

Lineo Warp!!  
for ConnectCore6 SBC 評価キット  
リリースノート  
Version 1.2



Rev.	日付	変更内容
1.0	2015/10/30	初版
1.1	2016/01/25	「2.2.1 拡張コマンド」に clear_bootf の説明を追加 「3.1 eMMC マップ」の修正 「4.3 Snapshot 消去」の追加 「5. 評価キットバイナリの書き込み手順」の修正
1.2	2016/02/10	「1.1.2 ソフトウェア」Android version の修正

#### 注意

- 本ソフトウェアおよびマニュアルの著作権は、リネオソリューションズ株式会社にあります。
- 本ソフトウェアおよびマニュアルの一部または全部を無断で使用、複製することはできません。
- 本ソフトウェアおよびマニュアルは、本製品の使用許諾契約書のもとのみ使用することができます。
- 本ソフトウェアおよびマニュアルを運用した結果に関して、リネオソリューションズ株式会社はいかなる保証も表明も行っていません。
- 本ソフトウェアの仕様およびマニュアルの記載されている内容は、予告なしに変更することがあります。

#### 商標

Lineo Solutions, Inc. はリネオソリューションズ株式会社の商標です。Linux は Linus Torvalds 氏が所有する登録商標です。その他記載されている会社名、製品名は、各社の商標および登録商標です。

## 目 次

1.	はじめに.....	1
1.1	評価キット構成.....	1
1.1.1	ハードウェア.....	1
1.1.2	ソフトウェア.....	1
2.	評価キットの操作方法.....	2
2.1	起動設定.....	2
2.1.1	起動選択.....	2
2.1.2	自動起動選択.....	2
2.2	u-boot の操作方法.....	2
2.2.1	拡張コマンド.....	2
3.	ソフトウェアについて.....	3
3.1	eMMC マップ.....	3
3.2	Warp!!対応済 Kernel Driver.....	3
4.	評価キットの動作について.....	4
4.1	Warp 起動後の動作.....	4
4.2	Snapshot 保存.....	4
4.3	Snapshot 消去.....	4
4.4	モジュールロード.....	4
5.	評価キットバイナリの書き込み手順.....	5
5.1	u-boot の書き込み手順.....	5
5.2	warp_emmc.bin の書き込み手順.....	5



## 1. はじめに

使用許諾書の内容を確認頂き、それに従って評価を実施して下さい。

本ドキュメントは、ConnectCore6SBC のドキュメントとなっております。該当機器の内容をご参照ください。

### 1.1 評価キット構成

#### 1.1.1 ハードウェア

Android 版

評価ボード	ConnectCore6 SBC Early Availability Kit
CPU	Freescale i.MX6 1GHz
SDRAM	1GByte (64bit)
Storage Device	eMMC 4GByte
LCD オプション	LCD Application Kit, including Fusion 10" WSVGA(1024x600)LCD panel with PCAP touch
HDMI	1920x1080 60fps

#### 1.1.2 ソフトウェア

Android 版

Warp	4.0
Boot loader	u-boot 2013.04 + Warp!! patch
Kernel	3.0.35 + Warp!! patch
Android	4.2.2

## 2. 評価キットの操作方法

### 2.1 起動設定

#### 2.1.1 起動選択

J5 により microSD 起動と eMMC 起動を切り替えます。

J5 オープン	eMMC 起動
J5 クローズ	microSD 起動

#### 2.1.2 自動起動選択

Warp 用 u-boot は自動的に warp 起動を行います。自動起動を停止し、u-boot のプロンプトに移行する場合は電源投入後シリアル入力を行ってください。

### 2.2 u-boot の操作方法

#### 2.2.1 拡張コマンド

下記のコマンドを拡張しています。

コマンド	説明
warp	スナップショットを使用して Warp!! 起動します。 使用例： => warp
clear_bootf	スナップショットを消去し、Warp!!起動を行えなくします。 使用例： => clear_bootf

### 3. ソフトウェアについて

#### 3.1 eMMC マップ

Android 版

Part	Start	End	Block	ID	説明
1	16	999999	499992	0x0C	kernel/ uramdisk.img
2	1000000	5999999	2500000	0x83	rootfs
3	6000000	7733247	866614	0xA0	Warp!!使用領域

注意) Warp!!使用領域を書き換えると Warp!!が機能しなくなります。

#### 3.2 Warp!!対応済 Kernel Driver

Warp!!起動後の動作確認がされているデバイスを下記に○印で示します。

Android 版

Device	Android
CPU : i.MX6	○
SDRAM	○
LCD	○
HDMI	○
UART3	○
USB(Host)	○
SD/MMC	○
SDMA	○
PMIC(I2C)	○
Touch Panel	○
SATA	○
Ethernet	○
WDT	○

○ : 対応

## 4. 評価キットの動作について

### 4.1 Warp 起動後の動作

評価キットには Warp!! 起動後に動作制限を設けており、Warp!! 起動から 3 分後にシステムがフリーズします。

### 4.2 Snapshot 保存

通常の Warp!! では任意の起動ポイントにて Snapshot 保存が行えます。評価キットは動作制限を設けており、Snapshot 保存が行えません。

また、通常の Warp!! では microSD カードへの Snapshot 保存もサポートしております。

### 4.3 Snapshot 消去

u-boot プロンプトにて以下のコマンドを実行し snapshot の消去を行います。

```
=> clear_bootf
```

snapshot を消去する事で通常起動を行います。snapshot をクリアした後に Warp!! 起動を行う場合は「5.2 warpemmc.bin の書き込み手順」の内容を実行し、eMMC の再構築を行う必要があります。

### 4.4 モジュールロード

通常の Warp!! ではモジュールロードが行えます。評価キットは動作制限を設けており、モジュールロードが行えません。



## 5. 評価キットバイナリの書き込み手順

バイナリの書き込みには起動用 microSD カードが必要です。起動用 microSD カードは 2GB 以上の microSD カードに bootsd.bin を書き込み作成します。

```
$ dd if=bootsd.bin of=/dev/sdX conv=notrunc
```

### 5.1 u-boot の書き込み手順

作成した microSD の第一パーティションに u-boot-ccimx6sbc-warp.imx を配置します。microSD 起動をし、u-boot プロンプトにて以下のコマンドを実行し書き込みを行います。

```
=> update uboot mmc 1:1 fat u-boot-ccimx6sbc-warp.imx
```

### 5.2 warp\_emmc.bin の書き込み手順

warp\_emmc.bin を USB ストレージに保存し、ターゲットに挿入します。USB ストレージは EXT4 フォーマットをお勧めします。microSD 起動をしカーネルプロンプトまで移行します。USB ストレージをマウント後、カーネルプロンプトにて以下のコマンドを実行し書き込みを行います。

Android 版

```
$ dd if=warp_emmc.bin of=/dev/block/mmcblk0 conv=notrunc; sync; sync
```