

Toradex TEZI使用マニュアル

本マニュアルは岡本無線電機株式会社が独自作成したものでありメーカーが保証した内容ではありません。万が一本マニュアルに間違いがあり、事故が生じたとしても岡本無線電機株式会社は一切の責任を問われないものとさせていただきます。

変更履歴

バージョン	更新日付	変更内容
1.00	2020/09/07	新規作成

本マニュアルについて

本マニュアルはトラデックスのCPUモジュールにOSイメージを書き込むために使用するTEZI(Toradex Easy Installer)の使用方法について記述しています。

本マニュアルに記載されていないTEZIに関する情報も多々あります。詳しくはToradexのWEBサイトをご参照ください。

参考:

<https://developer.toradex.com/software/toradex-easy-installer#os-and-demo-images>

1. 実行環境

本マニュアルの実行環境は下記です。

OS: Windows 10 1909

TEZI: 2.0b5

CPUモジュール: Colibri-iMX7D 512MB V1.1C

キャリアボード: Colibri 評価ボード Rev 3.2A+ アクセサリーキット

液晶: 感圧式7インチタッチパネル付き液晶(EDT 7.0)

BSP: Linux BSP v3.0.4

本マニュアルとは異なるモジュールや評価ボード以外のキャリアボードを使われても大雑把には同じ操作となります。

2. 注意点

TEZIはパスワードなしでシリアルコンソールを使用できますまたセキュリティ皆無のVNCが入っています。本番環境では安全なネットワークに接続してください。

TEZIは多くのオープンソースで作成されています。TEZIファイルシステムの/usr/share/common-licenses/配下にあるライセンス規約をご参照ください。/usr/share/common-licenses/license.manifestにライセンスの概要があります。

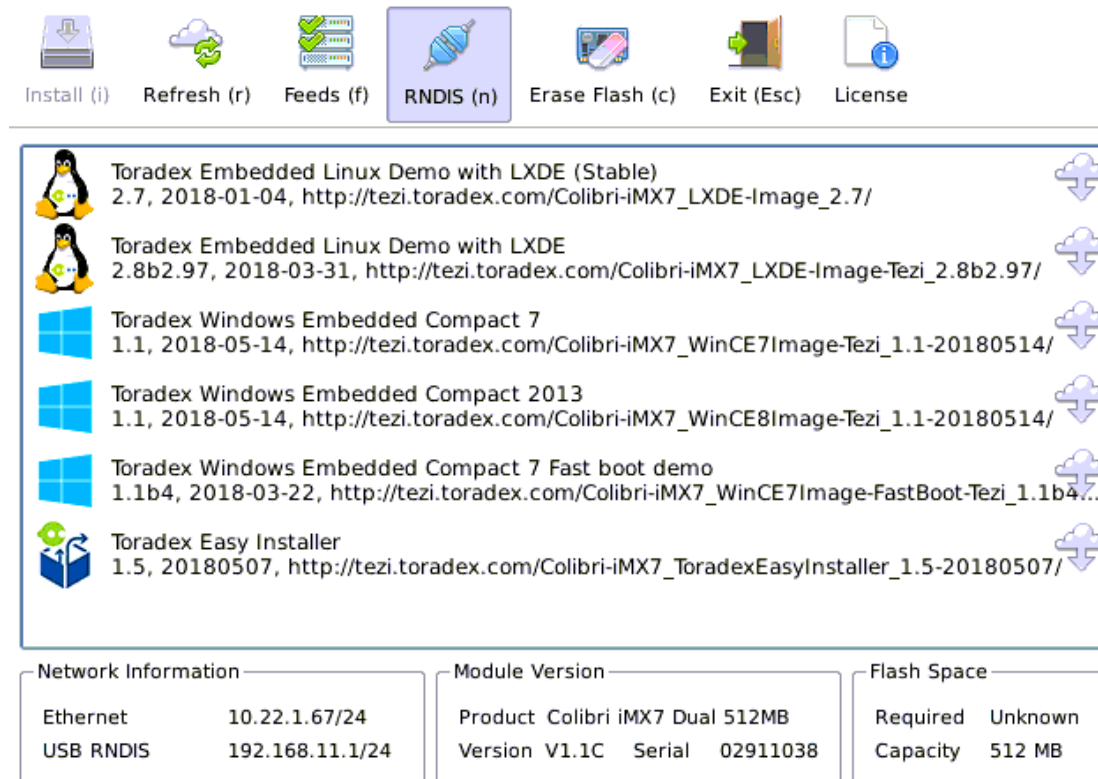
TEZIについて

TEZI(Toradex Easy Installer)はToradexのSOMの内部ROMにOSイメージを書き込むためのツールです。TEZIはLinux + アプリケーションで出来ています。SOM上で動作するものですのでパソコンなどがなくてもモジュール単体で書き込むことが可能です。SDやUSBメモリなどに格納したOSイメージや任意のサーバーからダウンロードしたものを書き込むこともできます。

TEZIが起動するとGUIが立ち上がり数クリックするだけで書き込みできるため誰でも利用することができます。

設定次第では電源を入れるだけで書き込みを自動で行うことも可能です。

SOMは出荷時にTEZIが書き込まれます。そのため量産工程で出荷時にインストールされているTEZIを利用して最終製品用のOSイメージを書き込みを行うこともできます。



Toradex Easy Installer 1.5 (gf4e21de) - Built: May 7 2018

TEZIの起動

TEZIを起動するには3つの方法があります。

1. TEZIがインストールされているモジュールを起動する。

購入直後はTEZIがインストールされているため電源評価ボードに電源を入れるだけでTEZIが起動します。

ただしTEZIでOSイメージを書き込むと内部のTEZIは消えてしまいますのでそれ以降はTEZIは起動せず書き込んだものが起動します。主にモジュール購入直後の評価開始のタイミングや量産時に利用します。

2. リカバリーモードで起動

リカバリーモードは通常起動とは別に内部ROMを使用せずに直接外部からRAMへプログラムを展開して起動するモードです。

主にブートローダーが潰れるなどモジュールが復帰できなくなった時に利用します。TEZIの起動も可能です。

3. Distro Bootで起動

U-BootにはDistro Bootという機能があります。SDカードなど内部ROMとは別の領域に格納されたOSイメージを起動させる機能です。SDカードなどにTEZIを起動させるファイルを格納してDistro Boot機能でTEZIを起動させることができます。

TEZIとOSイメージの入手

OSを書き込むにはTEZIとOSイメージ(TEZIフォーマット)が必要です。

・TEZIの入手

TEZIは下記からダウンロードできます。対象のモジュールのファイルをダウンロードしてください。

<https://developer.toradex.com/software/toradex-easy-installer#load-toradex-easy-installer>

・OSイメージの入手

下記にデモ用のOSイメージがあります。OSイメージをSDカードなどに格納してください。

<https://developer.toradex.com/software/toradex-easy-installer#os-and-demo-images>

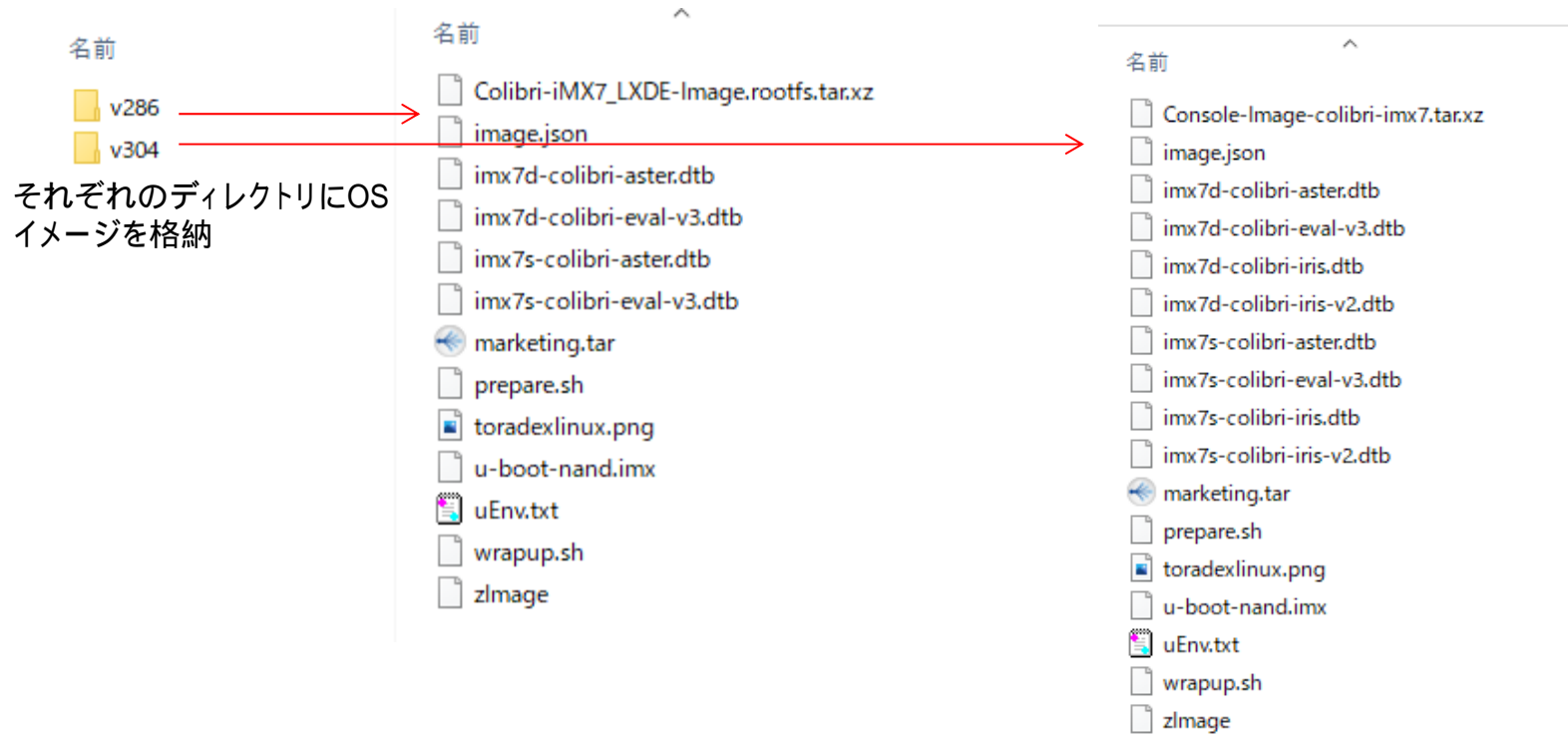
OSイメージをToradexのサーバーから書き込む場合はダウンロード不要です。書き込み時に自動でダウンロードされます。
自作カスタムOSを書き込む場合は作成したOSイメージを使用してください。

