

AM-3031G(5V)

AM-3031G(5-12V)

ハードウェア

取扱説明書

2019 年 02 月 21 日

第 2 . 0 1 版

山下システムズ株式会社

安全にご使用いただくために

本製品は安全に十分配慮して設計を行っていますが、誤った使い方をすると火災や感電などの事故につながり大変危険です。ご使用の際は、警告/注意事項を必ず守って下さい。

警告

- 製品の分解や改造等は、絶対に行わないでください。
- 無理に曲げる、落とす、傷つける、上に重い物を載せることは行わないでください。
- 製品が水・薬品・油等の液体によって濡れた場合、ショートによる火災や感電の恐れがあるため、使用しないでください。

注意

- 高温多湿の場所、温度差の激しい場所、チリやほこりの多い場所、振動や衝撃の加わる場所、磁気を帯びた物の近くで保管しないでください。
- 煙や異臭が発生した場合は、直ちに本製品の電源を切り、電源ケーブルも外してください。
- 取り付け時、鋭い部分で手を切らないよう、充分注意して作業を行ってください。
- 配線を誤ったことによる損失、逸失利益等が発生した場合でも、いかなる責任も負いかねます。

製品保証

- 保証期間はご購入後 5 年間です。
 - 本製品の故障、もしくは不具合により発生した付随的損害の責については負いかねます。
 - 保証期間中は、保証範囲に従って修理または交換させていただきます。
 - 保証期間が過ぎている場合は、ご要望により有償修理させていただきます。
- ただし、不具合内容によっては、修理不可能な場合があります。

以下の場合は保証期間内であっても有償または保証対象外となります。

- 納入後の輸送時の落下衝撃等により生じた故障・破損。
- 取扱説明書または納入仕様書に記載された警告/注意事項に反する取扱による故障・破損。
- 落雷・火災等の天災、災害による故障・破損。異常電圧、周波数等による故障・破損。
- 当社以外での修理または、改造。
- ソフトウェアのバグ・不具合による故障・破損。

はじめに

AM-3031G(5V)、AM-3031G(5-12V)は、NXP 社 i.MX6 を搭載した 3.5in サイズのシングルボードコンピュータです。

ビデオ(LVDS,HDMI)、Gb Ethernet、シリアル、USB2.0、I2C、SD、microSD のインターフェースを搭載しております。

AM-3031G(5-12V)は、主電源として DC5V～DC12V が使用出来ます。

AM-3031G (5 V) は、5V 電源が必要な大型 LCD パネル (LVDS) や 5V 電源が必要なバックライトを使用する用途に適合するようになっております。

本製品は、単一電源駆動、ファンレス、低消費電力といった特長により、幅広い応用が可能な組み込み CPU ボードとなっております。

AM-3031G(5V)と AM-3031G(5-12V)の仕様の違い

製品名 機能	AM-3031G (5V)	AM-3031G(5-12V)
電源入力	5V のみ	5V～12V
入力電源電圧低下検知機能	4.6V 以下で電源断	なし
LVDS J7 5V 定義ピン	5V 出力あり	5V 出力なし
LVDS J10 5V 定義ピン	5V 出力あり	5V 出力なし
バックライト J6	出力あり	出力あり
バックライト J13	出力なし	出力あり

注. <Linux>以下に示された例は、弊社で Yocto Project を使ってビルドされた Linux 上でのコマンドを使って各デバイスの動作を確認できるように示したものです。

◆ ◆ ◆ 寸法図 ◆ ◆ ◆

D2400816_300.PDF AM-3031G 外形寸法図
D2400817_300.PDF AM-3031G コネクタ寸法図
D2400818_300.PDF AM-3031G パネル側寸法図

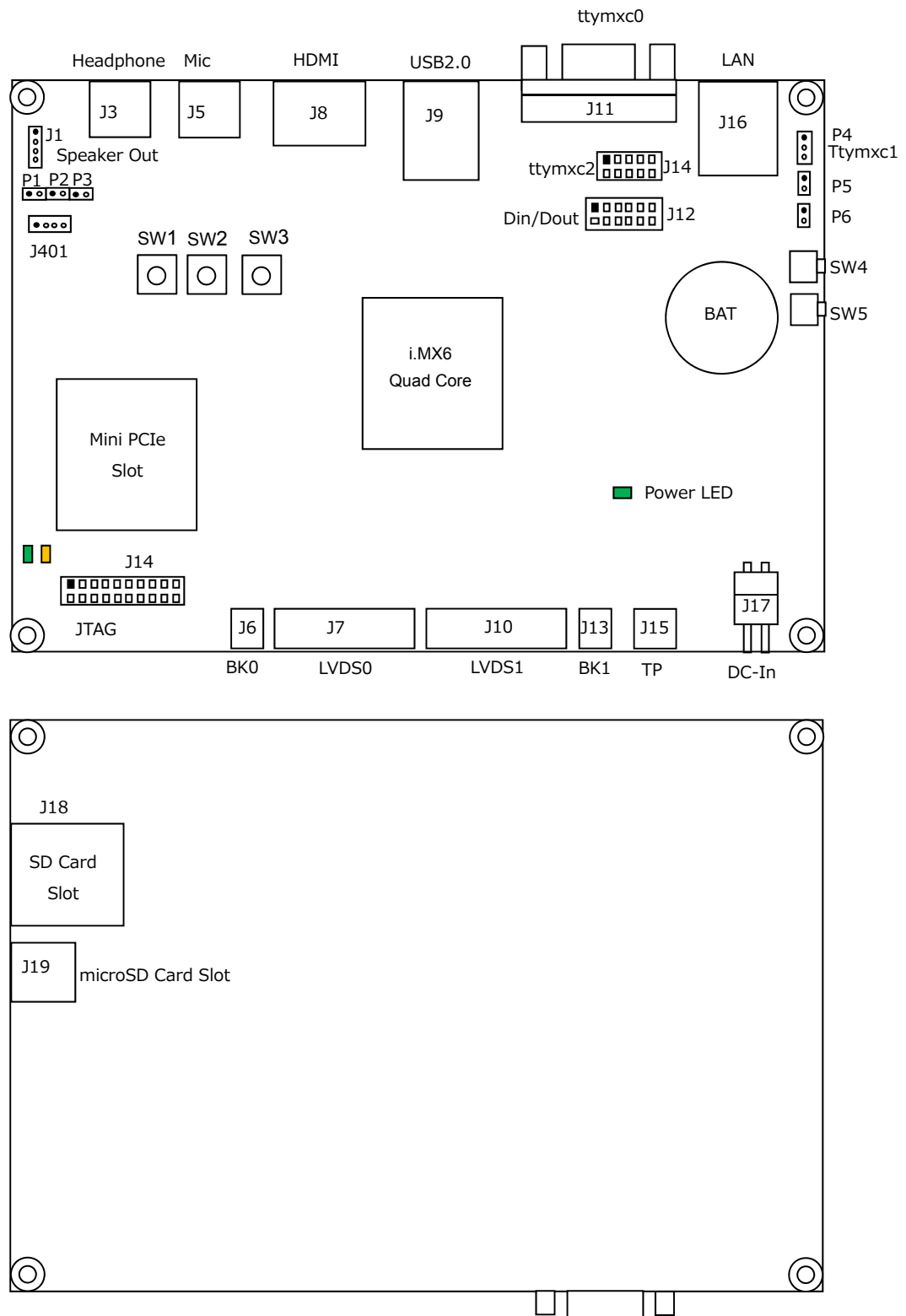
◆ ◆ ◆ 改版履歴 ◆ ◆ ◆

版数	発行日	事由
第 1.00 版	2017.05.17	Rev.A 対応
第 2.00 版	2018.05.30	Rev.C 対応 I2C コネクタ追加、ヒートシンク変更 (5V)と(5-12V)の取説を統合
第 2.01 版	2019.02.21	(5V) バックライト J6,J13 出力有無の訂正

