

「デジタルシステム設計実験」

1. 2週目：FPGAボードによる電子野球ゲームの設計

実験1：FPGA 統合開発環境 IDE (QuartusII) の操作修得

別紙テキスト*に従って practice1～practice7 の回路を設計し、FPGA ボード上で動作させよ。

●プロジェクトの作成と回路入力，FPGA へ送出する回路データの作成（コンパイル）

1. パソコンの電源ボタンを押して起動し、デスクトップ画面が出るまで待つ。
(ユーザー名やパスワードの入力は不要)
2. 設計ファイルを保存するフォルダを各自の USB メモリまたはデスクトップ上に作成する。(プロジェクトごとにフォルダ名を変更すると混乱しなくてよい)
3. FPGA 開発統合環境 QuartusII を起動し、プロジェクトを新規作成する。このとき、Available Device として「EP3C16F484C6」を選択する。(次回からフォルダ内の QPF (QuartusII Project File) ファイル (*.qpf) の青色のアイコンをダブルクリックすればプロジェクトが再開される)
4. QuartusII のメニューから File→New と進み、トップモジュールを含む必要な設計ファイルを作成し、すべて保存する。
5. [プリフィット] QuartusII のメニューから Processing→Start Compilation と進み、コンパイルを実行する(スタートボタン(▶)をクリックしても良い)。しばらく待つと、「Full compilation was successful」と表示され、コンパイルが完了する。エラーメッセージが出ていないことを確認して OK をクリックする。エラーメッセージが出ているときは修正して再度コンパイルを行う。
6. [ピン配置] QuartusII のメニューから Assignments→PinPlanner と進み、設計した回路の信号線と FPGA のピンの配置を結びつける。(未使用ピンの処理も忘れずに行う)
7. [ポストフィット] ピン配置が終わったら、もう一度コンパイルして FPGA に書き込む回路データを作成する。

●コンパイル結果のFPGA へのダウンロードと動作確認

1. FPGA ボードに AC アダプタを接続し、AC100V コンセントに接続する。
2. パソコンと FPGA ボードを USB ケーブルで接続する。
3. FPGA ボードの赤ボタンを押して電源を入れる。
4. QuartusII のメニューから Tools→Programmer と進み、Device 欄に「EP3C16F484」と表示されることを確認する(表示されないときは Programmer 画面の Auto Detect をクリックする。Auto Detect も表示されないときは Hardware Setup ボタンをクリックして表示されたウィンドウで Currently selected hardware で USB Blaster[USB-0]をクリックして選択する。Close ボタンをクリックしてウィンドウを閉じる。
5. 認識された Device の File 欄が空欄または<none>のときは、その部分をダブルクリックし、コンパイル結果として得られた SOF ファイル(*.sof)を選択して開く。
6. Program/Configure 欄のチェックボックスにチェックを入れ、Start をクリックすると、SOF ファイルの内容が FPGA にダウンロードされ、完了後すぐに FPGA が動作を開始する。

実験を行う際の注意：

本実験で使用する FPGA ボード「DE0」はテキストで想定している FPGA ボードとは異なるため、次ページの表を参考にして、対応するピン番号に読み替えて設計すること。また、Pin Planner で使用するハードウェアは、BlasterMVの代わりに USB Blaster を使用することに注意せよ。

* 別紙テキストは、<http://www.ee.secu.chukyo-u.ac.jp/> にアクセスし、『電気電子工学実験3「デジタルシステム設計実験」』の中に保存されている。

実験 2 : 野球ゲームの設計と開発

●モジュールの設計

別紙テキストに従って、野球ゲームの設計仕様を確認した後、野球ゲームを構成する部品（モジュール）の回路を設計せよ。設計したファイルは `baseball` という名前のフォルダに保存すること。

`base.vhd`

`baseball_pkg.vhd`

`clk_div.vhd`

`baseball_led_dec.vhd`

`batlatch.vhd`

`outcount.vhd`

`baseball_led_out.vhd`

`batting.vhd`

`score.vhd`

●トップモジュールの設計

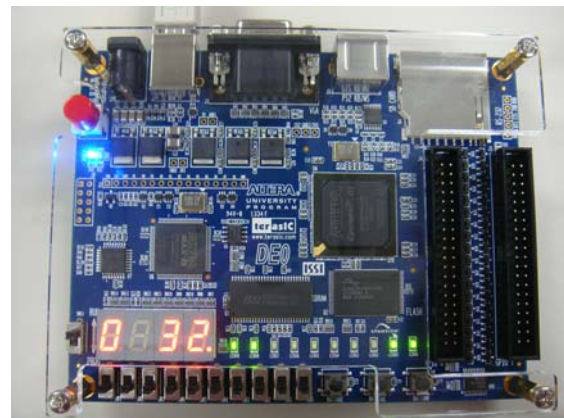
設計したモジュール間を接続するトップモジュールを設計し、`baseball` フォルダに保存せよ。