リカバリモードでTEZIを使用する方法

OSイメージ解凍後をFAT32フォーマットのSDカードやUSBメモリに格納してください。

ルートディレクトリに任意の名前でディレクトリを作成してその中に格納してください。ディレクトリを分けて複数のOSイメージを格納することもできます。例えばSDカードがFドライブであれば F:\v286、F:\v304配下など。

ToradexのサーバーからOSイメージを書き込む場合はSDカードは準備不要です。



ハードウェアの準備

以下を行います。

評価ボードにモジュールを挿します。

X24にVGAケーブルを指してディスプレイと繋がます。 オプションのX34に接続するLCDでも代用可能です。

X31かX32にUSBマウスを挿入します。

X30にUSBケーブルを差し込みパソコンと接続します。

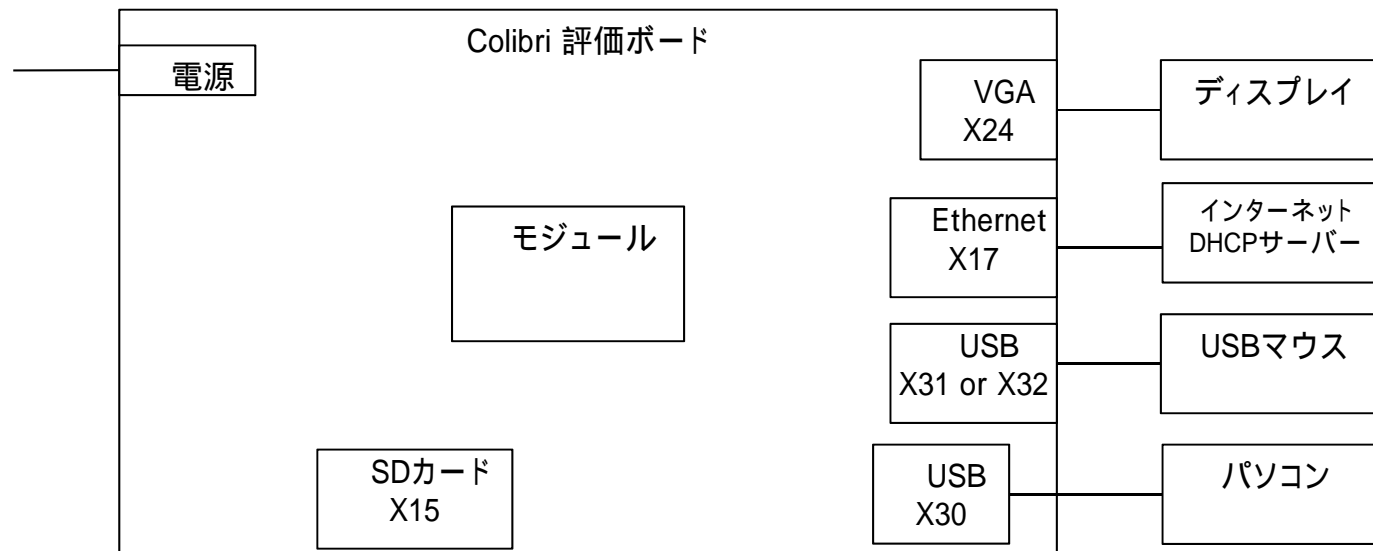
SDカードでアップデートする場合はSDカードを挿入します。

i.MX6系のモジュールの場合はSDカードを挿した状態ではリカバリーモードへ入れません。TEZI起動後に挿入します。

アップデートをインターネット越しで行う場合はX17にEthernetケーブルを挿入します。

インターネットに接続できる環境が必要です。モジュールはDHCPで接続を行いますのでDHCPサーバーが必要です。

評価ボードに電源を挿入します。



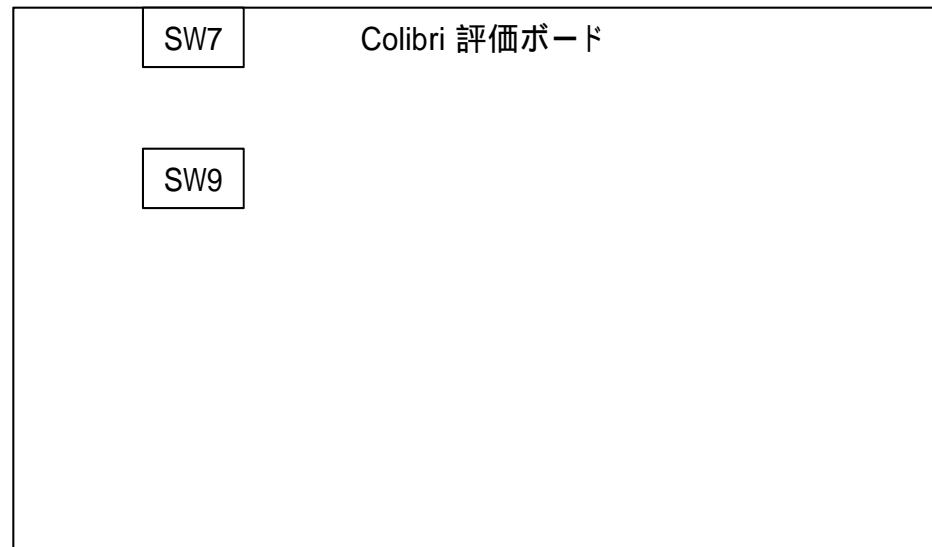
リカバリーモードで起動

Recovery Modeボタン(SW9)を押しながら電源ボタン(SW7)を押してください。

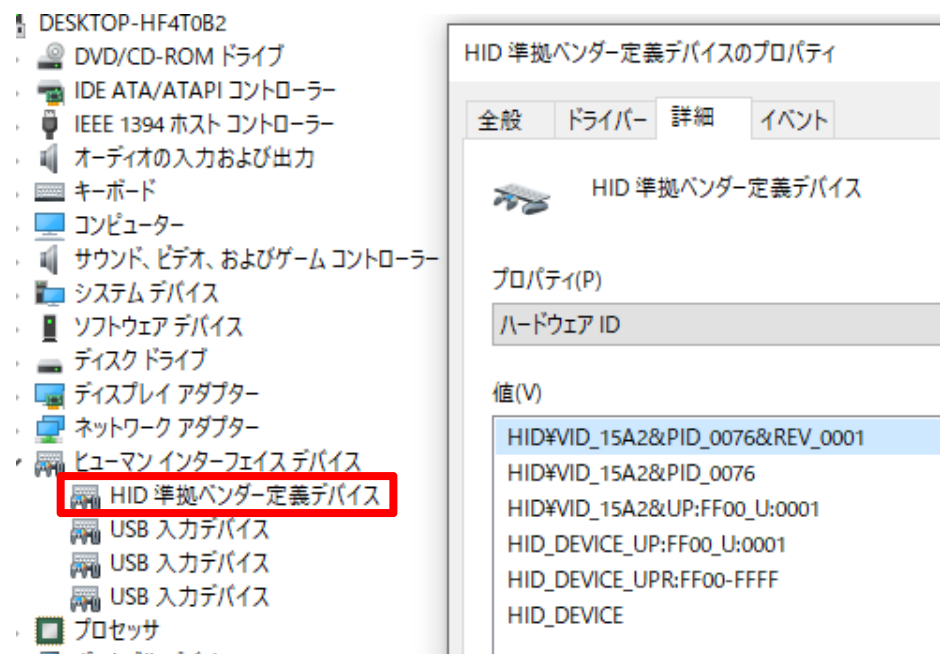
電源ボタンを離した後少しおいてRecovery Modeボタンを離してください。リカバリーモードで起動します。

Colibri-i.MX7以外のモジュールのリカバリーモードへの遷移方法は下記をご参照ください。

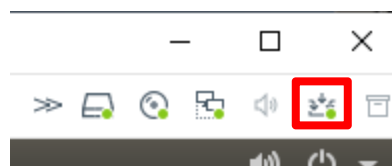
https://developer.toradex.com/knowledge-base/imx-recovery-mode#Enter_Recovery_Mode



リカバリモードで起動するとブートローダーが動作しないためteratermなどに起動ログが出力されません。
リカバリモードに入ったかどうかを確認する場合はWindows PCのデバイスマネージャを見ます。
下記のようなベンダーID,プロダクトIDのHID準拠デバイスが追加されています。



