

# **Getting Started**

# **Mpression Magnes Board**

Revision 1.0

2018/08/09



© Mpression by Macnica



# 目次

1. はじめにお読みください	_ 4
1.1. 重要事項	. 4
1.2. 開発元	. 5
1.3. お問い合わせ先	. 5
1.4. 免責、及び、ご使用上の注意	. 5
2. 安全上の注意	_ 6
2.1. 凡例	. 6
2.2. 注意事項	. 6
3. はじめに	8
4. セットアップ	9
4.1. リファレンス・デザインの入手	10
4.2. FPGA デザインの書き込み	11
4.2.1. JIC ファイルの書き込み	11
5. 事前準備	15
5.1. Linux のインストール	15
5.1.1. Ubuntu 16.04 LTS インストール・イメージの入手	15
5.1.2. インストール・メディアの作成	15
5.1.3. 各種接続	16
5.1.4. BIOS の設定	18
5.2. Ubuntu 16.04 LTS のインストール	20
6. リファレンス・デザイン	23
6.1. リファレンス・デザインの構成	23
6.1.1. PCIe & Video ブロックの詳細	24
6.1.2. トランシーバ・ブロックの詳細	25
6.2. PCle の BAR マッピング	26
6.3. その他	27
6.3.1. リファレンス・デザイン使用上の注意	27
7. リファレンス・デザインの実行	28
7.1. COMe モジュールの起動	28
7.2. サンプル・ソフトウェアのダウンロード	28
7.3. ドライバとユーザ・アプリケーションのビルド	29
7.4. ドライバのインストール	30
7.5. ユーザ・アプリケーションの実行	31
7.5.1. サポート・コマンド一覧	32
7.5.2. menu コマンド	33



7.5.3. led コマンド	34
7.5.4. memdump コマンド	35
7.5.5. memtest コマンド	36
7.5.6. pcidump コマンド	37
7.5.7. cfgread コマンド	38
7.5.8. barread コマンド	39
7.5.9. barwrite コマンド	40
7.5.10. exit コマンド	41
7.6. ドライバのアンインストール	42
7.7. OS のシャットダウン	42
7.8. トランシーバ・ブロックの確認	43
7.8.1. トランシーバ・ブロックの確認準備	43
7.8.2. トランシーバのコントロール・コンソール	44
7.9. HDMI 出力の確認	45
8. 参考情報4	46
9. 更新履歴4	47



# 1. はじめにお読みください

## 1.1. 重要事項

#### 最初にお読みください:

- 当製品のご使用前には必ず当ハードウェア・ユーザーマニュアルをお読みください
- 当ハードウェア・ユーザーマニュアルは、必要な時に参照できるよう保管してください
- 当製品の構成を十分に理解したうえで使用してください。

#### 当製品の用途:

● 当製品は、インテル社の Arria10 GX を使用したシステムの開発および検証を支援する装置です。当製品を使用して、ソフトウェア、ハードウェアの両面からシステム開発を支援します。 この用途に従って、当製品を正しく使用してください。

#### 当製品をご使用されると想定するお客様:

● 当製品は、本資料および Getting Started をよく読み、理解した人のみがご使用ください。 当製品を使用するうえで、FPGA、論理回路、電子回路、およびマイクロコンピュータの基本的な知識が必要です。

#### 当製品を使用する際の注意事項:

- 当製品は、お客様のプログラム開発および評価段階で使用するための開発支援装置です。お客様の 設計されたプログラムの量産時においては、当ボードをお客様の装置に組み込んで使用することはでき ません。また、開発済みの回路については、必ず統合試験、評価、または実験などにより実使用の可否 をご確認ください。
- 当製品の使用から生ずる一切の結果について、株式会社マクニカ(以降マクニカ)は責任を持ちません。
- マクニカは、本製品不具合に対する回避策の提示または、不具合改修などについて、有償もしくは無償の対応に努めます。ただし、いかなる場合でも回避策の提示または不具合改修を保証するものではありません。
- マクニカは、潜在的に内包されるすべての危険性を評価予期しているわけではありません。したがって 当ボードやハードウェア・ユーザーマニュアル内の警告や注意は、すべての警告や注意を含んでいるわ けではありません。お客様の責任で、本製品を正しく安全に使用して下さい。
- 当製品に搭載されているデバイスに不具合がある場合であっても、デバイスの不具合改修品には交換 しません。
- USB ペリフェラルデバイスのすべてのデバイスの動作を保証するものではありません。
- LAN インタフェース、HDMI 映像出力は、すべての機器との接続を保証するものではありません。
- 当製品の改造又は、お客様による製品の損傷時は、交換対応ができません。
- 当製品は、鉛フリー製品を使用した製品です。
- 当マニュアルに記載のある各ベンダの商標および登録済み商標の権利は、各ベンダに帰属します。

#### 製品改善のポリシー:

マクニカは、製品のデザイン、パフォーマンスおよび安全性に関して製品を常に改善しつづけます。マクニカは、お客様に予告なく、いつでも製品のドキュメント、ハードウェア・ユーザーマニュアル、デザインおよび仕様の一部またはすべてを変更する権利を保有します。



#### 製品の RMA について:

- 製品の納品後、30日以内の初期不良に関しましては無償交換にて対応させていただきます。ただし、
   以下の場合は無償交換の対応ができませんのでご了承ください。
  - (1) 製品の誤使用または、通常使用環境ではない状況での製品の損傷
  - (2) 製品の改造または補修
  - (3) 火災、地震、製品の落下やその他アクシデントによる損傷

#### 図および写真:

● 図や写真は、お手元にある実際の製品とは異なる可能性があります。

## 1.2. 開発元

株式会社マクニカ 〒222-8561 横浜市港北区新横浜 1-6-3

## 1.3. お問い合わせ先

ご購入いただいた販売代理店、もしくは下記 Web のお問い合わせフォームよりお問い合わせ下さい。 Mpression ブランド Web サイト内お問い合わせページ: <u>https://service.macnica.co.jp/contact</u>

## 1.4. 免責、及び、ご使用上の注意

本資料を入手されましたお客様におかれては、下記の使用上の注意を一読頂いたうえでご使用ください。

- 本資料内の図面など無断複製することを禁じます。
- 本資料は予告なく変更することがあります。
- 本資料の作成には万全を期していますが、万一ご不明な点や誤り、記載漏れなどお気づきの点がありましたら、下記までご一報いただければ幸いです。

株式会社マクニカ 戦略技術本部 Mpression 推進部 〒222-8561 横浜市港北区新横浜 1-6-3 http://www.m-pression.com

- 本資料で取り扱っている回路、技術、プログラムに関して運用した結果の影響については責任を負いか ねますので予めご了承ください。
- 製品をご使用になる場合は、各デバイスメーカの最新資料を併せてご参照ください。



# 2. 安全上の注意

ここに示した注意事項は、お使いになる人や、他の人への危害、財産への損害を未然に防ぐための内容を記 載していますので、必ずお守りください。

# 2.1. 凡例

<u> 危険</u> 危険	この表示は、取り扱いを誤った場合、「死亡また重傷を負う危険が切迫して生じること が想定される」内容です。
▲ 警告	この表示は、取り扱いを誤った場合、「死亡または重傷を負う可能性が想定される」 内容です。
1 注意	この表示は、取り扱いを誤った場合、「傷害を負う可能性が想定される場合および物 的損害のみの発生が想定される」内容です。

注、重要は、例外的な条件や注意を操作手順や説明記述の中で、ユーザに伝達する場合に使用しています。

## 2.2. 注意事項

Ń	危険	本マニュアルで指定された仕様に基づく AC アダプタ (梱包品) を使用してください。 指定の仕様を満たさない AC アダプタを使用した場合は、キットの発熱、破裂、発火の原因となります。
		強い衝撃を与えたり、投げつけたりしないでください。 発熱、破裂、発火や機器の故障、火災の原因となります。
		電子レンジなどの加熱調理機器や高圧容器に、本体や AC アダプタを入れないでく ださい。本体や AC アダプタの発熱、破裂、発火、発煙、部品の破壊、変更などの原 因となります。
		使用中の本体を布など熱のこもりやすいもので包んだりしないでください。 熱がこもり、発火、故障の原因となることがあります。
	警告	本体を廃棄する時は、他の一般ゴミと一緒に捨てないで下さい。 火中に投じると破裂する恐れがあります。廃棄方法については、廃棄物に関する各 種法律・法令・条例等に従ってください。
		電源コードを強く引っ張ったり、重いものを乗せたりしないでください。 電源コードを傷つけたり、破損したり、束ねたり、加工したりしないでください。 傷ついた部分から漏電して、火災・感電の原因になります。
		濡れた手で電源プラグを抜き差ししないでください。 感電による怪我や故障の原因になります。
		電源プラグはコンセントの奥までしっかりと差し込んでください。 しっかり差し込まないと、感電や発電による火災の原因となります。
		タコ配線を行ったり、AC アダプタの規格電圧以外の電源に接続したりしないでください。 故障や感電、発熱による火災の原因なります。



Â	<b>警告</b> (前項から継続)	<ul> <li>電源プラグのほこりを定期的に拭き取り、コンセント周辺のたまったほこりを取り除い てください。</li> <li>ほこりがたまったままで使用していると湿気などで、絶縁不良となり、火災の原因になります。</li> <li>電源プラグやコンセプト周辺のほこりは、乾いた布で拭き取ってください。</li> <li>当ボードにコップや花瓶など、水や液体が入った容器を置かないでください。</li> <li>当ボードに水や液体が入ると、故障や感電の原因になります。水などをこぼした場合 は、使用を中止し、電源を切って電源プラグを抜いてください。修理や技術的な相談 は購入元へお問い合わせください。</li> <li>当ボードや付属品は子供の手の届かないところに保管してください。子供の手の届く ところに保管すると、けがの原因になります。</li> </ul>
		ぐらついた台の上や傾いた場所等、不安定な場所には置かないでください。 落下して、けがや故障の原因になります。 直射日光の強い場所や炎天下の車内など高温の場所で使用、放置しないでください。 発熱、破損、発火、暴走、変形、故障の原因になります。また、機器の一部が熱くな
		9、欠傷の原因となる場合もあります。 極端な高温、低温、また温度変化の激しい場所で使用しないで下さい。 故障の原因となります。周囲温度は 5℃ ~ 35℃、湿度は 0% ~ 85%の範囲でご 使用ください。
		本体を組み込んだ装置の保守中は、電源を抜いて作業してください。 感電の危険性があります。
		当ボードに無理な力がかかるような場所に置かないでください。 基板の変形により、基板の破損、部品の脱落、故障の原因となります。
	计音	拡張ボードや他の周辺機器と一緒にお使いの場合には、それぞれ個別の取り扱い 説明書をよく読んで適正にお使いください。 本マニュアルに記載されているもの、また別途動作を確認できていることを公表して いるものの他は、特定の拡張ボードや周辺機器の相互動作は保証いたしかねます。
	江忌	当ボードを移動・接続するときは、電源スイッチを切ってください。 電源をいれたまま移動・接続すると、故障や感電の原因になります。
		ベンジンやシンナーなど化学薬品を含んだ雑巾で手入れしないでください。 当ボードが変質する可能性があります。科学雑巾を使用するときは、その注意書きに 従ってください。
		当ボードを箱から取り出した際、機器本体に結露が発生した場合は、すぐに電源を入 れないでください。 冷えた当ボードを暖かい部屋で箱から取り出すと、結露が発生することがあります。 結露があるまま電源を入れると、当ボードが破損したり、部品の寿命が短くなる場合 があります。 当ボードを取り出したら室温になじませてください。結露が発生した場合は、水滴が蒸 発してから設置や接続を行ってください。
		カスタマイズ可能と明示している部分以外の分解、解体、改変、改造、再生はしない でください。 当ボードはカスタマイズが可能な製品ですが、本マニュアルに指定された部分以外 は基本動作に必要な部分に何らか外部の手が加わることで製品全体の動作保証が 出来なくなります。本マニュアルに記載されているカスタマイズ可能部分以外のカスタ マイズをご希望の場合には、はじめに必ず購入元にご相談ください。



# 3. はじめに

当マニュアルでは、当ボードを使用し COM Express<sup>®</sup>(以下、COMe)コネクタに接続される CPU モジュール 上へ Linux をインストールする手順と COMe 側に実装したデバイスドライバとユーザ・アプリケーションを介 して FPGA に実装したリファレンス・デザインにアクセスする手順について解説します。

以下に当マニュアルで使用する機材を示します。

機材	必要数	備考
Magnes	1 枚	当ボード
AC アダプタ	1 個	当ボードに付属
インテル <sup>®</sup> FPGA ダウンロード・ケ	1 本	FPGA との JTAG 接続に使用
ーブル		(IE USB-Blaster™ / USB-Blaster II™)
COMe CPU モジュール	1 個	COMe-A41-CT6 - SECO COM Express <sup>®</sup> Type 6 Module
OS インストール用メディア	1 セット	USB メモリ / DVD ドライブ
キーボード	1 個	
マウス	1 個	
モニタ	2 枚	HDMI 入力をサポートしているモニタ
		CPU と FPGA 側を差し替えて使用する場合は
		1 枚でも問題なし
HDMI ケーブル	2 本	CPU と FPGA 側を差し替えて使用する場合は
		1 本でも問題なし
USB Type A to Type B ケーブル	1 本	FPGA 側 UART の接続に使用
イーサネット・ケーブル	1 本	COMe CPU のイーサネットに使用

表 3-1 必要な機材

以下に当マニュアルの記述に際して使用したツールのバージョンを示します。

- インテル<sup>®</sup> Quartus<sup>®</sup> Prime スタンダード・エディション v17.1.1 (B593)
- Nios<sup>®</sup> II エンベデッド・デザイン・スイート v17.1.1 (B593)