1	製品仕様	4
1-1	AM-3031G レイアウト.....	4
1-2	機能仕様.....	4
1-3	ブロックダイアグラム	6
1-4	電源仕様.....	7
1-5	環境仕様.....	7
1-6	構造仕様.....	7
2	コネクタ	8
2-1	背面パネルのコネクタ	8
2-1-1	RJ-45 LAN ポート(J16).....	8
2-1-2	ttymxc0 コネクタ(J11)	10
2-1-3	USB2.0 ポート(J9).....	11
2-1-4	HDMI ポート(J8)	12
2-1-5	Headphone ポート(J3).....	13
2-1-6	Microphone ポート(J5)	14
2-2	内部のコネクタ	15
2-2-1	電源入力コネクタ(J17).....	15
2-2-2	SD カードコネクタ(J18)	16
2-2-3	microSD カードコネクタ(J19)	16
2-2-4	ttymxc1 コネクタ(P4).....	17
2-2-5	ttymxc2 コネクタ(J14)	17
2-2-6	D-I/O コネクタ(J12)	18
2-2-7	LVDS0 コネクタ(J7)	21
2-2-8	LVDS1 コネクタ(J10)	22
2-2-9	BK0 コネクタ(J6).....	23
2-2-10	BK1 コネクタ(J13)	24
2-2-11	TP コネクタ(J15).....	25
2-2-12	Speaker コネクタ (J1).....	25
2-2-13	I2C コネクタ(J401).....	26
2-2-14	JTAG コネクタ(J4)	27
2-2-15	Mini PCIe Slot	28
2-2-16	RTC バックアップ	29
2-2-17	SW1～SW5.....	30

1 製品仕様

1-1 AM-3031G レイアウト



1-2 機能仕様

項 目	仕 様	
CPU	NXP i.MX6 Quad Core Cortex-A9 1GHz	
	32KB L1 Instruction Cache, 32KB L1 Data Cache 1MB unified I/D L2 Cache, shared by four cores	
System Memory	PC3-8500 DDR3L 1GB (実装済み)	
Video I/F	Processor Graphics	Vivante GC 2000+ Vivante GC 355+ Vivante GC 320
	VRAM	Unified Memory
	表示方式(最大解像度)	HDMI (1920x1080) 注 1
		LVDS 2ch (1920x1080) 注 2
Touch Panel I/F	TI TSC2007	4 線式抵抗膜方式 (シングルタッチ) 対応
Storage I/F	SD Card Slot x1	カード仕様 : SD SDHC,SDXC
	micro SD Card Slot x1	カード仕様: microSD
	容量、スピード	SDHC,SDXC,UHS-I (UHS-II 非対応)
Serial I/F	RS232C	9pin Dsub Connector(Male)
	RS232C	2mm Pitch 10pin Header
	RS232C (Debug Console)	2.54mm Pitch 3pin Header
LAN I/F	MICREL KSZ9031RNXIA	Gigabit Ethernet RJ-45
USB	USB2.0 (Back Panel x2)	
Audio	マイク入力	
	ヘッドフォン出力	
	スピーカ出力	
DIO	デジタル入力(4bit)、デジタル出力(4bit)	
I2C	x1 (3.3V 仕様) Standard Mode, 割込みサポートなし	
Switch	H/W Reset , Power On	
Misc	H/W Reset, Power On , S/W Reset	
	Power LED, Volume-Up, Volume-Down	
External RTC	TI BQ32000DR	
	ボード上のリチウム電池にてバックアップ	

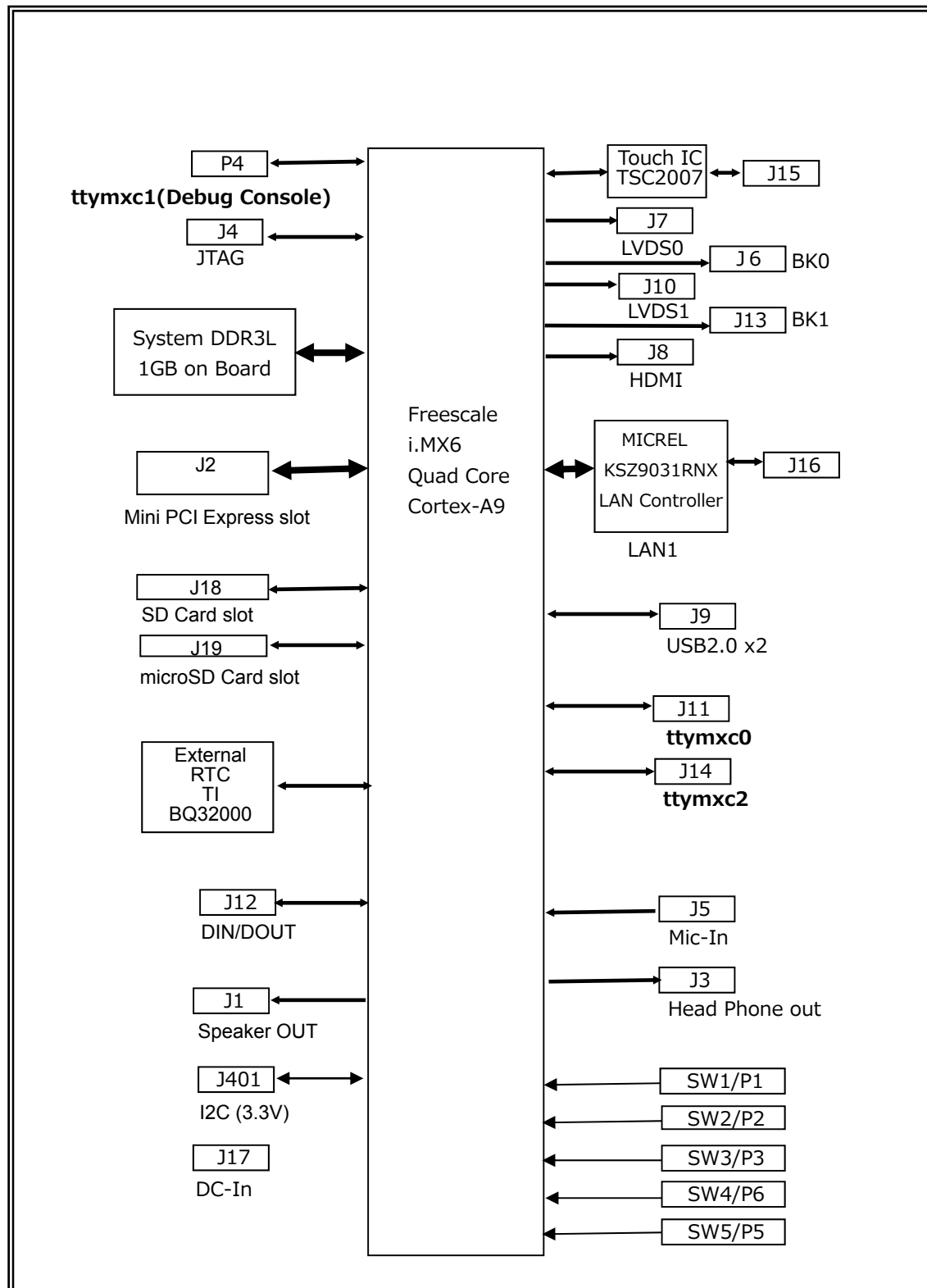
注 1. 640x480, 1024x768,1280x800,1366x768,1280x720p(16:9),1920x1080p(16:9)

注 2. LVDS 1ch : 800x480, 1024x600, 1024x768, 1280x800, 1366x768

LVDS 2ch: 1920x1080

(実際には各解像度に対応した linux OS を準備する必要がありますので、弊社にお問合せください。)

1-3 ブロックダイアグラム



1-4 電源仕様

AM-3031G(5V)の入力電源電圧範囲は **4.75VDC から 5.25VDC** です。

電源電圧	許容範囲	消費電流(平均値)	消費電流(瞬間最大)
DC +5V	±5%	1.5A	1.9A

AM-3031G(5-12V)の入力電源電圧範囲は **4.75VDC から 13.2VDC** です。

電源電圧	許容範囲	消費電流(平均値)	消費電流(瞬間最大)
DC +5V	±5%	1.5A	1.9A
DC +12V	±10%	0.6A	0.8A

注 1 AM-3031G(5V)には入力電圧が 4.6V まで降下すると、リセットを発生する回路を内蔵しています。

注 2 消費電流は下記条件における実測値です。

Qt5 サンプルプログラム Qt_CinematicExperience を実施中の実測値

消費電流に含まれる部材

マイクロ SD Card(16GB)