VMWAREが起動している場合が下記のようなアイコンが増えるのでそれで確認することも可能です。



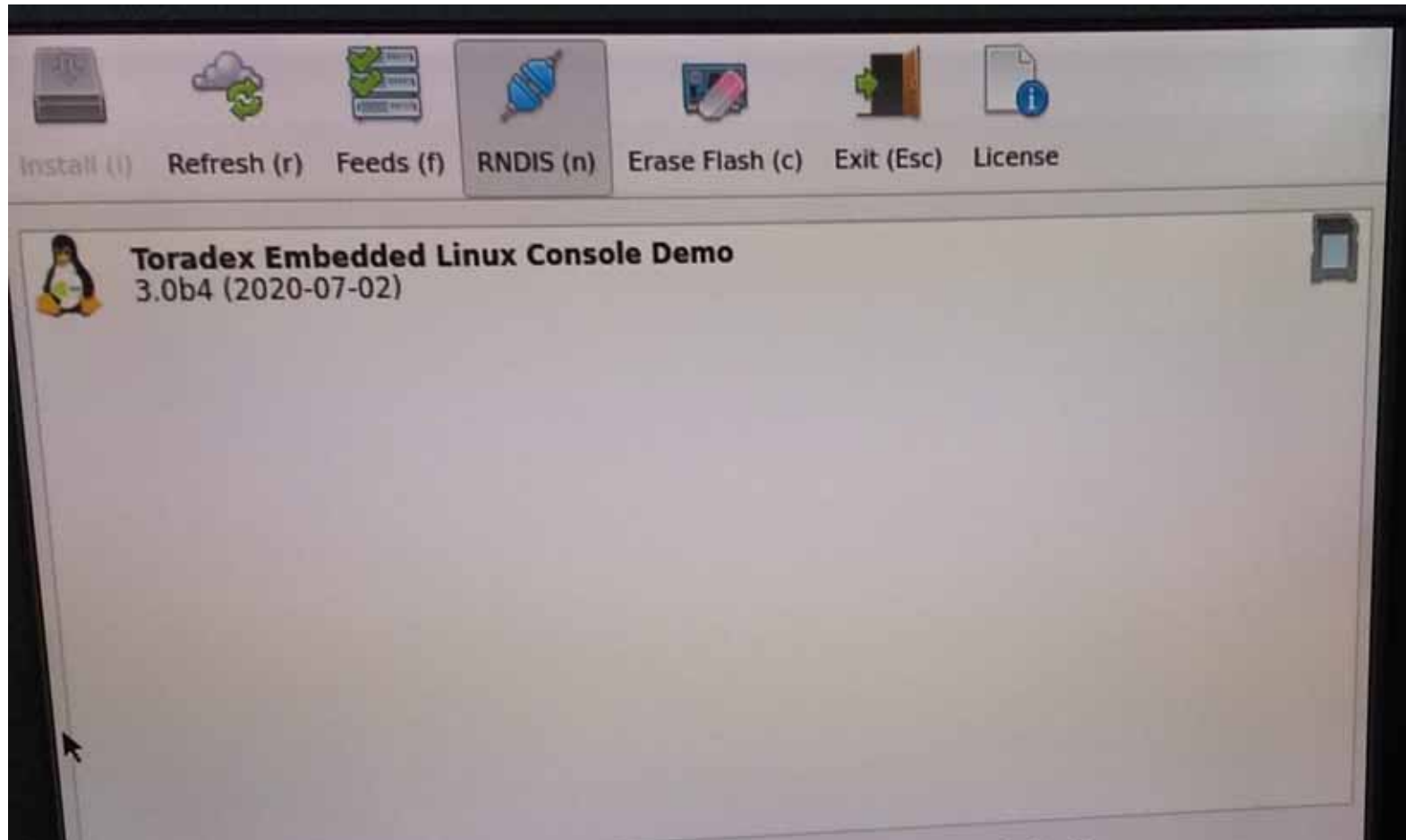
リカバリーモードで起動したらTEZIを起動します。ダウンロードしたTEZIの中にあるrecovery-windows.batを実行します。下記のようなログが出るとTEZIの起動成功です。

```
C:\Windows\system32\cmd.exe
jump_mode 0
jump_addr 0x00000000
== end work item
load_addr=82100000

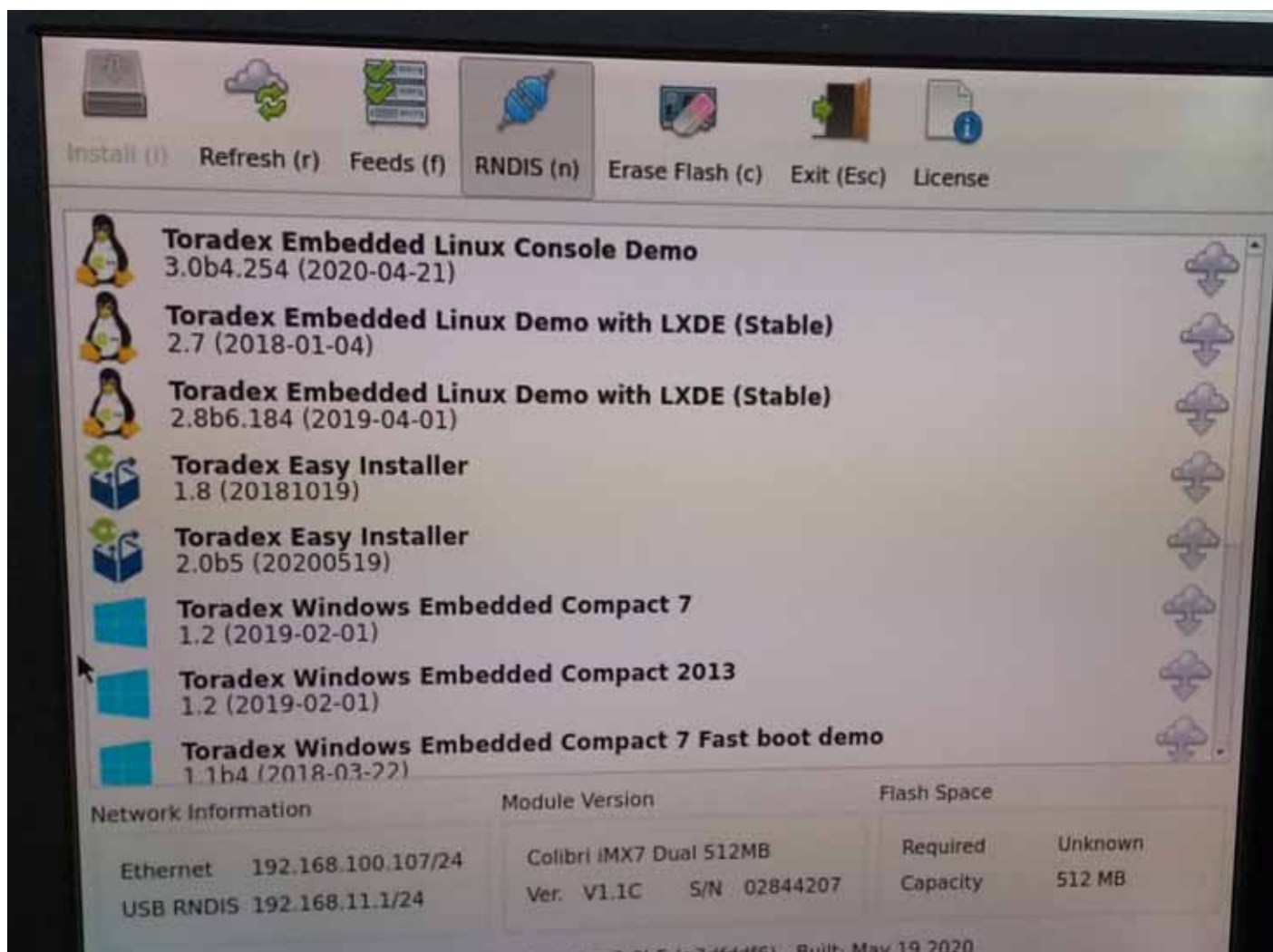
loading binary file(tezi.itb) to 82100000, skip=0, fsize=1d93b64 type=0
<<<31013732, 31013888 bytes>>>
succeeded (status 0x88888888)
HAB security state: development mode (0x56787856)
== work item
filename boot-sdp.scr
load_size 0 bytes
load_addr 0x82000000
dcd 0
clear_dcd 0
plug 0
jump_mode 1
jump_addr 0x82000000
== end work item
load_addr=82000000

loading binary file(boot-sdp.scr) to 82000000, skip=0, fsize=160 type=aa
<<<352, 1024 bytes>>>
succeeded (status 0x88888888)
jumping to 0x82000000
ECHO は <OFF> です。
Successfully downloaded Toradex Easy Installer.
続行するには何かキーを押してください...
```

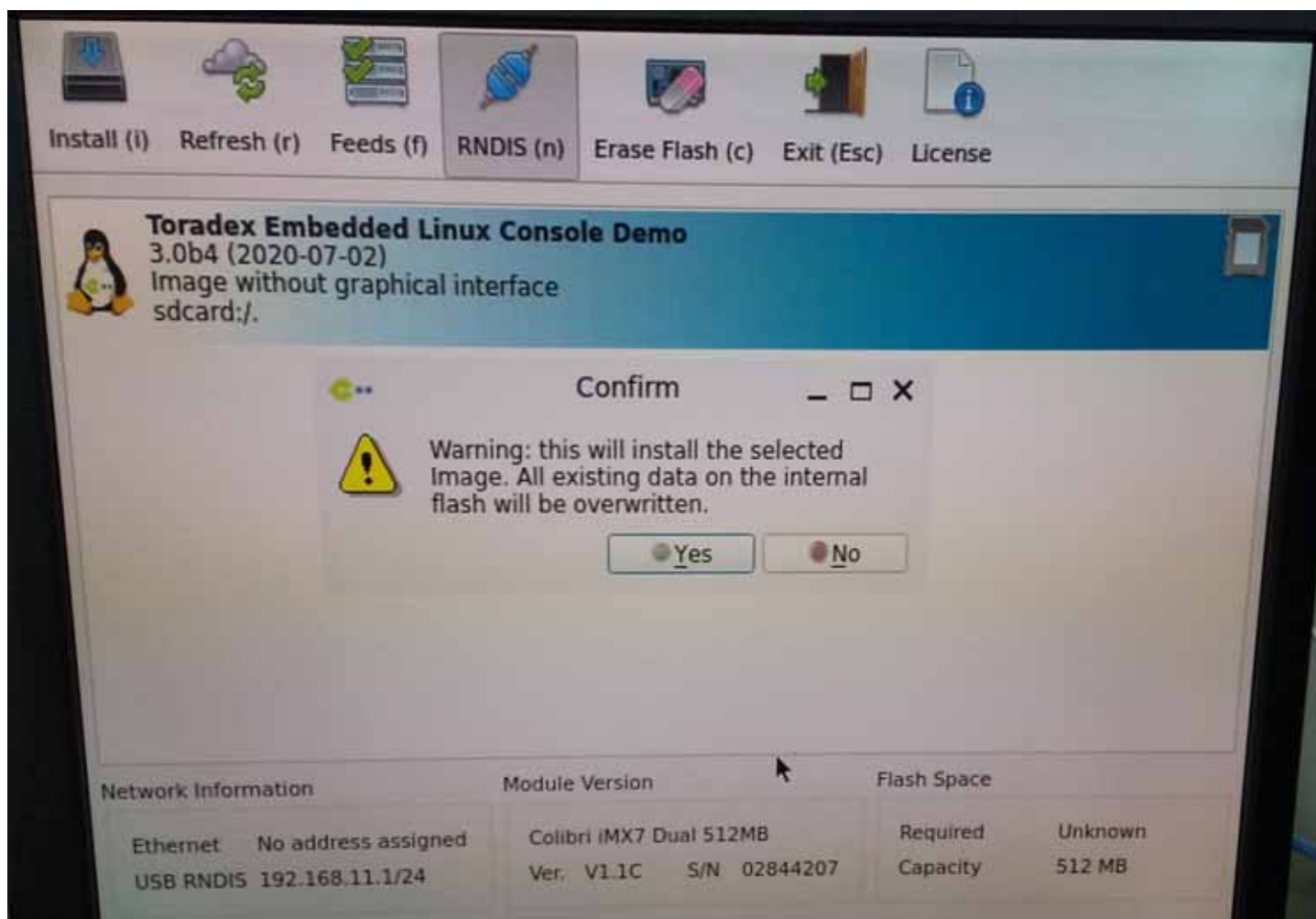
ディスプレイに下記のような画面が表示されます。SDカードに入れたOSイメージがリストされています。複数入れている場合は複数表示されます。