# 4. セットアップ

当ボードでリファレンス・デザインを動作させるために必要な事前準備について解説します。 以下は当ボードの外観図です。



図 4-1 基板外観(部品面)





ボード仕様に関しては別途リファレンス・マニュアルもご参照ください。

 Mpression Magnes ボード:ドキュメント&リファレンス・デザイン https://service.macnica.co.jp/library/127633

# 4.1. リファレンス・デザインの入手

リファレンス・デザインはマクニカオンラインサービスの下記ページよりダウンロードいただけます。

- Mpression Magnes ボード:ドキュメント&リファレンス・デザイン https://service.macnica.co.jp/library/127633
- ハードウェア・リファレンス・デザイン
   Magnes\_GHRD\_Std171\_r0\_\_<ver>.zip
- Linux OS 向けサンプル・ドライバ & ユーザ・アプリケーション
  - magnes\_pcie\_r0\_\_<ver>.zip

このセクションで使用するファイルは「Magnes\_GHRD\_Std171\_r0\_\_<ver>.zip」です。Linux OS 向けサンプル・ドライバ & ユーザ・アプリケーションについては後述のセクションで使用します。

• 注:弊社 Web の都合上ファイル名末尾のバージョン (<ver> の部分)の数値が変更されることがありま すが適宜読み替えてください。



## 4.2. FPGA デザインの書き込み

当ボードにはコンフィギュレーション ROM として EPCQL 1024 が搭載されています。この章では JTAG ポートから FPGA を経由して EPCQ デバイスにプログラミングを実施することが可能な JTAG Indirect Configuration (JIC) プログラミングの手法について説明します。

なお、JIC プログラミングの概要については、弊社 Web サイトも合わせてご覧ください。

 FPGA 経由で EPCQ デバイスヘプログラミング(JIC プログラミング) https://service.macnica.co.jp/library/122065

### 4.2.1.JIC ファイルの書き込み

本手順の開始前にインテル<sup>®</sup> FPGA ダウンロード・ケーブル(旧 USB-Blaster<sup>™</sup> / USB-Blaster II<sup>™</sup>)(以後、ダウ ンロード・ケーブル) が CN5 に接続されていることを確認します。 ダウンロード・ケーブルの向きにご注意く ださい。

以下の図は ダウンロード・ケーブルを接続した状態です。

なお、本作業は COMe モジュールを接続する前に実施することを推奨いたします。また、ダウンロード・ケー ブルの抜き差しを実施する際に当ボードの電源は必ず OFF の状態で実施してください。

#### 4.2.1.1. ダウンロード・ケーブルの接続



図 4-3 ダウンロード・ケーブルの接続



#### 4.2.1.2. 電源の接続

付属の AC アダプタを DIN コネクタ(CN3) に接続します。



図 4-4 電源入力コネクタ

電源を接続後、電源スイッチ(SW1)を ON 側にスライドさせて基板に電源を投入します。





#### 4.2.1.3. Quartus® Prime Programmer の起動

当ボードの電源を ON にし、Quartus<sup>®</sup> Prime の Tools メニュー  $\Rightarrow$  Programmer を起動します。



図 4-6 Quartus<sup>®</sup> Prime Programmer

Programmer が起動したら、以下の手順を実施します。

- 1. ダウンロード・ケーブル の選択: USB-Blaster II (使用する ダウンロード・ケーブル に合わせて選択)
- 2. Mode を選択: JTAG
- プログラミング・ファイルを指定
   「Add File...」をクリックしてリファレンス・デザイン内 (Magnes\_GHRD\_Std171\_r0)に予め用意されて
   いる output\_files/magnes.jic 指定してください。

dware Setup U	JSB-BlasterII [USB-1] o allow background p	asteril [USB-1] Mode JTAG Progres			ress:	is:					
Start	File	Device	Checksum	Usercode	Program/ Configure	Verify	Blank- Check	Examine	Security Bit	Erase	ISP CLAMF
Stop											
to Detect											
to Detect											
to Detect Delete Id File											
to Detect Delete Id File ave File											

#### 図 4-7 Programmer の設定

 Program/Configure にチェックを入れ、Start をクリック 書き込みには少し時間がかかります。書き込みの進捗については Progress バーで確認することができ ます。

File	Device	Checksum	Usercode	Program/ Configure	Verify	Blank- Check	Examine	Security Bit	Erase	ISP CLAMP
Factory default SFL image	10AX090N2	306F0D2A	FFFFFFF							
output_file.jic	EPCQL1024	0D270289								
	1									
EPCQL1024										
±	18									
ты										
	File Factory default SFL image output_file.jic	File Device Factory default SFL image output_file.jic EPCQL1024 EPCQL1024 TDI	File     Device     Checksum       Factory default SFL image output_file.jic     10AX090N2     306F0D2A       D0270289     0D270289	File     Device     Checksum     Usercode       Factory default SFL image     10AX090N2     306F0D2A     FFFFFFF       output_file.jic     EPCQL1024     0D270289	File Device Checksum Usercode Program/ Configure output_file.jic EPCQL1024 0D270289	File     Device     Checksum     Usercode     Program/ Configure       Factory default SFL image output_file.jic     10AX090N2     306F0D2A     FFFFFFFF     Image: Configure	File     Device     Checksum     Usercode     Program/ Configure     Verify     Blank- Check       Factory default SFL image output_file.jic     10AX090N2     306F0D2A     FFFFFFFF     Image: Check     Image: Check       Image: Check image     10AX090N2     306F0D2A     FFFFFFFF     Image: Check image:	File     Device     Checksum     Usercode     Program/ Configure     Verify     Blank- Check     Examine       Factory default SFL image output_file.jic     10AX090N2     306F0D2A     FFFFFFFF     Image: Check     Image: Check	File     Device     Checksum     Usercode     Program/ Configure     Verify     Blank- Check     Examine     Security Bit       Factory default SFL image output_file.jic     10AX090N2     306F0D2A     FFFFFFFF     Image: Checkson of the control of t	File     Device     Checksum     Usercode     Program/ Configure     Verify     Blank- Check     Examine     Security     Erase       Factory default SFL image output_file.jic     10AX090N2     306F0D2A     FFFFFFFF     Image: Check of the security     Image: Check of the security

図 4-8 Programming の実施

書き込みが完了したら、再度基板に電源を入れ直します。これにより、新しい FPGA のイメージがコンフィギュ レーションされ、動作を開始します。

LED1(CONF\_DONE) の点灯で書き込んだ FPGA のイメージが動作していることが確認いただけます。





# 5. 事前準備

## 5.1. Linux のインストール

このセクションでは CPU モジュールに OS をインストールする手順について解説します。当マニュアルでは SECO 社の COMe モジュール (<u>COMe-A41-CT6</u>) を使用します。

この COMe モジュールにはオンボードで eMMC が 32GB 実装されており、この eMMC に対して Ubuntu 16.04 LTS をインストールする手順について紹介します。

なお、当マニュアルでは eMMC に対して OS をインストールする例を紹介しますが当ボードに搭載される SATA コネクタや USB ポートを使用して HDD / SSD / USB メモリなどにインストールすることも可能です。

• SECO: <u>https://www.seco.com/</u>

## 5.1.1.Ubuntu 16.04 LTS インストール・イメージの入手

当マニュアルを作成するにあたり使用した OS のインストール・イメージ下記サイトよりダウンロードいただけます。

https://www.ubuntulinux.jp/News/ubuntu1604-ja-remix

## 5.1.2.インストール・メディアの作成

上記よりダウンロードした OS イメージをインストール用 DVD やブータブル USB スティックとしてご準備ください。本手順は割愛しますが参考として Ubuntu の公式ページのリンクを紹介いたします。

- Ubuntu 公式チュートリアル・ページ
   <u>https://tutorials.ubuntu.com/tutorial/try-ubuntu-before-you-install?ga=2.242128704.317878189.1525862376-1237821753.1523249530#0</u>
- Windows でブータブル USB スティックを作成する方法
  - https://tutorials.ubuntu.com/tutorial/tutorial-create-a-usb-stick-on-windows#0
- Windows で DVD インストール・メディアを作成する方法
  - https://tutorials.ubuntu.com/tutorial/tutorial-burn-a-dvd-on-windows#0



### 5.1.3.各種接続

CPU モジュールを接続します。COMe モジュールを基板中央の COMe コネクタに接続してください。 対向側のネジ部にはスペーサが実装されていないため接続状態には注意してください。(スペーサは当ボード に付属されません。スペーサを使用する場合には高さは 7mm / M2 のスペーサが推奨となります。)



図 5-1 COMe コネクタ(CN10/CN11)



続いて、キーボード、マウス、インストール・メディアを USB コネクタに、HDMI モニタを COMe 側の HDMI コネクタに接続し、イーサネット・ケーブルを COMe の LAN コネクタに接続します。



図 5-2 USB / HDMI / LAN コネクタ



### 5.1.4.BIOS の設定

COMe モジュール上の eMMC へ OS をインストールするため BIOS の設定で eMMC が有効になって いることを確認します。BIOS を起動するには電源を ON にした後 BIOS 画面が表示されるまでキーボード の ESC ボタンを何度か押下します。

BIOS が起動すると以下のような画面が表示されます。



図 5-3 BIOS 起動画面

SCU を選択し、Advanced タブに移ります。LPSS & SCC Configuration を選択し、SCC eMMC Boot Controller が Enable であることを確認してください。

PSS & SCC Devices Made PSS & SCC Auto Switch ide Unsupported LPSS devices	<acp1 hode=""> <enab1e></enab1e></acp1>
CC Configuration CC eMMC Boot Controller	
P10/SD Card Selection GP00	<gp10> <low></low></gp10>
GP01 GP02	<low> <low></low></low>
6203	<low></low>
PSS Configuration PSS DHA #1 Support LPSS HSHART #1 Support	<disabled></disabled>
LPSS HSUART #2 Support PSS DHA #2 Support	<disabled></disabled>
	Control Control Prof SD Card Selection GPO1 GPO1 GPO3 PSS Configuration PSS Configuration PSS HSUMRT #1 Support LPSS HSUMRT #2 Support LPSS DHA #2 Support

図 5-4 eMMC 設定



確認ができたら ESC を 2回押下しメインメニューに戻ります。 続いて、OS 起動後に不要な画面出力を Disable に変更します。

Video Configuration を選択します。 VBT Hook Configuration のセクションが以下に設定してください。 ・CRT: **Disabled** ・DDI0 Configured As: **HDMI/DVI** ・DDI1 Configured As: **No Device** 

nain Advanced Security Power Br	Advanced	Insyde
Boot Configuration PCI Express Configuration USB Configuration Audio Configuration Hiscellmeous Configuration Security Configuration Volce Configuration Chipset Configuration Mark Configuration Mark Configuration Mark Configuration Mark Configuration Mark Configuration Mark Configuration Mark Configuration Mark Configuration	Video Configuration VBIOS Selection VBT Hook Configuration CRT DD10 Configured As DD11 Configured As UEC7 Settings Primary Display Secondary Display Display Hode	<de fault=""> &lt;0/isebiled&gt; &lt;00fi20V12 &lt;00fi20V12 &lt;00fi20V12 &lt;00fi02 &lt;00fi02 &lt;00fi02 &lt;00fi02 &lt;00fi02 &lt;00fi02 &lt;00fi02</de>
the second s		

☑ 5-5 Video Configuration

設定が完了したら F10 を押下し、設定を保存して再起動します。BIOS が起動するまで ESC を押下し、起動したら設定した内容が保存されていることを確認してください。問題がなければ一度電源を落とします。

xit Saving Chan	ges?
[Yes]	[No ]

図 5-6 設定の保存

これで BIOS の設定は完了です。



# 5.2. Ubuntu 16.04 LTS のインストール

続いて OS のインストール手順について解説します。 作成したインストール・メディアをボードに接続し電源を入れます。

インストール・メディアが起動したら「Ubuntu をインストール」を選択しインストールを開始します。

- ワンポイント・アドバイス
  - 使用するネットワークがプロキシ下にある場合には「Ubuntu を試す」で一度起動し、プロキシの設定を実施後に OS のインストールを開始するとネットワークが有効となります。

使用したい言語を選択し「続ける」を押下します。ここでは日本語を使用します。

ようこそ		
<ul> <li>E11日1日日</li> <li>E200002e</li> <li>Exco</li> <li>Anan</li> <li>Anan<th>お望みならば <u>リリースノート</u>をご覧ください。</th><th>耗了(Q) 戻る(B) 続ける</th></li></ul>	お望みならば <u>リリースノート</u> をご覧ください。	耗了(Q) 戻る(B) 続ける
	•••••	

図 5-7 言語の選択

デフォルトの設定のままインストールを進めます

Ubuntu のインストール準備	
☑ Ubuntu のインストール中にアップデートをダウンロードする	
これにより、インストール後の処理を短縮することができます。	
□ グラフィックス、Wi-Fi 機器、Flash、MP3 やその他のメディアに必要なサードパーティーソフトウェ	ェアをインストールする
This software is subject to license terms included with its documentation. Some is proprietary.	
Fluendo MP3 ブラグインには、Fraunhofer IIS and Technicolor SAからライセンスされたMPEG Layer-3 音声デコード	技術が含まれています。
	授 J (Q) 戻 (B) 続ける

図 5-8 アップデートのダウンロード実施有無の設定



「ディスクを削除して Ubuntu をインストール」を選択し「インストール」を押下します。



図 5-9 インストール方法の選択

ロケーションを指定します。必要に応じて変更してください。



図 5-10 ロケーションの設定

キーボードのレイアウトを指定します。必要に応じて変更してください。

キーボードレイアウト	
キーボードレイアウトの選択:	
英語(は5) 英語(カルーン) 英語(オーナ) 英語(オーナ) 英語(オーナ) 英語(オーナ) 英語(オーナ) 英語(オーナ) 第二 第二 第二 第二 第二 第二 第二 第二 第二 第二	旧本語 日本語 - 日本語 (Dvorak) 日本語 - 日本語 (Macintosh) 日本語 - 日本語 (OADC 100A) 日本語 - 日本語 (かな 86) 日本語 - 日本語 (かな 96)
キーボード入力をここで試してください	
キーボードレイアウトを検出	
	戻る(B) 続ける
•••	•••

