

RaSCSI

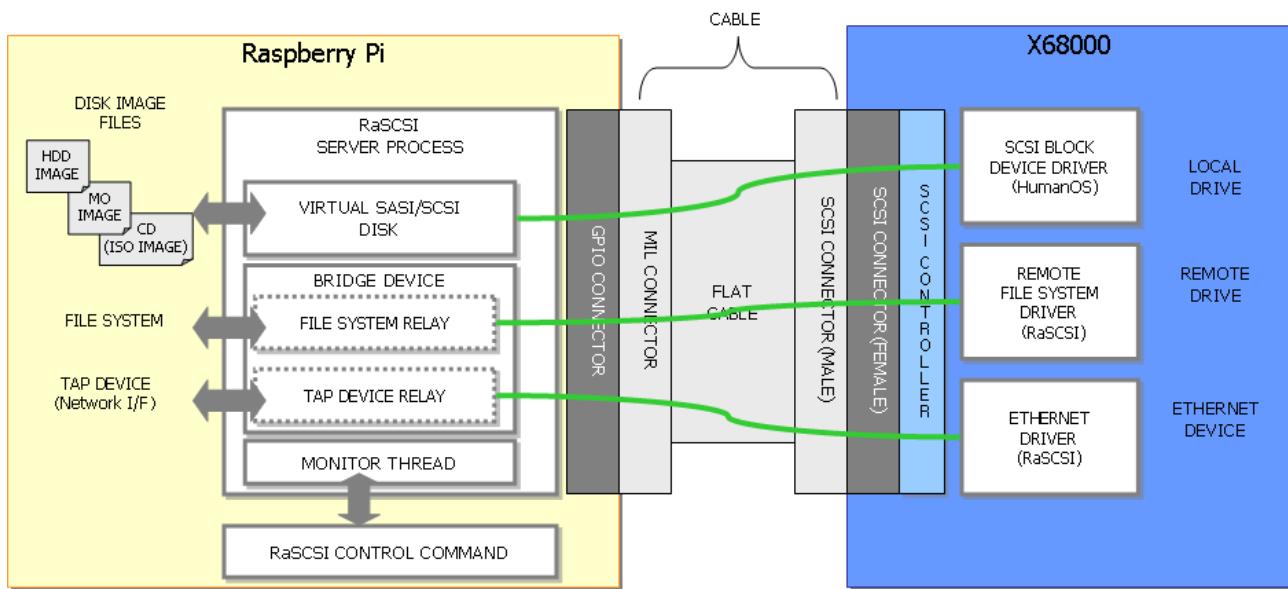
-Raspberry pi As a SCSI target device emulator for X68000-

Announcement

RaspberryPi4 is now available in stores, so I decided to upgrade. The recommended environment is 3B, but I think it will work with Zero/ZeroW/ZeroWH/2B/3A+/3B/3B+/4B.

Introduction

RaSCSI is a kind of emulator that virtually reproduces the SCSI devices (hard disk, MO, CD-ROM) that have become relics of the past. It was developed using the SCSI control of XM6 TypeG. The RPI behaves like a SCSI device when installed on a Raspberry Pi (hereinafter RPI). Use 18 RPI GPIO + 1 GND instead of SCSI connector. In order to connect RPI to X68000 (SCSI machine after SUPER), you need to make your own conversion cable. By the way, RaSCSI expands various functions by combining with X68000, but when used as a simple SCSI hard disk, I think that it can be used with other retro PCs that adopt SCSI such as FM TOWNS.



Feature

Virtual disk device (HDD, MO, CD)

Emulates a SCSI hard disk, magneto-optical (MO) disk, and CD-ROM. On the RPI side, the disk image file that can be created by XM6 is recognized as a virtual SCSI disk device, but on the X68000 side, it is recognized as if a physical device exists. We have officially supported SASI since version 1.24.

Ethernet communication

By installing the Ethernet driver provided by RaSCSI on the X68000 side, it becomes possible to communicate using the TAP device (virtual network I/F) through the bridge device on the RPI side. The driver on the X68000 side can be replaced with the Neptune-X driver.

Remote drive

By installing the remote file system driver provided by RaSCSI on the X68000 side, the RPI file system can be recognized as a drive through the bridge device on the RPI side. It provides the same functions as the WindrvXM of XM6 on a real machine.