Ethernetでアップデートする場合は下記のようにいくつかの項目がリストされます。
この項目はサーバーへ接続する時期により内容が変わります。



リストされている項目をダブルクリックすると下記のような確認がでできます。Yesをクリックするとインストールが始まります。



インストールが終了すると下記のような選択が表示されます。Rebootをクリックして書き込んだOSが起動するかどうかを確認します。



DistroBootでTEZIを使用する方法

Colibri-iMX7のLinux BSP3.0.4では不具合によりdistrobootが動きません。ほかのモジュールやBSPのバージョンなどでは動作します。

不具合を修正するパッチを適用してU-Bootをbitbakeし直すことでDistobootを使用することができます。

パッチをOpen Embeddedの該当ディレクトリにダウンロード

```
[ubuntu]$ wget -O /work/oe-core/layers/meta-toradex-nxp/recipes-bsp/u-boot/files/colibri-imx7/0001-FIT.patch "https://git.toradex.com/cgit/u-boot-toradex.git/patch/?id=b90d461f8277f03a498644d1252a061d245633ed"
```

bbappendにパッチ追加

```
[ubuntu]$ gedit /work/oe-core/layers/meta-toradex-nxp/recipes-bsp/u-boot/u-boot-toradex_2019.07.bbappend
```

内容

```
SRC_URI_append_mx7 = " file://0001-FIT.patch"
```

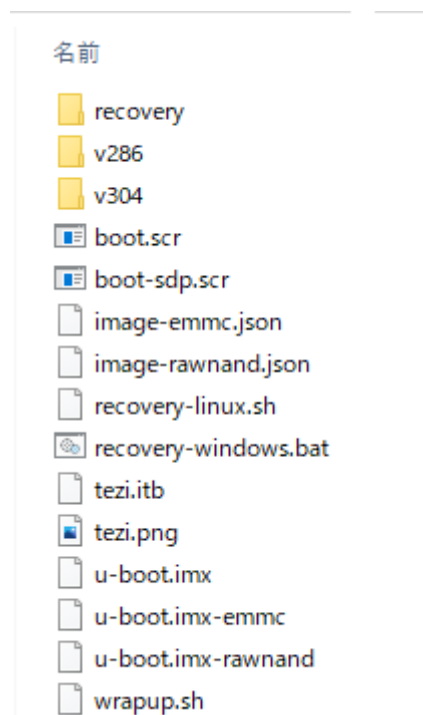
bitbake

```
[ubuntu]$ bitbake -c cleansstate virtual/bootloader && bitbake -c cleansstate console-tdx-image && bitbake console-tdx-image
```

できたOSイメージを書き込んでください。

TEZIを解凍してFAT32フォーマットのSDカードやUSBメモリのルートディレクトリに格納してください。
OSイメージ(TEZIフォーマット)も同様に解凍してリカバリーモード時同様に任意のディレクトリに格納してください。
ToradexのサーバーからOSイメージを書き込む場合はOSイメージの準備は不要です。

TEZIとOSイメージは各々を別のリムーバブルメディアに格納することもできます。



ハードウェアの準備

以下を行います。

評価ボードにモジュールを挿します。

X24にVGAケーブルを指してディスプレイと繋がます。 オプションのX34に接続するLCDでも代用可能です。

X31かX32にUSBマウスを挿入します。

RS232Cケーブルを評価ボードのUART A(X25下側)に挿入します。(パソコンと接続)

RS232Cの代わりにX27にUSBケーブルを挿入してもよいです。(FTDIのドライバが必要です。)

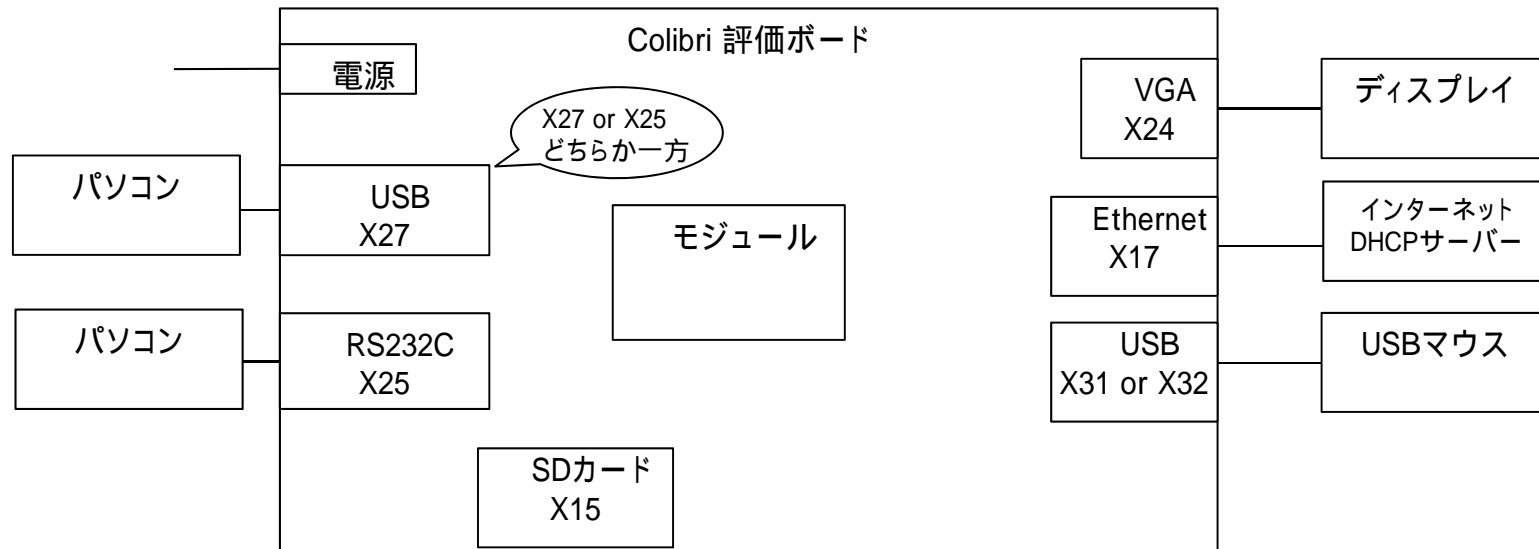
その場合JP17,19,20,21をそれぞれ2番ピンと3番ピンをショートさせてください。

SDカードでアップデートする場合はSDカードを挿入します。

アップデートをインターネット越しで行う場合はX17にEthernetケーブルを挿入します。

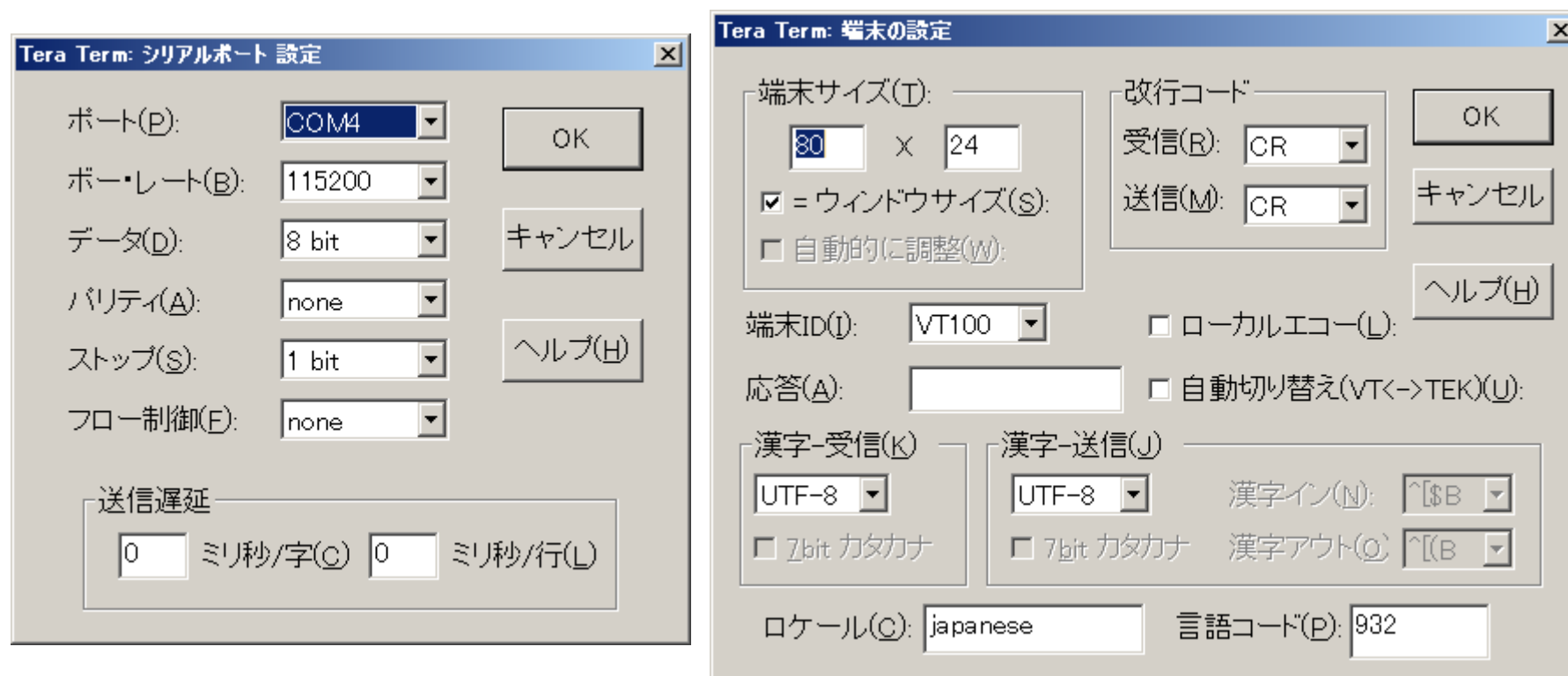
インターネットに接続できる環境が必要です。モジュールDHCPで接続を行いますのでDHCPサーバーが必要です。