図 5-11 キーボード・レイアウトの設定



ユーザ名とパスワードを指定します。パスワードは後ほど使用するため覚えておいてください。

あなたの情報を入力してください		
あなたの名前: コンピューターの名前: ユーザー名の入力: パスワードの承認: パスワードの確認:	<ul> <li>● のコンビューターと遠信するときに使用する名前です。</li> <li>● 自動的にログインする</li> <li>● ログイン時にバスワードを要求する</li> <li>● ホームフォルダーを暗号化する</li> </ul>	
		<b>戻る(B)</b> 続ける

図 5-12 ユーザ・アカウントの設定

インストールが完了するのを待ちます。



図 5-13 インストールの実行画面

インストールが完了したらインストール・メディアを取り出し再起動してください。 これで OS のインストールは完了です。



# 6. リファレンス・デザイン

このセクションではリファレンス・デザインの構成について解説します。

## 6.1. リファレンス・デザインの構成

以下にリファレンス・デザイン全体の概略のブロック図を示します。 リファレンス・デザインは大きく分けて以下 2 つのデザイン・ブロックから構成されております。

- PCle & Video ブロック (main.qsys)
- トランシーバ・ブロック (control.qsys)

PCle & Video ブロックは、PCle に各種ペリフェラルが接続されており CPU からのアクセスを受け付けます。 また、それとは独立したビデオ関連のブロックを内包しており 1 つのプラットフォーム・デザイナー(旧 Qsys) モジュール(main.qsys) として実装されます。

トランシーバ・ブロックは、トランシーバのループバック回路を実装したブロックがプラットフォーム・デザイナーのモジュール(control.qsys) として実装されます。

それぞれのブロックに関しての詳細は後述のセクションに記載いたします。



図 6-1 リファレンス・デザインのブロック図



### 6.1.1.PCle & Video ブロックの詳細

以下に PCle & Video 関連ブロックの詳細を示します。

COMe モジュールの CPU とは PCIe Gen2x1 で接続されており、PCIe を介して FPGA 内部のオンチッ プ・メモリや LED・ボタン等を制御するための PIO、FPGA 側に実装される DDR メモリ・コントローラに接続 されます。

また、画像出力として FPGA 内部でカラーパターン(1920 x 1080) を生成し HDMI インタフェースへ出力し ます。DDR をフレームバッファとして介在させることでデザインの変更にも対応しやすい構成となっております。



図 6-2 PCIe & Vidoe ブロックの詳細

なお、リファレンス・デザイン内のカラーパターンの出力にはビデオ / 画像処理 (VIP) スイート・インテル<sup>®</sup> <u>FPGA IP</u> (以下 VIP) を使用しており、本 IP を使用するには別途ライセンスが必要です。当ボードに付属の シリアルナンバーをお知らせいただければ 60 日間の評価ライセンスを発行いたします。 ライセンス発行が必要な場合には、販売代理店までお問い合わせください。

## 6.1.2.トランシーバ・ブロックの詳細

以下にトランシーバ・ブロックの詳細を示します。

UART を使用し FPGA 内部の Nios<sup>®</sup> II プロセッサーを介してトランシーバ・ブロックを制御します。 これによりテストパターンを利用したテストを FMC コネクタを介した外部ループバック、およびデバイス内部で のループバックを実行可能です。



図 6-3 トランシーバ・ブロックの詳細



## 6.2. PCle の BAR マッピング

本リファレンス・デザインの PCle & Video ブロックでは PCle IP に対して以下のように各種ペリフェラルがマップされています。BAR のマッピングに関してはデザインを追加・変更することで自由に変更が可能です。

BAR 番号	オフセット・ アドレス	ペリフェラル	備考
BAR0	0x0700_0000	On-Chip メモリ	512KB
BAR0	0x0800_0000	DDR メモリ	DDR は 2GB あり Host OS で直接マップできないた め Address_Span_Extender を介して接続し 32MB としてマップ
BAR2	0x0000_0000	PCIe IP Core CRA	PCle エンドポイントの各種設定用レジスタ (*Note1)
BAR2	0x0000_4000	LED PIO	LED7 - LED10 (緑色)に接続 (*Note2)
BAR2	0x0000_4100	DIP SW PIO	SW2: DIP スイッチに接続 (*Note2)
BAR2	0x0000_4200	Puch SW PIO	SW3 - SW6: プッシュスイッチに接続 (*Note2)
BAR2	0x0000_5000	Address_Span_Extender Ctrl	上記 DDR メモリの容量を小さく見せるために使用 マップ対象のオフセットを指定するためのレジスタ (*Note3)
BAR2	0x0600_0000	mSGDMA Descriptor	DMA のディスクリプタ登録用スレーブポート (*Note4)
BAR2	0x0660_0000	mSGDMA CSR	DMA の制御用スレーブポート (*Note4)

- Note1: Intel® Arria® 10 and Intel® Cyclone® 10 Avalon®-MM Interface for PCIe\* User Guide v17.1
- Note2: Embedded Peripherals IP User Guide PIO Core
- Note3: <u>Intel<sup>®</sup> Quartus<sup>®</sup> Prime Standard Edition Handbook Volume 1 v17.1</u>
- Note4: Embedded Peripherals IP User Guide Modular Scatter-Gather DMA Core
- Note5: Note2 / Note4 の資料は最新版(v18.0)へのリンクとなっております。
- Note6: リンクの変更や該当ページが変更になる可能性もございます。あらかじめご了承ください。

上記 PIO に接続される LED / プッシュスイッチ / DIP スイッチは以下に示す通りボード端部に実装されます。



図 6-4 FPGA ユーザ I/F



# 6.3. その他

### 6.3.1.リファレンス・デザイン使用上の注意

当ボードでは FPGA の PCIe ブロックへのリファレンスクロックが 2 系統接続されており、構成により切り替えることが可能です。

- COMe\_PCIE\_REFCLK :
  - COMe モジュールから供給されるクロック
  - > COMe モジュール使用時にはこちらを使用
- PCIE\_OB\_REFCLK :
  - 当ボード上にある発振器から供給されるクロック

リファレンス・デザインを使用する場合、COMe モジュールの有無で PCle IP へのリファレンスクロックの接続 を変更する必要がありますのでご注意ください。この変更を行わないと FPGA のトランシーバ・ブロックが正しく 動作しません。

デザインのトップファイル (golden\_top.v) を COMe モジュールの有無に合わせて、下記のようにそれぞれ設定してください。変更箇所は 2 か所です。

#### Line: 512

#### ↓ COMe 使用時 510 // NPOR Synchronize 511 reg [2:0] npor; 512 日 always @( posedge COME PCIE\_REFCLK /\*PCIE\_0B\_REFCLK\*/ ) begin //If using SECO, Assign to "COMe\_PCIE\_REFCLK". 513 npor <= {npor[1:0], COMe\_PCIE\_RESETN};

#### ↓COMe <u>未使用時</u>

510 511 512 513	<pre>// NPOR Synchronize reg [2:0] npor; always @( posedge /*COMe_PCIE_REFCLK*//PCIE_OB_REFCLK ) begin npor &lt;= {npor[1:0]. COMe_PCIE_RESEINT:</pre>	<pre>//If using SECO, Assign to "COMe_PCIE_REFCLK".</pre>
514	end	

#### Line: 769

#### ↓COMe <u>使用時</u>

766	// DCTo_Intenface	
700	// PCIe Incertace	
767	.npor_npor	( npor [2] ),
768	.npor_pin_perst	( COMO DELLE RESETTE ),
769	.refclk_clk	( COME_PCIE_REFCLK/*PCIE_OB_REFCLK*/), //If using SECO, Assign to "COME_PCIE_REFCLK",
770	.hip_serial_rx_in0	(COME_PCIE_RX_U)),

#### ↓COMe **未使用時**

766	// PCIe Interface	
767	.npor_npor	( npor [2] ),
768	.npor_pin_perst	( COMe_PCIE_RESETN
769	.refclk_clk	( /*COMe_PCIE_REFCL[*/PCIE_OB_REFCLK), //If using SECO, Assign to "COMe_PCIE_REFCLK".
770	hip_serial_rx_in0	(COME_PCIE_RX_0

#### 図 6-5 クロックの選択



# 7. リファレンス・デザインの実行

## 7.1. COMe モジュールの起動

「5.事前準備」でセットアップした Ubuntu 16.04 を起動します。

# 7.2. サンプル・ソフトウェアのダウンロード

当ボード上には LAN コネクタが実装されているためサンプル・ドライバの入手には起動した Linux 上から直 接ダウンロード可能です。

これはあくまでも一例であり別途ダウンロード済みのサンプル・ドライバを USB メモリ等で受け渡していただいても問題ありません。

「4.1 リファレンス・デザインの入手」で記載の通り、下記ダウンロード先よりサンプル・ソフトウェアをダウンロードしてください。

 Mpression Magnes ボード:ドキュメント&リファレンス・デザイン https://service.macnica.co.jp/library/127633

ダウンロードするファイルは「magnes\_pcie\_r0\_<ver>.zip」です。

• 注:弊社 Web の都合上ファイル名末尾のバージョン (<ver> の部分) の数値が変更されることがありま すが適宜読み替えてください。

本サンプル・ソフトウェアはあくまでも参考として提供されており動作を保証するものではございません。あらか じめご了承ください。



図 7-1 COMe LAN コネクタ



# 7.3. ドライバとユーザ・アプリケーションのビルド

ダウンロードしたサンプル・ドライバのアーカイブ・ファイルを展開します。(ここでは /home/work 以下に展開を した場合を例に紹介します)

\$ unzip magnes\_pcie\_r0\_\_<ver>.zip

#### 展開したディレクトリに移動します。

\$ cd magnes\_pcie

ドライバとユーザ・アプリケーションをビルドします。make コマンドを実行するとユーザ・アプリケーションとドラ イバの両方が順番にビルドされます。

\$ make

magnes@magnes:~/work/magnes pcle\$ make
make clean -C app
make[1]: ディレクトリ '/home/magnes/work/magnes pcie/app' に入ります
rm -rf ./obj ./obj/pci.d ./obj/main.d ./obj/cmd.d ./obj/mem util.d app
make[1]: ディレクトリ '/home/magnes/work/magnes pcie/app' がら出ます
make -C app
make[1]: ディレクトリ '/home/magnes/work/magnes_pcie/app' に入ります
mkdir -p ./obj
gcc -g -std=gnu99 -MMD -MP -Wall -Wextra -Winit-self -Wno-missing-field-initializers -DLINUX -II
/driver/ -o obj/pci.o -c pci.c
mkdir -p ./obj
gcc -g -std=gnu99 -MMD -MP -Wall -Wextra -Winit-self -Wno-missing-field-initializers -DLINUX -II
/driver/ -o obj/main.o -c main.c
mkdir -p ./obj
gcc -g -std=gnu99 -MMD -MP -Wall -Wextra -Winit-self -Wno-missing-field-initializers -DLINUX -II
/driver/ -o obj/cmd.o -c cmd.c
mkdir -p ./obj
gcc -g -std=gnu99 -MMD -MP -Wall -Wextra -Winit-self -Wno-missing-field-initializers -DLINUX -II
/driver/ -o obj/mem_util.o -c mem_util.c
gcc -o app_obj/pci.o obj/main.o obj/cmd.o obj/mem_util.o
make[1]: ディレクトリ '/home/magnes/work/magnes_pcie/app' から出ます
make cleanC driver
make[1]: ディレクトリ '/home/magnes/work/magnes_pcie/driver' に入ります
make -C /lib/modules/4.4.0-121-generic/build M=/home/magnes/work/magnes_pcie/driver clean
make[2]: ディレクトリ '/usr/src/linux-headers-4.4.0-121-generic' に入ります
CLEAN /home/magnes/work/magnes_pcie/driver/.tmp_versions
CLEAN _/home/magnes/work/magnes_pcie/driver/Module.symvers
make[2]: デイレクトリ '/usr/src/linux-headers-4.4.0-121-generic' から出ます
make[1]: ナイレクトリ '/home/magnes/work/magnes_pcie/driver' から出ます
make -C driver
make[1]: デイレクトリ /home/magnes/work/magnes_pcle/driver' に入ります
magnes_pcl.c magnes_tilelo.c
make -C /Lib/modules/4.4.0-121-generic/build M=/home/magnes/work/magnes_pcie/driver clean
make[2]: ディレクトリー/Usr/src/linux-headers-4.4.0-121-generic にんります
make 2]: デイレクトリー/Usr/src/Linux-neaders-4.4.0-121-generic から出まり
make - C / Clb/modules/4.4.0-121-generic/pulla #=/nome/magnes/Work/magnes/Ecle/artver modules
Make[2]: ノイレントリー/Usr/src/tinux-neaders-4.4.0-121-generic に入りまり
CC [m] /home/magnes/work/magnes_pcte/ortver/magnes_pct.o
LD [M] /home/magnes/work/magnes_pcie/driver/magnes_riteto.
Ruilding modulas stand 3
MODEST 1 modulor
CC /home/magnes/work/magnes prie/driver/magnes dry mod o
LD [M] /home/magnes/work/magnes_prie/driver/magnes_driv.hou.o
ーローフィース (All and All
make[1]: $f \neq \nu 2$ home/magnes/work/magnes prie/driver! the Hat

図 7-2 ドライバとユーザ・アプリケーションのビルド

ビルドが正常に終了し app ディレクトリ以下に app ファイルが、driver ディレクトリ以下に magnes\_drv.ko ファイルが生成されていることを確認します。



