# プロジェクト作成

• Quartus IIを起動



タイトルページが現れる。

[Create New Project] を選択

	and the second
Getting Started With Q	Quartus® II Software
Start Designing Designing with Quarter & software requires a project	Start Learning
Create a New Project Over Project Wittend) Open Existing Project Open Recent Project:	Open Interactive Tutional ここをクリックして新しいプロジェクトを作る
C: altera NyProject PwrsCstreycto C: altera NyProject NiesTest I/cycl C: altera NyProject FirSmp/cycleg	n Linning Oning Support
Don't show this screen again	AGENA.



## プロジェクト名の決定 ・プロジェクトという単位で開発を行います。 >ディレクトリ指定 >プロジェクト名の入力

トップレベルデザインを決定します (自動で決定される。)

通常プロジェクト名と同じ名前で モジュール名をつくり、そのモ ジュールがトップレベルデザインと なります。

What is the working directory	for this project?	
C:\altera\13.1		
What is the name of this proj	ect?	
PwmCtrol		
What is the name of the top-	evel design entity for this project? This name is case sensitive and must exactly r	match the entity name in the design file.
PwmCtrol		
Use Existing Project Setting	· · · ·	
<b>.</b>		• L L

< Back

Next >

Finish

Cancel

Help

Device	family				Show in 'Available devices' list				
Family:	Family: Cyclone III		•	Package:	Package: Any				
Devi	ces: Ali			*	Pin count:	484	4		•
	7779-107								
Target	device				Speed grad	de: 6		_	•
O Aut	to device selected by the I	Fitter			Name filter	;			
	cific douico coloctod io 'Au	usilabla dovici	oo' liet		Show a	advance	d devices		
-									
		$\sim$	∼ ∧ । ≁  ∖ē	32 エロ・	_ 7				
$\cdot$ $  -$	2C16E19	Q / C	~ ハ ! / ~ ¬ ፣	ᄄᅚ	すん				
EP	3C16F48	84C6	oNをz	些扒	する				
EP	3C16F48			些状	95 95	cimbed	lded multiplier 9-bit elements	; PLL	bal Clo
	3C16F48		5Nを1 555511/05 347	些 <b>伬</b>	9 5 11	cmbed	lded multiplier 9-bit elements	F PLL	bal Clo 20
EP3C160	3C16F48 -484C6 1.2V J484C6 1.2V	15408 15408	ON とし oser 1705 347 347	些 <b>伬</b> 516096 516096	9 5 7 DILS	cmbed	Ided multiplier 9-bit elements	6 PLL 4 4	<b>bal Clo</b> 20 20
EP3C160 EP3C160 EP3C160 EP3C400	<b>3C16F48</b> <b>484C6</b> 1.2V <b>484C6</b> 1.2V <b>484C6</b> 1.2V <b>484C6</b> 1.2V	84C6 15408 15408 39600 39600	347 347 332	516096 516096 1161216	9 5 9 11 11 25 25	2 12 12	lded multiplier 9-bit elements	5 PLL 4 4 4	bal Clo 20 20 20 20
