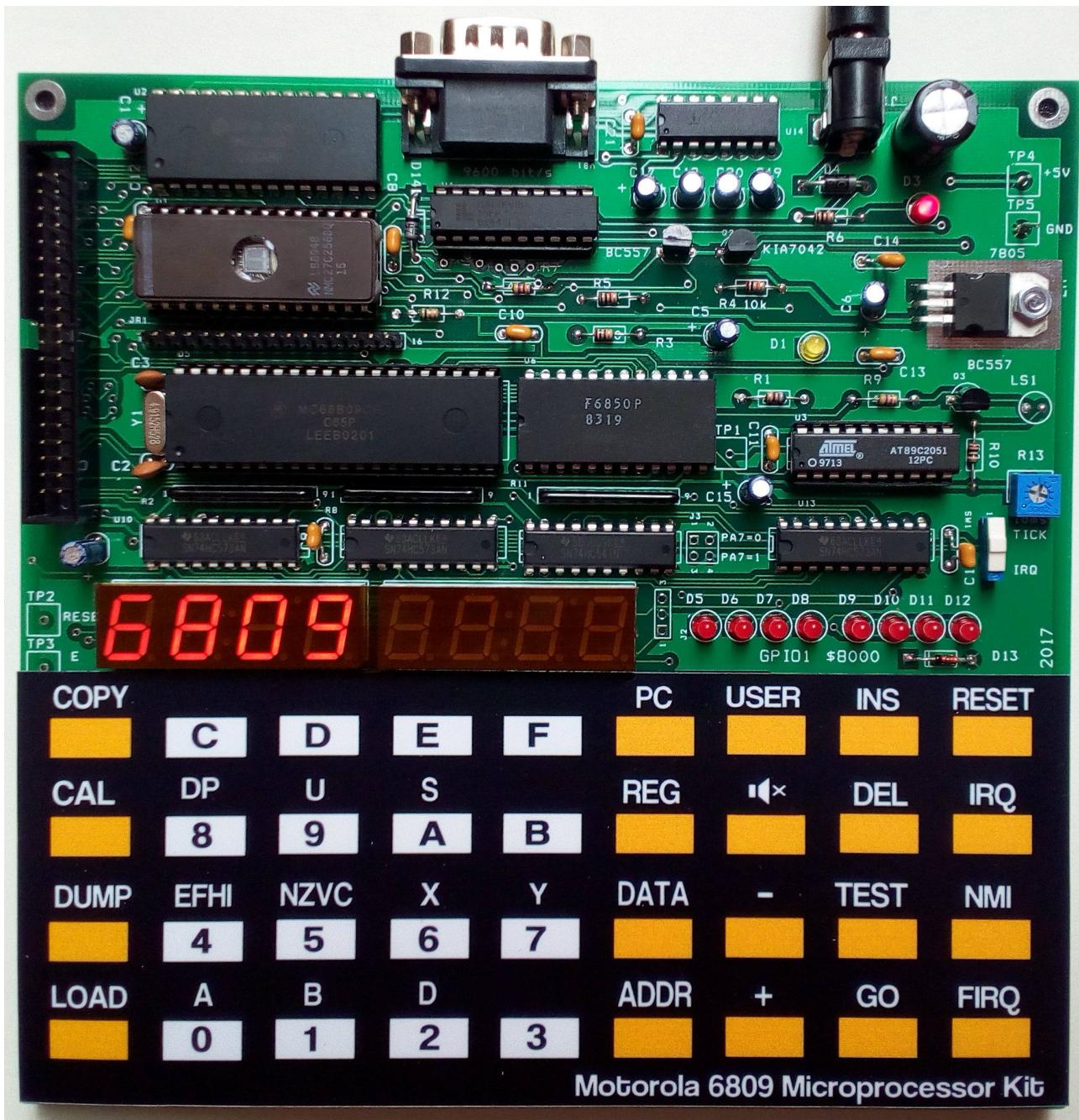


6809 Microprocessor Kit User's Manual



6809 MICROPROCESSOR KIT

CONTENTS

OVERVIEW.....	4
FUNCTIONAL BLOCK DIAGRAM.....	4
HARDWARE LAYOUT.....	5
KEYBOARD LAYOUT.....	7
HARDWARE FEATURES.....	9
MONITOR PROGRAM FEATURES.....	9
MEMORY AND I/O MAPS.....	10
INTERRUPT VECTORS.....	11
GETTING STARTED.....	12
HOW TO ENTER PROGRAM USING HEX CODE.....	14
USER REGISTERS DISPLAY.....	15
TEST CODE RUNNING WITH BREAK POINT.....	16
GPIO1 LED.....	17
CONNECTING KIT TO TERMINAL.....	18
EXPANSION BUS HEADER.....	23
10ms TICK GENERATOR.....	24
USING SYSTEM TICK	25
RS232C PORT.....	26
CONNECTING LCD MODULE.....	27
LOGIC PROBE POWER SUPPLY.....	29

CUSTOMIZING THE MONITOR PROGRAM.....30

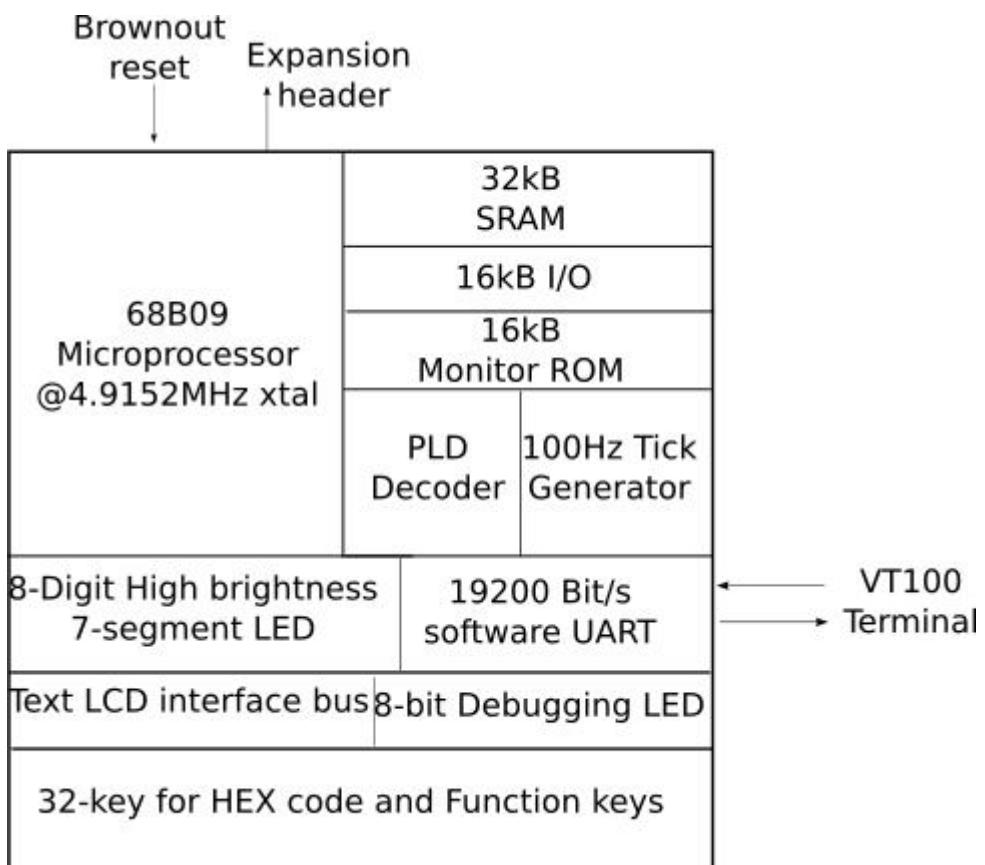
HARDWARE SCHEMATIC, PARTS LIST

MONITOR PROGRAM LISTINGS

OVERVIEW

The 6809 Microprocessor kit is a single board microcomputer designed for self learning the operation of the 6809 microprocessor. We can enter the 6809 instructions using hex key directly. The kit provides 32kB RAM for program testing. The 16kB ROM is monitor program. Tick generator provides 10ms tick signal for testing with interrupt. In addition, the UART provides serial interface for Motorola s-record loading.

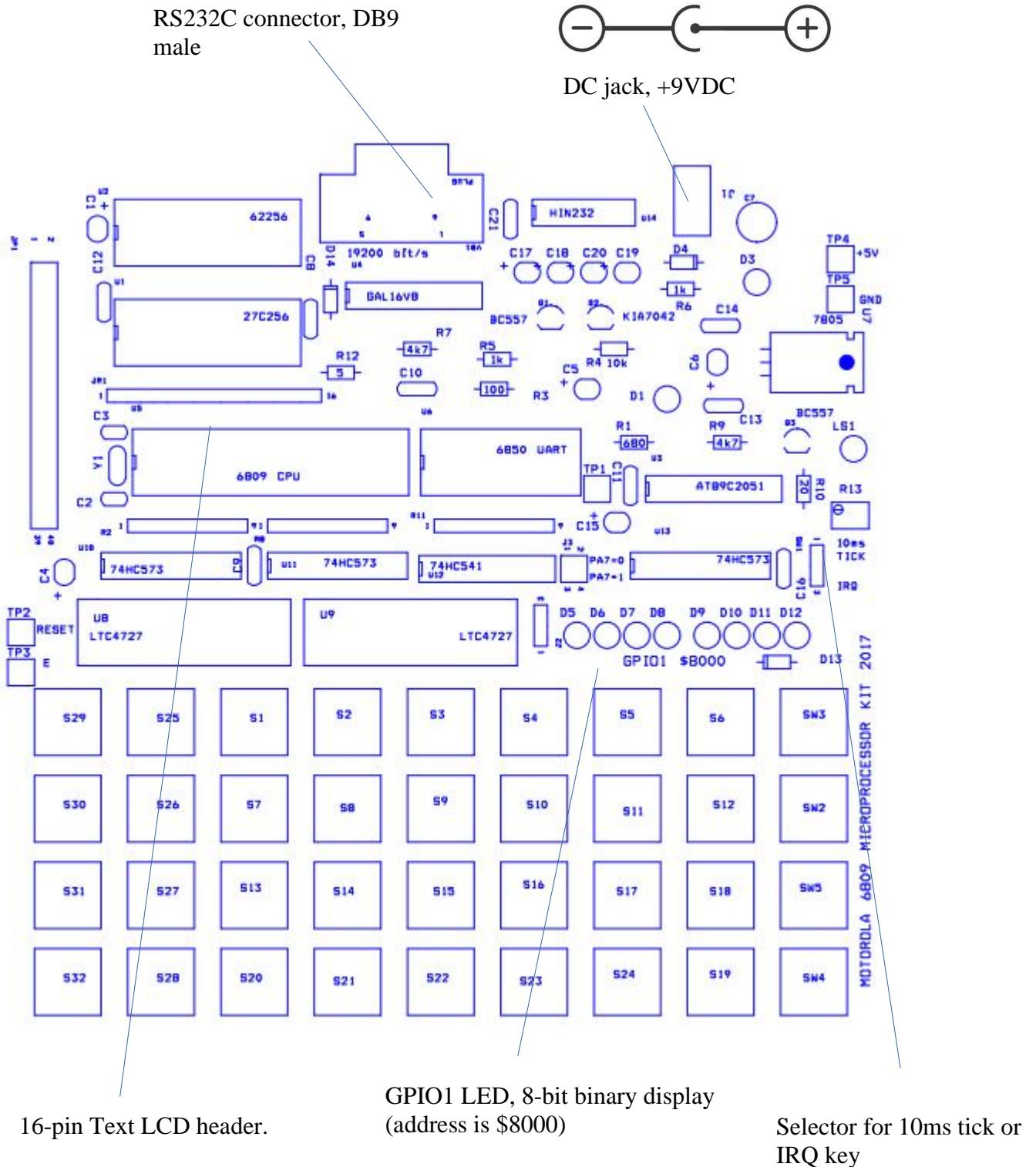
FUNCTIONAL BLOCK DIAGRAM



Notes

1. UART speed is 19200 bit/s.
2. The kit has LCD module interfacing with 6809 bus.
3. 100Hz Tick generator is for interrupt experiment.
4. Ports for display and keypad interfacing are built with discrete logic IC chips.
5. Memory and Port decoders are made with Programmable Logic Device, PLD.

HARDWARE LAYOUT



Important Notes

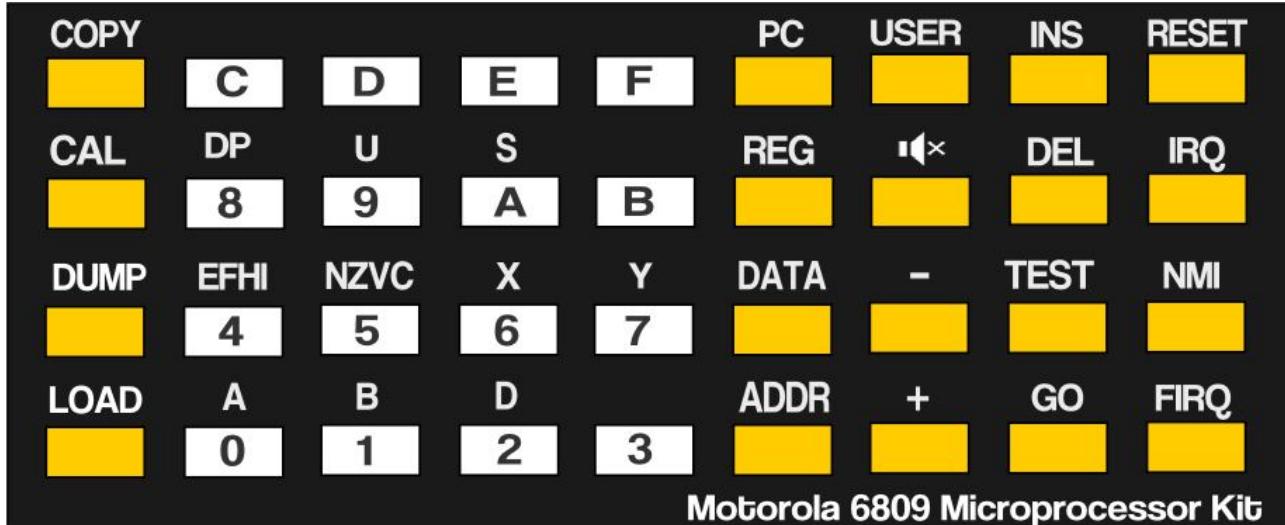
1. Insert or remove the LCD module must be done when the kit is powered off!
2. AC adapter should provide approx. +9VDC, higher voltage will cause the voltage regulator chip becomes hot.
3. The kit has diode protection for wrong polarity of adapter jack. If the center pin is not the positive (+), the diode will be reverse bias, preventing wrong polarity feeding to the voltage regulator.



4. The USB to 9VDC adapter can be powered the kit as well.



KEYBOARD LAYOUT



HEX keys Hexadecimal number 0 to F with associated user registers, A, B, D, X, Y, DP, U, S and flag bits, EFHI, NZVC

CPU control keys

RESET Reset the CPU, the 6809 will get vector from location \$FFFE. Kit provides the restart address at \$C000.

IRQ Make IRQ pin to logic low, used for experimenting with interrupt process

NMI Make NMI pin to logic low, used for experimenting with interrupt process

FIRQ Make FIRQ pin to logic low, used for experimenting with interrupt process

Monitor function keys

TEST Test 10ms tick interrupt, SW1 must be 10ms Tick position. Display shows hex number counting, the GPIO1 LED shows binary counting at 100Hz rate.

INS Insert one byte to the next location, the 512 bytes will be shifted down.

DEL Delete one byte at current display, the next 512 bytes will be moved up.

SPKR Turn off/On BEEP key

GO Jump from monitor program to user code at current ADDRESS

- Decrement current display address by one

+	Increment current display address by one
PC	Set display address to the current Program Counter
REG	Display user registers, used with HEX key.
0	A register
1	B register
2	D, or AB 16 bits register
4	Flag bits, EFHI, Condition Code high nibble
5	Flag bits, NZVC, Condition Code low nibble
6	X register
7	Y register
8	DP register
9	U register
A	S register
DATA	Set entry mode of hex keys to Data field
ADDR	Set entry mode of hex keys to Address field
COPY	Copy Block of memory, Enter start Address, used with key +, End address, and Destination, press key GO to copy then.
CAL	Calculate hex number addition +, or subtraction -, used with key +, or key -. Key GO will compute.
DUMP	Dump block memory using terminal display. Ket will need 19200 bit/s terminal interface.
LOAD	Load Motorola s-record. Set delay 1ms for character and line.
USER	User key, for user define the function. See the monitor source code in function key_exe().

HARDWARE FEATURES

Hardware features:

- CPU: Motorola 68B09, 8-bit Microprocessor @1.2288MHz instruction clock
- Memory: 32kB RAM, 16kB EPROM
- Memory and I/O Decoder chip: Programmable Logic Device GAL16V8D
- Display: high brightness 6-digit 7-segment LED
- Keyboard: 36 keys
- RS232 port: 6850 ACIA 19200 bit/s 8n1
- Debugging LED: 8-bit GPIO1 LED at location \$8000
- Tick: 10ms tick produced by 89C2051 for time trigger experiment
- Text LCD interface: direct CPU bus interface text LCD
- Brownout reset: KIA7042 reset chip for power brownout reset
- Expansion header: 40-pin header

MONITOR PROGRAM FEATURES

MONITOR program features:

- Simple hex code entering
- Insert and Delete byte
- User registers: A, B, X, Y, S, U, DP Condition code registers for storing CPU status after program execution
- HEX calculator for offset calculation
- Copy block of memory
- Motorola s-record S19 downloading
- Memory dump
- Beep ON/OFF
- TEST 10ms

MEMORY AND I/O MAPS

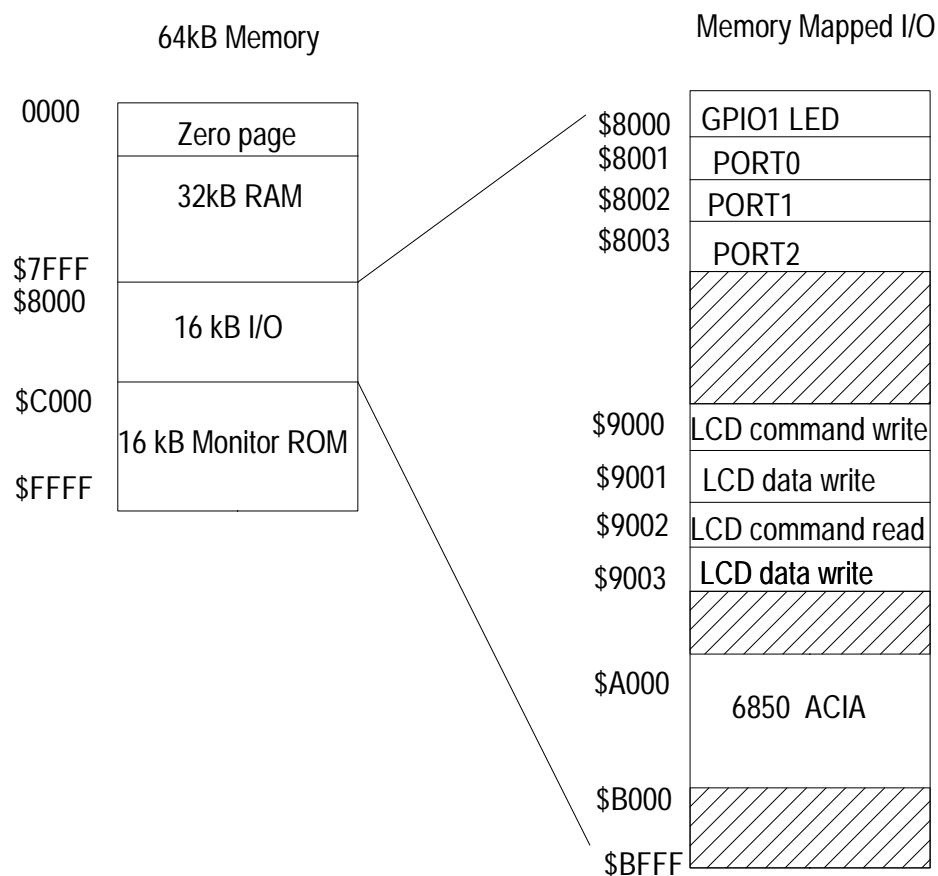
The 6809 chip can address up to 64kB using 16-bit address bus. The 64kB spaces are divided into 32kB RAM, 16kB memory mapped input/output, and 16 kB monitor ROM.

RAM starts from 0 to \$7FFF. The input/output starts from \$8000 to \$BFFF.

GPIO1 LED is the 8-bit binary display, useful for program debugging. The location is \$8000.

To see the contents of Accumulator A,

0200 86 AA	LDA #\$AA
0202 B7 80 00	STA \$8000
0205 3F	SWI



INTERRUPT VECTORS

Interrupts vectors are placed at high address from FFF0 to FFFF.

Restart interrupt vector, C000 is placed at FFFE and FFFF.

Interrupts	Vector location		6809 kit
	MS byte	LS byte	
Reset	FFFE	FFFF	C000
NMI	FFFC	FFFD	7FF3
SWI	FFFA	FFFB	C027
IRQ	FFF8	FFF9	7FF0
FIRQ	FFF6	FFF7	7FF6
SWI2	FFF4	FFF5	NA
SWI3	FFF2	FFF3	NA
Reserved	FFF0	FFF1	NA

SWI software interrupt, is used by monitor program to provide break-point running, by saving the CPU registers to user registers. The service routine is address C027.

NMI vector is relocated to RAM space, at 7FF3. Reset the kit, the instruction RTI code, 3b is put to address 7FF3. Students may replace it with the code to service the NMI directly.

Similarly to the FIRQ. We can check it with key ADDR and see these vectors easily.

GETTING STARTED

The kit accepts DC power supply with minimum voltage of +7.5V. It draws DC current approx. 300mA. However we can use +9VDC from any AC adapter. The example of AC adapter is shown below.



The center pin is positive. The outer is GND.



If your adapter is adjustable output voltage, try with approx. +9V. Higher voltage will make higher power loss at the voltage regulator, 7805. Dropping voltage across 7805 is approx. +2V. To get +5VDC for the kit, we thus need DC input >+7.5V.

When power up, we will see the text 6809.



6809

Press PC key, the display address will be 200. The data field will show its content.



0200 FF.

HOW TO ENTER PROGRAM USING HEX CODE

Let us try enter HEX CODE of the example program to the memory and test it. We write the program with 6809 instructions.

Address	Hex code	Label	Instruction	comment
0200	86 AA	MAIN	LDA #\$AA	Load A with value AA
0202	B7 80 00		STA \$8000	Write A to GPIO1 @ 8000
0205	3F		SWI	Jump back to monitor

Our test program has only three instructions.

The first instruction is

LDA #\$AA

Load A register with the 8-bit constant, AA or 10101010 in binary.

This instruction has two bytes hex code i.e., 86, and AA. 86 is instruction LDA #n and AA is n.

The 2nd instruction is

STA \$8000. Store A register to, gpio1 LED at location \$8000.

The instruction's machine code is B7. 80 00 is the location of GPIO1.

The last instruction is SWI, software interrupt. The hex code is 3F. It will jump back to monitor program.

The total of hex codes for this small program is 6 bytes that are, 86, AA, B7, 80, 00 and 3F.

The first byte will be entered to location 0200. And the following bytes will be entered at 0201, 0202, 0203, 0204, 0205. The last byte is 3F at 205.

Let us see how to enter these codes into the memory.

Step 1 Press RESET then key PC, the display will show current memory address and its contents.

0200 FF.

Shown the location 200 has data FF. There are small dots at the data field indicating the active field, ready for modifying the hex contents.

Step 2 Press key 8 and key 6. The new hex code 86 will be entered to the location 200.

0200 86.

Step 3 Press key + to increment the location from 200 to 201. Then enter hex key A, A.

0201 AA.

Repeat Step 3 until completed for the last location. We can verify the hex code with key + or key -.

To change the display location, press key ADDR. The dots will move to Address field. Any hex key pressed will change the display address.

USER REGISTERS DISPLAY

Before we test the code running, let us see how to examine user registers. User registers are the memory block in RAM that used to save the contents of CPU registers after completed a given program running. We can examine the user registers for checking our code running then.

Press key REG, then press key 0, it will show 8-bit content of register A.

FF A

For example, Key 0 to Key 3 are for A, B, AB or D..

Press key Reg, 4, 5, 6, 7 for Flag low nibble, high nibble, X and Y registers.

FCEF Y

TEST CODE RUNNING WITH BREAK POINT

Let us get back to the program we have just entered. The simple program loads accumulator A with AA and stores it to GPIO1 LED, \$8000. Last instruction, SWI will break the program running by returning to the monitor program.

0200 86 AA	LDA #\$AA
0202 B7 80 00	STA \$8000
0205 3F	SWI

Now press PC, the current display will show address 200. To run the code, press key GO.

What is happening at GPIO1 LED?

The display will show the address next location, 206

Check result in A register with key REG, 0.

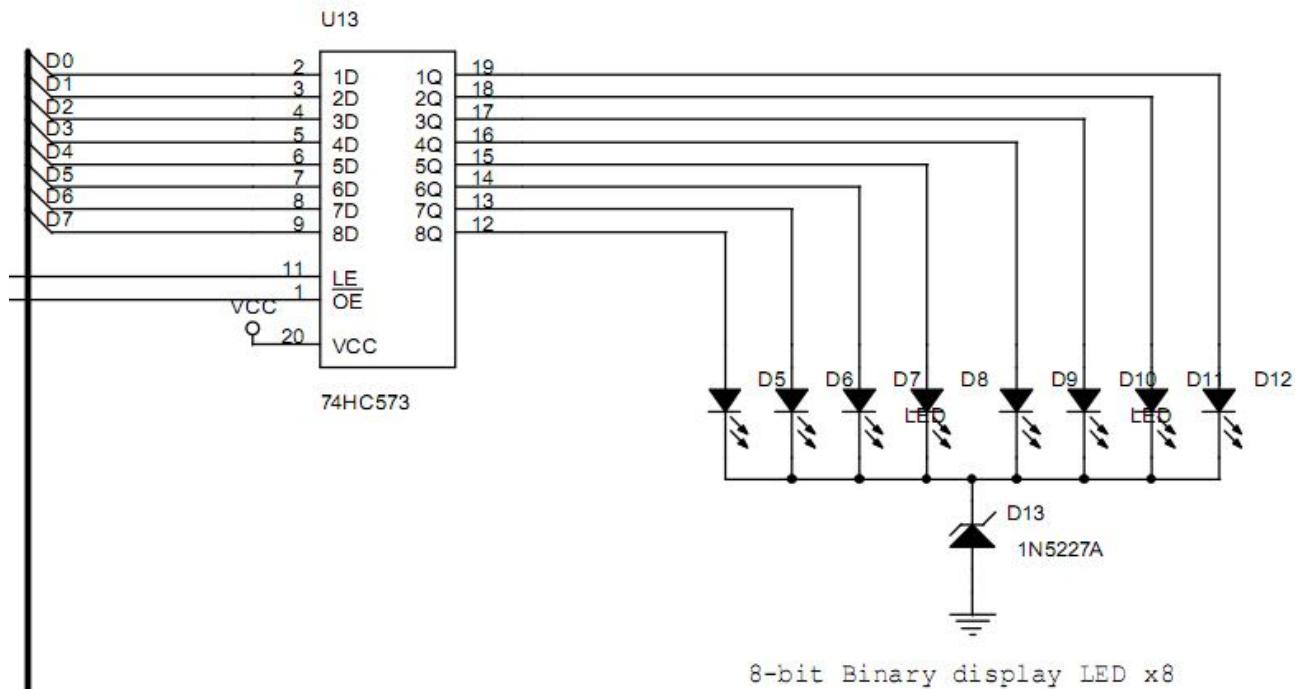
What is the value in A?

Can you change the byte being loaded from AA to another value, how?

SWI instruction is Software Interrupt. When executed, the CPU will get the locations to be called from the SWI vector located at FFFA and FFFB. All CPU registers and condition code are saved to stack memory. And will be copied to the user registers by monitor program. So we can examine them after program execution.

GPIO1 LED

The kit provides a useful 8-bit binary display. It can be used to debug the program or code running demonstration. The location is \$8000. The output port is 8-bit data flip-flop. Logic 1 at the output will make LED lit.



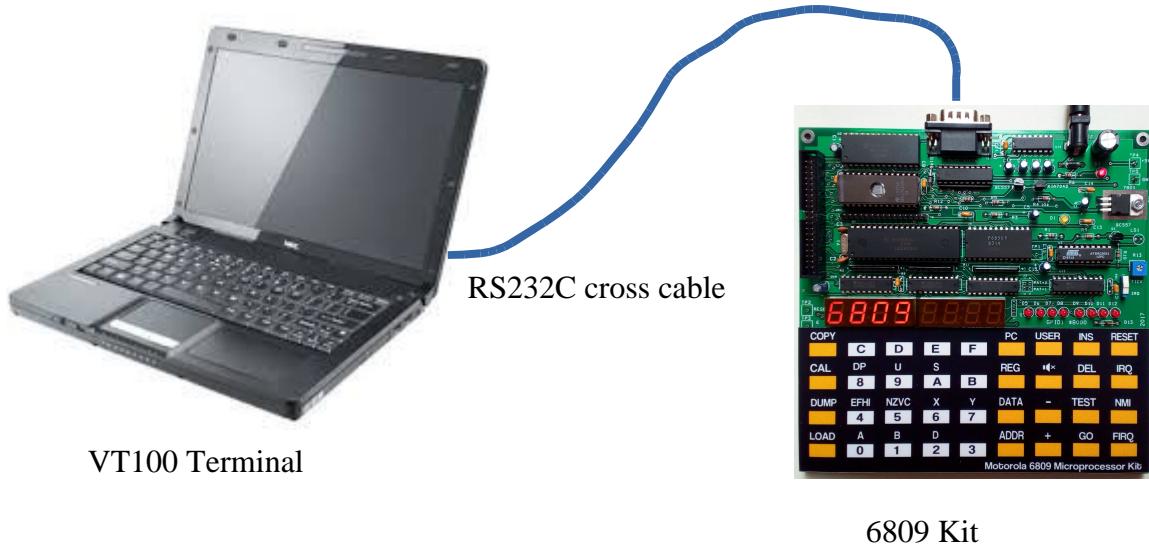
We can use instruction STA \$8000, to write the content of accumulator to this location.

Bit '1' will make the LED turn on and bit '0' will turn off. We can check 8-bit data in binary number directly.

The hex code for STA \$8000 is B7, 80, 00. Only three bytes and easy to remember.

CONNECTING 6809 KIT TO TERMINAL

We can connect the 6809 kit to a terminal by RS232C cross cable. You may download free terminal program, teraterm from this URL, <http://ttssh2.sourceforge.jp/index.html.en>

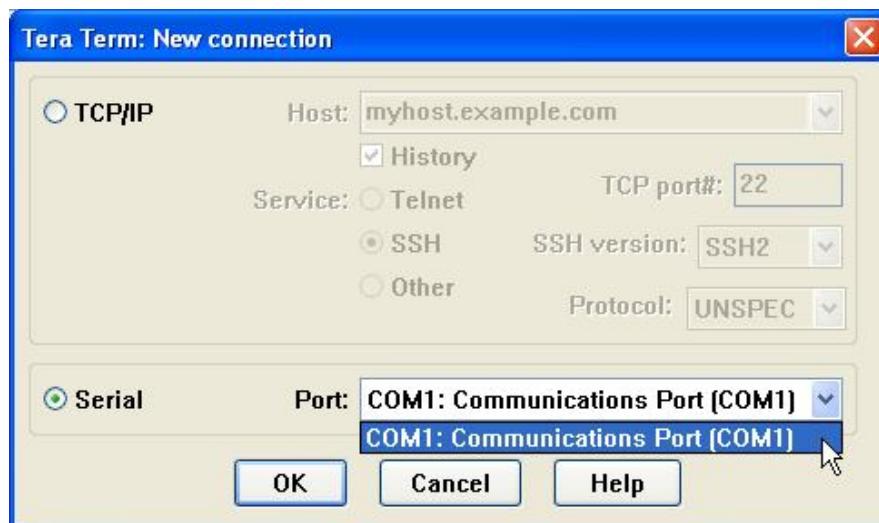


The example shows connecting laptop with COM1 port to the RS232C port of the 6809 kit.

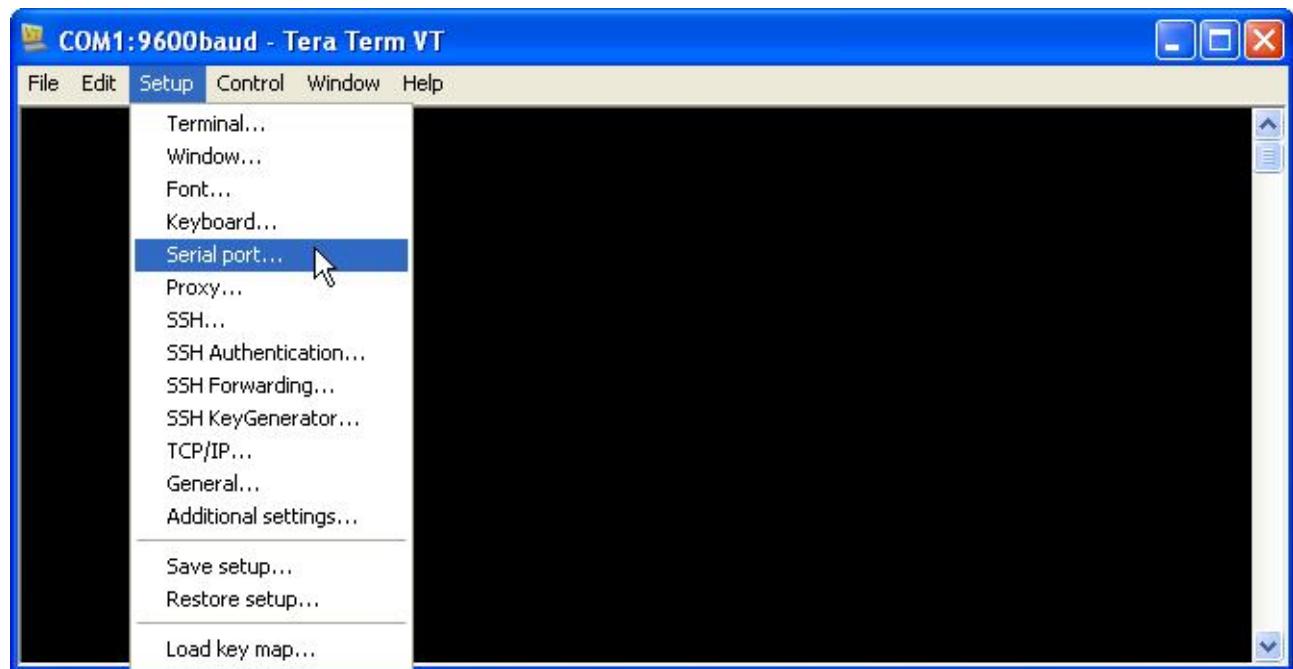
New laptop may not have the COM port, we can use the USB-RS232 adapter for converting the USB port to RS232 port.

To download Intel hex file that generated from the assembler or c compiler, set serial port speed to 19200 bit/s, 8-data bit, no parity, no flow control, one stop bit.

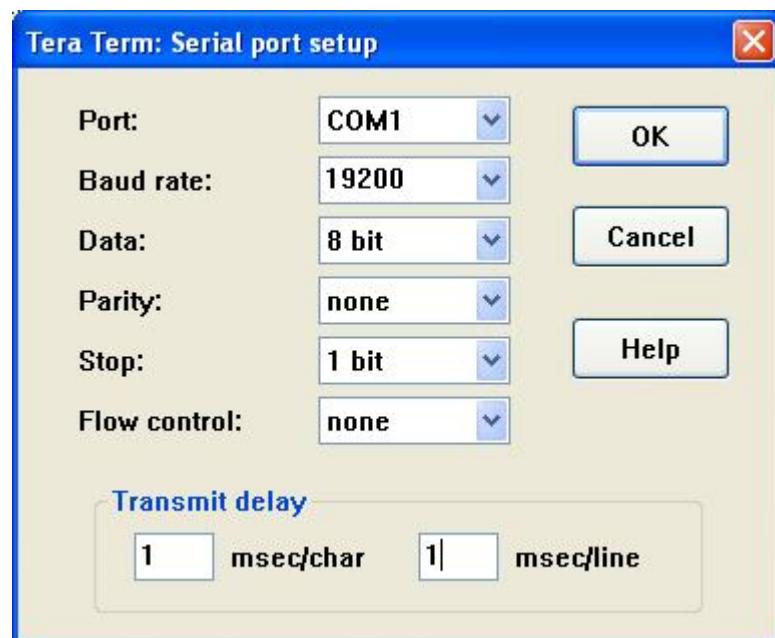
Step 1 Run teraterm, then click at Serial connection.



Step 2 Click setup>Serial port.



Step 3 Set serial port speed to 19200 and format as shown below. **Set transmit delay 1ms for character and line.**



Step 4 Press RESET on 6809 kit. The kit will send cold message on screen. Press key DUMP, the display will show the memory dump.

```

6809 MICROPROCESSOR KIT 2020
0200:86 AA B7 80 00 3F 3B 24 66 BB FF E7 3A 0A 4C 30 .....?;$f...:L0
0210:FF F7 7D BF 89 D3 68 02 CB 34 9B FB 59 0A 39 01 ..>..h..4..Y..9..
0220:FF FF 7F FD 81 12 14 40 9B AD D5 77 10 40 14 D0 .....@..w.@..
0230:FB D7 9B FC 6C 04 29 1E 99 EF BA 95 12 0D 00 80 .....1.>.....
0240:FF FF FE 12 64 84 10 BF 5F CE D2 00 04 44 18 .....d.....D..
0250:7F FF FF DF 82 16 00 40 0B 57 DF D9 88 82 01 21 .....@W.....!
0260:FF FF DE 07 40 41 00 75 86 5C 7F 02 03 84 08 .....@A.u\.....
0270:FF FF BB DC 01 4F 00 17 FF FF 7E 42 02 01 24 .....O...~B..$..
0280:F7 FD FF FE 02 A0 00 38 E6 FF 4E BE C4 8C 40 00 .....8..N..@..
0290:FF FD FF 5F 01 B0 80 40 BD 7A 4B 2E 3A 13 11 84 .....@zK.:...
02A0:FB FF BF EF 18 00 10 5D 6A 6F EE BF 42 0C 04 0D .....ljo..B...
02B0:FF FF 42 00 4C 0C BD 5F FD 3D 00 04 13 D8 .....B.L.=...:...
02C0:DB CF FF FB 10 69 06 31 7D 72 F7 E9 51 08 60 64 .....i.1>r..Q..d
02D0:F7 FF B7 FF 85 41 B8 8A F3 C7 FF A6 49 04 88 44 .....A.....I..D
02E0:F7 FF EF FF 0B 26 04 09 F9 FA D3 E7 40 41 48 02 .....&.....JAH..
02F0:DB FB FF F1 54 83 80 10 CF 2F 9D FE 00 08 0C 00 .....T...../

```

Step 5 Press key LOAD, the display will show “Load Motorola s-record”

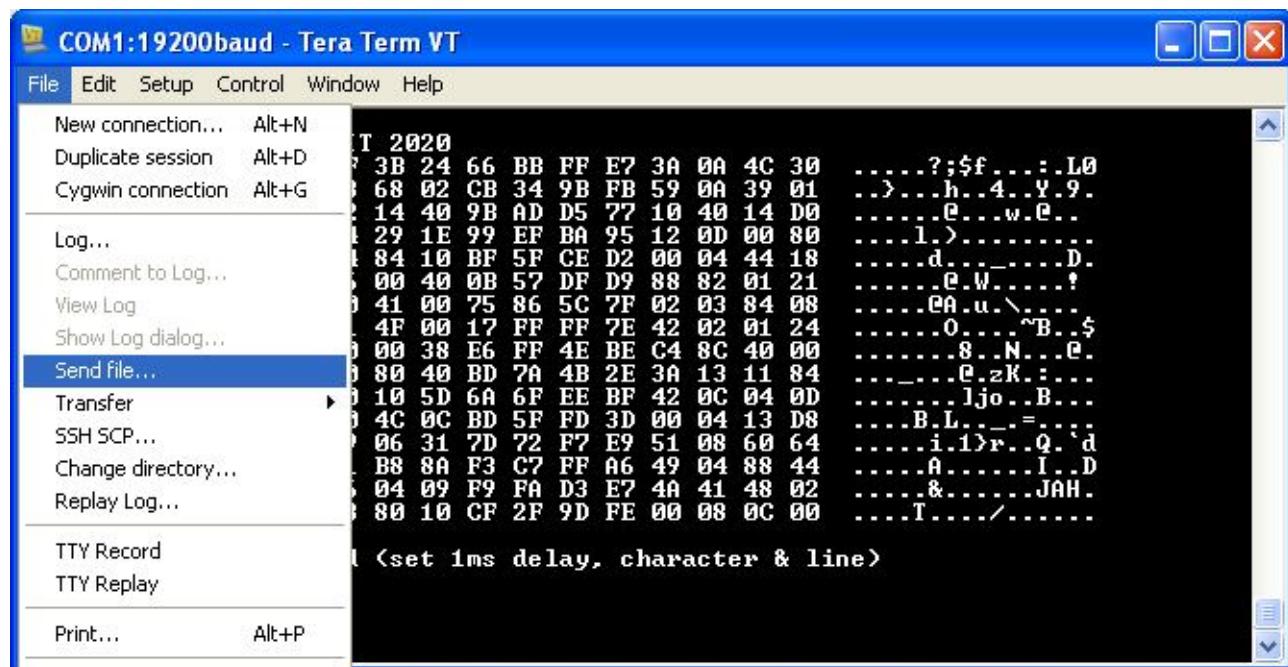
```

6809 MICROPROCESSOR KIT 2020
0200:10 CE 70 00 7E 02 0D 7E 02 52 7E 02 56 86 0C B7 ..p.^..~.R^..U...
0210:02 F3 B7 80 00 10 8E 00 00 31 3F 26 FC 86 00 B7 .....1?&...
0220:80 00 10 8E 03 E8 31 3F 26 FC BD 02 BF 20 DE C6 .....1?&...
0230:AA BD 02 5E 20 F9 3F 86 FE B4 02 F3 20 05 86 01 .....^ ?...
0240:BA 02 F3 B7 02 F3 B2 80 00 39 86 FD 20 EB 86 02 .....9...
0250:20 EE 86 FB 20 E3 86 F7 20 DF 86 0C 20 E2 34 12 .....82...4...
0260:8E 00 08 BD 02 BF 58 25 05 BD 02 4A 20 03 BD 02 .....82..J...
0270:4E BD 02 3E 7B 02 00 00 27 02 CA 01 BD 02 37 30 N..>{.....70
0280:1F 26 E0 35 12 7E 02 4A 34 12 B6 02 F3 8E 00 08 ..8.5.^J4...
0290:58 25 07 84 FD B7 80 00 20 05 8A 02 B7 80 00 8A X%.....
02A0:01 B7 80 00 7B 02 00 00 27 02 CA 01 84 FE B7 80 .....{.....
02B0:00 30 1F 26 DB 84 FD B7 80 00 B7 02 F3 35 92 10 ..0.&.....5..
02C0:8E 00 00 31 3F 26 FC 39 39 39 39 39 39 39 39 39 ..1?&.999999999999
02D0:39 39 39 39 39 39 39 39 39 39 39 39 39 39 39 39 999999999999999999
02E0:39 39 39 39 39 39 39 39 39 39 39 39 39 39 39 39 999999999999999999
02F0:39 39 39 00 00 00 00 00 00 00 00 00 00 00 00 00 999.....

```

Load Motorola s-record <set 1ms delay, character & line>

Step 6 On PC, Click file>Send File>TEST1.s19.



The kit will read the hex file, write to memory, when completed if no checksum error, the display will show 0 error.

