

2016/06/21

mxcard で始めるシステム開発 システム開発ボードの使い方

Prime Systems, Inc.

mxcard で始めるシステム開発

システム開発ボードの使い方

Smart-USB Plus 製品ファミリの「mxcard システム開発ボード」は、アルテラ社の MAX10 FPGA デバイスや USB2.0、不揮発性メモリの FRAM を利用して、各種データ収集等に利用できる製品です。

ここでは、実践的なシステム開発へのファーストステップとして、ボードの使い方と応用例を紹介します。

準備

デバイスドライバのインストール

製品付属の CD-ROM¥Software¥DPInst_Plus フォルダから、お使いの Windows バージョン (32bit または 64bit)に応じたフォルダ内にある"dpinst.exe"を実行して下さい。ここではボード を PC に USB 接続しないで下さい。

インストールの詳細はアプリケーションノート"SUA009.pdf"P.2<WinUSB デバイスドライバのイン ストール>を参照して下さい。

制御アプリのインストール

mxcard を制御するための制御アプリには 2 種類あります。Smart-USB Plus/Sigma 製品すべてに 共通の RefApp7.exe と mxcard 専用の RefMX,exe です。

RefMX.exe では、USB 経由で FPGA のコンフィグができます(メーカ専用の JTAG ダウンロードケ ーブル(例:USB Blaster)は不要です)。また、収集したアナログ波形をリアルタイムに画面表示 する波形ビューワを利用できます。

	RefApp7	RefMX
FPGA のコンフィグ(SRAM 領域)	○ (高速)	🔿 (JTAG)
コンフィグ ROM へのプログラム	×	🔿 (JTAG)
レジスタアクセス	0	0
拡張レジスタアクセス	0	×
メモリアクセス	0	0
TCL スクリプト機能	0	×
SmartAnalogIC(RAA730301)制御	×	0
アナログ波形ビューワ機能	×	0

<表 1. USB で制御できる機能 RefApp7 vs. RefMX>

RefApp7.exe と RefMX.exe を両方インストールすることもできますが、同時に起動するとどちらかのアプリしか動作しない現象が発生します。

RefApp7.exe をインストールする場合

製品付属の CD-ROM¥Software¥RefApp7_setup¥ RefApp7_setup.exe を実行して下さい。 Windows 32bit/64bit 共通です。

※インストールの詳細はアプリケーションノート"<u>SUA009.pdf</u>"P.4<RefApp7 のインストール>を参照して下さい。

インストール後には必ずアップデートを行って下さい。2016.6月現在、最新版は1.1.1.0です。

\varTheta Smart-USB Series : Refe	erence Application7 Ver1.1.1.0	- • •
ファイル(X) オプション(Y) ヘルプ(Z	Z)	
─転送レート測定│温度モニタ	GPIO FlashPath2 GPIF設定 拡張レジスタ	レシ [、] スタ操作(512)
レシ`スタ操作 メモリ操	作 USBコンフィクジ ボード情報 I2C操作	TCLスクリフト
ーレジスタ番号 -ビット幅	データ設定	
2 sbit v	00 00 (HEX)	Read Write
	データ設定	
3 × 8bit ×	00 00 (HEX)	Read Write
しシズタ番号 ビット幅	データ設定	
4	00 00 (HEX)	Read Write
ロケView再表示	7769Dutae1	終了
$\mp 55 \times 10^{-1}$ $\rightarrow 32.00 \text{KiBytes}[3]$	2706Bytesj	

<図 1. RefApp7 起動画面(ボード未接続)>

RefMX.exe をインストールする場合

製品付属の CD-ROM¥Software¥RefMX フォルダから、使用する Windows バージョン(32bit また は 64bit)に対応したフォルダにあるセットアップファイルを実行して下さい。例えば Windows 32bit 版なら、¥RefMX32_setup_1000¥ RefMX32_setup.exe です。