`baseball_top.vhd`

●野球ゲームのコンパイルと動作確認

1. FPGA 統合開発環境 QuartusII を起動し、`baseball` フォルダ内の設計ファイルをすべてプロジェクトに登録(Add Files)する。プロジェクトを `baseball_top` という名前で `baseball` フォルダ内に保存する。(次回から、`baseball_top`(QuartusII Project File)と書かれた青色のアイコンをダブルクリックすると、プロジェクトウィンドウが開き、設計を継続できる)
2. [プリフィット] [ピン配置] [ポストフィット] を行い、FPGA に書き込む回路データを作成する。(下記の「●ボード変更に伴う注意事項」にも注意すること)
3. コンパイル結果を FPGA へダウンロードし、動作を確認する。

ピンの名前	接続(テキストの記述)	ピン番号 (テキスト)	ピン番号 (DE0)	接続(DE0)
batter_led[0]	7セグメントLED(D3)の dp	25	PIN_G16	7セグメントLED(HEX3)の dp
batter_led[1]	7セグメントLED(D3)の g	22	PIN_G15	7セグメントLED(HEX3)の g
batter_led[2]	7セグメントLED(D3)の f	21	PIN_D19	7セグメントLED(HEX3)の f
batter_led[3]	7セグメントLED(D3)の e	24	PIN_C19	7セグメントLED(HEX3)の e
batter_led[4]	7セグメントLED(D3)の d	26	PIN_B19	7セグメントLED(HEX3)の d
batter_led[5]	7セグメントLED(D3)の c	23	PIN_A19	7セグメントLED(HEX3)の c
batter_led[6]	7セグメントLED(D3)の b	20	PIN_F15	7セグメントLED(HEX3)の b
batter_led[7]	7セグメントLED(D3)の a	5	PIN_B18	7セグメントLED(HEX3)の a
team0_led	7セグメントLED(D2)の dp	3	PIN_D13	7セグメントLED(HEX0)の dp
score0_led[1]	7セグメントLED(D2)の g	99	PIN_F13	7セグメントLED(HEX0)の g
score0_led[2]	7セグメントLED(D2)の f	98	PIN_F12	7セグメントLED(HEX0)の f
score0_led[3]	7セグメントLED(D2)の e	2	PIN_G12	7セグメントLED(HEX0)の e
score0_led[4]	7セグメントLED(D2)の d	4	PIN_H13	7セグメントLED(HEX0)の d
score0_led[5]	7セグメントLED(D2)の c	100	PIN_H12	7セグメントLED(HEX0)の c
score0_led[6]	7セグメントLED(D2)の b	97	PIN_F11	7セグメントLED(HEX0)の b
score0_led[7]	7セグメントLED(D2)の a	92	PIN_E11	7セグメントLED(HEX0)の a
team1_led	7セグメントLED(D1)の dp	90	PIN_B15	7セグメントLED(HEX1)の dp
score1_led[1]	7セグメントLED(D1)の g	84	PIN_A15	7セグメントLED(HEX1)の g
score1_led[2]	7セグメントLED(D1)の f	88	PIN_E14	7セグメントLED(HEX1)の f
score1_led[3]	7セグメントLED(D1)の e	89	PIN_B14	7セグメントLED(HEX1)の e
score1_led[4]	7セグメントLED(D1)の d	91	PIN_A14	7セグメントLED(HEX1)の d
score1_led[5]	7セグメントLED(D1)の c	87	PIN_C13	7セグメントLED(HEX1)の c
score1_led[6]	7セグメントLED(D1)の b	86	PIN_B13	7セグメントLED(HEX1)の b
score1_led[7]	7セグメントLED(D1)の a	85	PIN_A13	7セグメントLED(HEX1)の a
out_led1	LED(D8)	36	PIN_J1	LED(G0)
out_led2	LED(D9)	35	PIN_J2	LED(G1)
out_led3	LED(D7)	34	PIN_J3	LED(G2)
base1_led	LED(D6)	28	PIN_C2	LED(G7)
base2_led	LED(D5)	29	PIN_B2	LED(G8)
base3_led	LED(D4)	27	PIN_B1	LED(G9)
hitn	スイッチ(SW4)	37	PIN_F1	スイッチ(BUTTON2)
resetrn	スイッチ(SW3)	38	PIN_G3	スイッチ(BUTTON1)
clk	クロック発振器	10	PIN_G21	クロック発振器
resetrn	スイッチ(SW2)	39	PIN_H2	スイッチ(BUTTON0)
sw_in	スイッチ(SW1)	40	PIN_F1	スイッチ(BUTTON2で代用)