USB Keyboard / Mouse

平均消費電流は電源ラインをクランプ電流計にて測定

瞬間最大消費電流は電源ラインに電流プローブにて測定

- ※ 電源ユニットは、装置全体の消費電力を考慮し適切な容量の物をご使用ください。
大容量の電源ユニットの場合、装置全体の消費電力が少ないと電圧が正常に出力されない場合があります、誤動作の原因となる場合があります。
- ※ 電源ケーブルを本製品に接続してから、電源 ON にしてください。
電源 ON 状態にてケーブルを接続すると電圧スパイクが発生し、本製品が破損する恐れがあります。

1-5 環境仕様

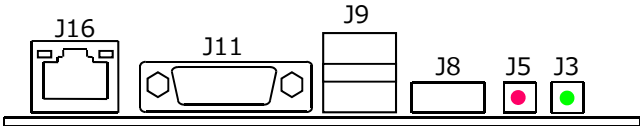
項 目	仕 様
使用温度範囲	-20 ～ 70℃ : 筐体内の放熱対策が施されていること
湿度範囲	20 ～ 80% : 結露せぬこと
腐食性ガス等	なきこと

1-6 構造仕様

項目	仕 様	
外形寸法	縦寸法	102mm : コネクタ張出部を含まず
	横寸法	145mm : コネクタ張出部を含まず
	高寸法	16.0mm : PCB 上面よりヒートシンク固定ピン上面 4.3mm: PCB 下面より
重量	153g 以下 (ヒートシンクを含む)	
PCB 材質等	FR-4(ULVO) : 1.6mm 厚	

2 コネクタ

2-1 背面パネルのコネクタ

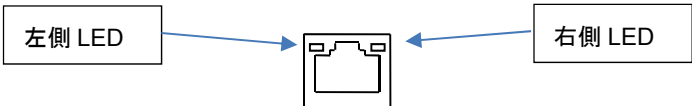


2-1-1 RJ-45 LAN ポート(J16)

LAN 接続用コネクタです。ネットワークを 1000BASE で運用される場合には、エンハンスドカテゴリ 5(CAT5e)以上のケーブルをご使用ください。

RJ45 コネクタに貼ってあるシールは、MAC アドレスの 00:50:01:XX:YY:ZZ の[XX:YY:ZZ]の部分を示しています。

コネクタの上部に配置されている左右の LED は以下の動作状態を表示しています。



Speed	左側 LED (黄色)		右側 LED (緑色、橙色)	
	Link Down 時	Link Up 時	Link Down 時	Link Up 時
1000/Full	消灯	点灯	消灯	緑色点灯
100/Full	消灯	点灯	消灯	橙色点灯
10/Half	消灯	点灯(注 1)	消灯	消灯

注. 通信量が多いと点滅する。

<Linux>

```
# ifconfig<Enter>      (ネットワークのコンフィグレーションを表示)

eth0      Link encap:Ethernet  HWaddr 00:50:01:3A:01:05
          inet addr:192.168.50.13  Bcast:192.168.50.255  Mask:255.255.255.0
          inet6 addr: fe80::250:1ff:fe3a:105/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:2091 errors:0 dropped:0 overruns:0 frame:0
          TX packets:356 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:207234 (202.3 KiB)  TX bytes:72934 (71.2 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
```

```
inet6 addr: ::1/128 Scope:Host
UP LOOPBACK RUNNING MTU:65536 Metric:1
RX packets:10 errors:0 dropped:0 overruns:0 frame:0
TX packets:10 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:700 (700.0 B) TX bytes:700 (700.0 B)
```

注. “HWaddr 00:50:01:3A:01:05”は 本ボードのMACアドレスを示します。

上位の “00:50:01”は弊社ベンダーコードです。

“inet addr:192.168.50.13”は IPアドレスです。

ping -c 4 192.168.50.1<Enter> (192.168.50.1 に対して 4 回 ping テストを実施)

```
PING 192.168.50.1 (192.168.50.1): 56 data bytes
64 bytes from 192.168.50.1: seq=0 ttl=128 time=0.780 ms
64 bytes from 192.168.50.1: seq=1 ttl=128 time=0.735 ms
64 bytes from 192.168.50.1: seq=2 ttl=128 time=0.694 ms
64 bytes from 192.168.50.1: seq=3 ttl=128 time=0.688 ms
```

--- 192.168.50.1 ping statistics ---

```
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 0.688/0.724/0.780 ms
```

ping -c 4 -s 1016 192.168.50.1<Enter> (192.168.50.1 に対して パケットサイズ 1016 バイトで 4 回 ping テストを実施例)

```
PING 192.168.50.1 (192.168.50.1): 1016 data bytes
1024 bytes from 192.168.50.1: seq=0 ttl=128 time=0.838 ms
1024 bytes from 192.168.50.1: seq=1 ttl=128 time=0.815 ms
1024 bytes from 192.168.50.1: seq=2 ttl=128 time=0.775 ms
1024 bytes from 192.168.50.1: seq=3 ttl=128 time=0.765 ms
```

--- 192.168.50.1 ping statistics ---

```
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 0.765/0.798/0.838 ms
```

注. Linux では **ifconfig** コマンド、Windows では **ipconfig** コマンド。

Linux の ping コマンドでは、Ctrl+C キーが押されるまで続行します。

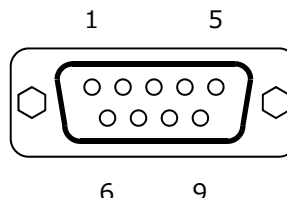
Windows の ping コマンドでは、4 回で終了します。

2-1-2 ttymxc0 コネクタ(J11)

RS232 接続用 Dsub(プラグ)コネクタです。Linux 上デバイスとしては **ttymxc0** となります。

シリアルポート コネクタ&ピン配列

No.	RS-232
1	NC
2	RxD
3	TxD
4	NC
5	GND
6	NC
7	RTS
8	CTS
9	NC



<Linux>

```
# stty -F /dev/ttymxc0<Enter> [ttymxc0 の設定状態表示]
speed 9600 baud; line = 0;
-brkint -imaxbel
```

```
# stty -F /dev/ttymxc0 -a<Enter> [ttymxc0 のすべての設定状態表示]
speed 9600 baud; rows 0; columns 0; line = 0;
intr = ^C; quit = ^\; erase = ^?; kill = ^U; eof = ^D; eol = <undef>;
eol2 = <undef>; swtch = <undef>; start = ^Q; stop = ^S; susp = ^Z; rprnt = ^R;
werase = ^W; lnxt = ^V; flush = ^O; min = 1; time = 0;
-parenb -parodd -cmspar cs8 hupcl -cstopb cread clocal -crtsets
-ignbrk -brkint -ignpar -parmrk -inpck -istrip -inlcr -igncr icrnl ixon -ixoff
-iuclc -ixany -imaxbel -iutf8
opost -olcuc -ocrnl onlcr -onocr -onlret -ofill -ofdel nl0 cr0 tab0 bs0 vt0 ff0
isig icanon iexten echo echoe echok -echonl -noflsh -xcase -tostop -echoprt
echoctl echoke
```

```
# cat /dev/ttymxc0<Enter> [ttymxc0 から文字列を受信とエコーバックあり(設定 echo)]
ABC [外部端末から"ABC<Enter>"、Terminal と外部端末に"ABC"表示]
[Ctrl+C キーで終了]

# stty -F /dev/ttymxc0 -echo<Enter> [エコーバックなし設定]
# cat /dev/ttymxc0<Enter> [ttymxc0 から文字列を受信とエコーバックなし(設定 -echo)]
ABC [外部端末から"ABC<Enter>"、Terminal に"ABC"表示]
[Ctrl+C キーで終了]
```

echo "hello world" > /dev/ttymx0<Enter> [ttymx0 へ文字列"hello world"を送信]

echo "!#\$%&()-=^~+;:*{}[]_<>" > /dev/ttymx0<Enter>
[ttymx0 へ"!#\$%&()-=^~+;:*{}[]_<>"を送信]

minicom -s [シリアル端末ソフトの初期設定]
[CLI 画面の serial port setup を選択し、Serial device ttymx0 の設定を行います。]

```
A - Serial Device      : /dev/ttymx0
B - Lockfile Location  : /var/lock
C - Callin Program     :
D - Callout Program    :
E - Bps/Par/Bits       : 9600 8N1
F - Hardware Flow Control : No
G - Software Flow Control : No

Change which setting?
```

