S U N Microsystems

Sun-2 CPU Board Diagnostic

User's Document

Ching-Tai Hu

August 30, 1988

Revision A

The user document of Sun-2 CPU Board Diagnostic is presented.

It is the intention of this document to inform the reader how to use the Sun-2 CPU Board Diagnostic.

Members of any of the following departments may find this document of interest for various reasons: (1) Design Engineering, (2) Manufacturing, (3) Field Service, and (4) Documentation. Using the Sun-2 CPU Board Diagnostic, the Design Engineer(s) of the Sun-2 CPU Board will be able to confirm the correctness of their design. Manufacturing and Field Service personnel will use Sun-2 CPU Board Diagnostic for testing and/or troubleshooting purposes. Finally, the Documentation department will use this document as a basis for developing the User's Guide which will be shipped with the product itself.

Revision A August 30, 1988 Initial release of this document.

VMEbus	Motorola bus interface connecting CPU board with other peripherals
Multibus	Intel bus interface connecting CPU board with other peripherals
DVMA	Direct Virtual Memory Access
SCSI	Small Computer System Interface
Async	Asynchronous transmission protocol
Bisync	IBM Binary synchronous transmission protocol
SDLC	IBM Synchronous Data Link Control protocol
SCC	Serial Communication Controller by Zilog
RTS	Request To Send
CTS	Clear To Send
DCD	Data Carrier Detect
DTR	Data Terminal Ready
TXD	Transmitted Data
RXD	Received Data
INT	Interrupt
CRC	Cyclic Redundancy Check

This document is meant to help you understand how the Sun-2 CPU Board Diagnostic works. It starts with a general description of hardware/software requirements, and moves on to a description of the user interface which includes error messages. Following is a description of the suggested testing sequence which will result in an accurate test.

The cpu2.diag has the ability to support both model-120/170 and model-50/160 workstations in the SUN-2 family. The CPU board is composed of several small units, each unit handles a certain function. The cpu2.diag is a package which concentrates on verifying these functions such as, MMU unit, Timer unit, DVMA unit, Clock unit, and Serial Communication Controller unit.

The minimum configuration of hardware required to run the Sun-2 CPU Board Diagnostic is:

(A). Sun-2/120 workstation:

1). Multibus card cage with power supply.
2). SUN-2/120 Processor board (the unit under tested).
3). At least one golden Multibus Memory board (one Mega bytes).
4). Sun video console (a golden multibus video controller board needed) or televideo-like terminal.
5). A golden SCSI host adapter with hard disk(Micropolis 1304).
6). Boot device, local disk (disk controller needed), local tape (tape controller needed), or remote disk via ethernet (ethernet controller board needed).

(B). Sun-2/50 workstation:

1). VMEbus card cage with power supply.
2). SUN-2/50 Processor board (the unit under tested).
3). Sun video console or televideo-like terminal.
4). A golden SCSI host adapter with an expansion board and hard disk(Micropolis 1304), if the interface of the VME DVMA is needed to be checked.

The standard standalone CPU Board Diagnostic (cpu.diag) is needed.

The standard firmware (rom monitor) is needed to boot the cpu2.diag program.

This is a standalone Sun-2 CPU Board Diagnostic package. The testing program is configured by itself between the model-120/170 and the model-50/160 of the SUN-2 family. The program prints the different header messages for the different models on the screen. After loading the cpu2.diag, the program does map those devices which are under tested by itself in case the prom monitor didn't do a good job.

The physical addresses for the on-board devices are:

SUN-2 120/170 CPU Board	
Device	Physical Address
SCC chip	0x002000
Timer chip	0x002800
Clock chip	0x003800
Video control register	0x781800
Encryption Processor	0x001000

SUN-2 50/160 CPU Board	
Device	Physical Address
SCC chip	0x7f2000
Timer chip	0x7f2800
Video control register	0x020000
Encryption Processor	0x7f1000

The CPU board supports seven autovector levels for the on-board devices to generate the interrupt signals. The jumper of each device should be installed before running the testing program, otherwise, the error message of time out will be generated. The following table shows the physical interrupt sources on the CPU board.

CPU Board Interrupt Level Configuration	
level	Source
level-1	system EN.INT1 (bit-2 of system enable register)
level-2	system EN.INT2 (bit-3 of system enable register)
level-3	system EN.INT3 (bit-4 of system enable register)
level-4	video interrupt
level-5	timer-2,3,4,5
level-6	SCC chip
level-7	timer-1

The layout of the external loopback cable for the Model-120/170 Serial Communication Controller is shown in the table as below. The leftest four columns are the layout of the 50 pins connector on the CPU board. The middle column shows the 50 lines parallel cable. The rightest two columns are the layout of the 25 pins female connectors of RS232.

SCC External Cable Configuration							
J.60 J1				CABLE	RS232 FEMALE CONNECTORS		
	1	2			1	14	
TXDA[]	3	4	DBA[]		2	15	
RXDA	5	6			3	16	
RTSA	7	8	DDA		4	17	
CTSA	9	10			5	18	
DSRA	11	12			6	19	CHANNEL-A
GND	13	14	DTRA		7	20	
DCDA	15	16			8	21	
	17	18			9	22	
	19	20			10	23	
	21	22	DAA		11	24	
	23	24	-5VR		12	25	
	25				13		
		26				1	
	27	28	TXDB[]		14	2	
DBB	29	30	RXDB		15	3	
	31	32	RTSB		16	4	
	33	34	CTSB		17	5	
	35	36	DSRB		18	6	CHANNEL-B
	37	38	GND		19	7	
DTRB	39	40	DCDB		20	8	
	41	42			21	9	
	43	44			22	10	
	45	46			23	11	
DAB	47	48			24	12	
-5VR	49	50			25	13	

The following table contains the format of the RS232 external loopback cable. The physical pin number and the transmission directions are shown as well.