🔎 RefMX Ver:	1.0.0.0 for V	Vin32				
ファイル(Y) Opt	ion ∿⊮7°(Z)				
レジスタ操作	メモリ操作	USBコンフィク	FlashPath	ボード情報	RAA730301	
Reg No	BIT Width	Read/Writ	e DATA Set	ting		
2	8bit 👻	00		01	(HEX)	Read Write
Reg No	BIT Width	Read/Writ	e DATA Set	ting		
3	8bit 👻	00		00	(HEX)	Read Write
Reg No	BIT Width	Read/Writ	e DATA Set	ting		
4	8bit 👻	00		00	(HEX)	Read Write
ロクView再表示	ħ					終了

<図 2. RefMX 起動画面(ボード未接続)>

mxcard ボート PC を USB 接続する

デバイスドライバと制御アプリをインストールできたら、PC と mxcard ボードを USB ケーブルで接続して下さい。ボードの3つの LED が緑に点灯したら、制御アプリからボード制御ができます。



左図、黄と青枠内に緑点灯の LED を配置しています。

青枠の LED は、FPGA のコンフィグ状態を示します。緑に 点灯していると FPGA にプログラムした回路が構成されて いる状態(使える状態)です。

右上の LED は、PC がボードを認識して制御アプリからボ ード制御が可能な状態を示します。緑に点灯していない場 合はデバイスドライバが正しくインストールできていない

可能性がありますので、再度確認して下さい。

左下の LED はボードに電源投入されると緑に点灯します。

RefMX.exe で FPGA をコンフィグする

ボード制御の準備ができたら、RefMX.exe を起動して FPGA をコンフィギュレーションできます。 製品出荷時には MAX10-FPGA にサンプル回路をコンフィグ済みですが、当社サーバから最新のサン プル FPGA 回路をダウンロードして FPGA にコンフィグ可能です。

MAX10-FPGA はコンフィグ(プログラム)用の FlashROM を内蔵したデバイスです。電源投入時、 この FlashROM から SRAM コンフィグビットに回路データをダウンロードして回路を構成する仕組 みです。

MAX10の SRAM 領域にコンフィグ

RefMX.exe では、内蔵 FlashROM にコンフィグせず、直接 SRAM コンフィグビットをプログラムできます。その際、SVF ファイルを使用します。以下に SVF ファイルの生成方法を示します。

SVF 生成方法

Quartus Prime ツールが生成する SOF ファイルから SVF ファイルを生成します。

- 1. Quatus Prime の Programmer を起動します。図1参照。
- 2. 図1で示す赤枠内に、生成した SOF ファイルをクリック&ドローしてください。
- 3. ツールバーの「File」から「Create JAM, JBC, SVF or ISC File...」を選択。図2参照。

4. 図 5 で示す様に File name を指定し、File Format を.svf 、TCK Frequency を 840KHz に 設定して下さい。OK ボタンをクリックすると File name で指定した SVF ファイルが生成でき ます。



<図 3. Programmer 起動画面>

Programmer - D:/project/qdesigns	/mxcard/GPIF_Verilog/GPIF_V		
File Edit View Processing Tools	Window Help 🤜		
New File Close	Ctrl+N	Create JAM, JBC, SVF	or ISC file 📃 💌
Save Save Create JAM, JBC, SVF or ISC File	Ctrl+S Device Ct	File name: GPIF_Verilog sv File format Serial Vector Fo Operation Program Verify	f Programming options Blank-check Verify
★ Delete → Add File		Clock frequecy TCK frequency 840 kHz Supply voltage: 3.3 volts	
<凶 4.5VF ノアイル生成選択>			OK Cancel

<図 5. 各種設定>

RefMX.exe でのコンフィグ手順(USB コンフィグ)

PC とボードを USB 接続し、RefMX.exe を起動します。「USB コンフィグ」タブを選択して生成した SVF ファイルを選択して下さい。

🖉 RefMX Ver1.0.0.0 for Win32		
7ァイル(Y) Option ヘルプ(Z)		
レシズタ操作 メモリ操作 USBコンフィク FlashPat	h ボード情報 RAA730301	
MAX10コンフィク [*] ファイル選択: E:¥DATA-CDROM¥SmartUSBPlus¥mxcard¥FPG/	A¥動作確認¥GPIF_Verilog_sof.svf	ファイル選択
PartName \rightarrow 10M08DA Manufacturer ID \rightarrow ALTERA Version \rightarrow 0h	SVF Execution Status \rightarrow - Config Status \rightarrow - Config Time \rightarrow -	
Device Detect		Configure
ロりView再表示		終了