評価ボードに電源を挿入します。



シリアル通信ソフトの準備

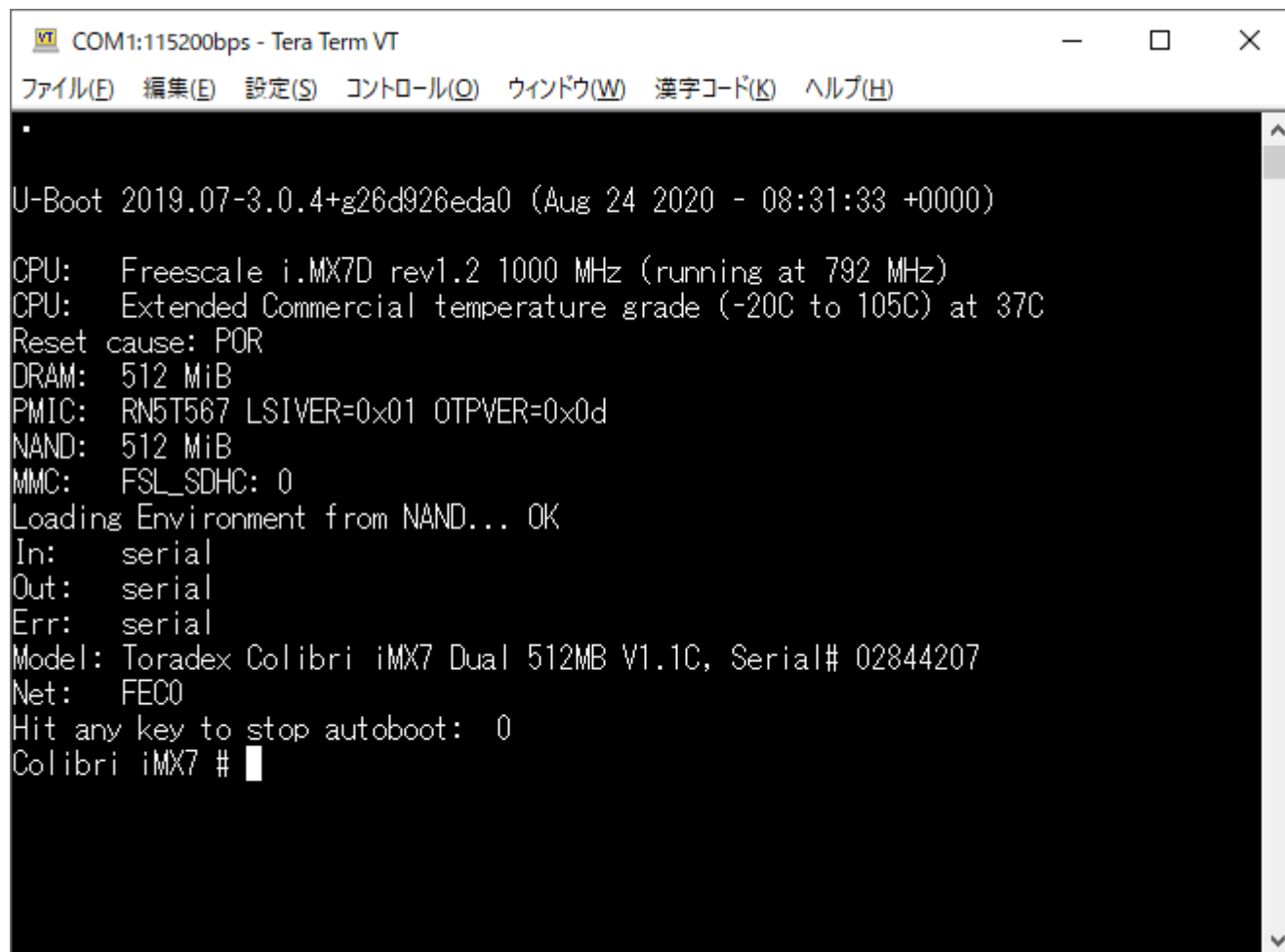
モジュールとシリアル通信を行います。本マニュアルではTeraTermを使います。
TeraTermを下記、設定でCOMポートを開きます。COMポート番号は任意です。



ブートローダーの起動

電源ボタン(SW7)を押し電源ONすると下記のように「Hit any key to stop autoboot:」と出力されます。

この時に何かしらのキーを入力するとブートローダー(U-Boot)で止まりますが入力しないとU-BootがLinuxを起動させます。(書き込まれている場合) 今回はブートローダーの機能を使ってDistroBootをするため何かしらのキーを入力しブートローダーで止めます。

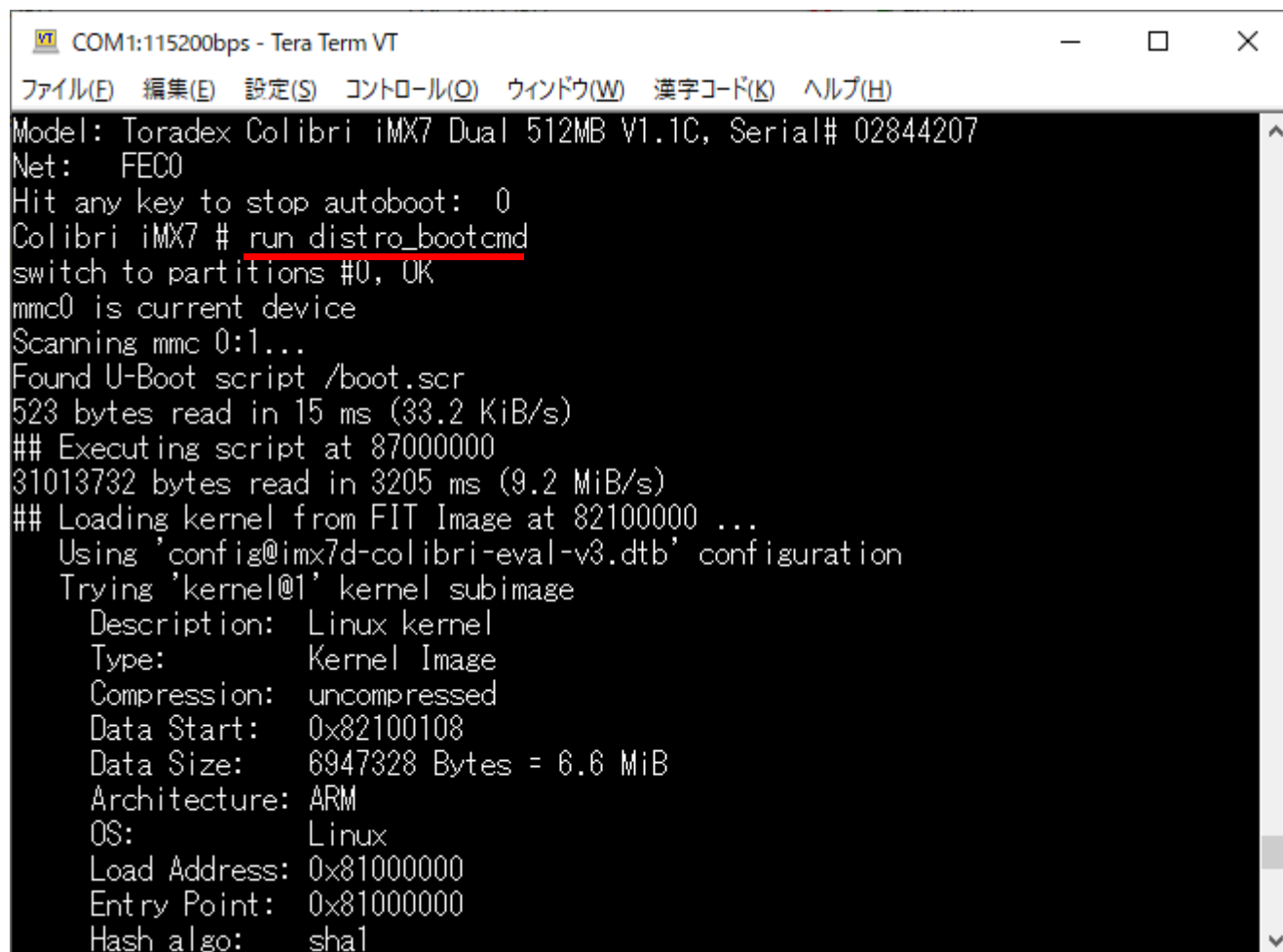


```
COM1:115200bps - Tera Term VT
ファイル(F) 編集(E) 設定(S) コントロール(O) ウィンドウ(W) 漢字コード(K) ヘルプ(H)

U-Boot 2019.07-3.0.4+g26d926eda0 (Aug 24 2020 - 08:31:33 +0000)

CPU: Freescale i.MX7D rev1.2 1000 MHz (running at 792 MHz)
CPU: Extended Commercial temperature grade (-20C to 105C) at 37C
Reset cause: POR
DRAM: 512 MiB
PMIC: RN5T567 LSIVER=0x01 OTPVER=0x0d
NAND: 512 MiB
MMC: FSL_SDHC: 0
Loading Environment from NAND... OK
In: serial
Out: serial
Err: serial
Model: Toradex Colibri iMX7 Dual 512MB V1.1C, Serial# 02844207
Net: FEC0
Hit any key to stop autoboot: 0
Colibri iMX7 #
```

run distro_bootcmdと入力するとdistroboot機能でSDカード内に格納されたTEZIが起動します。
あとはリカバリモード時と同様にOSイメージの書き込みを行ってください。

