

4.1

#### DMAを使用したサイン波生成

# DAC Signal Generation With DMA Control PSoC 3/5 Experiment Lab



DAC Sine Wave Generation by Constant Table Experiment Course Material 4.1 V1.50 June 25th. 2012 DMA\_DAC\_35.PPT (39 Slides)

Renji Mikami Renji\_Mikami@nifty.com



# DMA\_DAC\_35 DACサイン波信号生成(DMA使用)

#### DMAC使用

#### 演習ファイルと使用文字などの注意事項

各演習プロジェクトを置くための演習用ディレクトリを作成してく ださい。場所と名前は、演習中に指示します。(デフォルトは、 C:¥PSoC5\_Lab)

設計例(正解の例)プロジェクトは、演習で指示する場所の PSoC\_Lab\_MasterXXXにあります。

使用するファイルのあるディレクトリのパス名に英数字以外の 文字が含まれていないことを確認してください。(不可例参照)

ファイル名、プログラムのソースには、英数字と構文で許された記号以外は使用しないでください。(不可例参照)

WorkspaceとProjectを1対1に対応して作成(次スライド)

#### 不可例:全角文字、日本語文字、半角カナ

### 演習プロジェクト作成のグランドルール

設計は、プロジェクトをひとつの単位としますが、大規模設計では複数のプロジェクトをまとめて、ひとつのWorkspace内で管理することができます。

複数のプロジェクトを一つのWorkspaceに追加した場合、同じ名前のファイルを開いたときに、どのプロジェクトに含まれるものなのかを毎回確認する必要があります。例えば main.c などです。

ワークショップ中は、この混乱をさけるために、一つの Workspace には一つのプ ロジェクトしか配置しないように注意して下さい。

この方法は三つあります。

1.新プロジェクトを作成する前に、一度現在のPSoC Creatorを閉じる

2.プロジェクト作成する前に、FileメニューからClose Workspaceを実行して現在 開いているWorkspaceを閉じる

3.新規プロジェクトを作成する時、Advancedオプション内のWorkspaceをCreate New Workspaceに設定する。この方法については次のページに説明があります。

#### プロジェクト新規作成時の注意事項

プロジェクトを既に開いた状態で新規プロジェクトを作成した場合、デフォルトでは、 現在開いているWorkspaceに追加されます。ウィザード内のWorkspaceをCurrent New Workspaceに切り替えて下さい。

New Project	Add to	Current Workspaceを選択
Design Other PSoC Creator Installed Templates PBEmpty PSoC 3 Design PBEmpty PSoC 5 Design Advancedオプションを オープンする	↓ すると 現在開 新規プ <b>Create</b> と、	いているWorkspace内に コジェクトが作成されます。 <b>New Workspace</b> を選択する
Creates a PSoC 3, 8 bit, design project. Name: fue: Location D:¥MyCypress¥PSoC_Creator_Projects¥hoge Add to Current Workspace Workspace Name: Oreate New Workspace	新規Wa その中 ます。	orkspaceが作成され、 に新規プロジェクトが作成され
Device: CY8C3866AXI-040 - (Last Øsed PSoC Sheet Template: A4 (11.7" x 8.3") Application Type Normal	Advanced Workspace:	Add to Current Workspace
ОК	Workspace Name:	Add to Current Workspace Create New Workspace

### プロジェクトファイルの形式

PSoC Creator では、Workspaceの中に複数の Project Filesが含まれる、というファイル構成に なっています。

これは、関連するプロジェクトを一つにまとめて取り扱うことが出来るようにするものです。例えば以下のような場合に便利です。

送信側プロジェクトと受信側をひとまとめにして管 理したい時

コンポーネント毎の挙動を確認するためにテストプ ロジェクトを作成する時



### 新規プロジェクトを作成した場合

Creatorを起動した直後に"hoge"という新規プロジェクトを作成した場合、自動的に"hoge"というWorkspaceが生成され、その中に"hoge"というプロジェクトが生成されます(左図参照)。



既にプロジェクトファイルが開かれた状態で、新規プロジェク ト"piyo"を作成した場合、デフォルトではWorkspace "hoge"の中 にプロジェクトファイル"piyo"が作成されます。