Initiator mode

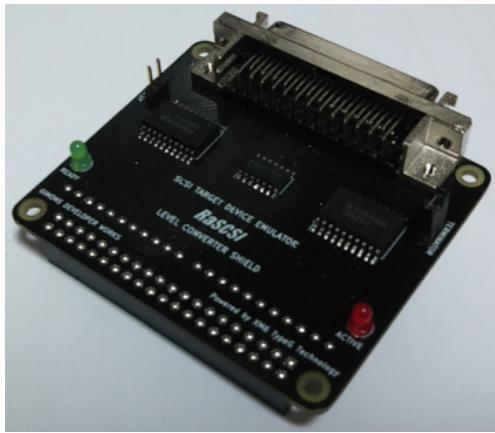
You can use the initiator mode to control the SCSI device (host side). This allows applications such as dumping and restoring physically connected SCSI hard disks. However, a conversion board that can use the direct connection or initiator mode described later is required.

Connection method (Conversion board)

You can create it yourself by referring to the circuit diagram below, or obtain it from a person who is distributed for a fee (consignment sale or BOOTH).



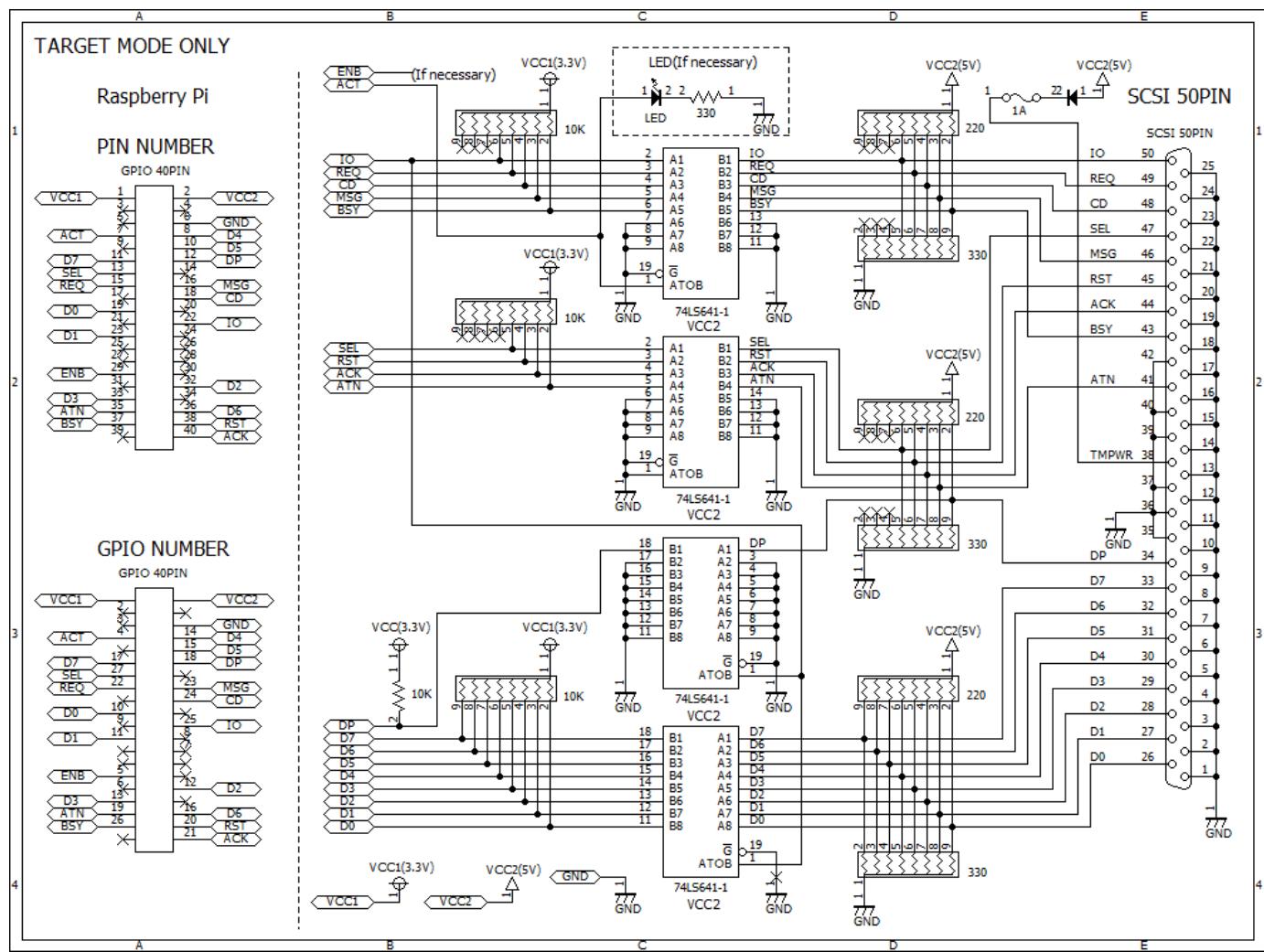
The author has also started to distribute it as an official version on BOOTH (it is irregular because the number is limited).
[GIMONS DEVELOPER WORKS\(BOOTH\)](#)