[Save setup as dfl を実行すると、 /etc/minirc.dfl として保存されます。]

minicom [シリアル端末ソフトを実行]

Welcome to minicom 2.7

OPTIONS: I|8n

Compiled on nov 21 2016. 15:47:41

Port /dev/ttymx0, 11:17:10

Press CTRL-A Z for help on special keys

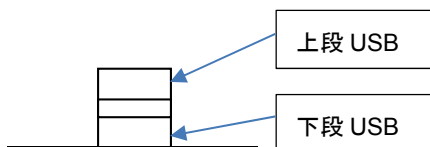
最下行に

CTRL-A Z for help | 9600 8 N1 |NOR|Minicom 2.7 |VT102|Offline| ttymx0

2-1-3 USB2.0 ポート(J9)

USB デバイス接続用コネクタです。上下段で 2 ポートです。

USB2.0/1.1 仕様をサポートしています。



注. USB の各ポートの電流値は 500mA 以下で使用します。

過電流保護回路は 700mA から 900mA の間で動作し、USB ポートの電源を OFF にします。

過電流保護回路は、電源を一度 OFF にするか、H/W RESET (SW5)を押すことで解除されます。

2-1-4 HDMI ポート(J8)

標準 HDMI 入力 ディスプレイ接続用コネクタです。HDMI 1.4 仕様をサポートします。

本 HDMI インターフェースの 5V が供給できる電流 55mA です。

注意として、55mA 以上の電流が必要な HDMI 機器を接続する場合は、外部から給電する必要があります。



注) 映像と音を出すことができます。

<Linux>

aplay -l<Enter> [オーディオ・デバイス一覧を表示]

**** List of PLAYBACK Hardware Devices ****

card 0: wm8962audio [wm8962-audio], device 0: HiFi wm8962-0 []

Subdevices: 1/1

Subdevice #0: subdevice #0

card 0: wm8962audio [wm8962-audio], device 1: HiFi-ASRC-FE (*) []

Subdevices: 1/1

Subdevice #0: subdevice #0

card 1: imxhdmisoc [imx-hdmi-soc], device 0: i.MX HDMI Audio Tx hdmi-hifi-0 []

Subdevices: 1/1

Subdevice #0: subdevice #0

aplay -D plughw:1,0 music.wav<Enter> [HDMI モニターに音出力]

Playing WAVE 'music.wav' : Signed 16bit Little Endian, Rate 44100Hz, Stereo

2-1-5 Headphone ポート(J3)

ヘッドフォン接続用コネクタ(3.5mmΦ)です。

注. ヘッドフォンを挿すと、スピーカ(J1)からの出力が止まり、
ヘッドフォンを抜くと、スピーカ(J1)から出力されます。

<Linux>

aplay -l<Enter> [オーディオ・デバイス一覧を表示] (*)はデフォルト値を示す。

**** List of PLAYBACK Hardware Devices ****

card 0: wm8962audio [wm8962-audio], device 0: HiFi wm8962-0 []

Subdevices: 1/1

Subdevice #0: subdevice #0

card 0: wm8962audio [wm8962-audio], device 1: HiFi-ASRC-FE (*) []

Subdevices: 1/1

Subdevice #0: subdevice #0

card 1: imxhdmi-soc [imx-hdmi-soc], device 0: i.MX HDMI Audio Tx hdmi-hifi-0 []

Subdevices: 1/1

Subdevice #0: subdevice #0

aplay -D plughw:0,1 music.wav <Enter> [ステレオスピーカ出力及びヘッドフォン出力]

Playing WAVE 'music.wav' : Signed 16bit Little Endian, Rate 44100Hz, Stereo

aplay -D plughw:0,0 music.wav<Enter> [ステレオスピーカ出力及びヘッドフォン出力]

Playing WAVE 'music.wav' : Signed 16bit Little Endian, Rate 44100Hz, Stereo

2-1-6 Microphone ポート(J5)

マイクrophon接続用コネクタ(3.5mmΦ)です。

<Linux>

以下のコマンドで録音と再生を行うことができます。

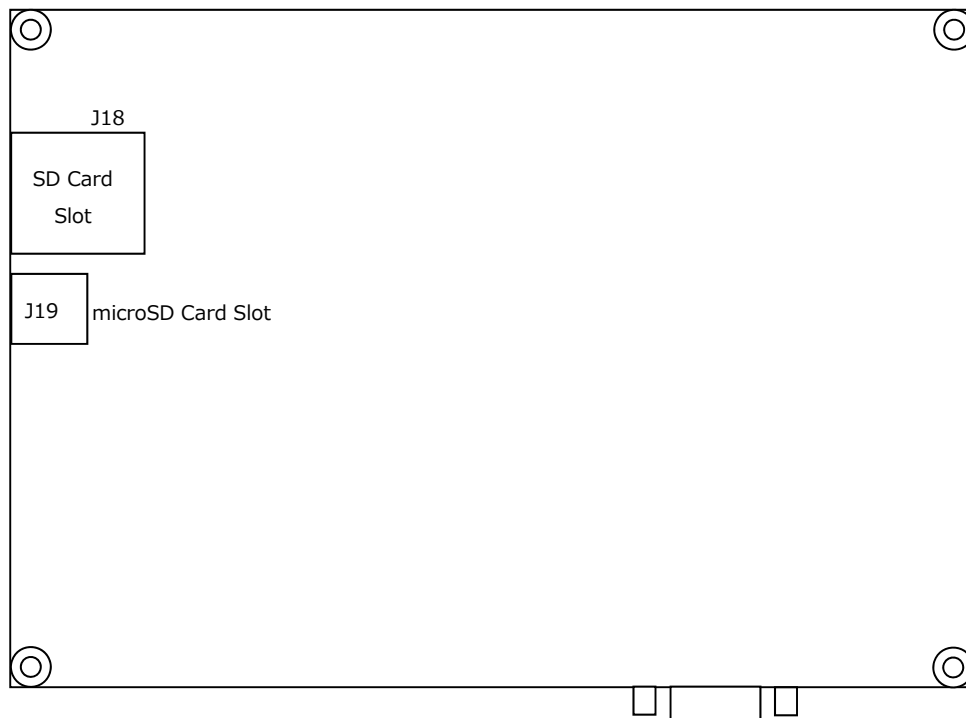
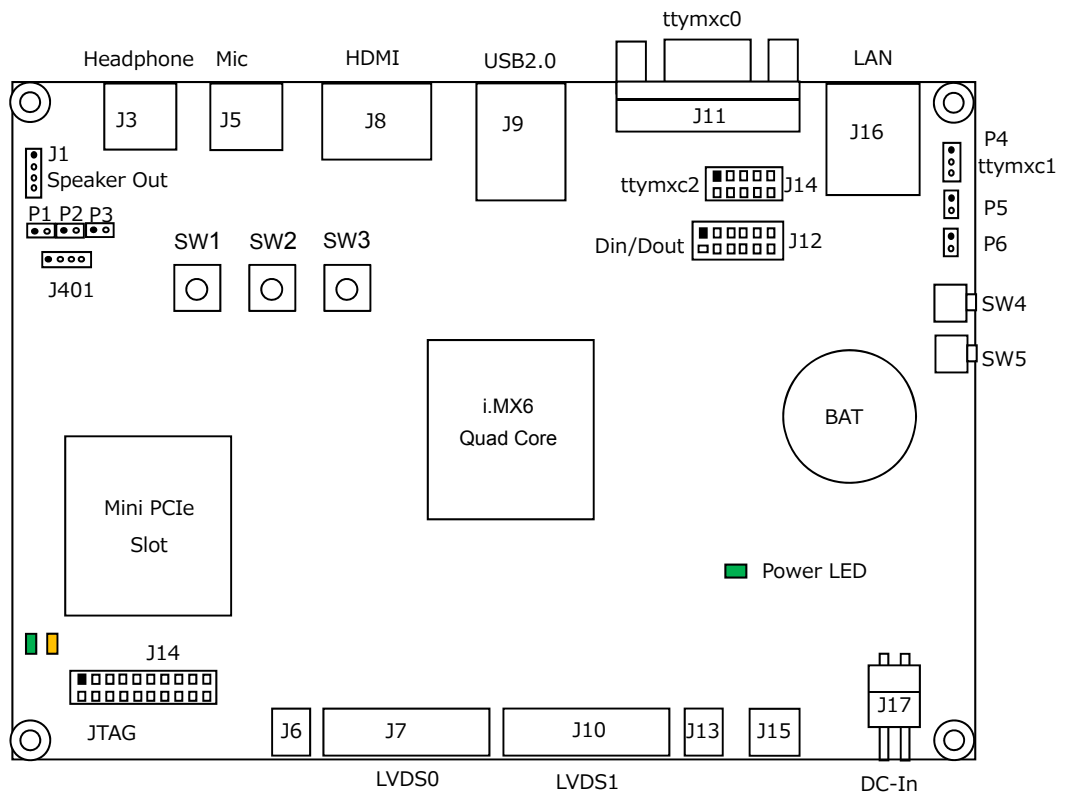
```
# arecord -d 10 -f cd ~/test.wav<Enter>    [コマンド入力後 10 秒間録音]
```