RS232 External Loopback Cable				
CHANNEL-A MALE CONNECTOR			CHANNEL-B MALE CONNECTOR	
NAME	PIN	DIRECTION	PIN	NAME
	1		1	
TXDA	2	------->	3	RXDB
RXDA	3	<-------	2	TXDA
RTSA	4	------->	5	CTSB
CTSA	5	<-------	4	RTSB
	6		6	
	7		7	
DCDA	8	<-------	20	DTRB
	9		9	
	10		10	
	11		11	
	12		12	
	13		13	
	14		14	
	15		15	
	16		16	
	17		17	
	18		18	
	19		19	
DTRA	20	------->	8	DCDB
	21		21	
	22		22	
	23		23	
	24		24	
	25		25	

A portion of the I/O routines provided by the standalone libraries are linked to the CPU Board Diagnostic. It is a sequencing, menu driven, and interruptible package and has parameterization capability. It is a useful tool for both engineers and technicians to troubleshoot and evaluate the operation of Sun-2 CPU board.

The testing program is written by the 'C' language, it is a portable package to other CPU boards which use the same devices except the physical addresses have to be changed.

When you turn on the system, after power-on rom diagnostics are run, the system automatically begins booting Unix. Then, break out of the boot sequence and return to the rom monitor by typing L1-a (hold down the "L-1" key while typing "a" key) on the Sun-2 console or by typing <break> key on the televideo-type terminal.

At this point, type K1 to the rom monitor to reset the memory maps to the initial state. We are now ready to boot the CPU Board Diagnostic.

There are several ways of booting

1). From local disk

Assuming the diagnostic 'cpu.diag' exists on your local disk in the /pub/stand (fileserver) or the /stand (standalone) directory, the CPU diagnostic is loaded by typing

```
> b stand/cpu.diag
```

2). From remote disk

Assuming the network fileserver has a partition reserved for the system under test (legitimate client) and the CPU diagnostic exists in the /pub/stand on the fileserver, the CPU diagnostic is loading by typing

```
> b stand/cpu.diag
```

If the system under test is not a client of the fileserver where the diagnostic lives in the /pub/stand directory, the CPU diagnostic is loading by typing the command (for the 3-COM ethernet controller) as below.

```
> b ec(,fileserver_host_net_#)stand/cpu.diag
```

or you can type the command (for the SUN-2 ethernet controller) as below.

```
> b ie(,fileserver_host_net_#)stand/cpu.diag
```

For Example, [b ec(,1a)stand/cpu.diag] indicates the fileserver is venus. 1). Separator (;)

The semicolon (;) mark is used between commands at the command line prompt. It must be isolated on the line and surrounded by spaces. It allows tests to be flexibly sequenced.

2). Default (.)

The period (.) mark is used as a place holder to indicate default values for some parameters.

3). Forever (*)

The star (*) mark is approximately equal to infinity or 0xffffffff times for loop count parameters.

4). Null ( )

If a parameter is not supplied, then, the default values are used.

The syntax of the command line and the parameters of the different menus will be discussed in the following sections.

You can specify several commands separated by a separator mark (;) on a single command line. This is equivalent to the traditional batch mode operation. The CPU diagnostic program will fetch each command and sequentially execute them until the end mark is reached. This feature gives users flexibility.

The cpu2.diag is a menu-driven program. The user could face up to three level menus. The first level menu is the main menu. It controls the procession of the CPU diagnostic. There are some functional tests which don't support the second and third level menus. The user has to specify command(s) and parameter(s) on the main menu level such as the diagnostic register test, the clock test, the MMU test, and the CPU interrupt test. The timer test, DVMA test and the SCC test have second level menu. The command(s) and parameter(s) can be specified in this level.

The SCC test supports the third level menu for changing the configuration of the Serial Communication Controller. Once a certain configuration has been established, it is unnecessary to go to the third menu.

In order to help the user to distinguish the current level of menu from other menus, each menu supports a different prompt sign, such as, the "CPU COMMAND :" for the main menu, the "SCC COMMAND :" for the second level menu of SCC test, the "SCC CONF COMMAND :" for the third level menu of SCC test, the "DVMA COMMAND :" for the second level menu of DVMA test, and the "TIMER COMMAND :" for the second level menu of Timer functional test.

If you specify the option of the automatic continuation on the main menu level, when error(s) occurred, the processing of the test will be continued beside displaying the error message. Otherwise, the users will be asked to abort or continue the procession of the test every time when error(s) occurred.

The cpu2.diag reserves several keys for the special functions, such as, 'h', program running level.

From any level of menu, you can abort the cpu2.diag by giving the 'q' command and the control of the system will be passed to the prom monitor. When you types the 'h' command, the help menu of the current level is displayed on the screen. If you specify the 'D' command, The current menu will be printed on the screen. After typing the 'u' command, the control of the system is passed to one of the upper level menu.

If you type the '^s' keys when the testing program is executed, the program will be suspended until the '^Q' keys are specified. If you type '^C' keys during the processing of the test, the current testing program will be aborted, and the control of the system is passed to the menu level.

After loading the CPU diagnostic, the control of the system is passed from the rom monitor to the CPU diagnostic. First, it prints main menu messages on the screen as follows. It waits the input of the command(s) and then executes them.

(1). Main menu of the model-120/170 workstation:

```
SUN2-120/170 CPU Board Diagnostic REV 1.1 9/25/86 Main Menu

        d - diagnostic register test
        c - real-time-clock test
        e - encryption test
        s - serial communication controller test
        t - timer test
        m - MMU check
        v - DVMA interface check
        i - CPU interrupt test
        T - default test
        a - auto continuation
        l - loop
        h - help
        q - back to monitor

CPU COMMAND :
```

(2). Main menu of the model-50/160 workstation:

```
SUN2-50/160 CPU Board Diagnostic  REV 1.1 9/25/86  Main Menu

        d - diagnostic register test
        c - real-time-clock test
        e - encryption test
        s - serial communication controller test
        t - timer test
        m - MMU check
        v - DVMA interface check
        i - CPU interrupt test
        T - default test
        a - auto continuation
        l - loop
        h - help
        q - back to monitor

CPU COMMAND :
```

The prompt sign of the main menu is shown by the "CPU Command :". After the sign is prompted, the user can specify a command and parameter(s) separated by spaces. If the selected command is not supported by the CPU diagnostic, the error message

(nonexist command) is an illegal option.

is printed. (nonexist command) is whatever you have typed. Also, the main menu is displayed again.

From now on, in the remaining sections of this document, the cpu2.diag will only show the header messages for the model-120/170. It is the same with the model 50/160 except the header messages. The commands and parameters of each subtest will be discussed in detail in the section of "The Contents Of Main Menu".