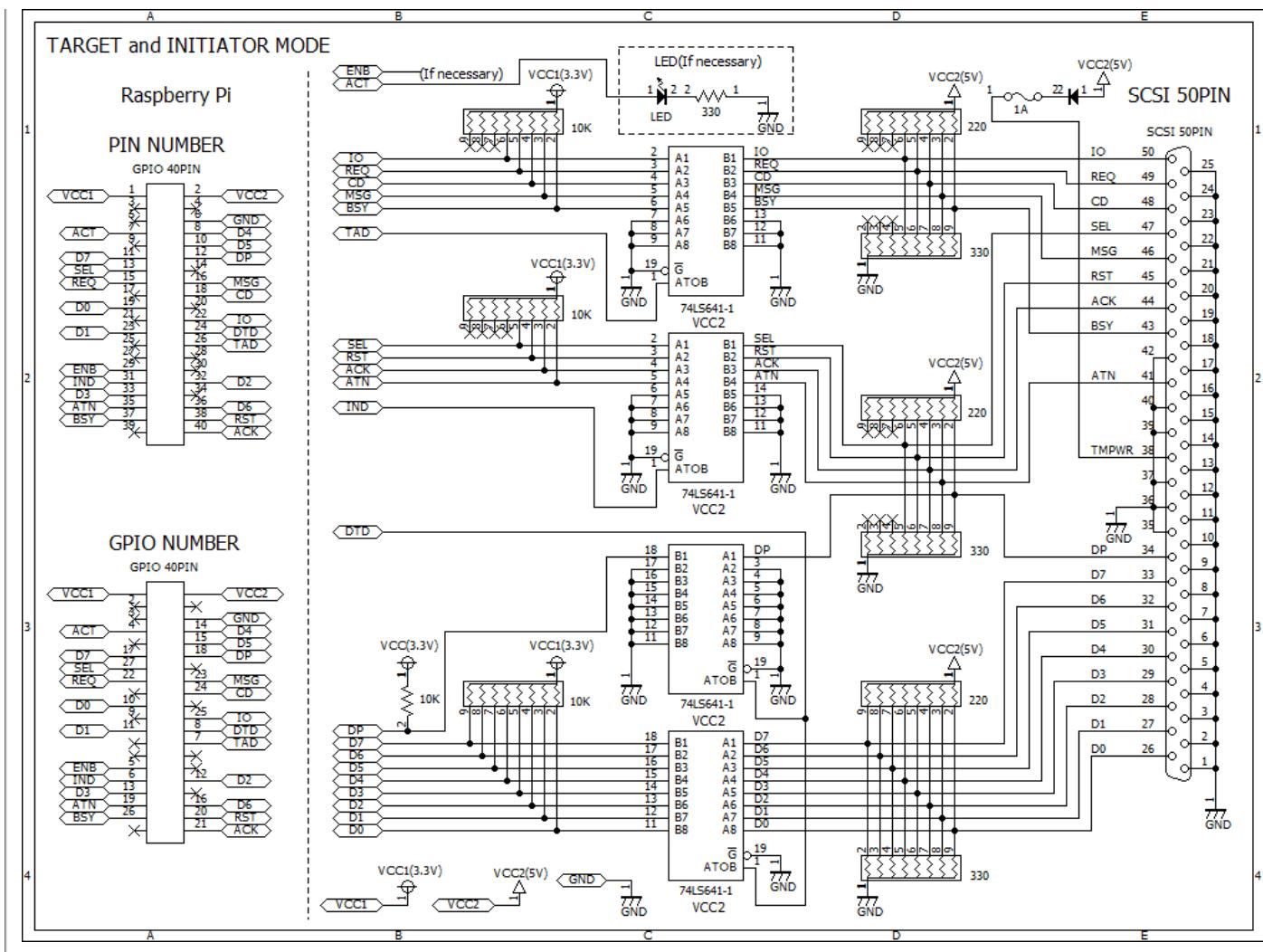
Circuit diagram of conversion board

For those who are trying to make their own conversion board, I am an amateur, but I will write down reference information. In SCSI, 5V is divided into 220Ω and 330Ω at the TTL level (in the case of a passive terminator), and a condition in which a voltage of less than 3V is applied to each signal line is in a steady state (the signal is negated). When the initiator side or the target side tries to assert the signal ($=0V$), a total current of $5000 \div 220 \times 2 = 45mA$ flows from the terminators on both ends (see the data sheet for the X68000 SCSI controller MB89352). As you can see, the sink current is $I_{OL} 48mA$. The RPI GPIO cannot absorb such a large sink current. To make an electrically safe connection, it is necessary to make a conversion board with a general-purpose logic IC. A general-purpose logic IC that can withstand a sink current of 48 mA uses an open collector such as 74LS06 or 07 with a high power type. The author performed basic SCSI direction control with 74HC541 $\times 3$, 74HC126 $\times 1$, 74HC04 $\times 1$ and further assembled a circuit for driving the bus using 74LS07 $\times 3$, and confirmed that it operated without problems. If you use 74LS641-1 which is a derivative of 74LS641, the circuit will be simple. Unlike the normal product, the sink current is compatible with 48mA, so there is no need to use the 74LS07. But availability is not so good.

It is an example of the circuit diagram of the conversion board that supports the target mode (HDD, MO, CD, etc.).



This is a circuit diagram example of a conversion board that supports target mode (HDD, MO, CD, etc.) and initiator mode (host function). I use three more GPIOs. The direction control of all 74LS641-1 is controlled from the RaSCSI side.



To those who distribute their own conversion board for a fee

If you want to distribute the conversion board for a fee, you do not need to obtain permission from the author as long as you follow the conditions below. Remember that the important thing is to provide enough information to the user who purchased the board.

1. Distribution price

Board production cost + Parts cost + Freight cost + (General social convention) fee.

2. Publishing the schematic

Please provide the schematic to the purchaser. At the same time as the circuit board distribution or separately from the website, you can download it from the home page.

3. Disclosure of verification results on the

X68000 actual machine It is mandatory to verify the operation of the X68000 actual machine environment. If you can't get the actual device, you may ask X68000 users to cooperate with the verification. For operation verification, please write or load test other than confirmation of startup. Please publish the verification result together with the environment used.

Connection method (Direct connection)

warning!