COM1:19200baud - Tera Term VT

File Edit Setup Control Window Help

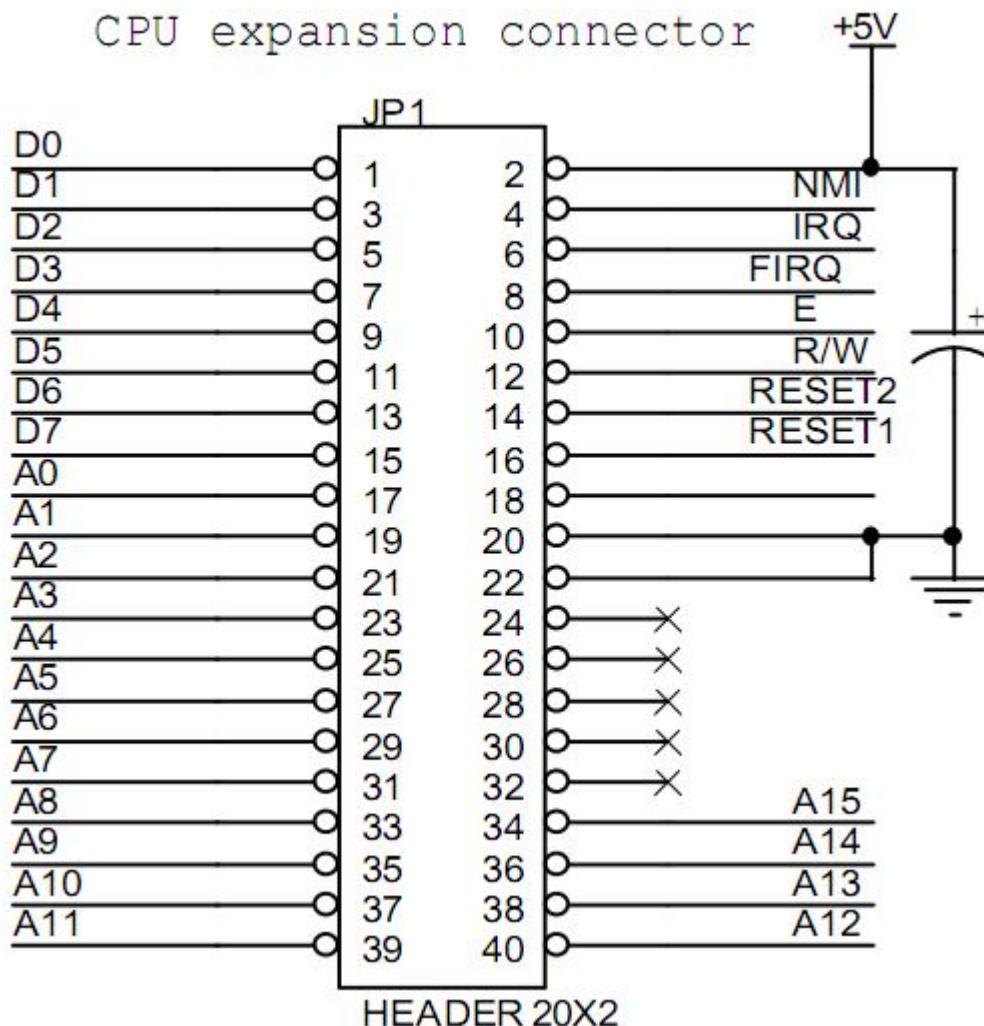
```
6809 MICROPROCESSOR KIT 2020
0200:86 AA B7 80 00 3F 3B 24 66 BB FF E7 3A 0A 4C 30 .....?,$f...:L0
0210:FF F7 7D BF 89 D3 68 02 CB 34 9B FB 59 0A 39 01 ..>..h..4..Y..9..
0220:FF FF 7F FD 81 12 14 40 9B AD D5 77 10 40 14 D0 .....@..w.@..
0230:FB D7 9B FC 6C 04 29 1E 99 EF BA 95 12 0D 00 80 .....i.>...
0240:FF FF BF FE 12 64 84 10 BF 5F CE D2 00 04 44 18 .....d..._.D..
0250:7F FF FF DF 82 16 00 40 0B 52 DF D9 88 82 01 21 .....@.W...!
0260:FF FF FF DE 07 40 41 00 75 86 5C 7F 02 03 84 08 .....@A.u.\...
0270:FF FF BB DC 01 4F 00 17 FF FF 7E 42 02 01 24 .....O...~B..$.
0280:F7 FD FF FE 02 A0 00 38 E6 FF 4E BE C4 8C 40 00 .....8..N..@..
0290:FF FD FF 5F 01 B0 80 40 BD 7A 4B 2E 3A 13 11 84 .....@.zK...:.
02A0:FB FF BF EF 18 00 10 5D 6A 6F EE BF 42 0C 04 0D .....ljo..B..
02B0:FF FF FF 42 00 4C 0C BD 5F FD 3D 00 04 13 D8 .....B.L...=...
02C0:DB CF FF FB 10 69 06 31 7D 72 F7 E9 51 08 60 64 .....i.1>r.Q..d
02D0:F7 FF 85 41 B8 8A F3 C7 FF A6 49 04 88 44 .....A....I..D
02E0:F7 FF EF FF 0B 26 04 09 F9 FA D3 E7 4A 41 48 02 .....&....JAH.
02F0:DB FB FF F1 54 83 80 10 CF 2F 9D FE 00 08 0C 00 .....T.../...
```

Load Motorola s-record (set 1ms delay, character & line)
0 errors...

Press RESET then, PC to set memory address 200, the press key GO to run the program.

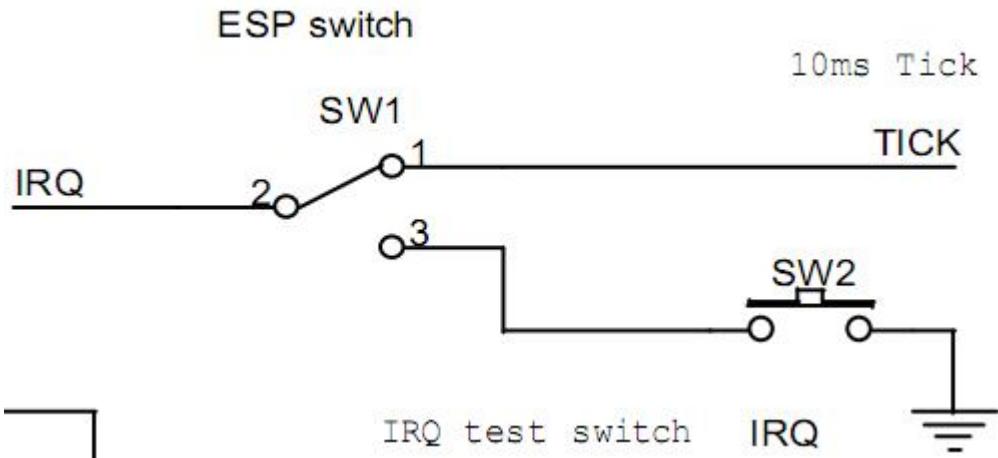
EXPANSION BUS HEADER

JP1, 40-pin header provides CPU bus signals for expansion or I/O interfacing. Students may learn how to make the simple I/O port, interfacing to Analog-to-Digital Converter, experimenting with interrupt process with FIRQ, IRQ and NMI.



10ms TICK GENERATOR

SW1 is a selector for interrupt source between key IRQ or 10ms tick produced by 89C2051 microcontroller. Tick generator is software controlled using timer0 interrupt in the 89C2051 chip. The active low tick signal is sent to P3.7. For tick running indicator, P1.7 drives D1 LED.



Tick is a 10ms periodic signal for triggering the 6809 IRQ pin. When select SW1 to Tick, the 6809 CPU can be triggered by the external interrupt. The 100Hz tick or 10ms tick can be used to produce tasks that executes with multiple of tick.



USING SYSTEM TICK

The monitor program provides 10ms system tick when IRQ is enabled and SW1 is set to 10ms position. The monitor program uses IRQ interrupt for producing 10ms time base.

System tick is one byte memory located at address \$700E in RAM. When IRQ is enabled, SW1 is set to 10ms, this memory location will be incremented by one every 10ms or 100Hz rate.

We can simply examine this variable by polling method. With 100Hz rate, if the Tick variable becomes 100, time will be 1000ms or one second exactly.

Below is the list of example program how to use 10ms system tick.

```
0001          * RUNNING CODE USING 10ms TICK
0002
0003 700E      TICK    EQU $700E
0004
0005 0200      ORG    $200
0006
0007 0200 86 00   MAIN    LDA #0
0008 0202 1F 8B      TFR A,DP    SET PAGE 0
0009
0010 0204 3C EF      CWAI #11101111 ENABLE IRQ
0011
0012 0206 B6 70 0E   LOOP    LDA TICK
0013 0209 81 64      CMPA #100
0014 020B 26 0A      BNE SKIP
0015 020D 7F 70 0E      CLR TICK
0016
0017 0210 0C 00      INC 0
0018 0212 96 00      LDA 0
0019 0214 B7 80 00      STA $8000
0020
0021 0217 20 ED      SKIP    BRA LOOP
0022
0023          END
```

Program starts with setting DP register to 0 for page zero accessing. Then the IRQ was enabled.

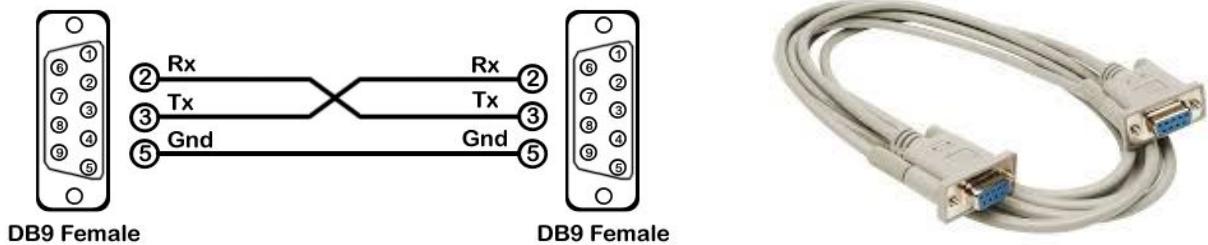
Main loop will poll the variable TICK and compare it, until reaches 100, then clear TICK to 0. The byte 0 in the page zero will be incremented, then write it to location \$8000.

Try enter the hex code from address 200 to 218.

What is happening at the gpio1 LED? Can you change the counting rate? How?

RS232C PORT

The RS232C port is for serial communication. We can use a cross cable or null MODEM cable to connect between the kit and terminal. The connector for both sides are DB9 female. We may build it or buying from computer stores.



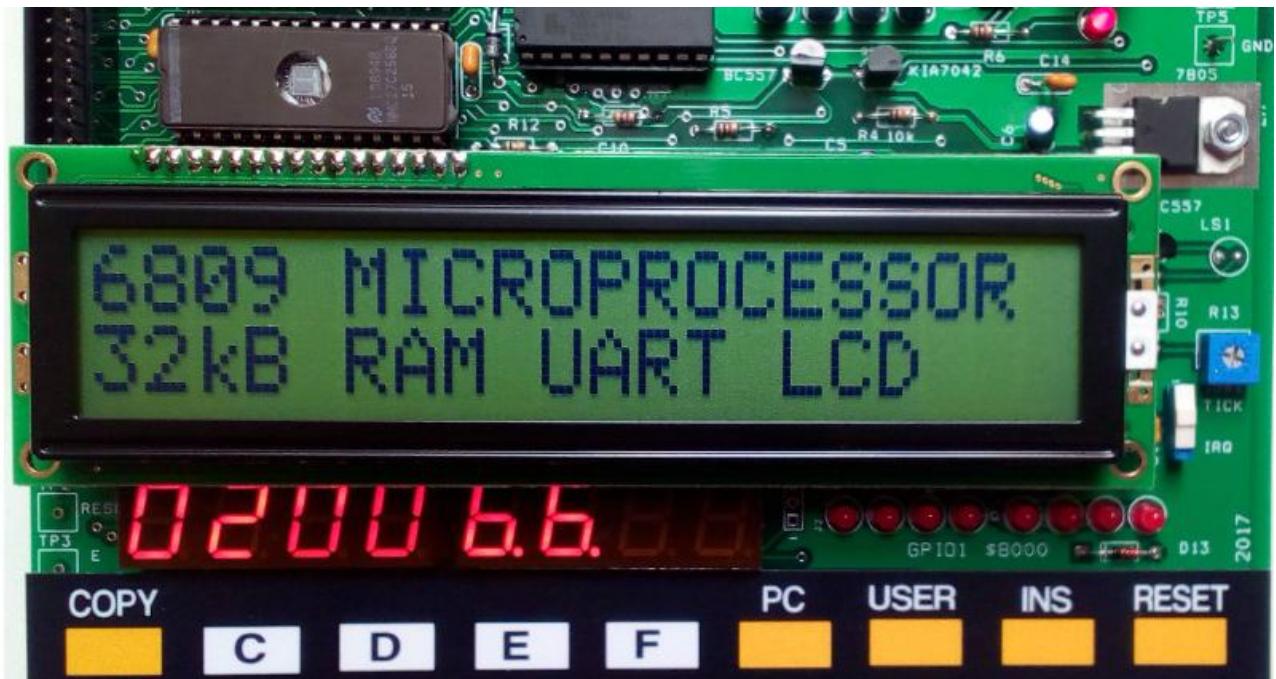
For new PC or laptop computer that have only USB port, we may use the USB to RS232 converter.

The example is CH340 USB 2.0 to RS232 COM Port Serial PDA 9 pin DB9 Wire Cable Adapter. Before we can use it, the hardware driver must be installed. Windows will assign the COM port number. On the terminal software, we can choose that port then.



CONNECTING LCD MODULE

JR1 is 16-pin header for connecting the LCD module. The example shows connecting the 20x2 line text LCD module. R12 is a current limit resistor for the back-light. R13 is trimmer POT for contrast adjustment. The LCD module is interfaced to the 6809 bus directly. The command and data registers are located in memory space having address from \$9000 to \$9003.



Important note: insert or remove the LCD module must be done when the kit is powered off. Ensure pin position, left hand is pin 1.

ON power up, the welcome message will be printed automatically.

Any text LCD with HD44780 compatible controller chip can be used.

Test program for the LCD display.

We will call the subroutines that setup the LCD and print string. The locations of these subroutines are shown the monitor program assembly code listing.

```
0001          * CALLING MONITOR C FUNCTION
0002          * DISPLAY TEXT ON LCD
0003
0004 0200          ORG $200
0005
0006 C228          INIT_LCD EQU $C228
0007 C26B          PSTRING EQU $C26B
0008
0009 0200 BD C2 28    MAIN     JSR INIT_LCD
0010
0011 0203 CC 02 0E          LDD #TEXT1
0012 0206 34 06          PSHS D
0013
0014 0208 BD C2 6B          JSR PSTRING
0015
0016 020B 32 62          LEAS 2,S
0017 020D 3F          SWI
0018
0019 020E 48 65 6C 6C 6F 20    TEXT1    FCC "Hello from 6809"
          66 72 6F 6D 20 36
          38 30 39
0020 021D 00          FCB 0
0021
0022          END
```

Try enter the hex code from address 200 to 21D. When completed, press PC then GO.

What is happening? Can you change the message? How?

LOGIC PROBE POWER SUPPLY

The kit provides test points TP4(+5V) and TP5(GND) for using the logic probe. Students may learn digital logic signals with logic probe easily. Tick signal is indicated by D1 LED blinking. Red clip is for +5V and Black clip for GND.



CUSTOMIZING THE MONITOR PROGRAM

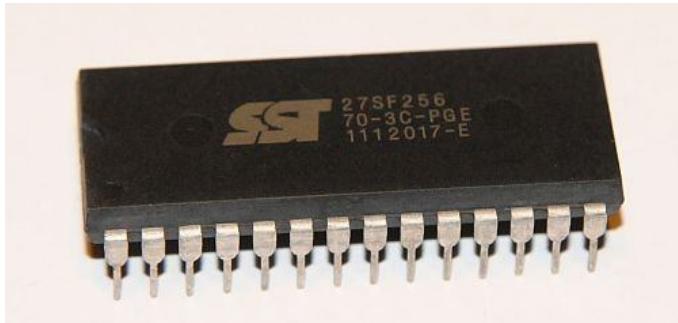
The latest monitor source code is available for download. This section explains how to modify the monitor program, compiling and programming the new flash EPROM.

Tool and parts:

1. MiniPRO Eprom programmer.



2. Flash EPROM, 27SF256, and 28 pin socket. (The 28 pin socket will make stronger the EPROM pins)



Steps for customizing the monitor program.

1. Download the source code and compiler tools from

<http://www.kswichit.com/6809/6809.htm>

2. Edit the source code.

3. Compile the source code.

4. Modify the object file from 64kB to 32kB.

5. Program the object file to the new EPROM.

6. Repeat step 2 if needed.

The compiler is cc09, the small-6809 c compiler V2.02.

Compiling the source code with cc09.

```
Small-6809 C compiler for MSDOS systems V2.02
Adapted to MSDOS and AS6809 by Brian Brown, Nov 1989

Compiling Source file: test4.c
Generating asm file: test4.asm
Assembler file is AS9 compatible

#include startRAM.h
#include startup.h
0 error(s) in compilation.
D:\6809\cc09>cc09 test4.c test4.asm -z
```

Under dos command line,

D:\6809\cc09>cc09 test4.c test4.asm -z

After compiling, the output file will be test4.asm

We will use AS9 to compile the test4.asm into the Motorola s-record.

The screenshot shows a Windows command prompt window titled 'C:\WINDOWS\system32\cmd.exe'. The title bar also displays the path 'C:\WINDOWS\system32\cmd.exe'. The window contains the following text:

```
Small-6809 C compiler for MSDOS systems V2.02
Adapted to MSDOS and AS6809 by Brian Brown, Nov 1989

Compiling Source file: test4.c
Generating asm file: test4.asm
Assembler file is AS9 compatible

#include startRAM.h
#include startup.h
0 error(s) in compilation.
D:\6809\cc09>as9 test4.asm -l > test4.lst
D:\6809\cc09>
```

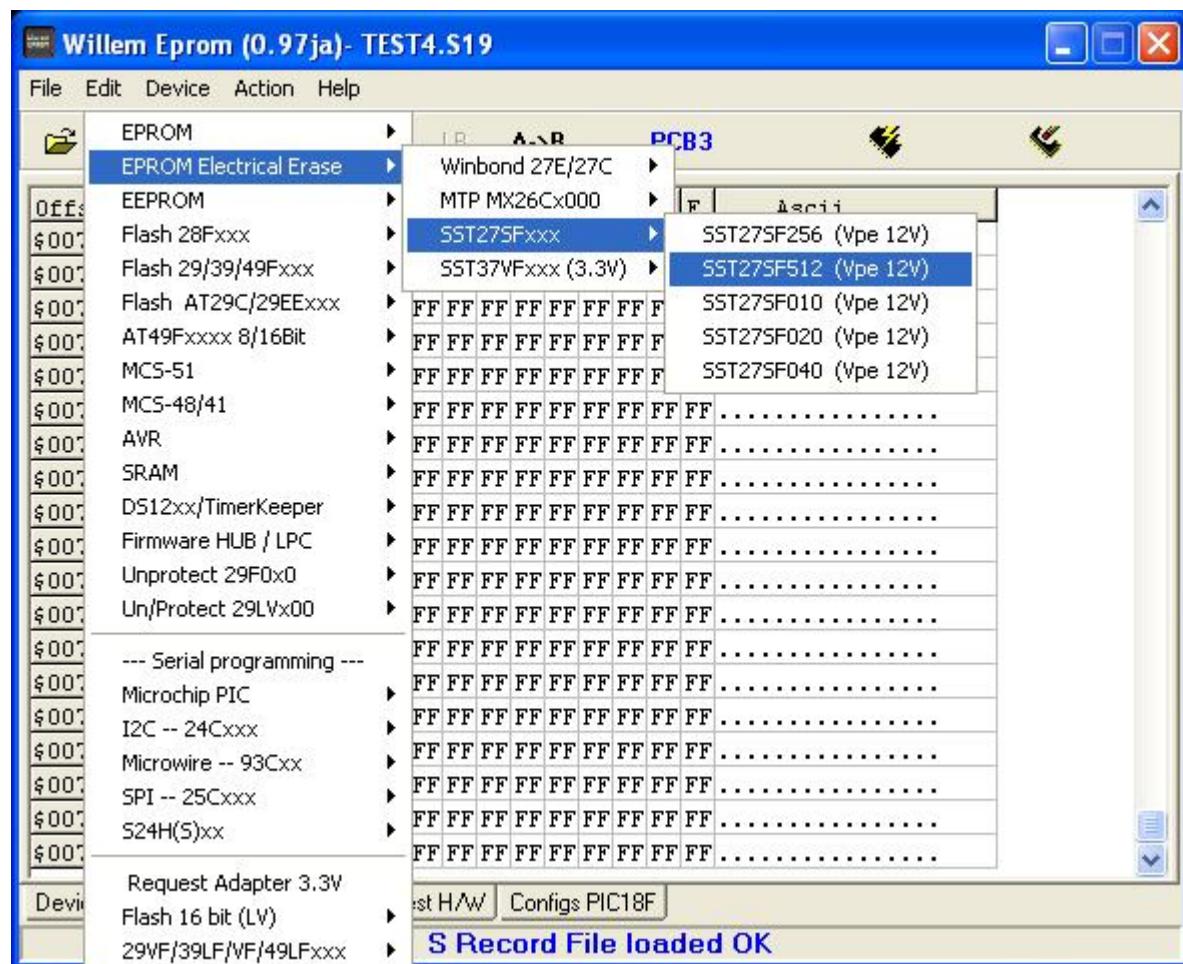
The optional -l will generate the List file.

We will get TEST4.s19, 64kB space from \$0000 to \$FFFF.

To program the 32kB EPROM, we must move the 32kB upper block to the lower block.

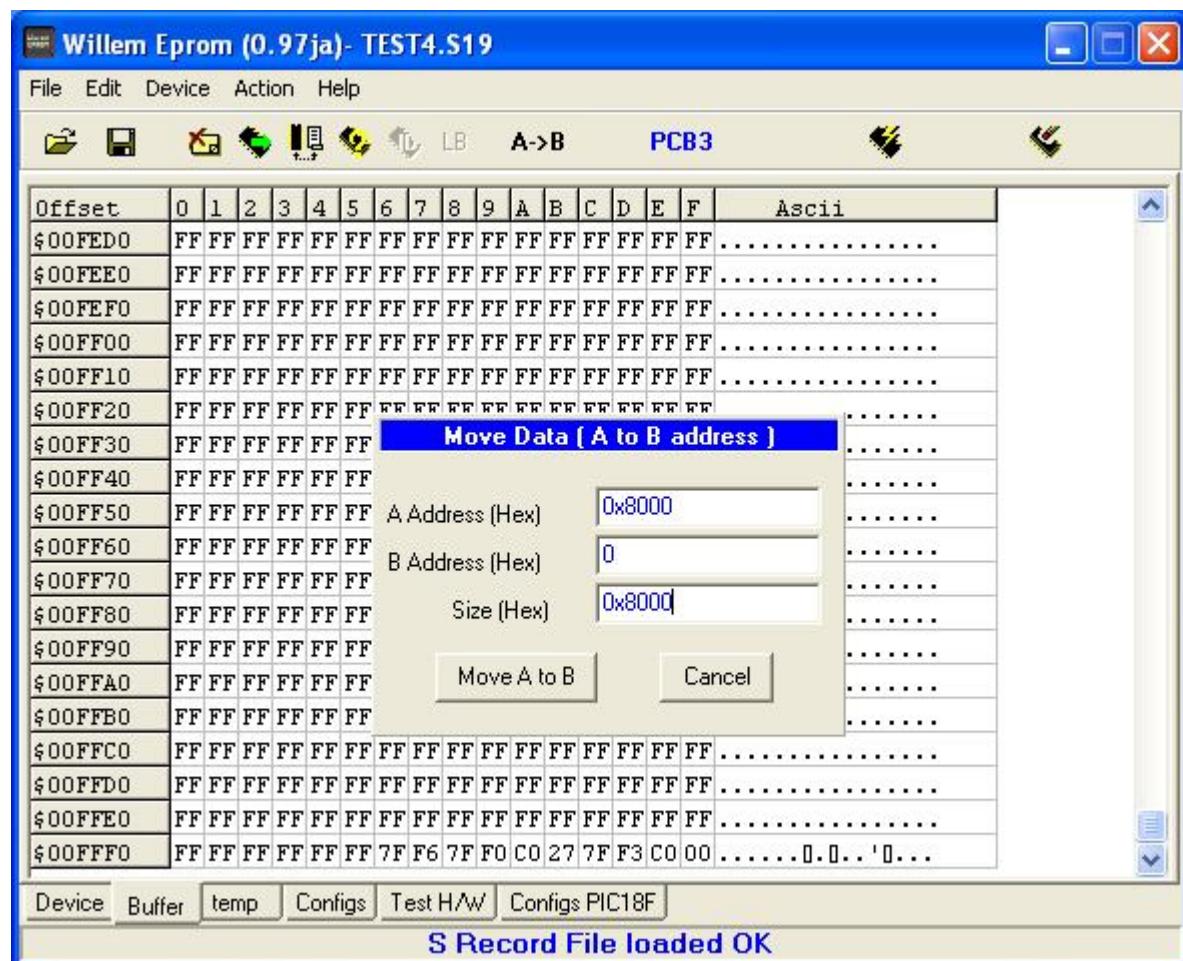
We can use Willem programmer to help modify the object file.

Since the object file is fit for 64kB, so we select the SST27SF512 device.



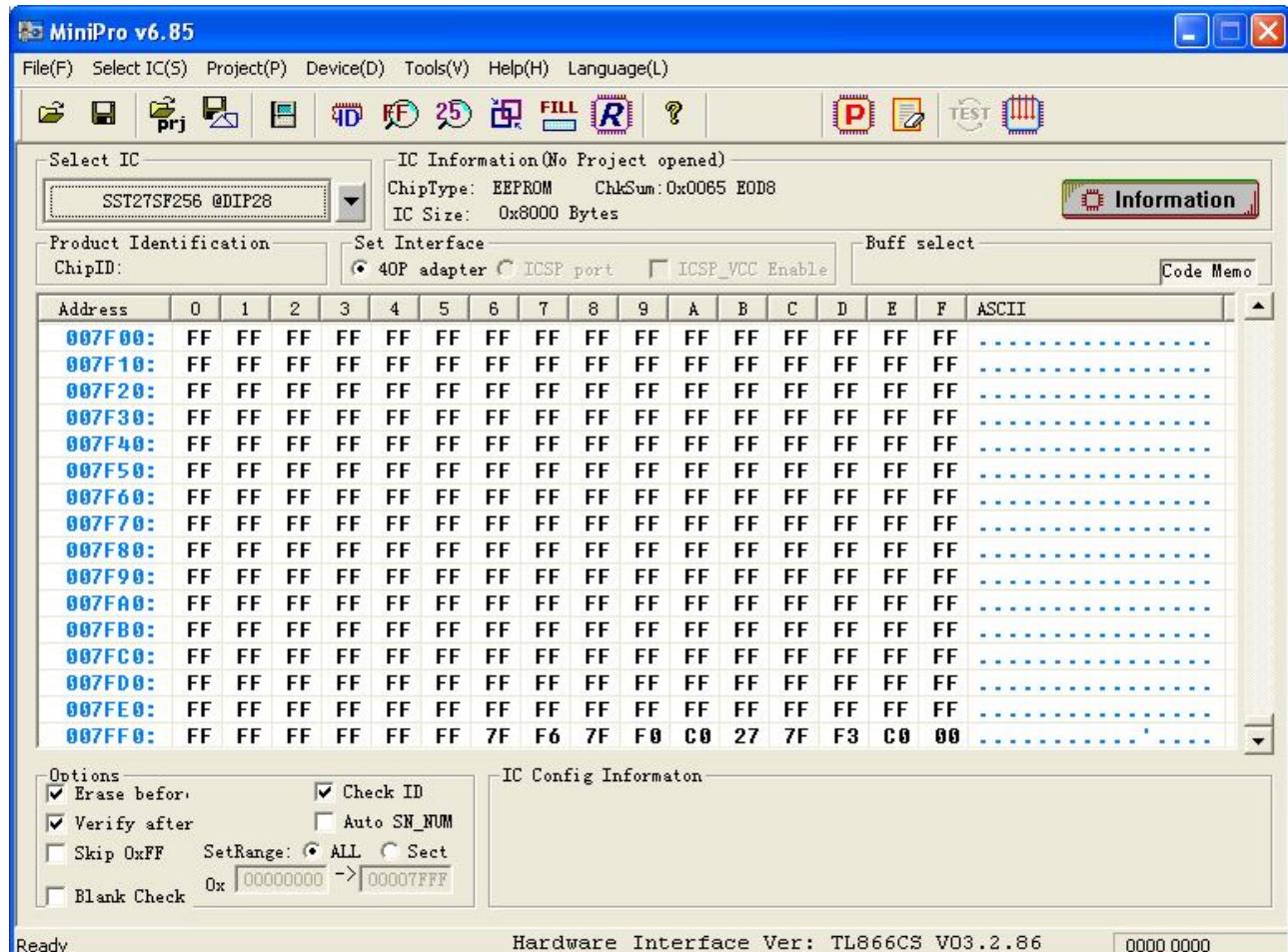
Then LOAD the Motorola s-record, TEST4.s19.

Select tool, A→B, copy the upper 32kB block to lower block.



Then save the hex file as Intel hex file.

Then load the Intel hex file, and use MiniPro to program the flash EPROM, SST27SF256.



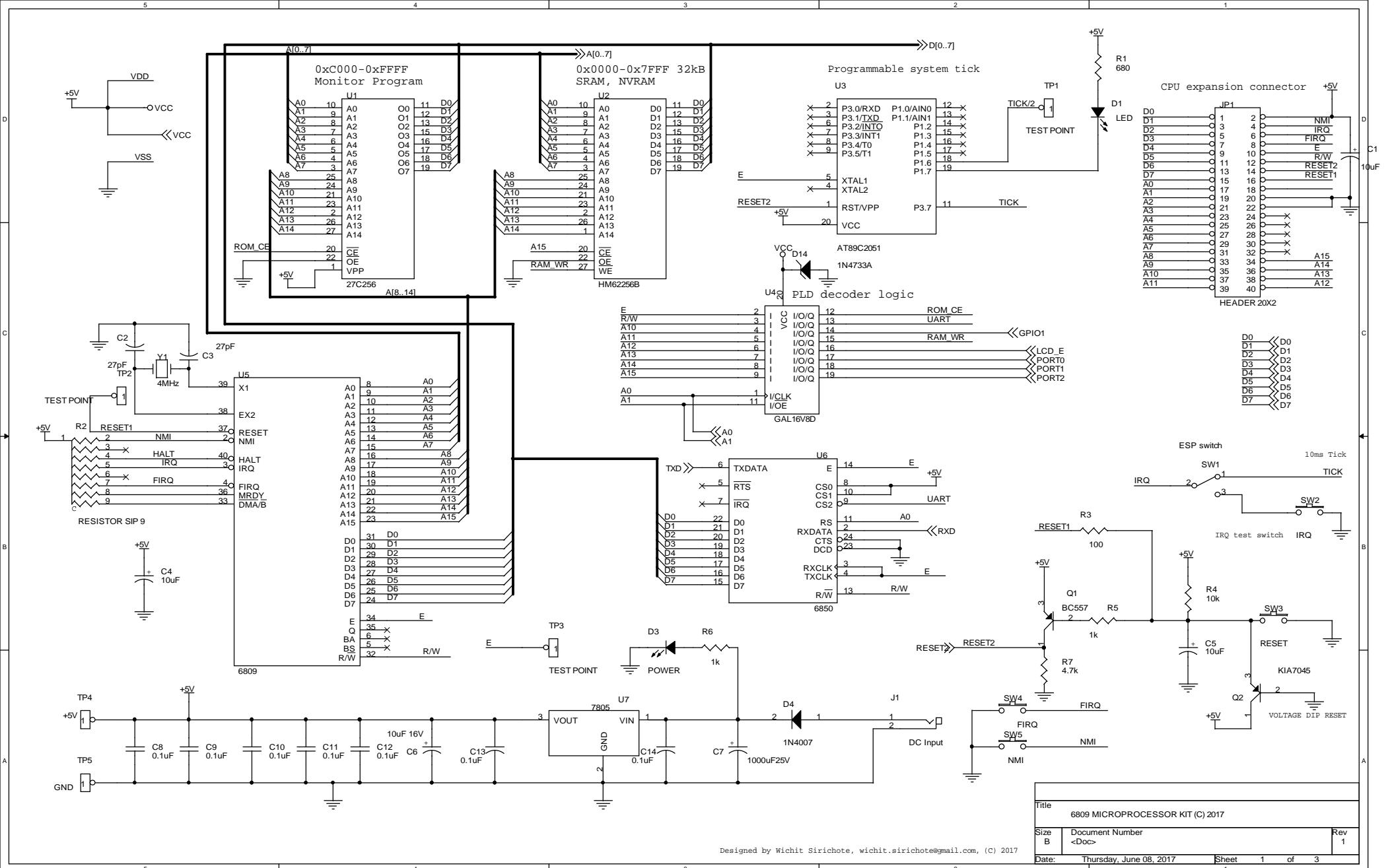
We can see the interrupt vectors for FIRQ, IRQ, SWI and RESTART C000 at FFFE.

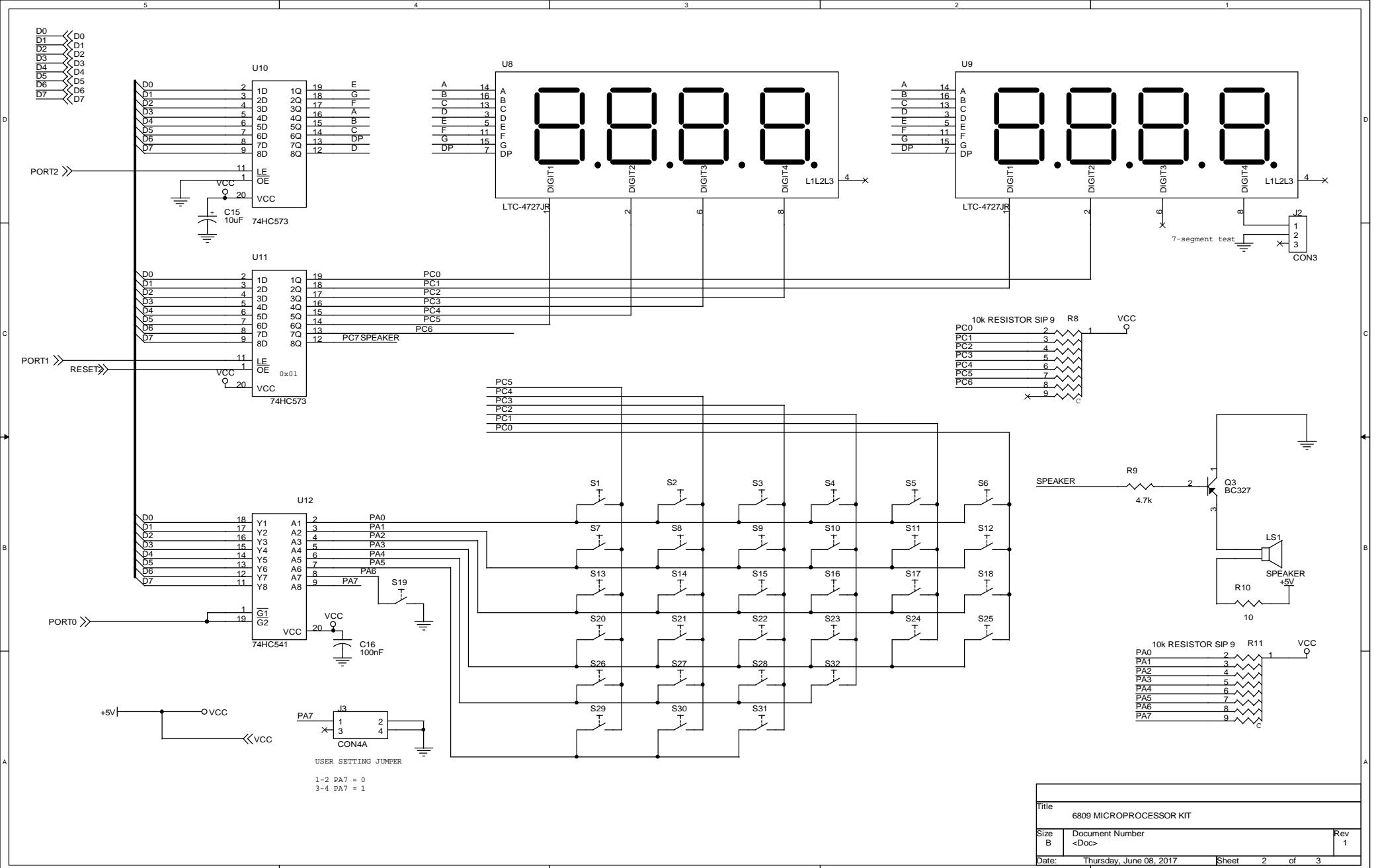
Ensure the kit has no power, then replace the monitor ROM with the new one.

Power up and see the new function.

The flash EPROM is electrically erasable, when program it, the chip will be erased using high voltage then will be programmed with the new hex file automatically.

HARDWARE SCHEMATIC, PARTS LIST





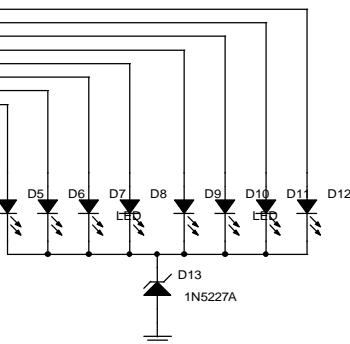
D0
D1
D2
D3
D4
D5
D6
D7

RESET >> GPIO1 >>

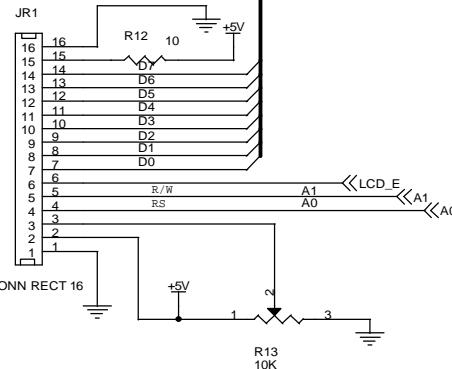
U13

D0
D1
D2
D3
D4
D5
D6
D7
1Q
18
2Q
17
3Q
16
4Q
15
5Q
14
6Q
13
7Q
12
8Q
11
LE
OE
VCC
VCC
20

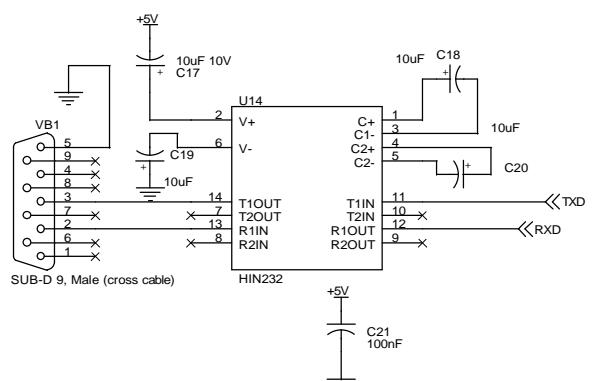
74HC573



HD44780 compatible Text LCD interface



8-bit Binary display LED x8



Title	
6809 MICROPROCESSOR KIT	

Size B	Document Number <Doc>
-----------	--------------------------

Rev
1

Date: Thursday, June 08, 2017

Sheet 3 of 3

PARTS LIST

Semiconductors

U1 27C256, 32kB Eprom
U2 HM62256B, 32kB SRAM
U3 AT89C2051, 8-bit microcontroller
U4 GAL16V8D, PLD
U5 Motorola 68B09, 8-bit microprocessor
U6 Motorola 6850, ACIA chip
U7 7805, voltage regulator
U9,U8 LTC-4727, 7-segment display
U10,U11,U13 74HC573
U12 74HC541
U14 HIN232, RS232 converter
Q1 BC557, PNP transistor
Q2 KIA7042, voltage detector
Q3 BC557 D4 1N4007
D13 1N5227A
D14 1N4733A
D1,D5,D6,D7,D8,D9,D10, LED
D11,D12
D3 3mm POWER LED

Resistors (all resistors are 1/8W +/-5%)

R1 680
R2 RESISTOR SIP 9
R3 100
R13,R4 10k
R6,R5 1k
R9,R7 4.7k
R11,R8 10k RESISTOR SIP 9
R12,R10 5, 20

Capacitors

C1 10uF electrolytic capacitor
C1,C4,C5,C15,C18,C19,C20 10uF
C3,C2 27pF
C6 10uF 16V
C7 1000uF 25V

C8,C9,C10,C11,C12 0.1uF
C13,C14 0.1uF
C21,C16 100nF
C17 10uF 16V

Additional parts

JP1 BLOCK HEADER 20X2
JR1 LCD CONN RECT 16
J1 DC Input
J2 CON3
J3 CON4A
LS1 SPEAKER
SW1 ESP switch
SW2 IRQ
SW3 RESET
SW4 FIRQ
SW5 NMI
S1,S2,S3,S4,S5,S6,S7,S8, SW
PUSHBUTTON
S9,S10,S11,S12,S13,S14,
S15,S16,S17,S18,S19,S20,
S21,S22,S23,S24,S25,S26,
S27,S28,S29,S30,S31,S32
VB1 SUB-D 9, Male (cross cable)
Y1 4.9152MHz XTAL
PCB double side plate through hole
LED color filter
Keyboard sticker printable SVG file