After typing the 'h' command, the following messages will help you understanding the syntax of commands and parameters of the main menu.

d - diag register test	d passcnt [0|1]
c - clock test	c passcnt
e - encryption test	e key[8] data[8] passcnt
s - serial port test	s
t - timer test	t
m - MMU check	m passcnt
v - DVMA interface check	v
i - CPU interrupt test	i passcnt level
T- default test	T
a - auto continuation	a [0|1]
l - loop	l loopcnt
h - help	h
q - back to monitor	q

There are several functional tests which have the submenu, such as, the SCC test, the Timer test, and the DVMA test. When one of these tests is selected, the submenu of the test is displayed on the screen.

The following sections describe the syntax of the commands and the parameters of each subtest which is supported by the CPU Diagnostic. A brief description of each subtest is given as well. Command syntax:

d passcnt mode

passcnt - loop count, 1 - 0xFFFFFFFF, the default value is '1'.
mode - auto or manual operation, 0 - 1, the default is auto mode.
'0' - automatic mode.
'1' - manual mode.

After selecting the 'd' command, the control of the system is passed to the diagnostic register subtest. It first prints the following message on the screen:

SUN-2 mode-120/170 CPU Diagnostic Register Test

The test checks the register of the diagnostic LEDs on the CPU board. Since the register is a read-only device, the result of the test should be verified by the user. The program supports two operational modes, automatic and manual.

The automatic mode is selected by typing '0' after the parameter of pass count. One of the LEDs will be lighted for a while, it then automatically shifts to the next LED until all of eight LEDs have been exercised.

The operation of the manual mode is selected by typing the '1' in the mode field. After the first LED is lighted, the following message is displayed on the screen to request the input of key-stroke:

Please hit any key to continue.

After the program recognizes the responsed action (it could be any key for this version), the next LED will be lighted. Otherwise, the previous selected LED will be lighted forever. The above sequence will be repeated until all of the LEDs been exercised.

Command syntax:

c passcnt tc

passcnt - loop count, 1 - 0xFFFFFFFF, the default is 1.
tc - the number of elapsed time (time constant = 1/17 second), the default is 1.

The clock test supports the functional check of the system real-time-clock. There is no such device on the CPU board of the model-50/160, therefore, no clock test supported for the model-50/160. If the user specifies the 'c' command. The error message

no such device such device on SUN-2 model-50/160 CPU Board.

will be displayed on the screen.

After typing the 'c' command, the control of the system is passed to the clock test. It first prints the header message as follows:

SUN-2 model-120/170 CPU Clock Test

It then displays the testing result on the screen.

clock (time(ref) = 0x39aa time(used) = 399a deviation= -0.3266

the time(ref) is the reference time in usec. the time(used) is the actual time used in usec.

Before running the test, the user should install the jumper for the autovector interrupt of the level-4 which is the source of the reference time. Otherwise, the error message of time out will be displayed on the screen.

The reference source comes from the vertical retrace interrupt signal of the video controller (the autovector interrupt level of video is level-4). For SUN-2 family, it has two different kind of monitors, one displays 1152X900 pixels on the screen , another displays 1024X1024 pixels on the screen. the elapsed time is different between these monitors.

elapsed time (1152X900) = 14992 usec
elapsed time (1024X1024) = 16976 usec

The program is intelligent to configure by itself which monitor is used, and the correct reference time will be given.

Command syntax:

e key[8] data[8] passcnt

key[8] -
data[8] -
passcnt - loop count, 1 - 0xFFFFFFFF, the default is 1.

Command syntax:

s

After typing the 's' command, the control of the tested system is passed to the CPU-SCC program. It first prints the CPU-SCC submenu on the screen. It then waits the command(s). The section of "Serial Communication Controller Submenu" will describe the details. Command syntax:

t

When the user types the 't' command, the control of the tested system is passed to the CPU-Timer testing program. It first prints the CPU-Timer submenu on the screen and then waits the input of the command(s). The details will be described in the section of the "Timer Functional Test Submenu". Command syntax:

m passcnt

passcnt -loop count, 1 - 0xFFFFFFFF, the default is 1.

Command syntax:

v

When the user specifies the 'v' command, the control of the tested system is passed to the CPU-DVMA testing program. It first prints the submenu of the CPU-DVMA test on the screen. It then waits the input of the command(s). The section of the "DVMA Functional Test Submenu" will describe the details.

Command syntax:

i passcnt level

passcnt - loop count, (1 - 0xFFFFFFFF), the default is 1.
level - autovector interrupt level index, (0 - 6), the default is 0.
'0' - level-1 to level-6.
'1' - level-1.
'2' - level-2.
'3' - level-3.
'4' - level-4.
'5' - level-5.
'6' - level-6.

After typing the 'i' command, The control of the system is passed to the CPU interrupt test. If the user specifies the autovector interrupt level index to be '0', it means to check all of the interrupt levels except level-0 and level-7. The following shows the different interrupt source for each interrupt level.

autovector level '1' -- system interrupt level-1 enable.
autovector level '2' -- system interrupt level-2 enable.
autovector level '3' -- system interrupt level-3 enable.
autovector level '4' -- video interrupt.
autovector level '5' -- system timer interrupt.
autovector level '6' -- On-board SCC interrupt.

The CPU-Interrupt test doesn't check the interrupt signal of level-7. If the level-7 interrupt doesn't work fine, the program won't reach to this point. This is a logical and reasonable assumption!

Before testing the interrupt signals, all of the jumpers related with that interrupt source should be installed, otherwise, the error message of time out of such interrupt will be displayed on the screen.

The testing program first prints the header message on the screen as follows:

SUN-2 model-120/170 CPU Interrupt Signals Test

If the testing program could recognize the interrupt signal(s), the run-time message will be displayed on the same line as follows:

pass 1 of CPU interrupt test : errors= 0, total errors= 0

Command syntax:

T

When the SCC subtest is selected, the control of the tested system is passed to this submenu. It first prints the submenu as below and then waits the input of the command(s). The following is the submenu for the Serial Communication Controller unit:

```
SUN-2 CPU Board Diagnostic REV Serial Communication Controller Submenu

                    I - interrupt signal test
                    A - async loopback test
                    B - bisync loopback test
                    S - SDLC loopback test
                    M - modem signals test
                    N - sync test
                    T - default test
                    d - dump
                    f - change configuration
                    D - display menu
                    u - up one level menu
                    l - loop
                    h - help menu
                    q - back to monitor

SCC COMMAND :
```

The prompt sign of the CPU-SCC submenu is shown by the "SCC Command :". If the selected command is not supported by the CPU-SCC diagnostic, the same error message as the main menu will be displayed on the screen.