LEDHEX3 (緑) : ルーレットを表示

LEDHEX0 (緑) : チーム0の得点と攻撃中を表示

LEDHEX1 (緑) : チーム1の得点と攻撃中を表示

LEDG7, G8, G9 (緑) : 1, 2, 3塁のランナーを表示

LEDG0~G2 : アウト数を表示

BUTTON2 : ルーレットの回転開始と停止

BUTTON1 : ゲームリセット (やり直し)

●ボード変更に伴う注意事項

- ・ピン番号は左表に従って読み替えること。
- ・LEDG0~G9は論理が逆なので0と1を逆にする。

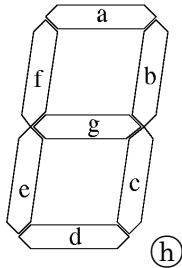
3週目開始前までにすべての設計ファイルを `baseball` フォルダ内に格納し、野球ゲームが動作することを確認してくること。もし動作がおかしい場合は設計ファイルの誤りを修正しておくこと。

3. 4週目：電子野球ゲームを改造してみよう！

下記は、簡単な改造例である。テキストにもあるように、外部回路を追加してサウンド（音）が出るようにしたり、内部回路の改造により打率を可変させたり、自由に改造を行い、レポートで報告せよ。

●改造その1：ルーレットのLED表示を変更してみよう

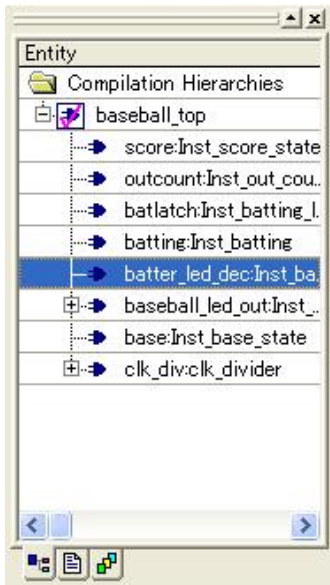
準備：ルーレットのLED表示で、点灯する場所を1、消灯する場所を0で記入せよ。



	a	b	c	d	e	f	g	h
1 (ヒット)								
2 (2 塁打)								
3 (3 塁打)								
H (ホームラン)								
O (アウト)								

改造のヒント：

FPGA 統合開発環境 QuartusII の左側に表示されている Entity ウィンドウの中にある baseball_top の中にあるファイル batter_led_dec をダブルクリックして開く。ファイルの中身を調べてみよ。



改造後のLED表示を記入せよ。(点灯=1, 消灯=0)

	a	b	c	d	e	f	g	h
1 (ヒット)								
2 (2 塁打)								
3 (3 塁打)								
H (ホームラン)								
O (アウト)								