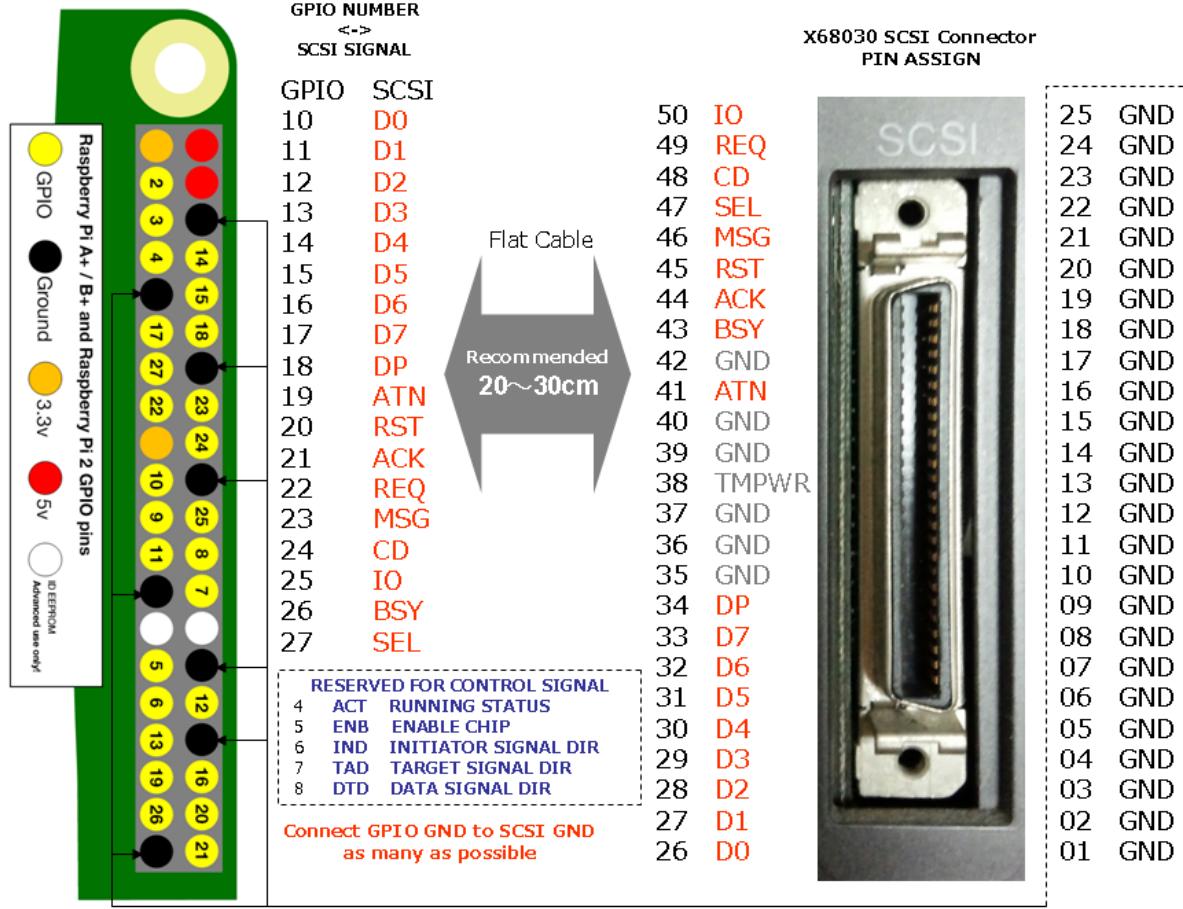
Direct connection is risky due to the difference in electrical specifications between GPIO and SCSI. A conversion board must be used to make the electrically safe connection. If you don't understand this risk, give up. Your X68000 is probably ok, but the Raspberry Pi is unpredictable. The unexpected load is the sink current explained in the converter board. If it is directly connected, there is no terminator, so a sink current of about 22mA will be generated, but since it exceeds the maximum drive capacity of GPIO, the