The 'D', 'u', 'h', and 'q' commands have the same meaning as the main menu.

When the SCC test is selected, the initiated state of the configuration of the SCC is shown as below:

m = 0,	poll mode.
L = 0,	local loopback mode.
b = 4800,	baud rate.
p = 1,	No parity.
c = 3,	8-bit character length.
s = 2,	1.5-bit stop bit.
x = 0x32,	sync character.
P = 0 0xAA,	constant with 0xAA pattern
z = 1024 bytes,	buffer size.
C = 1,	Channel-A.
T = 0xdb,	pad pattern.
k = 16,	X16 system clock.

If the user wants to check the SCC chip with the configuration other than the initiated state, she/he should change the configuration by typing the appropriated command(s) under the CPU-SCC CONF submenu. The sub-section of "CPU-SCC Configuration Submenu" will discuss more details about the commands to change the configuration.

The following sub-sections will discuss the HELP menu and all of the commands supported by the SCC submenu.

The following messages are used to help the user to understand the syntax of the different command(s) and parameter(s) of the SCC subtest.

I - interrupt signal test	I passcnt [0|1|2]
A - async loopback test	A passcnt
B - bisync loopback test	B passcnt
S - SDLC loopback test	S passcnt
M - modem signals test	M passcnt
N - sync test	N passcnt
T - default test	T
d - dump	d [0|1|2|4|5|6|7|8]
f - change configuration	f
D - display menu	D
u - up one level menu	u
l - loop	l loopcount
h - help menu	h
q - back to monitor	q

Command syntax:

I passcnt srcindex

passcnt - loop count, (1 - 0xFFFFFFFF), the default is 1.
srcindex - interrupt source index, (0- 2), the default is '0'.
0 - the source comes from either baud rate generator or modem signals.
1 - the source comes from baud rate generator.
2 - the source comes from modem signals.

After typing the 'I' command, the control of the tested system is passed to the SCC interrupt test program. If the user specifies the '1' in the srcindex field, when the zero count is reached, the ability of the SCC to generate the interrupt signals is checked.

If the user specifies the '0' in the srcindex field, when the state of the modem signals is changed, the ability of the SCC to generate the interrupt is checked. In order to change the state of the modem signals, the user should install the external loopback cable between channel-A and channel-B before running the test.

The testing program first prints the following message:

Install external loopback cable between SCC channels A & B
Type any key to continue

After recognizing any input of key-stroke, the interrupt capability of the SCC chip is checked. After one pass of the test, it displays the following message on the screen.

Pass 1 of SCC-Interrupt Test : errors= 0, total errors= 0

Command syntax:

A passcnt

passcnt - loop count, 1 - 0xFFFFFFFF, the default is 1.

When the user selects the Async loopback test, the control of the tested system is passed to this program. The SCC-Async protocol test checks the ability to transmit and receive the data under the control of the Asynchronous protocol. The user can change the mode of the test (such as, character length, parity type, stop bit length, baud rate, and clock mode) by changing the configurations. The sub-section of the "The CPU-SCC Configuration Submenu" will describe the details.

The following paragraphs are the summary of the abilities of the Asynchronous loopback test.

1. The internal logics of the Serial Communication Controller to transmit and receive the data can be checked by the Asynchronous loopback test with the poll mode enable. It could be running with either the local or external loopback and should be configured to be the half duplex operation. It is an useful troubleshooting tool to check the transmission capability when error occurred which related to the SCC chip.
2. The Asynchronous loopback test with the interrupt mode can check the interrupt logics beside the transmission abilities.
3. The capability of the transmission with different length character can be checked by the Asynchronous loopback test with the different length character. The selection can be done by specifying the change character length command - 'c 0' under the SCC-CONF submenu.
4. The parity check logics of the SCC chip can be checked by the Asynchronous loopback test with the different parity types. This option can be specified by the change parity type command 'p 0' under the SCC-CONF submenu.
5. The capability of the transmission with the different stop bit(s) can be checked by the Asynchronous loopback test with the different stop bit(s). This option can be specified by changing the stop bit command 's 0' under the SCC-CONF submenu.
6. The capability of the transmission with the different clock mode can be checked by the Asynchronous loopback test with the different clock mode. This option can be changed by the change clock command 'k yy' under the SCC-CONF submenu. The 'yy' is the clock mode. If the clock mode is specified to be X64, the baud rate should not exceed 19200. If the clock mode is specified to be X32, the baud rate should not exceed 38400.

After one pass of the test, the following message will be shown on the screen at the same line:

pass 1 of SCC-Async Loopback Test : errors= 0, total errors= 0

Command syntax:

B passcnt

passcnt - loop count, 1 - 0xFFFFFFFF, the default is 1.

After typing the 'B' command, the control of the system is passed to the SCC Bisync Loopback Test. It checks the transmission capability of the IBM Bisync protocol.

It could be running with either the local or external loopback. The character length won't be changed, it should be fixed 8-bit wide character. The interrupt mode should be enable. The size of the transmission blocks should be within the limit, 1024 bytes block is good enough. The longer is the transmission block, the higher probability to get the CRC errors.

Both the trailing pad pattern and the sync character can be changed as well.

When the testing program was running on our lab. machine in our department, the highest baud rate with the full duplex operation is 19200. If you specify the baud rate higher than the limitation of the system, the testing result will be meaningless.

After running one pass of the test, the following message will be shown on the screen at the same line:

pass 1 of SCC-Bisync Loopback Test : errors= 0, total errors= 0

Command syntax:

S passcnt

passcnt - loop count, 1 - 0xFFFFFFFF, the default is 1.

When the user selects the SCC-SDLC loopback test, the control of the system is passed to this test. It checks the transmission capability of the IBM SDLC protocol.