Workspace Explorer内のプロジェクト名を右クリックし、[Set As Active Project]を選択することで、Active Projectが切り替わります。

Activeなプロジェクトの切り替え方



<u>File Edit View D</u> ebug	<u>P</u> roject <u>B</u> uild	<u>T</u> ools <u>W</u> indow <u>H</u>
🗗 👌 👌 💕 月 🖉 🕘	3 1 % 1 8 8 >	(ାଏ ୯ ୍ରି 📑 - 📍
Microsoft Sans Serif	- 10 - B	/ U 🖉 🖉 🗐 🗚
Workspace Explorer	<b>- 4</b> ×	Start Page TopDe
1 😤		4
#Workspace 'hoge' (2 Project)	s)	
Project hoge [CY80	Set As Active Pr	oject
hoge cydwr 	Add	<b>\</b> ,
🦾 🖻 device h 🔛	Build hoge	14
E Cource Files	C <u>l</u> ean hoge	34
🕂 😼 Project 'piyo' [CY: 🎬	Clea <u>n</u> and Build	hoge
<u>Ba</u> TopDesign.cysch <mark></mark> piyo.cydwr	Update Compor	ients .
🖻 🧰 Header Files 👔	<u>С</u> ору	Ctrl+C
E Cource Files	Paste	Ctrl+V
i-C main.c	E <u>x</u> clude	12
	Rename	F2

hoge - PSoC Creator 1.0	) [D:¥¥PSc	C_Creator_
<u>Eile E</u> dit <u>V</u> iew <u>D</u> ebug	g <u>P</u> roject	<u>B</u> uild <u>T</u> ool
i 🗊 🔁 👸 🎽 🗐 🥵 🥵	Q % D	13 × 17
Microsoft Sans Serif	+ 10 +	BIU
Workspace Explorer		9 XSt
		3
<ul> <li>Project noge [CY8C:</li> <li>TopDesign.cysch</li> <li>Header Files</li> <li>Adevice h</li> <li>Source Files</li> <li>Project 'piyo' [CY8C386</li> <li>Project 'piyo' [CY8C386</li> <li>TopDesign.cysch</li> <li>Project Files</li> <li>Header Files</li> <li>Header Files</li> <li>Source Files</li> <li>Source Files</li> <li>Source Files</li> <li>Source Files</li> <li>Meader Files</li> <li>Source Files</li> <li>Meader Files</li> <li>Source Files</li> <li>Meader Files</li> </ul>	6AXI-040]	Source Components Results

#### DMA\_DAC\_35 ラボの目的

- DMAの使い方
- DACの使い方

前のプロジェクトをセーブし、 File>Close Workspace で 終了してから、次のプロジェクト に進みます。\

• 定数テーブルによるサイン波発生

注意事項:このラボはのPSoC5(32bit ARM)搭載の 050基板で動作します。PSoC3(8bit 8051)搭載の 030基板ではメモリ構成に合わせてコードを変更します。

プロジェクトをロードして再開する場合は、 File>Open>Project/Workspaceを実行 続いて、プロジェクと/ワークスペースを選択

作業を終了する場合には、 File>Close Workspace を実行して現在開いている Workspaceを閉じること All Project Files (\*,cypri;\*,cywrk)

キャンセル



ファイルの種類(T)

### 8CKIT-030基板での実測 (PSoC3)

DMA\_DAC (実測P0[0])

PSoC3



uint8 wave[100] = {

128, 136, 144, 152, 160, 167, 175, 182, 189, 196, 203, 209, 215, 221, 226, 231, 235, 239, 243, 246, 249, 251, 253, 254, 255, 255, 255, 254, 253, 251, 249, 246, 243, 239, 235, 231, 226, 221, 215, 209, 203, 196, 189, 182, 175, 167, 160, 152, 144, 136, 128, 120, 112, 104, 96, 89, 81, 74, 67, 60, 53, 47, 41, 35, 30, 25, 21, 17, 13, 10, 7, 5, 3, 2, 1, 1, 1, 2, 3, 5, 7, 10, 13, 17, 21, 25, 30, 35, 41, 47, 53, 60, 67, 74, 81, 89, 96, 104, 112, 120