EP3C160 EP3C160 EP3C400 EP3C400 EP3C400	<b>3C16F48</b> <b>484C6</b> 1.2V <b>484C6</b> 1.2V <b>484C6</b> 1.2V <b>484C6</b> 1.2V <b>484C6</b> 1.2V <b>5484C6</b> 1.2V <b>5484C6</b> 1.2V	15408 15408 39600 39600 55856	347 347 332 332 332	<u>516096</u> 516096 1161216 1161216 2396160	9 5 11 11 25 25 31	2 12 52 52	lded multiplier 9-bit elements	5 PLL 4 4 4 4 4	bal Clo 20 20 20 20 20 20 20
EP3C160 EP3C160 EP3C400 EP3C400 EP3C400 EP3C555 EP3C555	<b>3C16F48</b> <b>484C6</b> 1.2V <b>484C6</b> 1.2V <b>484C6</b> 1.2V <b>484C6</b> 1.2V <b>484C6</b> 1.2V <b>484C6</b> 1.2V <b>484C6</b> 1.2V <b>484C6</b> 1.2V	84C6 15408 15408 39600 39600 55856 55856	347 347 347 332 332 328 328	<u>516096</u> 516096 1161216 1161216 2396160 2396160	9 5 y bits 11 11 25 25 31 31	2 2 2 2 2 2 2 2 2 2 2	lded multiplier 9-bit elements	5 PLL 4 4 4 4 4 4 4 4 4	bal Clo 20 20 20 20 20 20 20 20 20
EP3C160 EP3C160 EP3C400 EP3C400 EP3C550 EP3C550 EP3C550	<b>3C16F48</b> <b>484C6</b> 1.2V <b>484C6</b> 1.2V <b>484C6</b> 1.2V <b>484C6</b> 1.2V <b>484C6</b> 1.2V <b>484C6</b> 1.2V <b>484C6</b> 1.2V <b>484C6</b> 1.2V <b>484C6</b> 1.2V	84C6 15408 15408 39600 39600 55856 55856 81264	347 347 347 332 332 328 328 328 296	<b>生</b> 516096 516096 1161216 1161216 2396160 2396160 2810880	9 5 y bits 11 11 25 25 31 31 31 48	2 12 12 12 12 12 12 12 12 12 13 8	lded multiplier 9-bit elements	5 PLL 4 4 4 4 4 4 4 4 4 4 4 4	bal Clo 20 20 20 20 20 20 20 20 20 20 20 20
EP3C160 EP3C160 EP3C400 EP3C400 EP3C400 EP3C550 EP3C800 EP3C800	<b>3C16F48</b> <b>484C6</b> 1.2V <b>484C6</b> 1.2V <b>484C6</b> 1.2V <b>484C6</b> 1.2V <b>484C6</b> 1.2V <b>484C6</b> 1.2V <b>484C6</b> 1.2V <b>484C6</b> 1.2V <b>484C6</b> 1.2V <b>484C6</b> 1.2V	84C6 15408 15408 39600 39600 55856 55856 81264 81264	347 347 347 332 332 328 328 328 296 296	<b>生</b> 516096 516096 1161216 1161216 2396160 2396160 2810880 2810880	9 5 y uics 11 11 25 25 31 31 48 48	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	lded multiplier 9-bit elements	<ul> <li>PLL</li> <li>4</li> </ul>	bal Clo 20 20 20 20 20 20 20 20 20 20 20 20