It could be running with either the local or external loopback. The character length won't be changed, it should be fixed 8-bit length character. The interrupt mode should be enable. The size of the transmission blocks should be within the limit, the maximum allowable error-freed block size should be less than 4048 bytes, but, the 1024 bytes block is good enough for testing.

The address of the secondary station is fixed, 0x96.

When the testing program was running on our lab. machine in our department, the highest baud rate with the full duplex operation is 19200. If you specify the baud rate higher than the limitation of the system, the testing result will be meaningless.

After one pass of the test, the following message will be shown on the screen at the same line:

pass 1 of SCC-SDLC Loopback Test : errors= 0, total errors= 0

Command syntax:

M passcnt

passcnt - loop count, 1 - 0xFFFFFFFF, the default is 1.

After typing the 'M' command, the control of the system is passed to the SCC-Modem test. It checks the ability to change the state of modem signals through the external loopback cable.

After one pass of the test, the following message is shown on the screen at the same line:

pass 1 of SCC-Modem Test : errors= 0, total errors= 0

Command syntax:

N passcnt

passcnt - loop count, 1 - 0xFFFFFFFF, the default is 1.

After typing the 'N' command, the control of the system is passed to the SCC-Sync character test. The test verifies the ability to transmit and receive the data with the different sync characters under the control of the IBM Bisync protocol.

It could be running with either the local or external loopback. The character length won't be changed, it should be fixed 8-bit length character. The interrupt mode should be enable. The size of the transmission blocks should be within the limit, the 1024 bytes block is good enough for testing.

Before running the test, you could change sync character by the change sync character command 'x 0xhh' under the SCC-CONF submenu. If you specify 0x00, the sync character sequence (from 0x01 to 0xFF) will be verified instead of 0x00.

When the testing program was running on our lab. machine in our department, the highest baud rate with the full duplex operation is 19200. If you specify the baud rate higher than the limitation of the system, the testing result will be meaningless.

After one pass of the test, the following message will be shown on the screen at the same line:

pass 1 of SCC-Sync Test : errors= 0, total errors= 0

In order to check the functionalities and the different combination of the configurations of the SCC chip, this package supports a submenu to change the configuration of the SCC chip very easily. The following table shows the CPU-SCC Configuration submenu:

SUN-2 model-120/170 CPU Board Diagnostic REV ?? SCC Configuration Submenu

m - poll/interrupt mode
L - local/external loopback
b - baud rate selection
p - parity type selection
c - character length
s - stop bit selection
x - sync character
P - pattern selection
z - buffer size selection
D - display menu
C - channels selection
T - trailing pad pattern
f - conf information
k - clock rate selection
u - up one level menu
h - help menu
q - back to monitor

SCC CONF COMMAND :

The prompt sign of the CPU-SCC CONF submenu is shown by the "SCC CONF Command :". If the selected command is not supported by the CPU-SCC CONF submenu, the same error message as the main menu will be shown on the screen.

The following table helps you to understand the syntax of the command(s) and the parameter(s) of the SCC Configuration submenu.

m - poll	interrupt mode
L - local	external loopback
b - baud rate selection	b baudrate
p - parity type selection	p [0|1|2|3]
c - character length	c charlen
s - stop bit selection	s 0|1|2|3]
x - sync character	x syncchar
P - pattern selection	P pattern_type pattern
z - buffer size selection	z buffer_size
D - display menu	D
C - channels selection	C [0|1|2|3|4]
T - trailing pad pattern	T pattern
f - conf information	f
k - clock rate selection	k [1|16|32|64]
u - up one level menu	u
h - help menu	h
q - back to monitor	q
SCC CONF Command :	

The 'D', 'u', 'h', and 'q' commands have the same meaning as the main menu. Command syntax:

m mode

mode - operation mode, (0 - 1), the default is '0'.
'0' - poll mode.
'1' - interrupt mode.

The 'm 1' command will swap the operation from the poll mode to the interrupt mode. The 'm 0' command selects the poll mode operation. Command syntax:

L mode

mode - loopback mode, (0 - 1), the default is '0'.
'0' - local loopback mode.
'1' - external loopback mode.

The 'L 1' command will swap the operation from the local loopback mode to the external loopback mode. The 'L 0' command selects the local loopback mode operation. Command syntax:

b baudrate

baudrate - baud rate, the default is '0'.

This command changes the baud rate. The baud rate could be any number, but, it is better to follow the industrial standard specification, such as,

(50, 75, 110, 150, 300, 600, 1200, 1800, 2000, 2400, 4800, 9600, 19200, 38400, 76800)

The 'b 0' command means to use the default baud rate sequence:

150, 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 76800

Command syntax:

p parity

parity - parity type, (0 - 3), the default is '0'.
'0' - parity sequence, (No parity, EVEN, ODD).
'1' - No parity, the parity check logic is not enable.
'2' - EVEN parity, the parity check logic is enable.
'3' - ODD parity, the parity check logic is enable.

The parity type can be changed by this command. The cpu2.diag supports NO, EVEN, or ODD parity type for the asynchronous protocol operation and NO parity type for the synchronous protocol operation. The 'p 0' means to use the default parity type sequence (NO, EVEN, ODD).

Examples:

p - It has the same meaning as the 'p 0'.
p 0 - Set the parity type to be sequence (NO, EVEN, ODD).
p 1 - Set the parity type to be NO.
p 2 - Set the parity type to be EVEN.
p 3 - Set the parity type to be ODD.

Command syntax:

c charlen

charlen - character length, (0 - 3), the default is '0'.
'0' - character length sequence (6, 7, 8 bits).
'1' - 6-bit length character.
'2' - 7-bit length character.
'3' - 8-bit length character.

The length of the transmission character can be changed by this command. The cpu2.diag has the option to use 6, 7, or 8-bit character length for the asynchronous protocol operation, but, the 8-bit character length for the synchronous protocol operation. The 'c 0' means to use the default character length sequence (6, 7, 8).

Examples:

c - It has the same meaning as the 'c 0'.
c 0 - Set the character length to be sequence (6, 7, 8)
c 1 - Set the character length to be 6-bit.
c 2 - Set the character length to be 7-bit.
c 3 - Set the character length to be 8-bit.