# 7.4. ドライバのインストール

ビルドしたドライバをインストールします。 ドライバのインストールには以下のコマンドを使用可能です。

\$ make install



図 7-3 ドライバのインストール

上記のコマンドにて insmod コマンド、およびインストールされたデバイス・ファイルに対して権限を付与します。 正常にインストールされると dmesg コマンドに以下のログが出力されることが確認できます。

\$ dmesg

🔊 🗊 magnes	@magnes: ~	
nes@magnes:	-\$ dmesg	
🔵 🛈 magnes	@magnes: ~	
413.926868]	/ <sup>-</sup> e / <sup>-</sup> l	
413.926873]	\$\$ \$ /\$\$	
413.926876]	\$\$\$ \$ /\$\$\$   /	\$ / \$ / \$ / \$ /
413.926879]	\$\$\$\$ /\$\$\$\$   \$\$\$\$\$ \$\$ \$\$ \$\$/\$\$   /	5  /555555  5555555  /5555555  /5555555/ 55  55   55  55   55  55  55  55  5
	\$\$  \$\$\$/ \$\$  /\$\$\$\$	ss iss s_ss iss i ss isssssss/ ssssss i
413.926887]	\$\$   \$/ \$\$  \$\$ cc/ ccccc	\$\$  \$\$   \$\$  \$\$   \$\$  \$\$  / \$\$/ ee/ eeeeee lee/ ee/ eeeeee/ eeeeee/
413.926893]		/ \$\$\$
413.926896]		\$\$ \$\$/
413.926899		\$\$\$\$\$\$/
413.926904]		
413.926907]	magnes_init	araba (day - exffffeee1720adeee aciid - exffffffffee2d1100)
413.926973]	vendor = 0x1172	
	device = 0xe001	
413.926980	class = 0xff0000 bus:slot.func = 01	• AA. AA
413.926991]	Upstream device 808	5/f48, bus:slot.func 00:1c.00
413.926994]	List of downstream of Link is operating at	t 5.0 GT/s with 1 lanes.
	Need to retrain.	
413.941404]	Link training comple	eted in 2 ms. GT/s with 1 lanes
413.941664]	BAR[0] 0x90000000-0	x9fffffff flags 0x0014220c
413.941670]	BAR[2] 0xa0000000-0	xa7ffffff flags 0x00040200
413.941758]	BAR[2] mapped at 0x1	ffffc90010000000 with length 134217728.
	Vender ID	= 0x1172
413.941778]	Device ID	= 0xe001
413.941788]	STATUS	= 0x0010
413.941792]	Rev ID	= 0x09
413.941797	Class Code Cache Line Size	= 0x0000009 = 0x10
	Mst Latency Timer	= 0x00
413.941812]	Header Type BIST	= 0x00 = 0x00
413,941822]	BARO	= 0x9000000c
413.941826]	BAR1	= 0x0000000
413.941836]	BAR3	= 0x00000000
413.941841]	BAR4	= 0x00000000
413.941846	BAR5 Cardbus CIS Pointer	= 0x00000000 = 0x00000000
413.941855]	SubSys Vender ID	= 0xa106
413.941860]	SubSys ID	= 0x2474
413.941805]	Capability List	= 0x50
413.941875]	Int Line	= 0x0b
413.941879]	Int Pin Min Gnt	= 0x01 = 0x00
413.941889]	Max Latency	= 0x00

図 7-4 dmesg コマンドの出力

# 7.5. ユーザ・アプリケーションの実行

ユーザ・アプリケーションを起動します。ユーザ・アプリケーションの起動には以下のコマンドが使用可能です。

\$ make run

```
magnes@magnes:~/work/magnes_pcie$ make run
```

図 7-5 ユーザ・アプリケーションの実行

本コマンドにより app ディレクトリ以下に生成されている app を起動しています。直接実行いただいても問題 ありません。

ユーザ・アプリケーションが起動すると下記のロゴと共にコンソールのメニューが表示されます。

\$\$ SS SS \$\$ \$\$ \$\$\$\$\$\$\$\$ SSSSSS SS \$\$ 155 SS 55 \$\$/ \$\$\$\$\$\$\$ 1\$\$/ \$\$/ SSSSSSS/ \$\$\$\$\$\$\$ \$\$\$ SS SS ss SSSS menu menu led <period(ms)> <times> led memdump <target(0:ddr/1:ocram)> <offset(HEX)> <size(HEX)> memdump memtest <target(0:ddr/1:ocram)> <mode(0:fixed/1:incremental)> memtest pcidump pcidump cfgread cfgread <target(0:end/1:root)> <offset(HEX)> barread barread <BAR:0/1/2..> <offset(HEX)> barwrite <BAR:0/1/2..> <offset(HEX)> <data(HEX)> barwrite exit exit Note: HEX Value does not need 0x Command:

図 7-6 ユーザ・アプリケーションの起動画面



## 7.5.1. サポート・コマンド一覧

ユーザ・アプリケーションには PCle を経由して各種アクセスを確認するためのコマンドが実装されます。実装されているコマンドと概要は以下の通りです。

表 7-1 サポート・コマンドー覧

コマンド	コマンド概要
menu	当ユーザ・アプリケーションに実装されているコマンドー覧を表示します
led	当ボード上に実装される LED の点滅を実行します
memdump	FPGA に接続される DDR メモリと FPGA 内部のオンチップメモリの内容を読み出します
memtest	FPGA に接続される DDR メモリと FPGA 内部のオンチップメモリに対してメモリテストを実行します
pcidump	PCle エンドポイントの PCl コンフィグレーション空間の情報を出力します
cfgread	PCI コンフィグレーション空間の指定したオフセットの情報を読み出します
barread	指定した BAR 番号とオフセットに接続されるペリフェラルから読み出しを実行します
barwrite	指定した BAR 番号とオフセットに接続されるペリフェラルに書き込みを実行します
exit	プログラムを終了します

### 7.5.2. menu コマンド

本コマンドを使用すると実装されているコマンドー覧を表示します。デフォルトでは繰り返しメニューが表示されるモードのため通常は使用しません。ログを残すことを目的としてモードを切り替えることが可能となっておりこの場合にメニューを再表示させたい場合に使用します。

Usage: menu

- コマンド名: menu
- 引数なし

/ 5 / 1 \$\$ /\$\$ / 5 / 1 \$\$5 \$ /\$\$\$   / 5 / 5 / 5 / 5 / 5 / 5 / 5 / 5 / 5 \$\$5 \$\$ /\$5\$   /\$5\$5\$5\$  /\$5\$5\$5\$  /\$5\$5\$5\$  /\$5\$5\$55 \$\$ 5\$ 5\$ 5\$ / 5\$   5\$ 15\$   5\$ 15\$ 15\$ 5   55 \$\$ 15\$ / 5\$   5\$ 55 5   5\$ 15\$ 15\$ 5   5\$ 5 \$\$ 15\$ / 5\$   5\$ 55 5   5\$ 15\$ 15\$ 15\$ 55 \$\$ 15\$ / 5\$   5\$ 55 5   5\$ 15\$ 15\$ 15\$ 55 \$\$ 15\$ / 5\$   5\$ 55 15\$   5\$ 15\$ 15\$ 15\$ 55 \$\$ 15\$ / 5\$ 15\$ 55 15\$   5\$ 15\$ 15\$ 15\$ 15\$ 15\$ \$\$ 15\$ / 5\$ 15\$ 55 15\$ 15\$ 15\$ 15\$ 15\$ 15\$ 15\$ 55 \$\$ 15\$ 5\$ 15\$ 5\$ 15\$ 15\$ 15\$ 15\$ 15\$ 15\$
<pre>menu : menu led : led <period(ms)> <times> memdump : memdump <target(0:ddr 1:ocram)=""> <offset(hex)> <size(hex)> memtest : memtest <target(0:ddr 1:ocram)=""> <mode(0:fixed 1:incremental)=""> pcidump : pcidump cfgread : cfgread <target(0:end 1:root)=""> <offset(hex)> barread : barread <bar:0 1="" 2=""> <offset(hex)> barrwrite : barwrite <bar:0 1="" 2=""> <offset(hex)> <data(hex)> exit : exit * Note: HEX Value does not need 0x</data(hex)></offset(hex)></bar:0></offset(hex)></bar:0></offset(hex)></target(0:end></mode(0:fixed></target(0:ddr></size(hex)></offset(hex)></target(0:ddr></times></period(ms)></pre>
Command: menu
menu
<pre>menu : menu led : led <period(ms)> <times> memdump : memdump <target(0:ddr 1:ocram)=""> <offset(hex)> <size(hex)> memtest : memtest <target(0:ddr 1:ocram)=""> <mode(0:fixed 1:incremental)=""> pcidump : pcidump cfgread : cfgread <target(0:end 1:root)=""> <offset(hex)> barread : barread <bar:0 1="" 2=""> <offset(hex)> barwrite : barwrite <bar:0 1="" 2=""> <offset(hex)> <data(hex)> exit : exit * Note: HEX Value does not need 0x</data(hex)></offset(hex)></bar:0></offset(hex)></bar:0></offset(hex)></target(0:end></mode(0:fixed></target(0:ddr></size(hex)></offset(hex)></target(0:ddr></times></period(ms)></pre>
Please Input Enter Key:

図 7-7 menu コマンドの実行例

継続ログモードに切り替える場合には app ディレクトリ以下に格納される main.c 内にある以下の define マクロを 1 にセットしてアプリケーションを再度ビルドしてください

変更前: #define CONTINUE\_LOG\_MODE (0) 変更後: #define CONTINUE\_LOG\_MODE (1)



### 7.5.3. led コマンド

本コマンドを使用して当ボード上の LED の点滅動作を確認できます。点滅する LED は「図 7-9 LED の配置」の LED7-LED10 の緑色の LED が該当します。コマンド実行時に点滅の周期と回数を指定します。

Usage: led <period(ms)> <times>

- コマンド名: led
- 第1引数: 点滅周期(ms)
- 第2引数: 点滅回数

/ 5 / 1 \$\$ 5 /\$5
\$\$_\$\$' \$\$\$\$\$ \$\$\$\$\$\$
menu : menu
<pre>led : led <period(ms)> <times></times></period(ms)></pre>
<pre>memdump : memdump <target(0:ddr 1:ocram)=""> <offset(hex)> <size(hex)></size(hex)></offset(hex)></target(0:ddr></pre>
<pre>memtest : memtest <target(0:ddr 1:ocram)=""> <mode(0:fixed 1:incremental)=""></mode(0:fixed></target(0:ddr></pre>
pcidump : pcidump
cfgread : cfgread <target(0:end 1:root)=""> <offset(hex)></offset(hex)></target(0:end>
<pre>barread : barread <bar:0 1="" 2=""> <offset(hex)></offset(hex)></bar:0></pre>
<pre>barwrite : barwrite <bar:0 1="" 2=""> <offset(hex)> <data(hex)></data(hex)></offset(hex)></bar:0></pre>
exit : exit
* Note: HEX Value does not need 0x
Command: led 100 10 led 100 10
led
Please Input Enter Key:

図 7-8 led コマンドの実行例



図 7-9 LED の配置



### 7.5.4. memdump コマンド

本コマンドを使用して FPGA 側に実装される DDR メモリと FPGA 内部のオンチップメモリの情報を読み出 すことが可能です。

Usage: memdump <target(0:ddr/1:ocram)> <offset(HEX)> <size(HEX)>

- コマンド名: memdump
- 第1引数: 読み出しを実行するメモリを選択。
  - > 0: DDR メモリ
  - > 1: On-Chip メモリ
- 第2引数: 読み出しを実行するオフセットを指定。0x は不要。HEX で指定。
- 第3引数: 読み出すサイズを指定。0x は不要。HEX で指定。