例えば、漢数字表示（一，二，三...）や、アウトを0と間違えないように小文字表示（O→o）に変更するなど、試してみよ。

ファイルを改造したら必ず保存し、もう一度コンパイルを行ってから回路データを FPGA ヘダウンロードすること。

●改造その2：ルーレットのスピードを変更してみよう

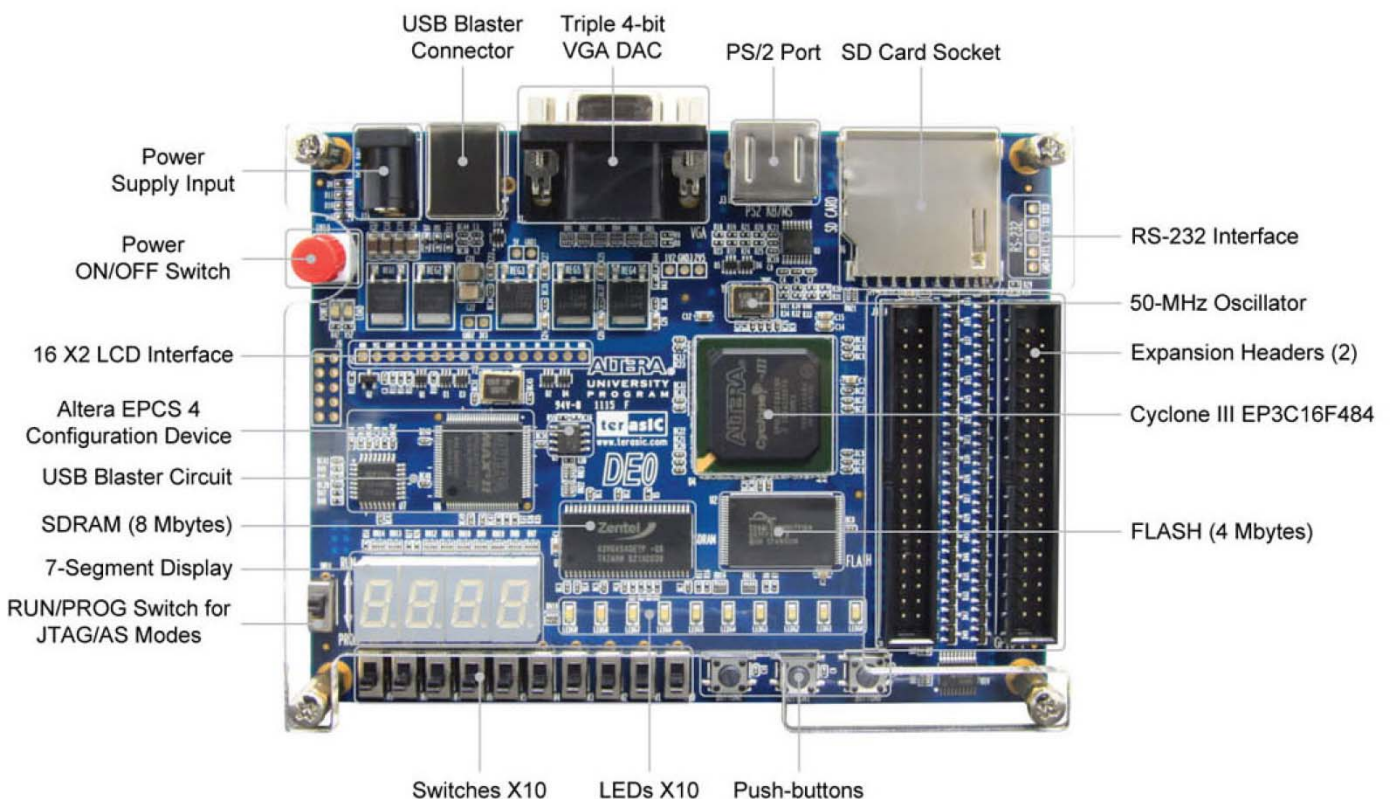
ルーレットは人間の目に見えないほど高速に動作している。

改造その1と同様に、baseball_top をダブルクリックし、145 行目付近の「DIV_BITS => 15」と書かれた数値を増減させ、好みのスピードに調整せよ。

●改造その3：電源を切っても回路を忘れずに記憶するようにしてみよう

FPGA という IC (LSI) は、一般に、回路を書き込んで動作する。しかし、電源を切るとその回路を忘れて元の状態に戻ってしまう。そこで、FPGA 内のメモリに回路を記憶させておき、電源を入れたときにその回路を自動的に読み込ませて常に同じ動作をするように改造する。

1. QuartusII の「Assignment」メニューから「Device」をクリックする。
2. 「Device and Pin Options...」ボタンをクリックし、Category から Configuration をクリックして選択し、Use configuration device にチェックを入れ、デバイスを「Auto」から「EPCS4」に変更する。OK ボタンを 2 回押して、ウィンドウを閉じる。
3. 一旦コンパイルする。
4. コンパイルが成功したら、FPGA ボードの電源を一旦切り、ボード上のスイッチ SW11 を PROG モードに切替えて電源を入れなおす。
5. 「Tools」メニューから「Programmer」を選択し、Mode を「JTAG」から「Active Serial Programing」に変更する。また、Add File ボタンをクリックして、コンパイル結果として得られたファイル baseball_top.pof を選択する。Program/Configure をチェックして Start ボタンをクリックしてしばらく待つ。
6. FPGA ボードの SW11 を RUN モードに切り替えると、電源を入れなおした際に、書き込んだ回路が読み込まれ、同じ動作をするようになる。



図：使用した FPGA ボード「DE0」

●レポート

今回のデジタルシステムの設計で行った改造の内容、方法、結果（成功・失敗等）について、通常のレポートと同様の体裁で報告せよ。