

ブラックジャックゲームの設計

中山 勝

筑波大学システム情報工学等技術室（装置開発班）

〒305-8573 茨城県つくば市天王台 1-1-1

概要

筑波大学情報学類・情報科学類の実験にプロセッサの設計開発というテーマがある。このテーマは、開発ツールを使って、ハードウェア記述言語（VHDL）で記述した論理回路を Field Programmable Gate Array (FPGA) に実装して行われる。そのため、開発ツールの使い方や、VHDL について学ぶ必要がある。

そこで学生が、論理回路設計に興味を持って取り組めるように、トランプゲームの、ブラックジャックゲームを設計させることを提案した。

キーワード：VHDL、FPGA、ブラックジャックゲーム

1. はじめに

ブラックジャックゲームは、トランプゲームの 1 つである。学生は、シミュレーションで各部の動作を確認しながら、VHDL で回路設計をする。回路が完成し動作に問題がないと判断したら、スパルタン 3 スタータキット（図 1）上の FPGA に実装し、動作させる。その過程において、開発ツールを使った、VHDL による論理回路設計と、シミュレーション方法を習得することができる。しかし、一からすべてを考え、設計させるには時間的に無理がある。そのため、基本回路の構成を提示し、その内容に沿って設計させることにした。

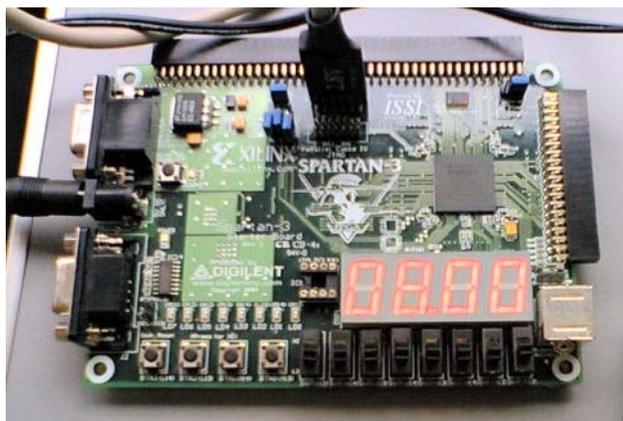


図 1. スパルタン 3 スタータキット

2. ブラックジャックゲームとは

表 1 にゲームの基本用語を示す。このゲームは、手札の合計がディーラーよりも大きい数で、かつ「21」を超えない範囲で「21」に近づけるトランプゲームである。まず、プレイヤーに 2 枚のカードを

オープンにして配る。その次にディーラーは、自分にオープンで 1 枚、伏せた状態で 1 枚配る。その後、プレイヤーがカードを続けて引き（引かなくてもよい）手札を決定する。次にディーラーは、手札の合計が 16 以下の場合、カードを引き続け、17 以上になったら Stand する。その際、絵札はすべて 10 と数えエースは 1 か、11 かを選べる。（今回の設計では、エースは 1 としか数えない。）

表 1 基本用語

用語	意味
プレイヤー	ボードを操作する人 (ボタンを押す)
ディーラー	カードを配る人 (論理回路)
Hit	カードを引く
Stand	これ以上カードを引かない
バースト	手札の合計が 21 を超えた

ゲームの進行中にプレイヤーがバーストした場合、無条件でディーラーの勝ちとなる。プレイヤーの手札の合計が 21 以下でディーラーの手札の合計より数が多い場合、もしくは、ディーラーがバーストしているときはプレイヤーの勝となり、双方同じ数の場合引き分けとなる。

3. ゲームの構成

ゲームを実現する上で、論理回路に必要な機能がいくつかある。このゲームは、プレイヤーとディーラーが 1 対 1 で対戦する。そこで論理回路には、ディーラーとしての判断ができる機能を持たせることになる。