sink current is too small to drive to around 0V. By the way, I have been running RaSCSI for several months, but so far it has not failed. Even if it breaks down, it is not expensive, so consider it a consumable item.

Cable connection diagram

Raspberry Pi (GPIO) <=> Connection (conversion) cable <=> (external SCSI connector) X68000 It

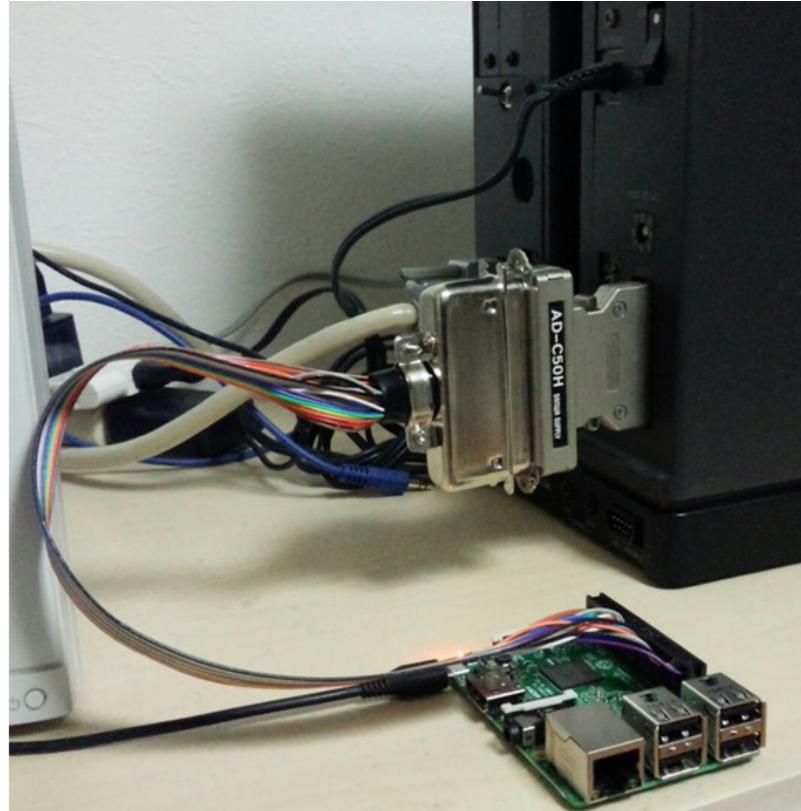
RASCSI PIN ASSIGN

UPDATE 2017/9/2



is stable if you pull out as many GND as possible on the SCSI connector side (8 if possible) and connect it to the GND on the RPI side. It seems to do.