以下は memtest コマンドを実行した後の DDR メモリをダンプした結果の例です。

•••••				
/ \$ \$\$ \$ /\$ \$\$\$ \$ /\$ \$\$\$\$ \$ /\$ \$\$ \$ \$\$	/ \$\$   \$\$   / \$\$   \$\$\$\$ \$\$   \$ \$\$  /\$\$\$ \$\$  \$\$ \$\$  \$\$ \$\$  \$\$	\$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$	\$ / \$55  \$55: \$5  \$5 \$5  \$5 \$5  \$5 \$55  \$5 \$555  \$5 \$555   \$55   \$55	5 / 5 / 1 5555  /555555  /5555555/   55  55   55  555555/ 555555     55  55  / 55/ 55/ 5555555/ 5555555/
menu	: mei	nu		
Led	: leo	d <period< td=""><td>(ms)&gt; <ti< td=""><td>mes&gt;</td></ti<></td></period<>	(ms)> <ti< td=""><td>mes&gt;</td></ti<>	mes>
memdump	: mer	moump <tar< td=""><td>get(0:dd</td><td>r/1:ocram)&gt; <offset(hex)> <size(hex)></size(hex)></offset(hex)></td></tar<>	get(0:dd	r/1:ocram)> <offset(hex)> <size(hex)></size(hex)></offset(hex)>
memtest	: mer	mtest <tai< td=""><td>get(0:dd</td><td>r/1:ocram)&gt; <mode(0:fixed 1:incremental)=""></mode(0:fixed></td></tai<>	get(0:dd	r/1:ocram)> <mode(0:fixed 1:incremental)=""></mode(0:fixed>
pclaump	: pc	Loump		
bassood		gread <tal< td=""><td>get(0:end</td><td>d/1:rool/&gt; <orrsel(hex)></orrsel(hex)></td></tal<>	get(0:end	d/1:rool/> <orrsel(hex)></orrsel(hex)>
bacwrite	i Da		D.0/1/2	> coffcot(HEX)> cdata(HEX)>
ovit	. Da	i+	AR:0/1/2.	.> COTISEC(HEX)> COBCB(HEX)>
* Note: I	HEX Value	does not	need Av	
Command: n memdump 0 mem_dump	memdump 0 0 100	0 100		
ADDRESS:	+0	+4	+8	_+C
	+	+	00000000	+
080000000	000000000	00000001	00000002	00000003
08000010:	00000004	00000005	000000000	0000007
08000020:	00000008	00000009	0000000A	00000008
080000000	00000000	000000000	000000012	00000001
080000401	00000010	00000011	00000012	00000015
08000060:	00000018	00000019	00000014	00000018
	00000010	0000015	0000011	
08000070:	0000001C	0000001D	0000001E	000001F
08000070: 08000080:	0000001C 000000 <u>20</u>	0000001D 00000021	0000001E 00000022	0000001F 00000023
08000070: 08000080: 0800009 <u>0:</u>	0000001C 00000020 00000024	0000001D 00000021 00000025	0000001E 00000022 00000026	0000001F 00000023 00000027
08000070: 08000080: 08000090: 08000040:	0000001C 00000020 00000024 00000028	0000001D 00000021 00000025 00000029	0000001E 00000022 00000026 0000002A	000001F 0000023 00000027 0000002B
08000070: 08000080: 08000090: 080000A0: 080000A0:	0000001C 00000020 00000024 00000028 0000002C	0000001D 00000021 00000025 00000029 0000002D	0000001E 00000022 00000026 0000002A 0000002E	0000001F 00000023 00000027 0000002B 0000002F
08000070: 08000080: 08000090: 080000A0: 080000B0: 080000B0:	0000001C 00000020 00000024 00000028 0000002C 00000030	0000001D 00000021 00000025 00000029 0000002D 0000002D	0000001E 00000022 00000026 0000002A 0000002E 00000032	0000001F 00000023 00000027 0000002B 0000002F 0000002F
08000070: 08000090: 08000090: 080000A0: 080000B0: 080000C0: 080000C0:	0000001C 00000020 00000024 00000028 0000002C 00000030 00000034	0000001D 00000021 00000025 00000029 0000002D 00000031 00000035	0000001E 00000022 00000026 0000002A 0000002E 00000032 00000036	0000001F 00000023 00000027 0000002B 0000002F 00000002F 00000033
08000070: 08000090: 08000090: 080000000: 080000000: 080000000: 080000000:	0000001C 00000020 00000024 00000028 0000002C 00000030 00000034 00000038	0000001D 00000021 00000025 00000029 00000002D 00000031 00000035 00000039	0000001E 00000022 00000026 0000002A 0000002E 00000032 00000036 0000003A	0000001F 00000023 00000027 0000002B 0000002F 00000037 00000037
08000070: 08000080: 08000090: 08000080: 08000080: 08000000: 08000000: 080000E0: 080000F0:	0000001C 00000020 00000024 00000022 0000002C 00000030 00000034 00000038 0000003C	0000001D 00000025 00000029 0000002D 00000031 00000035 00000039 0000003D	0000001E 00000022 00000026 0000002A 0000002E 00000032 00000036 0000003A 0000003E	0000001F 00000023 00000027 0000002F 00000033 00000037 0000003B 0000003F

図 7-10 memdump コマンドの実行例



#### 7.5.5. memtest コマンド

本コマンドを使用して FPGA 側に実装される DDR メモリと FPGA 内部のオンチップメモリに対してメモリテ ストの実行が可能です。

Usage: memtest <target(0:ddr/1:ocram)> <mode(0:fixed/1:incremental)>

- コマンド名: memtest
- 第1引数: テスト対象のメモリを選択。
  - DDR メモリ
  - ▶ 1: On-Chip メモリ
- 第2引数: テストパターンの選択。
  - ▶ 0: 固定データ・パターン(0xAAAA5555)
  - ▶ 1: インクリメンタル・データ・パターン

以下は DDR メモリに対してインクリメンタル・データ・パターンを使用したメモリテストを実行した例です。テストが成功すると Test Pass!! の表示と共にテスト対象のメモリの先頭部分のダンプを出力します。

	-			
\$ /				
SS S /S	is i			
SS S /SS	s i /	s /	s /	s /s /l
555 1555	5 5555	55 1/555	555 1555	1,22222221 2,2222221 2,2222221
	C 1			
> >> >> >> >> >>/>		22 22 1	22 22	
> >>>/ >>>	> //>>>>	222 122 P		>>  >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
\$   \$/ \$	\$ \$\$	\$\$ 155	\$\$ \$\$	\$\$  \$\$  / \$\$/
\$\$/ \$	\$\$/ \$\$\$\$	\$\$\$/ \$\$\$	5555  55/	\$\$/ \$\$\$\$\$\$/ \$\$\$\$\$\$/
		/ \$.	\$\$	
		\$\$	\$\$/	
		\$\$\$	\$\$\$/	
menu	: mei	nu		
led	: le	d <period< td=""><td>(ms)&gt; <tir< td=""><td>nes&gt;</td></tir<></td></period<>	(ms)> <tir< td=""><td>nes&gt;</td></tir<>	nes>
memdump	: mer	mdump <tai< td=""><td>rget(0:dd</td><td>r/1:ocram)&gt; <offset(hex)> <size(hex)></size(hex)></offset(hex)></td></tai<>	rget(0:dd	r/1:ocram)> <offset(hex)> <size(hex)></size(hex)></offset(hex)>
memtest	: mer	mtest <tar< td=""><td>rget(0:dd</td><td>r/1:ocram)&gt; <mode(0:fixed 1:incremental)=""></mode(0:fixed></td></tar<>	rget(0:dd	r/1:ocram)> <mode(0:fixed 1:incremental)=""></mode(0:fixed>
pcidump	: DC	idump	A CONTRACTOR OF	
cforead	· cf	aread star	raet(0:en	d/1:root)> <offset(hex)></offset(hex)>
baccoad			D:0/1/2	offrot/HEX)>
bacusite	: 22	TIEdu SDAI	10/1/2	<pre>&gt; collsec(nex)&gt; collsec(n</pre>
Darwrite	: Da	write <br< th=""><th>AR:0/1/2.</th><th>.&gt; <orrsel(mex)> <udld(mex)></udld(mex)></orrsel(mex)></th></br<>	AR:0/1/2.	.> <orrsel(mex)> <udld(mex)></udld(mex)></orrsel(mex)>
exit	: ex	ιτ		
* Note: H	IEX Value	does not	need 0x	
Command: m	nemtest 0	1		
nemtest 0	1			
nem test				
lemory Tes	t with I	ncremental	l Data Pa	ttern
est Pass!	!			
Print Top	of Targe	t Memory		
DDRESS	+0	+4	+8	+C
800000	00000000	00000001	00000000	0000003
0000000	00000000	00000001	00000002	00000003
8000010:	00000004	00000005	00000000	0000007
8000020:	00000008	00000009	0000000A	0000008
8000030:	0000000C	000000D	0000000E	000000F
8000040:	00000010	00000011	00000012	0000013
8000050:	00000014	00000015	00000016	00000017
8000060:	00000018	00000019	0000001A	000001B
8000070:	00000010	0000001D	0000001F	0000001F
800080	00000000	0000021	00000022	0000023
8000000	88888824	00000021	00000022	00000023
8000090:	00000024	00000025	00000020	0000027
80000A0:	00000028	00000029	0000002A	0000025
80000B0:	0000002C	0000002D	0000002E	000002F
8000000:	00000030	00000031	00000032	0000033
80000D0:	00000034	0000035	00000036	0000037
80000E0:	00000038	00000039	000000 <u>3A</u>	000003B
80000F0:	0000003C	000003D	000003E	000003F
lease Los	ut Enter	Key:		