<図 6. RefMX USB コンフィグ画面>

コンフィグファイル選択後、「Device Detect」ボタンをクリックして下さい。図6で示すように接続しているデバイス型番が表示されます。確認できたら「Configure...」ボタンをクリックしてコンフィグ動作が始まります。 図6の例では、約5秒程度でコンフィグが完了します。

SVF ファイルを使用した USB コンフィグの場合、ボードの電源を OFF にすると FPGA にコンフィグ したデータは揮発します。また、USB コンフィグすると、すでにコンフィグ用 FlashROM にプログ ラムしたコンフィグデータも初期化されて消えてしまいます。

MAX10の FlashROM 領域にコンフィグ

RefMX.exe では、アルテラ社 JTAG ケーブル USBBlaster を利用せずに、USB 経由で内蔵 FlashROM にコンフィグできます。その際、RPD ファイルを使用します。以下に RPD ファイルの生 成方法を示します。

RPD 生成方法

Quartus Prime ツールが生成する SOF ファイルから RPD ファイルを生成します。

1. Quartus Prime ツールバー「File」→「Convert Programming File」を選択します。

- 図 7 に示すように、Programming file type を.pof 、Mode を Internal Configuration に設定し、Create config data RPD にチェックを入れてください。File name 欄のファイル名は任意に設定して下さい。設定したファイル名で RPD ファイルが生成されます。
- 3. 図8では、①欄をクリックしてから②の「Add File...」ボタンをクリックして、③の sof ファイ ルを選択してください。最後に「Generate」ボタンをクリックして RPD ファイルを生成します。

生成された RPD ファイルは3つあります。File name 欄で指定したファイル名(例では output_file) に定型のファイル名称が追加される形式です。

> output_file_auto.rpd (このファイルだけ使用します) output_file_cfm0_auto.rpd (使用しないファイル) output_file_ufm_auto.rpd (使用しないファイル)

Tools Window			Search altera.com
ecify the input files to convert a u can also import input file infor ure use. Conversion setup files	nd the type of programming file to generat mation from other files and save the conv	te. ersion setup information created h	ere for
Open Cor	nversion Setup Data	Save	e Conversion Setup
utput programming file			
rogramming file type: Program	nmer Object File (.pof)		▼
Options/Boot info Configur	ation device: EPCE16	▼ Mode:	Internal Configuration
ile name: Output	file.pof		
Advanced Remote/I	Local update difference file: NC te Memory Map File (Generate output_file.	map)	▼
Advanced Remote/I	Local update difference file: [NC te Memory Map File (Generate output_file. te CvP files (Generate output_file.periph.p te config data RPD (Generate output_file_	map) of and output_file.core.rbf) auto.rpd)	•
Advanced Remote/I	Local update difference file: NC te Memory Map File (Generate output_file.) te CvP files (Generate output_file.periph.p te config data RPD (Generate output_file_ Properties	map) of and output_file.core.rbf) auto.rpd) Start Address	▼ Add Hex Data
Advanced Remote/I	Local update difference file: NC te Memory Map File (Generate output_file.) te CvP files (Generate output_file.periph.p te config data RPD (Generate output_file_ te config data RPD (Generate output_file_ Properties Page_0	map) of and output_file.core.rbf) auto.rpd) Start Address <auto></auto>	Add Hex Data Add Sof Page
Advanced Remote/I	Local update difference file: MC te Memory Map File (Generate output_file.) te CvP files (Generate output_file.periph.p te config data RPD (Generate output_file_ te config data RPD (Generate output_file_ Properties Page_0	map) of and output_file.core.rbf) auto.rpd) Start Address <auto></auto>	Add Hex Data Add Sof Page Add File
Advanced Remote/I	Local update difference file: [NC te Memory Map File (Generate output_file.) te CvP files (Generate output_file.periph.p te config data RPD (Generate output_file_i Properties Page_0	map) of and output_file.core.rbf) auto.rpd) Start Address <auto></auto>	Add Hex Data Add Sof Page Add File Remove
Advanced Remote/I	Local update difference file: MC te Memory Map File (Generate output_file. te CvP files (Generate output_file.periph.p te config data RPD (Generate output_file_i Properties Page_0	map) of and output_file.core.rbf) auto.rpd) Start Address <auto></auto>	Add Hex Data Add Sof Page Add File Remove
Advanced Remote/I	Local update difference file: MC te Memory Map File (Generate output_file.) te CvP files (Generate output_file.periph.p te config data RPD (Generate output_file_ Properties Page_0	map) of and output_file.core.rbf) auto.rpd) Start Address <auto></auto>	Add Hex Data Add Sof Page Add File Remove Up
Advanced Remote/I	Local update difference file: [NC te Memory Map File (Generate output_file.) te CvP files (Generate output_file.periph.p te config data RPD (Generate output_file_j Properties Page_0	map) of and output_file.core.rbf) auto.rpd) Start Address <auto></auto>	Add Hex Data Add Sof Page Add File Remove Up Down