Cable manufacturing example The



material used was a 40-pin MIL connector, Centronics 50-pin male, and flat cable. It is recommended to keep the flat cable within 30 cm (the length is longer than this and there is no actual result). The SCSI connector will be half size male (ribbon type) for X68030 and XVI Compact, and full size male for other SCSI machines. You can also use a conversion adapter as shown in the photo. GPIP pins aren't lined up neatly, and I'm using different pins to

make the app, so I'm worried about which way to adjust the connection. I usually soldered the SCSI connector side and crimped while checking the connections one by one on the MIL side.

Direct connection board example



There is also a direct connection board. Basically it is the same as the direct connection cable, but it seems that there is no problem if you use a SCSI cable because it is easy to electrically stabilize.

Operating environment (RPI)

It operates in the following operating environment.

Raspberry Pi

- Raspberry Pi Zero
- Raspberry Pi Zero W
- Raspberry Pi Zero WH
- Raspberry Pi 2 Model B
- Raspberry Pi 3 Model B (recommended)
- Raspberry Pi 3 Model A+
- Raspberry Pi 3 Model B+
- Raspberry Pi 4 Model B

Performance may be a little unstable with Zero/Zero W/Zero WH.

3 Model A+/3 Model B+/4 Model B has high performance, but the CPU clock may fluctuate due to the influence of heat, so it is necessary to take measures.

Supported OS

Developed and tested by RASPBIAN BUSTER. RaSCSI controls the SCSI signals using GPIO, so it should be used in the lowest latency possible. Therefore, we recommend that you use it in CUI mode.

Operating environment (X68000)

It operates in the following operating environment.

X68000

A SCSI machine of SUPER or later is recommended. Even models with additional SCSI boards have a track record of operation. Only SASI hard disk emulation can be used with the first generation, ACE, EXPERT, PRO and other SASI machines. We have not confirmed the operation of SxSI so much, so it would be helpful if you could report the operation. By the way, since parity is not used, it is not necessary to add a parity circuit.

Supported OS

Human 3.02 is recommended to use all functions, because the Ethernet function and the remote drive function provide only the Human device driver. If it is only SCSI storage, NetBSD etc. will be no problem.

Disk image

RaSCSI itself is a derivative of the XM6 TypeG, an X68000 emulator. Therefore, it is assumed that the disk image is created from the "Tools" menu of XM6 TypeG. Of course, it is also possible to create an empty image file with dd etc. according to the following specifications.

SCSI hard disk

HDS file format (extension HDS/HDN/HDI/NHD/HDA)

File size can be any size within the range of 10MB to 4095MB (however, 512-byte units).

If the extension is "HDN", NEC genuine hard disk emulation for the PC9800 series PC-9801-55 board is performed. There is a difference in the information returned by INQUIRY or MODE SENSE. When the extension is "HDI" or "NHD", the SCSI hard disk images of PC98 emulators Annex86 and T98Next are used. As with HDN, some information is converted for NEC (Thansk sava).

If the extension is "HDA", APPLE genuine hard disk emulation for Macintosh will be performed. There is a difference in the information returned by INQUIRY and MODE SENSE.

SASI hard disk

HDF file format (extension HDF)

) We recommend a file size of 10441728 bytes, 20748288 bytes, or 41496576 bytes (compatible with 10 MB drive, 20 MB drive, and 40 MB drive, respectively). Any file size from 10M to 512M can be mounted in 256-byte units.

Images from version 1.46 to 22437888 bytes are recognized as a 20MB image dedicated to the MZ-2500/MZ-2800 MZ-1F23 (a special image with a block size of 1024).

SCSI magneto-optical (MO) disk

MOS file format (extension MOS)

File size is one of the following 4 types.

128MB type (127,398,912 bytes)

230MB type (228,518,400 bytes)

540MB type (533,248,000 bytes)

640MB type (635,600,896 bytes)

128MB, 230MB, 540MB is 512 bytes / sector, 640MB will be 2048 bytes / sector.