MONITOR PROGRAM LISTINGS

```
1 /*
2  * monitor source code for 6809 MICROPROCRSSOR KIT 2020
3  * compiled with cc09 c compiler and as9 assembler
4  * monitor source code was written by Wichit Sirichote, wichit.sirichote@gm
5  *
6  * cc09>cc09 file.c file.asm -z
7  * cc09>as9 file.asm -l > file.lst
8  *
9  * The object file is Motorola S19 record
10 *
11 * more update and technical information, kswichit.com/6809/6809.htm
12 */
13
14 #include startRAM.h
15
16
17
18 int i;
19 int j;
20 char n;
21 char k;
22 char u,q,o,key;
23 char x;
24 char hit, positive;
25 char flag;
26 char tick;
27 int bcc;
28 int save_bcc, bcc_error;
29
30
31
32 int temp,temp16;
33
34 int PC, save_PC;
35 int num, start, end, desti;
36
37 int t;
38
39 char state;
40
41 int USER_PC,USER_U,USER_X,SAVE_SP;
42 int USER_Y;
43
44 char USER_A,USER_B,USER_P,USER_DP;
45
46 int timeout;
47 char buffer[6];
48
49 char *gpiol;
50 char *port2;
51 char *port1;
52 char *port0;
53 char *dptr;
54 char *dptr2;
55
56
```

```
57
58 #define AS9 1
59 #include startup.h
60
61 #define ACIAPORT 0xA000
62
63 #define LCD_cwr 0x9000
64 #define LCD_dwr 0x9001
65 #define LCD_crd 0x9002
66 #define LCD_drd 0x9003
67
68 #define BUSY 0x80
69
70 #define xon 17
71 #define xoff 19
72
73 #const
74
75 char convert[16] = {0xBD, 0x30, 0x9B, 0xBA, 0x36, 0xAE, 0xAF, 0x38,
76                      0xBF, 0xBE, 0x3F, 0xA7, 0x8D, 0xB3, 0x8F, 0x0F};
77
78 char cold_msg[12] = {0, 0, 0, 0, 0, 0, 0xae, 0xbf, 0xbd, 0xbe, 0, 0};
79
80
81 #code
82
83 /* LCD driver */
84
85 LcdReady()
86 {
87     timeout=0;
88     dptr = LCD_crd;
89
90     while((*dptr&0x80) && (timeout<500))
91         ++timeout;
92 }
93
94 clr_screen()
95 {
96     LcdReady();
97     dptr = LCD_cwr;
98     *dptr=0x01;
99 }
100
101
102 goto_xy(x,y)
103 int x, y;
104 {
105     LcdReady();
106     dptr = LCD_cwr;
107     switch(y)
108     {
109         case 0: *dptr=0x80+x; break;
110         case 1: *dptr=0xC0+x; break;
111         case 2: *dptr=0x94+x; break;
112         case 3: *dptr=0xd4+x; break;
```

```
113      }
114  }
115
116
117  InitLcd( )
118  {
119      LcdReady();
120      dptr = LCD_cwr;
121      *dptr=0x38;
122      LcdReady();
123      dptr = LCD_cwr;
124      *dptr=0x0c;
125      clr_screen();
126      goto_xy(0,0);
127      delay_ms(100);
128  }
129
130
131  PutLCD(str)
132
133  char *str;
134  {
135      char i;
136      for (i=0; str[i] != '\0'; i++)
137      {
138          LcdReady();
139          dptr = LCD_dwr;
140          *dptr=str[i];
141      }
142
143
144  }
145
146  putch_lcd(ch)
147  char ch;
148
149  {
150      LcdReady();
151      dptr= LCD_dwr;
152      *dptr=ch;
153  }
154
155
156
157
158 /* return internal code hex keys and function keys */
159
160  key_code(n)
161  char n;
162  {
163      char d;
164      if(n == 0x16) return 0;
165      if(n == 0x21) return 1;
166      if(n == 0x1b) return 2;
167      if(n == 0x15) return 3;
168      if(n == 0x1c) return 4;
```

```
169     if(n == 0x20) return 5;
170     if(n == 0x1a) return 6;
171     if(n == 0x14) return 7;
172     if(n == 0x22) return 8;
173     if(n == 0x1f) return 9;
174     if(n == 0x19) return 0xa;
175     if(n == 0x13) return 0xb;
176     if(n == 3) return 0xc;
177     if(n == 0x1e) return 0xd;
178     if(n == 0x18) return 0xe;
179     if(n == 0x12) return 0xf;
180
181     if(n == 0xc) return 0x10;
182     if(n == 0xd) return 0x11;
183     if(n == 0xe) return 0x12;
184     if(n == 0xf) return 0x13;
185
186     if(n == 6) return 0x14;
187     if(n == 7) return 0x15;
188     if(n == 8) return 0x16;
189     if(n == 9) return 0x17;
190
191     if(n == 0) return 0x18;
192     if(n == 1) return 0x19;
193     if(n == 2) return 0x1a;
194     if(n == 0x24) return 0x1b;
195
196     if(n == 0x23) return 0x1c;
197     if(n == 0x1d) return 0x1d;
198     if(n == 0x17) return 0x1e;
199     if(n == 0x10) return 0x1f;
200
201
202
203
204
205 }
206
207 delay_num1()
208 {
209     temp=0;
210     temp=0;
211 }
212
213 delay_ms(w)
214 int w;
215 {
216     for( n = 0; n < w; n++)
217         continue;
218
219 }
220
221
222 scan()
223 {
224     char key_pressed;
```

```
225
226     k = 1;
227     u = 0;
228     key = -1;
229     q = 0;
230
231     for(i=0; i<6; i++)
232     {
233         *port1 = ~k;
234
235         *port2 = buffer[i];
236
237         if(buffer[i] != 0x30 && buffer[i] != 0x38 && buffer[i] != 0x70) delay_ms
238         else delay_num1();
239
240         *port2 = 0;
241
242         delay_ms(1);
243
244         o= *port0;
245
246         for(n=0; n<6; n++)
247         {
248             if((o&1)==0)
249             {key=q;
250
251
252             }
253
254
255             else q++;
256             o = o >> 1;
257         }
258
259         k = k << 1;
260     }
261
262     o = *port0;
263
264     if((o&0x40)==0) key=0x24;
265
266     key_pressed=key;
267
268     return key_pressed;
269
270 }
271
272
273 dot_address()
274 {
275     buffer[0]=buffer[0]&~0x40;
276     buffer[1]=buffer[1]&~0x40;
277
278
279     buffer[2]=buffer[2]|0x40;
280     buffer[3]=buffer[3]|0x40;
```

```
281     buffer[4]=buffer[4]|0x40;
282     buffer[5]=buffer[5]|0x40;
283 }
284
285
286
287 dot_data()
288 {
289     buffer[0]=buffer[0]|0x40;
290     buffer[1]=buffer[1]|0x40;
291
292     buffer[2]=buffer[2]&~0x40;
293     buffer[3]=buffer[3]&~0x40;
294     buffer[4]=buffer[4]&~0x40;
295     buffer[5]=buffer[5]&~0x40;
296
297 }
298
299
300 hex4(h)
301 int h;
302 {
303     temp16 = h;
304     buffer[2]= convert[temp16&0xf];
305     temp16>>=4;
306     buffer[3]= convert[temp16&0xf];
307     temp16>>=4;
308     buffer[4]=convert[temp16&0xf];
309     temp16>>=4;
310     buffer[5]=convert[temp16&0xf];
311 }
312
313
314 address_display()
315 {
316     temp16 = PC;
317     hex4(temp16);
318
319 }
320
321
322
323
324 data_display()
325 {
326     dptr =PC;
327
328     n = *dptr;
329
330     buffer[0]= convert[n&0xf];
331     n = n>>4;
332     buffer[1]=convert[n&0xf];
333     dot_data();
334
335 }
336
```

```
337     read_memory( )
338     {
339         address_display();
340         data_display();
341     }
342
343
344     key_address( )
345     {
346         state = 1;
347
348         read_memory( );
349         dot_address( );
350         hit=0;
351
352     }
353
354
355     key_data( )
356     {
357
358         read_memory( );
359         dot_data();
360         hit=0;
361         state=2;
362
363     }
364
365     key_plus( )
366     {
367
368         if(state==1 || state==2)
369         {
370             PC++;
371             read_memory( );
372             key_data();
373         }
374         if(state==4)
375         {
376             start=num;
377             hit =0;
378             positive=1;
379
380         }
381
382
383         if(state==5)
384         {
385
386             state=6;
387             start = num;
388             hit=0;
389             buffer[0]=0x8f; /* end cursor */
390             return;
391
392         }
```

```
393
394
395     if(state==6)
396     {
397
398         state=7;
399         end = num;
400         hit=0;
401         buffer[0]=0xb3; /* destination cursor */
402
403         if(end <= start) print_error();
404
405     }
406
407 }
408
409
410
411 key_minus()
412 {
413     if(state==1 | state ==2)
414     {
415         PC--;
416         read_memory();
417         key_data();
418     }
419
420     if(state==4)
421     {
422         start=num;
423         hit =0;
424         positive=0;
425     }
426 }
427
428 data_hex()
429 {
430
431     dptr = PC;
432     x = *dptr;
433     if(hit==0) x=0;
434     {
435         hit =1;
436         x = x << 4;
437         x = x|key;
438
439         *dptr = x;
440
441         read_memory();
442
443         dot_data();
444     }
445 }
446
447 key_PC()
448 {
```

```
449     PC=save_PC;
450     key_data();
451 }
452
453 hex_address()
454 {
455     if(hit==0) PC=0;
456     {
457         hit=1;
458
459         PC<<=4;
460         PC |= key;
461         read_memory();
462         dot_address();
463     }
464 }
465
466 print_error()
467 {
468
469     buffer[5]= 0x8f;
470     buffer[4]= 3;
471     buffer[3]=3;
472     buffer[2]=0;
473     buffer[1]=0;
474     buffer[0]=0;
475     state=0;
476 }
477
478
479 key_go(){
480
481     if(state==1 || state==2)
482     {
483 #asm
484         STS SAVE_SP
485         LDU USER_U
486
487         LDD PC
488         PSHS D
489
490         LDA USER_P
491         TFR A,CC
492         LDA USER_DP
493         TFR A,DP
494
495         LDX USER_X
496         LDY USER_Y
497         LDB USER_B
498         LDA USER_A
499         RTS
500
501 #endasm
502     }
503
504     if(state==4)
```

```
505     {
506
507     desti = num;
508
509     if(positive==0) start= start-desti;
510     else start = start+desti;
511
512     hex4(start);
513     hit=0;
514
515 }
516
517 if(state==7)
518 {
519     desti = num;
520     temp = end-start;
521     dptr = start;
522     dptr2 = desti;
523
524     for(i=0; i<temp; i++)
525     {
526         *(dptr2+i)=*(dptr+i);
527     }
528     PC = desti;
529     read_memory();
530     dot_data();
531     state=2;
532
533
534
535 }
536
537
538 }
539
540
541 key_reg()
542 {
543     buffer[7]=0;
544     buffer[6]=0;
545     buffer[5]= 0x03;
546     buffer[4]= 0x8F;
547     buffer[3]= 0xad;
548     buffer[2]=0;
549     buffer[1]=0;
550     buffer[0]=0;
551
552     state = 3; /* register display state = 3 with hex key */
553
554 }
555
556
557 acca()
558 {
559     n = USER_A;
```

```
561     buffer[2]= convert[n&0xf];
562     n = n>>4;
563     buffer[3]=convert[n&0xf];
564     buffer[4]=0;
565     buffer[5]=0;
566     buffer[1]=0;
567     buffer[0]=0x3f;
568 }
569
570 accb()
571 {
572
573     n = USER_B;
574
575     buffer[2]= convert[n&0xf];
576     n = n>>4;
577     buffer[3]=convert[n&0xf];
578     buffer[4]=0;
579     buffer[5]=0;
580     buffer[1]=0;
581     buffer[0]=0xa7;
582 }
583
584 ab()
585 {
586
587     n = USER_B;
588
589     buffer[2]= convert[n&0xf];
590     n = n>>4;
591     buffer[3]=convert[n&0xf];
592
593     n = USER_A;
594     buffer[4]= convert[n&0xf];
595     n = n>>4;
596     buffer[5]=convert[n&0xf];
597
598
599     buffer[1]=0x3f;
600     buffer[0]=0xa7;
601 }
602
603
604 reg_x()
605 {
606     temp16 = USER_X;
607
608     hex4(temp16);
609
610
611     buffer[1] = 0;
612     buffer[0] = 0x13;
613
614 }
615
616 reg_y()
```

```
617 {  
618     temp16 = USER_Y;  
619     hex4(temp16);  
620  
621  
622     buffer[1] = 0;  
623     buffer[0] = 0xb6;  
624 }  
625  
626  
627 reg_u()  
{  
628     temp16 = USER_U;  
629     hex4(temp16);  
630  
631  
632     buffer[1] = 0;  
633     buffer[0] = 0xb5;  
634 }  
635 }  
636  
637  
638 reg_s()  
{  
639  
640     temp16 = SAVE_SP;  
641     hex4(temp16);  
642  
643  
644  
645     buffer[1] = 0;  
646     buffer[0] = 0xae;  
647 }  
648  
649 reg_dp()  
{  
650  
651     n = USER_DP;  
652  
653  
654     buffer[2]= convert[n&0xf];  
655     n = n>>4;  
656     buffer[3]=convert[n&0xf];  
657     buffer[4]=0;  
658     buffer[5]=0;  
659     buffer[1]=0xb3;  
660     buffer[0]=0x1F;  
661 }  
662  
663 low_cc()  
{  
664  
665     n = USER_P;  
666     if(n&1) buffer[2]=0x30; else buffer[2]=0xbd;  
667     if(n&2) buffer[3]=0x30; else buffer[3]=0xbd;  
668     if(n&4) buffer[4]=0x30; else buffer[4]=0xbd;  
669     if(n&8) buffer[5]=0x30; else buffer[5]=0xbd;  
670  
671     buffer[1]=0x8d;  
672     buffer[0]=0x85;
```

```
673     }
674
675     hi_cc()
676     {
677         n = USER_P;
678         if(n&0x10) buffer[2]=0x30; else buffer[2]=0xbd;
679         if(n&0x20) buffer[3]=0x30; else buffer[3]=0xbd;
680         if(n&0x40) buffer[4]=0x30; else buffer[4]=0xbd;
681         if(n&0x80) buffer[5]=0x30; else buffer[5]=0xbd;
682
683         buffer[1]=0x8d;
684         buffer[0]=0x37;
685
686     }
687
688
689
690     reg_display()
691     {
692
693         switch(key)
694         {
695             case 0: acca(); break;
696             case 1: accb(); break;
697             case 2: ab(); break;
698             case 6: reg_x(); break;
699             case 7: reg_y(); break;
700             case 9: reg_u(); break;
701             case 10: reg_s(); break;
702             case 8: reg_dp(); break;
703             case 5: low_cc(); break;
704             case 4: hi_cc(); break;
705         }
706     }
707
708     /* insert byte and shift 512 bytes down */
709
710     insert()
711     {
712         if(state==1 || state==2)
713         {
714             dptr=PC;
715             for(j=512; j>0; j--)
716             {
717                 *(dptr+j)=*(dptr+j-1);
718             }
719             *(dptr+1)=0; /* insert next byte */
720             PC++;
721             read_memory();
722             state=2;
723         }
724     }
725
726
727     /* delete current byte and shift 512 bytes up */
728
```

```
729 cut_byte()
730 {
731     if(state==1 || state==2)
732     {
733         dptr=PC;
734         for(j=0; j<512; j++)
735         {
736             *(dptr+j)=*(dptr+j+1);
737         }
738         read_memory();
739         state=2;
740     }
741 }
742 }
743
744 key_test()
745 {
746     /* clear I bit to enable irq */
747     #asm
748     andcc #$ef
749     #endasm
750     t=0;
751     buffer[0]=0;buffer[1]=0;
752
753     while(1)
754     {
755         while(tick<10)
756             scan();
757         tick=0;
758         hex4(t);
759         *gpiol = t;
760         t++;
761
762
763     }
764 }
765
766 }
767
768 clear_buffer()
769 {
770     for(i=0; i<6; i++)
771         *(buffer+i)=0;
772 }
773
774 key_cal()
775 {
776     state = 4;
777
778     clear_buffer();
779     buffer[2]= 0xbd;
780     start=0;
781     desti=0;
782     /*buffer[1] = 0x8d;
783     buffer[0] = 0x85;
784     */
```

```
785     hit = 0;
786
787 }
788
789
790
791 enter_num( )
792 {
793     if(hit==0) num=0;
794     {
795         hit=1;
796
797         num<<=4;
798         num |= key;
799         hex4(num);
800     }
801 }
802
803
804 key_copy( )
805 {
806     state=5;
807     hit=0;
808     clear_buffer();
809     buffer[2]= 0xbd;
810
811     buffer[0]=0xae;
812     buffer[1]=0;
813 }
814
815
816
817
818 key_exe( )
819 {
820
821     if(flag==0) beep();
822
823
824     if( key>15 )
825     {
826
827         switch(key)
828         {
829             case 0x13: key_address(); break;
830             case 0x12: key_data(); break;
831             case 0x17: key_plus(); break;
832             case 0x16: key_minus(); break;
833             case 0x10: key_PC(); break;
834             case 0x1b: key_go(); break;
835             case 0x11: key_reg(); break;
836             case 0x18: insert(); break;
837             case 0x19: cut_byte(); break;
838             case 0x15: flag = flag^1; break;
839             case 0x1a: key_test(); break;
840             case 0x1d: key_cal(); break;
```

```
841     case 0x1c: key_copy(); break;
842     case 0x1e: key_dump(); break;
843     case 0x1f: key_load(); break;
844
845 /*  case 0x14: key_user(); break; available for user key */
846
847 }
848 }
849
850 else
851 {
852
853     switch(state)
854     {
855         case 1: hex_address(); break;
856         case 2: data_hex(); break;
857         case 3: reg_display(); break;
858         case 4: enter_num(); break;
859         case 5: enter_num(); break;
860         case 6: enter_num(); break;
861         case 7: enter_num(); break;
862
863     }
864
865 }
866
867 }
868
869
870 delay_beep()
871 {
872     for(j=0; j<2; j++)
873         continue;
874 }
875
876
877 beep()
878 {
879     char x;
880
881     *port2=0;
882
883     for(x=0; x<60; x++)
884     {
885         *port1 = ~0x80;
886         delay_beep();
887         *port1 = 0xff;
888         delay_beep();
889     }
890
891 }
```

```
897
898     scan1()
899     {
900         while( scan() != -1)
901             continue;
902         delay_ms(3);
903
904         while(scan() == -1)
905             continue;
906         delay_ms(3);
907
908         key = scan();
909
910         key = key_code(key);
911
912         key_exe();
913
914
915
916     }
917
918
919 /* ACIA with 4.9152MHz CPU XTAL, bit rate is 19200, 8n1 */
920
921 initacia()
922 {
923     char dummy, *acia, reset, baudrate;
924     acia = ACIAPORT;
925     baudrate = 0x16;
926     reset = 3;
927     *(acia) = reset;
928     *(acia) = baudrate;
929     dummy = *(acia+1);           /*clear RBR*/
930 }
931
932 putchar(ch)
933 char ch;
934 {
935     char *acia;
936     acia = ACIAPORT;
937     while( ((*acia) & 0x02) == 0)      /* wait on TDRE */
938         continue;
939     *(acia+1) = ch;
940 }
941
942
943 puts(s)
944 char *s;
945 {
946     while( *s ) {
947         putchar(*s);
948         s++;
949     }
950
951 }
```

```
953     getchar( )
954     {
955         char ch;
956         char *acia;
957
958         acia=ACIAPORT;
959         while((*acia&1)==0)
960             continue;
961
962         ch = *(acia+1);           /*read 0xA001*/
963
964         return ch;
965     }
966
967     newline( )
968     {
969         putchar(0x0a);
970         putchar(0x0d);
971     }
972
973     send_hex(n)
974     char n;
975     {
976         k = n>>4;
977         k = k&0xf;
978
979         if (k>9) putchar(k+0x37); else putchar(k+0x30);
980         k= n&0xf;
981         if (k>9) putchar(k+0x37); else putchar(k+0x30);
982     }
983
984     send_word_hex(n)
985     int n;
986     {
987         temp16 = n>>8;
988         k = temp16&0xff;
989         send_hex(k);
990         k = n&0xff;
991         send_hex(k);
992     }
993
994
995
996     key_dump( )
997     {
998         int j,p;
999
1000        dptr = PC;
1001
1002        for(j=0; j<16; j++)
1003        {
1004            newline();
1005            send_word_hex(dptr);
1006            putchar(':');
1007            for(p=0; p<16; p++)
1008            {
```

```
1009         send_hex(* (dptr+p));
1010         putchar(0x20);
1011     }
1012
1013     putchar(0x20);
1014
1015     for (p=0; p<16; p++)
1016     {
1017         q=*(dptr+p);
1018         if(q >= 0x20 && q < 0x80) putchar(q);
1019         else putchar('.');
1020     }
1021
1022 }
1023
1024
1025     dptr+=16;
1026 }
1027 newline();
1028 PC = dptr;
1029 key_address();
1030 }
1031
1032 nibble2hex(c)
1033 char c;
1034 {
1035     char n;
1036     if(c<0x40) return (c-0x30);
1037     else return (c-0x37);
1038 }
1039
1040 gethex()
1041 {
1042     int a,b;
1043
1044     a = getchar2();
1045     b = getchar2();
1046
1047     a = nibble2hex(a)<<4;
1048     b = nibble2hex(b);
1049     a = a|b;
1050     bcc = bcc+a; /* compute check sum */
1051
1052     return (a);
1053 }
1054
1055 get16bitaddress()
1056 {
1057     int load_address;
1058
1059     load_address =0;
1060
1061     load_address |= gethex();
1062     load_address <<=8;
1063     load_address |= gethex();
1064 }
```

```
1065     return load_address;
1066 }
1067
1068
1069 read_record1()
1070 {
1071     char x;
1072     char byte_count;
1073
1074     int address16bit;
1075
1076     bcc =0;
1077
1078     byte_count = gethex()-3; /* only data record */
1079
1080     address16bit = get16bitaddress();
1081
1082     dptr = address16bit;
1083     for(x=0; x<byte_count; x++)
1084     {
1085         *(dptr+x) = gethex();
1086     }
1087
1088     bcc = ~bcc; /* one's complement */
1089
1090
1091
1092     save_bcc= bcc&0xff;
1093
1094     if(save_bcc != gethex()) bcc_error=1;
1095
1096 }
1097
1098
1099 get_s_record()
1100 {
1101     end =0;
1102     bcc_error=0;
1103
1104
1105     while(end==0)
1106     {
1107
1108
1109     while(getchar2() != 'S')
1110         continue;
1111
1112         switch(getchar2()) /* get record type */
1113     {
1114         case '0': end=0; break;
1115         case '1': read_record1(); break;
1116         case '5': end=1; break;
1117         case '9': end=1; break;
1118     }
1119 }
1120
```

```
1121     newline();
1122     if(bcc_error) puts("check sum errors!");
1123     else puts("0 error...");
1124
1125     key_data();
1126 }
1127
1128 key_load()
1129 {
1130     newline();
1131     puts("Load Motorola s-record (set lms delay, character & line)");
1132     get_s_record(); /* accept only S1 record */
1133
1134 }
1135
1136
1137
1138 initreg()
1139 {
1140     PC = 0x200;
1141     save_PC = 0x200;
1142     USER_U = 0x7F00;
1143     USER_DP = 0; /* page zero 6802 compatible */
1144
1145
1146     #asm
1147         TFR CC,A
1148         STA USER_P
1149     #endasm
1150
1151     gpiol = 0x8000;
1152     port2 = 0x8003;
1153     port1 = 0x8002;
1154     port0 = 0x8001;
1155
1156 }
1157
1158 waitls()
1159 {
1160     #asm
1161     andcc #$ef
1162     #endasm
1163     while(tick<100);
1164     tick=0;
1165 }
1166
1167 /* hardware flow control with RTS/CTS */
1168
1169 getchar2()
1170 {
1171     char ch;
1172     char *acia;
1173
1174     acia=ACIAPORT;
1175
1176     *acia = 0x16; /* enable receiving */
```

```
1177
1178     while((*acia&1)==0)
1179     ;
1180
1181     *acia = 0x56; /* stop sending */
1182
1183
1184     ch = *(acia+1);           /*read 0xA001*/
1185
1186     *gpiol=ch;
1187
1188     return ch;
1189 }
1190
1191
1192 main()
1193 {
1194     initreg();
1195
1196
1197     *gpiol=0;
1198
1199     *port2=0;
1200     *port1=0xff;
1201
1202     flag=0;
1203
1204     initacia();
1205
1206     newline();
1207     puts("6809 MICROPROCESSOR KIT 2020");
1208
1209     InitLcd();
1210     InitLcd();
1211
1212     PutLCD("6809 MICROPROCESSOR");
1213     goto_xy(0,1);
1214     PutLCD("32kB RAM UART LCD");
1215
1216     buffer[5]= convert[6];
1217     buffer[4]= convert[8];
1218     buffer[3]= convert[0];
1219     buffer[2]= convert[9];
1220     buffer[1]=0;
1221     buffer[0]=0;
1222
1223
1224     while(1)
1225     {
1226         scan1();
1227     }
1228
1229 }
1230
1231
1232
```