<図 7. SOF から RPD ファイルを生成するための設定画面>

10013 1111001						S	earch altera	.com
cify the input files can also import re use.	to convert and the Select Input	ne tvoe of programming file to ger ut File	nerate.		er to			x
onversion setup	Look in:	D:\project\qdesigns\mxcard\	GPIF_Verilog\output_files		•	000		
	🔊 My Com	nputer Name	*	Size	Туре	Date Modifie	ed	
utput programmin ogramming file t	TOOL	greybox_tmp GPIF_Verilog.sof	>	728 KB	Filder sof File	2016/02/23 2016/03/07	20:42:23 17:48:09	
Options/Boot infi	File name:						Ope	n
e name:	Files of type: (SRAM Object Files (*.sof)				-	Canc	el
Advanced	Charles and the second second							
	Create Me	emory Map File (Generate output_ vP files (Generate output_file.peri onfig data RPD (Generate output_	_file.map) iph.pof and output_file.core. file_auto.rpd)	rbf)				
out files to conver	Create M Create C Create C Create co	emory Map File (Generate output_ vP files (Generate output_file.peri onfig data RPD (Generate output_	_file.map) iph.pof and output_file.core. file_auto.rpd)	rbf)				
out files to conver File/I	Create M Create C Create C Create c t t	emory Map File (Generate output_ /P files (Generate output_file.peri onfig data RPD (Generate output_ Properties	_file.map) iph.pof and output_file.core. file_auto.rpd) Start Address	rbf)			Add	I Hex Data
out files to conver File/I SOF Data	Create Mo Create Co Create Co Create co Create co t	emory Map File (Generate output /P files (Generate output_file.peri onfig data RPD (Generate output_ Properties_ Page_0	_file.map) iph.pof and output_file.core. file_auto.rpd) Start Address <auto></auto>	rbf)			Add	Hex Data
out files to conver File/I SOF Data	Create M Create C Create C Create cc Create cc t	emory Map File (Generate output_ vP files (Generate output_file.peri onfig data RPD (Generate output_ Properties_ Page_0	_file.map) iph.pof and output_file.core. file_auto.rpd) Start Address <auto></auto>	rbf)			Add	Hex Data
out files to conver File/I SOF Data	Create Ma Create Co Create Co Create co t Data area	emory Map File (Generate output /P files (Generate output_file.peri onfig data RPD (Generate output_ Properties_ Page_0	_file.map) iph.pof and output_file.core. file_auto.rpd) Start Address <auto></auto>	rbf)			Add Add A	Hex Data Sof Page dd File
out files to conver File/I SOF Data	Create Mo Create Co Create Co Create co t Data area	emory Map File (Generate output_ rP files (Generate output_file.peri onfig data RPD (Generate output_ 	_file.map) iph.pof and output_file.core. file_auto.rpd) Start Address <auto></auto>	rbf)			Add Add F	Hex Data Sof Page dd File temove Up
out files to conver File/I SOF Data	Create Ma Create Co Create Co Create co t	emory Map File (Generate output /P files (Generate output_file.peri onfig data RPD (Generate output_ Properties_ Page_0	_file.map) iph.pof and output_file.core. file_auto.rpd) Start Address <auto></auto>	rbf)			Add Add	l Hex Data Sof Page dd File temove Up Down
Nut files to conver File/I SOF Data	Create Ma Create Co Create Co Create co t Data area	emory Map File (Generate output /P files (Generate output_file.peri onfig data RPD (Generate output_ Properties_ Page_0	_file.map) iph.pof and output_file.core. file_auto.rpd) Start Address <auto></auto>	rbf)			Add Add	I Hex Data Sof Page dd File Lemove Up Down operties