ソースを書く

ディジタル回路を記述するには、回 路図よりもハードウェア記述言語 HDLの一種であるVerilog HDLを使う

- メニューからFile→New を選ぶ
- ・ファイル名PwmCtrl.vとしてセーブ





#### 一旦、コンパイル「ソースコードのコンパイル」

ソースコードをコンパイルします。

Quartus IIのメニューバーに「Start Compilation」のボタンがあるのでクリック Successとでれば成功です。

🐇 Quartus II 32-bit - C:/DED Start Compilationボタンをクリック								
Eile Edit View Project Assign	Search altera.com							
🗋 🖆 🛃 🎒 🎒 🐰 🗎 🛍 🔒 🕫 🖼   PwmCtrl	v 💢 🐓 🖌 🏈 🧇 💷 📐 🕫 🐚 👯 🐔 🕘 😓 🖉 🐬							
Project Navigator Ø 🖉 🗙	PwmCtrl.v Start Compilation							
Entity	圖  桷 鎬 (7)   諄 諄  /4 🌤 🌤 🌾   🌒 🖾   🔂   錣 🖦 /   🔜 🖻 🖺							
🛆 Cyclone III: EP3C16F484C6 ▶ PwmCtrl கீத	<pre>1</pre>							

# ピンを割当てる

### •メニューのAssignments→Pin Planner を選択



#### PIN PLANNER のマニュアル

https://www.altima.jp/members/japanese/download/pinassign\_v10\_1.pdf

## Pin Planner

Pin Plannerが起動します。 さきほどのソースコードで 作成した CLK,LEDO,RST\_N が下の表に表示されています。

これを、FPGAのピンに割り当て を行います。 実験で使用するFPGAは、484個 のピンがあります。

そのうちいくつかのPINの用途は 決まっていますが、他は自由に つなぐことができます。

縦は A~AB まで22行あります 横は 1~22 まで22列あります。

ピンは行列であらわします。 例 右上「A22」



# DE0 の回路図

DEOボード上には、FPGAの他に LED、水晶発信機、プッシュボタン など、多くの電子部品が配置され ています。

それらは、FPGAのピンに事前に 接続されています。



FPGAの「どのピン」が「どの外部回路」に接続しているかを知らなければなりません。

DE0\_User\_Manual.pdf



RST\_N には 右下のプッシュボタン CLK には水晶発信の信号 LED0 には右端のLED を割り当てることにしましょう。

	Table 4	<ol><li>Pin assign</li></ol>	iments for the pushbutton switches
DEO の回路図より	Signal Name	FPGA Pin No.	Description
	BUTTON [0]	PIN_H2	Pushbutton[0]
DE0_User_Manual.pdf	BUTTON [1]	PIN_G3	Pushbutton[1]
	BUTTON [2]	PIN_F1	Pushbutton[2]
1	\$	Table 4.3. Pin	n assignments for the LEDs
	Signal Name	FPGA Pin No.	Description
	LEDG[0]	PIN_J1	LED Green[0]
	LEDG[1]	PIN_J2	LED Green[1]
LEDOIT J1	LEDG[2]	PIN_J3	LED Green[2]
水晶発信の信号は G21	LEDG[3]	PIN_H1	LED Green[3]
	LEDG[4]	PIN_F2	LED Green[4]
	LEDG[5]	PIN_E1	LED Green[5]
	LEDG[6]	PIN_C1	LED Green[6]
EP3C16F484C6N	LEDG[7]	PIN_C2	LED Green[7]
H2 Button0	LEDG[8]	PIN_B2	LED Green[8]
CLK	LEDG[9]	PIN_B1	LED Green[9]
G21 <b>≼</b> 50MHz			
LEDO	Tat	ole 4.5. Pin a	assignments for the clock inputs.
J1 LEDG0	Signal Name	FPGA Pin No.	Description
	CLOCK_50	PIN_G21	50 MHz clock input
_	CLOCK_50_2	PIN_B12	50 MHz clock input

#### it also T-11-40 D. . for the la la sett

Node Name	Direction	Location	Loodelollor	11272707	/// 0/	Standard
CLK	Input			PIN	_G2	2.5 V (default)
💷 LEDO	Output	/		PIN	E12 2	2.5 Y (default)
RST_N	Input	×				~
< <new node="">&gt;</new>		PIN_G16	IOBANK_7 Calumn I/O	DIFFIO_T31p		1
		PIN_G17	IOBANK_6 Row I/O	DIFFIO_R1p		
		PIN_G18	IOBANK_6 Row I/O	DIFFIO_R5n, PA	JDD23	
		PIN_G21	<ul> <li>IOBANK_6 Dedicated C</li> </ul>	lock CLK4, DIFFCLK_	2р	
		PIN G22	IOBANK 6 Dedicated C	lock CLK5, DIFFCLK	2n	
		PIN_H1	IOBANK_1_Row I/O	DIFFIO_L11n		
0		PIN H2	IOBANK_1 Row I/O	DIFFIO_L11p		
		PIN H5 1	IOBANK 1 Row I/O			
ī <		PIN H6	IOBANK 1 Row I/O	DIFFIO L5p		
		PIN HZ	IOBANK 1 Row I/O	VREFB1N0		~