3.1 ルーレット機能

ランダムにカードを配るために、ルーレットが必要となる。プレイヤーが 1 回ボタンを押すと回転し、もう 1 回押すと停止して配るカードが決まる。図 2 は、プレイヤーの 3 枚目のカードを配るため、ルーレットが回っている様子である。8 に見えるのは、1～d (16 進) を連続表示しているからで、ドットを点灯することで、ルーレットの回転と、配られたカードの値 8 を区別している。また、右の 2 つの値は、すでに配られたカードを示す。

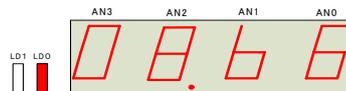


図 2. ルーレットが回転する様子

3.2 Stand 検出機能

プレイヤーの Stand は、ボタンが押されることで判断する。ディーラーの Stand は、カードを合計した値で決まる。論理回路により状況を判断し、正しくゲームを進めていく必要がある。

3.3 ゲーム進行機能

ゲームは、カードを配りながら進められる。論理回路は、次にどちらにカードを配るべきか知っている必要がある。また、配られたカードを把握し、ゲームの終了判定を行なう必要がある。

3.4 勝敗判定機能

ゲームが終了したとき、プレイヤー、ディーラーそれぞれのカードの合計を計算し勝敗判定をする。また、この時 16 進数で保存されている値を 10 進数で表示する (図 3)。



図 3. 勝敗表示

4. ユーザーインターフェイス

図 4 に示すように、スパルタン 3 スタータキットには出力として 7 セグメント LED が 4 個 (AN0~AN3)、発光ダイオードが 8 個 (LD0~LD7) あり、入力としてスライドスイッチが 8 個 (SW0~SW7)、押しボタンスイッチが 4 個 (BTN0~BTN3) ある。

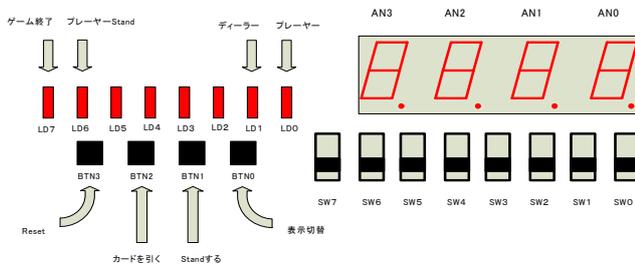


図 4. ボードの入出力

4.1 カード表示用 7 セグメント LED

7 セグメント LED は、カードの数字を表示する。7 セグメント LED の数は 4 個のため、プレイヤー、ディーラー各々最高 4 枚のカードで勝負するゲームルールにした。また、カードの数字は、A (エース)、2~10、J (ジャック)、Q (クイーン)、K (キング) だが、ここでは 1~d (16 進) となる。

4.2 状態表示用発光ダイオード

発光ダイオードは、現在の状態を表すために用いる。プレイヤー (LD0) とディーラー (LD1) は、現在どちらのカードが表示されているかを表す。また、最後の勝敗表示のときに勝ったほうが点灯する。

プレイヤー Stand (LD6) は、プレイヤーの手札が決定した際に点灯し、ゲーム終了 (LD7) は、プレイヤーとディーラーのすべてのカードが決定した際に点灯する。

4.3 押しボタンスイッチ

押しボタンスイッチは、論理回路に意思を伝えるため、またゲームを進めるための入力として用いる。それぞれ 4 つの押しボタンスイッチに、Reset (BTN3)、Hit (BTN2)、Stand (BTN1)、表示切り替え (BTN0) の機能を割り当てた。

5. 回路構成

ブラックジャックゲームを論理回路で構成するには、必要な機能毎にモジュールを分けたほうが得策である。そこで以降説明する 6 個のモジュールと 1 つのサブモジュールに分けることにした。