<図 8. SOF から RPD ファイルを生成する画面>

RefMX.exe でのコンフィグ手順(FlashPath)

PC とボードを USB 接続し、RefMX.exe を起動します。「FlashPath」タブを選択して生成した

RPD ファイルを選択して下さい(例では、output_file_auto.rpd)。

💋 RefMX Ver1.0.0.0 for Win32				
7ァイル(Y) Option ヘルプ(Z)				
レシ [*] スタ操作 メモリ操作 USBコンフィク [*] FlashPath	ホ [*] ート [*] 情報 RAA730301	FP		
プログラミングデータファイル選択(SOF-NAME_auto.rpd):				
E:¥DATA-CDROM¥SmartUSBPlus¥mxcard¥FPGA¥	「動作確認¥output_file_auto.rpd」ファイル選択	す。		
PartName → 10M08DA	Execute Time \rightarrow -			
Manufacturer ID \rightarrow ALTERA		<u>ت</u>		
Version \rightarrow 0h		(_L		
		に、		
0 %	Programing			
ロケView再表示	終了			

FPGA 型番が 10M08DA の場 合、約 9 秒程度で完了しま す。

ここでコンフィグしたデータ は、電源断後も揮発しません。

<図 9. RefMX FlashPath 画面>

アナログ機能を使う

MAX10-FPGA には、12bit 1MS/sの AD コアが内蔵されています。デバイスタイプにより、1 個または 2 個の AD コアが利用できます。

サンプル FPGA 回路では、1 個の AD コアを利用し、CH1 と CH8 を有効にしています。mxcard ボ ード出荷時設定では、CH8 はプログラマブル AMP に接続されているので、ボードの CN2.4 ピンに アナログ信号を入力してください。



サンプル回路ファイル名:mxcard_GPIF_Verilog_160620.zip

<図 10. アナログ入力経路>

SmartAnalogIC (RAA730301)の設定

RefMA.exe の「RAA730301」タブを選択して下さい。この画面の「設定ファイル」タブから、製品 添付の CD-ROM¥FPGA¥動作確認¥smartanalog 設定ファイル フォルダにある "r_sadesigner_reg.c"ファイルを選択して下さい。「一括設定」ボタンをクリックして SmartAnalogIC をゲイン 6dB に設定できます。 設定データは電源断やリセットで揮発します。

S RefMX Ver1.0.0.0 for Win32	
7ァイル(Y) Option ヘルプ(Z)	
レシブスタ操作 メモリ操作 USBコンフィクジ FlashPath ホートジ情報 RAA730301	
Common AMP DAC TEMP Sensor/LDO 設定ファイル	
設定ファイル:r_sadesigner_reg.c	
E:¥Project¥SmartUSBPlus¥mxcard¥r_sadesigner_reg.c	ファイル選択
	一括設定
L	
	<u><u></u><u></u><u></u><u></u><u></u><u></u><u></u><u></u><u></u><u></u><u></u><u></u><u></u><u></u><u></u><u></u><u></u><u></u><u></u></u>
	ारः ।

<図 11. SmartAnalogIC(プログラマブル AMP)設定画面>

8•

アナログ波形のリアルタイム表示

mxcard のコネクタ CN2.4 ピンに入力したアナログ波形をリアルタイムで表示させるには、レジス タ No.4(16bit)の[0]ビットに"1"を書き込んで下さい。次に、RefMX.exe のツールバー「Option」 から波形 View 表示を選択します。