};

/\* Variable declarations for DMA\_1 \*/

/\* Move these variable declarations to the top of the function \*/

uint8 DMA\_1\_Chan;

uint8 DMA\_1\_TD[1];

/\* DMA Configuration for DMA\_1 \*/ #define DMA\_1\_BYTES\_PER\_BURST 1 #define DMA\_1\_REQUEST\_PER\_BURST 1 #define DMA\_1\_SRC\_BASE (CYDEV\_SRAM\_BASE) #define DMA\_1\_DST\_BASE (CYDEV\_PERIPH\_BASE) DMA\_1\_Chan = DMA\_1\_DmaInitialize(DMA\_1\_BYTES\_PER\_BURST, DMA\_1\_REQUEST\_PER\_BURST, HI16(DMA\_1\_SRC\_BASE), HI16(DMA\_1\_DST\_BASE)); DMA\_1\_TD[0] = CyDmaTdAllocate(); CyDmaTdSetConfiguration(DMA\_1\_TD[0], sizeof(wave), DMA\_1\_TD[0], TD\_INC\_SRC\_ADR); CyDmaTdSetAddress(DMA\_1\_TD[0], LO16((uint32)wave), LO16((uint32)VDAC8\_1\_Data\_PTR)); CyDmaChSetInitialTd(DMA\_1\_Chan, DMA\_1\_TD[0]); CyDmaChEnable(DMA\_1\_Chan, 1);

© Renji Mikami – 2012 PSoC 5 / 3 CUA Workshop

DMA制御参考コード

#### DMA初期化ファンクションの意味

#### txDmaChan = TxDMA\_DmaInitialize

・バーストコンテンツ: DMAの転送を1byteずつに設定。 最大で127要素 まで転送可能

・バーストが終わった後のリクエストをハードウェアから出すかどうかの設定

・ソースアドレスとデスティネーションアドレスの設定。ここでは上位16bitのみが設定される。両方とも SRAM上なら0x00

txTd[0] = CyDmaTdAllocate();

・Tdを確保。この値がInvalidだと確保に失敗しているので待機する必要がある。

CyDmaTdSetConfiguration(txTd[0], sizeof(buffer), txTd[0], TD\_INC\_SRC\_ADR);

Td の設定。

·対象となるTd

·Tdで転送するbyte数

·別にTdを選択する場合には、ここにほかのTdをセットする。

・アドレスのインクリメントおよび転送終了の有り、無しを選択

CyDmaTdSetAddress(txTd[0], LO16((uint32)buffer), (uint16)I2S\_Tx\_dpTx\_u0\_\_F0\_REG); ・Tdを選択

・データの転送元アドレスを設定

・データの転送先アドレスを設定

CyDmaChSetInitialTd(txDmaChan, txTd[0]); 確保したTDに対して、最終的に関連づけを行う。

CyDmaChEnable(txDmaChan, 1); 最後にチャンネルをイネーブルにする。 ・1だと転送完了後に自動的に設定が初期化される。0だとそのまま。

#### Step1.PSoC Creator Softwareの起動

PSoC Creator 1 0



### 各ウィンドウの説明





ウィンドウ番号	機能
1,Main Window	回路の表記、ピンアサイン、コードの記述などを行います。Work space exploreから開いたものが、メイン ウィンドウに表示されます。立ち上がった状態(画面の状態)ではSchimatic Windowが表示され回路の表 記Windowが表示されています。機能ごとに TAB形式で開かれ、TABが二つ以上ある場合にはTABを右 クリックしてNew XXX windowを選択することで縦、横に分割することもできます。
2,Work Space Explore	SourceTAB にはプロジェクトに必要なファイルが表示されます。また、ここから既存のファイルを追加する なども可能です。Componentsには自分で作成したモジュールなどのファイルが表示されます。Resultには 各種ログファイルが表示されます。
3,Component Catalog	標準ではCypress社の用意したモジュールが表示されます。コンポーネントのデータシートなども、ここから 参照できます。
4,Output	現在進行中のLogファイルがここに表示されます。 コンパイル結果なども同様に、 Output Windowに表示さ れます。