SCSI CD-ROM

ISO file format (extension ISO, ISO9660 solid image)

Mode 1 (2048 bytes/sector) supports both files that store only data and files that are recorded in RAW format.

download

RaSCSI(version 1.47)

RaSCSI RPI program and X68000 drivers and documents. The source code for each program is also included in the archive.

[Download RaSCSI version 1.47 \(1,120,067 bytes\)](#)

Operation verification information

Information about the board that GIMONS has verified the operation.

Please check the operation results for other than X68000 from the homepage of each board manufacturer.

RaSCSI Version
1.44p1

Board author	type	model	RPI	target	Start (READ)	Copy (READ/WRITE)	format	RASDRV	stress test
GIMONS DEVELOPER WORKS	Direct link	Version1.0	3B	PRO(SASI)	○	○	○	○	DiskBench
				Compact (SCSI)	○	○	○	○	DiskBench
GIMONS DEVELOPER WORKS	Full spec	Version1.0	3B	PRO(SASI)	○	○	○	○	DiskBench
				Compact (SCSI)	○	○	○	○	DiskBench
Aibomu Products	Full spec	RaSCSI Adapter	3B	PRO(SASI)	○	○	○	○	DiskBench
				Compact (SCSI)	○	○	○	○	DiskBench
GAMERnium.com	Full spec	RaSCSI EX	3B	PRO(SASI)	○	○	○	○	DiskBench
				Compact (SCSI)	○	○	○	○	DiskBench
@naka_mobile version	Full spec	Board number ND161B	3B	PRO(SASI)	○	○	○	○	DiskBench
				Compact (SCSI)	○	○	○	○	DiskBench
Project MPS	Full spec	RaSCSI Zero Anfenoru half-pitch version	ZeroW	Via PRO (SASI) connector conversion	○	○	○	○	DiskBench
				Compact (SCSI)	○	○	○	○	DiskBench

screenshot

Disk full

SCSI ID can be freely assigned from 0 to 7. Usually, the PC itself has an ID assigned by the initiator, so the number is actually up to 7.



It
can also be used as a boot disk boot device.



The DSKBENCH
benchmark also hits a fair number. It's no different from a physical HDD (SCSI1, though).



It

was developed for the X68000 that can be used with FM TOWNS II, but it can be used normally with FM TOWNS. The power of RASDRV allows you to enjoy the feeling of WindrvXM even though it is



a real machine that uses the file of Razipi as it is.



Ethernet connection is also available

. TELNET from R6 to X68000, I don't need Neptune-X anymore

```

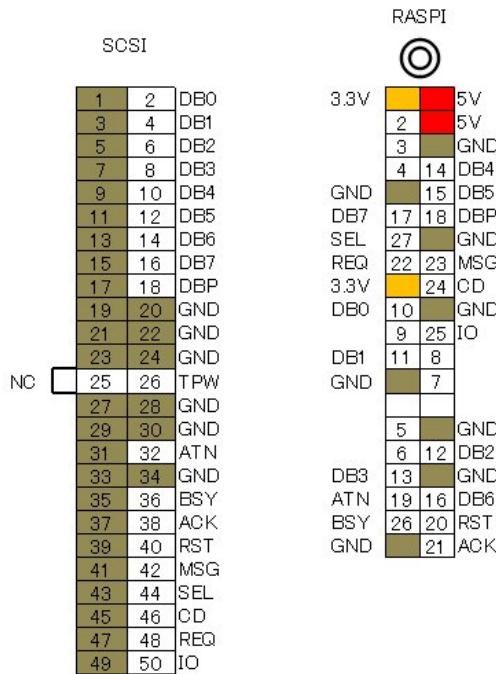
root      812  1 0 04:32 ttym1    00:00:00 /sbin/agetty --noclear ttym1 linux
root      862  740 0 04:35 ?      00:00:00 sshd: pi [priv]
pi       868  862 0 04:35 ?      00:00:00 sshd: pi@pts/0
pi       870  868 0 04:35 pts/0    00:00:00 -bash
root      898  2 0 04:35 ?      00:00:00 [kworker/0:1H]
root      962  740 0 04:41 ?      00:00:00 sshd: pi [priv]
root      966  2 0 04:41 ?      00:00:01 [kworker/3:0]
pi       971  962 0 04:41 ?      00:00:00 sshd: pi@pts/1
pi       973  971 0 04:41 pts/1    00:00:00 -bash
root     1081  2 0 04:57 ?      00:00:04 [kworker/u8:1]
pi       1711  870 0 06:25 pts/0    00:00:02 ftp 192.168.3.13
root     2164  2 0 07:13 ?      00:00:01 [kworker/u8:4]
root     2757  2 0 07:23 ?      00:00:00 [kworker/1:1]
pi       3005  1711 0 07:28 pts/0    00:00:00 bash
root     3014  2 0 07:28 ?      00:00:00 [kworker/0:0]
root     3015  2 0 07:28 ?      00:00:00 [kworker/3:2]
root     3020  973 0 07:28 pts/1    00:00:00 sudo ./rascsi -ID0 SCSIHD2.HDS -ID2 ETHER
root     3024  3020 38 07:28 pts/1    00:02:45 ./rascsi -ID0 SCSIHD2.HDS -ID2 ETHER
root     3091  2 0 07:29 ?      00:00:00 [kworker/2:0]
root     3135  2 0 07:34 ?      00:00:00 [kworker/u8:0]
telnetd  3136  414 0 07:35 ?      00:00:00 in.telnetd: 192.168.68.3
root     3137  3136 0 07:35 pts/2    00:00:00 login -h 192.168.68.3 -p
root     3138  2 0 07:35 ?      00:00:00 [kworker/1:0]
pi       3145  3137 1 07:35 pts/2    00:00:00 -bash
pi       3172  3145 0 07:35 pts/2    00:00:00 ps -ef
pi@raspberrypi:~
```