1233
1234
1235
1236
1237

```
1 0001
2 0002 * small-c V2.3
3 0003 7000 ORG $7000
4 0004
5 0005 * .global i
6 0006 i:
7 0007 7000 RMB 2
8 0008
9 0009 * .global j
10 0010 j:
11 0011 7002 RMB 2
12 0012
13 0013 * .global n
14 0014 n:
15 0015 7004 RMB 1
16 0016
17 0017 * .global k
18 0018 k:
19 0019 7005 RMB 1
20 0020
21 0021 * .global u
22 0022 u:
23 0023 7006 RMB 1
24 0024
25 0025 * .global q
26 0026 q:
27 0027 7007 RMB 1
28 0028
29 0029 * .global o
30 0030 o:
31 0031 7008 RMB 1
32 0032
33 0033 * .global key
34 0034 key:
35 0035 7009 RMB 1
36 0036
37 0037 * .global x
38 0038 x:
39 0039 700a RMB 1
40 0040
41 0041 * .global hit
42 0042 hit:
43 0043 700b RMB 1
44 0044
45 0045 * .global positive
46 0046 positive:
47 0047 700c RMB 1
48 0048
49 0049 * .global flag
50 0050 flag:
51 0051 700d RMB 1
52 0052
53 0053 * .global tick
54 0054 tick:
55 0055 700e RMB 1
56 0056
```

```
57 0057          * .global bcc
58 0058          bcc:
59 0059 700f      RMB 2
60 0060
61 0061          * .global save_bcc
62 0062          save_bcc:
63 0063 7011      RMB 2
64 0064
65 0065          * .global bcc_errno
66 0066          bcc_errno:
67 0067 7013      RMB 2
68 0068
69 0069          * .global temp
70 0070          temp:
71 0071 7015      RMB 2
72 0072
73 0073          * .global temp16
74 0074          temp16:
75 0075 7017      RMB 2
76 0076
77 0077          * .global PC
78 0078          PC:
79 0079 7019      RMB 2
80 0080
81 0081          * .global save_PC
82 0082          save_PC:
83 0083 701b      RMB 2
84 0084
85 0085          * .global num
86 0086          num:
87 0087 701d      RMB 2
88 0088
89 0089          * .global start
90 0090          start:
91 0091 701f      RMB 2
92 0092
93 0093          * .global end
94 0094          end:
95 0095 7021      RMB 2
96 0096
97 0097          * .global desti
98 0098          desti:
99 0099 7023      RMB 2
100 0100
101 0101          * .global t
102 0102          t:
103 0103 7025      RMB 2
104 0104
105 0105          * .global state
106 0106          state:
107 0107 7027      RMB 1
108 0108
109 0109          * .global USER_PC
110 0110          USER_PC:
111 0111 7028      RMB 2
112 0112
```

```
113 0113 * .global USER_U
114 0114 USER_U:
115 0115 702a RMB 2
116 0116
117 0117 * .global USER_X
118 0118 USER_X:
119 0119 702c RMB 2
120 0120
121 0121 * .global SAVE_SP
122 0122 SAVE_SP:
123 0123 702e RMB 2
124 0124
125 0125 * .global USER_Y
126 0126 USER_Y:
127 0127 7030 RMB 2
128 0128
129 0129 * .global USER_A
130 0130 USER_A:
131 0131 7032 RMB 1
132 0132
133 0133 * .global USER_B
134 0134 USER_B:
135 0135 7033 RMB 1
136 0136
137 0137 * .global USER_P
138 0138 USER_P:
139 0139 7034 RMB 1
140 0140
141 0141 * .global USER_DP
142 0142 USER_DP:
143 0143 7035 RMB 1
144 0144
145 0145 * .global timeout
146 0146 timeout:
147 0147 7036 RMB 2
148 0148
149 0149 * .global buffer
150 0150 buffer:
151 0151 7038 RMB 6
152 0152
153 0153 * .global gpiol
154 0154 gpiol:
155 0155 703e 00 00 FDB 0
156 0156
157 0157 * .global port2
158 0158 port2:
159 0159 7040 00 00 FDB 0
160 0160
161 0161 * .global port1
162 0162 port1:
163 0163 7042 00 00 FDB 0
164 0164
165 0165 * .global port0
166 0166 port0:
167 0167 7044 00 00 FDB 0
168 0168
```

```
169 0169          * .global dptr
170 0170          dptr:
171 0171 7046 00 00      FDB 0
172 0172
173 0173          * .global dptr2
174 0174          dptr2:
175 0175 7048 00 00      FDB 0
176 0176 704a          ADR1: RMB 2
177 0177 704c          ADR2: RMB 2
178 0178 704e          ADR3: RMB 1
179 0179 704f          ADR31: RMB 3
180 0180 fffe          ORG $ffffe
181 0181 fffe c0 00      FDB $c000
182 0182 fff8          org $fff8
183 0183          *   fdb irq_serv
184 0184 fff8 7f f0      fdb $7ff0      use ram vector
185 0185 fffc          org $fffc
186 0186 fffc 7f f3      fdb $7ff3      nmi vector
187 0187 fff6          org $fff6
188 0188 fff6 7f f6      fdb $7ff6      firq vector
189 0189 fffa          org $fffa
190 0190 fffa c0 27      fdb swi_serv
191 0191 c000          ORG $c000
192 0192 c000 10 ce 7f f0      lds #$7ff0
193 0193 c004 86 7e      lda #$7e      store jmp instruction
194 0194 c006 b7 7f f0      sta $7ff0
195 0195 c009 8e c0 23      ldx #irq_serv
196 0196 c00c bf 7f f1      stx $7ff1
197 0197 c00f 86 3b      lda #$3b      store rti instruction
198 0198 c011 b7 7f f3      sta $7ff3
199 0199 c014 b7 7f f6      sta $7ff6
200 0200 c017 8e 00 00      ldx #0
201 0201 c01a bf 7f f4      stx $7ff4
202 0202 c01d bf 7f f7      stx $7ff7
203 0203 c020 7e e8 10      jmp main
204 0204          irq_serv
205 0205 c023 7c 70 0e      inc tick
206 0206 c026 3b          rti
207 0207          swi_serv
208 0208 c027 35 02      PULS A
209 0209 c029 b7 70 34      STA USER_P
210 0210 c02c 35 02      PULS A
211 0211 c02e b7 70 32      STA USER_A
212 0212 c031 35 02      PULS A
213 0213 c033 b7 70 33      STA USER_B
214 0214 c036 35 02      PULS A
215 0215 c038 b7 70 35      STA USER_DP
216 0216 c03b 35 10      PULS X
217 0217 c03d bf 70 2c      STX USER_X
218 0218 c040 35 10      PULS X
219 0219 c042 bf 70 30      STX USER_Y
220 0220 c045 35 10      PULS X
221 0221 c047 bf 70 2a      STX USER_U
222 0222 c04a 35 10      PULS X
223 0223 c04c bf 70 28      STX USER_PC
224 0224 c04f bf 70 1b      stx save_PC
```

```
225 0225 c052 10 fe 70 2e      LDS SAVE_SP
226 0226 c056 bd d0 5b      JSR key_PC
227 0227 c059 39      rts
228 0228      *
229 0229      * 16 bit division
230 0230      *
231 0231      *
232 0232      * D=X/D
233 0233      *
234 0234 c05a 7f 70 4f      ccd1: CLR ADR31
235 0235 c05d bf 70 4a      STX ADR1
236 0236 c060 fd 70 4c      STD ADR2
237 0237 c063 2e 0b      ccd1: BGT ccd2
238 0238 c065 73 70 4f      COM ADR31
239 0239 c068 43      COMA
240 0240 c069 53      COMB
241 0241 c06a c3 00 01      ADDD #1
242 0242 c06d fd 70 4c      STD ADR2
243 0243 c070 fc 70 4a      ccd2: LDD ADR1
244 0244 c073 10 83 00 00      CMPD #0
245 0245 c077 2e 0b      BGT ccd3
246 0246 c079 73 70 4f      COM ADR31
247 0247 c07c 43      COMA
248 0248 c07d 53      COMB
249 0249 c07e c3 00 01      ADDD #1
250 0250 c081 fd 70 4a      STD ADR1
251 0251 c084 86 10      ccd3: LDA #16
252 0252 c086 b7 70 4e      STA ADR3
253 0253 c089 4f      CLRA
254 0254 c08a 5f      CLR B
255 0255      DIVD:
256 0256 c08b 78 70 4b      ASL ADR1+1
257 0257 c08e 79 70 4a      ROL ADR1
258 0258 c091 59      ROLB
259 0259 c092 49      ROLA
260 0260 c093 10 b3 70 4c      CMPD ADR2
261 0261 c097 25 06      BLO NOSUB
262 0262 c099 b3 70 4c      SUBD ADR2
263 0263 c09c 7c 70 4b      INC ADR1+1
264 0264 c09f 7a 70 4e      NOSUB: DEC ADR3
265 0265 c0a2 26 e7      BNE DIVD
266 0266 c0a4 1f 01      TFR D,X
267 0267 c0a6 fc 70 4a      LDD ADR1
268 0268 c0a9 7d 70 4f      TST ADR31
269 0269 c0ac 27 05      BEQ ccd4
270 0270 c0ae 43      COMA
271 0271 c0af 53      COMB
272 0272 c0b0 c3 00 01      ADDD #1
273 0273 c0b3 39      ccd4: RTS
274 0274      *
275 0275      * 16 bit multiplication
276 0276      *
277 0277 c0b4 bf 70 4a      ccmult: STX ADR1
278 0278 c0b7 fd 70 4c      STD ADR2
279 0279 c0ba 7f 70 4e      CLR ADR3
280 0280 c0bd 7f 70 4f      CLR ADR3+1
```

```
281 0281 c0c0 b6 70 4b      LDA ADR1+1
282 0282 c0c3 f6 70 4d      LDB ADR2+1
283 0283 c0c6 3d           MUL
284 0284 c0c7 fd 70 50      STD ADR3+2
285 0285 c0ca b6 70 4a      LDA ADR1
286 0286 c0cd f6 70 4d      LDB ADR2+1
287 0287 c0d0 3d           MUL
288 0288 c0d1 f3 70 4f      ADDD ADR3+1
289 0289 c0d4 fd 70 4f      STD ADR3+1
290 0290 c0d7 b6 70 4b      LDA ADR1+1
291 0291 c0da f6 70 4c      LDB ADR2
292 0292 c0dd 3d           MUL
293 0293 c0de f3 70 4f      ADDD ADR3+1
294 0294 c0e1 fd 70 4f      STD ADR3+1
295 0295 c0e4 24 03         BCC NOCARRY
296 0296 c0e6 7c 70 4e      INC ADR3
297 0297 c0e9 b6 70 4a      NOCARRY: LDA ADR1
298 0298 c0ec f6 70 4c      LDB ADR2
299 0299 c0ef 3d           MUL
300 0300 c0f0 f3 70 4e      ADDD ADR3
301 0301 c0f3 1f 01         TFR D,X
302 0302 c0f5 fc 70 50      LDD ADR3+2
303 0303 c0f8 39           RTS
304 0304
305 0305           * switch statement
306 0306
307 0307           * D = switch value
308 0308           * S = switch table
309 0309           *   FDB addr1, value1
310 0310           *   FDB addr2, value2
311 0311           *
312 0312           *   ...
313 0313           *   [JMP DEFAULT]
314 0314           * continuation
315 0315           *
316 0316 c0f9 35 10         cccswitch: PULS X get stack adr
317 0317 c0fb 10 ae 81      SWLOOP: LDY ,X++ get address
318 0318 c0fe 10 8c 00 00      CMPY #0 if 0
319 0319 c102 27 07           BEQ SWEND then it is the default
320 0320 c104 10 a3 81           CMPD ,X++ else, if it is not the swit
321 0321 c107 26 f2           BNE SWLOOP try next one
322 0322 c109 6e a4           JMP ,Y else we found it
323 0323 c10b 6e 84           SWEND: JMP ,X default exit
324 0324
325 0325           * .area CONST (REL,CON)
326 0326
327 0327           * .global convert
328 0328           convert:
329 0329 c10d bd 30 9b ba 36 ae   FCB 189,48,155,186,54,174,175,56,191,190
330           af 38 bf be
331 0330 c117 3f a7 8d b3 8f 0f   FCB 63,167,141,179,143,15
332 0331
333 0332           * .global cold_msg
334 0333           cold_msg:
335 0334 c11d 00 00 00 00 00 00 00   FCB 0,0,0,0,0,0,174,191,189,190
336           ae bf bd be
```

```
337 0335 c127 00 00          FCB 0,0
338 0336
339 0337          * .area ROMCODE (REL,CON)
340 0338
341 0339          * .module LcdReady
342 0340
343 0341          * .global LcdReady
344 0342          LcdReady:
345 0343 c129 cc 00 00          ldd #0
346 0344 c12c fd 70 36          std timeout
347 0345 c12f cc 90 02          ldd #-28670
348 0346 c132 fd 70 46          std dptr
349 0347          cc2:
350 0348 c135 fc 70 46          ldd dptr
351 0349 c138 34 06          pshs d
352 0350 c13a e6 f1          ldb [,s++]
353 0351 c13c 1d          sex
354 0352 c13d 34 06          pshs d
355 0353 c13f cc 00 80          ldd #128
356 0354 c142 a4 e0          anda ,s+
357 0355 c144 e4 e0          andb ,s+
358 0356 c146 10 83 00 00          cmpd #0
359 0357 c14a 10 27 00 20          lbeq cc4 ;_ instruction flagged for non opt
360 0358 c14e fc 70 36          ldd timeout
361 0359 c151 34 06          pshs d
362 0360 c153 cc 01 f4          ldd #500
363 0361 c156 10 a3 e1          cmpd ,s++
364 0362 c159 2e 05          bgt *+7
365 0363 c15b cc 00 00          ldd #0
366 0364 c15e 20 03          bra *+5
367 0365 c160 cc 00 01          ldd #1
368 0366 c163 10 83 00 00          cmpd #0
369 0367 c167 10 27 00 03          lbeq cc4 ;_ instruction flagged for non opt
370 0368 c16b cc 00 01          ldd #1
371 0369          cc4:
372 0370 c16e 10 83 00 00          cmpd #0
373 0371 c172 10 27 00 0c          lbeq cc3
374 0372 c176 fc 70 36          ldd timeout
375 0373 c179 c3 00 01          addd #1
376 0374 c17c fd 70 36          std timeout
377 0375 c17f 7e c1 35          jmp cc2
378 0376          cc3:
379 0377 c182 39          rts
380 0378
381 0379          * .global clr_scre
382 0380          clr_scre:
383 0381 c183 bd c1 29          jsr LcdReady
384 0382 c186 cc 90 00          ldd #-28672
385 0383 c189 fd 70 46          std dptr
386 0384 c18c fc 70 46          ldd dptr
387 0385 c18f 34 06          pshs d
388 0386 c191 cc 00 01          ldd #1
389 0387 c194 e7 f1          stb [,s++]
390 0388 c196 39          rts
391 0389
392 0390          * .global goto_xy
```

```
393 0391
394 0392 c197 bd c1 29
395 0393 c19a cc 90 00
396 0394 c19d fd 70 46
397 0395 c1a0 31 62
398 0396 c1a2 1f 20
399 0397 c1a4 34 06
400 0398 c1a6 ec f1
401 0399 c1a8 7e c2 12
402 0400
403 0401 c1ab fc 70 46
404 0402 c1ae 34 06
405 0403 c1b0 cc 00 80
406 0404 c1b3 34 06
407 0405 c1b5 31 68
408 0406 c1b7 1f 20
409 0407 c1b9 34 06
410 0408 c1bb ec f1
411 0409 c1bd e3 e1
412 0410 c1bf e7 f1
413 0411 c1c1 7e c2 27
414 0412
415 0413 c1c4 fc 70 46
416 0414 c1c7 34 06
417 0415 c1c9 cc 00 c0
418 0416 c1cc 34 06
419 0417 c1ce 31 68
420 0418 c1d0 1f 20
421 0419 c1d2 34 06
422 0420 c1d4 ec f1
423 0421 c1d6 e3 e1
424 0422 c1d8 e7 f1
425 0423 c1da 7e c2 27
426 0424
427 0425 c1dd fc 70 46
428 0426 c1e0 34 06
429 0427 c1e2 cc 00 94
430 0428 c1e5 34 06
431 0429 c1e7 31 68
432 0430 c1e9 1f 20
433 0431 c1eb 34 06
434 0432 c1ed ec f1
435 0433 c1ef e3 e1
436 0434 c1f1 e7 f1
437 0435 c1f3 7e c2 27
438 0436
439 0437 c1f6 fc 70 46
440 0438 c1f9 34 06
441 0439 c1fb cc 00 d4
442 0440 c1fe 34 06
443 0441 c200 31 68
444 0442 c202 1f 20
445 0443 c204 34 06
446 0444 c206 ec f1
447 0445 c208 e3 e1
448 0446 c20a e7 f1

goto_xy:
    jsr LcdReady
    ldd #-28672
    std dptr
    leay 2,s
    tfr y,d
    pshs d
    ldd [,s++]
    jmp cc7
cc8:
    ldd dptr
    pshs d
    ldd #128
    pshs d
    leay 8,s
    tfr y,d
    pshs d
    ldd [,s++]
    addd ,s++
    stb [,s++]
    jmp cc6
cc9:
    ldd dptr
    pshs d
    ldd #192
    pshs d
    leay 8,s
    tfr y,d
    pshs d
    ldd [,s++]
    addd ,s++
    stb [,s++]
    jmp cc6
cc10:
    ldd dptr
    pshs d
    ldd #148
    pshs d
    leay 8,s
    tfr y,d
    pshs d
    ldd [,s++]
    addd ,s++
    stb [,s++]
    jmp cc6
cc11:
    ldd dptr
    pshs d
    ldd #212
    pshs d
    leay 8,s
    tfr y,d
    pshs d
    ldd [,s++]
    addd ,s++
    stb [,s++]
```

```
449 0447 c20c 7e c2 27      jmp cc6
450 0448 c20f 7e c2 27      jmp cc6
451 0449                      cc7:
452 0450 c212 bd c0 f9      jsr ccswitch
453 0451 c215 c1 ab 00 00    FDB cc8,0
454 0452 c219 c1 c4 00 01    FDB cc9,1
455 0453 c21d c1 dd 00 02    FDB cc10,2
456 0454 c221 c1 f6 00 03    FDB cc11,3
457 0455 c225 00 00          FDB 0
458 0456                      cc6:
459 0457 c227 39            rts
460 0458
461 0459                      * .global InitLCD
462 0460                      InitLCD:
463 0461 c228 bd c1 29      jsr LcdReady
464 0462 c22b cc 90 00      ldd #-28672
465 0463 c22e fd 70 46      std dptr
466 0464 c231 fc 70 46      ldd dptr
467 0465 c234 34 06          pshs d
468 0466 c236 cc 00 38      ldd #56
469 0467 c239 e7 f1          stb [,s++]
470 0468 c23b bd c1 29      jsr LcdReady
471 0469 c23e cc 90 00      ldd #-28672
472 0470 c241 fd 70 46      std dptr
473 0471 c244 fc 70 46      ldd dptr
474 0472 c247 34 06          pshs d
475 0473 c249 cc 00 0c      ldd #12
476 0474 c24c e7 f1          stb [,s++]
477 0475 c24e bd c1 83      jsr clr_scre
478 0476 c251 cc 00 00      ldd #0
479 0477 c254 34 06          pshs d
480 0478 c256 cc 00 00      ldd #0
481 0479 c259 34 06          pshs d
482 0480 c25b bd c1 97      jsr goto_xy
483 0481 c25e 32 64          leas 4,s
484 0482 c260 cc 00 64      ldd #100
485 0483 c263 34 06          pshs d
486 0484 c265 bd c8 41      jsr delay_ms
487 0485 c268 32 62          leas 2,s
488 0486 c26a 39            rts
489 0487
490 0488                      * .global PutLCD
491 0489                      PutLCD:
492 0490 c26b 32 7f          leas -1,s
493 0491 c26d 31 e4          leay 0,s
494 0492 c26f 1f 20          tfr y,d
495 0493 c271 34 06          pshs d
496 0494 c273 cc 00 00      ldd #0
497 0495 c276 e7 f1          stb [,s++]
498 0496                      cc14:
499 0497 c278 31 63          leay 3,s
500 0498 c27a 1f 20          tfr y,d
501 0499 c27c 34 06          pshs d
502 0500 c27e ec f1          ldd [,s++]
503 0501 c280 34 06          pshs d
504 0502 c282 31 62          leay 2,s
```

```
505 0503 c284 1f 20          tfr y,d
506 0504 c286 34 06          pshs d
507 0505 c288 e6 f1          ldb [ ,s++ ]
508 0506 c28a 1d             sex
509 0507 c28b e3 e1          addd ,s++
510 0508 c28d 34 06          pshs d
511 0509 c28f e6 f1          ldb [ ,s++ ]
512 0510 c291 1d             sex
513 0511 c292 34 06          pshs d
514 0512 c294 cc 00 00          ldd #0
515 0513 c297 10 a3 e1          cmpd ,s++
516 0514 c29a 26 05          bne *+7
517 0515 c29c cc 00 00          ldd #0
518 0516 c29f 20 03          bra *+5
519 0517 c2a1 cc 00 01          ldd #1
520 0518 c2a4 10 83 00 00          cmpd #0
521 0519 c2a8 10 27 00 46          lbeq cc13
522 0520 c2ac 7e c2 c5          jmp cc15
523 0521                         cc12:
524 0522 c2af 31 e4          leay 0,s
525 0523 c2b1 1f 20          tfr y,d
526 0524 c2b3 34 06          pshs d
527 0525 c2b5 34 06          pshs d
528 0526 c2b7 e6 f1          ldb [ ,s++ ]
529 0527 c2b9 1d             sex
530 0528 c2ba c3 00 01          addd #1
531 0529 c2bd e7 f1          stb [ ,s++ ]
532 0530 c2bf 83 00 01          subd #1
533 0531 c2c2 7e c2 78          jmp cc14
534 0532                         cc15:
535 0533 c2c5 bd c1 29          jsr LcdReady
536 0534 c2c8 cc 90 01          ldd #-28671
537 0535 c2cb fd 70 46          std dptr
538 0536 c2ce fc 70 46          ldd dptr
539 0537 c2d1 34 06          pshs d
540 0538 c2d3 31 65          leay 5,s
541 0539 c2d5 1f 20          tfr y,d
542 0540 c2d7 34 06          pshs d
543 0541 c2d9 ec f1          ldd [ ,s++ ]
544 0542 c2db 34 06          pshs d
545 0543 c2dd 31 64          leay 4,s
546 0544 c2df 1f 20          tfr y,d
547 0545 c2e1 34 06          pshs d
548 0546 c2e3 e6 f1          ldb [ ,s++ ]
549 0547 c2e5 1d             sex
550 0548 c2e6 e3 e1          addd ,s++
551 0549 c2e8 34 06          pshs d
552 0550 c2ea e6 f1          ldb [ ,s++ ]
553 0551 c2ec 1d             sex
554 0552 c2ed e7 f1          stb [ ,s++ ]
555 0553 c2ef 7e c2 af          jmp cc12
556 0554                         cc13:
557 0555 c2f2 32 61          leas 1,s
558 0556 c2f4 39             rts
559 0557
560 0558             * .global putch_lc
```

```
561 0559
562 0560 c2f5 bd c1 29
563 0561 c2f8 cc 90 01
564 0562 c2fb fd 70 46
565 0563 c2fe fc 70 46
566 0564 c301 34 06
567 0565 c303 31 65
568 0566 c305 1f 20
569 0567 c307 34 06
570 0568 c309 e6 f1
571 0569 c30b 1d
572 0570 c30c e7 f1
573 0571 c30e 39
574 0572
575 0573
576 0574
577 0575 c30f 32 7f
578 0576 c311 31 64
579 0577 c313 1f 20
580 0578 c315 34 06
581 0579 c317 e6 f1
582 0580 c319 1d
583 0581 c31a 34 06
584 0582 c31c cc 00 16
585 0583 c31f 10 a3 e1
586 0584 c322 27 05
587 0585 c324 cc 00 00
588 0586 c327 20 03
589 0587 c329 cc 00 01
590 0588 c32c 10 83 00 00
591 0589 c330 10 27 00 06
592 0590 c334 cc 00 00
593 0591 c337 32 61
594 0592 c339 39
595 0593
596 0594 c33a 31 64
597 0595 c33c 1f 20
598 0596 c33e 34 06
599 0597 c340 e6 f1
600 0598 c342 1d
601 0599 c343 34 06
602 0600 c345 cc 00 21
603 0601 c348 10 a3 e1
604 0602 c34b 27 05
605 0603 c34d cc 00 00
606 0604 c350 20 03
607 0605 c352 cc 00 01
608 0606 c355 10 83 00 00
609 0607 c359 10 27 00 06
610 0608 c35d cc 00 01
611 0609 c360 32 61
612 0610 c362 39
613 0611
614 0612 c363 31 64
615 0613 c365 1f 20
616 0614 c367 34 06

putch_lc:
    jsr LcdReady
    ldd #-28671
    std dptr
    ldd dptr
    pshs d
    leay 5,s
    tfr y,d
    pshs d
    ldb [ ,s++ ]
    sex
    stb [ ,s++ ]
    rts

* .global key_code
key_code:
    leas -1,s
    leay 4,s
    tfr y,d
    pshs d
    ldb [ ,s++ ]
    sex
    pshs d
    ldd #22
    cmpd ,s++
    beq *+7
    ldd #0
    bra *+5
    ldd #1
    cmpd #0
    lbeq cc16
    ldd #0
    leas 1,s
    rts

cc16:
    leay 4,s
    tfr y,d
    pshs d
    ldb [ ,s++ ]
    sex
    pshs d
    ldd #33
    cmpd ,s++
    beq *+7
    ldd #0
    bra *+5
    ldd #1
    cmpd #0
    lbeq cc17
    ldd #1
    leas 1,s
    rts

cc17:
    leay 4,s
    tfr y,d
    pshs d
```

```
617 0615 c369 e6 f1           ldb  [ ,s++]
618 0616 c36b 1d             sex
619 0617 c36c 34 06          pshs d
620 0618 c36e cc 00 1b          ldd #27
621 0619 c371 10 a3 e1          cmpd ,s++
622 0620 c374 27 05          beq *+7
623 0621 c376 cc 00 00          ldd #0
624 0622 c379 20 03          bra *+5
625 0623 c37b cc 00 01          ldd #1
626 0624 c37e 10 83 00 00          cmpd #0
627 0625 c382 10 27 00 06          lbeq cc18
628 0626 c386 cc 00 02          ldd #2
629 0627 c389 32 61          leas 1,s
630 0628 c38b 39             rts
631 0629                   cc18:
632 0630 c38c 31 64          leay 4,s
633 0631 c38e 1f 20          tfr y,d
634 0632 c390 34 06          pshs d
635 0633 c392 e6 f1          ldb  [ ,s++]
636 0634 c394 1d             sex
637 0635 c395 34 06          pshs d
638 0636 c397 cc 00 15          ldd #21
639 0637 c39a 10 a3 e1          cmpd ,s++
640 0638 c39d 27 05          beq *+7
641 0639 c39f cc 00 00          ldd #0
642 0640 c3a2 20 03          bra *+5
643 0641 c3a4 cc 00 01          ldd #1
644 0642 c3a7 10 83 00 00          cmpd #0
645 0643 c3ab 10 27 00 06          lbeq cc19
646 0644 c3af cc 00 03          ldd #3
647 0645 c3b2 32 61          leas 1,s
648 0646 c3b4 39             rts
649 0647                   cc19:
650 0648 c3b5 31 64          leay 4,s
651 0649 c3b7 1f 20          tfr y,d
652 0650 c3b9 34 06          pshs d
653 0651 c3bb e6 f1          ldb  [ ,s++]
654 0652 c3bd 1d             sex
655 0653 c3be 34 06          pshs d
656 0654 c3c0 cc 00 1c          ldd #28
657 0655 c3c3 10 a3 e1          cmpd ,s++
658 0656 c3c6 27 05          beq *+7
659 0657 c3c8 cc 00 00          ldd #0
660 0658 c3cb 20 03          bra *+5
661 0659 c3cd cc 00 01          ldd #1
662 0660 c3d0 10 83 00 00          cmpd #0
663 0661 c3d4 10 27 00 06          lbeq cc20
664 0662 c3d8 cc 00 04          ldd #4
665 0663 c3db 32 61          leas 1,s
666 0664 c3dd 39             rts
667 0665                   cc20:
668 0666 c3de 31 64          leay 4,s
669 0667 c3e0 1f 20          tfr y,d
670 0668 c3e2 34 06          pshs d
671 0669 c3e4 e6 f1          ldb  [ ,s++]
672 0670 c3e6 1d             sex
```

```
673 0671 c3e7 34 06      pshs d
674 0672 c3e9 cc 00 20    ldd #32
675 0673 c3ec 10 a3 e1    cmpd ,s++
676 0674 c3ef 27 05      beq *+7
677 0675 c3f1 cc 00 00    ldd #0
678 0676 c3f4 20 03      bra *+5
679 0677 c3f6 cc 00 01    ldd #1
680 0678 c3f9 10 83 00 00  cmpd #0
681 0679 c3fd 10 27 00 06  lbeq cc21
682 0680 c401 cc 00 05    ldd #5
683 0681 c404 32 61      leas 1,s
684 0682 c406 39        rts
685 0683                  cc21:
686 0684 c407 31 64      leay 4,s
687 0685 c409 1f 20      tfr y,d
688 0686 c40b 34 06      pshs d
689 0687 c40d e6 f1      ldb [,s++]
690 0688 c40f 1d        sex
691 0689 c410 34 06      pshs d
692 0690 c412 cc 00 1a    ldd #26
693 0691 c415 10 a3 e1    cmpd ,s++
694 0692 c418 27 05      beq *+7
695 0693 c41a cc 00 00    ldd #0
696 0694 c41d 20 03      bra *+5
697 0695 c41f cc 00 01    ldd #1
698 0696 c422 10 83 00 00  cmpd #0
699 0697 c426 10 27 00 06  lbeq cc22
700 0698 c42a cc 00 06    ldd #6
701 0699 c42d 32 61      leas 1,s
702 0700 c42f 39        rts
703 0701                  cc22:
704 0702 c430 31 64      leay 4,s
705 0703 c432 1f 20      tfr y,d
706 0704 c434 34 06      pshs d
707 0705 c436 e6 f1      ldb [,s++]
708 0706 c438 1d        sex
709 0707 c439 34 06      pshs d
710 0708 c43b cc 00 14    ldd #20
711 0709 c43e 10 a3 e1    cmpd ,s++
712 0710 c441 27 05      beq *+7
713 0711 c443 cc 00 00    ldd #0
714 0712 c446 20 03      bra *+5
715 0713 c448 cc 00 01    ldd #1
716 0714 c44b 10 83 00 00  cmpd #0
717 0715 c44f 10 27 00 06  lbeq cc23
718 0716 c453 cc 00 07    ldd #7
719 0717 c456 32 61      leas 1,s
720 0718 c458 39        rts
721 0719                  cc23:
722 0720 c459 31 64      leay 4,s
723 0721 c45b 1f 20      tfr y,d
724 0722 c45d 34 06      pshs d
725 0723 c45f e6 f1      ldb [,s++]
726 0724 c461 1d        sex
727 0725 c462 34 06      pshs d
728 0726 c464 cc 00 22    ldd #34
```

```
729 0727 c467 10 a3 e1      cmpd ,s++
730 0728 c46a 27 05      beq *+7
731 0729 c46c cc 00 00      ldd #0
732 0730 c46f 20 03      bra *+5
733 0731 c471 cc 00 01      ldd #1
734 0732 c474 10 83 00 00      cmpd #0
735 0733 c478 10 27 00 06      lbeq cc24
736 0734 c47c cc 00 08      ldd #8
737 0735 c47f 32 61      leas 1,s
738 0736 c481 39      rts
739 0737      cc24:
740 0738 c482 31 64      leay 4,s
741 0739 c484 1f 20      tfr y,d
742 0740 c486 34 06      pshs d
743 0741 c488 e6 f1      ldb [,s++]
744 0742 c48a 1d      sex
745 0743 c48b 34 06      pshs d
746 0744 c48d cc 00 1f      ldd #31
747 0745 c490 10 a3 e1      cmpd ,s++
748 0746 c493 27 05      beq *+7
749 0747 c495 cc 00 00      ldd #0
750 0748 c498 20 03      bra *+5
751 0749 c49a cc 00 01      ldd #1
752 0750 c49d 10 83 00 00      cmpd #0
753 0751 c4a1 10 27 00 06      lbeq cc25
754 0752 c4a5 cc 00 09      ldd #9
755 0753 c4a8 32 61      leas 1,s
756 0754 c4aa 39      rts
757 0755      cc25:
758 0756 c4ab 31 64      leay 4,s
759 0757 c4ad 1f 20      tfr y,d
760 0758 c4af 34 06      pshs d
761 0759 c4b1 e6 f1      ldb [,s++]
762 0760 c4b3 1d      sex
763 0761 c4b4 34 06      pshs d
764 0762 c4b6 cc 00 19      ldd #25
765 0763 c4b9 10 a3 e1      cmpd ,s++
766 0764 c4bc 27 05      beq *+7
767 0765 c4be cc 00 00      ldd #0
768 0766 c4c1 20 03      bra *+5
769 0767 c4c3 cc 00 01      ldd #1
770 0768 c4c6 10 83 00 00      cmpd #0
771 0769 c4ca 10 27 00 06      lbeq cc26
772 0770 c4ce cc 00 0a      ldd #10
773 0771 c4d1 32 61      leas 1,s
774 0772 c4d3 39      rts
775 0773      cc26:
776 0774 c4d4 31 64      leay 4,s
777 0775 c4d6 1f 20      tfr y,d
778 0776 c4d8 34 06      pshs d
779 0777 c4da e6 f1      ldb [,s++]
780 0778 c4dc 1d      sex
781 0779 c4dd 34 06      pshs d
782 0780 c4df cc 00 13      ldd #19
783 0781 c4e2 10 a3 e1      cmpd ,s++
784 0782 c4e5 27 05      beq *+7
```

```
785 0783 c4e7 cc 00 00      ldd #0
786 0784 c4ea 20 03      bra *+5
787 0785 c4ec cc 00 01      ldd #1
788 0786 c4ef 10 83 00 00      cmpd #0
789 0787 c4f3 10 27 00 06      lbeq cc27
790 0788 c4f7 cc 00 0b      ldd #11
791 0789 c4fa 32 61      leas 1,s
792 0790 c4fc 39      rts
793 0791      cc27:
794 0792 c4fd 31 64      leay 4,s
795 0793 c4ff 1f 20      tfr y,d
796 0794 c501 34 06      pshs d
797 0795 c503 e6 f1      ldb [ ,s++ ]
798 0796 c505 1d      sex
799 0797 c506 34 06      pshs d
800 0798 c508 cc 00 03      ldd #3
801 0799 c50b 10 a3 e1      cmpd ,s++
802 0800 c50e 27 05      beq *+7
803 0801 c510 cc 00 00      ldd #0
804 0802 c513 20 03      bra *+5
805 0803 c515 cc 00 01      ldd #1
806 0804 c518 10 83 00 00      cmpd #0
807 0805 c51c 10 27 00 06      lbeq cc28
808 0806 c520 cc 00 0c      ldd #12
809 0807 c523 32 61      leas 1,s
810 0808 c525 39      rts
811 0809      cc28:
812 0810 c526 31 64      leay 4,s
813 0811 c528 1f 20      tfr y,d
814 0812 c52a 34 06      pshs d
815 0813 c52c e6 f1      ldb [ ,s++ ]
816 0814 c52e 1d      sex
817 0815 c52f 34 06      pshs d
818 0816 c531 cc 00 1e      ldd #30
819 0817 c534 10 a3 e1      cmpd ,s++
820 0818 c537 27 05      beq *+7
821 0819 c539 cc 00 00      ldd #0
822 0820 c53c 20 03      bra *+5
823 0821 c53e cc 00 01      ldd #1
824 0822 c541 10 83 00 00      cmpd #0
825 0823 c545 10 27 00 06      lbeq cc29
826 0824 c549 cc 00 0d      ldd #13
827 0825 c54c 32 61      leas 1,s
828 0826 c54e 39      rts
829 0827      cc29:
830 0828 c54f 31 64      leay 4,s
831 0829 c551 1f 20      tfr y,d
832 0830 c553 34 06      pshs d
833 0831 c555 e6 f1      ldb [ ,s++ ]
834 0832 c557 1d      sex
835 0833 c558 34 06      pshs d
836 0834 c55a cc 00 18      ldd #24
837 0835 c55d 10 a3 e1      cmpd ,s++
838 0836 c560 27 05      beq *+7
839 0837 c562 cc 00 00      ldd #0
840 0838 c565 20 03      bra *+5
```

```
841 0839 c567 cc 00 01      ldd #1
842 0840 c56a 10 83 00 00    cmpd #0
843 0841 c56e 10 27 00 06    lbeq cc30
844 0842 c572 cc 00 0e      ldd #14
845 0843 c575 32 61        leas 1,s
846 0844 c577 39        rts
847 0845                    cc30:
848 0846 c578 31 64        leay 4,s
849 0847 c57a 1f 20        tfr y,d
850 0848 c57c 34 06        pshs d
851 0849 c57e e6 f1        ldb [ ,s++ ]
852 0850 c580 1d        sex
853 0851 c581 34 06        pshs d
854 0852 c583 cc 00 12    ldd #18
855 0853 c586 10 a3 e1    cmpd ,s++
856 0854 c589 27 05    beq *+7
857 0855 c58b cc 00 00    ldd #0
858 0856 c58e 20 03    bra *+5
859 0857 c590 cc 00 01    ldd #1
860 0858 c593 10 83 00 00    cmpd #0
861 0859 c597 10 27 00 06    lbeq cc31
862 0860 c59b cc 00 0f    ldd #15
863 0861 c59e 32 61    leas 1,s
864 0862 c5a0 39        rts
865 0863                    cc31:
866 0864 c5a1 31 64        leay 4,s
867 0865 c5a3 1f 20        tfr y,d
868 0866 c5a5 34 06        pshs d
869 0867 c5a7 e6 f1        ldb [ ,s++ ]
870 0868 c5a9 1d        sex
871 0869 c5aa 34 06        pshs d
872 0870 c5ac cc 00 0c    ldd #12
873 0871 c5af 10 a3 e1    cmpd ,s++
874 0872 c5b2 27 05    beq *+7
875 0873 c5b4 cc 00 00    ldd #0
876 0874 c5b7 20 03    bra *+5
877 0875 c5b9 cc 00 01    ldd #1
878 0876 c5bc 10 83 00 00    cmpd #0
879 0877 c5c0 10 27 00 06    lbeq cc32
880 0878 c5c4 cc 00 10    ldd #16
881 0879 c5c7 32 61    leas 1,s
882 0880 c5c9 39        rts
883 0881                    cc32:
884 0882 c5ca 31 64        leay 4,s
885 0883 c5cc 1f 20        tfr y,d
886 0884 c5ce 34 06        pshs d
887 0885 c5d0 e6 f1        ldb [ ,s++ ]
888 0886 c5d2 1d        sex
889 0887 c5d3 34 06        pshs d
890 0888 c5d5 cc 00 0d    ldd #13
891 0889 c5d8 10 a3 e1    cmpd ,s++
892 0890 c5db 27 05    beq *+7
893 0891 c5dd cc 00 00    ldd #0
894 0892 c5e0 20 03    bra *+5
895 0893 c5e2 cc 00 01    ldd #1
896 0894 c5e5 10 83 00 00    cmpd #0
```

```
897 0895 c5e9 10 27 00 06      lbeq cc33
898 0896 c5ed cc 00 11      ldd #17
899 0897 c5f0 32 61      leas 1,s
900 0898 c5f2 39      rts
901 0899      cc33:
902 0900 c5f3 31 64      leay 4,s
903 0901 c5f5 1f 20      tfr y,d
904 0902 c5f7 34 06      pshs d
905 0903 c5f9 e6 f1      ldb [ ,s++ ]
906 0904 c5fb 1d      sex
907 0905 c5fc 34 06      pshs d
908 0906 c5fe cc 00 0e      ldd #14
909 0907 c601 10 a3 e1      cmpd ,s++
910 0908 c604 27 05      beq *+7
911 0909 c606 cc 00 00      ldd #0
912 0910 c609 20 03      bra *+5
913 0911 c60b cc 00 01      ldd #1
914 0912 c60e 10 83 00 00      cmpd #0
915 0913 c612 10 27 00 06      lbeq cc34
916 0914 c616 cc 00 12      ldd #18
917 0915 c619 32 61      leas 1,s
918 0916 c61b 39      rts
919 0917      cc34:
920 0918 c61c 31 64      leay 4,s
921 0919 c61e 1f 20      tfr y,d
922 0920 c620 34 06      pshs d
923 0921 c622 e6 f1      ldb [ ,s++ ]
924 0922 c624 1d      sex
925 0923 c625 34 06      pshs d
926 0924 c627 cc 00 0f      ldd #15
927 0925 c62a 10 a3 e1      cmpd ,s++
928 0926 c62d 27 05      beq *+7
929 0927 c62f cc 00 00      ldd #0
930 0928 c632 20 03      bra *+5
931 0929 c634 cc 00 01      ldd #1
932 0930 c637 10 83 00 00      cmpd #0
933 0931 c63b 10 27 00 06      lbeq cc35
934 0932 c63f cc 00 13      ldd #19
935 0933 c642 32 61      leas 1,s
936 0934 c644 39      rts
937 0935      cc35:
938 0936 c645 31 64      leay 4,s
939 0937 c647 1f 20      tfr y,d
940 0938 c649 34 06      pshs d
941 0939 c64b e6 f1      ldb [ ,s++ ]
942 0940 c64d 1d      sex
943 0941 c64e 34 06      pshs d
944 0942 c650 cc 00 06      ldd #6
945 0943 c653 10 a3 e1      cmpd ,s++
946 0944 c656 27 05      beq *+7
947 0945 c658 cc 00 00      ldd #0
948 0946 c65b 20 03      bra *+5
949 0947 c65d cc 00 01      ldd #1
950 0948 c660 10 83 00 00      cmpd #0
951 0949 c664 10 27 00 06      lbeq cc36
952 0950 c668 cc 00 14      ldd #20
```

```
953 0951 c66b 32 61           leas 1,s
954 0952 c66d 39           rts
955 0953                   cc36:
956 0954 c66e 31 64           leay 4,s
957 0955 c670 1f 20           tfr y,d
958 0956 c672 34 06           pshs d
959 0957 c674 e6 f1           ldb [ ,s++ ]
960 0958 c676 1d           sex
961 0959 c677 34 06           pshs d
962 0960 c679 cc 00 07           ldd #7
963 0961 c67c 10 a3 e1           cmpd ,s++
964 0962 c67f 27 05           beq *+7
965 0963 c681 cc 00 00           ldd #0
966 0964 c684 20 03           bra *+5
967 0965 c686 cc 00 01           ldd #1
968 0966 c689 10 83 00 00           cmpd #0
969 0967 c68d 10 27 00 06           lbeq cc37
970 0968 c691 cc 00 15           ldd #21
971 0969 c694 32 61           leas 1,s
972 0970 c696 39           rts
973 0971                   cc37:
974 0972 c697 31 64           leay 4,s
975 0973 c699 1f 20           tfr y,d
976 0974 c69b 34 06           pshs d
977 0975 c69d e6 f1           ldb [ ,s++ ]
978 0976 c69f 1d           sex
979 0977 c6a0 34 06           pshs d
980 0978 c6a2 cc 00 08           ldd #8
981 0979 c6a5 10 a3 e1           cmpd ,s++
982 0980 c6a8 27 05           beq *+7
983 0981 c6aa cc 00 00           ldd #0
984 0982 c6ad 20 03           bra *+5
985 0983 c6af cc 00 01           ldd #1
986 0984 c6b2 10 83 00 00           cmpd #0
987 0985 c6b6 10 27 00 06           lbeq cc38
988 0986 c6ba cc 00 16           ldd #22
989 0987 c6bd 32 61           leas 1,s
990 0988 c6bf 39           rts
991 0989                   cc38:
992 0990 c6c0 31 64           leay 4,s
993 0991 c6c2 1f 20           tfr y,d
994 0992 c6c4 34 06           pshs d
995 0993 c6c6 e6 f1           ldb [ ,s++ ]
996 0994 c6c8 1d           sex
997 0995 c6c9 34 06           pshs d
998 0996 c6cb cc 00 09           ldd #9
999 0997 c6ce 10 a3 e1           cmpd ,s++
1000 0998 c6d1 27 05           beq *+7
1001 0999 c6d3 cc 00 00           ldd #0
1002 1000 c6d6 20 03           bra *+5
1003 1001 c6d8 cc 00 01           ldd #1
1004 1002 c6db 10 83 00 00           cmpd #0
1005 1003 c6df 10 27 00 06           lbeq cc39
1006 1004 c6e3 cc 00 17           ldd #23
1007 1005 c6e6 32 61           leas 1,s
1008 1006 c6e8 39           rts
```

```
1009 1007
1010 1008 c6e9 31 64
1011 1009 c6eb 1f 20
1012 1010 c6ed 34 06
1013 1011 c6ef e6 f1
1014 1012 c6f1 1d
1015 1013 c6f2 34 06
1016 1014 c6f4 cc 00 00
1017 1015 c6f7 10 a3 e1
1018 1016 c6fa 27 05
1019 1017 c6fc cc 00 00
1020 1018 c6ff 20 03
1021 1019 c701 cc 00 01
1022 1020 c704 10 83 00 00
1023 1021 c708 10 27 00 06
1024 1022 c70c cc 00 18
1025 1023 c70f 32 61
1026 1024 c711 39
1027 1025
1028 1026 c712 31 64
1029 1027 c714 1f 20
1030 1028 c716 34 06
1031 1029 c718 e6 f1
1032 1030 c71a 1d
1033 1031 c71b 34 06
1034 1032 c71d cc 00 01
1035 1033 c720 10 a3 e1
1036 1034 c723 27 05
1037 1035 c725 cc 00 00
1038 1036 c728 20 03
1039 1037 c72a cc 00 01
1040 1038 c72d 10 83 00 00
1041 1039 c731 10 27 00 06
1042 1040 c735 cc 00 19
1043 1041 c738 32 61
1044 1042 c73a 39
1045 1043
1046 1044 c73b 31 64
1047 1045 c73d 1f 20
1048 1046 c73f 34 06
1049 1047 c741 e6 f1
1050 1048 c743 1d
1051 1049 c744 34 06
1052 1050 c746 cc 00 02
1053 1051 c749 10 a3 e1
1054 1052 c74c 27 05
1055 1053 c74e cc 00 00
1056 1054 c751 20 03
1057 1055 c753 cc 00 01
1058 1056 c756 10 83 00 00
1059 1057 c75a 10 27 00 06
1060 1058 c75e cc 00 1a
1061 1059 c761 32 61
1062 1060 c763 39
1063 1061
1064 1062 c764 31 64
```

```
cc39:
    leay 4,s
    tfr y,d
    pshs d
    ldb [ ,s++ ]
    sex
    pshs d
    ldd #0
    cmpd ,s++
    beq *+7
    ldd #0
    bra *+5
    ldd #1
    cmpd #0
    lbeq cc40
    ldd #24
    leas 1,s
    rts
cc40:
    leay 4,s
    tfr y,d
    pshs d
    ldb [ ,s++ ]
    sex
    pshs d
    ldd #1
    cmpd ,s++
    beq *+7
    ldd #0
    bra *+5
    ldd #1
    cmpd #0
    lbeq cc41
    ldd #25
    leas 1,s
    rts
cc41:
    leay 4,s
    tfr y,d
    pshs d
    ldb [ ,s++ ]
    sex
    pshs d
    ldd #2
    cmpd ,s++
    beq *+7
    ldd #0
    bra *+5
    ldd #1
    cmpd #0
    lbeq cc42
    ldd #26
    leas 1,s
    rts
cc42:
    leay 4,s
```

```
1065 1063 c766 1f 20          tfr y,d
1066 1064 c768 34 06          pshs d
1067 1065 c76a e6 f1          ldb [ ,s++ ]
1068 1066 c76c 1d             sex
1069 1067 c76d 34 06          pshs d
1070 1068 c76f cc 00 24          ldd #36
1071 1069 c772 10 a3 e1          cmpd ,s++
1072 1070 c775 27 05          beq *+7
1073 1071 c777 cc 00 00          ldd #0
1074 1072 c77a 20 03          bra *+5
1075 1073 c77c cc 00 01          ldd #1
1076 1074 c77f 10 83 00 00          cmpd #0
1077 1075 c783 10 27 00 06          lbeq cc43
1078 1076 c787 cc 00 1b          ldd #27
1079 1077 c78a 32 61          leas 1,s
1080 1078 c78c 39             rts
1081 1079                      cc43:
1082 1080 c78d 31 64          leay 4,s
1083 1081 c78f 1f 20          tfr y,d
1084 1082 c791 34 06          pshs d
1085 1083 c793 e6 f1          ldb [ ,s++ ]
1086 1084 c795 1d             sex
1087 1085 c796 34 06          pshs d
1088 1086 c798 cc 00 23          ldd #35
1089 1087 c79b 10 a3 e1          cmpd ,s++
1090 1088 c79e 27 05          beq *+7
1091 1089 c7a0 cc 00 00          ldd #0
1092 1090 c7a3 20 03          bra *+5
1093 1091 c7a5 cc 00 01          ldd #1
1094 1092 c7a8 10 83 00 00          cmpd #0
1095 1093 c7ac 10 27 00 06          lbeq cc44
1096 1094 c7b0 cc 00 1c          ldd #28
1097 1095 c7b3 32 61          leas 1,s
1098 1096 c7b5 39             rts
1099 1097                      cc44:
1100 1098 c7b6 31 64          leay 4,s
1101 1099 c7b8 1f 20          tfr y,d
1102 1100 c7ba 34 06          pshs d
1103 1101 c7bc e6 f1          ldb [ ,s++ ]
1104 1102 c7be 1d             sex
1105 1103 c7bf 34 06          pshs d
1106 1104 c7c1 cc 00 1d          ldd #29
1107 1105 c7c4 10 a3 e1          cmpd ,s++
1108 1106 c7c7 27 05          beq *+7
1109 1107 c7c9 cc 00 00          ldd #0
1110 1108 c7cc 20 03          bra *+5
1111 1109 c7ce cc 00 01          ldd #1
1112 1110 c7d1 10 83 00 00          cmpd #0
1113 1111 c7d5 10 27 00 06          lbeq cc45
1114 1112 c7d9 cc 00 1d          ldd #29
1115 1113 c7dc 32 61          leas 1,s
1116 1114 c7de 39             rts
1117 1115                      cc45:
1118 1116 c7df 31 64          leay 4,s
1119 1117 c7e1 1f 20          tfr y,d
1120 1118 c7e3 34 06          pshs d
```

```
1121 1119 c7e5 e6 f1           ldb  [ ,s++]
1122 1120 c7e7 1d             sex
1123 1121 c7e8 34 06          pshs d
1124 1122 c7ea cc 00 17          ldd #23
1125 1123 c7ed 10 a3 e1          cmpd ,s++
1126 1124 c7f0 27 05          beq *+7
1127 1125 c7f2 cc 00 00          ldd #0
1128 1126 c7f5 20 03          bra *+5
1129 1127 c7f7 cc 00 01          ldd #1
1130 1128 c7fa 10 83 00 00          cmpd #0
1131 1129 c7fe 10 27 00 06          lbeq cc46
1132 1130 c802 cc 00 1e          ldd #30
1133 1131 c805 32 61          leas 1,s
1134 1132 c807 39          rts
1135 1133                      cc46:
1136 1134 c808 31 64          leay 4,s
1137 1135 c80a 1f 20          tfr y,d
1138 1136 c80c 34 06          pshs d
1139 1137 c80e e6 f1          ldb [ ,s++]
1140 1138 c810 1d             sex
1141 1139 c811 34 06          pshs d
1142 1140 c813 cc 00 10          ldd #16
1143 1141 c816 10 a3 e1          cmpd ,s++
1144 1142 c819 27 05          beq *+7
1145 1143 c81b cc 00 00          ldd #0
1146 1144 c81e 20 03          bra *+5
1147 1145 c820 cc 00 01          ldd #1
1148 1146 c823 10 83 00 00          cmpd #0
1149 1147 c827 10 27 00 06          lbeq cc47
1150 1148 c82b cc 00 1f          ldd #31
1151 1149 c82e 32 61          leas 1,s
1152 1150 c830 39          rts
1153 1151                      cc47:
1154 1152 c831 32 61          leas 1,s
1155 1153 c833 39          rts
1156 1154
1157 1155                      * .global delay_nu
1158 1156                      delay_nu:
1159 1157 c834 cc 00 00          ldd #0
1160 1158 c837 fd 70 15          std temp
1161 1159 c83a cc 00 00          ldd #0
1162 1160 c83d fd 70 15          std temp
1163 1161 c840 39          rts
1164 1162
1165 1163                      * .global delay_ms
1166 1164                      delay_ms:
1167 1165 c841 cc 00 00          ldd #0
1168 1166 c844 f7 70 04          stb n
1169 1167                      cc50:
1170 1168 c847 f6 70 04          ldb n
1171 1169 c84a 1d             sex
1172 1170 c84b 34 06          pshs d
1173 1171 c84d 31 64          leay 4,s
1174 1172 c84f 1f 20          tfr y,d
1175 1173 c851 34 06          pshs d
1176 1174 c853 ec f1          ldd [ ,s++]
```

```
1177 1175 c855 10 a3 e1      cmpd ,s++
1178 1176 c858 2e 05        bgt *+7
1179 1177 c85a cc 00 00      ldd #0
1180 1178 c85d 20 03        bra *+5
1181 1179 c85f cc 00 01      ldd #1
1182 1180 c862 10 83 00 00    cmpd #0
1183 1181 c866 10 27 00 19    lbeq cc49
1184 1182 c86a 7e c8 7d      jmp cc51
1185 1183                      cc48:
1186 1184 c86d f6 70 04      ldb n
1187 1185 c870 1d            sex
1188 1186 c871 c3 00 01      addd #1
1189 1187 c874 f7 70 04      stb n
1190 1188 c877 83 00 01      subd #1
1191 1189 c87a 7e c8 47      jmp cc50
1192 1190                      cc51:
1193 1191 c87d 7e c8 6d      jmp cc48
1194 1192 c880 7e c8 6d      jmp cc48
1195 1193                      cc49:
1196 1194 c883 39            rts
1197 1195
1198 1196                      * .global scan
1199 1197                      scan:
1200 1198 c884 32 7f          leas -1,s
1201 1199 c886 cc 00 01      ldd #1
1202 1200 c889 f7 70 05      stb k
1203 1201 c88c cc 00 00      ldd #0
1204 1202 c88f f7 70 06      stb u
1205 1203 c892 cc 00 01      ldd #1
1206 1204 c895 43            coma
1207 1205 c896 53            comb
1208 1206 c897 c3 00 01      addd #1
1209 1207 c89a f7 70 09      stb key
1210 1208 c89d cc 00 00      ldd #0
1211 1209 c8a0 f7 70 07      stb q
1212 1210 c8a3 cc 00 00      ldd #0
1213 1211 c8a6 fd 70 00      std i
1214 1212                      cc54:
1215 1213 c8a9 fc 70 00      ldd i
1216 1214 c8ac 34 06        pshs d
1217 1215 c8ae cc 00 06      ldd #6
1218 1216 c8b1 10 a3 e1      cmpd ,s++
1219 1217 c8b4 2e 05        bgt *+7
1220 1218 c8b6 cc 00 00      ldd #0
1221 1219 c8b9 20 03        bra *+5
1222 1220 c8bb cc 00 01      ldd #1
1223 1221 c8be 10 83 00 00    cmpd #0
1224 1222 c8c2 10 27 01 9e    lbeq cc53
1225 1223 c8c6 7e c8 d8      jmp cc55
1226 1224                      cc52:
1227 1225 c8c9 fc 70 00      ldd i
1228 1226 c8cc c3 00 01      addd #1
1229 1227 c8cf fd 70 00      std i
1230 1228 c8d2 83 00 01      subd #1
1231 1229 c8d5 7e c8 a9      jmp cc54
1232 1230                      cc55:
```

```
1233 1231 c8d8 fc 70 42      ldd port1
1234 1232 c8db 34 06        pshs d
1235 1233 c8dd f6 70 05      ldb k
1236 1234 c8e0 1d          sex
1237 1235 c8e1 43          coma
1238 1236 c8e2 53          comb
1239 1237 c8e3 e7 f1        stb [ ,s++ ]
1240 1238 c8e5 fc 70 40      ldd port2
1241 1239 c8e8 34 06        pshs d
1242 1240 c8ea cc 70 38      ldd #buffer
1243 1241 c8ed 34 06        pshs d
1244 1242 c8ef fc 70 00      ldd i
1245 1243 c8f2 e3 e1        addd ,s++
1246 1244 c8f4 34 06        pshs d
1247 1245 c8f6 e6 f1        ldb [ ,s++ ]
1248 1246 c8f8 1d          sex
1249 1247 c8f9 e7 f1        stb [ ,s++ ]
1250 1248 c8fb cc 70 38      ldd #buffer
1251 1249 c8fe 34 06        pshs d
1252 1250 c900 fc 70 00      ldd i
1253 1251 c903 e3 e1        addd ,s++
1254 1252 c905 34 06        pshs d
1255 1253 c907 e6 f1        ldb [ ,s++ ]
1256 1254 c909 1d          sex
1257 1255 c90a 34 06        pshs d
1258 1256 c90c cc 00 30      ldd #48
1259 1257 c90f 10 a3 e1      cmpd ,s++
1260 1258 c912 26 05        bne *+7
1261 1259 c914 cc 00 00      ldd #0
1262 1260 c917 20 03        bra *+5
1263 1261 c919 cc 00 01      ldd #1
1264 1262 c91c 10 83 00 00    cmpd #0
1265 1263 c920 10 27 00 2c    lbeq cc57 ;_ instruction flagged for non op
1266 1264 c924 cc 70 38      ldd #buffer
1267 1265 c927 34 06        pshs d
1268 1266 c929 fc 70 00      ldd i
1269 1267 c92c e3 e1        addd ,s++
1270 1268 c92e 34 06        pshs d
1271 1269 c930 e6 f1        ldb [ ,s++ ]
1272 1270 c932 1d          sex
1273 1271 c933 34 06        pshs d
1274 1272 c935 cc 00 38      ldd #56
1275 1273 c938 10 a3 e1      cmpd ,s++
1276 1274 c93b 26 05        bne *+7
1277 1275 c93d cc 00 00      ldd #0
1278 1276 c940 20 03        bra *+5
1279 1277 c942 cc 00 01      ldd #1
1280 1278 c945 10 83 00 00    cmpd #0
1281 1279 c949 10 27 00 03    lbeq cc57 ;_ instruction flagged for non op
1282 1280 c94d cc 00 01      ldd #1
1283 1281                      cc57:
1284 1282 c950 10 83 00 00    cmpd #0
1285 1283 c954 10 27 00 2c    lbeq cc58 ;_ instruction flagged for non op
1286 1284 c958 cc 70 38      ldd #buffer
1287 1285 c95b 34 06        pshs d
1288 1286 c95d fc 70 00      ldd i
```

```
1289 1287 c960 e3 e1           addd ,s++
1290 1288 c962 34 06           pshs d
1291 1289 c964 e6 f1           ldb [ ,s++]
1292 1290 c966 1d             sex
1293 1291 c967 34 06           pshs d
1294 1292 c969 cc 00 70           ldd #112
1295 1293 c96c 10 a3 e1           cmpd ,s++
1296 1294 c96f 26 05           bne *+7
1297 1295 c971 cc 00 00           ldd #0
1298 1296 c974 20 03           bra *+5
1299 1297 c976 cc 00 01           ldd #1
1300 1298 c979 10 83 00 00           cmpd #0
1301 1299 c97d 10 27 00 03           lbeq cc58 ; instruction flagged for non op
1302 1300 c981 cc 00 01           ldd #1
1303 1301                   cc58:
1304 1302 c984 10 83 00 00           cmpd #0
1305 1303 c988 10 27 00 0d           lbeq cc56
1306 1304 c98c cc 00 02           ldd #2
1307 1305 c98f 34 06           pshs d
1308 1306 c991 bd c8 41           jsr delay_ms
1309 1307 c994 32 62           leas 2,s
1310 1308 c996 7e c9 9c           jmp cc59
1311 1309                   cc56:
1312 1310 c999 bd c8 34           jsr delay_nu
1313 1311                   cc59:
1314 1312 c99c fc 70 40           ldd port2
1315 1313 c99f 34 06           pshs d
1316 1314 c9a1 cc 00 00           ldd #0
1317 1315 c9a4 e7 f1           stb [ ,s++]
1318 1316 c9a6 cc 00 01           ldd #1
1319 1317 c9a9 34 06           pshs d
1320 1318 c9ab bd c8 41           jsr delay_ms
1321 1319 c9ae 32 62           leas 2,s
1322 1320 c9b0 fc 70 44           ldd port0
1323 1321 c9b3 34 06           pshs d
1324 1322 c9b5 e6 f1           ldb [ ,s++]
1325 1323 c9b7 1d             sex
1326 1324 c9b8 f7 70 08           stb o
1327 1325 c9bb cc 00 00           ldd #0
1328 1326 c9be f7 70 04           stb n
1329 1327                   cc62:
1330 1328 c9c1 f6 70 04           ldb n
1331 1329 c9c4 1d             sex
1332 1330 c9c5 34 06           pshs d
1333 1331 c9c7 cc 00 06           ldd #6
1334 1332 c9ca 10 a3 e1           cmpd ,s++
1335 1333 c9cd 2e 05           bgt *+7
1336 1334 c9cf cc 00 00           ldd #0
1337 1335 c9d2 20 03           bra *+5
1338 1336 c9d4 cc 00 01           ldd #1
1339 1337 c9d7 10 83 00 00           cmpd #0
1340 1338 c9db 10 27 00 6b           lbeq cc61
1341 1339 c9df 7e c9 f2           jmp cc63
1342 1340                   cc60:
1343 1341 c9e2 f6 70 04           ldb n
1344 1342 c9e5 1d             sex
```

```
1345 1343 c9e6 c3 00 01      addd #1
1346 1344 c9e9 f7 70 04      stb n
1347 1345 c9ec 83 00 01      subd #1
1348 1346 c9ef 7e c9 c1      jmp cc62
1349 1347                      cc63:
1350 1348 c9f2 f6 70 08      ldb o
1351 1349 c9f5 1d            sex
1352 1350 c9f6 34 06        pshs d
1353 1351 c9f8 cc 00 01      ldd #1
1354 1352 c9fb a4 e0        anda ,s+
1355 1353 c9fd e4 e0        andb ,s+
1356 1354 c9ff 34 06        pshs d
1357 1355 ca01 cc 00 00      ldd #0
1358 1356 ca04 10 a3 e1      cmpd ,s++
1359 1357 ca07 27 05        beq *+7
1360 1358 ca09 cc 00 00      ldd #0
1361 1359 ca0c 20 03        bra *+5
1362 1360 ca0e cc 00 01      ldd #1
1363 1361 ca11 10 83 00 00    cmpd #0
1364 1362 ca15 10 27 00 0a    lbeq cc64
1365 1363 ca19 f6 70 07      ldb q
1366 1364 ca1c 1d            sex
1367 1365 ca1d f7 70 09      stb key
1368 1366 ca20 7e ca 30      jmp cc65
1369 1367                      cc64:
1370 1368 ca23 f6 70 07      ldb q
1371 1369 ca26 1d            sex
1372 1370 ca27 c3 00 01      addd #1
1373 1371 ca2a f7 70 07      stb q
1374 1372 ca2d 83 00 01      subd #1
1375 1373                      cc65:
1376 1374 ca30 f6 70 08      ldb o
1377 1375 ca33 1d            sex
1378 1376 ca34 34 06        pshs d
1379 1377 ca36 cc 00 01      ldd #1
1380 1378 ca39 5a            decb
1381 1379 ca3a 2d 06        blt *+8
1382 1380 ca3c 64 e4        lsr ,s
1383 1381 ca3e 66 61        ror 1,s
1384 1382 ca40 20 f7        bra *-7
1385 1383 ca42 35 06        puls d
1386 1384 ca44 f7 70 08      stb o
1387 1385 ca47 7e c9 e2      jmp cc60
1388 1386                      cc61:
1389 1387 ca4a f6 70 05      ldb k
1390 1388 ca4d 1d            sex
1391 1389 ca4e 34 06        pshs d
1392 1390 ca50 cc 00 01      ldd #1
1393 1391 ca53 5a            decb
1394 1392 ca54 2d 06        blt *+8
1395 1393 ca56 68 61        asl 1,s
1396 1394 ca58 69 e4        rol ,s
1397 1395 ca5a 20 f7        bra *-7
1398 1396 ca5c 35 06        puls d
1399 1397 ca5e f7 70 05      stb k
1400 1398 ca61 7e c8 c9      jmp cc52
```

```
1401 1399
1402 1400 ca64 fc 70 44 cc53:
1403 1401 ca67 34 06 pshs d
1404 1402 ca69 e6 f1 ldb [ ,s++ ]
1405 1403 ca6b 1d sex
1406 1404 ca6c f7 70 08 stb o
1407 1405 ca6f f6 70 08 ldb o
1408 1406 ca72 1d sex
1409 1407 ca73 34 06 pshs d
1410 1408 ca75 cc 00 40 ldd #64
1411 1409 ca78 a4 e0 anda ,s+
1412 1410 ca7a e4 e0 andb ,s+
1413 1411 ca7c 34 06 pshs d
1414 1412 ca7e cc 00 00 ldd #0
1415 1413 ca81 10 a3 e1 cmpd ,s++
1416 1414 ca84 27 05 beq *+7
1417 1415 ca86 cc 00 00 ldd #0
1418 1416 ca89 20 03 bra *+5
1419 1417 ca8b cc 00 01 ldd #1
1420 1418 ca8e 10 83 00 00 cmpd #0
1421 1419 ca92 10 27 00 06 lbeq cc66
1422 1420 ca96 cc 00 24 ldd #36
1423 1421 ca99 f7 70 09 stb key
1424 1422 cc66:
1425 1423 ca9c 31 e4 leay 0,s
1426 1424 ca9e 1f 20 tfr y,d
1427 1425 caa0 34 06 pshs d
1428 1426 caa2 f6 70 09 ldb key
1429 1427 caa5 1d sex
1430 1428 caa6 e7 f1 stb [ ,s++ ]
1431 1429 caa8 31 e4 leay 0,s
1432 1430 caaa 1f 20 tfr y,d
1433 1431 caac 34 06 pshs d
1434 1432 caae e6 f1 ldb [ ,s++ ]
1435 1433 cab0 1d sex
1436 1434 cab1 32 61 leas 1,s
1437 1435 cab3 39 rts
1438 1436
1439 1437 * .global dot_addr
1440 1438 dot_addr:
1441 1439 cab4 cc 70 38 ldd #buffer
1442 1440 cab7 34 06 pshs d
1443 1441 cab9 cc 00 00 ldd #0
1444 1442 cabc e3 e1 addd ,s++
1445 1443 cabe 34 06 pshs d
1446 1444 cac0 cc 70 38 ldd #buffer
1447 1445 cac3 34 06 pshs d
1448 1446 cac5 cc 00 00 ldd #0
1449 1447 cac8 e3 e1 addd ,s++
1450 1448 caca 34 06 pshs d
1451 1449 cacc e6 f1 ldb [ ,s++ ]
1452 1450 cace 1d sex
1453 1451 cacf 34 06 pshs d
1454 1452 cad1 cc 00 40 ldd #64
1455 1453 cad4 43 coma
1456 1454 cad5 53 comb
```

1457	1455	cad6	a4	e0	anda ,s+
1458	1456	cad8	e4	e0	andb ,s+
1459	1457	cada	e7	f1	stb [,s++]
1460	1458	cadc	cc	70 38	ldd #buffer
1461	1459	cadf	34	06	pshs d
1462	1460	cae1	cc	00 01	ldd #1
1463	1461	cae4	e3	e1	addir ,s++
1464	1462	cae6	34	06	pshs d
1465	1463	cae8	cc	70 38	ldd #buffer
1466	1464	caeb	34	06	pshs d
1467	1465	caed	cc	00 01	ldd #1
1468	1466	caf0	e3	e1	addir ,s++
1469	1467	caf2	34	06	pshs d
1470	1468	caf4	e6	f1	ldb [,s++]
1471	1469	caf6	1d		sex
1472	1470	caf7	34	06	pshs d
1473	1471	caf9	cc	00 40	ldd #64
1474	1472	cafC	43		coma
1475	1473	cafD	53		comb
1476	1474	cafe	a4	e0	anda ,s+
1477	1475	cb00	e4	e0	andb ,s+
1478	1476	cb02	e7	f1	stb [,s++]
1479	1477	cb04	cc	70 38	ldd #buffer
1480	1478	cb07	34	06	pshs d
1481	1479	cb09	cc	00 02	ldd #2
1482	1480	cb0c	e3	e1	addir ,s++
1483	1481	cb0e	34	06	pshs d
1484	1482	cb10	cc	70 38	ldd #buffer
1485	1483	cb13	34	06	pshs d
1486	1484	cb15	cc	00 02	ldd #2
1487	1485	cb18	e3	e1	addir ,s++
1488	1486	cb1a	34	06	pshs d
1489	1487	cb1c	e6	f1	ldb [,s++]
1490	1488	cb1e	1d		sex
1491	1489	cb1f	34	06	pshs d
1492	1490	cb21	cc	00 40	ldd #64
1493	1491	cb24	aa	e0	ora ,s+
1494	1492	cb26	ea	e0	orb ,s+
1495	1493	cb28	e7	f1	stb [,s++]
1496	1494	cb2a	cc	70 38	ldd #buffer
1497	1495	cb2d	34	06	pshs d
1498	1496	cb2f	cc	00 03	ldd #3
1499	1497	cb32	e3	e1	addir ,s++
1500	1498	cb34	34	06	pshs d
1501	1499	cb36	cc	70 38	ldd #buffer
1502	1500	cb39	34	06	pshs d
1503	1501	cb3b	cc	00 03	ldd #3
1504	1502	cb3e	e3	e1	addir ,s++
1505	1503	cb40	34	06	pshs d
1506	1504	cb42	e6	f1	ldb [,s++]
1507	1505	cb44	1d		sex
1508	1506	cb45	34	06	pshs d
1509	1507	cb47	cc	00 40	ldd #64
1510	1508	cb4a	aa	e0	ora ,s+
1511	1509	cb4c	ea	e0	orb ,s+
1512	1510	cb4e	e7	f1	stb [,s++]

```
1513 1511 cb50 cc 70 38      ldd #buffer
1514 1512 cb53 34 06       pshs d
1515 1513 cb55 cc 00 04      ldd #4
1516 1514 cb58 e3 e1       addd ,s++
1517 1515 cb5a 34 06       pshs d
1518 1516 cb5c cc 70 38      ldd #buffer
1519 1517 cb5f 34 06       pshs d
1520 1518 cb61 cc 00 04      ldd #4
1521 1519 cb64 e3 e1       addd ,s++
1522 1520 cb66 34 06       pshs d
1523 1521 cb68 e6 f1       ldb [,s++]
1524 1522 cb6a 1d          sex
1525 1523 cb6b 34 06       pshs d
1526 1524 cb6d cc 00 40      ldd #64
1527 1525 cb70 aa e0       ora ,s+
1528 1526 cb72 ea e0       orb ,s+
1529 1527 cb74 e7 f1       stb [,s++]
1530 1528 cb76 cc 70 38      ldd #buffer
1531 1529 cb79 34 06       pshs d
1532 1530 cb7b cc 00 05      ldd #5
1533 1531 cb7e e3 e1       addd ,s++
1534 1532 cb80 34 06       pshs d
1535 1533 cb82 cc 70 38      ldd #buffer
1536 1534 cb85 34 06       pshs d
1537 1535 cb87 cc 00 05      ldd #5
1538 1536 cb8a e3 e1       addd ,s++
1539 1537 cb8c 34 06       pshs d
1540 1538 cb8e e6 f1       ldb [,s++]
1541 1539 cb90 1d          sex
1542 1540 cb91 34 06       pshs d
1543 1541 cb93 cc 00 40      ldd #64
1544 1542 cb96 aa e0       ora ,s+
1545 1543 cb98 ea e0       orb ,s+
1546 1544 cb9a e7 f1       stb [,s++]
1547 1545 cb9c 39          rts
1548 1546
1549 1547
1550 1548
1551 1549 cb9d cc 70 38      * .global dot_data
1552 1550 cba0 34 06       dot_data:
1553 1551 cba2 cc 00 00      ldd #buffer
1554 1552 cba5 e3 e1       pshs d
1555 1553 cba7 34 06       ldd #0
1556 1554 cba9 cc 70 38      addd ,s++
1557 1555 cbac 34 06       pshs d
1558 1556 cbae cc 00 00      pshs d
1559 1557 cbb1 e3 e1       ldd #buffer
1560 1558 cbb3 34 06       pshs d
1561 1559 cbb5 e6 f1       pshs d
1562 1560 cbb7 1d          ldd #0
1563 1561 cbb8 34 06       addd ,s++
1564 1562 cbba cc 00 40      ldb [,s++]
1565 1563 cbbd aa e0       sex
1566 1564 cbff ea e0       pshs d
1567 1565 cbc1 e7 f1       ldd #64
1568 1566 cbc3 cc 70 38      ora ,s+
1569 1567 cbcc 34 06       orb ,s+
1570 1571 cbcd aa e0       stb [,s++]
1571 1572 cbce ea e0       ldd #buffer
```