### Step2.新しいプロジェクトの作成

File->New->Projectを選択してください Project名は、DMA\_DAC\_35等にしてください Locationは、演習で指示しますが、デフォルトは、 C:¥PSoC5\_Labとします。

H H	ELLO_WORLD_35 - PSoC Creator	2.0 [C:¥PSoC5_Lab¥HELLO_WC
<u>F</u> ile	e <u>E</u> dit <u>V</u> iew <u>D</u> ebu <mark>g P</mark> roject	<u>B</u> uild <u>T</u> ools <u>W</u> indow <u>H</u> elp
	New +	Project
	Open •	1 <u>File</u>
	Add	Start Page TopDesign.cysch mai
i <sup>t,</sup>	Close Ctrl+F4 Close Workspace	10 * 11 */ 12 #include <device.h 13</device.h 

# Step2.1 新しいプロジェクトの設定(続き)







© Renji Mikami – 2012 PSoC 5 / 3 CUA Workshop



#### Step3.2 Pinコンポーネントを追加



me: VDAC8_1	
Configure Built-in	
Range       Speed         Image       Image         Im	VDAC8のシンボルをダブルクリック コンフィギュレーション <mark>設定</mark>
with with with with with with with with	Configure 'VDAC8'
8 bit Hex: 64 Note: Changing any value field recalculates the other Begister Write	Name: VDAC8_1  Configure Built-in Parameter Value  CY_MAJOR_VERSION 1
	CY_MINOR_VERSION 70
	CY_REMOVE false
	CY_SUPPRESS_API_GEN false
	CY_VERSION PSoC Creator 2.0
	Parameter Information CY_MAJOR_VERSION: Major Version Value: 1 Type: string

. . .

### Step3.4 DMACのコンフィグレーション

Configure 'cy_dma' Name: DMA_1 Basic Built-in Hardware Request:	Derived	2 DMACの コンフィ	▲ Dシンボルをダブルクリックして ィギュレーションを設定、確認
Hardware Termination:	Disabled Derived Rising Edge Level	Configure 'cy_dma'	
		Basi Built-in Parameter	4 ⊅ Value
Datasheet	OK	CY_MINOR_VERSION CY_REMOVE CY SUPPRESS API GEN	50 false false
		CY_VERSION	PSoC Creator 2.0
DMA 1 DMA nrq		Parameter Information CY_MAJOR_VERSION: Value: 1 Type: string Datasheet	Major Version

### Step3.5 Clockのコンフィグレーション

Configure 'cy_ Name:	clock'	2 ×	Clock_1
Clock Type: Source: Specify:	e Clock    Advanced    Built-in   New   Existing <auto>      Frequency   100   kHz +   Tolerance:   -   5%</auto>	Clockのシ コンフィギ	ンボルをダブルクリックして シュレーションを <mark>設定、確認</mark>
API Ge Uses ( By default, Design Wide	Name: Clock_1 Configure Clock Advanced Built-in Force clock to be Analog Clock. (This option provides ar Sync with MASTER_CLK	Configure 'cy_clock' Name: Clock_1 Configure Clock Adv Parameter	vanced Built-in 4 b
Datashe	The clock distribution network produces a master clock, MA resynchronization. This clock is not intended for clocking cir distribution network. Output clocks can be phase aligned to MASTER_CLK should be the highest frequency clock in the Generally, all clocks used in the chip must be derived from synchronized to the main fast clk_sync clock (MASTER_CLK	CY_MAJOR_VERSION CY_MINOR_VERSION CY_REMOVE CY_SUPPRESS_API_GEN CY_VERSION	1 60 false false PSoC Creator 2.0
	By setting this parameter to false this clock becomes an un Datasheet	Parameter Information <b>GY_MAJOR_VERSION:</b> <b>Value:</b> 1 <b>Type:</b> string Datasheet	Major Version

### Step4 コンポーネントの配線



### Step5 デバイスのピンアサイン

左側のWorkspace Explorerから、DMA\_DAC\_35.cydwr をクリック、 の回面下左端のPinsを選んでピンアサイン画面が開く。



### Step6.main.cにコードを記述(その1)



正弦波の100ポイントのデータの記述 (ペーストソースあり)