図 7-11 memtest コマンドの実行例



## 7.5.6. pcidump コマンド

本コマンドを実行すると PCI コンフィグレーション空間の情報を表示します。

Usage: pcidump

• コマンド名: pcidump

引数なし

5s       5       /5s       /5s       5       /5s         5sss       /5ss       5       /5ssss       5       5       1         5sss       /5sss       5       5       5       5       5       5       5         5ss       /5sss       5       15       5       15       5       5       5       5       5         5ss       /5       55       15       5       15       5	/c /1	
SSS 5       /SSS 1       /SSSS 5       /SSSS 5       /SSSS 5       /SSSS 5       /SSSSS 5       /SSSSS 5       /SSSSSS 5       /SSSSSSS 5       /SSSSSS 5       /SSSSSS 5       /SSSSSS 5       /SSSSSS 5       /SSSSSS 5       /SSSSSS 5       /SSSSSSS 5       /SSSSSS 5       /SSSSSSS 5       /SSSSSSS 5       /SS		
3335       1/3535       1/355555       1/355555       1/355555       1/355555         355       1/355555       1/355555       1/355555       1/355555       1/355555         355       1/3555555       1/3555555       1/3555555       1/3555555       1/3555555         355       1/3555555       1/3555555       1/3555555       1/3555555       1/3555555         351       351       351       351       355       1/355555       1/3555555         351       355       1/355555       1/3555555       1/355555       1/355555       1/355555         351       351       355       1/355555       1/355555       1/355555       1/3555555         351       351       355       355       355       355       355       355         351       355       3555       355       355       355       355       355         351       355<		
SSS SS 55/SS   // SS   SS   SS   SS   SS		1/2222222/1/2222222/1/2222222/1/2222222/1/222222
35       1553       155       1		
32       153       15		
SS/ SS/ SSSSSS/ SSSSSSS IS/ SS/ SSSSSS/ SSSSSS/ SSSSSS/ SS/ SSSSSS/ SSSSSSS IS/ SS/ SSSSSS/ SSSSSS/ SSSSSS/ mentum : menu led : led <period(ns)> <times> mendump : mendump <target(0:dd liocram)=""> <offset(hex)> <size(hex)> mentest : mentump <target(0:dd liocram)=""> <offset(hex)>  stred/lincremental)&gt; pcidump : pcidump cfgread <tbr></tbr>to pcidump  cfgread <tbr></tbr>to pcidump  sett = barwrite  sett = exit * Note: HEX Value does not need 0x Command: pcidump  pcidump  pcidump  pcidump  pcidump  pcidump  pcidump  pcidump  pcidump  pcidump  </br></br></br></br></br></br></br></br></offset(hex)></target(0:dd></size(hex)></offset(hex)></target(0:dd></times></period(ns)>		
<pre>very transformer is the set of the set</pre>	22222 122 122	(22222222 (22222222 (22222222 (22222222
<pre>ssss/ sssss/ ssssss/ mendump : mendump <times> mendump : mendump <target(0:ddr l:ocram)=""> <offset(hex)> <size(hex)> mentest : mentest <target(0:ddr l:ocram)=""> <mode(0:fixed l:incremental)=""> pcidump : pcidump cfgread : barread  shread : barread  shreit : barread sbAR:0/1/2&gt; <offset(hex)>  soffset(HEX)&gt; barread : cfgread <target(0:end l:root)=""> <offset(hex)> exit : exit * Note: HEX Value does not need 0x Command: pcidump pcidump PCI End Config Space Dump Vender ID = 0x1172 Device ID = 0xe001 CMD = 0x0010 Rev ID = 0x000 Cache Line Size = 0x10 Mst Latency Timer = 0x00 Header Type = 0x00 BAR0 = 0x0000000 BAR1 = 0x0000000 BAR1 = 0x0000000 BAR2 = 0x0000000 BAR3 = 0x00000000 BAR4 = 0x00000000 BAR4 = 0x00000000 BAR4 = 0x00000000 Cardbus CIS Pointer = 0x00 BAR5 = 0x0000000 BAR4 = 0x00000000 Cardbus CIS Pointer = 0x00 Hext = 0x0000000 Cardbus CIS Pointer = 0x00 Hext = 0x0000000 Cardbus CIS Pointer = 0x0000000 Cardbus CIS Pointer = 0x0000000 Cardbus CIS Pointer = 0x0000000 BAR5 = 0x0000000 BAR4 = 0x0000000 Cardbus CIS Pointer = 0x0000000 BAR5 = 0x0000000 BAR5 = 0x0000000 BAR4 = 0x0000000 Cardbus CIS Pointer = 0x0000000 Max Latency = 0x00 Min Cnt = 0x00 Please Input Enter Key: Please Input Enter Key:</br></br></br></offset(hex)></target(0:end></offset(hex)></mode(0:fixed></target(0:ddr></size(hex)></offset(hex)></target(0:ddr></times></pre>		
menu       :       menu         led       :       led cperiod(ns)> <times>         mendump       :       mendump <target(0:ddr l:ocram)=""> <offset(hex)> <size(hex)>         mentest       :       mentest <target(0:ddr l:ocram)=""> <mode(0:fixed l:incremental)="">         pctdump       :       pctdump(0:fixed/l:noram)&gt; <mode(0:fixed l:incremental)="">         pctdump       :       pctdump(0:fixed/l:noram)&gt; <mode(0:fixed l:incremental)="">         barread       :       barread  <tbr></tbr>exit       &gt;         exit       :       parwrite  exit       barread  exit         * Note:       HEX Value does not need 0x         Command:       pcidump         pcidump       pcidump         PCI End Config Space Dump       =         Vender ID       = 0x1172         Device ID       = 0x0010         Rev ID       = 0x0010         Rev ID       = 0x0010         Rev ID       = 0x00         Header Type       = 0x00         Header Type       = 0x00         BAR0       = 0x0000000         BAR1       = 0x00000000         BAR2       = 0x00000000         BAR3       = 0x00000000         BAR4       = 0x00000000      <t< td=""><td></td><td>ss</td></t<></mode(0:fixed></mode(0:fixed></mode(0:fixed></target(0:ddr></size(hex)></offset(hex)></target(0:ddr></times>		ss
<pre>menu : menu led : led <pre>certicd(ms)&gt; <times> memdump : memdump <target(0:ddr l:ocram)=""> <offset(hex)> <size(hex)> memtest : memtest <target(0:end 1:root)=""> <offset(hex)> <size(hex)> memtest : memtest <target(0:end 1:root)=""> <offset(hex)> barread : barread <bar:0 1="" 2=""> <offset(hex)> barread : barread <bar:0 1="" 2=""> <offset(hex)>  data(HEX)&gt; exit : exit * Note: HEX Value does not need 0x Command: pcidump pcidump PCI End Config Space Dump Vender ID = 0x1072 Device ID = 0x0001 CMD = 0x0010 Rev ID = 0x0010 Rev ID = 0x00 Cabs Code = 0xff000 Cabs Code = 0x000 BAR1 = 0x0000000 BAR1 = 0x0000000 BAR1 = 0x0000000 BAR2 = 0x0000000 BAR3 = 0x0000000 BAR4 = 0x0000000 BAR4 = 0x0000000 BAR4 = 0x0000000 BAR4 = 0x0000000 Cardbus CIS Pointer = 0x000 BAR5 ID = 0x2474 Expansion LOM Base = 0x000000 Capability List = 0x00 Max Latency = 0x00 Max Max Max Max Max Max Max Max Max Max</br></offset(hex)></bar:0></offset(hex)></bar:0></offset(hex)></target(0:end></size(hex)></offset(hex)></target(0:end></size(hex)></offset(hex)></target(0:ddr></times></pre></pre>		\$\$\$\$\$\$
<pre>menu : menu led : led <period(ms)> <times> memdump :: mendump <target(0:ddr l:ocram)=""> <offset(hex)> <size(hex)> memtest :: mentest <target(0:ddr l:ocram)=""> <mode(0:fixed 1:incremental)=""> pcidump :: pcidump cfgread :: cfgread <target(0:end 1:root)=""> <offset(hex)> barread :: barread <bar:0 1="" 2=""> <offset(hex)>  softset(HEX)&gt; barread :: barread <bar:0 1="" 2=""> <offset(hex)> <data(hex)> exit :: exit * Note: HEX Value does not need 0x Command: pcidump pcidump PCI End Config Space Dump Vender ID = 0x1172 Device ID = 0x0006 STATUS = 0x0006 STATUS = 0x000 Cache Line Size = 0x1f000 Cache Line Size = 0x16 Mst Latency Timer = 0x00 Header Type = 0x00 BAR1 = 0x00000000 BAR3 = 0x00000000 BAR3 = 0x00000000 BAR3 = 0x00000000 BAR4 = 0x00000000 BAR5 = 0x00000000 BAR5 = 0x00000000 Cardbus CIS Pointer = 0x000 BAR5 = 0x00000000 Cardbus CIS Pointer = 0x0000000 Cardbus CIS Pointer = 0x0000000 Cardbus CIS Pointer = 0x0000000 BAR5 = 0x00000000 Cardbus CIS Pointer = 0x0000000 Cardbus CIS Pointer = 0x0000000 Cardbus CIS Pointer = 0x00000000 Cardbus CIS Pointer = 0x0000000000000000000000000000000000</br></data(hex)></offset(hex)></bar:0></offset(hex)></bar:0></offset(hex)></target(0:end></mode(0:fixed></target(0:ddr></size(hex)></offset(hex)></target(0:ddr></times></period(ms)></pre>		******/
<pre>menu : menu led : led <pre>cyperiod(ms)&gt; <times> memdump : mendump <target(0:ddr 1:ocram)=""> <offset(hex)> <size(hex)> memtest :: memtest <target(0:ddr 1:ocram)=""> <mode(0:fixed 1:incremental)=""> pcidump :: pcidump cfgread :: cfgread <target(0:end 1:root)=""> <offset(hex)> barread :: barread <bar:0 1="" 2=""> <offset(hex)>  stread :: exit</offset(hex)></bar:0></offset(hex)></target(0:end></mode(0:fixed></target(0:ddr></size(hex)></offset(hex)></target(0:ddr></times></pre></pre>		
led       ied       cperiod(ms)> <ttmes>         memdump       mendump <target(0:ddr 1:ocram)=""> <offset(hex)> <size(hex)>         peidump       pcidump         cfgread       cfgread(1:ocram)&gt; <mode(0:fixed 1:iccremental)="">         pcidump       barread         cfgread       cbread         barread       barread         exit       exit         * Note:       HEX Value does not need 0x         Command:       pcidump         pcidump       exit         * Note:       HEX Value does not need 0x         Command:       pcidump         pcidump       pcidump         PCI End Config Space Dump         Vender ID       = 0x1172         Device ID       = 0x0006         STATUS       = 0x0010         Rev ID       = 0x00         Cass Code       = 0xff0000         Cass Code       = 0x00         BIST       = 0x00         BARA       = 0x0000000         BAR3       = 0x00000000         BAR4       = 0x00000000         BAR4       = 0x00000000         BAR5       = 0x00000000         BAR4       = 0x00000000         SubSys Vender ID       = 0</mode(0:fixed></size(hex)></offset(hex)></target(0:ddr></ttmes>	menu : men	
<pre>mendump :: mendump <target(0:ddr 1:ocram)=""> <offset(hex)> <size(hex)> memtest :: memtest <target(0:ddr 1:ocram)=""> <mode(0:fixed 1:incremental)=""> pcidump :: pcidump cfgread :: cfgread <target(0:end 1:root)=""> <offset(hex)> barread :: barread <bar:0 1="" 2=""> <offset(hex)> barwite :: barwrite <bar:0 1="" 2=""> <offset(hex)> <data(hex)> exit :: exit * Note: HEX Value does not need 0x Command: pcidump PCI End Config Space Dump Vender ID = 0x1172 Device ID = 0x8001 CMD = 0x000 Cache Line Size = 0x10 Mst Latency Timer = 0x00 Header Type = 0x00 BAR0 = 0x9000000 BAR2 = 0x00000000 BAR3 = 0x00000000 BAR4 = 0x00000000 BAR5 = 0x0000000 BAR5 = 0x000000 BAR5 = 0x0000000 BAR5 = 0x0000000 BAR5 = 0x0</data(hex)></offset(hex)></bar:0></offset(hex)></bar:0></offset(hex)></target(0:end></mode(0:fixed></target(0:ddr></size(hex)></offset(hex)></target(0:ddr></pre>	led : led	<pre>cperiod(ms)&gt; <times></times></pre>
<pre>mentest : mentest <target(0:ddr 1:ocram)=""> <mode(0:fixed 1:incremental)=""> pcidump cfgread : cfgread <target(0:end 1:root)=""> <offset(hex)> barread : barread <bar:0 1="" 2=""> <offset(hex)> barwrite : barwrite <bar:0 1="" 2=""> <offset(hex)> cata(HEX)&gt; exit : exit * Note: HEX Value does not need 0x Command: pcidump pcidump PCI End Config Space Dump Vender ID = 0x1172 Device ID = 0x000 CAMD = 0x0006 STATUS = 0x0010 Rev ID = 0x10 Mst Latency Timer = 0x00 BAR0 = 0x0000000 BAR2 = 0x00000000 BAR3 = 0x00000000 BAR4 = 0x00000000 Cardbus CIS Pointer = 0x00 SubSys Vender ID = 0x274 Expansion LOM Base = 0x0000000 Cardbus CIS Pointer = 0x00 Int Line = 0x00 Please Input Enter Key: Please Input Enter Key: </offset(hex)></bar:0></offset(hex)></bar:0></offset(hex)></target(0:end></mode(0:fixed></target(0:ddr></pre>	memdump : mem	<pre>dump <target(0:ddr 1:ocram)=""> <offset(hex)> <size(hex)></size(hex)></offset(hex)></target(0:ddr></pre>
<pre>pcidump : pcidump cfgread : cfgread <target(0:end 1:root)=""> <offset(hex)> barread <bar:0 1="" 2=""> <offset(hex)> barwite : barwrite <bar:0 1="" 2=""> <offset(hex)> <data(hex)> exit : exit * Note: HEX Value does not need 0x Command: pcidump pcidump PCI End Config Space Dump Vender ID = 0x1172 Device ID = 0xe001 CMD = 0x0006 STATUS = 0x0010 Rev ID = 0x809 Class Code = 0xff0000 Cache Line Size = 0x10 Mist Latency Timer = 0x00 BAR1 = 0x0000000 BAR1 = 0x00000000 BAR2 = 0x00000000 BAR3 = 0x00000000 BAR4 = 0x00000000 BAR4 = 0x00000000 Cardbus CIS Pointer = 0x00 BAR5 = 0x00000000 Cardbus CIS Pointer = 0x000 Cardbus CIS Pointer = 0x000 DATE = 0x0000000 Cardbus CIS Pointer = 0x00000000 Cardbus CIS Pointer = 0x0000000 Cardbus CIS Pointer = 0x00000000 Cardbus CIS Pointer = 0x0000000000000000000000000000000000</data(hex)></offset(hex)></bar:0></offset(hex)></bar:0></offset(hex)></target(0:end></pre>	memtest : mem	<pre>test <target(0:ddr 1:ocram)=""> <mode(0:fixed 1:incremental)=""></mode(0:fixed></target(0:ddr></pre>
<pre>cfgread : cfgread <target(0:end 1:root)=""> <offset(hex)> barread : barread <bar:0 1="" 2=""> <offset(hex)> exit : exit * Note: HEX Value does not need 0x Command: pcidump pcidump PCI End Config Space Dump Vender ID = 0x1172 Device ID = 0x000 CMD = 0x00060 STATUS = 0x0006 STATUS = 0x000 Cache Line Size = 0x10 Mst Latency Timer = 0x00 BIST = 0x00 BAR1 = 0x00000000 BAR2 = 0x00000000 BAR3 = 0x00000000 BAR4 = 0x00000000 BAR5 = 0x00000000 BAR5 = 0x00000000 Cardbus CIS Pointer = 0x000 BAR4 = 0x00000000 Cardbus CIS Pointer = 0x000 BAR5 = 0x00000000 Cardbus CIS Pointer = 0x0000000 Cardbus CIS Pointer = 0x00000000 Cardbus CIS Pointer = 0x0000000000000000000000000000000000</offset(hex)></bar:0></offset(hex)></target(0:end></pre>	pcidump : pci	dump
barread : barread <bar:0 1="" 2=""> <offset(hex)> barwrite : barwrite <bar:0 1="" 2=""> <offset(hex)> <data(hex)> exit : exit * Note: HEX Value does not need 0x Command: pcidump pcidump PCI End Config Space Dump Vender ID = 0x1172 Device ID = 0xe001 CMD = 0x0006 STATUS = 0x0010 Rev ID = 0x09 Class Code = 0xff0000 Cache Line Size = 0x10 Mst Latency Timer = 0x00 Header Type = 0x00 BAR0 = 0x90000000 BAR1 = 0x0000000 BAR3 = 0x0000000 BAR3 = 0x0000000 BAR4 = 0x0000000 Cardbus CIS Pointer = 0x000 Cardbus CIS Pointer = 0x00 DAR5 = 0x0000000 Cardbus CIS Pointer = 0x000000 Cardbus CIS Pointer = 0x0000000 Cardbus CIS Pointer = 0x000 Int Line = 0x00 Int Line = 0x00 Int Pin = 0x00 Max Latency = 0x00 Please Input Enter Key:</data(hex)></offset(hex)></bar:0></offset(hex)></bar:0>	cfgread : cfg	read <target(0:end 1:root)=""> <offset(hex)></offset(hex)></target(0:end>
barwrite : barwrite <bar:0 1="" 2=""> <offset(hex)> <data(hex)> exit : exit * Note: HEX Value does not need 0x Command: pcidump pcidump PCI End Config Space Dump Vender ID = 0x1172 Device ID = 0xe001 CMD = 0x0006 STATUS = 0x0010 Rev ID = 0x000 Cache Line Size = 0x10 Mst Latency Timer = 0x00 Header Type = 0x00 BAR0 = 0x90000000 BAR1 = 0x00000000 BAR2 = 0x00000000 BAR3 = 0x00000000 BAR4 = 0x00000000 BAR5 = 0x00000000 Cardbus CIS Pointer = 0x00 BAR5 = 0x00000000 Cardbus CIS Pointer = 0x00 SubSys Vender ID = 0xa106 SubSys Vender ID = 0xa106 SubSys Vender ID = 0xa106 SubSys Vender ID = 0x00 Int Line = 0x00 Max Latency = 0x00</data(hex)></offset(hex)></bar:0>	barread : bar	read <bar:0 1="" 2=""> <offset(hex)></offset(hex)></bar:0>