Command syntax:

s stopbit

stopbit - stop bit, (0 - 3), the default is '0'.
'0' - stop bit sequence (1, 1.5, and 2 bit(s)).
'1' - select 1 stop bit.
'2' - select 1.5 stop bits.
'3' - select 2 stop bits.

This command only affects the asynchronous protocol operation, it is ignored by all of the synchronous protocols operation. The cpu2.diag supports 1, 1.5, or 2 stop bit(s). The 's 0' means to use the default stop bit sequence (1, 1.5, 2).

Examples:

s - It has the same meaning as the 's 0'.
s 0 - Set the stop bit to be sequence (1, 1.5, 2).
s 1 - Set the stop bit to be 1 stop bit.
s 2 - Set the stop bit to be 1.5 stop bit.
s 3 - Set the stop bit to be 2 stop bit.

Command syntax:

x syncchar

syncchar - sync character, (0 - 0xFF), the default is 0x32.

This command only affects the IBM Bisync protocol operation, it is ignored by the asynchronous and IBM SDLC protocol operations. The 'x 0' means to use the default sync character sequence (0x01 - 0xFF).
Command syntax:

P pattern_type pattern

pattern_type - type of pattern, (0 - 3), the default is '0'.
'0' - constant pattern.
'1' - incremented with the initial pattern of second parameter.
'2' - decremented with the initial pattern of second parameter.
'3' - random pattern.

pattern - testing pattern, (0 - 0xFF), the default is 0xAA.

This command changes the testing pattern(s).

Examples:

P - It has the same meaning as the 'P 0 AA'.
P 0 AA - Set the pattern to be constant 0xAA.
P 1 BB - Set the pattern to be incremented with the initial value 0xBB.
P 2 22 - Set the pattern to be decremented with the initial value 0x22.
P 3 5A - Set the pattern to be random with the initial value 0x5A.

Command syntax:

z buffersize

buffersize - the size of transmitting and receiving buffers, the default is 1024 bytes.

This command changes the size of the testing buffers. Command syntax:

C channel

channel - select tested channel(s), (0 - 4), the default is '0'.
'0' - CH-A -> CH-A and CH-B -> CH-B, full duplex mode with local loopback.
'1' - CH-A -> CH-A, half duplex mode with local loopback.
'2' - CH-B -> CH-B, half duplex mode with local loopback.
'3' - CH-A -> Ch-B, half duplex mode with external loopback.
'4' - CH-B -> CH-A, half duplex mode with external loopback.

This command changes the direction of the operation. Command syntax.

T pattern

pattern - pad pattern, (0 - 0xFF), the default is 0xdb.

In order to get the correct CRCs from the SCC chip for the IBM Bisync protocol test, we have to send some extra trailing pad patterns. This command changes the pad pattern.

Command syntax:

k mode

mode - clock mode, (1|16|32|64), the default is '16'.
'1' - X1 system clock.
'16' - X16 system clock.
'32' - X32 system clock.
'64' - X64 system clock.

For the asynchronous protocol operation, the user could select one of X16 (default), X32, or X64 system clock. For the synchronous protocol operation, the X1 system clock is only choice.

Examples:

k - It has the same meaning as the 'k 16'.
k 1 - Set the clock mode to be X1 system clock.
k 16 - Set the clock mode to be X16 system clock.
k 32 - Set the clock mode to be X32 system clock.
k 64 - Set the clock mode to be X64 system clock.

CPU SCC current configuration

operation:	ASYNCHRONOUS protocol with LOCAL loopback
mode:	POLL mode
direction:	txc(CHA)->rxc(CHA), half duplex mode
baud rate:	19200
character length:	8
parity type:	EVEN
stop bit:	1.5-bit
buffer size:	1024 testing pattern: constant with 0xff
sync pattern:	0x32 pad pattern: 0xdb
cable setup flag:	ON

The above table shows the current configuration of the SCC chip. It is running under the Asynchronous protocol with internal (local) loopback. The interrupt logics are not enable, it is the poll mode. The transmission direction is from the transmitter of the channel-A to the receiver of the same channel with the half duplex mode. The current baud rate is 19200. The length of each character is 8-bit wide. The parity logics have been enable, the parity type is EVEN parity. The stop bit is 1.5-bit long. The current transmitting block is 1024 bytes with the constant data of 0xFF. The others don't mean anything for the Asynchronous protocol.

When the DVMA subtest is selected, the control of the tested system is passed to this submenu. It first prints the submenu as below. It then waits the input of the command(s). The following is the submenu for the DVMA functional test:

```
SUN2-120/170 CPU Board Diagnostic REV DVMA Submenu

                S - SCSI DVMA test
                p - pattern selection
                d - dump buffer
                D - display menu
                u - up one level menu
                l - loop
                h - help menu
                q - back to monitor

DVMA COMMAND :
```

The prompt sign of the CPU-DVMA submenu is shown by the "DVMA Command :". If the selected command is not supported by the CPU-DVMA diagnostic, the same error message as the main menu will be displayed on the screen.

The 'D', 'u', 'h', and 'q' commands have the same meaning as the main menu.

After typing the 'h' command, the following messages are used to help you understanding the syntax of command(s) and parameter(s) of the CPU-DVMA menu.

S - SCSI DVMA test	S passcnt block# W
p - pattern selection	p [0|1|2|3] pattern
d - dump buffer	d [0|1|2] size
D - display menu	D
u - up one level menu	u
l - loop	l loopcount
h - help menu	h
q - back to monitor	q

Command syntax:

S passcnt block# W/rflag

passcnt - loop count, 1 - 0xFFFFFFFF, the default is 1.
block# - The tested block number, the default is 0.
W/rflag - The flag of Write/Read operation, 0 - 2, the default is 0.
'0' - Write, read and verify the data.
'1' - Write operation only.
'2' - Read operation only.

The default value of the block# field is zero. The testing program checks the interface twice with two extremely different block sizes. It first checks the interface with one block (512 bytes) and then checks the interface with 128 blocks (64K bytes) which is the maximum length of the DMA transmission.

Both write and read operations access the data on the "swap" partition instead of the "user" partition, in terms the user data won't be destroyed. The devices used are the Micropolis 1304 drives.