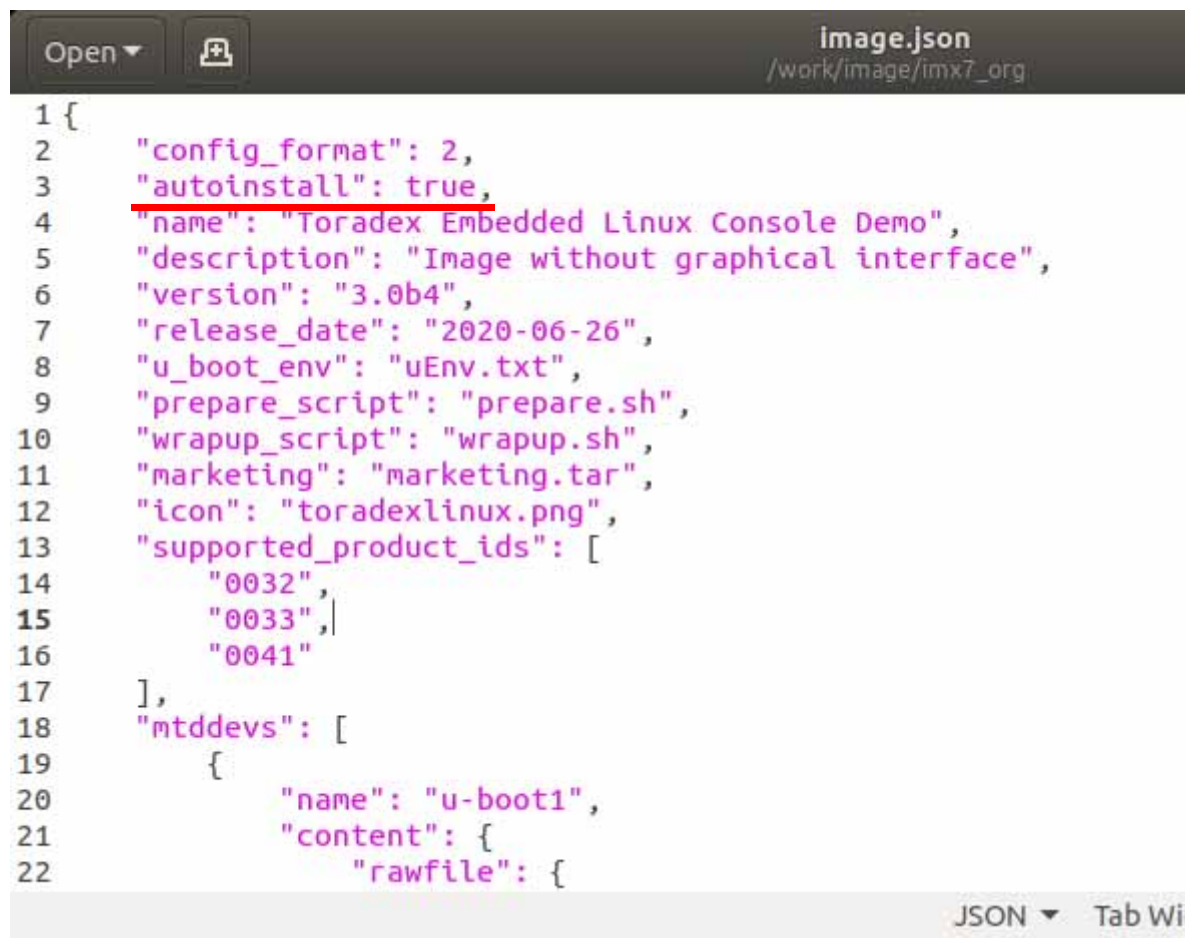


```
COM1:115200bps - Tera Term VT
ファイル(F) 編集(E) 設定(S) コントロール(Q) ウィンドウ(W) 漢字コード(K) ヘルプ(H)
Model: Toradex Colibri iMX7 Dual 512MB V1.1C, Serial# 02844207
Net: FEC0
Hit any key to stop autoboot: 0
Colibri iMX7 # run distro_bootcmd
switch to partitions #0, OK
mmc0 is current device
Scanning mmc 0:1...
Found U-Boot script /boot.scr
523 bytes read in 15 ms (33.2 KiB/s)
## Executing script at 87000000
31013732 bytes read in 3205 ms (9.2 MiB/s)
## Loading kernel from FIT Image at 82100000 ...
Using 'config@imx7d-colibri-eval-v3.dtb' configuration
Trying 'kernel@1' kernel subimage
Description: Linux kernel
Type: Kernel Image
Compression: uncompressed
Data Start: 0x82100108
Data Size: 6947328 Bytes = 6.6 MiB
Architecture: ARM
OS: Linux
Load Address: 0x81000000
Entry Point: 0x81000000
Hash algo: sha1
```

自動インストール機能

OSイメージの中にimage.jsonというファイルがあります。この中にautoinstallという設定があります。この設定をtrueに変更すると自動でインストールするようになります。TEZIが起動した後リムーバルメディアをスキャンしてautoinstallがtrueがあればそのままインストールされます。この設定を使用すると量産工程などでGUIなどを操作することなく電源ONからインストールまでが自動で行われます。複数OSイメージがある場合はどれがインストールされるかは保証されていないためひとつだけtrueに設定してください。

TEZIで扱うテキストファイルはgeditで編集することをお勧めします。改行コードは¥nで処理されます。¥rが含まれると誤動作します。



```
1 {
2   "config_format": 2,
3   "autoinstall": true,
4   "name": "Toradex Embedded Linux Console Demo",
5   "description": "Image without graphical interface",
6   "version": "3.0b4",
7   "release_date": "2020-06-26",
8   "u_boot_env": "uEnv.txt",
9   "prepare_script": "prepare.sh",
10  "wrapup_script": "wrapup.sh",
11  "marketing": "marketing.tar",
12  "icon": "toradexlinux.png",
13  "supported_product_ids": [
14    "0032",
15    "0033",
16    "0041"
17  ],
18  "mtddevs": [
19    {
20      "name": "u-boot1",
21      "content": {
22        "rawfile": {
```

TEZIの自動アップデート機能

OSイメージのBSPのバージョンが上がると古いTEZIでは書き込めなくなる場合があります。(config_formatの設定に依存します。)Toradexのモジュールを購入したときにインストールされているTEZIのバージョンでは書き込めない場合があります。この問題を解決するためにTEZIの自動アップデート機能があります。TEZIのimage.jsonにはisinstallerという設定がありtrueになっています。(Colibri-iMX7にはROMがNANDタイプとMMCタイプがあり、それぞれimage-rawnand.jsonとimage-mmc.jsonに分かれています。該当するタイプの方をimage.jsonに書き換えてください。)

isinstallerがtrueかつautoinstallがtrueの場合、OSイメージよりも優先してTEZIの書き込みが行われます。実行中のTEZIのバージョン(versionの設定)と比較してリムーバブルメディアにあるTEZIのバージョンが新しければ最初にTEZIをインストールして再起動が行われます。

再起動後は新しいTEZIが立ち上がりバージョンが同じになるため再度TEZIがインストールされることはありません。次にOSイメージのインストールが行われます。

```


1 {
2   "config_format": 1,
3   "autoinstall": true,
4   "name": "Toradex Easy Installer",
5   "description": "Toradex Easy Installer for colibri-imx7 machi
6   "version": "2.0b5",
7   "release_date": "20200519",
8   "wrapup_script": "wrapup.sh",
9   "icon": "tezi.png",
10  "isinstaller": true,
11  "supported_product_ids": [
12    "0032",
13    "0033",
14    "0041"
15  ],
16  "mtddevs": [
17    {
18      "name": "u-boot1",
19      "content": {
20        "rawfile": {

```

U-Boot環境変数設定ファイル

image.jsonのu_boot_envに設定されたファイル(デフォルトはuEnv.txt)にU-Bootの環境変数の設定をします。U-Bootのヘッダーを参照して出力されるためU-Bootのソースコードを修正してもそのまま使用できます。U-Bootのソースコードに設定された値はenv defaultで初期化した時に設定される値です。個別で設定を追加したい場合はこのファイルに追加するだけで設定されます。

フラッシュ上に有効なU-Boot環境変数がすでにある場合、新しい変数は追加され、既存の変数は上書きされ、空の変数は削除され、他の変数は変更されません。

A screenshot of a text editor window titled 'uEnv.txt' with a path of '/work/image/Colibri-iMX7_Console-Image-Tezi_3.0b4'. The editor contains a series of environment variable assignments and boot commands for U-Boot. The variables include bootcmd, bootdelay, baudrate, ethprime, ipaddr, serverip, netmask, loadaddr, arch, cpu, board, board_name, vendor, and soc. There are also conditional boot commands for mmc, usb, and ubifs, and a final load command for the efi dtb. The status bar at the bottom indicates 'Plain Text', 'Tab Width: 8', 'Ln 1, Col 1', and 'INS' mode.

```
bootcmd=run ubiboot ; echo ; echo ubiboot failed ; setenv fdtfile ${soc}-colibri-${fdt_board}.dtb && run distro_bootcmd;
bootdelay=1
baudrate=115200
ethprime=FEC
ipaddr=192.168.10.2
serverip=192.168.10.1
netmask=255.255.255.0
loadaddr=0x80800000
arch=arm
cpu=armv7
board=colibri_imx7
board_name=colibri_imx7
vendor=toradex
soc=mx7
mmc_boot=if mmc dev ${devnum}; then devtype=mmc; run scan_dev_for_boot_part; fi
boot_net_usb_start=usb start
usb_boot=usb start; if usb dev ${devnum}; then devtype=usb; run scan_dev_for_boot_part; fi
ubifs_boot=env exists bootubipart || env set bootubipart UBI; env exists bootubivol || env set bootubivol boot; if ubi part ${bootubipart} && ubismount ubi${devnum}:${bootubivol}; then devtype=ubi; run scan_dev_for_boot; fi
boot_efi_binary=if fdt addr ${fdt_addr_r}; then bootefi bootmgr ${fdt_addr_r};else bootefi bootmgr ${fdtcontroladdr};fi;load ${devtype} ${devnum}:${distro_bootpart} ${kernel_addr_r} efi/boot/bootarm.efi; if fdt addr ${fdt_addr_r}; then bootefi ${kernel_addr_r} ${fdt_addr_r};else bootefi ${kernel_addr_r} ${fdtcontroladdr};fi
load_efi_dtb=load ${devtype} ${devnum}:${distro_bootpart} ${fdt_addr_r} $
```


スクリプト実行機能

prepare_script、wrapup_script、error_scriptにシェルフファイルを設定することでそれぞれインストール前、インストール後、エラー発生時に自動でスクリプトを実行することが可能です。このシェルのパラメータは\$1,\$2,\$3,\$4にそれぞれプロダクトID、ボードリビジョン、シリアルナンバー、イメージのフォルダーが入っています。

例えば下記のような記述をするとOSイメージインストール後SDカードにログを出力して再起動します。TEZIはリムーバルメディアを読み込み専用でマウントします。そのままではログの保存ができないため一度、書き込み可能でマウントし直して最後に読み込み専用に戻しています。

TEZIが内部でどのようなことをしているかは/var/volatile/tezi.logにログが出力されています。

下記シェルではtezi.logも保存しています。



```
1 #!/bin/sh
2
3 MOUNT_POINT=/run/media/src
4 mount -o remount,rw $MOUNT_POINT
5
6 LOG_DIR=$MOUNT_POINT/log/$3
7 mkdir -p $LOG_DIR
8 echo "PRODUCT_ID:$1" >> $LOG_DIR/log
9 echo "BOARD_REV:$2" >> $LOG_DIR/log
10 echo "SERIAL:$3" >> $LOG_DIR/log
11 echo "IMAGE_FOLDER:$4" >> $LOG_DIR/log
12 cp -prf /var/volatile/tezi.log $LOG_DIR
13 sync
14 mount -o remount,ro $MOUNT_POINT
15 reboot -f
16
17 exit 0
```

書き込み時間の短縮

書き込み時間はOSイメージの大きさに依存します。できるだけOSイメージから不要な機能を削除することでOSイメージのサイズを小さくすることで書き込み時間を短くすることができます。