1569	1567	cbc6	34	06	pshs d
1570	1568	cbc8	cc	00	01 ldd #1
1571	1569	cbcb	e3	e1	addd ,s++
1572	1570	cbcd	34	06	pshs d
1573	1571	cbcfc	cc	70	38 ldd #buffer
1574	1572	cbd2	34	06	pshs d
1575	1573	cbd4	cc	00	01 ldd #1
1576	1574	cbd7	e3	e1	addd ,s++
1577	1575	cbd9	34	06	pshs d
1578	1576	cbdb	e6	f1	ldb [,s++]
1579	1577	cbdd	1d		sex
1580	1578	cbde	34	06	pshs d
1581	1579	cbe0	cc	00	40 ldd #64
1582	1580	cbe3	aa	e0	ora ,s+
1583	1581	cbe5	ea	e0	orb ,s+
1584	1582	cbe7	e7	f1	stb [,s++]
1585	1583	cbe9	cc	70	38 ldd #buffer
1586	1584	cbec	34	06	pshs d
1587	1585	cbee	cc	00	02 ldd #2
1588	1586	cbf1	e3	e1	addd ,s++
1589	1587	cbf3	34	06	pshs d
1590	1588	cbf5	cc	70	38 ldd #buffer
1591	1589	cbf8	34	06	pshs d
1592	1590	cbfa	cc	00	02 ldd #2
1593	1591	cbfd	e3	e1	addd ,s++
1594	1592	cbff	34	06	pshs d
1595	1593	cc01	e6	f1	ldb [,s++]
1596	1594	cc03	1d		sex
1597	1595	cc04	34	06	pshs d
1598	1596	cc06	cc	00	40 ldd #64
1599	1597	cc09	43		coma
1600	1598	cc0a	53		comb
1601	1599	cc0b	a4	e0	anda ,s+
1602	1600	cc0d	e4	e0	andb ,s+
1603	1601	cc0f	e7	f1	stb [,s++]
1604	1602	cc11	cc	70	38 ldd #buffer
1605	1603	cc14	34	06	pshs d
1606	1604	cc16	cc	00	03 ldd #3
1607	1605	cc19	e3	e1	addd ,s++
1608	1606	cc1b	34	06	pshs d
1609	1607	cc1d	cc	70	38 ldd #buffer
1610	1608	cc20	34	06	pshs d
1611	1609	cc22	cc	00	03 ldd #3
1612	1610	cc25	e3	e1	addd ,s++
1613	1611	cc27	34	06	pshs d
1614	1612	cc29	e6	f1	ldb [,s++]
1615	1613	cc2b	1d		sex
1616	1614	cc2c	34	06	pshs d
1617	1615	cc2e	cc	00	40 ldd #64
1618	1616	cc31	43		coma
1619	1617	cc32	53		comb
1620	1618	cc33	a4	e0	anda ,s+
1621	1619	cc35	e4	e0	andb ,s+
1622	1620	cc37	e7	f1	stb [,s++]
1623	1621	cc39	cc	70	38 ldd #buffer
1624	1622	cc3c	34	06	pshs d

```

1625 1623 cc3e cc 00 04      ldd #4
1626 1624 cc41 e3 e1      addd ,s++
1627 1625 cc43 34 06      pshs d
1628 1626 cc45 cc 70 38      ldd #buffer
1629 1627 cc48 34 06      pshs d
1630 1628 cc4a cc 00 04      ldd #4
1631 1629 cc4d e3 e1      addd ,s++
1632 1630 cc4f 34 06      pshs d
1633 1631 cc51 e6 f1      ldb [,s++]
1634 1632 cc53 1d      sex
1635 1633 cc54 34 06      pshs d
1636 1634 cc56 cc 00 40      ldd #64
1637 1635 cc59 43      coma
1638 1636 cc5a 53      comb
1639 1637 cc5b a4 e0      anda ,s+
1640 1638 cc5d e4 e0      andb ,s+
1641 1639 cc5f e7 f1      stb [,s++]
1642 1640 cc61 cc 70 38      ldd #buffer
1643 1641 cc64 34 06      pshs d
1644 1642 cc66 cc 00 05      ldd #5
1645 1643 cc69 e3 e1      addd ,s++
1646 1644 cc6b 34 06      pshs d
1647 1645 cc6d cc 70 38      ldd #buffer
1648 1646 cc70 34 06      pshs d
1649 1647 cc72 cc 00 05      ldd #5
1650 1648 cc75 e3 e1      addd ,s++
1651 1649 cc77 34 06      pshs d
1652 1650 cc79 e6 f1      ldb [,s++]
1653 1651 cc7b 1d      sex
1654 1652 cc7c 34 06      pshs d
1655 1653 cc7e cc 00 40      ldd #64
1656 1654 cc81 43      coma
1657 1655 cc82 53      comb
1658 1656 cc83 a4 e0      anda ,s+
1659 1657 cc85 e4 e0      andb ,s+
1660 1658 cc87 e7 f1      stb [,s++]
1661 1659 cc89 39      rts
1662 1660
1663 1661      * .global hex4
1664 1662      hex4:
1665 1663 cc8a 31 62      leay 2,s
1666 1664 cc8c 1f 20      tfr y,d
1667 1665 cc8e 34 06      pshs d
1668 1666 cc90 ec f1      ldd [,s++]
1669 1667 cc92 fd 70 17      std temp16
1670 1668 cc95 cc 70 38      ldd #buffer
1671 1669 cc98 34 06      pshs d
1672 1670 cc9a cc 00 02      ldd #2
1673 1671 cc9d e3 e1      addd ,s++
1674 1672 cc9f 34 06      pshs d
1675 1673 cca1 cc c1 0d      ldd #convert
1676 1674 cca4 34 06      pshs d
1677 1675 cca6 fc 70 17      ldd temp16
1678 1676 cca9 34 06      pshs d
1679 1677 ccab cc 00 0f      ldd #15
1680 1678 ccae a4 e0      anda ,s+

```

1681	1679	ccb0	e4	e0	andb ,s+
1682	1680	ccb2	e3	e1	addd ,s++
1683	1681	ccb4	34	06	pshs d
1684	1682	ccb6	e6	f1	ldb [,s++]
1685	1683	ccb8	1d		sex
1686	1684	ccb9	e7	f1	stb [,s++]
1687	1685	ccbb	cc	70 17	ldd #temp16
1688	1686	ccbe	34	06	pshs d
1689	1687	ccc0	cc	00 04	ldd #4
1690	1688	ccc3	ae	f4	ldx [,s]
1691	1689	ccc5	34	10	pshs x
1692	1690	ccc7	5a		decb
1693	1691	ccc8	2d	06	blt *+8
1694	1692	cccc	64	e4	lsr ,s
1695	1693	cccc	66	61	ror 1,s
1696	1694	ccce	20	f7	bra *-7
1697	1695	ccd0	35	06	puls d
1698	1696	ccd2	fd	70 17	std temp16
1699	1697	ccd5	cc	70 38	ldd #buffer
1700	1698	ccd8	34	06	pshs d
1701	1699	ccda	cc	00 03	ldd #3
1702	1700	ccdd	e3	e1	addd ,s++
1703	1701	ccdf	34	06	pshs d
1704	1702	cce1	cc	c1 0d	ldd #convert
1705	1703	cce4	34	06	pshs d
1706	1704	cce6	fc	70 17	ldd temp16
1707	1705	cce9	34	06	pshs d
1708	1706	cceb	cc	00 0f	ldd #15
1709	1707	ccee	a4	e0	anda ,s+
1710	1708	ccf0	e4	e0	andb ,s+
1711	1709	ccf2	e3	e1	addd ,s++
1712	1710	ccf4	34	06	pshs d
1713	1711	ccf6	e6	f1	ldb [,s++]
1714	1712	ccf8	1d		sex
1715	1713	ccf9	e7	f1	stb [,s++]
1716	1714	ccfb	cc	70 17	ldd #temp16
1717	1715	ccfe	34	06	pshs d
1718	1716	cd00	cc	00 04	ldd #4
1719	1717	cd03	ae	f4	ldx [,s]
1720	1718	cd05	34	10	pshs x
1721	1719	cd07	5a		decb
1722	1720	cd08	2d	06	blt *+8
1723	1721	cd0a	64	e4	lsr ,s
1724	1722	cd0c	66	61	ror 1,s
1725	1723	cd0e	20	f7	bra *-7
1726	1724	cd10	35	06	puls d
1727	1725	cd12	fd	70 17	std temp16
1728	1726	cd15	cc	70 38	ldd #buffer
1729	1727	cd18	34	06	pshs d
1730	1728	cd1a	cc	00 04	ldd #4
1731	1729	cd1d	e3	e1	addd ,s++
1732	1730	cd1f	34	06	pshs d
1733	1731	cd21	cc	c1 0d	ldd #convert
1734	1732	cd24	34	06	pshs d
1735	1733	cd26	fc	70 17	ldd temp16
1736	1734	cd29	34	06	pshs d

```

1737 1735 cd2b cc 00 0f      ldd #15
1738 1736 cd2e a4 e0      anda ,s+
1739 1737 cd30 e4 e0      andb ,s+
1740 1738 cd32 e3 e1      addd ,s++
1741 1739 cd34 34 06      pshs d
1742 1740 cd36 e6 f1      ldb [,s++]
1743 1741 cd38 1d      sex
1744 1742 cd39 e7 f1      stb [,s++]
1745 1743 cd3b cc 70 17      ldd #temp16
1746 1744 cd3e 34 06      pshs d
1747 1745 cd40 cc 00 04      ldd #4
1748 1746 cd43 ae f4      ldx [,s]
1749 1747 cd45 34 10      pshs x
1750 1748 cd47 5a      decb
1751 1749 cd48 2d 06      blt *+8
1752 1750 cd4a 64 e4      lsr ,s
1753 1751 cd4c 66 61      ror 1,s
1754 1752 cd4e 20 f7      bra *-7
1755 1753 cd50 35 06      puls d
1756 1754 cd52 fd 70 17      std temp16
1757 1755 cd55 cc 70 38      ldd #buffer
1758 1756 cd58 34 06      pshs d
1759 1757 cd5a cc 00 05      ldd #5
1760 1758 cd5d e3 e1      addd ,s++
1761 1759 cd5f 34 06      pshs d
1762 1760 cd61 cc c1 0d      ldd #convert
1763 1761 cd64 34 06      pshs d
1764 1762 cd66 fc 70 17      ldd temp16
1765 1763 cd69 34 06      pshs d
1766 1764 cd6b cc 00 0f      ldd #15
1767 1765 cd6e a4 e0      anda ,s+
1768 1766 cd70 e4 e0      andb ,s+
1769 1767 cd72 e3 e1      addd ,s++
1770 1768 cd74 34 06      pshs d
1771 1769 cd76 e6 f1      ldb [,s++]
1772 1770 cd78 1d      sex
1773 1771 cd79 e7 f1      stb [,s++]
1774 1772 cd7b 32 66      leas 6,s
1775 1773 cd7d 39      rts
1776
1777 1775      * .global address_
1778 1776      address_:
1779 1777 cd7e fc 70 19      ldd PC
1780 1778 cd81 fd 70 17      std temp16
1781 1779 cd84 fc 70 17      ldd temp16
1782 1780 cd87 34 06      pshs d
1783 1781 cd89 bd cc 8a      jsr hex4
1784 1782 cd8c 32 62      leas 2,s
1785 1783 cd8e 39      rts
1786
1787 1785      * .global data_dis
1788 1786      data_dis:
1789 1787 cd8f fc 70 19      ldd PC
1790 1788 cd92 fd 70 46      std dptr
1791 1789 cd95 fc 70 46      ldd dptr
1792 1790 cd98 34 06      pshs d

```

```
1793 1791 cd9a e6 f1           ldb  [,s++]
1794 1792 cd9c 1d             sex
1795 1793 cd9d f7 70 04       stb  n
1796 1794 cda0 cc 70 38       ldd  #buffer
1797 1795 cda3 34 06         pshs d
1798 1796 cda5 cc 00 00       ldd  #0
1799 1797 cda8 e3 e1         addd ,s++
1800 1798 cdaa 34 06         pshs d
1801 1799 cdac cc c1 0d       ldd  #convert
1802 1800 cdaf 34 06         pshs d
1803 1801 cdb1 f6 70 04       ldb  n
1804 1802 cdb4 1d             sex
1805 1803 cdb5 34 06         pshs d
1806 1804 cdb7 cc 00 0f       ldd  #15
1807 1805 cdba a4 e0         anda ,s+
1808 1806 cdbc e4 e0         andb ,s+
1809 1807 cdbe e3 e1         addd ,s++
1810 1808 cdc0 34 06         pshs d
1811 1809 cdc2 e6 f1         ldb  [,s++]
1812 1810 cdc4 1d             sex
1813 1811 cdc5 e7 f1         stb  [,s++]
1814 1812 cdc7 f6 70 04       ldb  n
1815 1813 cdca 1d             sex
1816 1814 cdcb 34 06         pshs d
1817 1815 ccd4 cc 00 04       ldd  #4
1818 1816 cdd0 5a             decb
1819 1817 cdd1 2d 06         blt  *+8
1820 1818 cdd3 64 e4         lsr  ,s
1821 1819 cdd5 66 61         ror  1,s
1822 1820 cdd7 20 f7         bra  *-7
1823 1821 cdd9 35 06         puls d
1824 1822 cddb f7 70 04       stb  n
1825 1823 cdde cc 70 38       ldd  #buffer
1826 1824 cde1 34 06         pshs d
1827 1825 cde3 cc 00 01       ldd  #1
1828 1826 cde6 e3 e1         addd ,s++
1829 1827 cde8 34 06         pshs d
1830 1828 cdea cc c1 0d       ldd  #convert
1831 1829 cded 34 06         pshs d
1832 1830 cdef f6 70 04       ldb  n
1833 1831 cdf2 1d             sex
1834 1832 cdf3 34 06         pshs d
1835 1833 cdf5 cc 00 0f       ldd  #15
1836 1834 cdf8 a4 e0         anda ,s+
1837 1835 cdfa e4 e0         andb ,s+
1838 1836 cdfc e3 e1         addd ,s++
1839 1837 cefe 34 06         pshs d
1840 1838 ce00 e6 f1         ldb  [,s++]
1841 1839 ce02 1d             sex
1842 1840 ce03 e7 f1         stb  [,s++]
1843 1841 ce05 bd cb 9d       jsr  dot_data
1844 1842 ce08 39             rts
1845 1843
1846 1844                   * .global read_mem
1847 1845                   read_mem:
1848 1846 ce09 bd cd 7e       jsr  address_
```

```
1849 1847 ce0c bd cd 8f          jsr data_dis
1850 1848 ce0f 39              rts
1851 1849
1852 1850                      * .global key_addr
1853 1851                      key_addr:
1854 1852 ce10 cc 00 01          ldd #1
1855 1853 ce13 f7 70 27          stb state
1856 1854 ce16 bd ce 09          jsr read_mem
1857 1855 ce19 bd ca b4          jsr dot_addr
1858 1856 ce1c cc 00 00          ldd #0
1859 1857 ce1f f7 70 0b          stb hit
1860 1858 ce22 39              rts
1861 1859
1862 1860                      * .global key_data
1863 1861                      key_data:
1864 1862 ce23 bd ce 09          jsr read_mem
1865 1863 ce26 bd cb 9d          jsr dot_data
1866 1864 ce29 cc 00 00          ldd #0
1867 1865 ce2c f7 70 0b          stb hit
1868 1866 ce2f cc 00 02          ldd #2
1869 1867 ce32 f7 70 27          stb state
1870 1868 ce35 39              rts
1871 1869
1872 1870                      * .global key_plus
1873 1871                      key_plus:
1874 1872 ce36 f6 70 27          ldb state
1875 1873 ce39 1d              sex
1876 1874 ce3a 34 06          pshs d
1877 1875 ce3c cc 00 01          ldd #1
1878 1876 ce3f 10 a3 e1          cmpd ,s++
1879 1877 ce42 27 05          beq *+7
1880 1878 ce44 cc 00 00          ldd #0
1881 1879 ce47 20 03          bra *+5
1882 1880 ce49 cc 00 01          ldd #1
1883 1881 ce4c 10 83 00 00          cmpd #0
1884 1882 ce50 10 26 00 24          lbne cc68
1885 1883 ce54 f6 70 27          ldb state
1886 1884 ce57 1d              sex
1887 1885 ce58 34 06          pshs d
1888 1886 ce5a cc 00 02          ldd #2
1889 1887 ce5d 10 a3 e1          cmpd ,s++
1890 1888 ce60 27 05          beq *+7
1891 1889 ce62 cc 00 00          ldd #0
1892 1890 ce65 20 03          bra *+5
1893 1891 ce67 cc 00 01          ldd #1
1894 1892 ce6a 10 83 00 00          cmpd #0
1895 1893 ce6e 10 26 00 06          lbne cc68
1896 1894 ce72 cc 00 00          ldd #0
1897 1895 ce75 7e ce 7b          jmp cc69
1898 1896                      cc68:
1899 1897 ce78 cc 00 01          ldd #1
1900 1898                      cc69:
1901 1899 ce7b 10 83 00 00          cmpd #0
1902 1900 ce7f 10 27 00 12          lbeq cc67
1903 1901 ce83 fc 70 19          ldd PC
1904 1902 ce86 c3 00 01          addd #1
```

1905	1903	ce89	fd	70	19	std PC
1906	1904	ce8c	83	00	01	subd #1
1907	1905	ce8f	bd	ce	09	jsr read_mem
1908	1906	ce92	bd	ce	23	jsr key_data
1909	1907					cc67:
1910	1908	ce95	f6	70	27	ldb state
1911	1909	ce98	1d			sex
1912	1910	ce99	34	06		pshs d
1913	1911	ce9b	cc	00	04	ldd #4
1914	1912	ce9e	10	a3	e1	cmpd ,s++
1915	1913	cea1	27	05		beq *+7
1916	1914	cea3	cc	00	00	ldd #0
1917	1915	cea6	20	03		bra *+5
1918	1916	cea8	cc	00	01	ldd #1
1919	1917	ceab	10	83	00 00	cmpd #0
1920	1918	ceaf	10	27	00 12	lbeq cc70
1921	1919	ceb3	fc	70	1d	ldd num
1922	1920	ceb6	fd	70	1f	std start
1923	1921	ceb9	cc	00	00	ldd #0
1924	1922	cebc	f7	70	0b	stb hit
1925	1923	cebf	cc	00	01	ldd #1
1926	1924	cec2	f7	70	0c	stb positive
1927	1925					cc70:
1928	1926	cec5	f6	70	27	ldb state
1929	1927	cec8	1d			sex
1930	1928	cec9	34	06		pshs d
1931	1929	cecb	cc	00	05	ldd #5
1932	1930	cece	10	a3	e1	cmpd ,s++
1933	1931	ced1	27	05		beq *+7
1934	1932	ced3	cc	00	00	ldd #0
1935	1933	ced6	20	03		bra *+5
1936	1934	ced8	cc	00	01	ldd #1
1937	1935	cedb	10	83	00 00	cmpd #0
1938	1936	cedf	10	27	00 24	lbeq cc71
1939	1937	cee3	cc	00	06	ldd #6
1940	1938	cee6	f7	70	27	stb state
1941	1939	cee9	fc	70	1d	ldd num
1942	1940	ceec	fd	70	1f	std start
1943	1941	ceef	cc	00	00	ldd #0
1944	1942	cef2	f7	70	0b	stb hit
1945	1943	cef5	cc	70	38	ldd #buffer
1946	1944	cef8	34	06		pshs d
1947	1945	cefa	cc	00	00	ldd #0
1948	1946	cefd	e3	e1		addd ,s++
1949	1947	ceff	34	06		pshs d
1950	1948	cf01	cc	00	8f	ldd #143
1951	1949	cf04	e7	f1		stb [,s++]
1952	1950	cf06	39			rts
1953	1951					cc71:
1954	1952	cf07	f6	70	27	ldb state
1955	1953	cf0a	1d			sex
1956	1954	cf0b	34	06		pshs d
1957	1955	cf0d	cc	00	06	ldd #6
1958	1956	cf10	10	a3	e1	cmpd ,s++
1959	1957	cf13	27	05		beq *+7
1960	1958	cf15	cc	00	00	ldd #0

```
1961 1959 cf18 20 03          bra *+5
1962 1960 cf1a cc 00 01      ldd #1
1963 1961 cf1d 10 83 00 00    cmpd #0
1964 1962 cf21 10 27 00 43    lbeq cc72
1965 1963 cf25 cc 00 07      ldd #7
1966 1964 cf28 f7 70 27      stb state
1967 1965 cf2b fc 70 1d      ldd num
1968 1966 cf2e fd 70 21      std end
1969 1967 cf31 cc 00 00      ldd #0
1970 1968 cf34 f7 70 0b      stb hit
1971 1969 cf37 cc 70 38      ldd #buffer
1972 1970 cf3a 34 06        pshs d
1973 1971 cf3c cc 00 00      ldd #0
1974 1972 cf3f e3 e1        addd ,s++
1975 1973 cf41 34 06        pshs d
1976 1974 cf43 cc 00 b3      ldd #179
1977 1975 cf46 e7 f1        stb [,s++]
1978 1976 cf48 fc 70 21      ldd end
1979 1977 cf4b 34 06        pshs d
1980 1978 cf4d fc 70 1f      ldd start
1981 1979 cf50 10 a3 e1      cmpd ,s++
1982 1980 cf53 2c 05        bge *+7
1983 1981 cf55 cc 00 00      ldd #0
1984 1982 cf58 20 03        bra *+5
1985 1983 cf5a cc 00 01      ldd #1
1986 1984 cf5d 10 83 00 00    cmpd #0
1987 1985 cf61 10 27 00 03    lbeq cc73
1988 1986 cf65 bd d0 c6      jsr print_er
1989 1987                      cc73:
1990 1988                      cc72:
1991 1989 cf68 39            rts
1992 1990
1993 1991                      * .global key_minu
1994 1992                      key_minu:
1995 1993 cf69 f6 70 27      ldb state
1996 1994 cf6c 1d            sex
1997 1995 cf6d 34 06        pshs d
1998 1996 cf6f cc 00 01      ldd #1
1999 1997 cf72 10 a3 e1      cmpd ,s++
2000 1998 cf75 27 05        beq *+7
2001 1999 cf77 cc 00 00      ldd #0
2002 2000 cf7a 20 03        bra *+5
2003 2001 cf7c cc 00 01      ldd #1
2004 2002 cf7f 34 06        pshs d
2005 2003 cf81 f6 70 27      ldb state
2006 2004 cf84 1d            sex
2007 2005 cf85 34 06        pshs d
2008 2006 cf87 cc 00 02      ldd #2
2009 2007 cf8a 10 a3 e1      cmpd ,s++
2010 2008 cf8d 27 05        beq *+7
2011 2009 cf8f cc 00 00      ldd #0
2012 2010 cf92 20 03        bra *+5
2013 2011 cf94 cc 00 01      ldd #1
2014 2012 cf97 aa e0        ora ,s+
2015 2013 cf99 ea e0        orb ,s+
2016 2014 cf9b 10 83 00 00    cmpd #0
```

```
2017 2015 cf9f 10 27 00 12      lbeq cc74
2018 2016 cfa3 fc 70 19        ldd PC
2019 2017 cfa6 83 00 01        subd #1
2020 2018 cfa9 fd 70 19        std PC
2021 2019 cfac c3 00 01        addd #1
2022 2020 cfaf bd ce 09        jsr read_mem
2023 2021 cfb2 bd ce 23        jsr key_data
2024 2022                      cc74:
2025 2023 cfb5 f6 70 27        ldb state
2026 2024 cfb8 1d              sex
2027 2025 cfb9 34 06          pshs d
2028 2026 cfb9 cc 00 04        ldd #4
2029 2027 cfbe 10 a3 e1        cmpd ,s++
2030 2028 cfc1 27 05          beq *+7
2031 2029 cfc3 cc 00 00        ldd #0
2032 2030 cfc6 20 03          bra *+5
2033 2031 cfc8 cc 00 01        ldd #1
2034 2032 cfc9 10 83 00 00     cmpd #0
2035 2033 cfcf 10 27 00 12     lbeq cc75
2036 2034 cfd3 fc 70 1d        ldd num
2037 2035 cfd6 fd 70 1f        std start
2038 2036 cfd9 cc 00 00        ldd #0
2039 2037 cfdc f7 70 0b        stb hit
2040 2038 cfdf cc 00 00        ldd #0
2041 2039 cfe2 f7 70 0c        stb positive
2042 2040                      cc75:
2043 2041 cfe5 39              rts
2044 2042
2045 2043                      * .global data_hex
2046 2044                      data_hex:
2047 2045 cfe6 fc 70 19        ldd PC
2048 2046 cfe9 fd 70 46        std dptr
2049 2047 cfec fc 70 46        ldd dptr
2050 2048 cfef 34 06          pshs d
2051 2049 cff1 e6 f1          ldb [,s++]
2052 2050 cff3 1d              sex
2053 2051 cff4 f7 70 0a        stb x
2054 2052 cff7 f6 70 0b        ldd hit
2055 2053 cffa 1d              sex
2056 2054 cffb 34 06          pshs d
2057 2055cffd cc 00 00        ldd #0
2058 2056 d000 10 a3 e1        cmpd ,s++
2059 2057 d003 27 05          beq *+7
2060 2058 d005 cc 00 00        ldd #0
2061 2059 d008 20 03          bra *+5
2062 2060 d00a cc 00 01        ldd #1
2063 2061 d00d 10 83 00 00     cmpd #0
2064 2062 d011 10 27 00 06     lbeq cc76
2065 2063 d015 cc 00 00        ldd #0
2066 2064 d018 f7 70 0a        stb x
2067 2065                      cc76:
2068 2066 d01b cc 00 01        ldd #1
2069 2067 d01e f7 70 0b        stb hit
2070 2068 d021 f6 70 0a        ldb x
2071 2069 d024 1d              sex
2072 2070 d025 34 06          pshs d
```

```

2073 2071 d027 cc 00 04      ldd #4
2074 2072 d02a 5a          decb
2075 2073 d02b 2d 06      blt *+8
2076 2074 d02d 68 61      asl 1,s
2077 2075 d02f 69 e4      rol ,s
2078 2076 d031 20 f7      bra *-7
2079 2077 d033 35 06      puls d
2080 2078 d035 f7 70 0a      stb x
2081 2079 d038 f6 70 0a      ldb x
2082 2080 d03b 1d          sex
2083 2081 d03c 34 06      pshs d
2084 2082 d03e f6 70 09      ldb key
2085 2083 d041 1d          sex
2086 2084 d042 aa e0      ora ,s+
2087 2085 d044 ea e0      orb ,s+
2088 2086 d046 f7 70 0a      stb x
2089 2087 d049 fc 70 46      ldd dptr
2090 2088 d04c 34 06      pshs d
2091 2089 d04e f6 70 0a      ldb x
2092 2090 d051 1d          sex
2093 2091 d052 e7 f1      stb [,s++]
2094 2092 d054 bd ce 09      jsr read_mem
2095 2093 d057 bd cb 9d      jsr dot_data
2096 2094 d05a 39          rts
2097
2098 2096
2099 2097
2100 2098 d05b fc 70 1b      * .global key_PC
2101 2099 d05e fd 70 19      key_PC:
2102 2100 d061 bd ce 23      ldd save_PC
2103 2101 d064 39          std PC
2104
2102
2103
2104
2105 2105 d065 f6 70 0b      jsr key_data
2106 2106 d068 1d          rts
2107
2103
2104
2107 2105 d065 f6 70 0b      * .global hex_addr
2108 2106 d068 1d          hex_addr:
2109 2107 d069 34 06      ldb hit
2110 2108 d06b cc 00 00      sex
2111 2109 d06e 10 a3 e1      pshs d
2112 2110 d071 27 05      ldd #0
2113 2111 d073 cc 00 00      cmpd ,s++
2114 2112 d076 20 03      beq *+7
2115 2113 d078 cc 00 01      ldd #0
2116 2114 d07b 10 83 00 00      bra *+5
2117 2115 d07f 10 27 00 06      ldd #1
2118 2116 d083 cc 00 00      cmpd #0
2119 2117 d086 fd 70 19      lbeq cc77
2120 2118
2119
2120
2121 2119 d089 cc 00 01      ldd #0
2122 2120 d08c f7 70 0b      stb hit
2123 2121 d08f cc 70 19      ldd #PC
2124 2122 d092 34 06      pshs d
2125 2123 d094 cc 00 04      ldd #4
2126 2124 d097 ae f4      ldx [,s]
2127 2125 d099 34 10      pshs x
2128 2126 d09b 5a          decb

```

```
2129 2127 d09c 2d 06          blt  *+8
2130 2128 d09e 68 61          asl  1,s
2131 2129 d0a0 69 e4          rol  ,s
2132 2130 d0a2 20 f7          bra  *-7
2133 2131 d0a4 35 06          puls d
2134 2132 d0a6 fd 70 19      std  PC
2135 2133 d0a9 cc 70 19      ldd  #PC
2136 2134 d0ac 34 06          pshs d
2137 2135 d0ae f6 70 09      ldb  key
2138 2136 d0b1 1d             sex
2139 2137 d0b2 ae f4          ldx  [,s]
2140 2138 d0b4 34 10          pshs x
2141 2139 d0b6 aa e0          ora  ,s+
2142 2140 d0b8 ea e0          orb  ,s+
2143 2141 d0ba fd 70 19      std  PC
2144 2142 d0bd bd ce 09      jsr  read_mem
2145 2143 d0c0 bd ca b4      jsr  dot_addr
2146 2144 d0c3 32 64          leas 4,s
2147 2145 d0c5 39             rts
2148 2146
2149 2147
2150 2148
2151 2149 d0c6 cc 70 38      * .global print_er
2152 2150 d0c9 34 06          print_er:
2153 2151 d0cb cc 00 05      ldd  #buffer
2154 2152 d0ce e3 e1          pshs d
2155 2153 d0d0 34 06          ldd  #5
2156 2154 d0d2 cc 00 8f      addd ,s++
2157 2155 d0d5 e7 f1          pshs d
2158 2156 d0d7 cc 70 38      ldd  #143
2159 2157 d0da 34 06          stb  [,s++]
2160 2158 d0dc cc 00 04      ldd  #buffer
2161 2159 d0df e3 e1          pshs d
2162 2160 d0e1 34 06          ldd  #4
2163 2161 d0e3 cc 00 03      addd ,s++
2164 2162 d0e6 e7 f1          pshs d
2165 2163 d0e8 cc 70 38      ldd  #3
2166 2164 d0eb 34 06          stb  [,s++]
2167 2165 d0ed cc 00 03      ldd  #buffer
2168 2166 d0f0 e3 e1          pshs d
2169 2167 d0f2 34 06          ldd  #3
2170 2168 d0f4 cc 00 03      addd ,s++
2171 2169 d0f7 e7 f1          pshs d
2172 2170 d0f9 cc 70 38      ldd  #0
2173 2171 d0fc 34 06          stb  [,s++]
2174 2172 d0fe cc 00 02      ldd  #buffer
2175 2173 d101 e3 e1          pshs d
2176 2174 d103 34 06          ldd  #2
2177 2175 d105 cc 00 00      addd ,s++
2178 2176 d108 e7 f1          pshs d
2179 2177 d10a cc 70 38      ldd  #1
2180 2178 d10d 34 06          stb  [,s++]
2181 2179 d10f cc 00 01      pshs d
2182 2180 d112 e3 e1          ldd  #0
2183 2181 d114 34 06          addd ,s++
2184 2182 d116 cc 00 00      pshs d
                                         ldd  #0
```

```
2185 2183 d119 e7 f1          stb  [,s++]
2186 2184 d11b cc 70 38       ldd  #buffer
2187 2185 d11e 34 06         pshs d
2188 2186 d120 cc 00 00       ldd  #0
2189 2187 d123 e3 e1         addd ,s++
2190 2188 d125 34 06         pshs d
2191 2189 d127 cc 00 00       ldd  #0
2192 2190 d12a e7 f1         stb  [,s++]
2193 2191 d12c cc 00 00       ldd  #0
2194 2192 d12f f7 70 27       stb  state
2195 2193 d132 39            rts
2196
2197 2195                      * .global key_go
2198 2196                      key_go:
2199 2197 d133 f6 70 27       ldb  state
2200 2198 d136 1d             sex
2201 2199 d137 34 06         pshs d
2202 2200 d139 cc 00 01       ldd  #1
2203 2201 d13c 10 a3 e1       cmpd ,s++
2204 2202 d13f 27 05         beq  *+7
2205 2203 d141 cc 00 00       ldd  #0
2206 2204 d144 20 03         bra  *+5
2207 2205 d146 cc 00 01       ldd  #1
2208 2206 d149 10 83 00 00    cmpd #0
2209 2207 d14d 10 26 00 24    lbne cc79
2210 2208 d151 f6 70 27       ldb  state
2211 2209 d154 1d             sex
2212 2210 d155 34 06         pshs d
2213 2211 d157 cc 00 02       ldd  #2
2214 2212 d15a 10 a3 e1       cmpd ,s++
2215 2213 d15d 27 05         beq  *+7
2216 2214 d15f cc 00 00       ldd  #0
2217 2215 d162 20 03         bra  *+5
2218 2216 d164 cc 00 01       ldd  #1
2219 2217 d167 10 83 00 00    cmpd #0
2220 2218 d16b 10 26 00 06    lbne cc79
2221 2219 d16f cc 00 00       ldd  #0
2222 2220 d172 7e d1 78       jmp  cc80
2223 2221                      cc79:
2224 2222 d175 cc 00 01       ldd  #1
2225 2223                      cc80:
2226 2224 d178 10 83 00 00    cmpd #0
2227 2225 d17c 10 27 00 24    lbeq cc78
2228 2226 d180 10 ff 70 2e    STS SAVE_SP
2229 2227 d184 fe 70 2a       LDU USER_U
2230 2228 d187 fc 70 19       LDD PC
2231 2229 d18a 34 06         PSHS D
2232 2230 d18c b6 70 34       LDA USER_P
2233 2231 d18f 1f 8a         TFR A,CC
2234 2232 d191 b6 70 35       LDA USER_DP
2235 2233 d194 1f 8b         TFR A,DP
2236 2234 d196 be 70 2c       LDX USER_X
2237 2235 d199 10 be 70 30    LDY USER_Y
2238 2236 d19d f6 70 33       LDB USER_B
2239 2237 d1a0 b6 70 32       LDA USER_A
2240 2238 d1a3 39            RTS
```

```
2241 2239
2242 2240 d1a4 f6 70 27
2243 2241 d1a7 1d
2244 2242 d1a8 34 06
2245 2243 d1aa cc 00 04
2246 2244 d1ad 10 a3 e1
2247 2245 d1b0 27 05
2248 2246 d1b2 cc 00 00
2249 2247 d1b5 20 03
2250 2248 d1b7 cc 00 01
2251 2249 d1ba 10 83 00 00
2252 2250 d1be 10 27 00 57
2253 2251 d1c2 fc 70 1d
2254 2252 d1c5 fd 70 23
2255 2253 d1c8 f6 70 0c
2256 2254 d1cb 1d
2257 2255 d1cc 34 06
2258 2256 d1ce cc 00 00
2259 2257 d1d1 10 a3 e1
2260 2258 d1d4 27 05
2261 2259 d1d6 cc 00 00
2262 2260 d1d9 20 03
2263 2261 d1db cc 00 01
2264 2262 d1de 10 83 00 00
2265 2263 d1e2 10 27 00 16
2266 2264 d1e6 fc 70 1f
2267 2265 d1e9 34 06
2268 2266 d1eb fc 70 23
2269 2267 d1ee 35 10
2270 2268 d1f0 34 06
2271 2269 d1f2 1f 10
2272 2270 d1f4 a3 e1
2273 2271 d1f6 fd 70 1f
2274 2272 d1f9 7e d2 09
2275 2273
2276 2274 d1fc fc 70 1f
2277 2275 d1ff 34 06
2278 2276 d201 fc 70 23
2279 2277 d204 e3 e1
2280 2278 d206 fd 70 1f
2281 2279
2282 2280 d209 fc 70 1f
2283 2281 d20c 34 06
2284 2282 d20e bd cc 8a
2285 2283 d211 32 62
2286 2284 d213 cc 00 00
2287 2285 d216 f7 70 0b
2288 2286
2289 2287 d219 f6 70 27
2290 2288 d21c 1d
2291 2289 d21d 34 06
2292 2290 d21f cc 00 07
2293 2291 d222 10 a3 e1
2294 2292 d225 27 05
2295 2293 d227 cc 00 00
2296 2294 d22a 20 03

cc78:
    ldb state
    sex
    pshs d
    ldd #4
    cmpd ,s++
    beq *+7
    ldd #0
    bra *+5
    ldd #1
    cmpd #0
    lbeq cc81
    ldd num
    std desti
    ldb positive
    sex
    pshs d
    ldd #0
    cmpd ,s++
    beq *+7
    ldd #0
    bra *+5
    ldd #1
    cmpd #0
    lbeq cc82
    ldd start
    pshs d
    ldd desti
    puls x
    pshs d
    tfr x,d
    subd ,s++
    std start
    jmp cc83
cc82:
    ldd start
    pshs d
    ldd desti
    addd ,s++
    std start
cc83:
    ldd start
    pshs d
    jsr hex4
    leas 2,s
    ldd #0
    stb hit
cc81:
    ldb state
    sex
    pshs d
    ldd #7
    cmpd ,s++
    beq *+7
    ldd #0
    bra *+5
```

```
2297 2295 d22c cc 00 01      ldd #1
2298 2296 d22f 10 83 00 00    cmpd #0
2299 2297 d233 10 27 00 8c    lbeq cc84
2300 2298 d237 fc 70 1d      ldd num
2301 2299 d23a fd 70 23      std desti
2302 2300 d23d fc 70 21      ldd end
2303 2301 d240 34 06      pshs d
2304 2302 d242 fc 70 1f      ldd start
2305 2303 d245 35 10      puls x
2306 2304 d247 34 06      pshs d
2307 2305 d249 1f 10      tfr x,d
2308 2306 d24b a3 e1      subd ,s++
2309 2307 d24d fd 70 15      std temp
2310 2308 d250 fc 70 1f      ldd start
2311 2309 d253 fd 70 46      std dptr
2312 2310 d256 fc 70 23      ldd desti
2313 2311 d259 fd 70 48      std dptr2
2314 2312 d25c cc 00 00      ldd #0
2315 2313 d25f fd 70 00      std i
2316 2314      cc87:
2317 2315 d262 fc 70 00      ldd i
2318 2316 d265 34 06      pshs d
2319 2317 d267 fc 70 15      ldd temp
2320 2318 d26a 10 a3 e1      cmpd ,s++
2321 2319 d26d 2e 05      bgt *+7
2322 2320 d26f cc 00 00      ldd #0
2323 2321 d272 20 03      bra *+5
2324 2322 d274 cc 00 01      ldd #1
2325 2323 d277 10 83 00 00    cmpd #0
2326 2324 d27b 10 27 00 32    lbeq cc86
2327 2325 d27f 7e d2 91      jmp cc88
2328 2326      cc85:
2329 2327 d282 fc 70 00      ldd i
2330 2328 d285 c3 00 01      addd #1
2331 2329 d288 fd 70 00      std i
2332 2330 d28b 83 00 01      subd #1
2333 2331 d28e 7e d2 62      jmp cc87
2334 2332      cc88:
2335 2333 d291 fc 70 48      ldd dptr2
2336 2334 d294 34 06      pshs d
2337 2335 d296 fc 70 00      ldd i
2338 2336 d299 e3 e1      addd ,s++
2339 2337 d29b 34 06      pshs d
2340 2338 d29d fc 70 46      ldd dptr
2341 2339 d2a0 34 06      pshs d
2342 2340 d2a2 fc 70 00      ldd i
2343 2341 d2a5 e3 e1      addd ,s++
2344 2342 d2a7 34 06      pshs d
2345 2343 d2a9 e6 f1      ldb [ ,s++ ]
2346 2344 d2ab 1d      sex
2347 2345 d2ac e7 f1      stb [ ,s++ ]
2348 2346 d2ae 7e d2 82      jmp cc85
2349 2347      cc86:
2350 2348 d2b1 fc 70 23      ldd desti
2351 2349 d2b4 fd 70 19      std PC
2352 2350 d2b7 bd ce 09      jsr read_mem
```

```
2353 2351 d2ba bd cb 9d          jsr dot_data
2354 2352 d2bd cc 00 02          ldd #2
2355 2353 d2c0 f7 70 27          stb state
2356 2354                               cc84:
2357 2355 d2c3 39          rts
2358 2356
2359 2357          * .global key_reg
2360 2358          key_reg:
2361 2359 d2c4 cc 70 38          ldd #buffer
2362 2360 d2c7 34 06          pshs d
2363 2361 d2c9 cc 00 07          ldd #7
2364 2362 d2cc e3 e1          addd ,s++
2365 2363 d2ce 34 06          pshs d
2366 2364 d2d0 cc 00 00          ldd #0
2367 2365 d2d3 e7 f1          stb [,s++]
2368 2366 d2d5 cc 70 38          ldd #buffer
2369 2367 d2d8 34 06          pshs d
2370 2368 d2da cc 00 06          ldd #6
2371 2369 d2dd e3 e1          addd ,s++
2372 2370 d2df 34 06          pshs d
2373 2371 d2e1 cc 00 00          ldd #0
2374 2372 d2e4 e7 f1          stb [,s++]
2375 2373 d2e6 cc 70 38          ldd #buffer
2376 2374 d2e9 34 06          pshs d
2377 2375 d2eb cc 00 05          ldd #5
2378 2376 d2ee e3 e1          addd ,s++
2379 2377 d2f0 34 06          pshs d
2380 2378 d2f2 cc 00 03          ldd #3
2381 2379 d2f5 e7 f1          stb [,s++]
2382 2380 d2f7 cc 70 38          ldd #buffer
2383 2381 d2fa 34 06          pshs d
2384 2382 d2fc cc 00 04          ldd #4
2385 2383 d2ff e3 e1          addd ,s++
2386 2384 d301 34 06          pshs d
2387 2385 d303 cc 00 8f          ldd #143
2388 2386 d306 e7 f1          stb [,s++]
2389 2387 d308 cc 70 38          ldd #buffer
2390 2388 d30b 34 06          pshs d
2391 2389 d30d cc 00 03          ldd #3
2392 2390 d310 e3 e1          addd ,s++
2393 2391 d312 34 06          pshs d
2394 2392 d314 cc 00 ad          ldd #173
2395 2393 d317 e7 f1          stb [,s++]
2396 2394 d319 cc 70 38          ldd #buffer
2397 2395 d31c 34 06          pshs d
2398 2396 d31e cc 00 02          ldd #2
2399 2397 d321 e3 e1          addd ,s++
2400 2398 d323 34 06          pshs d
2401 2399 d325 cc 00 00          ldd #0
2402 2400 d328 e7 f1          stb [,s++]
2403 2401 d32a cc 70 38          ldd #buffer
2404 2402 d32d 34 06          pshs d
2405 2403 d32f cc 00 01          ldd #1
2406 2404 d332 e3 e1          addd ,s++
2407 2405 d334 34 06          pshs d
2408 2406 d336 cc 00 00          ldd #0
```

```
2409 2407 d339 e7 f1          stb  [,s++]
2410 2408 d33b cc 70 38       ldd  #buffer
2411 2409 d33e 34 06         pshs d
2412 2410 d340 cc 00 00       ldd  #0
2413 2411 d343 e3 e1         addd ,s++
2414 2412 d345 34 06         pshs d
2415 2413 d347 cc 00 00       ldd  #0
2416 2414 d34a e7 f1         stb  [,s++]
2417 2415 d34c cc 00 03       ldd  #3
2418 2416 d34f f7 70 27       stb  state
2419 2417 d352 39           rts
2420
2418
2421 2419                   * .global acca
2422 2420                   acca:
2423 2421 d353 f6 70 32       ldb  USER_A
2424 2422 d356 1d           sex
2425 2423 d357 f7 70 04       stb  n
2426 2424 d35a cc 70 38       ldd  #buffer
2427 2425 d35d 34 06         pshs d
2428 2426 d35f cc 00 02       ldd  #2
2429 2427 d362 e3 e1         addd ,s++
2430 2428 d364 34 06         pshs d
2431 2429 d366 cc c1 0d       ldd  #convert
2432 2430 d369 34 06         pshs d
2433 2431 d36b f6 70 04       ldb  n
2434 2432 d36e 1d           sex
2435 2433 d36f 34 06         pshs d
2436 2434 d371 cc 00 0f       ldd  #15
2437 2435 d374 a4 e0         anda ,s+
2438 2436 d376 e4 e0         andb ,s+
2439 2437 d378 e3 e1         addd ,s++
2440 2438 d37a 34 06         pshs d
2441 2439 d37c e6 f1         ldb  [,s++]
2442 2440 d37e 1d           sex
2443 2441 d37f e7 f1         stb  [,s++]
2444 2442 d381 f6 70 04       ldb  n
2445 2443 d384 1d           sex
2446 2444 d385 34 06         pshs d
2447 2445 d387 cc 00 04       ldd  #4
2448 2446 d38a 5a           decb
2449 2447 d38b 2d 06         blt  *+8
2450 2448 d38d 64 e4         lsr  ,s
2451 2449 d38f 66 61         ror  1,s
2452 2450 d391 20 f7         bra  *-7
2453 2451 d393 35 06         puls d
2454 2452 d395 f7 70 04       stb  n
2455 2453 d398 cc 70 38       ldd  #buffer
2456 2454 d39b 34 06         pshs d
2457 2455 d39d cc 00 03       ldd  #3
2458 2456 d3a0 e3 e1         addd ,s++
2459 2457 d3a2 34 06         pshs d
2460 2458 d3a4 cc c1 0d       ldd  #convert
2461 2459 d3a7 34 06         pshs d
2462 2460 d3a9 f6 70 04       ldb  n
2463 2461 d3ac 1d           sex
2464 2462 d3ad 34 06         pshs d
```

```
2465 2463 d3af cc 00 0f      ldd #15
2466 2464 d3b2 a4 e0        anda ,s+
2467 2465 d3b4 e4 e0        andb ,s+
2468 2466 d3b6 e3 e1        addd ,s++
2469 2467 d3b8 34 06        pshs d
2470 2468 d3ba e6 f1        ldb [,s++]
2471 2469 d3bc 1d          sex
2472 2470 d3bd e7 f1        stb [,s++]
2473 2471 d3bf cc 70 38      ldd #buffer
2474 2472 d3c2 34 06        pshs d
2475 2473 d3c4 cc 00 04      ldd #4
2476 2474 d3c7 e3 e1        addd ,s++
2477 2475 d3c9 34 06        pshs d
2478 2476 d3cb cc 00 00      ldd #0
2479 2477 d3ce e7 f1        stb [,s++]
2480 2478 d3d0 cc 70 38      ldd #buffer
2481 2479 d3d3 34 06        pshs d
2482 2480 d3d5 cc 00 05      ldd #5
2483 2481 d3d8 e3 e1        addd ,s++
2484 2482 d3da 34 06        pshs d
2485 2483 d3dc cc 00 00      ldd #0
2486 2484 d3df e7 f1        stb [,s++]
2487 2485 d3e1 cc 70 38      ldd #buffer
2488 2486 d3e4 34 06        pshs d
2489 2487 d3e6 cc 00 01      ldd #1
2490 2488 d3e9 e3 e1        addd ,s++
2491 2489 d3eb 34 06        pshs d
2492 2490 d3ed cc 00 00      ldd #0
2493 2491 d3f0 e7 f1        stb [,s++]
2494 2492 d3f2 cc 70 38      ldd #buffer
2495 2493 d3f5 34 06        pshs d
2496 2494 d3f7 cc 00 00      ldd #0
2497 2495 d3fa e3 e1        addd ,s++
2498 2496 d3fc 34 06        pshs d
2499 2497 d3fe cc 00 3f      ldd #63
2500 2498 d401 e7 f1        stb [,s++]
2501 2499 d403 39          rts
2502 2500
2503 2501
2504 2502
2505 2503 d404 f6 70 33      * .global accb
2506 2504 d407 1d          accb:
2507 2505 d408 f7 70 04      ldb USER_B
2508 2506 d40b cc 70 38      sex
2509 2507 d40e 34 06        stb n
2510 2508 d410 cc 00 02      ldd #buffer
2511 2509 d413 e3 e1        pshs d
2512 2510 d415 34 06        addd ,s++
2513 2511 d417 cc c1 0d      pshs d
2514 2512 d41a 34 06        ldd #convert
2515 2513 d41c f6 70 04      pshs d
2516 2514 d41f 1d          ldb n
2517 2515 d420 34 06        sex
2518 2516 d422 cc 00 0f      pshs d
2519 2517 d425 a4 e0        ldd #15
2520 2518 d427 e4 e0        anda ,s+
                                andb ,s+
```