After one pass of the test, the following message will be shown on the screen at the same line:

Pass 1 of SCSI DVMA Test : errors= 0, total errors= 0

When the Timer subtest is selected, the control of the tested system is passed to this submenu. It first prints the submenu as below. It then waits the input of the command(s). The following is the submenu for the Timer functional test:

```
SUN2-120/170 CPU Board Diagnostic REV ?? AMD9513 Timer Submenu

                    I - Timer Interrupt test
                    A - Timer Accuracy test
                    T - Timer Tick test
                    D - display menu
                    u - up one level menu
                    l - loop
                    h - help menu
                    q - back to monitor

TIMER COMMAND :
```

The prompt sign of the CPU-Timer submenu is shown by the "TIMER Command :". If the selected command is not supported by the CPU-Timer diagnostic, the same error message as the main menu will be displayed on the screen.

The 'D', 'u', 'h', and 'q' commands have the same meaning as the main menu.

After typing the 'h' command, the following messages are used to help you understanding the syntax of commands and parameters of the CPU-Timer submenu.

I - Timer Interrupt test	I passcnt index
A - Timer Accuracy test	A passcnt index tc
T - Timer Tick test	T passcnt index
D - display menu	D
u - up one level menu	u
l - loop	l loopcount
h - help menu	h
q - back to monitor	q

Command syntax:

S passcnt index

passcnt - loop count, 1 - 0xFFFFFFFF, the default is 1.
index - Timer index, 0 - 5, the default is 0.

The system timing controller contains five timers. The timer-1 has special function, we don't check it. The timer index indicates which timer should be verified. If the user specifies the timer index to be zero, that means to check the interrupt ability of all the timers.

The timer interrupt test checks the ability of the selected timer to generate the interrupt signal when the time constant is reached zero. After one pass of this test, the following message will be displayed on the screen:

Pass 1 of Timer Interrupt Test : errors= 0, total errors= 0

If error(s) occurred during the timer interrupt test, the error message will be displayed on the screen as below:

Timer-X TIMEOUT, couldn't generate INT signal

The 'X' is the number of the timer under tested. Command syntax:

S passcnt index

passcnt - loop count, 1 - 0xFFFFFFFF, the default is 1.
index - Timer index, 0 - 5, the default is 0.

The timer index has the same meaning as above. If the user specifies the timer index to be zero, all of four timers will be verified at one time, that means they are turned on or off at one time.

The timer accuracy test checks the accuracy of the selected timer. The reference source comes from the vertical retrace signal of the video controller (the autovector interrupt level of video is level-4). The elapsed times of the different monitor are shown below:

elapsed time (for 1152X900 pixels monitor) = 14992 usec
elapsed time (for 1024X1024 pixels monitor) = 16976 usec

The run time message is shown below.

timer-X time(ref) = 0x39a0 time(used) = 0x39a8 deviation= +0.0800

After one pass of this test, the following message will be displayed on the screen:

Pass 1 of Timer Accuracy Test : errors= 0, total errors= 0

If error(s) occurred during the timer accuracy test, the error message will be displayed on the screen as below:

Timer-X TIMEOUT, couldn't generate INT signal

The 'X' is the number of the timer under tested.

Command syntax:

S passcnt index

passcnt - loop count, 1 - 0xFFFFFFFF, the default is 1.
index - Timer index, 0 - 5, the default is 0.

The timer index has the same meaning as above. If the user specifies the timer index to be zero, all of four timers will be verified at one time, that means they are turned on or off at one time.

After one pass of this test, the following message will be displayed on the screen:

Pass 1 of Timer Tick Test : errors= 0, total errors= 0

If error(s) occurred during the timer tick test, the error message will be displayed on the screen as below:

Timer-X couldn't tick

The 'X' is the number of the timer under tested.

The default test sequentially checks the following devices.

1. The MMU subtest checks the MMU logics.
2. The interrupt subtest verifies the abilities of the different devices on the CPU board to generate the interrupt signals.
3. The timer subtest checks the accuracy and the tick ability of the timers.
4. The SCC subtest checks the transmission capabilities of the SCC with the different configurations.

The following shows the default sequence:

m - It does a MMU test.
i - It does a interrupt test.
t ; T ; A ; I
- It jumps to the timer submenu, and does a tick, accuracy, and interrupt test of timers.
u - It jumps back to the main menu.
s - It jumps to the SCC submenu.
f ; b 38400 ; z 5 ; c 0 ; p 0 ; s 0
- It configures the baud rate to be 38400, the buffer size be 5 bytes, the character length sequence (6, 7, 8), the parity type sequence (NO, EVEN, ODD), the stop bit sequence (1, 1.5, 2).
u ; A - It does a Asynchronous loopback test with the poll mode enable.
f ; C 2 - It swaps the tested channel to be channel-B.
u ; A - Ditto the above loopback test.
f ; b 2400 ; p 1 ; c 3 ; s 2 ; z 1
- It reconfigures the baud rate to be 2400 (the medium rate), the parity type to be No parity, the character length to be 8_bit, the the stop bit to be 1.5_bit, the buffer size to be One byte.
u ; A - Ditto the above loopback test.
f ; C 1 - It then swaps the tested channel to be the channel-A.
u ; A - Ditto the above loopback test.
f ; b 600
- It reconfigures the baud rate to be 600.
u ; A - Ditto the above loopback test.
f ; C 2 - It swaps the tested channel to be the channel-B.
u ; A - Ditto the above loopback test.
f ; b 19200 ; k 64 ; m 1 ; z 128 ; C 0
- It reconfigures the baud rate to be 19200, the clock mode to be X64, the interrupt mode, the buffer size to be 16 bytes, the tested channels to be both channels.
u ; A - Ditto for the above loopback test.
f ; z 2048
- It reconfigures the buffer size to be 2048 bytes.
u ; A - Ditto for the above loopback test.
f ; b 38400 ; k 32
- It reconfigures the baud rate to be 38400, the clock mode to be X32.
u ; A - Ditto for the above loopback test.
f ; z 16- It reconfigures the buffer size to be 16 bytes.
u ; A - Ditto for the above loopback test.
f ; b 76800 ; k 16
- It reconfigures the baud rate to be 76800, the clock mode to be X16.
u ; A - Ditto for the above loopback test.
f ; z 2048
- It reconfigures the buffer size to be 2048 bytes.
u ; A - Ditto the above loopback test.
f ; b 19200 ; C 1
- It changes the tested channel to be the channel-A.
u ; B ; S
- It does the bisync and the SDLC tests sequentially.
f ; C 2 - It swaps the tested channel to be the channel-B.
u ; B ; S
- Ditto the above loopback tests.
f ; z 2048
- It reconfigures the buffer size to be 2048 bytes.
u ; B ; S

- Ditto for the above loopback tests.

u - It jump back to the main menu.