💋 RefMX Ve	er1.0.0.0 for Win64		
7711(Y) O	ption <pre>\N/7°(Z)</pre>		
レシェスタ損	LOGフォルダを開く(L)	-ド情報 RAA730301	
-Reg No-	波形フォルダを開く(W)		
2	ログ View表示(V)	(HEX)	Read Write
Reg No	波形View表示(I)		
3	メモリリード時にスクランブル化を行う(X) メモリライト時にスクランブル化を行う(Y)	(HEX)	Read Write
Reg No	拡張レジスタアクセスをエミュレートする(Z)		
4	16bit • 0001	0001 (HEX)	Read Write
ロクView再表	ফন		終了
レジスリード成功			

<図 12. アナログ波形表示設定>

Ø WaveForm 4,000 3,500 735 3,000 824 930 1,025 1,123 1,228 1,338 1,445 1,543 1,638 1,716 1,778 2,500 2,000 1,500 1,000 500 n 1,800 200 400 600 800 1,000 1,200 1,400 1,600 2,000 解析情報: 取込データバイト数: FileSize: -2000 (HEX) 有効ADCチャンネル数:2 CH 初期表示 更新回数: 2225 データ数 ID データ数 ID NAME NAME 波形保存 閾値: -0 CH00 2048 8 CH08 2048 -- 波形ON/OFF --CH01 0 CH09 0 9 1 CH8 CH0 2 CH02 CH10 0 10 0 CH1 CH9 3 CH03 0 11 CH11 0 CH2 CH10 4 CH04 0 CH12 0 CH3 CH4 12 CH11 CH12 5 CH05 0 13 CH13 0 連続 CH5 CH13 6 CH06 CH14 0 14 0 CH14 CH6 停止 7 CH07 0 15 CH15 0 CH7 データファイルからの波形描画: データファイル: D:¥utilities¥data¥ad_read.bin ファイル選択 グラフ描画

図 13 の波形ビューワ画面で、「連続」ボタンをクリックすると波形を表示します。

<図 13. 波形ビューワ画面>

アナログ波形の表示

RefMX.exe のメモリ操作タブで読み出したバイナリデータを図 13の画面で波形表示することもできます。

S RefMX Ver1.0.0.0 for Win64	
7ァイル(Y) Option ヘルプ(Z)	
レシ [*] スタ操作 メモリ操作 USBコンフィク [*] FlashPath ホ [*] ート [*] 情報 RAA730301	
Memory Setting	
Start Address(Register0) 00000000 00000000 (HEX) Read	Write
Target Memory(Register1) 0 Transfer Length 002000	(HEX)
File Setting	
D:¥utilities¥data¥ad_read.bin	ファイル選択
Memory Width 8bit MemRead MemWrite Split Transfer	開
ロケ ⁱ View再表示	終了
Transfer Langth → 8.00KiBytes[8192Bytes]	

<図 14. AD 変換後のデータをファイル化する操作画面>

Transfer Length 欄で読み出したいデータ量(バイト)を設定し、File Setting 欄にファイル名を指定して「MenRead」ボタンをクリックすれば、AD 変換後のバイナリデータをファイル化して保存できます。 図 13 の画面の最下部にあるファイル選択で、このファイルを選択し、「グラフ描画」すると波形が表示されます。

ADC コアのカスタマイズ

この章では、サンプル FPGA 回路をベースにした AD コア(IP)のカスタマイズによるチャネル数の 増減について解説します。

サンプル FPGA 回路では CH 0 と CH8 の 2 入力を有効にし、交互に信号をサンプリングする構成で す。このため、CH8 のサンプリングレートは 1MHz の半分 500KHz サンプリングになります。使用 するチャネル数が増えればサンプリング速度は低下します。

※ Quartus Prime ver.16 以降、AD コアのサンプリング速度を選択できるようになっています。 図 15 の画面は Qsys ツールを起動し、AD コアの設定画面を表示した例です。