OSイメージを出力するとrootfsやbootfsはLZMA2アルゴリズム(xzファイル)で出力されます。SDカードやUSBメモリなど高速に読み込みできる場所にある場合、読み込速度に対して解凍の方が時間がかかるため遅くなります。(インターネット越しでダウンロードする場合は解凍時間に比べてダウンロード速度のほうが遅いため圧縮は効果的です。)

tar.xzを解凍してtarにした状態でインストールすると大幅に早くなります。この場合image.jsonのファイル名をtar.xzからtarに修正してください。xzの解凍は下記のコマンドで行います。

```
xz -d Console-Image-colibri-imx7.tar.xz
```



```
image.json
/work/image/Colibri-iMX7_Console-Image-Tezi_3.0b4

    "product_ids": "0033"
  },
  {
    "filename": "imx7d-colibri-eval-v3.dtb",
    "product_ids": "0041"
  }
],
"size_kib": 128,
"type": "static"
},
{
  "name": "m4firmware",
  "size_kib": 896,
  "type": "static"
},
{
  "name": "rootfs",
  "content": {
    "filesystem_type": "ubifs",
    "filename": "Console-Image-colibri-imx7.tar",
    "uncompressed_size": 181.69140625
  }
}
]
}
```

JSON Tab Width: 8 Ln 93, Col 2 INS

ディスプレイの設定変更

TEZIはオリジナルのソースコードは入手できませんが多少のカスタムは可能です。使用するディスプレイに合わせて設定変更を行いたい場合はデバイスツリーを変更することで対応することができます。

カスタムするためにTEZIをUbuntuに持っていきます。本マニュアルでは/work/tezi配下をワーキングディレクトリとします。

/work/tezi/orgにオリジナルのTEZIを格納します。

下記コマンドでu-boot-toolsをインストールします。

```
sudo apt-get install -y u-boot-tools
```

出力ディレクトリ作成

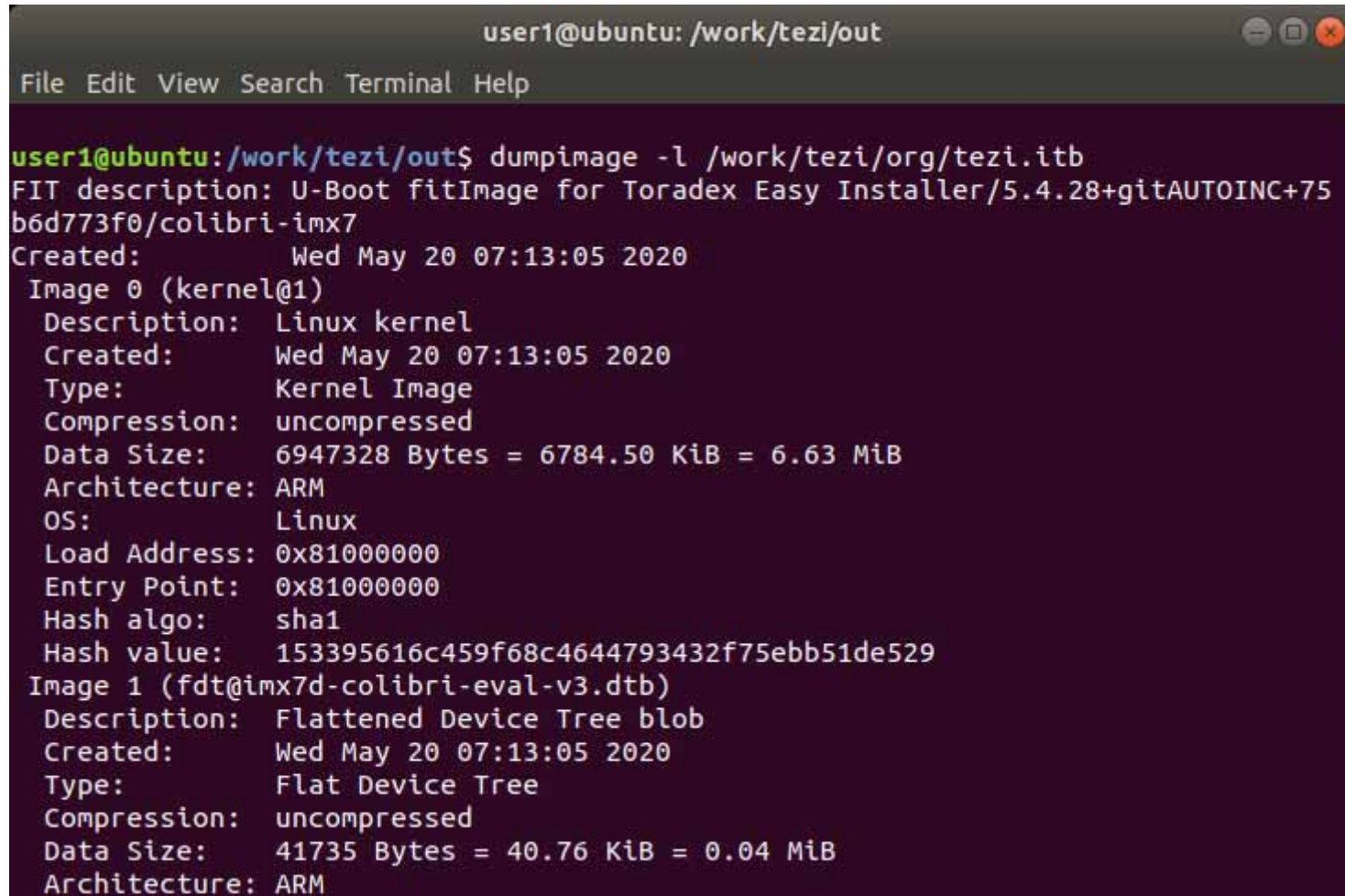
```
mkdir -p /work/tezi/out
```

```
cd /work/tezi/out
```

TEZIはitb形式でできています。itb形式のファイルにはカーネルとデバスタツリーとファイルシステムが含まれます。

下記コマンドでtezi.itbの中身を情報を表示することができます。

```
dumpimage -l /work/tezi/org/tezi.itb
```

A terminal window titled 'user1@ubuntu: /work/tezi/out' with a menu bar (File, Edit, View, Search, Terminal, Help). The terminal shows the command 'dumpimage -l /work/tezi/org/tezi.itb' and its output. The output describes a FIT (Flattened Image Tree) for U-Boot, containing two images: a Linux kernel (Image 0) and a Flattened Device Tree blob (Image 1).

```
user1@ubuntu:/work/tezi/out$ dumpimage -l /work/tezi/org/tezi.itb
FIT description: U-Boot fitImage for Toradex Easy Installer/5.4.28+gitAUTOINC+75
b6d773f0/colibri-imx7
Created:          Wed May 20 07:13:05 2020
Image 0 (kernel@1)
  Description:    Linux kernel
  Created:        Wed May 20 07:13:05 2020
  Type:           Kernel Image
  Compression:    uncompressed
  Data Size:      6947328 Bytes = 6784.50 KiB = 6.63 MiB
  Architecture:   ARM
  OS:             Linux
  Load Address:  0x81000000
  Entry Point:    0x81000000
  Hash algo:      sha1
  Hash value:     153395616c459f68c4644793432f75ebb51de529
Image 1 (fdt@imx7d-colibri-eval-v3.dtb)
  Description:    Flattened Device Tree blob
  Created:        Wed May 20 07:13:05 2020
  Type:           Flat Device Tree
  Compression:    uncompressed
  Data Size:      41735 Bytes = 40.76 KiB = 0.04 MiB
  Architecture:   ARM
```

dumpimageで表示した内容からimage0-4までありそれぞれ

0.Kernel

1.Device Tree (imx7d-colibri-eval-v3.dtb)

2.Device Tree (imx7s-colibri-eval-v3.dtb)

3.Device Tree (imx7d-colibri-mmc-eval-v3.dtb)

4.Filesystem(ramdisk)

とわかります。その下にConfigurationが0-2までありデフォルトはimx7d-colibri-eval-v3.dtbとなっています。

下記のコマンドでimage0をzImageというファイルで出力することができます。

```
dumpimage /work/tezi/org/tezi.itb -T flat_dt -p 0 -o zImage
```

同様に下記コマンドでimage1のDeviceTreeとimage4のFilesystemを取り出します。

```
dumpimage /work/tezi/org/tezi.itb -T flat_dt -p 1 -o imx7d-colibri-eval-v3.dtb
```

```
dumpimage /work/tezi/org/tezi.itb -T flat_dt -p 4 -o ramdisk
```

デバイツリーは使用するモジュールに合わせて取り出すものを変えてください。

取り出したデバイスツリーはimx7d-colibri-eval-v3.dtbを下記コマンドでdtsに変換して修正します。

```
sudo dtc -I dtb -O dts -o ./imx7d-colibri-eval-v3.dts ./imx7d-colibri-eval-v3.dtb
```

```
sudo gedit ./imx7d-colibri-eval-v3.dts
```

TEZIのデバイスツリーはBSP3.0.4のものと大きく異なりますのでご注意ください。

ディスプレイの設定を変更するにはpanel-dpiの中にあるpanel-timingを変更します。

```
panel-dpi {  
    compatible = "panel-dpi";  
    backlight = <0x65>;  
    power-supply = <0x39>;  
    width-mm = <0x73>;  
    height-mm = <0x56>;  
    data-mapping = "bgr666";  
  
    panel-timing {  
        clock-frequency = <0x18023d8>;  
        hactive = <0x280>;  
        vactive = <0x1e0>;  
        hsync-len = <0x60>;  
        hfront-porch = <0x10>;  
        hback-porch = <0x30>;  
        vsync-len = <0x2>;  
        vfront-porch = <0xa>;  
        vback-porch = <0x21>;  
    };  
};
```

EDT 7.0を使用する場合下記の設定になります。修正後ファイルを保存します。

```
panel-dpi {  
    compatible = "panel-dpi";  
    backlight = <0x65>;  
    power-supply = <0x39>;  
    width-mm = <0x73>;  
    height-mm = <0x56>;  
    data-mapping = "bgr666";  
  
    panel-timing {  
        clock-frequency = <0x1fb81e0>;  
        hactive = <0x320>;  
        vactive = <0x1e0>;  
        hback-porch = <0xd8>;  
        hfront-porch = <0x28>;  
        vback-porch = <0x23>;  
        vfront-porch = <0xa>;  
        hsync-len = <0x80>;  
        vsync-len = <0x2>;  
        de-active = <0x1>;  
        hsync-active = <0x0>;  
        vsync-active = <0x0>;  
        pixelclk-active = <0x0>;  
    };  
};
```

dtsを再度dtbに戻します。

```
sudo dtc -I dts -O dtb -o ./imx7d-colibri-eval-v3.dtb ./imx7d-colibri-eval-v3.dts
```