2521	2519	d429	e3	e1	addd ,s++
2522	2520	d42b	34	06	pshs d
2523	2521	d42d	e6	f1	ldb [,s++]
2524	2522	d42f	1d		sex
2525	2523	d430	e7	f1	stb [,s++]
2526	2524	d432	f6	70 04	ldb n
2527	2525	d435	1d		sex
2528	2526	d436	34	06	pshs d
2529	2527	d438	cc	00 04	ldd #4
2530	2528	d43b	5a		decb
2531	2529	d43c	2d	06	blt *+8
2532	2530	d43e	64	e4	lsr ,s
2533	2531	d440	66	61	ror 1,s
2534	2532	d442	20	f7	bra *-7
2535	2533	d444	35	06	puls d
2536	2534	d446	f7	70 04	stb n
2537	2535	d449	cc	70 38	ldd #buffer
2538	2536	d44c	34	06	pshs d
2539	2537	d44e	cc	00 03	ldd #3
2540	2538	d451	e3	e1	addd ,s++
2541	2539	d453	34	06	pshs d
2542	2540	d455	cc	c1 0d	ldd #convert
2543	2541	d458	34	06	pshs d
2544	2542	d45a	f6	70 04	ldb n
2545	2543	d45d	1d		sex
2546	2544	d45e	34	06	pshs d
2547	2545	d460	cc	00 0f	ldd #15
2548	2546	d463	a4	e0	anda ,s+
2549	2547	d465	e4	e0	andb ,s+
2550	2548	d467	e3	e1	addd ,s++
2551	2549	d469	34	06	pshs d
2552	2550	d46b	e6	f1	ldb [,s++]
2553	2551	d46d	1d		sex
2554	2552	d46e	e7	f1	stb [,s++]
2555	2553	d470	cc	70 38	ldd #buffer
2556	2554	d473	34	06	pshs d
2557	2555	d475	cc	00 04	ldd #4
2558	2556	d478	e3	e1	addd ,s++
2559	2557	d47a	34	06	pshs d
2560	2558	d47c	cc	00 00	ldd #0
2561	2559	d47f	e7	f1	stb [,s++]
2562	2560	d481	cc	70 38	ldd #buffer
2563	2561	d484	34	06	pshs d
2564	2562	d486	cc	00 05	ldd #5
2565	2563	d489	e3	e1	addd ,s++
2566	2564	d48b	34	06	pshs d
2567	2565	d48d	cc	00 00	ldd #0
2568	2566	d490	e7	f1	stb [,s++]
2569	2567	d492	cc	70 38	ldd #buffer
2570	2568	d495	34	06	pshs d
2571	2569	d497	cc	00 01	ldd #1
2572	2570	d49a	e3	e1	addd ,s++
2573	2571	d49c	34	06	pshs d
2574	2572	d49e	cc	00 00	ldd #0
2575	2573	d4a1	e7	f1	stb [,s++]
2576	2574	d4a3	cc	70 38	ldd #buffer

```
2577 2575 d4a6 34 06          pshs d
2578 2576 d4a8 cc 00 00          ldd #0
2579 2577 d4ab e3 e1          addd ,s++
2580 2578 d4ad 34 06          pshs d
2581 2579 d4af cc 00 a7          ldd #167
2582 2580 d4b2 e7 f1          stb [,s++]
2583 2581 d4b4 39          rts
2584 2582
2585 2583          * .global ab
2586 2584          ab:
2587 2585 d4b5 f6 70 33          ldb USER_B
2588 2586 d4b8 1d          sex
2589 2587 d4b9 f7 70 04          stb n
2590 2588 d4bc cc 70 38          ldd #buffer
2591 2589 d4bf 34 06          pshs d
2592 2590 d4c1 cc 00 02          ldd #2
2593 2591 d4c4 e3 e1          addd ,s++
2594 2592 d4c6 34 06          pshs d
2595 2593 d4c8 cc c1 0d          ldd #convert
2596 2594 d4cb 34 06          pshs d
2597 2595 d4cd f6 70 04          ldb n
2598 2596 d4d0 1d          sex
2599 2597 d4d1 34 06          pshs d
2600 2598 d4d3 cc 00 0f          ldd #15
2601 2599 d4d6 a4 e0          anda ,s+
2602 2600 d4d8 e4 e0          andb ,s+
2603 2601 d4da e3 e1          addd ,s++
2604 2602 d4dc 34 06          pshs d
2605 2603 d4de e6 f1          ldb [,s++]
2606 2604 d4e0 1d          sex
2607 2605 d4e1 e7 f1          stb [,s++]
2608 2606 d4e3 f6 70 04          ldb n
2609 2607 d4e6 1d          sex
2610 2608 d4e7 34 06          pshs d
2611 2609 d4e9 cc 00 04          ldd #4
2612 2610 d4ec 5a          decb
2613 2611 d4ed 2d 06          blt *+8
2614 2612 d4ef 64 e4          lsr ,s
2615 2613 d4f1 66 61          ror 1,s
2616 2614 d4f3 20 f7          bra *-7
2617 2615 d4f5 35 06          puls d
2618 2616 d4f7 f7 70 04          stb n
2619 2617 d4fa cc 70 38          ldd #buffer
2620 2618 d4fd 34 06          pshs d
2621 2619 d4ff cc 00 03          ldd #3
2622 2620 d502 e3 e1          addd ,s++
2623 2621 d504 34 06          pshs d
2624 2622 d506 cc c1 0d          ldd #convert
2625 2623 d509 34 06          pshs d
2626 2624 d50b f6 70 04          ldb n
2627 2625 d50e 1d          sex
2628 2626 d50f 34 06          pshs d
2629 2627 d511 cc 00 0f          ldd #15
2630 2628 d514 a4 e0          anda ,s+
2631 2629 d516 e4 e0          andb ,s+
2632 2630 d518 e3 e1          addd ,s++
```

2633	2631 d51a 34 06	pshs d
2634	2632 d51c e6 f1	ldb [,s++]
2635	2633 d51e 1d	sex
2636	2634 d51f e7 f1	stb [,s++]
2637	2635 d521 f6 70 32	ldb USER_A
2638	2636 d524 1d	sex
2639	2637 d525 f7 70 04	stb n
2640	2638 d528 cc 70 38	ldd #buffer
2641	2639 d52b 34 06	pshs d
2642	2640 d52d cc 00 04	ldd #4
2643	2641 d530 e3 e1	addd ,s++
2644	2642 d532 34 06	pshs d
2645	2643 d534 cc c1 0d	ldd #convert
2646	2644 d537 34 06	pshs d
2647	2645 d539 f6 70 04	ldb n
2648	2646 d53c 1d	sex
2649	2647 d53d 34 06	pshs d
2650	2648 d53f cc 00 0f	ldd #15
2651	2649 d542 a4 e0	anda ,s+
2652	2650 d544 e4 e0	andb ,s+
2653	2651 d546 e3 e1	addd ,s++
2654	2652 d548 34 06	pshs d
2655	2653 d54a e6 f1	ldb [,s++]
2656	2654 d54c 1d	sex
2657	2655 d54d e7 f1	stb [,s++]
2658	2656 d54f f6 70 04	ldb n
2659	2657 d552 1d	sex
2660	2658 d553 34 06	pshs d
2661	2659 d555 cc 00 04	ldd #4
2662	2660 d558 5a	decb
2663	2661 d559 2d 06	blt *+8
2664	2662 d55b 64 e4	lsr ,s
2665	2663 d55d 66 61	ror 1,s
2666	2664 d55f 20 f7	bra *-7
2667	2665 d561 35 06	puls d
2668	2666 d563 f7 70 04	stb n
2669	2667 d566 cc 70 38	ldd #buffer
2670	2668 d569 34 06	pshs d
2671	2669 d56b cc 00 05	ldd #5
2672	2670 d56e e3 e1	addd ,s++
2673	2671 d570 34 06	pshs d
2674	2672 d572 cc c1 0d	ldd #convert
2675	2673 d575 34 06	pshs d
2676	2674 d577 f6 70 04	ldb n
2677	2675 d57a 1d	sex
2678	2676 d57b 34 06	pshs d
2679	2677 d57d cc 00 0f	ldd #15
2680	2678 d580 a4 e0	anda ,s+
2681	2679 d582 e4 e0	andb ,s+
2682	2680 d584 e3 e1	addd ,s++
2683	2681 d586 34 06	pshs d
2684	2682 d588 e6 f1	ldb [,s++]
2685	2683 d58a 1d	sex
2686	2684 d58b e7 f1	stb [,s++]
2687	2685 d58d cc 70 38	ldd #buffer
2688	2686 d590 34 06	pshs d

```
2689 2687 d592 cc 00 01      ldd #1
2690 2688 d595 e3 e1      addd ,s++
2691 2689 d597 34 06      pshs d
2692 2690 d599 cc 00 3f      ldd #63
2693 2691 d59c e7 f1      stb [,s++]
2694 2692 d59e cc 70 38      ldd #buffer
2695 2693 d5a1 34 06      pshs d
2696 2694 d5a3 cc 00 00      ldd #0
2697 2695 d5a6 e3 e1      addd ,s++
2698 2696 d5a8 34 06      pshs d
2699 2697 d5aa cc 00 a7      ldd #167
2700 2698 d5ad e7 f1      stb [,s++]
2701 2699 d5af 39      rts
2702 2700
2703 2701      * .global reg_x
2704 2702      reg_x:
2705 2703 d5b0 fc 70 2c      ldd USER_X
2706 2704 d5b3 fd 70 17      std temp16
2707 2705 d5b6 fc 70 17      ldd temp16
2708 2706 d5b9 34 06      pshs d
2709 2707 d5bb bd cc 8a      jsr hex4
2710 2708 d5be 32 62      leas 2,s
2711 2709 d5c0 cc 70 38      ldd #buffer
2712 2710 d5c3 34 06      pshs d
2713 2711 d5c5 cc 00 01      ldd #1
2714 2712 d5c8 e3 e1      addd ,s++
2715 2713 d5ca 34 06      pshs d
2716 2714 d5cc cc 00 00      ldd #0
2717 2715 d5cf e7 f1      stb [,s++]
2718 2716 d5d1 cc 70 38      ldd #buffer
2719 2717 d5d4 34 06      pshs d
2720 2718 d5d6 cc 00 00      ldd #0
2721 2719 d5d9 e3 e1      addd ,s++
2722 2720 d5db 34 06      pshs d
2723 2721 d5dd cc 00 13      ldd #19
2724 2722 d5e0 e7 f1      stb [,s++]
2725 2723 d5e2 39      rts
2726 2724
2727 2725      * .global reg_y
2728 2726      reg_y:
2729 2727 d5e3 fc 70 30      ldd USER_Y
2730 2728 d5e6 fd 70 17      std temp16
2731 2729 d5e9 fc 70 17      ldd temp16
2732 2730 d5ec 34 06      pshs d
2733 2731 d5ee bd cc 8a      jsr hex4
2734 2732 d5f1 32 62      leas 2,s
2735 2733 d5f3 cc 70 38      ldd #buffer
2736 2734 d5f6 34 06      pshs d
2737 2735 d5f8 cc 00 01      ldd #1
2738 2736 d5fb e3 e1      addd ,s++
2739 2737 d5fd 34 06      pshs d
2740 2738 d5ff cc 00 00      ldd #0
2741 2739 d602 e7 f1      stb [,s++]
2742 2740 d604 cc 70 38      ldd #buffer
2743 2741 d607 34 06      pshs d
2744 2742 d609 cc 00 00      ldd #0
```

```

2745 2743 d60c e3 e1      addd ,s++
2746 2744 d60e 34 06      pshs d
2747 2745 d610 cc 00 b6      ldd #182
2748 2746 d613 e7 f1      stb [,s++]
2749 2747 d615 39      rts

2750 2748
2751 2749
2752 2750
2753 2751 d616 fc 70 2a      * .global reg_u
2754 2752 d619 fd 70 17      reg_u:
2755 2753 d61c fc 70 17      ldd USER_U
2756 2754 d61f 34 06      std temp16
2757 2755 d621 bd cc 8a      ldd temp16
2758 2756 d624 32 62      pshs d
2759 2757 d626 cc 70 38      jsr hex4
2760 2758 d629 34 06      leas 2,s
2761 2759 d62b cc 00 01      ldd #buffer
2762 2760 d62e e3 e1      pshs d
2763 2761 d630 34 06      ldd #1
2764 2762 d632 cc 00 00      addd ,s++
2765 2763 d635 e7 f1      pshs d
2766 2764 d637 cc 70 38      ldd #0
2767 2765 d63a 34 06      stb [,s++]
2768 2766 d63c cc 00 00      ldd #buffer
2769 2767 d63f e3 e1      pshs d
2770 2768 d641 34 06      ldd #0
2771 2769 d643 cc 00 b5      addd ,s++
2772 2770 d646 e7 f1      pshs d
2773 2771 d648 39      ldd #181
2774 2772      stb [,s++]
2775 2773      rts

2776 2774
2777 2775 d649 fc 70 2e      * .global reg_s
2778 2776 d64c fd 70 17      reg_s:
2779 2777 d64f fc 70 17      ldd SAVE_SP
2780 2778 d652 34 06      std temp16
2781 2779 d654 bd cc 8a      ldd temp16
2782 2780 d657 32 62      pshs d
2783 2781 d659 cc 70 38      jsr hex4
2784 2782 d65c 34 06      leas 2,s
2785 2783 d65e cc 00 01      ldd #buffer
2786 2784 d661 e3 e1      pshs d
2787 2785 d663 34 06      ldd #1
2788 2786 d665 cc 00 00      addd ,s++
2789 2787 d668 e7 f1      pshs d
2790 2788 d66a cc 70 38      ldd #0
2791 2789 d66d 34 06      stb [,s++]
2792 2790 d66f cc 00 00      ldd #buffer
2793 2791 d672 e3 e1      pshs d
2794 2792 d674 34 06      ldd #1
2795 2793 d676 cc 00 ae      addd ,s++
2796 2794 d679 e7 f1      pshs d
2797 2795 d67b 39      ldd #174
2798 2796      stb [,s++]
2799 2797      rts

2800 2798

```

2801	2799	d67c	f6	70	35	ldb	USER_DP
2802	2800	d67f	1d			sex	
2803	2801	d680	f7	70	04	stb	n
2804	2802	d683	cc	70	38	ldd	#buffer
2805	2803	d686	34	06		pshs	d
2806	2804	d688	cc	00	02	ldd	#2
2807	2805	d68b	e3	e1		addd	,s++
2808	2806	d68d	34	06		pshs	d
2809	2807	d68f	cc	c1	0d	ldd	#convert
2810	2808	d692	34	06		pshs	d
2811	2809	d694	f6	70	04	ldb	n
2812	2810	d697	1d			sex	
2813	2811	d698	34	06		pshs	d
2814	2812	d69a	cc	00	0f	ldd	#15
2815	2813	d69d	a4	e0		anda	,s+
2816	2814	d69f	e4	e0		andb	,s+
2817	2815	d6a1	e3	e1		addd	,s++
2818	2816	d6a3	34	06		pshs	d
2819	2817	d6a5	e6	f1		ldb	[,s++]
2820	2818	d6a7	1d			sex	
2821	2819	d6a8	e7	f1		stb	[,s++]
2822	2820	d6aa	f6	70	04	ldb	n
2823	2821	d6ad	1d			sex	
2824	2822	d6ae	34	06		pshs	d
2825	2823	d6b0	cc	00	04	ldd	#4
2826	2824	d6b3	5a			decb	
2827	2825	d6b4	2d	06		blt	*+8
2828	2826	d6b6	64	e4		lsr	,s
2829	2827	d6b8	66	61		ror	1,s
2830	2828	d6ba	20	f7		bra	*-7
2831	2829	d6bc	35	06		puls	d
2832	2830	d6be	f7	70	04	stb	n
2833	2831	d6c1	cc	70	38	ldd	#buffer
2834	2832	d6c4	34	06		pshs	d
2835	2833	d6c6	cc	00	03	ldd	#3
2836	2834	d6c9	e3	e1		addd	,s++
2837	2835	d6cb	34	06		pshs	d
2838	2836	d6cd	cc	c1	0d	ldd	#convert
2839	2837	d6d0	34	06		pshs	d
2840	2838	d6d2	f6	70	04	ldb	n
2841	2839	d6d5	1d			sex	
2842	2840	d6d6	34	06		pshs	d
2843	2841	d6d8	cc	00	0f	ldd	#15
2844	2842	d6db	a4	e0		anda	,s+
2845	2843	d6dd	e4	e0		andb	,s+
2846	2844	d6df	e3	e1		addd	,s++
2847	2845	d6e1	34	06		pshs	d
2848	2846	d6e3	e6	f1		ldb	[,s++]
2849	2847	d6e5	1d			sex	
2850	2848	d6e6	e7	f1		stb	[,s++]
2851	2849	d6e8	cc	70	38	ldd	#buffer
2852	2850	d6eb	34	06		pshs	d
2853	2851	d6ed	cc	00	04	ldd	#4
2854	2852	d6f0	e3	e1		addd	,s++
2855	2853	d6f2	34	06		pshs	d
2856	2854	d6f4	cc	00	00	ldd	#0

```
2857 2855 d6f7 e7 f1          stb  [,s++]
2858 2856 d6f9 cc 70 38      ldd  #buffer
2859 2857 d6fc 34 06          pshs d
2860 2858 d6fe cc 00 05      ldd  #5
2861 2859 d701 e3 e1          addd ,s++
2862 2860 d703 34 06          pshs d
2863 2861 d705 cc 00 00      ldd  #0
2864 2862 d708 e7 f1          stb  [,s++]
2865 2863 d70a cc 70 38      ldd  #buffer
2866 2864 d70d 34 06          pshs d
2867 2865 d70f cc 00 01      ldd  #1
2868 2866 d712 e3 e1          addd ,s++
2869 2867 d714 34 06          pshs d
2870 2868 d716 cc 00 b3      ldd  #179
2871 2869 d719 e7 f1          stb  [,s++]
2872 2870 d71b cc 70 38      ldd  #buffer
2873 2871 d71e 34 06          pshs d
2874 2872 d720 cc 00 00      ldd  #0
2875 2873 d723 e3 e1          addd ,s++
2876 2874 d725 34 06          pshs d
2877 2875 d727 cc 00 1f      ldd  #31
2878 2876 d72a e7 f1          stb  [,s++]
2879 2877 d72c 39             rts
2880
2881
2882
2883 2881 d72d f6 70 34      * .global low_cc
2884 2882 d730 1d             low_cc:
2885 2883 d731 f7 70 04      ldb  USER_P
2886 2884 d734 f6 70 04      sex
2887 2885 d737 1d             stb  n
2888 2886 d738 34 06          ldb  n
2889 2887 d73a cc 00 01      sex
2890 2888 d73d a4 e0          pshs d
2891 2889 d73f e4 e0          ldd  #1
2892 2890 d741 10 83 00 00    anda ,s+
2893 2891 d745 10 27 00 14    andb ,s+
2894 2892 d749 cc 70 38      cmpd #0
2895 2893 d74c 34 06          lbeq cc89
2896 2894 d74e cc 00 02      ldd  #buffer
2897 2895 d751 e3 e1          pshs d
2898 2896 d753 34 06          ldd  #2
2899 2897 d755 cc 00 30      addd ,s++
2900 2898 d758 e7 f1          pshs d
2901 2899 d75a 7e d7 6e      ldd  #48
2902 2900                      stb  [,s++]
2903 2901 d75d cc 70 38      jmp  cc90
2904 2902 d760 34 06          cc89:
2905 2903 d762 cc 00 02      ldd  #buffer
2906 2904 d765 e3 e1          pshs d
2907 2905 d767 34 06          ldd  #2
2908 2906 d769 cc 00 bd      addd ,s++
2909 2907 d76c e7 f1          pshs d
2910 2908                      ldd  #189
2911 2909 d76e f6 70 04      stb  [,s++]
2912 2910 d771 1d             cc90:
                                ldb  n
                                sex
```

```
2913 2911 d772 34 06          pshs d
2914 2912 d774 cc 00 02          ldd #2
2915 2913 d777 a4 e0          anda ,s+
2916 2914 d779 e4 e0          andb ,s+
2917 2915 d77b 10 83 00 00          cmpd #0
2918 2916 d77f 10 27 00 14          lbeq cc91
2919 2917 d783 cc 70 38          ldd #buffer
2920 2918 d786 34 06          pshs d
2921 2919 d788 cc 00 03          ldd #3
2922 2920 d78b e3 e1          addd ,s++
2923 2921 d78d 34 06          pshs d
2924 2922 d78f cc 00 30          ldd #48
2925 2923 d792 e7 f1          stb [,s++]
2926 2924 d794 7e d7 a8          jmp cc92
2927 2925
2928 2926 d797 cc 70 38          cc91:
2929 2927 d79a 34 06          ldd #buffer
2930 2928 d79c cc 00 03          pshs d
2931 2929 d79f e3 e1          ldd #3
2932 2930 d7a1 34 06          addd ,s++
2933 2931 d7a3 cc 00 bd          pshs d
2934 2932 d7a6 e7 f1          ldd #189
2935 2933                      stb [,s++]
2936 2934 d7a8 f6 70 04          cc92:
2937 2935 d7ab 1d          ldb n
2938 2936 d7ac 34 06          sex
2939 2937 d7ae cc 00 04          pshs d
2940 2938 d7b1 a4 e0          ldd #4
2941 2939 d7b3 e4 e0          anda ,s+
2942 2940 d7b5 10 83 00 00          andb ,s+
2943 2941 d7b9 10 27 00 14          cmpd #0
2944 2942 d7bd cc 70 38          lbeq cc93
2945 2943 d7c0 34 06          ldd #buffer
2946 2944 d7c2 cc 00 04          pshs d
2947 2945 d7c5 e3 e1          ldd #4
2948 2946 d7c7 34 06          addd ,s++
2949 2947 d7c9 cc 00 30          pshs d
2950 2948 d7cc e7 f1          ldd #48
2951 2949 d7ce 7e d7 e2          stb [,s++]
2952 2950                      jmp cc94
2953 2951 d7d1 cc 70 38          cc93:
2954 2952 d7d4 34 06          ldd #buffer
2955 2953 d7d6 cc 00 04          pshs d
2956 2954 d7d9 e3 e1          ldd #4
2957 2955 d7db 34 06          addd ,s++
2958 2956 d7dd cc 00 bd          pshs d
2959 2957 d7e0 e7 f1          ldd #189
2960 2958                      stb [,s++]
2961 2959 d7e2 f6 70 04          cc94:
2962 2960 d7e5 1d          ldb n
2963 2961 d7e6 34 06          sex
2964 2962 d7e8 cc 00 08          pshs d
2965 2963 d7eb a4 e0          ldd #8
2966 2964 d7ed e4 e0          anda ,s+
2967 2965 d7ef 10 83 00 00          andb ,s+
2968 2966 d7f3 10 27 00 14          cmpd #0
                                         lbeq cc95
```

```
2969 2967 d7f7 cc 70 38      ldd #buffer
2970 2968 d7fa 34 06        pshs d
2971 2969 d7fc cc 00 05      ldd #5
2972 2970 d7ff e3 e1        addd ,s++
2973 2971 d801 34 06        pshs d
2974 2972 d803 cc 00 30      ldd #48
2975 2973 d806 e7 f1        stb [,s++]
2976 2974 d808 7e d8 1c      jmp cc96
2977 2975                      cc95:
2978 2976 d80b cc 70 38      ldd #buffer
2979 2977 d80e 34 06        pshs d
2980 2978 d810 cc 00 05      ldd #5
2981 2979 d813 e3 e1        addd ,s++
2982 2980 d815 34 06        pshs d
2983 2981 d817 cc 00 bd      ldd #189
2984 2982 d81a e7 f1        stb [,s++]
2985 2983                      cc96:
2986 2984 d81c cc 70 38      ldd #buffer
2987 2985 d81f 34 06        pshs d
2988 2986 d821 cc 00 01      ldd #1
2989 2987 d824 e3 e1        addd ,s++
2990 2988 d826 34 06        pshs d
2991 2989 d828 cc 00 8d      ldd #141
2992 2990 d82b e7 f1        stb [,s++]
2993 2991 d82d cc 70 38      ldd #buffer
2994 2992 d830 34 06        pshs d
2995 2993 d832 cc 00 00      ldd #0
2996 2994 d835 e3 e1        addd ,s++
2997 2995 d837 34 06        pshs d
2998 2996 d839 cc 00 85      ldd #133
2999 2997 d83c e7 f1        stb [,s++]
3000 2998 d83e 39          rts
3001 2999
3002 3000
3003 3001
3004 3002 d83f f6 70 34      * .global hi_cc
3005 3003 d842 1d          hi_cc:
3006 3004 d843 f7 70 04      ldb USER_P
3007 3005 d846 f6 70 04      sex
3008 3006 d849 1d          stb n
3009 3007 d84a 34 06        ldb n
3010 3008 d84c cc 00 10      sex
3011 3009 d84f a4 e0        pshs d
3012 3010 d851 e4 e0        ldd #16
3013 3011 d853 10 83 00 00  anda ,s+
3014 3012 d857 10 27 00 14  andb ,s+
3015 3013 d85b cc 70 38      cmpd #0
3016 3014 d85e 34 06        lbeq cc97
3017 3015 d860 cc 00 02      ldd #buffer
3018 3016 d863 e3 e1        pshs d
3019 3017 d865 34 06        addd ,s++
3020 3018 d867 cc 00 30      pshs d
3021 3019 d86a e7 f1        ldd #48
3022 3020 d86c 7e d8 80      stb [,s++]
3023 3021                      cc97:
3024 3022 d86f cc 70 38      jmp cc98
                                ldd #buffer
```

```
3025 3023 d872 34 06          pshs d
3026 3024 d874 cc 00 02          ldd #2
3027 3025 d877 e3 e1          addd ,s++
3028 3026 d879 34 06          pshs d
3029 3027 d87b cc 00 bd          ldd #189
3030 3028 d87e e7 f1          stb [,s++]
3031 3029
3032 3030 d880 f6 70 04          cc98:
3033 3031 d883 1d          ldb n
3034 3032 d884 34 06          sex
3035 3033 d886 cc 00 20          pshs d
3036 3034 d889 a4 e0          ldd #32
3037 3035 d88b e4 e0          anda ,s+
3038 3036 d88d 10 83 00 00          andb ,s+
3039 3037 d891 10 27 00 14          cmpd #0
3040 3038 d895 cc 70 38          lbeq cc99
3041 3039 d898 34 06          ldd #buffer
3042 3040 d89a cc 00 03          pshs d
3043 3041 d89d e3 e1          ldd #3
3044 3042 d89f 34 06          addd ,s++
3045 3043 d8a1 cc 00 30          pshs d
3046 3044 d8a4 e7 f1          ldd #48
3047 3045 d8a6 7e d8 ba          stb [,s++]
3048 3046          jmp cc100
3049 3047 d8a9 cc 70 38          cc99:
3050 3048 d8ac 34 06          ldd #buffer
3051 3049 d8ae cc 00 03          pshs d
3052 3050 d8b1 e3 e1          ldd #3
3053 3051 d8b3 34 06          addd ,s++
3054 3052 d8b5 cc 00 bd          pshs d
3055 3053 d8b8 e7 f1          ldd #189
3056 3054          stb [,s++]
3057 3055 d8ba f6 70 04          cc100:
3058 3056 d8bd 1d          ldb n
3059 3057 d8be 34 06          sex
3060 3058 d8c0 cc 00 40          pshs d
3061 3059 d8c3 a4 e0          ldd #64
3062 3060 d8c5 e4 e0          anda ,s+
3063 3061 d8c7 10 83 00 00          andb ,s+
3064 3062 d8cb 10 27 00 14          cmpd #0
3065 3063 d8cf cc 70 38          lbeq cc101
3066 3064 d8d2 34 06          ldd #buffer
3067 3065 d8d4 cc 00 04          pshs d
3068 3066 d8d7 e3 e1          ldd #4
3069 3067 d8d9 34 06          addd ,s++
3070 3068 d8db cc 00 30          pshs d
3071 3069 d8de e7 f1          ldd #48
3072 3070 d8e0 7e d8 f4          stb [,s++]
3073 3071          jmp cc102
3074 3072 d8e3 cc 70 38          cc101:
3075 3073 d8e6 34 06          ldd #buffer
3076 3074 d8e8 cc 00 04          pshs d
3077 3075 d8eb e3 e1          ldd #4
3078 3076 d8ed 34 06          addd ,s++
3079 3077 d8ef cc 00 bd          pshs d
3080 3078 d8f2 e7 f1          ldd #189
3081
3082
3083
3084
3085
3086
3087
3088
3089
3090
3091
3092
3093
3094
3095
3096
3097
3098
3099
3100
3101
3102
3103
3104
3105
3106
3107
3108
3109
3110
3111
3112
3113
3114
3115
3116
3117
3118
3119
3120
3121
3122
3123
3124
3125
3126
3127
3128
3129
3130
3131
3132
3133
3134
3135
3136
3137
3138
3139
3140
3141
3142
3143
3144
3145
3146
3147
3148
3149
3150
3151
3152
3153
3154
3155
3156
3157
3158
3159
3160
3161
3162
3163
3164
3165
3166
3167
3168
3169
3170
3171
3172
3173
3174
3175
3176
3177
3178
3179
3180
3181
3182
3183
3184
3185
3186
3187
3188
3189
3190
3191
3192
3193
3194
3195
3196
3197
3198
3199
3200
3201
3202
3203
3204
3205
3206
3207
3208
3209
3210
3211
3212
3213
3214
3215
3216
3217
3218
3219
3220
3221
3222
3223
3224
3225
3226
3227
3228
3229
3230
3231
3232
3233
3234
3235
3236
3237
3238
3239
3240
3241
3242
3243
3244
3245
3246
3247
3248
3249
3250
3251
3252
3253
3254
3255
3256
3257
3258
3259
3260
3261
3262
3263
3264
3265
3266
3267
3268
3269
3270
3271
3272
3273
3274
3275
3276
3277
3278
3279
3280
3281
3282
3283
3284
3285
3286
3287
3288
3289
3290
3291
3292
3293
3294
3295
3296
3297
3298
3299
3300
3301
3302
3303
3304
3305
3306
3307
3308
3309
3310
3311
3312
3313
3314
3315
3316
3317
3318
3319
3320
3321
3322
3323
3324
3325
3326
3327
3328
3329
3330
3331
3332
3333
3334
3335
3336
3337
3338
3339
3340
3341
3342
3343
3344
3345
3346
3347
3348
3349
3350
3351
3352
3353
3354
3355
3356
3357
3358
3359
3360
3361
3362
3363
3364
3365
3366
3367
3368
3369
3370
3371
3372
3373
3374
3375
3376
3377
3378
3379
3380
3381
3382
3383
3384
3385
3386
3387
3388
3389
3390
3391
3392
3393
3394
3395
3396
3397
3398
3399
3400
3401
3402
3403
3404
3405
3406
3407
3408
3409
3410
3411
3412
3413
3414
3415
3416
3417
3418
3419
3420
3421
3422
3423
3424
3425
3426
3427
3428
3429
3430
3431
3432
3433
3434
3435
3436
3437
3438
3439
3440
3441
3442
3443
3444
3445
3446
3447
3448
3449
3450
3451
3452
3453
3454
3455
3456
3457
3458
3459
3460
3461
3462
3463
3464
3465
3466
3467
3468
3469
3470
3471
3472
3473
3474
3475
3476
3477
3478
3479
3480
3481
3482
3483
3484
3485
3486
3487
3488
3489
3490
3491
3492
3493
3494
3495
3496
3497
3498
3499
3500
3501
3502
3503
3504
3505
3506
3507
3508
3509
3510
3511
3512
3513
3514
3515
3516
3517
3518
3519
3520
3521
3522
3523
3524
3525
3526
3527
3528
3529
3530
3531
3532
3533
3534
3535
3536
3537
3538
3539
3540
3541
3542
3543
3544
3545
3546
3547
3548
3549
3550
3551
3552
3553
3554
3555
3556
3557
3558
3559
3560
3561
3562
3563
3564
3565
3566
3567
3568
3569
3570
3571
3572
3573
3574
3575
3576
3577
3578
3579
3580
3581
3582
3583
3584
3585
3586
3587
3588
3589
3590
3591
3592
3593
3594
3595
3596
3597
3598
3599
3600
3601
3602
3603
3604
3605
3606
3607
3608
3609
3610
3611
3612
3613
3614
3615
3616
3617
3618
3619
3620
3621
3622
3623
3624
3625
3626
3627
3628
3629
3630
3631
3632
3633
3634
3635
3636
3637
3638
3639
3640
3641
3642
3643
3644
3645
3646
3647
3648
3649
3650
3651
3652
3653
3654
3655
3656
3657
3658
3659
3660
3661
3662
3663
3664
3665
3666
3667
3668
3669
3670
3671
3672
3673
3674
3675
3676
3677
3678
3679
3680
3681
3682
3683
3684
3685
3686
3687
3688
3689
3690
3691
3692
3693
3694
3695
3696
3697
3698
3699
3700
3701
3702
3703
3704
3705
3706
3707
3708
3709
3710
3711
3712
3713
3714
3715
3716
3717
3718
3719
3720
3721
3722
3723
3724
3725
3726
3727
3728
3729
3730
3731
3732
3733
3734
3735
3736
3737
3738
3739
3740
3741
3742
3743
3744
3745
3746
3747
3748
3749
3750
3751
3752
3753
3754
3755
3756
3757
3758
3759
3760
3761
3762
3763
3764
3765
3766
3767
3768
3769
3770
3771
3772
3773
3774
3775
3776
3777
3778
3779
3780
3781
3782
3783
3784
3785
3786
3787
3788
3789
3790
3791
3792
3793
3794
3795
3796
3797
3798
3799
3800
3801
3802
3803
3804
3805
3806
3807
3808
3809
38010
38011
38012
38013
38014
38015
38016
38017
38018
38019
38020
38021
38022
38023
38024
38025
38026
38027
38028
38029
38030
38031
38032
38033
38034
38035
38036
38037
38038
38039
38040
38041
38042
38043
38044
38045
38046
38047
38048
38049
38050
38051
38052
38053
38054
38055
38056
38057
38058
38059
38060
38061
38062
38063
38064
38065
38066
38067
38068
38069
38070
38071
38072
38073
38074
38075
38076
38077
38078
38079
38080
38081
38082
38083
38084
38085
38086
38087
38088
38089
380810
380811
380812
380813
380814
380815
380816
380817
380818
380819
380820
380821
380822
380823
380824
380825
380826
380827
380828
380829
380830
380831
380832
380833
380834
380835
380836
380837
380838
380839
380840
380841
380842
380843
380844
380845
380846
380847
380848
380849
380850
380851
380852
380853
380854
380855
380856
380857
380858
380859
380860
380861
380862
380863
380864
380865
380866
380867
380868
380869
380870
380871
380872
380873
380874
380875
380876
380877
380878
380879
380880
380881
380882
380883
380884
380885
380886
380887
380888
380889
3808810
3808811
3808812
3808813
3808814
3808815
3808816
3808817
3808818
3808819
3808820
3808821
3808822
3808823
3808824
3808825
3808826
3808827
3808828
3808829
3808830
3808831
3808832
3808833
3808834
3808835
3808836
3808837
3808838
3808839
38088310
38088311
38088312
38088313
38088314
38088315
38088316
38088317
38088318
38088319
38088320
38088321
38088322
38088323
38088324
38088325
38088326
38088327
38088328
38088329
38088330
38088331
38088332
38088333
38088334
38088335
38088336
38088337
38088338
38088339
38088340
38088341
38088342
38088343
38088344
38088345
38088346
38088347
38088348
38088349
38088350
38088351
38088352
38088353
38088354
38088355
38088356
38088357
38088358
38088359
38088360
38088361
38088362
38088363
38088364
38088365
38088366
38088367
38088368
38088369
38088370
38088371
38088372
38088373
38088374
38088375
38088376
38088377
38088378
38088379
38088380
38088381
38088382
38088383
38088384
38088385
38088386
38088387
38088388
38088389
380883810
380883811
380883812
380883813
380883814
380883815
380883816
380883817
380883818
380883819
380883820
380883821
380883822
380883823
380883824
380883825
380883826
380883827
380883828
380883829
380883830
380883831
380883832
380883833
380883834
380883835
380883836
380883837
380883838
380883839
380883840
380883841
380883842
380883843
380883844
380883845
380883846
380883847
380883848
380883849
380883850
380883851
380883852
380883853
380883854
380883855
380883856
380883857
380883858
380883859
380883860
380883861
380883862
380883863
380883864
380883865
380883866
380883867
380883868
380883869
380883870
380883871
380883872
380883873
380883874
380883875
380883876
380883877
380883878
380883879
380883880
380883881
380883882
380883883
380883884
380883885
380883886
380883887
380883888
380883889
3808838810
3808838811
3808838812
3808838813
3808838814
3808838815
3808838816
3808838817
3808838818
3808838819
3808838820
3808838821
3808838822
3808838823
3808838824
3808838825
3808838826
3808838827
3808838828
3808838829
3808838830
3808838831
3808838832
3808838833
3808838834
3808838835
3808838836
3808838837
3808838838
3808838839
3808838840
3808838841
3808838842
3808838843
3808838844
3808838845
3808838846
3808838847
3808838848
3808838849
3808838850
3808838851
3808838852
3808838853
3808838854
3808838855
3808838856
3808838857
3808838858
3808838859
3808838860
3808838861
3808838862
3808838863
3808838864
3808838865
3808838866
3808838867
3808838868
3808838869
3808838870
3808838871
3808838872
3808838873
3808838874
3808838875
3808838876
3808838877
3808838878
3808838879
3808838880
3808838881
3808838882
3808838883
3808838884
3808838885
3808838886
3808838887
3808838888
3808838889
38088388810
38088388811
38088388812
38088388813
38088388814
38088388815
38088388816
38088388817
38088388818
38088388819
38088388820
38088388821
38088388822
38088388823
38088388824
38088388825
38088388826
38088388827
38088388828
38088388829
38088388830
38088388831
38088388832
38088388833
38088388834
38088388835
38088388836
38088388837
38088388838
38088388839
38088388840
38088388841
38088388842
38088388843
38088388844
38088388845
38088388846
38088388847
38088388848
38088388849
38088388850
38088388851
38088388852
38088388853
38088388854
38088388855
38088388856
38088388857
38088388858
38088388859
38088388860
38088388861
38088388862
38088388863
38088388864
38088388865
38088388866
38088388867
38088388868
38088388869
38088388870
38088388871
38088388872
38088388873
38088388874
38088388875
38088388876
38088388877
38088388878
38088388879
38088388880
38088388881
38088388882
38088388883
38088388884
38088388885
38088388886
38088388887
38088388888
38088388889
380883888810
380883888811
380883888812
380883888813
380883888814
380883888815
380883888816
380883888817
380883888818
380883888819
380883888820
380883888821
380883888822
380883888823
380883888824
380883888825
380883888826
380883888827
380883888828
380883888829
380883888830
380883888831
380883888832
380883888833
380883888834
380883888835
380883888836
380883888837
380883888838
380883888839
380883888840
380883888841
380883888842
380883888843
380883888844
380883888845
380883888846
380883888847
380883888848
380883888849
380883888850
380883888851
380883888852
380883888853
380883888854
380883888855
380883888856
380883888857
380883888858
380883888859
380883888860
380883888861
380883888862
380883888863
380883888864
380883888865
380883888866
380883888867
380883888868
380883888869
380883888870
380883888871
380883888872
380883888873
380883888874
380883888875
380883888876
380883888877
380883888878
380883888879
380883888880
380883888881
380883888882
380883888883
380883888884
380883888885
380883888886
380883888887
380883888888
380883888889
3808838888810
3808838888811
3808838888812
3808838888813
3808838888814
3808838888815
3808838888816
3808838888817
3808838888818
3808838888819
3808838888820
3808838888821
3808838888822
3808838888823
3808838888824
3808838888825
3808838888826
3808838888827
3808838888828
3808838888829
3808838888830
3808838888831
3808838888832
3808838888833
3808838888834
3808838888835
3808838888836
3808838888837
3808838888838
3808838888839
3808838888840
3808838888841
3808838888842
3808838888843
3808838888844
3808838888845
3808838888846
3808838888847
3808838888848
3808838888849
3808838888850
3808838888851
3808838888852
380883888
```

```
3081 3079
3082 3080 d8f4 f6 70 04
3083 3081 d8f7 1d
3084 3082 d8f8 34 06
3085 3083 d8fa cc 00 80
3086 3084 d8fd a4 e0
3087 3085 d8ff e4 e0
3088 3086 d901 10 83 00 00
3089 3087 d905 10 27 00 14
3090 3088 d909 cc 70 38
3091 3089 d90c 34 06
3092 3090 d90e cc 00 05
3093 3091 d911 e3 e1
3094 3092 d913 34 06
3095 3093 d915 cc 00 30
3096 3094 d918 e7 f1
3097 3095 d91a 7e d9 2e
3098 3096
3099 3097 d91d cc 70 38
3100 3098 d920 34 06
3101 3099 d922 cc 00 05
3102 3100 d925 e3 e1
3103 3101 d927 34 06
3104 3102 d929 cc 00 bd
3105 3103 d92c e7 f1
3106 3104
3107 3105 d92e cc 70 38
3108 3106 d931 34 06
3109 3107 d933 cc 00 01
3110 3108 d936 e3 e1
3111 3109 d938 34 06
3112 3110 d93a cc 00 8d
3113 3111 d93d e7 f1
3114 3112 d93f cc 70 38
3115 3113 d942 34 06
3116 3114 d944 cc 00 00
3117 3115 d947 e3 e1
3118 3116 d949 34 06
3119 3117 d94b cc 00 37
3120 3118 d94e e7 f1
3121 3119 d950 39
3122 3120
3123 3121
3124 3122
3125 3123 d951 f6 70 09
3126 3124 d954 1d
3127 3125 d955 7e d9 97
3128 3126
3129 3127 d958 bd d3 53
3130 3128 d95b 7e d9 c4
3131 3129
3132 3130 d95e bd d4 04
3133 3131 d961 7e d9 c4
3134 3132
3135 3133 d964 bd d4 b5
3136 3134 d967 7e d9 c4

cc102:
    ldb  n
    sex
    pshs d
    ldd  #128
    anda ,s+
    andb ,s+
    cmpd #0
    lbeq cc103
    ldd  #buffer
    pshs d
    ldd  #5
    addd ,s++
    pshs d
    ldd  #48
    stb  [,s++]
    jmp  cc104

cc103:
    ldd  #buffer
    pshs d
    ldd  #5
    addd ,s++
    pshs d
    ldd  #189
    stb  [,s++]

cc104:
    ldd  #buffer
    pshs d
    ldd  #1
    addd ,s++
    pshs d
    ldd  #141
    stb  [,s++]
    ldd  #buffer
    pshs d
    ldd  #0
    addd ,s++
    pshs d
    ldd  #55
    stb  [,s++]
    rts

* .global reg_disp
reg_disp:
    ldb  key
    sex
    jmp  cc107

cc108:
    jsr  acca
    jmp  cc106

cc109:
    jsr  accb
    jmp  cc106

cc110:
    jsr  ab
    jmp  cc106
```

```
3137 3135          cc111:  
3138 3136 d96a bd d5 b0      jsr reg_x  
3139 3137 d96d 7e d9 c4      jmp cc106  
3140 3138          cc112:  
3141 3139 d970 bd d5 e3      jsr reg_y  
3142 3140 d973 7e d9 c4      jmp cc106  
3143 3141          cc113:  
3144 3142 d976 bd d6 16      jsr reg_u  
3145 3143 d979 7e d9 c4      jmp cc106  
3146 3144          cc114:  
3147 3145 d97c bd d6 49      jsr reg_s  
3148 3146 d97f 7e d9 c4      jmp cc106  
3149 3147          cc115:  
3150 3148 d982 bd d6 7c      jsr reg_dp  
3151 3149 d985 7e d9 c4      jmp cc106  
3152 3150          cc116:  
3153 3151 d988 bd d7 2d      jsr low_cc  
3154 3152 d98b 7e d9 c4      jmp cc106  
3155 3153          cc117:  
3156 3154 d98e bd d8 3f      jsr hi_cc  
3157 3155 d991 7e d9 c4      jmp cc106  
3158 3156 d994 7e d9 c4      jmp cc106  
3159 3157          cc107:  
3160 3158 d997 bd c0 f9      jsr ccswitch  
3161 3159 d99a d9 58 00 00    FDB cc108,0  
3162 3160 d99e d9 5e 00 01    FDB cc109,1  
3163 3161 d9a2 d9 64 00 02    FDB cc110,2  
3164 3162 d9a6 d9 6a 00 06    FDB cc111,6  
3165 3163 d9aa d9 70 00 07    FDB cc112,7  
3166 3164 d9ae d9 76 00 09    FDB cc113,9  
3167 3165 d9b2 d9 7c 00 0a    FDB cc114,10  
3168 3166 d9b6 d9 82 00 08    FDB cc115,8  
3169 3167 d9ba d9 88 00 05    FDB cc116,5  
3170 3168 d9be d9 8e 00 04    FDB cc117,4  
3171 3169 d9c2 00 00          FDB 0  
3172 3170          cc106:  
3173 3171 d9c4 39            rts  
3174 3172  
3175 3173          * .global insert  
3176 3174          insert:  
3177 3175 d9c5 f6 70 27      ldb state  
3178 3176 d9c8 1d            sex  
3179 3177 d9c9 34 06          pshs d  
3180 3178 d9cb cc 00 01      ldd #1  
3181 3179 d9ce 10 a3 e1      cmpd ,s++  
3182 3180 d9d1 27 05          beq *+7  
3183 3181 d9d3 cc 00 00      ldd #0  
3184 3182 d9d6 20 03          bra *+5  
3185 3183 d9d8 cc 00 01      ldd #1  
3186 3184 d9db 10 83 00 00    cmpd #0  
3187 3185 d9df 10 26 00 24    lbne cc119  
3188 3186 d9e3 f6 70 27      ldb state  
3189 3187 d9e6 1d            sex  
3190 3188 d9e7 34 06          pshs d  
3191 3189 d9e9 cc 00 02      ldd #2  
3192 3190 d9ec 10 a3 e1      cmpd ,s++
```

```
3193 3191 d9ef 27 05          beq    *+7
3194 3192 d9f1 cc 00 00      ldd    #0
3195 3193 d9f4 20 03          bra    *+5
3196 3194 d9f6 cc 00 01      ldd    #1
3197 3195 d9f9 10 83 00 00    cmpd   #0
3198 3196 d9fd 10 26 00 06    lbne   cc119
3199 3197 da01 cc 00 00      ldd    #0
3200 3198 da04 7e da 0a      jmp    cc120
3201 3199
3202 3200 da07 cc 00 01      cc119:
3203 3201
3204 3202 da0a 10 83 00 00    ldd    #1
3205 3203 da0e 10 27 00 8e    cc120:
3206 3204 da12 fc 70 19      cmpd   #0
3207 3205 da15 fd 70 46      lbeq   cc118
3208 3206 da18 cc 02 00      ldd    PC
3209 3207 da1b fd 70 02      std    dptr
3210 3208
3211 3209 da1e fc 70 02      ldd    #512
3212 3210 da21 34 06          pshs   d
3213 3211 da23 cc 00 00      ldd    #0
3214 3212 da26 10 a3 e1      cmpd   ,s++
3215 3213 da29 2d 05          blt    *+7
3216 3214 da2b cc 00 00      ldd    #0
3217 3215 da2e 20 03          bra    *+5
3218 3216 da30 cc 00 01      ldd    #1
3219 3217 da33 10 83 00 00    cmpd   #0
3220 3218 da37 10 27 00 3f    lbeq   cc122
3221 3219 da3b 7e da 4d      jmp    cc124
3222 3220
3223 3221 da3e fc 70 02      cc121:
3224 3222 da41 83 00 01      ldd    j
3225 3223 da44 fd 70 02      subd   #1
3226 3224 da47 c3 00 01      std    j
3227 3225 da4a 7e da 1e      addd   #1
3228 3226
3229 3227 da4d fc 70 46      jmp    cc123
3230 3228 da50 34 06          cc124:
3231 3229 da52 fc 70 02      ldd    dptra
3232 3230 da55 e3 e1          pshs   d
3233 3231 da57 34 06          ldd    j
3234 3232 da59 fc 70 46      addd   ,s++
3235 3233 da5c 34 06          pshs   d
3236 3234 da5e fc 70 02      ldd    dptra
3237 3235 da61 e3 e1          pshs   d
3238 3236 da63 34 06          ldd    ,s++
3239 3237 da65 cc 00 01      puls   x
3240 3238 da68 35 10          pshs   d
3241 3239 da6a 34 06          tfr    x,d
3242 3240 da6c 1f 10          subd   ,s++
3243 3241 da6e a3 e1          pshs   d
3244 3242 da70 34 06          ldb    [ ,s++ ]
3245 3243 da72 e6 f1          sex
3246 3244 da74 1d            stb    [ ,s++ ]
3247 3245 da75 e7 f1          jmp    cc121
3248 3246 da77 7e da 3e
```

```
3249 3247 cc122:  
3250 3248 da7a fc 70 46 ldd dptr  
3251 3249 da7d 34 06 pshs d  
3252 3250 da7f cc 00 01 ldd #1  
3253 3251 da82 e3 e1 addd ,s++  
3254 3252 da84 34 06 pshs d  
3255 3253 da86 cc 00 00 ldd #0  
3256 3254 da89 e7 f1 stb [,s++]  
3257 3255 da8b fc 70 19 ldd PC  
3258 3256 da8e c3 00 01 addd #1  
3259 3257 da91 fd 70 19 std PC  
3260 3258 da94 83 00 01 subd #1  
3261 3259 da97 bd ce 09 jsr read_mem  
3262 3260 da9a cc 00 02 ldd #2  
3263 3261 da9d f7 70 27 stb state  
3264 3262 cc118:  
3265 3263 daa0 39 rts  
3266 3264  
3267 3265 * .global cut_byte  
3268 3266 cut_byte:  
3269 3267 daa1 f6 70 27 ldb state  
3270 3268 daa4 1d sex  
3271 3269 daa5 34 06 pshs d  
3272 3270 daa7 cc 00 01 ldd #1  
3273 3271 daaa 10 a3 e1 cmpd ,s++  
3274 3272 daad 27 05 beq *+7  
3275 3273 daaf cc 00 00 ldd #0  
3276 3274 dab2 20 03 bra *+5  
3277 3275 dab4 cc 00 01 ldd #1  
3278 3276 dab7 10 83 00 00 cmpd #0  
3279 3277 dabb 10 26 00 24 lbne cc126  
3280 3278 dabf f6 70 27 ldb state  
3281 3279 dac2 1d sex  
3282 3280 dac3 34 06 pshs d  
3283 3281 dac5 cc 00 02 ldd #2  
3284 3282 dac8 10 a3 e1 cmpd ,s++  
3285 3283 dacb 27 05 beq *+7  
3286 3284 dacd cc 00 00 ldd #0  
3287 3285 dad0 20 03 bra *+5  
3288 3286 dad2 cc 00 01 ldd #1  
3289 3287 dad5 10 83 00 00 cmpd #0  
3290 3288 dad9 10 26 00 06 lbne cc126  
3291 3289 dadd cc 00 00 ldd #0  
3292 3290 dae0 7e da e6 jmp cc127  
3293 3291 cc126:  
3294 3292 dae3 cc 00 01 ldd #1  
3295 3293 cc127:  
3296 3294 dae6 10 83 00 00 cmpd #0  
3297 3295 daea 10 27 00 6b lbeq cc125  
3298 3296 daee fc 70 19 ldd PC  
3299 3297 daf1 fd 70 46 std dptr  
3300 3298 daf4 cc 00 00 ldd #0  
3301 3299 daf7 fd 70 02 std j  
3302 3300 cc130:  
3303 3301 dafa fc 70 02 ldd j  
3304 3302 dafd 34 06 pshs d
```

```
3305 3303 daff cc 02 00      ldd #512
3306 3304 db02 10 a3 e1      cmpd ,s++
3307 3305 db05 2e 05      bgt *+7
3308 3306 db07 cc 00 00      ldd #0
3309 3307 db0a 20 03      bra *+5
3310 3308 db0c cc 00 01      ldd #1
3311 3309 db0f 10 83 00 00      cmpd #0
3312 3310 db13 10 27 00 39      lbeq cc129
3313 3311 db17 7e db 29      jmp cc131
3314 3312                      cc128:
3315 3313 db1a fc 70 02      ldd j
3316 3314 db1d c3 00 01      addd #1
3317 3315 db20 fd 70 02      std j
3318 3316 db23 83 00 01      subd #1
3319 3317 db26 7e da fa      jmp cc130
3320 3318                      cc131:
3321 3319 db29 fc 70 46      ldd dptr
3322 3320 db2c 34 06      pshs d
3323 3321 db2e fc 70 02      ldd j
3324 3322 db31 e3 e1      addd ,s++
3325 3323 db33 34 06      pshs d
3326 3324 db35 fc 70 46      ldd dptr
3327 3325 db38 34 06      pshs d
3328 3326 db3a fc 70 02      ldd j
3329 3327 db3d e3 e1      addd ,s++
3330 3328 db3f 34 06      pshs d
3331 3329 db41 cc 00 01      ldd #1
3332 3330 db44 e3 e1      addd ,s++
3333 3331 db46 34 06      pshs d
3334 3332 db48 e6 f1      ldb [ ,s++]
3335 3333 db4a 1d      sex
3336 3334 db4b e7 f1      stb [ ,s++]
3337 3335 db4d 7e db 1a      jmp cc128
3338 3336                      cc129:
3339 3337 db50 bd ce 09      jsr read_mem
3340 3338 db53 cc 00 02      ldd #2
3341 3339 db56 f7 70 27      stb state
3342 3340                      cc125:
3343 3341 db59 39      rts
3344 3342
3345 3343                      * .global key_test
3346 3344      key_test:
3347 3345 db5a 1c ef      andcc #$ef
3348 3346 db5c cc 00 00      ldd #0
3349 3347 db5f fd 70 25      std t
3350 3348 db62 cc 70 38      ldd #buffer
3351 3349 db65 34 06      pshs d
3352 3350 db67 cc 00 00      ldd #0
3353 3351 db6a e3 e1      addd ,s++
3354 3352 db6c 34 06      pshs d
3355 3353 db6e cc 00 00      ldd #0
3356 3354 db71 e7 f1      stb [ ,s++]
3357 3355 db73 cc 70 38      ldd #buffer
3358 3356 db76 34 06      pshs d
3359 3357 db78 cc 00 01      ldd #1
3360 3358 db7b e3 e1      addd ,s++
```

```
3361 3359 db7d 34 06          pshs d
3362 3360 db7f cc 00 00      ldd #0
3363 3361 db82 e7 f1        stb [,s++]
3364 3362                      cc132:
3365 3363 db84 cc 00 01      ldd #1
3366 3364 db87 10 83 00 00    cmpd #0
3367 3365 db8b 10 27 00 4d    lbeq cc133
3368 3366                      cc134:
3369 3367 db8f f6 70 0e      ldb tick
3370 3368 db92 1d            sex
3371 3369 db93 34 06          pshs d
3372 3370 db95 cc 00 0a      ldd #10
3373 3371 db98 10 a3 e1      cmpd ,s++
3374 3372 db9b 2e 05        bgt *+7
3375 3373 db9d cc 00 00      ldd #0
3376 3374 dba0 20 03        bra *+5
3377 3375 dba2 cc 00 01      ldd #1
3378 3376 dba5 10 83 00 00    cmpd #0
3379 3377 dba9 10 27 00 06    lbeq cc135
3380 3378 dbad bd c8 84      jsr scan
3381 3379 dbb0 7e db 8f      jmp cc134
3382 3380                      cc135:
3383 3381 dbb3 cc 00 00      ldd #0
3384 3382 dbb6 f7 70 0e      stb tick
3385 3383 dbb9 fc 70 25      ldd t
3386 3384 dbbc 34 06        pshs d
3387 3385 dbbe bd cc 8a      jsr hex4
3388 3386 dbc1 32 62        leas 2,s
3389 3387 dbc3 fc 70 3e      ldd gpio1
3390 3388 dbc6 34 06        pshs d
3391 3389 dbc8 fc 70 25      ldd t
3392 3390 dbcb e7 f1        stb [,s++]
3393 3391 dbcd fc 70 25      ldd t
3394 3392 dbd0 c3 00 01      addd #1
3395 3393 dbd3 fd 70 25      std t
3396 3394 dbd6 83 00 01      subd #1
3397 3395 dbd9 7e db 84      jmp cc132
3398 3396                      cc133:
3399 3397 dbdc 39            rts
3400 3398
3401 3399                      * .global clear_bu
3402 3400                      clear_bu:
3403 3401 dbdd cc 00 00      ldd #0
3404 3402dbe0 fd 70 00      std i
3405 3403                      cc138:
3406 3404dbe3 fc 70 00      ldd i
3407 3405dbe6 34 06        pshs d
3408 3406dbe8 cc 00 06      ldd #6
3409 3407dbeb 10 a3 e1      cmpd ,s++
3410 3408dbee 2e 05        bgt *+7
3411 3409dbf0 cc 00 00      ldd #0
3412 3410dbf3 20 03        bra *+5
3413 3411dbf5 cc 00 01      ldd #1
3414 3412dbf8 10 83 00 00    cmpd #0
3415 3413dbfc 10 27 00 26    lbeq cc137
3416 3414dc00 7e dc 12      jmp cc139
```

```
3417 3415          cc136:  
3418 3416 dc03 fc 70 00      ldd i  
3419 3417 dc06 c3 00 01      addd #1  
3420 3418 dc09 fd 70 00      std i  
3421 3419 dc0c 83 00 01      subd #1  
3422 3420 dc0f 7e db e3      jmp cc138  
3423 3421          cc139:  
3424 3422 dc12 cc 70 38      ldd #buffer  
3425 3423 dc15 34 06        pshs d  
3426 3424 dc17 fc 70 00      ldd i  
3427 3425 dc1a e3 e1        addd ,s++  
3428 3426 dc1c 34 06        pshs d  
3429 3427 dc1e cc 00 00      ldd #0  
3430 3428 dc21 e7 f1        stb [,s++]  
3431 3429 dc23 7e dc 03      jmp cc136  
3432 3430          cc137:  
3433 3431 dc26 39          rts  
3434 3432  
3435 3433          * .global key_cal  
3436 3434          key_cal:  
3437 3435 dc27 cc 00 04      ldd #4  
3438 3436 dc2a f7 70 27      stb state  
3439 3437 dc2d bd db dd      jsr clear_bu  
3440 3438 dc30 cc 70 38      ldd #buffer  
3441 3439 dc33 34 06        pshs d  
3442 3440 dc35 cc 00 02      ldd #2  
3443 3441 dc38 e3 e1        addd ,s++  
3444 3442 dc3a 34 06        pshs d  
3445 3443 dc3c cc 00 bd      ldd #189  
3446 3444 dc3f e7 f1        stb [,s++]  
3447 3445 dc41 cc 00 00      ldd #0  
3448 3446 dc44 fd 70 1f      std start  
3449 3447 dc47 cc 00 00      ldd #0  
3450 3448 dc4a fd 70 23      std desti  
3451 3449 dc4d cc 00 00      ldd #0  
3452 3450 dc50 f7 70 0b      stb hit  
3453 3451 dc53 39          rts  
3454 3452  
3455 3453          * .global enter_nu  
3456 3454          enter_nu:  
3457 3455 dc54 f6 70 0b      ldb hit  
3458 3456 dc57 1d          sex  
3459 3457 dc58 34 06        pshs d  
3460 3458 dc5a cc 00 00      ldd #0  
3461 3459 dc5d 10 a3 e1      cmpd ,s++  
3462 3460 dc60 27 05        beq *+7  
3463 3461 dc62 cc 00 00      ldd #0  
3464 3462 dc65 20 03        bra *+5  
3465 3463 dc67 cc 00 01      ldd #1  
3466 3464 dc6a 10 83 00 00    cmpd #0  
3467 3465 dc6e 10 27 00 06    lbeq cc140  
3468 3466 dc72 cc 00 00      ldd #0  
3469 3467 dc75 fd 70 1d      std num  
3470 3468          cc140:  
3471 3469 dc78 cc 00 01      ldd #1  
3472 3470 dc7b f7 70 0b      stb hit
```

```
3473 3471 dc7e cc 70 1d      ldd #num
3474 3472 dc81 34 06      pshs d
3475 3473 dc83 cc 00 04      ldd #4
3476 3474 dc86 ae f4      ldx [ ,s]
3477 3475 dc88 34 10      pshs x
3478 3476 dc8a 5a      decb
3479 3477 dc8b 2d 06      blt *+8
3480 3478 dc8d 68 61      asl 1,s
3481 3479 dc8f 69 e4      rol ,s
3482 3480 dc91 20 f7      bra *-7
3483 3481 dc93 35 06      puls d
3484 3482 dc95 fd 70 1d      std num
3485 3483 dc98 cc 70 1d      ldd #num
3486 3484 dc9b 34 06      pshs d
3487 3485 dc9d f6 70 09      ldb key
3488 3486 dca0 1d      sex
3489 3487 dca1 ae f4      ldx [ ,s]
3490 3488 dca3 34 10      pshs x
3491 3489 dca5 aa e0      ora ,s+
3492 3490 dca7 ea e0      orb ,s+
3493 3491 dca9 fd 70 1d      std num
3494 3492 dcac fc 70 1d      ldd num
3495 3493 dcac 34 06      pshs d
3496 3494 dcbb bd cc 8a      jsr hex4
3497 3495 dcbb 32 62      leas 2,s
3498 3496 dcbb 32 64      leas 4,s
3499 3497 dcbb 39      rts
3500 3498
3501 3499      * .global key_copy
3502 3500      key_copy:
3503 3501 dcbb 00 05      ldd #5
3504 3502 dcbb f7 70 27      stb state
3505 3503 dcbb cc 00 00      ldd #0
3506 3504 dcbb f7 70 0b      stb hit
3507 3505 dcbb bd db dd      jsr clear_bu
3508 3506 dcbb cc 70 38      ldd #buffer
3509 3507 dcbb 34 06      pshs d
3510 3508 dcbb cc 00 02      ldd #2
3511 3509 dcbb e3 e1      addd ,s++
3512 3510 dcbb 34 06      pshs d
3513 3511 dcbb 00 bd      ldd #189
3514 3512 dcbb e7 f1      stb [ ,s++ ]
3515 3513 dcbb cc 70 38      ldd #buffer
3516 3514 dcbb 34 06      pshs d
3517 3515 dcbb cc 00 00      ldd #0
3518 3516 dcbb e3 e1      addd ,s++
3519 3517 dcbb 34 06      pshs d
3520 3518 dcbb cc 00 ae      ldd #174
3521 3519 dcbb e7 f1      stb [ ,s++ ]
3522 3520 dcbb cc 70 38      ldd #buffer
3523 3521 dcbb 34 06      pshs d
3524 3522 dcbb cc 00 01      ldd #1
3525 3523 dcbb e3 e1      addd ,s++
3526 3524 dcbb 34 06      pshs d
3527 3525 dcbb cc 00 00      ldd #0
3528 3526 dcbb e7 f1      stb [ ,s++ ]
```

```
3529 3527 dcfb 39          rts
3530 3528
3531 3529          * .global key_exe
3532 3530          key_exe:
3533 3531 dcfc f6 70 0d    ldb flag
3534 3532 dcff 1d          sex
3535 3533 dd00 34 06      pshs d
3536 3534 dd02 cc 00 00    ldd #0
3537 3535 dd05 10 a3 e1    cmpd ,s++
3538 3536 dd08 27 05      beq *+7
3539 3537 dd0a cc 00 00    ldd #0
3540 3538 dd0d 20 03      bra *+5
3541 3539 dd0f cc 00 01    ldd #1
3542 3540 dd12 10 83 00 00  cmpd #0
3543 3541 dd16 10 27 00 03  lbeq cc141
3544 3542 dd1a bd de 82    jsr beep
3545 3543
3546 3544 dd1d f6 70 09    cc141:
3547 3545 dd20 1d          ldb key
3548 3546 dd21 34 06      sex
3549 3547 dd23 cc 00 0f      pshs d
3550 3548 dd26 10 a3 e1    ldd #15
3551 3549 dd29 2d 05      cmpd ,s++
3552 3550 dd2b cc 00 00    blt *+7
3553 3551 dd2e 20 03      ldd #0
3554 3552 dd30 cc 00 01    bra *+5
3555 3553 dd33 10 83 00 00  ldd #1
3556 3554 dd37 10 27 00 b5  cmpd #0
3557 3555 dd3b f6 70 09    lbeq cc142
3558 3556 dd3e 1d          ldb key
3559 3557 dd3f 7e dd ac    sex
3560 3558          cc145:
3561 3559 dd42 bd ce 10    jsr key_addr
3562 3560 dd45 7e dd ed    jmp cc144
3563 3561          cc146:
3564 3562 dd48 bd ce 23    jsr key_data
3565 3563 dd4b 7e dd ed    jmp cc144
3566 3564          cc147:
3567 3565 dd4e bd ce 36    jsr key_plus
3568 3566 dd51 7e dd ed    jmp cc144
3569 3567          cc148:
3570 3568 dd54 bd cf 69    jsr key_minu
3571 3569 dd57 7e dd ed    jmp cc144
3572 3570          cc150:
3573 3571 dd5a bd d0 5b    jsr key_PC
3574 3572 dd5d 7e dd ed    jmp cc144
3575 3573          cc151:
3576 3574 dd60 bd d1 33    jsr key_go
3577 3575 dd63 7e dd ed    jmp cc144
3578 3576          cc152:
3579 3577 dd66 bd d2 c4    jsr key_reg
3580 3578 dd69 7e dd ed    jmp cc144
3581 3579          cc153:
3582 3580 dd6c bd d9 c5    jsr insert
3583 3581 dd6f 7e dd ed    jmp cc144
3584 3582          cc154:
```

```
3585 3583 dd72 bd da a1          jsr cut_byte
3586 3584 dd75 7e dd ed          jmp cc144
3587 3585
3588 3586 dd78 f6 70 0d          cc155:
3589 3587 dd7b 1d               ldb flag
3590 3588 dd7c 34 06               sex
3591 3589 dd7e cc 00 01               pshs d
3592 3590 dd81 a8 e0               ldd #1
3593 3591 dd83 e8 e0               eora ,s+
3594 3592 dd85 f7 70 0d               eorb ,s+
3595 3593 dd88 7e dd ed               stb flag
3596 3594
3597 3595 dd8b bd db 5a          jmp cc144
3598 3596 dd8e 7e dd ed          cc156:
3599 3597
3600 3598 dd91 bd dc 27          jsr key_test
3601 3599 dd94 7e dd ed          jmp cc144
3602 3600
3603 3601 dd97 bd dc b9          cc157:
3604 3602 dd9a 7e dd ed          jsr key_cal
3605 3603
3606 3604 dd9d bd e2 21          jmp cc144
3607 3605 dda0 7e dd ed          cc158:
3608 3606
3609 3607 dda3 bd e6 d0          jsr key_copy
3610 3608 dda6 7e dd ed          jmp cc144
3611 3609 dda9 7e dd ed          cc159:
3612 3610
3613 3611 ddac bd c0 f9          jsr key_dump
3614 3612 ddaf dd 42 00 13          jmp cc144
3615 3613 ddb3 dd 48 00 12          cc160:
3616 3614 ddb7 dd 4e 00 17          jsr key_load
3617 3615 ddbb dd 54 00 16          jmp cc144
3618 3616 ddbf dd 5a 00 10          cc145:
3619 3617 ddc3 dd 60 00 1b          jsr ccswitch
3620 3618 ddc7 dd 66 00 11          FDB cc146,19
3621 3619 ddcb dd 6c 00 18          FDB cc147,18
3622 3620 ddcf dd 72 00 19          FDB cc148,23
3623 3621 ddd3 dd 78 00 15          FDB cc149,22
3624 3622 ddd7 dd 8b 00 1a          FDB cc150,16
3625 3623 dddb dd 91 00 1d          FDB cc151,27
3626 3624 dddf dd 97 00 1c          FDB cc152,17
3627 3625 dde3 dd 9d 00 1e          FDB cc153,24
3628 3626 dde7 dd a3 00 1f          FDB cc154,25
3629 3627 ddeb 00 00          FDB cc155,21
3630 3628
3631 3629 dded 7e de 45          FDB cc156,26
3632 3630
3633 3631 ddf0 f6 70 27          FDB cc157,29
3634 3632 ddf3 1d               FDB cc158,28
3635 3633 ddf4 7e de 24          FDB cc159,30
3636 3634
3637 3635 ddf7 bd d0 65          FDB cc160,31
3638 3636 ddfa 7e de 45          FDB 0
3639 3637
3640 3638 ddfd bd cf e6          cc144:

```

```

3641 3639 de00 7e de 45
3642 3640
3643 3641 de03 bd d9 51
3644 3642 de06 7e de 45
3645 3643
3646 3644 de09 bd dc 54
3647 3645 de0c 7e de 45
3648 3646
3649 3647 de0f bd dc 54
3650 3648 de12 7e de 45
3651 3649
3652 3650 de15 bd dc 54
3653 3651 de18 7e de 45
3654 3652
3655 3653 de1b bd dc 54
3656 3654 de1e 7e de 45
3657 3655 de21 7e de 45
3658 3656
3659 3657 de24 bd c0 f9
3660 3658 de27 dd f7 00 01
3661 3659 de2b dd fd 00 02
3662 3660 de2f de 03 00 03
3663 3661 de33 de 09 00 04
3664 3662 de37 de 0f 00 05
3665 3663 de3b de 15 00 06
3666 3664 de3f de 1b 00 07
3667 3665 de43 00 00
3668 3666
3669 3667
3670 3668 de45 39
3671 3669
3672 3670
3673 3671
3674 3672 de46 cc 00 00
3675 3673 de49 fd 70 02
3676 3674
3677 3675 de4c fc 70 02
3678 3676 de4f 34 06
3679 3677 de51 cc 00 02
3680 3678 de54 10 a3 e1
3681 3679 de57 2e 05
3682 3680 de59 cc 00 00
3683 3681 de5c 20 03
3684 3682 de5e cc 00 01
3685 3683 de61 10 83 00 00
3686 3684 de65 10 27 00 18
3687 3685 de69 7e de 7b
3688 3686
3689 3687 de6c fc 70 02
3690 3688 de6f c3 00 01
3691 3689 de72 fd 70 02
3692 3690 de75 83 00 01
3693 3691 de78 7e de 4c
3694 3692
3695 3693 de7b 7e de 6c
3696 3694 de7e 7e de 6c

cc167:
    jsr reg_disp
    jmp cc163

cc168:
    jsr enter_nu
    jmp cc163

cc169:
    jsr enter_nu
    jmp cc163

cc170:
    jsr enter_nu
    jmp cc163

cc171:
    jsr enter_nu
    jmp cc163

cc164:
    jsr ccswitch
    FDB cc165,1
    FDB cc166,2
    FDB cc167,3
    FDB cc168,4
    FDB cc169,5
    FDB cc170,6
    FDB cc171,7
    FDB 0

cc163:
cc161:
    rts

* .global delay_be
delay_be:
    ldd #0
    std j

cc174:
    ldd j
    pshs d
    ldd #2
    cmpd ,s++
    bgt *+7
    ldd #0
    bra *+5
    ldd #1
    cmpd #0
    lbeq cc173
    jmp cc175

cc172:
    ldd j
    addd #1
    std j
    subd #1
    jmp cc174

cc175:
    jmp cc172
    jmp cc172

```

```
3697 3695 cc173:  
3698 3696 de81 39 rts  
3699 3697  
3700 3698 * .global beep  
3701 3699 beep:  
3702 3700 de82 32 7f leas -1,s  
3703 3701 de84 fc 70 40 ldd port2  
3704 3702 de87 34 06 pshs d  
3705 3703 de89 cc 00 00 ldd #0  
3706 3704 de8c e7 f1 stb [ ,s++ ]  
3707 3705 de8e 31 e4 leay 0,s  
3708 3706 de90 1f 20 tfr y,d  
3709 3707 de92 34 06 pshs d  
3710 3708 de94 cc 00 00 ldd #0  
3711 3709 de97 e7 f1 stb [ ,s++ ]  
3712 3710 cc178:  
3713 3711 de99 31 e4 leay 0,s  
3714 3712 de9b 1f 20 tfr y,d  
3715 3713 de9d 34 06 pshs d  
3716 3714 de9f e6 f1 ldb [ ,s++ ]  
3717 3715 deal 1d sex  
3718 3716 dea2 34 06 pshs d  
3719 3717 dea4 cc 00 3c ldd #60  
3720 3718 dea7 10 a3 e1 cmpd ,s++  
3721 3719 deaa 2e 05 bgt *+7  
3722 3720 deac cc 00 00 ldd #0  
3723 3721 deaf 20 03 bra *+5  
3724 3722 deb1 cc 00 01 ldd #1  
3725 3723 deb4 10 83 00 00 cmpd #0  
3726 3724 deb8 10 27 00 38 lbeq cc177  
3727 3725 debc 7e de d5 jmp cc179  
3728 3726 cc176:  
3729 3727 debf 31 e4 leay 0,s  
3730 3728 dec1 1f 20 tfr y,d  
3731 3729 dec3 34 06 pshs d  
3732 3730 dec5 34 06 pshs d  
3733 3731 dec7 e6 f1 ldb [ ,s++ ]  
3734 3732 dec9 1d sex  
3735 3733 deca c3 00 01 addd #1  
3736 3734 decd e7 f1 stb [ ,s++ ]  
3737 3735 decf 83 00 01 subd #1  
3738 3736 ded2 7e de 99 jmp cc178  
3739 3737 cc179:  
3740 3738 ded5 fc 70 42 ldd port1  
3741 3739 ded8 34 06 pshs d  
3742 3740 deda cc 00 80 ldd #128  
3743 3741 dedd 43 coma  
3744 3742 dede 53 comb  
3745 3743 dedf e7 f1 stb [ ,s++ ]  
3746 3744 dee1 bd de 46 jsr delay_be  
3747 3745 dee4 fc 70 42 ldd port1  
3748 3746 dee7 34 06 pshs d  
3749 3747 dee9 cc 00 ff ldd #255  
3750 3748 deec e7 f1 stb [ ,s++ ]  
3751 3749 deee bd de 46 jsr delay_be  
3752 3750 def1 7e de bf jmp cc176
```

```
3753 3751 cc177:  
3754 3752 def4 32 61 leas 1,s  
3755 3753 def6 39 rts  
3756 3754  
3757 3755 * .global scan1  
3758 3756 scan1:  
3759 3757 cc180:  
3760 3758 def7 bd c8 84 jsr scan  
3761 3759 defa 34 06 pshs d  
3762 3760 defc cc 00 01 ldd #1  
3763 3761 deff 43 coma  
3764 3762 df00 53 comb  
3765 3763 df01 c3 00 01 addd #1  
3766 3764 df04 10 a3 e1 cmpd ,s++  
3767 3765 df07 26 05 bne *+7  
3768 3766 df09 cc 00 00 ldd #0  
3769 3767 df0c 20 03 bra *+5  
3770 3768 df0e cc 00 01 ldd #1  
3771 3769 df11 10 83 00 00 cmpd #0  
3772 3770 df15 10 27 00 06 lbeq cc181  
3773 3771 df19 7e de f7 jmp cc180  
3774 3772 df1c 7e de f7 jmp cc180  
3775 3773 cc181:  
3776 3774 df1f cc 00 03 ldd #3  
3777 3775 df22 34 06 pshs d  
3778 3776 df24 bd c8 41 jsr delay_ms  
3779 3777 df27 32 62 leas 2,s  
3780 3778 cc182:  
3781 3779 df29 bd c8 84 jsr scan  
3782 3780 df2c 34 06 pshs d  
3783 3781 df2e cc 00 01 ldd #1  
3784 3782 df31 43 coma  
3785 3783 df32 53 comb  
3786 3784 df33 c3 00 01 addd #1  
3787 3785 df36 10 a3 e1 cmpd ,s++  
3788 3786 df39 27 05 beq *+7  
3789 3787 df3b cc 00 00 ldd #0  
3790 3788 df3e 20 03 bra *+5  
3791 3789 df40 cc 00 01 ldd #1  
3792 3790 df43 10 83 00 00 cmpd #0  
3793 3791 df47 10 27 00 06 lbeq cc183  
3794 3792 df4b 7e df 29 jmp cc182  
3795 3793 df4e 7e df 29 jmp cc182  
3796 3794 cc183:  
3797 3795 df51 cc 00 03 ldd #3  
3798 3796 df54 34 06 pshs d  
3799 3797 df56 bd c8 41 jsr delay_ms  
3800 3798 df59 32 62 leas 2,s  
3801 3799 df5b bd c8 84 jsr scan  
3802 3800 df5e f7 70 09 stb key  
3803 3801 df61 f6 70 09 ldb key  
3804 3802 df64 1d sex  
3805 3803 df65 34 06 pshs d  
3806 3804 df67 bd c3 0f jsr key_code  
3807 3805 df6a 32 62 leas 2,s  
3808 3806 df6c f7 70 09 stb key
```

```
3809 3807 df6f bd dc fc          jsr key_exe
3810 3808 df72 39                rts
3811 3809
3812 3810                      * .global initacia
3813 3811 initacia:
3814 3812 df73 32 7b          leas -5,s
3815 3813 df75 31 62          leay 2,s
3816 3814 df77 1f 20          tfr y,d
3817 3815 df79 34 06          pshs d
3818 3816 df7b cc a0 00          ldd #-24576
3819 3817 df7e ed f1          std [,s++]
3820 3818 df80 31 e4          leay 0,s
3821 3819 df82 1f 20          tfr y,d
3822 3820 df84 34 06          pshs d
3823 3821 df86 cc 00 16          ldd #22
3824 3822 df89 e7 f1          stb [,s++]
3825 3823 df8b 31 61          leay 1,s
3826 3824 df8d 1f 20          tfr y,d
3827 3825 df8f 34 06          pshs d
3828 3826 df91 cc 00 03          ldd #3
3829 3827 df94 e7 f1          stb [,s++]
3830 3828 df96 31 62          leay 2,s
3831 3829 df98 1f 20          tfr y,d
3832 3830 df9a 34 06          pshs d
3833 3831 df9c ec f1          ldd [,s++]
3834 3832 df9e 34 06          pshs d
3835 3833 dfa0 31 63          leay 3,s
3836 3834 dfa2 1f 20          tfr y,d
3837 3835 dfa4 34 06          pshs d
3838 3836 dfa6 e6 f1          ldb [,s++]
3839 3837 dfa8 1d            sex
3840 3838 dfa9 e7 f1          stb [,s++]
3841 3839 dfab 31 62          leay 2,s
3842 3840 dfad 1f 20          tfr y,d
3843 3841 dfaf 34 06          pshs d
3844 3842 dfb1 ec f1          ldd [,s++]
3845 3843 dfb3 34 06          pshs d
3846 3844 dfb5 31 62          leay 2,s
3847 3845 dfb7 1f 20          tfr y,d
3848 3846 dfb9 34 06          pshs d
3849 3847 dfbb e6 f1          ldb [,s++]
3850 3848 dfbd 1d            sex
3851 3849 dfbe e7 f1          stb [,s++]
3852 3850 dfc0 31 64          leay 4,s
3853 3851 dfc2 1f 20          tfr y,d
3854 3852 dfc4 34 06          pshs d
3855 3853 dfc6 31 64          leay 4,s
3856 3854 dfc8 1f 20          tfr y,d
3857 3855 dfca 34 06          pshs d
3858 3856 dfcc ec f1          ldd [,s++]
3859 3857 dfce 34 06          pshs d
3860 3858 dfd0 cc 00 01          ldd #1
3861 3859 dfd3 e3 e1          addd ,s++
3862 3860 dfd5 34 06          pshs d
3863 3861 dfd7 e6 f1          ldb [,s++]
3864 3862 dfd9 1d            sex
```

```
3865 3863 dfda e7 f1          stb  [,s++]
3866 3864 dfdc 32 65          leas 5,s
3867 3865 dfde 39          rts
3868 3866
3869 3867          * .global putchar
3870 3868          putchar:
3871 3869 dfdf 32 7e          leas -2,s
3872 3870 dfe1 31 e4          leay 0,s
3873 3871 dfe3 1f 20          tfr  y,d
3874 3872 dfe5 34 06          pshs d
3875 3873 dfe7 cc a0 00          ldd  #-24576
3876 3874 dfea ed f1          std  [,s++]
3877 3875          cc184:
3878 3876 dfec 31 e4          leay 0,s
3879 3877 dfee 1f 20          tfr  y,d
3880 3878 dff0 34 06          pshs d
3881 3879 dff2 ec f1          ldd  [,s++]
3882 3880 dff4 34 06          pshs d
3883 3881 dff6 e6 f1          ldb  [,s++]
3884 3882 dff8 1d          sex
3885 3883 dff9 34 06          pshs d
3886 3884 dffb cc 00 02          ldd  #2
3887 3885 dffe a4 e0          anda ,s+
3888 3886 e000 e4 e0          andb ,s+
3889 3887 e002 34 06          pshs d
3890 3888 e004 cc 00 00          ldd  #0
3891 3889 e007 10 a3 e1          cmpd ,s++
3892 3890 e00a 27 05          beq  *+7
3893 3891 e00c cc 00 00          ldd  #0
3894 3892 e00f 20 03          bra  *+5
3895 3893 e011 cc 00 01          ldd  #1
3896 3894 e014 10 83 00 00          cmpd #0
3897 3895 e018 10 27 00 06          lbeq cc185
3898 3896 e01c 7e df ec          jmp  cc184
3899 3897 e01f 7e df ec          jmp  cc184
3900 3898          cc185:
3901 3899 e022 31 e4          leay 0,s
3902 3900 e024 1f 20          tfr  y,d
3903 3901 e026 34 06          pshs d
3904 3902 e028 ec f1          ldd  [,s++]
3905 3903 e02a 34 06          pshs d
3906 3904 e02c cc 00 01          ldd  #1
3907 3905 e02f e3 e1          addd ,s++
3908 3906 e031 34 06          pshs d
3909 3907 e033 31 67          leay 7,s
3910 3908 e035 1f 20          tfr  y,d
3911 3909 e037 34 06          pshs d
3912 3910 e039 e6 f1          ldb  [,s++]
3913 3911 e03b 1d          sex
3914 3912 e03c e7 f1          stb  [,s++]
3915 3913 e03e 32 62          leas 2,s
3916 3914 e040 39          rts
3917 3915
3918 3916          * .global puts
3919 3917          puts:
3920 3918          cc186:
```

```
3921 3919 e041 31 62          leay 2,s
3922 3920 e043 1f 20          tfr y,d
3923 3921 e045 34 06          pshs d
3924 3922 e047 ec f1          ldd [ ,s++ ]
3925 3923 e049 34 06          pshs d
3926 3924 e04b e6 f1          ldb [ ,s++ ]
3927 3925 e04d 1d             sex
3928 3926 e04e 10 83 00 00    cmpd #0
3929 3927 e052 10 27 00 29    lbeq cc187
3930 3928 e056 31 62          leay 2,s
3931 3929 e058 1f 20          tfr y,d
3932 3930 e05a 34 06          pshs d
3933 3931 e05c ec f1          ldd [ ,s++ ]
3934 3932 e05e 34 06          pshs d
3935 3933 e060 e6 f1          ldb [ ,s++ ]
3936 3934 e062 1d             sex
3937 3935 e063 34 06          pshs d
3938 3936 e065 bd df df      jsr putchar
3939 3937 e068 32 62          leas 2,s
3940 3938 e06a 31 62          leay 2,s
3941 3939 e06c 1f 20          tfr y,d
3942 3940 e06e 34 06          pshs d
3943 3941 e070 34 06          pshs d
3944 3942 e072 ec f1          ldd [ ,s++ ]
3945 3943 e074 c3 00 01      addd #1
3946 3944 e077 ed f1          std [ ,s++ ]
3947 3945 e079 83 00 01      subd #1
3948 3946 e07c 7e e0 41      jmp cc186
3949 3947                         cc187:
3950 3948 e07f 39             rts
3951 3949
3952 3950                         * .global getchar
3953 3951                         getchar:
3954 3952 e080 32 7d          leas -3,s
3955 3953 e082 31 e4          leay 0,s
3956 3954 e084 1f 20          tfr y,d
3957 3955 e086 34 06          pshs d
3958 3956 e088 cc a0 00      ldd #-24576
3959 3957 e08b ed f1          std [ ,s++ ]
3960 3958                         cc188:
3961 3959 e08d 31 e4          leay 0,s
3962 3960 e08f 1f 20          tfr y,d
3963 3961 e091 34 06          pshs d
3964 3962 e093 ec f1          ldd [ ,s++ ]
3965 3963 e095 34 06          pshs d
3966 3964 e097 e6 f1          ldb [ ,s++ ]
3967 3965 e099 1d             sex
3968 3966 e09a 34 06          pshs d
3969 3967 e09c cc 00 01      ldd #1
3970 3968 e09f a4 e0          anda ,s+
3971 3969 e0a1 e4 e0          andb ,s+
3972 3970 e0a3 34 06          pshs d
3973 3971 e0a5 cc 00 00      ldd #0
3974 3972 e0a8 10 a3 e1      cmpd ,s++
3975 3973 e0ab 27 05          beq *+7
3976 3974 e0ad cc 00 00      ldd #0
```

```
3977 3975 e0b0 20 03          bra  *+5
3978 3976 e0b2 cc 00 01          ldd  #1
3979 3977 e0b5 10 83 00 00          cmpd #0
3980 3978 e0b9 10 27 00 06          lbeq cc189
3981 3979 e0bd 7e e0 8d          jmp  cc188
3982 3980 e0c0 7e e0 8d          jmp  cc188
3983 3981                         cc189:
3984 3982 e0c3 31 62          leay 2,s
3985 3983 e0c5 1f 20          tfr  y,d
3986 3984 e0c7 34 06          pshs d
3987 3985 e0c9 31 62          leay 2,s
3988 3986 e0cb 1f 20          tfr  y,d
3989 3987 e0cd 34 06          pshs d
3990 3988 e0cf ec f1          ldd  [,s++]
3991 3989 e0d1 34 06          pshs d
3992 3990 e0d3 cc 00 01          ldd  #1
3993 3991 e0d6 e3 e1          addd ,s++
3994 3992 e0d8 34 06          pshs d
3995 3993 e0da e6 f1          ldb  [,s++]
3996 3994 e0dc 1d             sex
3997 3995 e0dd e7 f1          stb  [,s++]
3998 3996 e0df 31 62          leay 2,s
3999 3997 e0e1 1f 20          tfr  y,d
4000 3998 e0e3 34 06          pshs d
4001 3999 e0e5 e6 f1          ldb  [,s++]
4002 4000 e0e7 1d             sex
4003 4001 e0e8 32 63          leas 3,s
4004 4002 e0ea 39             rts
4005 4003
4006 4004
4007 4005
4008 4006 e0eb cc 00 0a          * .global newline
4009 4007 e0ee 34 06          newline:
4010 4008 e0f0 bd df df          ldd  #10
4011 4009 e0f3 32 62          pshs d
4012 4010 e0f5 cc 00 0d          jsr  putchar
4013 4011 e0f8 34 06          leas 2,s
4014 4012 e0fa bd df df          ldd  #13
4015 4013 e0fd 32 62          pshs d
4016 4014 e0ff 39             jsr  putchar
4017 4015
4018 4016
4019 4017
4020 4018 e100 31 63          leas 2,s
4021 4019 e102 1f 20          rts
4022 4020 e104 34 06          * .global send_hex
4023 4021 e106 e6 f1          send_hex:
4024 4022 e108 1d             leay 3,s
4025 4023 e109 34 06          tfr  y,d
4026 4024 e10b cc 00 04          pshs d
4027 4025 e10e 5a             ldb  [,s++]
4028 4026 e10f 2d 06          sex
4029 4027 e111 64 e4          pshs d
4030 4028 e113 66 61          ldd  #4
4031 4029 e115 20 f7          decb
4032 4030 e117 35 06          blt  *+8
                                lsr  ,s
                                ror  1,s
                                bra  *-7
                                puls d
```