Recording WAVE 'home/root/test.wav': Signed 16bit Little Endian, Rate 44100hz, Stereo

```
# aplay ~/test.wav<Enter>    [録音データファイル test.wav から右スピーカから再生]
```

Playing WAVE 'test.wav': Signed 16bit Little Endian, Rate 44100Hz, Stereo

2-2 内部のコネクタ



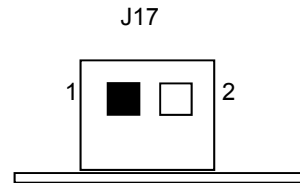
2-2-1 電源入力コネクタ(J17)

電源入力コネクタと電源を接続した後、規定範囲電圧を供給してご使用下さい。
電源入力コネクタは 2Pin 3.96mm Pitch コネクタです。

電源入力コネクタ & ピン配列

AM-3031G(5V)

No	信号名
1	5V
2	GND



AM-3031G(5-12V)

No	信号名
1	5V~12V
2	GND

適合コネクタ

ハウジング : VHR-2N JST

コンタクト : SVH-41T-P1.1 JST

2-2-2 SD カードコネクタ(J18)

SD カード接続用コネクタです。

<Linux>

```
# cd /system/class/mmc_host/mmc2<Enter>
```

```
# ls<Enter>
```

```
device mmc2:59b4
```

```
power subsystem uevent
```

```
# cd mmc2:59b4<Enter>
```

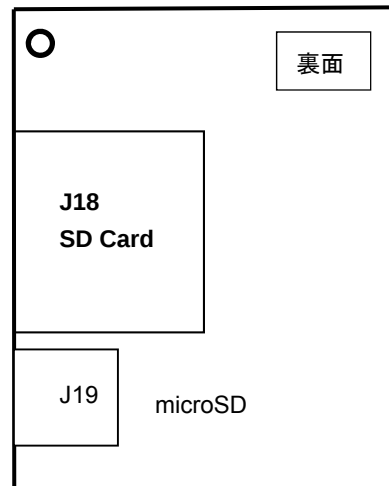
```
# cat name<Enter>
```

```
SD2G
```

```
# cat type<Enter>
```

```
SD
```

注. 抜くときは、奥に押すと SD カードが出てきます。



2-2-3 microSD カードコネクタ(J19)

microSD カード接続コネクタです。

<Linux>

```
# cd /system/class/mmc_host/mmc1<Enter>
```

```
# ls<Enter>
```

```
device mmc1:8fe8 power subsystem uevent
```

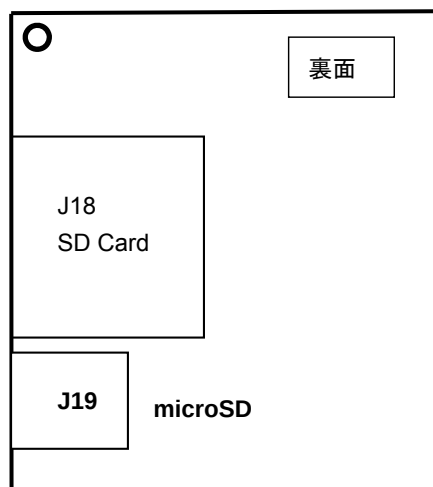
```
#cd mmc1:8fe8<Enter>
```

```
# cat name<Enter>
```

```
SD2G
```

```
# cat type<Enter>
```

```
SD
```



注. 抜くときは、奥に押すと microSD カードが出てきます。

2-2-4 ttymxc1 コネクタ(P4)

ttymxc1 コネクタは RS232 シリアルコンソールポート(ttymxc1)です。
3Pin 2.54mm Pitch Pin Header です。

No.	RS-232
1	RxD
2	TxD
3	GND



適合コネクタ

ハウジング : RE-03 JST
 コンタクト : RF-SC2210(LF)(SN) (金メッキ) JST
 適合電線 : AWG24~AWG30

<linux>

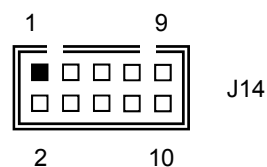
```
# stty -F /dev/ttymxc1<Enter> [ttymxc1 の設定状態表示]
speed 115200 baud; line = 0;
eol = ^J;
-brkint ixoff -imaxbel
-iexten
```

2-2-5 ttymxc2 コネクタ(J14)

ttymxc2 コネクタは RS232 シリアルポート(ttymxc2)です。
10Pin 2.00mm Pitch Box Header で、オプションのケーブル(SL812226910)により
Dsub 9 ピンに変換することができます。

COM コネクタ&ピン配列

No.	RS-232
1	NC
2	NC
3	RxD
4	RTS
5	TxD
6	CTS
7	NC
8	NC
9	GND
10	NC



適合コネクタ

ハウジング : DF11-10DS-2C HIROSE
 コンタクト : DF11-22SCA (金メッキ) HIROSE
 適合電線 : AWG26~AWG28

<Linux>

```
# stty -F /dev/ttymx2<Enter> [ttymx2 の設定状態表示]
Speed 9600 baud; line = 0;
-brkint -imaxbel

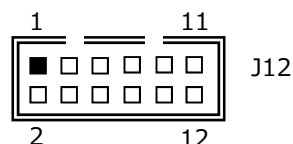
# cat /dev/ttymx2<Enter> [ttymx2 から文字列を受信]
# echo hello > /dev/ttymx2<Enter> [ttymx2 へ文字列"hello"を送信]
```

2-2-6 D-I/O コネクタ(J12)

12Pin 2.00mm Pitch Box Header です。
入力ポートには、8.2K Ω で **+3.3V** にプルアップされています。
出力ポートには DTC113ZUA を使用し、オープンコレクタ出力です。

D-I/O ポート コネクタ&ピン配列

No	信号名	No	信号名
1	NC	2	NC
3	Din A0	4	Dout A0
5	Din A1	6	Dout A1
7	Din A2	8	Dout A2
9	Din A3	10	Dout A3
11	GND	12	GND



適合コネクタ

ハウジング : DF11-12DS-2C HIROSE
コンタクト : DF11-22SCA (金メッキ) HIROSE
適合電線 : AWG26~AWG28

<Linux>

```
//使用するポート宣言
# echo "40" > /sys/class/gpio/export <Enter>
# echo "41" > /sys/class/gpio/export <Enter>
# echo "42" > /sys/class/gpio/export <Enter>
# echo "43" > /sys/class/gpio/export <Enter>
# echo "44" > /sys/class/gpio/export <Enter>
# echo "45" > /sys/class/gpio/export <Enter>
# echo "46" > /sys/class/gpio/export <Enter>
# echo "47" > /sys/class/gpio/export <Enter>

# ls /sys/class/gpio/gpio4* <Enter>
gpio40 gpio41 gpio42 gpio43 gpio44 gpio45 gpio46 gpio47
```