<pre>exit : exit * Note: HEX Value does not need 0x Command: pcidump pcidump PCI End Config Space Dump Vender ID = 0x1172 Device ID = 0xe001 CMD = 0x0006 STATUS = 0x0010 Rev ID = 0x09 Class Code = 0xf10 Mst Latency Timer = 0x00 Header Type = 0x00 BIST = 0x000 BAR0 = 0x90000000 BAR1 = 0x00000000 BAR2 = 0x00000000 BAR3 = 0x00000000 BAR3 = 0x00000000 BAR4 = 0x00000000 Cardbus CIS Pointer = 0x0000000 SubSys Vender ID = 0x2474 Expansion LOM Base = 0x00000000 Capability List = 0x50 Int Line = 0x00 Max Latency = 0x00 Please Input Enter Key:</pre>	barwrite : bar	write <bar:0 1="" 2=""> <offset(hex)> <data(hex)></data(hex)></offset(hex)></bar:0>
<pre>* Note: HEX Value does not need 0x Command: pcidump pcidump PCI End Config Space Dump Vender ID = 0x1172 Device ID = 0xe001 CMD = 0x0006 STATUS = 0x0010 Rev ID = 0x609 Class Code = 0xff0000 Cache Line Size = 0x10 Mst Latency Timer = 0x00 Header Type = 0x00 BAR0 = 0x90000000 BAR1 = 0x00000000 BAR2 = 0x00000000 BAR3 = 0x00000000 BAR3 = 0x00000000 BAR5 = 0x00000000 Cardbus CIS Pointer = 0x0000000 SubSys Vender ID = 0x106 SubSys ID = 0x2474 Expansion LOM Base = 0x0000000 Int Line = 0x00 Max Latency = 0x00</pre>	exit : exi	t
Command: pcidump pcidump PCI End Config Space Dump Vender ID = 0x1172 Device ID = 0xe001 CMD = 0x0006 STATUS = 0x0010 Rev ID = 0x09 Class Code = 0xff0000 Cache Line Size = 0x10 Mst Latency Timer = 0x00 Header Type = 0x00 BAR0 = 0x90000000 BAR1 = 0x00000000 BAR2 = 0x00000000 BAR2 = 0x00000000 BAR3 = 0x00000000 BAR4 = 0x00000000 BAR5 = 0x00000000 Cardbus CIS Pointer = 0x00000000 SubSys Vender ID = 0x106 SubSys ID = 0x2474 Expansion LOM Base = 0x000 Int Line = 0x00 Int Line = 0x00 Max Latency = 0x00 Please Input Enter Key:	* Note: HEX Value	does not need 0x
Command: pcidump pcidump PCI End Config Space Dump Vender ID = 0x1172 Device ID = 0x0006 STATUS = 0x0010 Rev ID = 0x09 Class Code = 0xff0000 Cache Line Size = 0x10 Header Type = 0x00 Header Type = 0x00 BAR0 = 0x90000000 BAR1 = 0x0000000 BAR2 = 0x0000000 BAR2 = 0x0000000 BAR3 = 0x0000000 BAR4 = 0x0000000 Cardbus CIS Pointer = 0x0000000 Cardbus CIS Pointer = 0x0000000 Capability List = 0x50 Int Line = 0x00 Max Latency = 0x00		
pcidump         PCI End Config Space Dump         Vender ID       = 0x1172         Device ID       = 0x0006         STATUS       = 0x0010         Rev ID       = 0x09         Class Code       = 0xff0000         Cache Line Size       = 0x10         Mst Latency Timer       = 0x00         Header Type       = 0x00         BIST       = 0x0000000         BAR0       = 0x00000000         BAR1       = 0x00000000         BAR3       = 0x00000000         BAR4       = 0x00000000         BAR5       = 0x00000000         Cardbus CIS Pointer       = 0x00000000         SubSys ID       = 0x2474         Expansion LOM Base       = 0x00         Int Line       = 0x00         Int Pin       = 0x00         Min Gnt       = 0x00         Max Latency       = 0x00         Please Input Enter Key:	Command: pcidump	
PCI End Config Space Dump         Vender ID       = 0x1172         Device ID       = 0x0006         STATUS       = 0x0010         Rev ID       = 0x09         Class Code       = 0xff0000         Cache Line Size       = 0x10         Mst Latency Timer       = 0x00         BIST       = 0x000000         BAR0       = 0x90000000         BAR1       = 0x00000000         BAR3       = 0x00000000         BAR4       = 0x00000000         Cardbus CIS Pointer       = 0x0000000         SubSys ID       = 0x2474         Expansion LOM Base       = 0x00         Int Line       = 0x00         Min Gnt       = 0x00         Max Latency       = 0x00	pcidump	
PCI End Config Space Dump         Vender ID       = 0x1172         Device ID       = 0x0006         STATUS       = 0x0010         Rev ID       = 0x09         Class Code       = 0xff0000         Cache Line Size       = 0x10         Mst Latency Timer       = 0x00         Header Type       = 0x00         BAR0       = 0x9000000         BAR1       = 0x0000000         BAR2       = 0x0000000         BAR3       = 0x0000000         BAR4       = 0x0000000         BAR5       = 0x0000000         BAR4       = 0x0000000         BAR5       = 0x0000000         BAR4       = 0x0000000         Cardbus CIS Pointer = 0x00000000       SubSys Vender ID         Expansion LOM Base       = 0x0000000         Capability List       = 0x50         Int Line       = 0x000         Min Gnt       = 0x00         Max Latency       = 0x00		
Vender ID = 0x1172 Device ID = 0xe001 CMD = 0x0006 STATUS = 0x0010 Rev ID = 0x09 Class Code = 0xff0000 Cache Line Size = 0x10 Mst Latency Timer = 0x00 BIST = 0x00 BAR0 = 0x9000000 BAR1 = 0x00000000 BAR2 = 0xa0000000 BAR3 = 0x00000000 BAR3 = 0x00000000 BAR4 = 0x00000000 Cardbus CIS Pointer = 0x00000000 SubSys Vender ID = 0xa106 SubSys ID = 0x2474 Expansion L0M Base = 0x0000000 Capability List = 0x50 Int Line = 0x00 Min Gnt = 0x00 Max Latency = 0x00	PCI End Config Space	e Dump
Vender ID       = 0x1172         Device ID       = 0x0001         CMD       = 0x0006         STATUS       = 0x0010         Rev ID       = 0x09         Class Code       = 0xff0000         Cache Line Size       = 0x10         Mst Latency Timer       = 0x00         Header Type       = 0x00         BIST       = 0x0000000         BAR0       = 0x9000000         BAR1       = 0x00000000         BAR2       = 0x00000000         BAR3       = 0x00000000         BAR4       = 0x00000000         BAR5       = 0x00000000         Carbus CIS Pointer       = 0x00000000         SubSys Vender ID       = 0x2474         Expansion LOM Base       = 0x00000000         Capability List       = 0x050         Int Pin       = 0x00         Min Gnt       = 0x00         Max Latency       = 0x00		
Device ID = 0xe001 CMD = 0x0006 STATUS = 0x0010 Rev ID = 0x09 Class Code = 0xff0000 Cache Line Size = 0x10 Mst Latency Timer = 0x00 Header Type = 0x00 BIST = 0x00 BAR0 = 0x9000000 BAR1 = 0x00000000 BAR2 = 0x00000000 BAR3 = 0x00000000 BAR4 = 0x00000000 Cardbus CIS Pointer = 0x00000000 Cardbus CIS Pointer = 0x00000000 SubSys Vender ID = 0xa106 SubSys ID = 0x2474 Expansion LOM Base = 0x00000000 Capability List = 0x50 Int Line = 0x00 Min Gnt = 0x00 Max Latency = 0x00	Vender ID	= 0x1172
CMD       = 0x00060         STATUS       = 0x0010         Rev ID       = 0x09         Class Code       = 0xff0000         Cache Line Size       = 0x10         Mst Latency Timer       = 0x00         Header Type       = 0x00         BAR0       = 0x9000000         BAR1       = 0x0000000         BAR2       = 0x0000000         BAR3       = 0x0000000         BAR4       = 0x0000000         BAR5       = 0x0000000         BAR4       = 0x0000000         BAR5       = 0x00000000         Cardbus CIS Pointer       = 0x0000000         SubSys Vender ID       = 0xa106         SubSys ID       = 0x2474         Expansion LOM Base       = 0x0000000         Capability List       = 0x50         Int Line       = 0x00         Min Gnt       = 0x00         Max Latency       = 0x00	Device ID	= 0xe001
STATUS       = 0x0010         Rev ID       = 0x09         Class Code       = 0x10         Mst Latency Timer       = 0x00         Header Type       = 0x00         BIST       = 0x00         BAR0       = 0x9000000         BAR1       = 0x0000000         BAR3       = 0x0000000         BAR4       = 0x0000000         BAR5       = 0x0000000         BAR5       = 0x00000000         Cardbus CIS Pointer       = 0x00000000         SubSys Vender ID       = 0x2474         Expansion LOM Base       = 0x0000000         Capability List       = 0x50         Int Line       = 0x00         Min Gnt       = 0x00         Max Latency       = 0x00	CMD	= 0x0006
Rev ID       = 0x09         Class Code       = 0x160000         Cache Line Size       = 0x10         Mst Latency Timer       = 0x00         Header Type       = 0x00         BIST       = 0x0000000         BAR0       = 0x0000000         BAR1       = 0x0000000         BAR2       = 0xa0000000         BAR3       = 0x00000000         BAR4       = 0x00000000         BAR5       = 0x00000000         Carbus CIS Pointer       = 0x00000000         SubSys Vender ID       = 0x2474         Expansion LOM Base       = 0x00000000         Capability List       = 0x050         Int Line       = 0x01         Min Gnt       = 0x00         Max Latency       = 0x00	STATUS	= 0x0010
Class Code = 0xTT0000 Cache Line Size = 0x10 Mst Latency Timer = 0x00 Header Type = 0x00 BIST = 0x00 BAR0 = 0x90000000 BAR1 = 0x00000000 BAR2 = 0x00000000 BAR3 = 0x00000000 Cardbus CIS Pointer = 0x00000000 SubSys Vender ID = 0x2474 Expansion LOM Base = 0x00000000 Capability List = 0x50 Int Line = 0x00 Min Gnt = 0x00 Max Latency = 0x00	Rev ID	= 0x09
Cache Line Size = 0x10 Mst Latency Timer = 0x00 Header Type = 0x00 BIST = 0x00 BAR0 = 0x90000000 BAR1 = 0x00000000 BAR2 = 0x00000000 BAR3 = 0x00000000 BAR5 = 0x00000000 Cardbus CIS Pointer = 0x00000000 SubSys Vender ID = 0xa106 SubSys ID = 0x2474 Expansion LOM Base = 0x0000000 Capability List = 0x50 Int Line = 0x00 Min Gnt = 0x00 Max Latency = 0x00 Please Input Enter Key:	Class Code	= 0xtt0000
Mst Latency Timer = 0x00 Header Type = 0x00 BIST = 0x00 BAR0 = 0x9000000 BAR1 = 0x00000000 BAR2 = 0xa0000000 BAR3 = 0x00000000 BAR3 = 0x00000000 Cardbus CIS Pointer = 0x00000000 Cardbus CIS Pointer = 0x00000000 SubSys Vender ID = 0xa106 SubSys ID = 0x2474 Expansion LOM Base = 0x00000000 Capability List = 0x50 Int Line = 0x00 Min Gnt = 0x00 Max Latency = 0x00 Please Input Enter Key:	Cache Line Size	= 0x10
Header Type       = 0x00         BIST       = 0x00         BAR0       = 0x0000000         BAR1       = 0x0000000         BAR2       = 0x0000000         BAR3       = 0x0000000         BAR4       = 0x0000000         BAR5       = 0x0000000         Cardbus CIS Pointer       = 0x0000000         SubSys Vender ID       = 0x2474         Expansion LOM Base       = 0x00000000         Capability List       = 0x50         Int Line       = 0x01         Min Gnt       = 0x00         Max Latency       = 0x00         Please Input Enter Key:	Mst Latency Timer	
BISI       = 0x00         BAR0       = 0x9000000         BAR1       = 0x0000000         BAR2       = 0x0000000         BAR3       = 0x0000000         BAR4       = 0x0000000         BAR5       = 0x0000000         Cardbus CIS Pointer       = 0x0000000         SubSys Vender ID       = 0x2474         Expansion LOM Base       = 0x00000000         Capability List       = 0x207         Int Line       = 0x00         Min Gnt       = 0x00         Max Latency       = 0x00	Header Type	= 0x00
BAR0       = 0x90000000         BAR1       = 0x00000000         BAR3       = 0x00000000         BAR4       = 0x00000000         BAR5       = 0x00000000         Cardbus CIS Pointer       = 0x00000000         Cardbus CIS Pointer       = 0x00000000         SubSys Vender ID       = 0xa106         SubSys ID       = 0x2474         Expansion LOM Base       = 0x0000000         Capability List       = 0x50         Int Line       = 0x00         Min Gnt       = 0x00         Max Latency       = 0x00	B151	= 0,000
BAR1       = 0x0000000         BAR2       = 0x0000000         BAR3       = 0x0000000         BAR4       = 0x0000000         BAR5       = 0x0000000         Cardbus CIS Pointer       = 0x0000000         Cardbus CIS Pointer       = 0x0000000         SubSys Vender ID       = 0xa106         SubSys ID       = 0x2474         Expansion LOM Base       = 0x0000000         Capability List       = 0x50         Int Line       = 0x00         Min Gnt       = 0x00         Max Latency       = 0x00         Please Input Enter Key:	PAD1	
BAR2       = 0x80000000         BAR3       = 0x00000000         BAR4       = 0x00000000         BAR5       = 0x00000000         Cardbus CIS Pointer       = 0x00000000         Subsys Vender ID       = 0x2474         Expansion LOM Base       = 0x00000000         Capability List       = 0x50         Int Line       = 0x01         Min Gnt       = 0x00         Max Latency       = 0x00         Please Input Enter Key:	RAD2	
BAR3       = 0x00000000         BAR4       = 0x000000000         BAR5       = 0x000000000         Cardbus CIS Pointer       = 0x00000000         SubSys Vender ID       = 0x2474         Expansion LOM Base       = 0x000000000         Capability List       = 0x2474         Expansion LOM Base       = 0x00000000         Capability List       = 0x20         Int Line       = 0x00         Min Gnt       = 0x00         Max Latency       = 0x00         Please Input Enter Key:	RADA	- 0x80000000
BAR5       = 0x00000000         Cardbus CIS Pointer = 0x00000000         SubSys Vender ID       = 0xa106         SubSys ID       = 0x2474         Expansion LOM Base       = 0x00000000         Capability List       = 0x50         Int Line       = 0x00         Min Gnt       = 0x00         Max Latency       = 0x00         Please Input Enter Key:	BAD4	- 0x0000000
Dradbus CIS Pointer = 0x00000000         SubSys Vender ID = 0x106         SubSys ID = 0x2474         Expansion LOM Base = 0x00000000         Capability List = 0x50         Int Line = 0x0b         Int Pin = 0x01         Min Gnt = 0x00         Max Latency = 0x00         Please Input Enter Key:	BARS	- 0x00000000
SubSys Vender ID       = 0x030000         SubSys ID       = 0x2474         Expansion LOM Base       = 0x00000000         Capability List       = 0x50         Int Line       = 0x00         Min Gnt       = 0x00         Max Latency       = 0x00         Please Input Enter Key:	Cardbus CIS Pointer	= 0x00000000
SubSys       ID       = 0x2474         Expansion LOM Base       = 0x00000000         Capability List       = 0x50         Int Line       = 0x0b         Int Pin       = 0x01         Min Gnt       = 0x00         Max Latency       = 0x00         Please Input Enter Key:	SubSys Vender ID	= 0xa106
Expansion LOM Base = 0x00000000 Capability List = 0x50 Int Line = 0x0b Int Pin = 0x01 Min Gnt = 0x00 Max Latency = 0x00 Please Input Enter Key:	SubSys ID	= 0x2474
Capability List = 0x50 Int Line = 0x0b Int Pin = 0x01 Min Gnt = 0x00 Max Latency = 0x00 Please Input Enter Key:	Expansion LOM Base	= 0x0000000
Int Line = 0x0b Int Pin = 0x01 Min Gnt = 0x00 Max Latency = 0x00 Please Input Enter Key:	Capability List	= 0x50
Int Pin = 0x01 Min Gnt = 0x00 Max Latency = 0x00 Please Input Enter Key:	Int Line	= 0x0b
Min Gnt = 0x00 Max Latency = 0x00 Please Input Enter Key:	Int Pin	= 0x01
Max Latency = 0x00 Please Input Enter Key:	Min Gnt	= 0x00
Please Input Enter Key:	Max Latency	= 0×00
Please Input Enter Key:		
Please Input Enter Key:		
	Please Input Enter	Key:

図 7-12 pcidump コマンドの実行例



### 7.5.7. cfgread コマンド

本コマンドはアドレスを指定して PCI コンフィグレーション空間の読み出しが可能です。

Usage: cfgread <target(0:end/1:root)> <offset(HEX)>

- コマンド名: cfgread
- 第1引数:読み出し対象の選択
  - D: エンドポイント
  - 1: ルート・コンプレックス
- 第2引数: 読み出し対象の PCI コンフィグレーション空間のオフセットの選択

以下の例ではエンドポイント側の PCI コンフィグレーション空間の 0x0 番地を読み出した結果です。ベンダ ID である 0x1172 とデバイス ID の 0xe001 が読み出せていることが確認できます。

/ \$ / 1 \$\$ \$ /\$\$   \$\$ \$ /\$\$   \$\$\$ \$ /\$\$   \$\$5 \$ /\$\$\$   \$\$55 /\$\$5   \$\$ 55 5 /\$\$5   \$\$ 55 55 55 1 / \$\$ 55 55 5 5 1 / \$\$ 55 55 5 5 1 / \$\$ 55 55 55 55 1 / \$\$ 55 55 55 55 1 / \$\$ 55 55 55 55 1 / \$\$ 55 55 1
\$\$\$\$\$\$/
manu y manu
led · led coeriod(ms)> ctimes>
memdump : memdump ctarget(0:ddr/1:ocram)> coffset(HEX)> csize(HEX)>
memory i memory carget(0.dd//1.ocram) correct(nEx) < Stretc(nEx)
pridump : pridump
cforead < cforead <target(0:end 1:root)=""> <offset(uex)></offset(uex)></target(0:end>
harroad · harroad RADEA(1/2 > coffset(HEX)>
barried , barried $AR(0)/1/2$ , $Arrive (REY)$ , $Adata(HEY)$
avit avit
* Note: HEY Value does not need ay
Note. HEA Value does not need of
Command: cfgread 0 0 cfgread 0 0
cfgread cfgread: End read_data: e0011172
Please Input Enter Key:

図 7-13 cfgread コマンドの実行例



### 7.5.8. barread コマンド

本コマンドを使用して指定した BAR の番号とオフセットに対してリードを実行可能です。

Usage: barread <BAR:0/1/2..> <offset(HEX)>

- コマンド名: barread
- 第1引数: BAR 番号の選択
- 第2引数: BAR の先頭からのオフセットをの指定

以下は BAR2:0x4000 オフセットにある LED 用の PIO ペリフェラルにアクセスした際の例です。

/ \$ / [ \$\$ \$ /\$\$   \$\$\$ \$ /\$\$   \$\$\$ \$ /\$\$\$  \$ / \$ / \$ / \$ / \$ / [ \$\$\$\$ \$ /\$\$\$   555555   5555555   55555555   5555555	
<pre>menu : menu led : led <period(ms)> <times> memdump : memdump <target(0:ddr 1:ocram)=""> <offset(hex)> <size(hex)> memtest : memtest <target(0:ddr 1:ocram)=""> <mode(0:fixed 1:incremental)=""> pcidump : pcidump cfgread : cfgread <target(0:end 1:root)=""> <offset(hex)> barread : barread <bar:0 1="" 2=""> <offset(hex)> barread : barreid <bar:0 1="" 2=""> <offset(hex)> cati : exit * Note: HEX Value does not need 0x</offset(hex)></bar:0></offset(hex)></bar:0></offset(hex)></target(0:end></mode(0:fixed></target(0:ddr></size(hex)></offset(hex)></target(0:ddr></times></period(ms)></pre>	
Command: barread 2 4000 barread 2 4000	
barX_read read_data: 0000000f	
Please Input Enter Key:	

図 7-14 barread コマンドの実行例



#### 7.5.9. barwrite コマンド

本コマンドを使用して指定した BAR の番号とオフセットに対してライトを実行可能です。

Usage: barwrite <BAR:0/1/2..> <offset(HEX)> <data(HEX)>

- コマンド名: barwrite
- 第1引数: BAR 番号の選択
- 第2引数: BAR の先頭からのオフセットを指定
- 第3引数:書き込み対象のデータ(32bit)

以下は BAR2:0x4000 オフセットに実装される LED 用の PIO ペリフェラルに対して 0xF をライトした際の 例です。

•••••	
/ S / S SS S /SS SSS S /SSS SSSS /SSSS SS SS SS/SS SS   SS/SS SS   S/ SS SS / SS/	   /
menu	Menu
led	led <period(ms)> <times></times></period(ms)>
memdump	<pre>memdump <target(0:ddr 1:ocram)=""> <offset(hex)> <size(hex)></size(hex)></offset(hex)></target(0:ddr></pre>
memtest :	<pre>memtest <target(0:ddr 1:ocram)=""> <mode(0:fixed 1:incremental)=""></mode(0:fixed></target(0:ddr></pre>
pcidump :	pcidump
cfgread :	cfgread <target(0:end 1:root)=""> <offset(hex)></offset(hex)></target(0:end>
barread :	barread <bar:0 1="" 2=""> <offset(hex)></offset(hex)></bar:0>
barwrite :	barwrite <bar:0 1="" 2=""> <offset(hex)> <data(hex)></data(hex)></offset(hex)></bar:0>
exit :	exit
* Note: HE>	Value does not need 0x
Command: bar barwrite 2 4	write 2 4000 f 1000 f
barX write	
write data:	0000000
data.	
Please Input	Enter Key:

図 7-15 barwrite コマンドの実行例



## 7.5.10. exit コマンド

本コマンドを使用するとユーザ・アプリケーションが終了します。

Usage: exit

- コマンド名: exit
- 引数なし

/ - 5 / - 1 55 5 /555   5 / 5 / 1 5555 /5555   5555555   /5555555   /5555555   /5555555/ 55 55 55/55   / 55   55   55   55   55
<pre>menu : menu led : led <period(ms)> <times> memdump : memdump <target(0:ddr 1:ocram)=""> <offset(hex)> <size(hex)> memtest : memtest <target(0:ddr 1:ocram)=""> <mode(0:fixed 1:incremental)=""> pcidump : pcidump cfgread : cfgread <target(0:end 1:root)=""> <offset(hex)> barread : barread <bar:0 1="" 2=""> <offset(hex)> barwrite : barwrite <bar:0 1="" 2=""> <offset(hex)> <data(hex)> exit : exit * Note: HEX Value does not need 0x</data(hex)></offset(hex)></bar:0></offset(hex)></bar:0></offset(hex)></target(0:end></mode(0:fixed></target(0:ddr></size(hex)></offset(hex)></target(0:ddr></times></period(ms)></pre>
Command: exit
exit
Exit loop
magnes@magnes:~/work/magnes_pcle\$

図 7-16 exit コマンドの実行例



# 7.6. ドライバのアンインストール

ドライバのアンインストールには以下のコマンドを使用します。

\$ make remove

```
magnes@magnes:~/work/magnes_pcie$ make remove
sudo rmmod magnes_drv
```

# 7.7. OS のシャットダウン

OS のシャットダウンを実施するには画面右上の 🍄 アイコンより「シャットダウン」を選択します。



図 7-17 シャットダウン・メニュー



# 7.8. トランシーバ・ブロックの確認

当ボードで 10.8Gbps のトランシーバ・ブロックをコントロールするための手順を解説します。 (10.8Gbps シリアル・データには 64B66B エンコード/デコードが適用されております)

## 7.8.1.トランシーバ・ブロックの確認準備

本手順の開始前に PC と USB Type A to Type B ケーブルが CN16 に接続されていることを確認し、UART 経由でコントロールするためのターミナル・ソフトを用意してください。 (USB Type A to Type B ケーブルは同梱されておりませんのでご注意ください)



図 7-18 CN16 USB UART コネクタ

なお、本マニュアルではターミナル・ソフトは Tera Term を使用しています。 以下は設定例です。ポート番号は使用している PC にあわせてください。(改行コードの設定にご注意ください。)

Tera Term: シリアルボート 設定 🔀	Tera Term: 端末の設定
ボート(P): <u>COM11</u> OK スピード(E): 115200 ● データ(D): 3 bit ● <u>キャンセル</u> パリティ(A): none ● ストップビット(S): 1 bit ● ヘルブ(H)	端末サイズ(I): 図7 × 57 マ = ウィンドウサイズ(S): □ 自動的に調整(W): 端末D(I): ▼T100 ▼ □ ローカルエコー(L): ○ 次行コード ⑦(行コード ◎(「コード ◎(「コード ◎(「コード ◎(「コード ◎(「コード ◎(「コード ◎(「コード ◎(「コード ◎(「コード ◎(「コード ◎(「コード) ◎(「コード ◎(「コード ◎(「コード ◎(「コード ◎(「コード ◎(「コート) ◎(「コード ◎(「コード ◎(「コード) ◎(「コード ◎(「コード) ◎(「コード ◎(「コード) ◎(「コー) ◎(□)
フロー制御(F): none 💌	応答(A): 「自動切り替え(VT<->TEK)(U):
送信遅延 0 ミリ秒/字(C) 0 ミリ秒/行(L)	漢字-受信(K) UTF-8 ● 「Zbit 力タカナ 漢字マケ(M) 「&B ● 「7bit 力タカナ 漢字アウト(G) 「(B ●
	ロケール( <u>C</u> ): japanese 言語コード( <u>P</u> ): 932

図 7-19 Tera Term の設定例



## 7.8.2.トランシーバのコントロール・コンソール

トランシーバをコントロールするためのコンソールについて解説します。

Channel0~9 は FPGA-FMC コネクタ(HPC)、 Channel A は FPGA-FMC コネクタ(LPC)にピン・アサインして います。

Number of received bits[63:32], [31:0]で受信ビット数をカウントし、Number of received Error bits[63:32], [31:0]で受信エラービットをカウントします。

PMA Internal Serial Loopback はトランシーバ内部でトランスミッタ⇒レシーバーでループバックさせるデータ パスを有効/無効をコントロールします。

初期設定では全てのトランシーバが無効になっており、FPGA-FMC コネクタがオープン状態でトランシーバへの受信ビットを確認するためにはループバックパスを有効にする必要があります。

「Select Action」以下がトランシーバをコントロールするコマンドになり、0 でデータ・パターン/チェッカーを有効、 1 でデータ・パターン/チェッカー無効かつ、受信ビットカウンタをクリアします。最後に2 でループバックパスの有 効/無効を切り替えます。なお、コマンド 0~2 の決定後、チャネル番号 0~a を指定する必要があります。



図 7-20 実行例



## 7.9. HDMI 出力の確認

当ボードには FPGA 側にも HDMI インタフェースが搭載されておりリファレンス・デザインではカラーパター ンが出力されます。出力される解像度は FullHD 1920 x 1080 となります。 上記をサポートするモニタと当ボードの CN6 を HDMI ケーブルにて接続いただくことでカラーパターンの出 力をご確認いただけます。



図 7-21 FPGA-HDMI 出力コネクタ



図 7-22 カラーパターンの出力

リファレンス・デザイン内のカラーパターン出力には VIP を使用しており、本 IP を使用するには別途ライセン スが必要です。当ボードに付属のシリアルナンバーをお知らせいただければ 60 日間の評価ライセンスを発 行いたします。

ライセンス発行が必要な場合には、販売代理店までお問い合わせください。



# 8. 参考情報

- インテル<sup>®</sup> FPGA の開発フロー/FPGA トップページ <u>https://service.macnica.co.jp/library/109705</u>
- インテル<sup>®</sup> FPGA で PCI Express <u>https://service.macnica.co.jp/library/118473</u>
- FPGA 経由で EPCQ デバイスヘプログラミング(JIC プログラミング) https://service.macnica.co.jp/library/122065



# 9. 更新履歴

日付	版	更新概要
2018年8月9日	1.0	初版