4033	4031	e119	f7	70	05	stb	k	
4034	4032	e11c	f6	70	05	ldb	k	
4035	4033	e11f	1d			sex		
4036	4034	e120	34	06		pshs	d	
4037	4035	e122	cc	00	0f	ldd	#15	
4038	4036	e125	a4	e0		anda	,s+	
4039	4037	e127	e4	e0		andb	,s+	
4040	4038	e129	f7	70	05	stb	k	
4041	4039	e12c	f6	70	05	ldb	k	
4042	4040	e12f	1d			sex		
4043	4041	e130	34	06		pshs	d	
4044	4042	e132	cc	00	09	ldd	#9	
4045	4043	e135	10	a3	e1	cmpd	,s++	
4046	4044	e138	2d	05		blt	*+7	
4047	4045	e13a	cc	00	00	ldd	#0	
4048	4046	e13d	20	03		bra	*+5	
4049	4047	e13f	cc	00	01	ldd	#1	
4050	4048	e142	10	83	00	00	cmpd	#0
4051	4049	e146	10	27	00	15	lbeq	cc190
4052	4050	e14a	f6	70	05	ldb	k	
4053	4051	e14d	1d			sex		
4054	4052	e14e	34	06		pshs	d	
4055	4053	e150	cc	00	37	ldd	#55	
4056	4054	e153	e3	e1		addd	,s++	
4057	4055	e155	34	06		pshs	d	
4058	4056	e157	bd	df	df	jsr	putchar	
4059	4057	e15a	32	62		leas	2,s	
4060	4058	e15c	7e	e1	71	jmp	cc191	
4061	4059					cc190:		
4062	4060	e15f	f6	70	05	ldb	k	
4063	4061	e162	1d			sex		
4064	4062	e163	34	06		pshs	d	
4065	4063	e165	cc	00	30	ldd	#48	
4066	4064	e168	e3	e1		addd	,s++	
4067	4065	e16a	34	06		pshs	d	
4068	4066	e16c	bd	df	df	jsr	putchar	
4069	4067	e16f	32	62		leas	2,s	
4070	4068					cc191:		
4071	4069	e171	31	63		leay	3,s	
4072	4070	e173	1f	20		tfr	y,d	
4073	4071	e175	34	06		pshs	d	
4074	4072	e177	e6	f1		ldb	[,s++]	
4075	4073	e179	1d			sex		
4076	4074	e17a	34	06		pshs	d	
4077	4075	e17c	cc	00	0f	ldd	#15	
4078	4076	e17f	a4	e0		anda	,s+	
4079	4077	e181	e4	e0		andb	,s+	
4080	4078	e183	f7	70	05	stb	k	
4081	4079	e186	f6	70	05	ldb	k	
4082	4080	e189	1d			sex		
4083	4081	e18a	34	06		pshs	d	
4084	4082	e18c	cc	00	09	ldd	#9	
4085	4083	e18f	10	a3	e1	cmpd	,s++	
4086	4084	e192	2d	05		blt	*+7	
4087	4085	e194	cc	00	00	ldd	#0	
4088	4086	e197	20	03		bra	*+5	

```
4089 4087 e199 cc 00 01      ldd #1
4090 4088 e19c 10 83 00 00    cmpd #0
4091 4089 e1a0 10 27 00 15    lbeq cc192
4092 4090 e1a4 f6 70 05      ldb k
4093 4091 e1a7 1d           sex
4094 4092 e1a8 34 06      pshs d
4095 4093 e1aa cc 00 37      ldd #55
4096 4094 e1ad e3 e1      addd ,s++
4097 4095 e1af 34 06      pshs d
4098 4096 e1b1 bd df df    jsr putchar
4099 4097 e1b4 32 62      leas 2,s
4100 4098 e1b6 7e e1 cb    jmp cc193
4101 4099                   cc192:
4102 4100 e1b9 f6 70 05    ldb k
4103 4101 e1bc 1d           sex
4104 4102 e1bd 34 06      pshs d
4105 4103 e1bf cc 00 30      ldd #48
4106 4104 e1c2 e3 e1      addd ,s++
4107 4105 e1c4 34 06      pshs d
4108 4106 e1c6 bd df df    jsr putchar
4109 4107 e1c9 32 62      leas 2,s
4110 4108                   cc193:
4111 4109 e1cb 39           rts
4112 4110
4113 4111                   * .global send_wor
4114 4112                   send_wor:
4115 4113 e1cc 31 62      leay 2,s
4116 4114 e1ce 1f 20      tfr y,d
4117 4115 e1d0 34 06      pshs d
4118 4116 e1d2 ec f1      ldd [,s++]
4119 4117 e1d4 34 06      pshs d
4120 4118 e1d6 cc 00 08      ldd #8
4121 4119 e1d9 5a           decb
4122 4120 e1da 2d 06      blt *+8
4123 4121 e1dc 64 e4      lsr ,s
4124 4122 e1de 66 61      ror 1,s
4125 4123 e1e0 20 f7      bra *-7
4126 4124 e1e2 35 06      puls d
4127 4125 e1e4 fd 70 17    std temp16
4128 4126 e1e7 fc 70 17    ldd temp16
4129 4127 e1ea 34 06      pshs d
4130 4128 e1ec cc 00 ff      ldd #255
4131 4129 e1ef a4 e0      anda ,s+
4132 4130 e1f1 e4 e0      andb ,s+
4133 4131 e1f3 f7 70 05    stb k
4134 4132 e1f6 f6 70 05    ldb k
4135 4133 e1f9 1d           sex
4136 4134 e1fa 34 06      pshs d
4137 4135 e1fc bd e1 00    jsr send_hex
4138 4136 e1ff 32 62      leas 2,s
4139 4137 e201 31 62      leay 2,s
4140 4138 e203 1f 20      tfr y,d
4141 4139 e205 34 06      pshs d
4142 4140 e207 ec f1      ldd [,s++]
4143 4141 e209 34 06      pshs d
4144 4142 e20b cc 00 ff      ldd #255
```

```
4145 4143 e20e a4 e0           anda ,s+
4146 4144 e210 e4 e0           andb ,s+
4147 4145 e212 f7 70 05       stb k
4148 4146 e215 f6 70 05       ldb k
4149 4147 e218 1d             sex
4150 4148 e219 34 06           pshs d
4151 4149 e21b bd e1 00       jsr send_hex
4152 4150 e21e 32 62           leas 2,s
4153 4151 e220 39             rts
4154
4155 4153
4156 4154
4157 4155 e221 32 7c           * .global key_dump
4158 4156 e223 fc 70 19       key_dump:
4159 4157 e226 fd 70 46       leas -4,s
4160 4158 e229 31 62           ldd PC
4161 4159 e22b 1f 20           std dptr
4162 4160 e22d 34 06           leay 2,s
4163 4161 e22f cc 00 00       tfr y,d
4164 4162 e232 ed f1           pshs d
4165 4163
4166 4164 e234 31 62           ldd #0
4167 4165 e236 1f 20           std [,s++]
4168 4166 e238 34 06           cc196:
4169 4167 e23a ec f1           leay 2,s
4170 4168 e23c 34 06           tfr y,d
4171 4169 e23e cc 00 10       pshs d
4172 4170 e241 10 a3 e1       ldd [,s++]
4173 4171 e244 2e 05           bgt *+7
4174 4172 e246 cc 00 00       ldd #0
4175 4173 e249 20 03           bra *+5
4176 4174 e24b cc 00 01       ldd #1
4177 4175 e24e 10 83 00 00     cmpd #0
4178 4176 e252 10 27 01 7a     lbeq cc195
4179 4177 e256 7e e2 6e       jmp cc197
4180 4178
4181 4179 e259 31 62           cc194:
4182 4180 e25b 1f 20           leay 2,s
4183 4181 e25d 34 06           tfr y,d
4184 4182 e25f 34 06           pshs d
4185 4183 e261 ec f1           pshs d
4186 4184 e263 c3 00 01       ldd [,s++]
4187 4185 e266 ed f1           addd #1
4188 4186 e268 83 00 01       std [,s++]
4189 4187 e26b 7e e2 34       subd #1
4190 4188
4191 4189 e26e bd e0 eb       jmp cc196
4192 4190 e271 fc 70 46       cc197:
4193 4191 e274 34 06           jsr newline
4194 4192 e276 bd e1 cc       ldd dptr
4195 4193 e279 32 62           pshs d
4196 4194 e27b cc 00 3a       jsr send_wor
4197 4195 e27e 34 06           leas 2,s
4198 4196 e280 bd df df       ldd #58
4199 4197 e283 32 62           pshs d
4200 4198 e285 31 e4           jsr putchar
                                         leas 2,s
                                         leay 0,s
```

```
4201 4199 e287 1f 20          tfr y,d
4202 4200 e289 34 06          pshs d
4203 4201 e28b cc 00 00          ldd #0
4204 4202 e28e ed f1          std [,s++]
4205 4203                      cc200:
4206 4204 e290 31 e4          leay 0,s
4207 4205 e292 1f 20          tfr y,d
4208 4206 e294 34 06          pshs d
4209 4207 e296 ec f1          ldd [,s++]
4210 4208 e298 34 06          pshs d
4211 4209 e29a cc 00 10          ldd #16
4212 4210 e29d 10 a3 e1          cmpd ,s++
4213 4211 e2a0 2e 05          bgt *+7
4214 4212 e2a2 cc 00 00          ldd #0
4215 4213 e2a5 20 03          bra *+5
4216 4214 e2a7 cc 00 01          ldd #1
4217 4215 e2aa 10 83 00 00          cmpd #0
4218 4216 e2ae 10 27 00 40          lbeq cc199
4219 4217 e2b2 7e e2 ca          jmp cc201
4220 4218                      cc198:
4221 4219 e2b5 31 e4          leay 0,s
4222 4220 e2b7 1f 20          tfr y,d
4223 4221 e2b9 34 06          pshs d
4224 4222 e2bb 34 06          pshs d
4225 4223 e2bd ec f1          ldd [,s++]
4226 4224 e2bf c3 00 01          addd #1
4227 4225 e2c2 ed f1          std [,s++]
4228 4226 e2c4 83 00 01          subd #1
4229 4227 e2c7 7e e2 90          jmp cc200
4230 4228                      cc201:
4231 4229 e2ca fc 70 46          ldd dptr
4232 4230 e2cd 34 06          pshs d
4233 4231 e2cf 31 62          leay 2,s
4234 4232 e2d1 1f 20          tfr y,d
4235 4233 e2d3 34 06          pshs d
4236 4234 e2d5 ec f1          ldd [,s++]
4237 4235 e2d7 e3 e1          addd ,s++
4238 4236 e2d9 34 06          pshs d
4239 4237 e2db e6 f1          ldb [,s++]
4240 4238 e2dd 1d          sex
4241 4239 e2de 34 06          pshs d
4242 4240 e2e0 bd e1 00          jsr send_hex
4243 4241 e2e3 32 62          leas 2,s
4244 4242 e2e5 cc 00 20          ldd #32
4245 4243 e2e8 34 06          pshs d
4246 4244 e2ea bd df df          jsr putchar
4247 4245 e2ed 32 62          leas 2,s
4248 4246 e2ef 7e e2 b5          jmp cc198
4249 4247                      cc199:
4250 4248 e2f2 cc 00 20          ldd #32
4251 4249 e2f5 34 06          pshs d
4252 4250 e2f7 bd df df          jsr putchar
4253 4251 e2fa 32 62          leas 2,s
4254 4252 e2fc 31 e4          leay 0,s
4255 4253 e2fe 1f 20          tfr y,d
4256 4254 e300 34 06          pshs d
```

```
4257 4255 e302 cc 00 00      ldd #0
4258 4256 e305 ed f1      std [ ,s++ ]
4259 4257
4260 4258 e307 31 e4      cc204:
4261 4259 e309 1f 20      leay 0,s
4262 4260 e30b 34 06      tfr y,d
4263 4261 e30d ec f1      pshs d
4264 4262 e30f 34 06      ldd [ ,s++ ]
4265 4263 e311 cc 00 10      pshs d
4266 4264 e314 10 a3 e1      ldd #16
4267 4265 e317 2e 05      cmpd ,s++ 
4268 4266 e319 cc 00 00      bgt *+7
4269 4267 e31c 20 03      ldd #0
4270 4268 e31e cc 00 01      bra *+5
4271 4269 e321 10 83 00 00      ldd #1
4272 4270 e325 10 27 00 91      cmpd #0
4273 4271 e329 7e e3 41      lbeq cc203
4274 4272      jmp cc205
4275 4273 e32c 31 e4      cc202:
4276 4274 e32e 1f 20      leay 0,s
4277 4275 e330 34 06      tfr y,d
4278 4276 e332 34 06      pshs d
4279 4277 e334 ec f1      pshs d
4280 4278 e336 c3 00 01      ldd [ ,s++ ]
4281 4279 e339 ed f1      addd #1
4282 4280 e33b 83 00 01      std [ ,s++ ]
4283 4281 e33e 7e e3 07      subd #1
4284 4282      jmp cc204
4285 4283 e341 fc 70 46      cc205:
4286 4284 e344 34 06      ldd dptr
4287 4285 e346 31 62      pshs d
4288 4286 e348 1f 20      leay 2,s
4289 4287 e34a 34 06      tfr y,d
4290 4288 e34c ec f1      pshs d
4291 4289 e34e e3 e1      ldd [ ,s++ ]
4292 4290 e350 34 06      addd ,s++ 
4293 4291 e352 e6 f1      pshs d
4294 4292 e354 1d      ldb [ ,s++ ]
4295 4293 e355 f7 70 07      sex
4296 4294 e358 f6 70 07      stb q
4297 4295 e35b 1d      ldb q
4298 4296 e35c 34 06      sex
4299 4297 e35e cc 00 20      pshs d
4300 4298 e361 10 a3 e1      ldd #32
4301 4299 e364 2f 05      cmpd ,s++ 
4302 4300 e366 cc 00 00      ble *+7
4303 4301 e369 20 03      ldd #0
4304 4302 e36b cc 00 01      bra *+5
4305 4303 e36e 10 83 00 00      ldd #1
4306 4304 e372 10 27 00 21      cmpd #0
4307 4305 e376 f6 70 07      lbeq cc207 ;_ instruction flagged for non o
4308 4306 e379 1d      ldb q
4309 4307 e37a 34 06      sex
4310 4308 e37c cc 00 80      pshs d
4311 4309 e37f 10 a3 e1      ldd #128
4312 4310 e382 2e 05      cmpd ,s++ 
4313 4311 e384 34 06      bgt *+7
```

```

4313 4311 e384 cc 00 00      ldd #0
4314 4312 e387 20 03      bra *+5
4315 4313 e389 cc 00 01      ldd #1
4316 4314 e38c 10 83 00 00      cmpd #0
4317 4315 e390 10 27 00 03      lbeq cc207 ;_ instruction flagged for non o:
4318 4316 e394 cc 00 01      ldd #1
4319 4317      cc207:
4320 4318 e397 10 83 00 00      cmpd #0
4321 4319 e39b 10 27 00 0e      lbeq cc206
4322 4320 e39f f6 70 07      ldb q
4323 4321 e3a2 1d      sex
4324 4322 e3a3 34 06      pshs d
4325 4323 e3a5 bd df df      jsr putchar
4326 4324 e3a8 32 62      leas 2,s
4327 4325 e3aa 7e e3 b7      jmp cc208
4328 4326      cc206:
4329 4327 e3ad cc 00 2e      ldd #46
4330 4328 e3b0 34 06      pshs d
4331 4329 e3b2 bd df df      jsr putchar
4332 4330 e3b5 32 62      leas 2,s
4333 4331      cc208:
4334 4332 e3b7 7e e3 2c      jmp cc202
4335 4333      cc203:
4336 4334 e3ba cc 70 46      ldd #dptr
4337 4335 e3bd 34 06      pshs d
4338 4336 e3bf cc 00 10      ldd #16
4339 4337 e3c2 ae f4      ldx [ ,s]
4340 4338 e3c4 34 10      pshs x
4341 4339 e3c6 e3 e1      addd ,s++
4342 4340 e3c8 fd 70 46      std dptr
4343 4341 e3cb 32 62      leas 2,s
4344 4342 e3cd 7e e2 59      jmp cc194
4345 4343      cc195:
4346 4344 e3d0 bd e0 eb      jsr newline
4347 4345 e3d3 fc 70 46      ldd dptr
4348 4346 e3d6 fd 70 19      std PC
4349 4347 e3d9 bd ce 10      jsr key_addr
4350 4348 e3dc 32 64      leas 4,s
4351 4349 e3de 39      rts
4352 4350
4353 4351      * .global nibble2h
4354 4352      nibble2h:
4355 4353 e3df 32 7f      leas -1,s
4356 4354 e3e1 31 64      leay 4,s
4357 4355 e3e3 1f 20      tfr y,d
4358 4356 e3e5 34 06      pshs d
4359 4357 e3e7 e6 f1      ldb [ ,s++ ]
4360 4358 e3e9 1d      sex
4361 4359 e3ea 34 06      pshs d
4362 4360 e3ec cc 00 40      ldd #64
4363 4361 e3ef 10 a3 e1      cmpd ,s++
4364 4362 e3f2 2e 05      bgt *+7
4365 4363 e3f4 cc 00 00      ldd #0
4366 4364 e3f7 20 03      bra *+5
4367 4365 e3f9 cc 00 01      ldd #1
4368 4366 e3fc 10 83 00 00      cmpd #0

```

```
4369 4367 e400 10 27 00 19      lbeq cc209
4370 4368 e404 31 64      leay 4,s
4371 4369 e406 1f 20      tfr y,d
4372 4370 e408 34 06      pshs d
4373 4371 e40a e6 f1      ldb [ ,s++ ]
4374 4372 e40c 1d      sex
4375 4373 e40d 34 06      pshs d
4376 4374 e40f cc 00 30      ldd #48
4377 4375 e412 35 10      puls x
4378 4376 e414 34 06      pshs d
4379 4377 e416 1f 10      tfr x,d
4380 4378 e418 a3 e1      subd ,s++ 
4381 4379 e41a 32 61      leas 1,s
4382 4380 e41c 39      rts
4383 4381      cc209:
4384 4382 e41d 31 64      leay 4,s
4385 4383 e41f 1f 20      tfr y,d
4386 4384 e421 34 06      pshs d
4387 4385 e423 e6 f1      ldb [ ,s++ ]
4388 4386 e425 1d      sex
4389 4387 e426 34 06      pshs d
4390 4388 e428 cc 00 37      ldd #55
4391 4389 e42b 35 10      puls x
4392 4390 e42d 34 06      pshs d
4393 4391 e42f 1f 10      tfr x,d
4394 4392 e431 a3 e1      subd ,s++ 
4395 4393 e433 32 61      leas 1,s
4396 4394 e435 39      rts
4397 4395      cc210:
4398 4396 e436 32 61      leas 1,s
4399 4397 e438 39      rts
4400 4398
4401 4399      * .global gethex
4402 4400      gethex:
4403 4401 e439 32 7c      leas -4,s
4404 4402 e43b 31 62      leay 2,s
4405 4403 e43d 1f 20      tfr y,d
4406 4404 e43f 34 06      pshs d
4407 4405 e441 bd e7 7a      jsr getchar2
4408 4406 e444 ed f1      std [ ,s++ ]
4409 4407 e446 31 e4      leay 0,s
4410 4408 e448 1f 20      tfr y,d
4411 4409 e44a 34 06      pshs d
4412 4410 e44c bd e7 7a      jsr getchar2
4413 4411 e44f ed f1      std [ ,s++ ]
4414 4412 e451 31 62      leay 2,s
4415 4413 e453 1f 20      tfr y,d
4416 4414 e455 34 06      pshs d
4417 4415 e457 31 64      leay 4,s
4418 4416 e459 1f 20      tfr y,d
4419 4417 e45b 34 06      pshs d
4420 4418 e45d ec f1      ldd [ ,s++ ]
4421 4419 e45f 34 06      pshs d
4422 4420 e461 bd e3 df      jsr nibble2h
4423 4421 e464 32 62      leas 2,s
4424 4422 e466 34 06      pshs d
```

```

4425 4423 e468 cc 00 04      ldd #4
4426 4424 e46b 5a          decb
4427 4425 e46c 2d 06      blt *+8
4428 4426 e46e 68 61      asl 1,s
4429 4427 e470 69 e4      rol ,s
4430 4428 e472 20 f7      bra *-7
4431 4429 e474 35 06      puls d
4432 4430 e476 ed f1      std [,s++]
4433 4431 e478 31 e4      leay 0,s
4434 4432 e47a 1f 20      tfr y,d
4435 4433 e47c 34 06      pshs d
4436 4434 e47e 31 62      leay 2,s
4437 4435 e480 1f 20      tfr y,d
4438 4436 e482 34 06      pshs d
4439 4437 e484 ec f1      ldd [,s++]
4440 4438 e486 34 06      pshs d
4441 4439 e488 bd e3 df      jsr nibble2h
4442 4440 e48b 32 62      leas 2,s
4443 4441 e48d ed f1      std [,s++]
4444 4442 e48f 31 62      leay 2,s
4445 4443 e491 1f 20      tfr y,d
4446 4444 e493 34 06      pshs d
4447 4445 e495 31 64      leay 4,s
4448 4446 e497 1f 20      tfr y,d
4449 4447 e499 34 06      pshs d
4450 4448 e49b ec f1      ldd [,s++]
4451 4449 e49d 34 06      pshs d
4452 4450 e49f 31 64      leay 4,s
4453 4451 e4a1 1f 20      tfr y,d
4454 4452 e4a3 34 06      pshs d
4455 4453 e4a5 ec f1      ldd [,s++]
4456 4454 e4a7 aa e0      ora ,s+
4457 4455 e4a9 ea e0      orb ,s+
4458 4456 e4ab ed f1      std [,s++]
4459 4457 e4ad fc 70 0f      ldd bcc
4460 4458 e4b0 34 06      pshs d
4461 4459 e4b2 31 64      leay 4,s
4462 4460 e4b4 1f 20      tfr y,d
4463 4461 e4b6 34 06      pshs d
4464 4462 e4b8 ec f1      ldd [,s++]
4465 4463 e4ba e3 e1      addd ,s++
4466 4464 e4bc fd 70 0f      std bcc
4467 4465 e4bf 31 62      leay 2,s
4468 4466 e4c1 1f 20      tfr y,d
4469 4467 e4c3 34 06      pshs d
4470 4468 e4c5 ec f1      ldd [,s++]
4471 4469 e4c7 32 64      leas 4,s
4472 4470 e4c9 39      rts
4473 4471
4474 4472      * .global get16bit
4475 4473      get16bit:
4476 4474 e4ca 32 7e      leas -2,s
4477 4475 e4cc 31 e4      leay 0,s
4478 4476 e4ce 1f 20      tfr y,d
4479 4477 e4d0 34 06      pshs d
4480 4478 e4d2 cc 00 00      ldd #0

```

```
4481 4479 e4d5 ed f1           std  [,s++]
4482 4480 e4d7 31 e4           leay 0,s
4483 4481 e4d9 1f 20           tfr  y,d
4484 4482 e4db 34 06           pshs d
4485 4483 e4dd bd e4 39       jsr  gethex
4486 4484 e4e0 ae f4           ldx  [,s]
4487 4485 e4e2 34 10           pshs x
4488 4486 e4e4 aa e0           ora  ,s+
4489 4487 e4e6 ea e0           orb  ,s+
4490 4488 e4e8 ed f1           std  [,s++]
4491 4489 e4ea 31 e4           leay 0,s
4492 4490 e4ec 1f 20           tfr  y,d
4493 4491 e4ee 34 06           pshs d
4494 4492 e4f0 cc 00 08       ldd  #8
4495 4493 e4f3 ae f4           ldx  [,s]
4496 4494 e4f5 34 10           pshs x
4497 4495 e4f7 5a             decb
4498 4496 e4f8 2d 06           blt  *+8
4499 4497 e4fa 68 61           asl  1,s
4500 4498 e4fc 69 e4           rol  ,s
4501 4499 e4fe 20 f7           bra  *-7
4502 4500 e500 35 06           puls d
4503 4501 e502 ed f1           std  [,s++]
4504 4502 e504 31 e4           leay 0,s
4505 4503 e506 1f 20           tfr  y,d
4506 4504 e508 34 06           pshs d
4507 4505 e50a bd e4 39       jsr  gethex
4508 4506 e50d ae f4           ldx  [,s]
4509 4507 e50f 34 10           pshs x
4510 4508 e511 aa e0           ora  ,s+
4511 4509 e513 ea e0           orb  ,s+
4512 4510 e515 ed f1           std  [,s++]
4513 4511 e517 31 e4           leay 0,s
4514 4512 e519 1f 20           tfr  y,d
4515 4513 e51b 34 06           pshs d
4516 4514 e51d ec f1           ldd  [,s++]
4517 4515 e51f 32 62           leas 2,s
4518 4516 e521 39             rts
4519
4520 4518
4521 4519
4522 4520 e522 32 7c           * .global read_rec
4523 4521 e524 cc 00 00         read_rec:
4524 4522 e527 fd 70 0f         leas -4,s
4525 4523 e52a 31 62           ldd  #0
4526 4524 e52c 1f 20           std  bcc
4527 4525 e52e 34 06           leay 2,s
4528 4526 e530 bd e4 39       tfr  y,d
4529 4527 e533 34 06           pshs d
4530 4528 e535 cc 00 03       puls x
4531 4529 e538 35 10           ldd  #3
4532 4530 e53a 34 06           pshs d
4533 4531 e53c 1f 10           tfr  x,d
4534 4532 e53e a3 e1           subd ,s++
4535 4533 e540 e7 f1           stb  [,s++]
4536 4534 e542 31 e4           leay 0,s
```

```
4537 4535 e544 1f 20          tfr y,d
4538 4536 e546 34 06          pshs d
4539 4537 e548 bd e4 ca       jsr get16bit
4540 4538 e54b ed f1          std [,s++]
4541 4539 e54d 31 e4          leay 0,s
4542 4540 e54f 1f 20          tfr y,d
4543 4541 e551 34 06          pshs d
4544 4542 e553 ec f1          ldd [,s++]
4545 4543 e555 fd 70 46       std dptr
4546 4544 e558 31 63          leay 3,s
4547 4545 e55a 1f 20          tfr y,d
4548 4546 e55c 34 06          pshs d
4549 4547 e55e cc 00 00       ldd #0
4550 4548 e561 e7 f1          stb [,s++]
4551 4549                         cc213:
4552 4550 e563 31 63          leay 3,s
4553 4551 e565 1f 20          tfr y,d
4554 4552 e567 34 06          pshs d
4555 4553 e569 e6 f1          ldb [,s++]
4556 4554 e56b 1d             sex
4557 4555 e56c 34 06          pshs d
4558 4556 e56e 31 64          leay 4,s
4559 4557 e570 1f 20          tfr y,d
4560 4558 e572 34 06          pshs d
4561 4559 e574 e6 f1          ldb [,s++]
4562 4560 e576 1d             sex
4563 4561 e577 10 a3 e1       cmpd ,s++
4564 4562 e57a 2e 05          bgt *+7
4565 4563 e57c cc 00 00       ldd #0
4566 4564 e57f 20 03          bra *+5
4567 4565 e581 cc 00 01       ldd #1
4568 4566 e584 10 83 00 00     cmpd #0
4569 4567 e588 10 27 00 33     lbeq cc212
4570 4568 e58c 7e e5 a5       jmp cc214
4571 4569                         cc211:
4572 4570 e58f 31 63          leay 3,s
4573 4571 e591 1f 20          tfr y,d
4574 4572 e593 34 06          pshs d
4575 4573 e595 34 06          pshs d
4576 4574 e597 e6 f1          ldb [,s++]
4577 4575 e599 1d             sex
4578 4576 e59a c3 00 01       addd #1
4579 4577 e59d e7 f1          stb [,s++]
4580 4578 e59f 83 00 01       subd #1
4581 4579 e5a2 7e e5 63       jmp cc213
4582 4580                         cc214:
4583 4581 e5a5 fc 70 46       ldd dptr
4584 4582 e5a8 34 06          pshs d
4585 4583 e5aa 31 65          leay 5,s
4586 4584 e5ac 1f 20          tfr y,d
4587 4585 e5ae 34 06          pshs d
4588 4586 e5b0 e6 f1          ldb [,s++]
4589 4587 e5b2 1d             sex
4590 4588 e5b3 e3 e1          addd ,s++
4591 4589 e5b5 34 06          pshs d
4592 4590 e5b7 bd e4 39       jsr gethex
```

```
4593 4591 e5ba e7 f1          stb  [,s++]
4594 4592 e5bc 7e e5 8f        jmp  cc211
4595 4593                         cc212:
4596 4594 e5bf fc 70 0f        ldd  bcc
4597 4595 e5c2 43             coma
4598 4596 e5c3 53             comb
4599 4597 e5c4 fd 70 0f        std  bcc
4600 4598 e5c7 fc 70 0f        ldd  bcc
4601 4599 e5ca 34 06          pshs d
4602 4600 e5cc cc 00 ff        ldd  #255
4603 4601 e5cf a4 e0          anda ,s+
4604 4602 e5d1 e4 e0          andb ,s+
4605 4603 e5d3 fd 70 11        std  save_bcc
4606 4604 e5d6 fc 70 11        ldd  save_bcc
4607 4605 e5d9 34 06          pshs d
4608 4606 e5db bd e4 39        jsr  gethex
4609 4607 e5de 10 a3 e1        cmpd ,s++
4610 4608 e5e1 26 05          bne  *+7
4611 4609 e5e3 cc 00 00        ldd  #0
4612 4610 e5e6 20 03          bra  *+5
4613 4611 e5e8 cc 00 01        ldd  #1
4614 4612 e5eb 10 83 00 00      cmpd #0
4615 4613 e5ef 10 27 00 06      lbeq cc215
4616 4614 e5f3 cc 00 01        ldd  #1
4617 4615 e5f6 fd 70 13        std  bcc_errno
4618 4616                         cc215:
4619 4617 e5f9 32 64          leas 4,s
4620 4618 e5fb 39             rts
4621 4619
4622 4620                         * .global get_s_re
4623 4621                         get_s_re:
4624 4622 e5fc cc 00 00        ldd  #0
4625 4623 e5ff fd 70 21        std  end
4626 4624 e602 cc 00 00        ldd  #0
4627 4625 e605 fd 70 13        std  bcc_errno
4628 4626                         cc216:
4629 4627 e608 fc 70 21        ldd  end
4630 4628 e60b 34 06          pshs d
4631 4629 e60d cc 00 00        ldd  #0
4632 4630 e610 10 a3 e1        cmpd ,s++
4633 4631 e613 27 05          beq  *+7
4634 4632 e615 cc 00 00        ldd  #0
4635 4633 e618 20 03          bra  *+5
4636 4634 e61a cc 00 01        ldd  #1
4637 4635 e61d 10 83 00 00      cmpd #0
4638 4636 e621 10 27 00 65      lbeq cc217
4639 4637                         cc218:
4640 4638 e625 bd e7 7a        jsr  getchar2
4641 4639 e628 34 06          pshs d
4642 4640 e62a cc 00 53        ldd  #83
4643 4641 e62d 10 a3 e1        cmpd ,s++
4644 4642 e630 26 05          bne  *+7
4645 4643 e632 cc 00 00        ldd  #0
4646 4644 e635 20 03          bra  *+5
4647 4645 e637 cc 00 01        ldd  #1
4648 4646 e63a 10 83 00 00      cmpd #0
```

```
4649 4647 e63e 10 27 00 06      lbeq cc219
4650 4648 e642 7e e6 25      jmp cc218
4651 4649 e645 7e e6 25      jmp cc218
4652 4650      cc219:
4653 4651 e648 bd e7 7a      jsr getchar2
4654 4652 e64b 7e e6 72      jmp cc222
4655 4653      cc223:
4656 4654 e64e cc 00 00      ldd #0
4657 4655 e651 fd 70 21      std end
4658 4656 e654 7e e6 87      jmp cc221
4659 4657      cc224:
4660 4658 e657 bd e5 22      jsr read_rec
4661 4659 e65a 7e e6 87      jmp cc221
4662 4660      cc225:
4663 4661 e65d cc 00 01      ldd #1
4664 4662 e660 fd 70 21      std end
4665 4663 e663 7e e6 87      jmp cc221
4666 4664      cc226:
4667 4665 e666 cc 00 01      ldd #1
4668 4666 e669 fd 70 21      std end
4669 4667 e66c 7e e6 87      jmp cc221
4670 4668 e66f 7e e6 87      jmp cc221
4671 4669      cc222:
4672 4670 e672 bd c0 f9      jsr ccswitch
4673 4671 e675 e6 4e 00 30      FDB cc223,48
4674 4672 e679 e6 57 00 31      FDB cc224,49
4675 4673 e67d e6 5d 00 35      FDB cc225,53
4676 4674 e681 e6 66 00 39      FDB cc226,57
4677 4675 e685 00 00      FDB 0
4678 4676      cc221:
4679 4677 e687 7e e6 08      jmp cc216
4680 4678      cc217:
4681 4679 e68a bd e0 eb      jsr newline
4682 4680 e68d fc 70 13      ldd bcc_errno
4683 4681 e690 10 83 00 00      cmpd #0
4684 4682 e694 10 27 00 0d      lbeq cc227
4685 4683 e698 cc e6 b3      ldd #cc1+0
4686 4684 e69b 34 06      pshs d
4687 4685 e69d bd e0 41      jsr puts
4688 4686 e6a0 32 62      leas 2,s
4689 4687 e6a2 7e e6 af      jmp cc228
4690 4688      cc227:
4691 4689 e6a5 cc e6 c5      ldd #cc1+18
4692 4690 e6a8 34 06      pshs d
4693 4691 e6aa bd e0 41      jsr puts
4694 4692 e6ad 32 62      leas 2,s
4695 4693      cc228:
4696 4694 e6af bd ce 23      jsr key_data
4697 4695 e6b2 39      rts
4698 4696      cc1:
4699 4697 e6b3 63 68 65 63 6b 20      FCB 99,104,101,99,107,32,115,117,109,32
4700          73 75 6d 20
4701 4698 e6bd 65 72 72 6f 72 73      FCB 101,114,114,111,114,115,33,0,48,32
4702          21 00 30 20
4703 4699 e6c7 65 72 72 6f 72 2e      FCB 101,114,114,111,114,46,46,46,0
4704          2e 2e 00
```

```
4705 4700
4706 4701
4707 4702
4708 4703 e6d0 bd e0 eb * .global key_load
4709 4704 e6d3 cc e6 e1 key_load:
4710 4705 e6d6 34 06 jsr newline
4711 4706 e6d8 bd e0 41 ldd #cc229+0
4712 4707 e6db 32 62 pshs d
4713 4708 e6dd bd e5 fc jsr puts
4714 4709 e6e0 39 leas 2,s
4715 4710 jsr get_s_re
4716 4711 e6e1 4c 6f 61 64 20 4d rts
4717 6f 74 6f 72 cc229:
4718 4712 e6eb 6f 6c 61 20 73 2d FCB 76,111,97,100,32,77,111,116,111,114
4719 72 65 63 6f
4720 4713 e6f5 72 64 20 28 73 65 FCB 111,108,97,32,115,45,114,101,99,111
4721 74 20 31 6d
4722 4714 e6ff 73 20 64 65 6c 61 FCB 114,100,32,40,115,101,116,32,49,109
4723 79 2c 20 63
4724 4715 e709 68 61 72 61 63 74 FCB 115,32,100,101,108,97,121,44,32,99
4725 65 72 20 26
4726 4716 e713 20 6c 69 6e 65 29 FCB 104,97,114,97,99,116,101,114,32,38
4727 00
4728 4717
4729 4718
4730 4719 * .global initreg
4731 4720 e71a cc 02 00 initreg:
4732 4721 e71d fd 70 19 ldd #512
4733 4722 e720 cc 02 00 std PC
4734 4723 e723 fd 70 1b ldd #512
4735 4724 e726 cc 7f 00 std save_PC
4736 4725 e729 fd 70 2a ldd #32512
4737 4726 e72c cc 00 00 std USER_U
4738 4727 e72f f7 70 35 ldd #0
4739 4728 e732 1f a8 stb USER_DP
4740 4729 e734 b7 70 34 TFR CC,A
4741 4730 e737 cc 80 00 STA USER_P
4742 4731 e73a fd 70 3e ldd #-32768
4743 4732 e73d cc 80 03 std gpio1
4744 4733 e740 fd 70 40 ldd #-32765
4745 4734 e743 cc 80 02 std port2
4746 4735 e746 fd 70 42 ldd #-32766
4747 4736 e749 cc 80 01 std port1
4748 4737 e74c fd 70 44 ldd #-32767
4749 4738 e74f 39 std port0
4750 4739 rts
4751 4740
4752 4741 * .global wait1s
4753 4742 e750 1c ef wait1s:
4754 4743 andcc #$ef
4755 4744 e752 f6 70 0e cc231:
4756 4745 e755 1d ldb tick
4757 4746 e756 34 06 sex
4758 4747 e758 cc 00 64 pshs d
4759 4748 e75b 10 a3 e1 ldd #100
4760 4749 e75e 2e 05 cmpd ,s++
4761 bgt *+7
```

```

4761 4750 e760 cc 00 00
4762 4751 e763 20 03
4763 4752 e765 cc 00 01
4764 4753 e768 10 83 00 00
4765 4754 e76c 10 27 00 03
4766 4755 e770 7e e7 52
4767 4756
4768 4757 e773 cc 00 00
4769 4758 e776 f7 70 0e
4770 4759 e779 39
4771 4760
4772 4761
4773 4762
4774 4763 e77a 32 7d
4775 4764 e77c 31 e4
4776 4765 e77e 1f 20
4777 4766 e780 34 06
4778 4767 e782 cc a0 00
4779 4768 e785 ed f1
4780 4769 e787 31 e4
4781 4770 e789 1f 20
4782 4771 e78b 34 06
4783 4772 e78d ec f1
4784 4773 e78f 34 06
4785 4774 e791 cc 00 16
4786 4775 e794 e7 f1
4787 4776
4788 4777 e796 31 e4
4789 4778 e798 1f 20
4790 4779 e79a 34 06
4791 4780 e79c ec f1
4792 4781 e79e 34 06
4793 4782 e7a0 e6 f1
4794 4783 e7a2 1d
4795 4784 e7a3 34 06
4796 4785 e7a5 cc 00 01
4797 4786 e7a8 a4 e0
4798 4787 e7aa e4 e0
4799 4788 e7ac 34 06
4800 4789 e7ae cc 00 00
4801 4790 e7b1 10 a3 e1
4802 4791 e7b4 27 05
4803 4792 e7b6 cc 00 00
4804 4793 e7b9 20 03
4805 4794 e7bb cc 00 01
4806 4795 e7be 10 83 00 00
4807 4796 e7c2 10 27 00 03
4808 4797 e7c6 7e e7 96
4809 4798
4810 4799 e7c9 31 e4
4811 4800 e7cb 1f 20
4812 4801 e7cd 34 06
4813 4802 e7cf ec f1
4814 4803 e7d1 34 06
4815 4804 e7d3 cc 00 56
4816 4805 e7d6 e7 f1

        ldd  #0
        bra *+5
        ldd #1
        cmpd #0
        lbeq cc232
        jmp cc231
cc232:
        ldd #0
        stb tick
        rts

* .global getchar2
getchar2:
        leas -3,s
        leay 0,s
        tfr y,d
        pshs d
        ldd #-24576
        std [,s++]
        leay 0,s
        tfr y,d
        pshs d
        ldd [,s++]
        pshs d
        ldd #22
        stb [,s++]

cc233:
        leay 0,s
        tfr y,d
        pshs d
        ldd [,s++]
        pshs d
        ldb [,s++]
        sex
        pshs d
        ldd #1
        anda ,s+
        andb ,s+
        pshs d
        ldd #0
        cmpd ,s++
        beq *+7
        ldd #0
        bra *+5
        ldd #1
        cmpd #0
        lbeq cc234
        jmp cc233

cc234:
        leay 0,s
        tfr y,d
        pshs d
        ldd [,s++]
        pshs d
        ldd #86
        stb [,s++]

```

```
4817 4806 e7d8 31 62          leay 2,s
4818 4807 e7da 1f 20          tfr y,d
4819 4808 e7dc 34 06          pshs d
4820 4809 e7de 31 62          leay 2,s
4821 4810 e7e0 1f 20          tfr y,d
4822 4811 e7e2 34 06          pshs d
4823 4812 e7e4 ec f1          ldd [ ,s++ ]
4824 4813 e7e6 34 06          pshs d
4825 4814 e7e8 cc 00 01          ldd #1
4826 4815 e7eb e3 e1          addd ,s++
4827 4816 e7ed 34 06          pshs d
4828 4817 e7ef e6 f1          ldb [ ,s++ ]
4829 4818 e7f1 1d             sex
4830 4819 e7f2 e7 f1          stb [ ,s++ ]
4831 4820 e7f4 fc 70 3e          ldd gpiol
4832 4821 e7f7 34 06          pshs d
4833 4822 e7f9 31 64          leay 4,s
4834 4823 e7fb 1f 20          tfr y,d
4835 4824 e7fd 34 06          pshs d
4836 4825 e7ff e6 f1          ldb [ ,s++ ]
4837 4826 e801 1d             sex
4838 4827 e802 e7 f1          stb [ ,s++ ]
4839 4828 e804 31 62          leay 2,s
4840 4829 e806 1f 20          tfr y,d
4841 4830 e808 34 06          pshs d
4842 4831 e80a e6 f1          ldb [ ,s++ ]
4843 4832 e80c 1d             sex
4844 4833 e80d 32 63          leas 3,s
4845 4834 e80f 39             rts
4846 4835
4847 4836          * .global main
4848 4837          main:
4849 4838 e810 bd e7 1a          jsr initreg
4850 4839 e813 fc 70 3e          ldd gpiol
4851 4840 e816 34 06          pshs d
4852 4841 e818 cc 00 00          ldd #0
4853 4842 e81b e7 f1          stb [ ,s++ ]
4854 4843 e81d fc 70 40          ldd port2
4855 4844 e820 34 06          pshs d
4856 4845 e822 cc 00 00          ldd #0
4857 4846 e825 e7 f1          stb [ ,s++ ]
4858 4847 e827 fc 70 42          ldd port1
4859 4848 e82a 34 06          pshs d
4860 4849 e82c cc 00 ff          ldd #255
4861 4850 e82f e7 f1          stb [ ,s++ ]
4862 4851 e831 cc 00 00          ldd #0
4863 4852 e834 f7 70 0d          stb flag
4864 4853 e837 bd df 73          jsr initacia
4865 4854 e83a bd e0 eb          jsr newline
4866 4855 e83d cc e9 18          ldd #cc230+0
4867 4856 e840 34 06          pshs d
4868 4857 e842 bd e0 41          jsr puts
4869 4858 e845 32 62          leas 2,s
4870 4859 e847 bd c2 28          jsr InitLcd
4871 4860 e84a bd c2 28          jsr InitLcd
4872 4861 e84d cc e9 35          ldd #cc230+29
```

4873	4862	e850	34	06	pshs d	
4874	4863	e852	bd	c2	6b	jsr PutLCD
4875	4864	e855	32	62	leas 2,s	
4876	4865	e857	cc	00	00	ldd #0
4877	4866	e85a	34	06	pshs d	
4878	4867	e85c	cc	00	01	ldd #1
4879	4868	e85f	34	06	pshs d	
4880	4869	e861	bd	c1	97	jsr goto_xy
4881	4870	e864	32	64	leas 4,s	
4882	4871	e866	cc	e9	49	ldd #cc230+49
4883	4872	e869	34	06	pshs d	
4884	4873	e86b	bd	c2	6b	jsr PutLCD
4885	4874	e86e	32	62	leas 2,s	
4886	4875	e870	cc	70	38	ldd #buffer
4887	4876	e873	34	06	pshs d	
4888	4877	e875	cc	00	05	ldd #5
4889	4878	e878	e3	e1	addd ,s++	
4890	4879	e87a	34	06	pshs d	
4891	4880	e87c	cc	c1	0d	ldd #convert
4892	4881	e87f	34	06	pshs d	
4893	4882	e881	cc	00	06	ldd #6
4894	4883	e884	e3	e1	addd ,s++	
4895	4884	e886	34	06	pshs d	
4896	4885	e888	e6	f1	ldb [,s++]	
4897	4886	e88a	1d		sex	
4898	4887	e88b	e7	f1	stb [,s++]	
4899	4888	e88d	cc	70	38	ldd #buffer
4900	4889	e890	34	06	pshs d	
4901	4890	e892	cc	00	04	ldd #4
4902	4891	e895	e3	e1	addd ,s++	
4903	4892	e897	34	06	pshs d	
4904	4893	e899	cc	c1	0d	ldd #convert
4905	4894	e89c	34	06	pshs d	
4906	4895	e89e	cc	00	08	ldd #8
4907	4896	e8a1	e3	e1	addd ,s++	
4908	4897	e8a3	34	06	pshs d	
4909	4898	e8a5	e6	f1	ldb [,s++]	
4910	4899	e8a7	1d		sex	
4911	4900	e8a8	e7	f1	stb [,s++]	
4912	4901	e8aa	cc	70	38	ldd #buffer
4913	4902	e8ad	34	06	pshs d	
4914	4903	e8af	cc	00	03	ldd #3
4915	4904	e8b2	e3	e1	addd ,s++	
4916	4905	e8b4	34	06	pshs d	
4917	4906	e8b6	cc	c1	0d	ldd #convert
4918	4907	e8b9	34	06	pshs d	
4919	4908	e8bb	cc	00	00	ldd #0
4920	4909	e8be	e3	e1	addd ,s++	
4921	4910	e8c0	34	06	pshs d	
4922	4911	e8c2	e6	f1	ldb [,s++]	
4923	4912	e8c4	1d		sex	
4924	4913	e8c5	e7	f1	stb [,s++]	
4925	4914	e8c7	cc	70	38	ldd #buffer
4926	4915	e8ca	34	06	pshs d	
4927	4916	e8cc	cc	00	02	ldd #2
4928	4917	e8cf	e3	e1	addd ,s++	

```
4929 4918 e8d1 34 06          pshs d
4930 4919 e8d3 cc c1 0d          ldd #convert
4931 4920 e8d6 34 06          pshs d
4932 4921 e8d8 cc 00 09          ldd #9
4933 4922 e8db e3 e1          addd ,s++
4934 4923 e8dd 34 06          pshs d
4935 4924 e8df e6 f1          ldb [ ,s++ ]
4936 4925 e8e1 1d          sex
4937 4926 e8e2 e7 f1          stb [ ,s++ ]
4938 4927 e8e4 cc 70 38          ldd #buffer
4939 4928 e8e7 34 06          pshs d
4940 4929 e8e9 cc 00 01          ldd #1
4941 4930 e8ec e3 e1          addd ,s++
4942 4931 e8ee 34 06          pshs d
4943 4932 e8f0 cc 00 00          ldd #0
4944 4933 e8f3 e7 f1          stb [ ,s++ ]
4945 4934 e8f5 cc 70 38          ldd #buffer
4946 4935 e8f8 34 06          pshs d
4947 4936 e8fa cc 00 00          ldd #0
4948 4937 e8fd e3 e1          addd ,s++
4949 4938 e8ff 34 06          pshs d
4950 4939 e901 cc 00 00          ldd #0
4951 4940 e904 e7 f1          stb [ ,s++ ]
4952 4941          cc235:
4953 4942 e906 cc 00 01          ldd #1
4954 4943 e909 10 83 00 00          cmpd #0
4955 4944 e90d 10 27 00 06          lbeq cc236
4956 4945 e911 bd de f7          jsr scan1
4957 4946 e914 7e e9 06          jmp cc235
4958 4947          cc236:
4959 4948 e917 39          rts
4960 4949          cc230:
4961 4950 e918 36 38 30 39 20 4d          FCB 54,56,48,57,32,77,73,67,82,79
4962          49 43 52 4f
4963 4951 e922 50 52 4f 43 45 53          FCB 80,82,79,67,69,83,83,79,82,32
4964          53 4f 52 20
4965 4952 e92c 4b 49 54 20 32 30          FCB 75,73,84,32,50,48,50,48,0,54
4966          32 30 00 36
4967 4953 e936 38 30 39 20 4d 49          FCB 56,48,57,32,77,73,67,82,79,80
4968          43 52 4f 50
4969 4954 e940 52 4f 43 45 53 53          FCB 82,79,67,69,83,83,79,82,0,51
4970          4f 52 00 33
4971 4955 e94a 32 6b 42 20 52 41          FCB 50,107,66,32,82,65,77,32,85,65
4972          4d 20 55 41
4973 4956 e954 52 54 20 4c 43 44          FCB 82,84,32,76,67,68,0
4974          00
4975 4957          END
4976
```

NOTE