### エラー発生時には、修正して再度Build

#### Build -> Build XXX(XXXはプロジェクト名)を選択して Buildを再実行

DMA_DAC_35 - PSoC Creator 2.0	[C:¥PSoC5_Lab¥DMA_DAC_35¥DMA_DA	C_35.cydsn¥ma
<u>Eile E</u> dit <u>V</u> iew <u>D</u> ebug <u>P</u> roject	<u>B</u> uild Tools <u>W</u> indow <u>H</u> elp	
10000000000000000000000000000000000000	Build DMA_DAC_35 Shift+F6	•
通 • 孟 参 😭 👹 兼 🖕 律 律 Workspace Explorer (1 proj • 年 🗙	Clean DMA_DAC_35	*main.c Start
	Cancel Build     Ctrl+Break       Image: Second Seco	ANY, THE YI
TopDesign.cysch  DMA_DAC_35.cydwr  Header Files  Component Source Files  Comp	Generate Application 6 7 * CONFIDENTIAL AND F 8 * WHICH IS THE PROPE 9 * 10 * * * * * * * * *	ED SOFTWAR

# Step7.デバイスにプログラムを書き込み Debug->Programを選択

HELLO_WORLD_3	5 - PSoC Creator 2.0 [C:¥PSoC5_Lab¥HELLO_W
<u>File E</u> dit <u>V</u> iew	Debug Project Build Tools Window Help
1 <b>67 17 67 16</b> 16	Windows
- 🖂 🆃 💕 🤻	Program Ctrl+F5
Workspace Explorer	😹 Select Debug <u>T</u> arget
	養 <u>D</u> ebug F5
Workspace 'HELLO_'	
TopDesign.c	Attach to Running Target
HELLO_WOR	Toggle Breakpoint F9
hedder med	New Breakpoint
ja - Cource Files	Delete All Breakpoints Ctrl+Shift+F9
Generated_S	Enable All Breakpoints
🗄 🛅 PSoC5 🗌	0 21 /* Place

## プログラムの進行

#### プログラムの進行状況はOutput ウインドウに表示 正常に終了した場合はSucessfully programmed…と表示されます



各演習プロジェクトが終了時には、必ず現在のワークスペースを閉じてから (File>Close Workspace)次のプロジェクトの作成や読み込みを行ってください。

#### プロジェクト/ワークスペースのクローズについて

1.File > Close Workspaceを実行



プロジェクトをロードして再開する場合は、 File>Open>Project/Workspaceを実行 プロジェクト/ワークスペースを選択



## オシロスコープ(WS)で波形の観測

#### DMA\_DAC (実測P0[0])





オシロスコープやWave Spectra等がない場合には、周波 数を1から10Hzに落とし、電圧のスイングを0から5Vに上げ て、LEDの点灯で確認します。2箇所のパラメータの変更で、 周波数とスイング電圧を変更します。

- 1.スライド22 Step3.3 のRenge の変更
- 2.スライド24 Step3.5 のClock Frequency の変更 各自で変更し、LEDでサイン波の明暗変化を確認して ください

セーブ後は、File>Close Workspace で終了します。

#### Lab DMA\_DAC\_35

終了

この資料は、デバイスがES1, ソフトウェアPSoC Creater 1.0SP2 /2.0 をベースに作成しています。 エラッタやバージョンの違いで操作や動作が 異なる場合があります。

#### Memo

#### フォローアップURL

http://mikamir.web.fc2.com/?/?.htm

#### ?に入る文字列は、講義中に示します。

本資料は、米国および日本サイプレス社の協力と情報の提供 により作成されおり、著作権は以下に帰属します。 内容は定期的に改訂されます。引用や再使用の場合はご連絡ください。

#### 担当講師

ミカミ設計コンサルティング

〒142-0042 東京都品川区豊町 2-17-8

#### 三上廉司(みかみれんじ)

Renji\_Mikami@nifty.com

http://homepage3.nifty.com/western/mikamiconsult.htm 電話 080-5422-2503(au)