#### Locationをクリック してピンを割当てる



6	Node Name	Direction	Location	I/O Bank	VREF Group	Fitter Location	I/O Standard	
	🖻 CLK	Input						٧
	🗇 LEDO	Output	PIN_G15	IOBANK_7 Column (	I/O DIFFIO_T3	On		~
	🗈 RST N	Input	PIN_G16	[OBANK_7 Column]	I/O DIFFIO_T3	1p		
	< <new node="">&gt;</new>		PIN_G17 PIN_G18	IOBANK_6 Row I/O IOBANK 6 Row I/O	DIFFIO_R1 DIFFIO_R5	ip in, PADD23		
			PIN_GZ1	IOBANK_6 Dedicate	d Clock, CLK4, DIFF	·CLK_Zp		
			PIN_G22 📉	[OBANK_6 Dedicate	d Clock CLK5, DIFF	CLK_2n		
			PIN_H1	[OBANK_1 Row I/O	DIFFIO_L1	1n		- 1
			PIN_H2	[OBANK_1 Row I/O	DIFFIO_L1	1p		- 1
			PIN_H5	[OBANK_1 Row I/O				
5			PIN_H6	IOBANK_1 Row I/O	DIFFIO_L5	p		$\sim$
ব	<		III					
							0%	%

e g	Node Name	Direction	Location		I/O Bank	VREF Group	Fitter Location	I/O :
	🕩 ακ	Input	PIN_G21	6		B6_N1	PIN_G2	2.5 V (d
	🐵 ledo	Output	PIN_J1	*			PIN_E12	2.5 V (d
	RST_N	Input	FIN_H2	1		B1_N1	PIN_G1	2.5 V (d
	< <new node="">&gt;</new>		15					

もう一度コンパイル

PwmC	Str I/PwmC	trl – PwmCtrl			- (	Start Compilation TAX + AllwA
; <u>I</u> oo	ls <u>W</u> indow	Help 💎				Start Compliation
mCtrl		<b>~</b> 🔀	¥ 🖌 🏈 🤻	🦻 🧇 🛛	1	10 🛈 😫 🗶 🗶 🕐
ð×	•	PwmCtrl.v	×	٨	Con Start (	
	i 📑 🕅	🏠 🔂 🕴 💷 🗐	6 % % %	🔏 🛛 🖉		267 268 ab/
	1 [	module PwmCtr	1 (			

コンパイルによってVerilog HDL で書かれた回路はゲート・レベル (AND, OR, フリップフロップといった基本ハードウェアウエア要 素)の回路に変換されます(これを論理合成と言う).

🐇 Quar	rtus II	×
<b>(</b>	Full Compilation was successful (11 warning	js)
	ОК	

<ul> <li>Device</li> <li>Timing Models</li> <li>Total logic elements</li> <li>Total combinational functions</li> <li>Dedicated logic registers</li> <li>Total registers</li> <li>Total pins</li> <li>Total virtual pins</li> </ul>	EP3C16F484C6 Final 28 / 15,408 ( < 1 % ) 28 / 15,408 ( <1 % ) 28 / 15,408 ( <1 % ) 28 3 / 347 ( <1 % ) 0	ロジカルエレメント(論理 要素)LE は28個しか 使っていない
<ul> <li>Total memory bits</li> <li>Embedded Multiplier 9-bit elements</li> <li>Total PLLs</li> </ul>	0/516,096(0%) 0/112(0%) 0/4(0%)	347ピンのうち3つ使用した

## DEO に電源を入れ、さらにパソコンと USB ケーブルでつなぎます.

- FPGAとPCの接続
- 実習用のPCには事前に USB-Blasterのドライバーが インストールされており、 自動的にPCとFPGAとの接 続が確立します。

液晶ディスプレイを接続 (カメラ画像の確認用) VGAを用いた実習で必要。 今回は、接続しなくても良い。



開発用PCとUSBケーブルで接続 (プログラムインストール用) FPGAを用いた組込みシステムの協

PGAを用いた組込みシステムの前 調設計 FPGAスタンドア ローンでの実行に 必要。今回は接続 しなくても可

- 1.「Programmer」起動 右上の アイコンを クリック
- 対象HWが登録を確認
   USB-Blasterが選択されている
   ことを確認。されていない場合
   「HardwareSetup」を押し、選択
- 3. SOFファイルの追加 4. SOFファイルの実行



### 実行結果 LED0が約5.37秒周期で点滅



## マイコンを用いた協調設計

- AND,OR,フリップフロップ、加算器、乗算器という基本ハードウェア要素をつなぎ合わせるだけでは、設計に限界がある。
- 柔軟でわかりやすいシ ステム設計には「マイコ ン」が便利