最後にkernelとDevice TreeとFilesystemを結合してitbファイルに戻します。itbにするにはitsファイルという定義ファイルが必要です。下記にサンプルを記述します。

```
----- /dts-v1/;
```

```
/ {  
    description = "U-Boot fitImage for Toradex Easy Installer";  
    #address-cells = <1>;  
  
    images {  
        kernel@1 {  
            description = "Linux Kernel";  
            data = /incbin("./zImage");  
            type = "kernel";  
            arch = "arm";  
            os = "linux";  
            compression = "none";  
            load = <replace with address from dumpimage>;  
            entry = <replace with address from dumpimage>;  
            hash@1 {  
                algo = "sha1";  
            };  
        };  
    };  
};
```



```

fdt@imx7d-colibri-eval-v3.dtb {
    description = "Flattened Device Tree blob";
    data = /incbin/("./imx7d-colibri-eval-v3.dtb");
    type = "flat_dt";
    arch = "arm";
    compression = "none";
    hash@1 {
        algo = "sha1";
    };
};

ramdisk@1 {
    description = "tezi-initramfs";
    data = /incbin/("./ramdisk");
    type = "ramdisk";
    arch = "arm";
    os = "linux";
    compression = "none";
    hash@1 {
        algo = "sha1";
    };
};
};

```

```
configurations {  
    default = "config@imx7d-colibri-eval-v3.dtb";  
    config@imx7d-colibri-eval-v3.dtb {  
        description = "1 Linux kernel, FDT blob, ramdisk";  
        kernel = "kernel@1";  
        fdt = "fdt@imx7d-colibri-eval-v3.dtb";  
        ramdisk = "ramdisk@1";  
        hash@1 {  
            algo = "sha1";  
        };  
    };  
};
```

カーネルの設定にある<replace with address from dumpimage>の部分はdumpimage -l tezi.itbで表示された内容を記述します。

下記の場合

load = <0x81000000>;

entry = <0x81000000>;

になります。

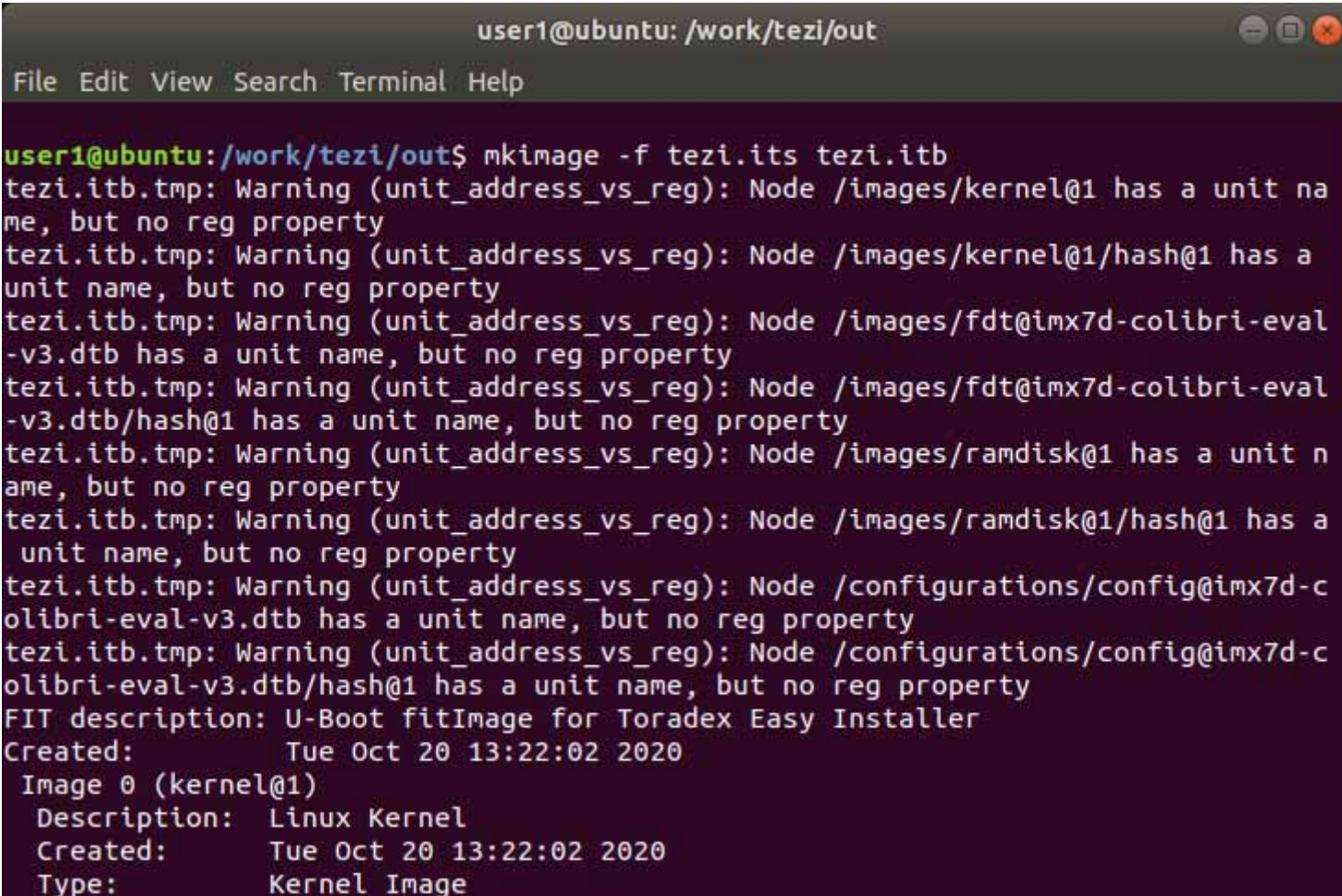
```
user1@ubuntu: /work/tezi/out
File Edit View Search Terminal Help

user1@ubuntu:/work/tezi/out$ dumpimage -l /work/tezi/org/tezi.itb
FIT description: U-Boot fitImage for Toradex Easy Installer/5.4.28+gitAUTOINC+75
b6d773f0/colibri-imx7
Created:      Wed May 20 07:13:05 2020
Image 0 (kernel@1)
  Description: Linux kernel
  Created:    Wed May 20 07:13:05 2020
  Type:      Kernel Image
  Compression: uncompressed
  Data Size: 6947328 Bytes = 6784.50 KiB = 6.63 MiB
  Architecture: ARM
  OS:      Linux
  Load Address: 0x81000000
  Entry Point: 0x81000000
  Hash algo: sha1
  Hash value: 153395616c459f68c4644793432f75ebb51de529
Image 1 (fdt@imx7d-colibri-eval-v3.dtb)
  Description: Flattened Device Tree blob
  Created:    Wed May 20 07:13:05 2020
  Type:      Flat Device Tree
  Compression: uncompressed
  Data Size: 41735 Bytes = 40.76 KiB = 0.04 MiB
  Architecture: ARM
```

itsファイルができたらitbファイルを下記コマンドで作成します。

```
mkimage -f tezi.its tezi.itb
```

作成したtezi.itbで置き換えて起動すると解像度を変更されているのが確認できます。



```
user1@ubuntu: /work/tezi/out
File Edit View Search Terminal Help

user1@ubuntu:/work/tezi/out$ mkimage -f tezi.its tezi.itb
tezi.itb.tmp: Warning (unit_address_vs_reg): Node /images/kernel@1 has a unit name, but no reg property
tezi.itb.tmp: Warning (unit_address_vs_reg): Node /images/kernel@1/hash@1 has a unit name, but no reg property
tezi.itb.tmp: Warning (unit_address_vs_reg): Node /images/fdt@imx7d-colibri-eval-v3.dtb has a unit name, but no reg property
tezi.itb.tmp: Warning (unit_address_vs_reg): Node /images/fdt@imx7d-colibri-eval-v3.dtb/hash@1 has a unit name, but no reg property
tezi.itb.tmp: Warning (unit_address_vs_reg): Node /images/ramdisk@1 has a unit name, but no reg property
tezi.itb.tmp: Warning (unit_address_vs_reg): Node /images/ramdisk@1/hash@1 has a unit name, but no reg property
tezi.itb.tmp: Warning (unit_address_vs_reg): Node /configurations/config@imx7d-colibri-eval-v3.dtb has a unit name, but no reg property
tezi.itb.tmp: Warning (unit_address_vs_reg): Node /configurations/config@imx7d-colibri-eval-v3.dtb/hash@1 has a unit name, but no reg property
FIT description: U-Boot fitImage for Toradex Easy Installer
Created:      Tue Oct 20 13:22:02 2020
Image 0 (kernel@1)
  Description: Linux Kernel
  Created:    Tue Oct 20 13:22:02 2020
  Type:      Kernel Image
```

インストール中の背景の変更

OSイメージのファイル一式の中にmarketing.tarがあります。このファイルはインストール中の背景画像が入っています。この中のファイルを変更することでインストール中の背景画像を変更することができます。

下記コマンドで解凍します。

```
tar xf marketing.tar
```

slides_vgaというディレクトリに下記が入っています。これらの画像が順番に表示されていきます。

Embedded-Linux-Splash-Screen_V3_01.png

Embedded-Linux-Splash-Screen_V3_02.png

Embedded-Linux-Splash-Screen_V3_03.png

Embedded-Linux-Splash-Screen_V3_04.png

Embedded-Linux-Splash-Screen_V3_05.png

使用したい画像をpng形式で格納してください。

再度アーカイブ化してオリジナルに上書きしてください。

```
tar cf ./marketing.tar ./slides_vga/
```

インストール中の背景が変わっていることが確認できます。

Linux起動中からTEZIの起動

下記のコマンドでLinux起動中からU-Bootでコマンドを実行せずにTEZIを実行することができます。

```
fw_setenv bootcmd "run distro_bootcmd"  
reboot
```

fw_setenvはU-Bootの設定領域を書き換えるコマンドです。参照するにはfw_printenvを使います。

上記はfw_setenvコマンドでU-Bootのbootcmdを"run distro_bootcmd"に書き換えています。

bootcmdはU-Bootが起動したときに実行するスクリプトです。通常は内部ROMから起動するスクリプトになっていますがこれをDistro Bootに書き換えています。次にリブートしたときにはDistro Bootが行われて、SDなどにTEZIがあればTEZIが起動されます。

TEZI起動後にアップデートを行えばbootcmdは元に戻りアップデート後には内部ROMから起動します。

TEZIのオートインストール機能とインストール後のシェルでrebootを行えば全自動でアップデートが行われたあとリブートします。

このコマンドを利用することでアプリケーションにも容易にOSアップデート機能を搭載することができます。

OSアップデートを行うと内部ROMのデータは消えてしまいます。ユーザーデータを保持したい場合はSDカードに一度退避してアップデート後に戻すことで保持できます。