The following are the error messages which will be described in groups in more detail.

1. The following group of the error messages are related with the transmission capabilities of the SCC.

   "transmitter time out"

   is displayed when the testing program is trying to send a data over the chip, but, it can't see the transmitter ready flag for a long time.

   "receiver time out"

   is displayed when the testing program is expecting a data received, but, it couldn't see the flag of the received data available for a long time.

   "parity error"

   is displayed when the parity check logics of the SCC detected a parity error during the data transmission.

   "receiver data overrun error"

   is displayed when the data buffers of the SCC have been over written.

   "asynchronous framing error"

   is displayed when the starting bit was lost in the data transmission of the asynchronous protocol mode.

   "soft error, data error"

   is an error message when the received data didn't match the sending data.

2. The following group of the error messages are related with the modem signals of the SCC.

   "can not see cts change when rts changed from low to high"
   "can not see cts change when rts changed from high to low"
   "can not see dcd change when dtr changed from low to high"
   "can not see dcd change when dtr changed from high to low"

3. The following group of the error messages are related with the SCC.

   "channel-A rcv int - no such protocol"
   "channel-A xmit int - no such protocols"
   "channel-A SDLC rcv int is not enable"
   "channel-A SDLC xmit int is not enable"
   "channel-A Bisync rcv int is not enable"
   "channel-A Bisync xmit int is not enable"
   "channel-A Async rcv int is not enable"
   "channel-A Async xmit int is not enable"

   "channel-B rcv int - no such protocol"
   "channel-B xmit int - no such protocol"
   "channel-B SDLC rcv int is not enable"
   "channel-B SDLC xmit int is not enable"
   "channel-B Bisync rcv int is not enable"
   "channel-B Bisync xmit int is not enable"
   "channel-B Async rcv int is not enable"
   "channel-B Async xmit int is not enable"

   are displayed due to the interrupts occurred when the SCC is not enabled.

   "channel-A special conditions error"
   "channel-B special conditions error"

   are displayed when the special condition(s) occurred in the data transmission, such as, the starting bit

was lost, the CRC error, the parity error, and the receiver overrun error.

4. The following group of the error messages are related with the interrupt signals of the CPU board.

"No LEVEL-1 interrupt, check LEVEL-1 jumper"
"No LEVEL-2 interrupt, check LEVEL-2 jumper"
"No LEVEL-3 interrupt, check LEVEL-3 jumper"
"No LEVEL-4 interrupt, check LEVEL-4 jumper"

"Timer-2 couldn't generate interrupt signal"
"Timer-3 couldn't generate interrupt signal"
"Timer-4 couldn't generate interrupt signal"
"Timer-5 couldn't generate interrupt signal"

"cts-interrupt time out when rts changed from high to low"
"cts-interrupt time out when rts changed from low to high"
"dcd-interrupt time out when dtr changed from high to low"
"dcd-interrupt time out when dtr changed from low to high",
"baud rate zero count interrupt time out"

are displayed when the individual source is expecting an interrupt occurred, but, its never happened.

"Unexpected LEVEL-4 interrupt happened"
"Unexpected LEVEL-1 interrupt happened"
"Unexpected LEVEL-2 interrupt happened"
"Unexpected LEVEL-3 interrupt happened"

"cts interrupt happened when cts interrupt enable mask"
"dcd interrupt happened when cts interrupt enable mask"
"baud rate zero count interrupt happened when enable mask"
"unallowable interrupt happened"
"unexpected cts interrupt"
"unexpected dcd interrupt"
"unexpected baud rate zero count interrupt"

are displayed when the individual source is not expecting an interrupt, but, its occurred.

5. The following group of the error messages are related with the Timer of the CPU board.

"Timer-2 couldn't tick"
"Timer-3 couldn't tick"
"Timer-4 couldn't tick"
"Timer-5 couldn't tick"

are displayed when the timer couldn't decrement the counter.

"Timer-2 couldn't count accuracy"
"Timer-3 couldn't count accuracy"
"Timer-4 couldn't count accuracy"
"Timer-5 couldn't count accuracy"

are displayed when the time counted is not accurate.

It is recommended to follow the testing sequence below.

1. Run the MMU subtest to verify the MMU logic.
2. Run the interrupt subtest to check the capability of each source to generate the interrupt signal.
3. Run the timer subtest to check the accuracy and the ability to decrement the counter.
4. Run the asynchronous loopback subtest with different configurations (such as, the different baud rate, the different character bit, the different parity type, and the different stop bit). It is better started with the poll mode, then the interrupt mode.
5. Run the bisync loopback subtest with the different baud rate and the interrupt mode.
6. Run the SDLC loopback subtest with the interrupt mode and the different baud rate.
7. Run the clock subtest if there is the device on the CPU board.
8. Run the diagnostic register subtest to verify the LEDs.
9. Run the DVMA interface subtest to check the bus interface if the specified SCSI-DVMA devices are installed.

The following areas could be considered been enhanced in the future:

1. The DVMA test could be enhanced with different drives, such as, the large hard disk drive with SMD interface.
2. The point-to-point operation of the SCC test with different protocols could be enhanced.
3. The further analysis of the deviation of the clock and the timer tests.

Sun-2 Architecture Manual, Draft Version 0.5, 22 May 1984.

Theory of Operation Manual for the Sun-2/120 CPU Board, Version 28, Sept. 1984

Engineering Manual Sun-2 Model 50, REvision 01 of 12, October 1984

The Design Document of Sun-XXX CPU Board Diagnostic, Revision A, 1984.

Sun SCSI Theory of Operation.

Sun SCSI Programmers' Manual.

MM58167 Microprocessor Compatible Real Time Clock.

ZILOG Z8030/Z8530 SCC Serial Communications Controller Manual.

The Am9513 System Timing Controller, Advanced Micro Device, 1981

Analog and Communications Products, Advanced Micro Device, 1983.