//出力方向の指定

```
# echo "out" > /sys/class/gpio/gpio40/direction <Enter>
# echo "out" > /sys/class/gpio/gpio41/direction <Enter>
# echo "out" > /sys/class/gpio/gpio42/direction <Enter>
# echo "out" > /sys/class/gpio/gpio43/direction <Enter>
```

//入力方向の指定

```
# echo "in" > /sys/class/gpio/gpio44/direction <Enter>
# echo "in" > /sys/class/gpio/gpio45/direction <Enter>
# echo "in" > /sys/class/gpio/gpio46/direction <Enter>
# echo "in" > /sys/class/gpio/gpio47/direction <Enter>
```

// 値の出力

```
# echo 1 > /sys/class/gpio/gpio40/value <Enter>    [オープンコレクタ出力 ON]
# echo 1 > /sys/class/gpio/gpio41/value <Enter>    [オープンコレクタ出力 ON]
# echo 1 > /sys/class/gpio/gpio42/value <Enter>    [オープンコレクタ出力 ON]
# echo 1 > /sys/class/gpio/gpio43/value <Enter>    [オープンコレクタ出力 ON]
# echo 0 > /sys/class/gpio/gpio40/value <Enter>    [オープンコレクタ出力 OFF]
# echo 0 > /sys/class/gpio/gpio41/value <Enter>    [オープンコレクタ出力 OFF]
# echo 0 > /sys/class/gpio/gpio42/value <Enter>    [オープンコレクタ出力 OFF]
# echo 0 > /sys/class/gpio/gpio43/value <Enter>    [オープンコレクタ出力 OFF]
```

//値の入力

```
# cat /sys/class/gpio/gpio44/value <Enter>    [入力 Low]
0
# cat /sys/class/gpio/gpio45/value <Enter>    [入力 Low]
0
# cat /sys/class/gpio/gpio46/value <Enter>    [入力 Low]
0
# cat /sys/class/gpio/gpio47/value <Enter>    [入力 Low]
0
# cat /sys/class/gpio/gpio44/value <Enter>    [入力 High]
4096
# cat /sys/class/gpio/gpio45/value <Enter>    [入力 High]
8192
# cat /sys/class/gpio/gpio46/value <Enter>    [入力 High]
16384
# cat /sys/class/gpio/gpio47/value <Enter>    [入力 High]
32768
```

//使用するポート宣言の解除

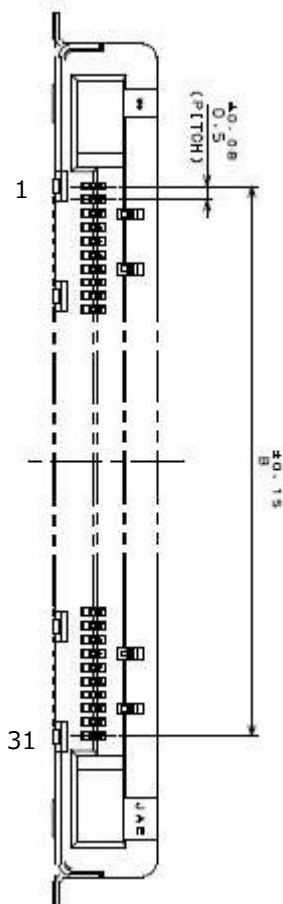
```
# echo "40" > /sys/class/gpio/unexport <Enter>
# echo "41" > /sys/class/gpio/unexport <Enter>
# echo "42" > /sys/class/gpio/unexport <Enter>
# echo "43" > /sys/class/gpio/unexport <Enter>
# echo "44" > /sys/class/gpio/unexport <Enter>
# echo "45" > /sys/class/gpio/unexport <Enter>
# echo "46" > /sys/class/gpio/unexport <Enter>
# echo "47" > /sys/class/gpio/unexport <Enter>
```

2-2-7 LVDS0 コネクタ(J7)

LVDS0 の LCD ケーブル接続用コネクタです。AM-3031G(5V)は LVDS+5V の出力がありますが、AM-3031G(5-12V)は LVDS+5V の出力はありません。J7,J10 両方で LVDS+3.3V 360mA 最大 です。

LVDS0 コネクタ J7&ピン配列

No	信号名
1	NC
2	LVDS+5V
3	LVDS+5V
4	GND
5	LVDS+5V
6	LVDS+5V
7	GND
8	LVDS+5V
9	LVDS+5V
10	GND
11	LVDS0 Tx3+
12	LVDS0 Tx3-
13	GND
14	LVDS0 CLK+
15	LVDS0 CLK-
16	GND
17	LVDS0 Tx2+
18	LVDS0 Tx2-
19	GND
20	LVDS0 Tx1+
21	LVDS0 Tx1-
22	GND
23	LVDS0 Tx0+
24	LVDS0 Tx0-
25	NC
26	NC
27	NC
28	NC
29	LVDS +3.3V
30	LVDS +3.3V
31	GND



基板側コネクタ

FI-RE31S-HF-R1500 (JAE)

適合コネクタ

ケーブル側圧着タイプ

圧着ハウジング FI-RE31HL

圧着ハウジング用上段コンタクト FI-RC3-1A-15000

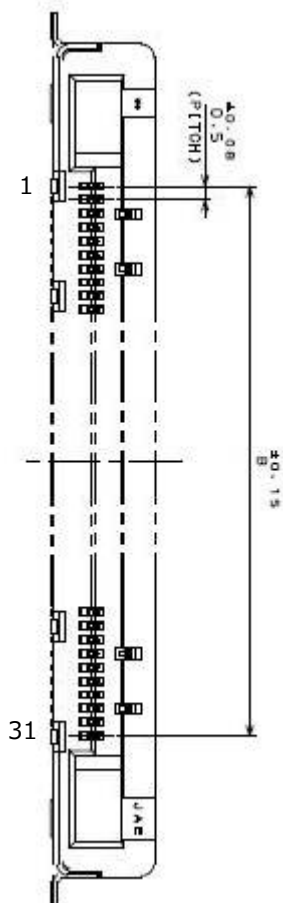
圧着ハウジング用下段コンタクト FI-RC3-1B-15000

2-2-8 LVDS1 コネクタ(J10)

LVDS1 の LCD ケーブル接続用コネクタです。**AM-3031G(5V)**は **LVDS+5V** の出力がありますが、**AM-3031G(5-12V)**は **LVDS+5V** の出力はありません。J7,J10 両方で LVDS+3.3V 360mA 最大 です。

LVDS1 コネクタ J10&ピン配列

No	信号名
1	NC
2	LVDS+5V
3	LVDS+5V
4	GND
5	LVDS+5V
6	LVDS+5V
7	GND
8	LVDS+5V
9	LVDS+5V
10	GND
11	LVDS1 Tx3+
12	LVDS1 Tx3-
13	GND
14	LVDS1 CLK+
15	LVDS1 CLK-
16	GND
17	LVDS1 Tx2+
18	LVDS1 Tx2-
19	GND
20	LVDS1 Tx1+
21	LVDS1Tx1-
22	GND
23	LVDS1 Tx0+
24	LVDS1 Tx0-
25	NC
26	NC
27	NC
28	NC
29	LVDS +3.3V
30	LVDS +3.3V
31	GND



基板側コネクタ
FI-RE31S-HF-R1500 (JAE)

適合コネクタ

ケーブル側圧着タイプ

圧着ハウジング FI-RE31HL

圧着ハウジング用上段コンタクト FI-RC3-1A-15000

圧着ハウジング用下段コンタクト FI-RC3-1B-15000

2-2-9 BK0 コネクタ(J6)