MITSUBISHI

Reference information

Signal line of

上から見た図



SCSI flat cable Information given by RaSCSI Warrior connected by SCSI flat cable.

History

2020/4/11 Version 1.47 released (Avoid the phenomenon that it does not start in 030 turbo 030 mode, other bug fixes)

2019/12/29 Version 1.46 released (Raspberry Pi 4 compatible, other bug fixes)

2019/4/28 Version 1.45 released (Source integration with XM6 TypeG, timeout set for busy release wait at SELECT)

2019/3/22 Version 1.44p1 released (Improved stability of rasdump)

2019/3/12 Version 1.44 released (Performance improvement, stability improvement in RPI Zero/w, correspondence of HD 0-16 designation)

February 17, 2019 Operation verification information added

2019/2/16 Version 1.43 released (SASI format problem and startup problem on PC98 genuine 55 solved)

2019/2/11 Version 1.42 released (SASI machine, slow SCSI machine, RASDRV stabilization, and many other fixes)

2019/2/4 Version 1.41 is released (processing of SASI \$C2 SPECIFY command is corrected)

2019/2/3 Version 1.40 released (kernel module abolished, many other fixes)

2018/12/11 Target & Initiator circuit diagram deleted (because it is complicated)

2018/4/26 Version 1.34 released (rasdump improvement, 256 bytes/sector released experimentally with HDN and HDI, merge of NetBSD compatible patch)

2018/2/10 Version 1.33 released (GAMERnium.com full spec version supported, timing correction)

2017/12/3 Version 1.32 is released (PC98 enhanced support, timing improvement, full spec version binary addition, other bug fixes)

2017/9/5 Version 1.31 released (initiator mode supported, RASDRV improved)

2017/7/1 Version 1.30 released (PC-9801-55 genuine NEC HD compatible, Macintosh Apple genuine HD compatible)

2017/6/25 Version 1.26 released (55 boards expansion for PC98, MAC support experiment)

2017/6/18 Version 1.25L10 released (Urgent response to changes in cputinfo specifications of RASPBIAN)

2017/6/18 Version 1.25 released (Transfer error processing and RST signal reception processing corrected)

2017/6/17 Version 1.24 released (Stability improvement by kernel driver)

2017/5/21 Version 1.23 is released (PC98 55 board compatible experiment, SASI hard disk compatible experiment, @132sync version, tomcat version binary added for conversion board)

2017/5/1 Version 1.22 released (Processing when receiving ATN signal corrected, ENABLE signal output added, SCSI control logic for conversion board added * Recompilation required)

2017/4/10 Version 1.21 is released.(Unnecessary code deleted, pin assignment changed, active signal output added, host file system bug fixed)

2017/4/09 Added circuit diagram to "Consideration of electrically safe connection method"

2017/3/23 Added "Consideration of electrically safe connection method" etc.

[EOF]