IP Parameter Editor - ADC_top.q	sys (D:¥project¥qdesigns¥mxcard¥mxcard_GPIF_Verilog_160616¥ADC	_top.qsys)	
File Edit System Generate View	Tools Help		
Narameters 🛛		- 🗗 🗆	🔋 Details 🛛 Block Symbol 🕅 🗕 🗗 🗖
System: ADC_top Path: modular_adc	0		
Altera Modular ADC core altera_modular_adc		Details	Altera Modular ADC core
Debug Path:	Disabl	^	ADC Sample Rate
T Clocks			Effective ADC Sampling Rate
ADC Sample Rate: ADC Input Clock: Reference Voltage Reference Voltage Source: External Reference Voltage: Logic Simulation Enable user created expected output Channels Sequencer	1 Mhz ▼ 1 Mhz ▼ 500 Khz 250 Khz 200 Khz 200 Khz 100 Khz ▼ 50 Khz ▼ 50 Khz ▼ 100 Khz ▼ 100 Khz ▼ 100 Khz ▼ 100 Khz ▼	E	
CH0 CH1 CH2 CH3 CH4 CH8	5 CH6 CH7 CH8 CH9 CH10 CH11 CH12 CH13 CH14 CH15 CH16 T	SD	🍯 Presets 🛛 🗕 🗗 🗖
✓ Channel 0 ✓ Use Channel 0 (Dedicated and	ilog input pin - ANAIN)		Presets for modular_adc_0
X≣ Messages ⊗		- 🗗 🗆	
Type Path Message			
•	III	•	Apply Update Delete New
0 Errors, 0 Warnings			Generate HDL Finish

<図 15. Qsys AD コア設定画面>

アナログ入力チャネルを有効にするには「Channels」タブで各チャネルのチェックボックスにチェ ックを入れます。次に、有効チャネルをどのような順番(Slot)でサンプリングするか「Sequencer」 タブで設定します。

シングル AD コアの場合、17ch を有効にできますが、サンプル FPGA 回路では AD 出力データの 12bit にチャネルデータの 4bit を加えて {16bit/サンプリング} という単位でデータを扱っています。

<サンプリングデータフォーマット>

16bit データ	[15:12]	[11:0]
内容	AD コアが出力するチャネルデ	AD コアが出力する 12bit サンプリングデータ
	ータ[4:0]のうち[3:0]を適用	

この 16bit データを 1MHz 周期で FIFO(16K ワード、32KB)に入力し、48MHz クロックで読み出 します。

このため、有効にできるチャネル数は 16ch です。17ch 有効にする場合には、回路の変更が必要で、 RefMX の波形ビューワ機能で 17ch 表示ができません。

サンプル FPGA プロジェクトについて

サンプル FPGA 回路では、0ch(アナログ入力専用ピン)と8ch(プリスケーラ機能有効)の2ch を 交互にサンプリングしています。ADC のサンプリング周波数を1MHz に設定しているので、各 ch は 500KHz/s のサンプリングレートです。

また、サンプル回路では MAX10-FPGA の 8ch に SIN 波(0~1.5V)を入力する設定になっています。 図 10 を参照して下さい。

RefMX.exe の波形ビューワを利用する際には、以下の設定が必要です。

- レジスタ No.4(16bit)にアナログデータを取り込むためのトリガ値を WR します。
 例:100h
- 2. RAA730301 に設定ファイルを書き込み(Gain=6dB 設定)

レジスタ No.4 でトリガ値が正しくないと、波形ビューワで「連続」ボタンをクリックしても動作しません。現状アプリは固まってしまいますので強制的に終了して下さい。

レベル信号等を観測する場合は、mxcard ボード ディップスイッチ SW3 ビット[1]を ON にすると、 トリガ機能を無効にできます。

*RAA730301 設定ファイルでは、ゲイン 2 倍(6dB)設定ですが、実際には 1.25 倍程度になりま す。('16.6.21 現在)

お問い合わせ

有限会社プライムシステムズ

e-mail : mailto:info@prime-sys.co.jp

http://www.prime-sys.co.jp

http://www.smartusb.info (技術サポートサイト)

改版履歴

′16.6.21:初版リリース