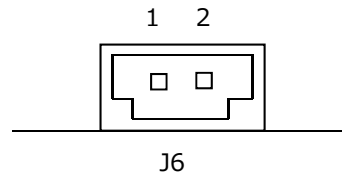
LVDS0 の LCD バックライト接続用コネクタです。

適合 LED バックライトの仕様は 9.6V/320mA (ILED=20mA/LED)です。

2 pin 1.25mmPitch コネクタです。

ただし、AM-3031G(5V),AM-3031G(5-12V)とも BK0 コネクタ(J6)を使用することができます。

No.	信号
1	LED-
2	LED+



適合コネクタ

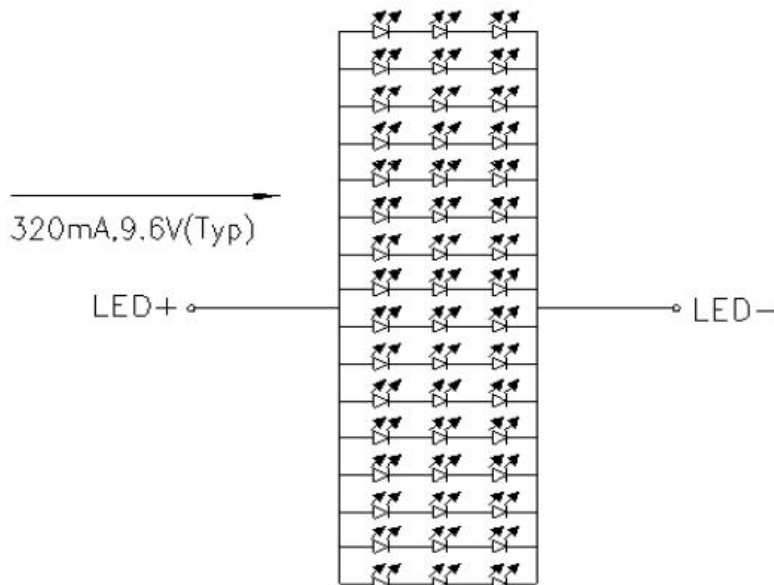
ハウジング : 0511460200

Molex

コンタクト : 0506418141 (金メッキ)

Molex

適合電線 : AWG28~AWG30



<Linux>

/sys/class/backlight/backlight.24/bl_power

LVDS0 backlight OFF

```
# echo 1 > /sys/class/backlight/backlight.24/bl_power<Enter>
```

```
# cat /sys/class/backlight/backlight.24/bl_power<Enter>
```

1

LVDS0 backlight ON

```
# echo 0 > /sys/class/backlight/backlight.24/bl_power<Enter>
```

```
# cat /sys/class/backlight/backlight.24/bl_power<Enter>
```

0

2-2-10 BK1 コネクタ(J13)

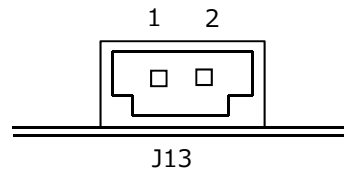
LVDS1 の LCD バックライト接続用コネクタです。

適合 LED バックライトの仕様は 9.6V/320mA (ILED=20mA/LED)です。

2 pin 1.25mmPitch コネクタです

**ただし、AM-3031G(5-12V)は BK1 コネクタ(J13)を使用することができますが、
AM-3031G(5V)では BK1 コネクタ (J13) を使用することはできません。**

No.	信号
1	LED-
2	LED+



適合コネクタ

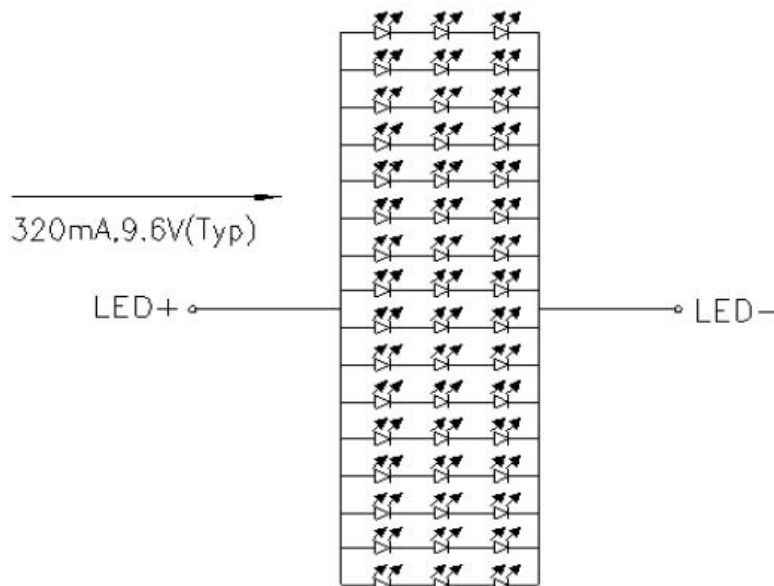
ハウジング : 0511460200

Molex

コンタクト : 0506418141 (金メッキ)

Molex

適合電線 : AWG28~AWG30



<linux>

```
/sys/class/backlight/backlight.25/bl_power
```

LVDS1 backlight OFF

```
# echo 1 > /sys/class/backlight/backlight.25/bl_power<Enter>
```

```
# cat /sys/class/backlight/backlight.25/bl_power<Enter>
```

1

LVDS1 backlight ON

```
# echo 0 > /sys/class/backlight/backlight.25/bl_power<Enter>
```

```
# cat /sys/class/backlight/backlight.25/bl_power<Enter>
```

0

2-2-11 TP コネクタ(J15)

タッチパネル接続用コネクタです。

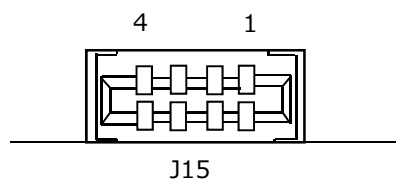
4Pin 1.0mm Pitch Non-ZIF コネクタです。

タッチパネルコントローラには TI 製 TSC2007 を搭載しており、

4 線式抵抗膜方式のタッチパネルに対応しています。

TSC2007 は /dev/i2c-1 に接続されており、Slave address は<0x48>です。

No.	信号
1	TP-X1 (RIGHT)
2	TP-Y1 (TOP)
3	TP-X2 (LEFT)
4	TP-Y2 (BOTTOM)



基板側コネクタ 04FMS-1.0SP-TF(LF)(SN) JST

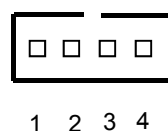
<Linux>

Desktop [Calibrate Touchscreen] Icon をクリックして、タッチパネルのキャリブレーション実行

2-2-12 Speaker コネクタ (J1)

8Ω 1W クラスのスピーカ(R,L)を接続するための 4Pin 2mm Pitch コネクタです。

No.	信号
1	SPKOUT-L-N
2	SPKOUT-L-P
3	SPKOUT-R-N
4	SPKOUT-R-P



適合コネクタ

ハウジング : PHR-4 JST

コンタクト : SPH-002T-P0.5S JST

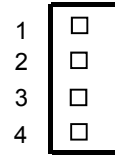
適合電線 : AWG24~AWG30

2-2-13 I2C コネクタ(J401)

外部 I2C デバイス(3.3V)に接続するための 4Pin 2mm Pitch コネクタです。Standard mode で動作します。

内部 **I2C-2**(3.3V)と外部 I2C(3.3V)の間は I2C-bus repeater (PCA9157ADP)を経由して接続されます。

No.	信号
1	SCL-3.3V
2	SDA-3.3V
3	3.3V
4	GND



適合コネクタ

ハウジング : PHR-4 JST

コンタクト : SPH-002T-P0.5S JST

適合電線 : AWG24~AWG30

<Linux>

NXP 製 Remote 8-bit I/O expander PCF8574 を接続した場合の例を示します。

Slave address は <0x20>に設定しています。

```
# i2cdetect 2 <Enter>
```

WARNING! This program can confuse your I2C bus, cause data loss and worse!

I will probe file /dev/i2c-2.

I will probe address range 0x03-0x77.

Continue? [Y/n] y

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
00:				--	--	--	--	--	--	--	--	--	--	--	--	--
10:	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
20:	20-	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
30:	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
40:	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
50:	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
60:	--	--	--	--	--	--	--	--	UU	--	--	--	--	--	--	--
70:	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

```
#i2cset -y 2 0x20 0x00 0x55<Enter> (/dev/i2c-2 の Slave address 0x20 にデータ 0x55 を書き込む)
```

```
#i2cget -y 2 0x20<Enter> (/dev/i2c-2 の Slave address 0x20 のデータを読み込む)
```

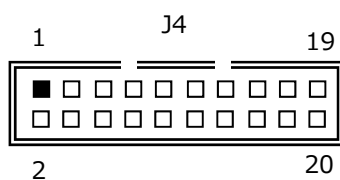
0x55

2-2-14 JTAG コネクタ(J4)

JTAG 接続用コネクタです。

20Pin 2 列 2.00mm Pitch コネクタです。

No	信号名	No	信号名
1	JTAG VREF	2	+3.3V
3	JTAG nTRST	4	GND
5	JTAG TDI	6	GND
7	JTAG TMS	8	GND
9	JTAG TCK	10	GND
11	JTAG RTCK	12	GND
13	JTAG TDO	14	GND
15	JTAG nSRST	16	GND
17	JTAG DE	18	GND
19	JTAG DACK	20	GND



適合コネクタ

ハウジング : DF11-20DS-2C HIROSE
コンタクト : DF11-22SCA (金メッキ) HIROSE
適合電線 : AWG26~AWG28

2-2-15 Mini PCIe Slot

Mini PCIe Slot は ハーフサイズの Mini PCIe モジュールを装着するためのスロットです。

Mini PCIe Slot のピン配列と接続状況

No	信号名	接続	No	信号名	接続
1	WAKE#	WAKE#	2	3.3V_PCIe	3.3V
3	COEX1	No Connect	4	GND	GND
5	COEX2	No Connect	6	1.5V_VGEN1	1.5V
7	CLKREQ#	No Connect	8	USIM_PWR	No Connect
9	GND	GND	10	USIM_DATA	No Connect
11	REFCLK-	REFCLK-	12	USIM_CLK	No Connect
13	REFCLK+	REFCLK+	14	USIM_RESET	No Connect
15	GND	GND	16	USIM_VPP	No Connect
17	Reserved/UIM_C8	No Connect	18	GND	GND
19	Reserved/UIM_C4	No Connect	20	W_DISABLE#	W_DISABLE#
21	GND	GND	22	PERST#	PERST#
23	PERn0/SATA+B	PERn0	24	3.3V_PCIe	3.3V
25	PERp0/SATA-B	PERp0	26	GND	GND
27	GND	GND	28	1.5V_VGEN1	1.5V
29	GND	GND	30	SMB_CLK	SMB_CLK
31	PETn0/SATA-A	PETn0	32	SMB_DATA	SMB_DATA
33	PETp0/SATA+A	PETp0	34	GND	GND
35	GND	GND	36	USB_D-	No Connect
37	GND	GND	38	USB_D+	No Connect
39	3.3V_PCIe	3.3V	40	GND	GND
41	3.3V_PCIe	3.3V	42	LED_WWAN#	No Connect
43	GND	GND	44	LED_WLAN#	LED_WLAN#
45	Reserved	No Connect	46	LED_WPAN#	LED_WPAN#
47	Reserved	No Connect	48	1.5V_VGEN1	1.5V
49	Reserved	No Connect	50	GND	GND
51	Reserved	No Connect	52	3.3V_PCIe	3.3V

注.本ボードの Mini PCIe slot は、USB が接続されていませんので、無線 LAN + Bluetooth モジュールでは Bluetooth 機能は使用できません。

LED1: WPAN (Wireless Personal Area Network)

LED2: WLAN (Wireless Local Area Network)

2-2-16 RTC バックアップ

SoC 内の RTC は使用せず、外部 RTC(TI 製 BQ32000DR)を搭載しています。

この RTC は、ボード上のリチウム電池によりバックアップされています。

この RTC は **/dev/ic2-3** に接続されており、Slave address は <0x68>です。

また、使用済み電池は地域の環境規則に従って処理する必要があります。

電池が消耗した場合の動作については保証対象外とさせていただきます。

電池タイプ	コイン型リチウム電池
型式	BR2032
電池容量	200mAh
消費電流	1.5uA / Typ
計算寿命	$200\text{mA} \div 1.5\text{uA} \div 24\text{h} \div 365\text{d} \approx 15\text{y}$
(計算上の寿命です。保証寿命では有りません。)	

<Linux>

```
# date -s "2015/12/1 14:50:00" <Enter>      システム RTC での日時の設定
Wed Dec  2 14:50:00 UTC 2015
# hwclock -w <Enter>   システム RTC を外部 RTC に書き込む
# hwclock -r <Enter>   外部 RTC の日時を表示
Wed Dec  2 14:50 2015 0.000000 seconds
```

注. date コマンドでの日時の設定は、

1970 年 1 月 1 日 00:00 を基準にした値をシステム・タイマーに書き込むのみです。

外部 RTC に日時を設定には hwclock -w を使います。

2-2-17 SW1～SW5

SW1 : VOL-UP (P1 は SW1 と同じ機能を外部スイッチに接続するためのポスト)
SW2 : VOL-DOWN (P2 は SW2 と同じ機能を外部スイッチに接続するためのポスト)
SW3: RESET (P3 は SW3 と同じ機能を外部スイッチに接続するためのポスト)
SW4: SHUTDOWN (P6 は SW4 と同じ機能を外部スイッチに接続するためのポスト)
SW5: POWER ON (P5 は SW5 と同じ機能を外部スイッチに接続するためのポスト)

注) 外部スイッチはモーメンタリ・スイッチを使用します。

適合コネクタ

ハウジング : RE-02 JST
コンタクト : RF-SC2210(LF)(SN) (金メッキ) JST
適合電線 : AWG24～AWG30

P1,P2,P3 を一つのコネクタで外部に取り出す場合の適合コネクタ

ハウジング : RF-06 JST
コンタクト : RF-SC2210(LF)(SN) (金メッキ) JST
適合電線 : AWG24～AWG30

<Linux>

下記コマンドを実行して GPIO キーのイベントを確認できる。

```
# cat /sys/class/input/event*/device/name<Enter>
```

TSC2007 Touchscreen
ELECOM USB mouse with wheel
HID 0566:3029
HID 0566:3029
WM8962 Beep Generator
gpio-keys.18

⇒表示された上から順に event0, event1, …に該当する

event0	TSC2007 Touchscreen
event1	ELECOM USB mouse with wheel
event2	HID 0566:3029
event3	HID 0566:3029
event4	WM8962 Beep Generator
event5	gpio-key.18

・下記コマンドの XX に「gpio-keys.18」に該当する event 番号を指定して実行し、下記ボタンを押してターミナルにログが表示されることを確認できる。

```
# hexdump /dev/input/eventXX <enter>
```

- ・ボリュームアップボタン

- ・ボリュームダウンボタン

```
# hexdump /dev/input/event5<Return>
```

<ボリュームアップボタン **SW1** を押すと以下を表示>

```
000000 f821 55de 8e31 000e 0001 0073 0001 0000
000010 f821 55da 8e31 000e 0000 0000 0000 0000
000020 f822 55de d292 0000 0001 0073 0000 0000
000030 f822 55de d292 0000 0000 0000 0000 0000
```

<ボリュームダウンボタン **SW2** を押すと以下を表示>

```
000040 f8da 55de 0b07 0002 0001 0072 0001 0000
000050 f8da 55de 06ae 0002 0000 0000 0000 0000
000060 f8da 55de 06ae 0004 0001 0072 0000 0000
000070 f8da 55de 06ae 0004 0000 0000 0000 0000
```

```
# hexdump /dev/input/event5<Return>
```

<POWER ON ボタン **SW5**>を押すと以下を表示>

```
000000 fbba 55de 592b 0002 0001 0074 0001 0000
000010 fbba 55de 592b 0002 0000 0000 0000 0000
000020 fbba 55de 54cf 0004 0001 0074 0000 0000
000030 fbba 55de 54cf 0004 0000 0000 0000 0000
```

- ・Linux OS 起動中に

下記ボタンを押して OS 再起動が行われる。

- ・S/W リセットボタン SW3

- ・H/W リセットボタン SW5

一度起動した後、シャットダウン状態にある時

- ・S/W リセットボタン SW3 は反応なし。

- ・H/W リセットボタン SW5 で OS 再起動が行われる。

- ・POWER ON ボタン SW4 を長押し(4 秒以上)して電源が切れる。

再度 POWER ON ボタン SW4 を押して電源が入る。

- ・OS 起動中、POWER ON ボタン SW4 を押してもシャットダウン動作はなし。