5.1 serve モジュール

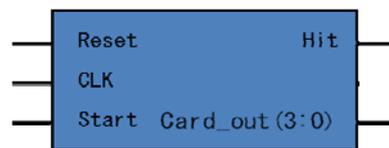


図 5. serve モジュール

図 5 に serve モジュールの入出力を示す。このモジュールは図 6 のようなステートマシンで、配られるカードを決める機能を持つ。Start 入力が 1 になるとルーレットが回転し、1~d (16 進) の値を連続的に出力する。Start 入力が 0 になると停止しその時の値を保持する。

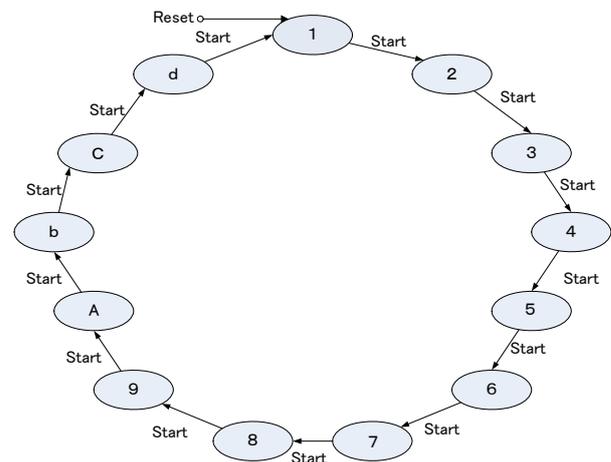


図 6. カードの値を出力するステートマシンの状態遷移図

また、カードが決まったことを知らせる信号として、1 クロックサイクルの信号 (Hit) を、図 7 に示すステートマシンで出力する。

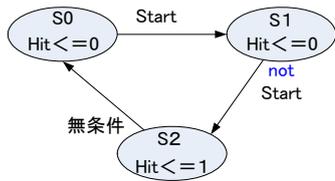


図 7. Hit 信号出力

5.2 phase モジュール

phase モジュールは、ゲームの進行状況を管理するモジュールで、図 8 に示す入出力を持つ。このモジュールも状態マシンからなり、図 9 に示す状態遷移図で表現できる。

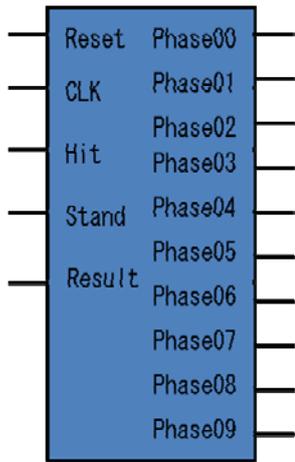


図 8. phase モジュール

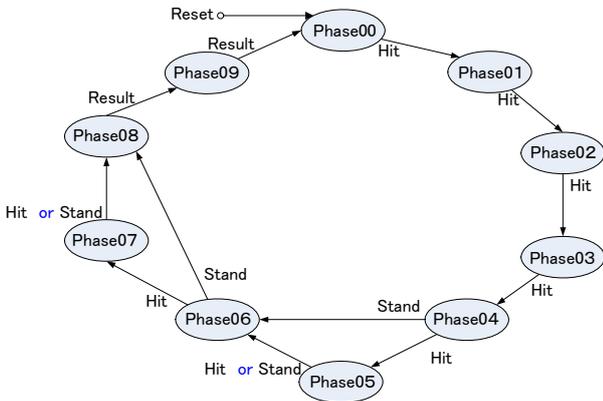


図 9. phase モジュールの状態遷移図

phase モジュールは、ゲームの進行中には Hit 信号または Stand 信号で遷移し、結果表示と初期状態に戻るときは Result 信号で遷移する。表 2 に、それぞれの状態の意味を示す。

表 2 Phase モジュールの状態の意味

状態	意味
Phase00	初期状態
Phase01	プレイヤーのカードが 1 枚配られた状態
Phase02	プレイヤーのカードが 2 枚配られた状態
Phase03	ディーラーのカードが 1 枚配られた状態
Phase04	ディーラーのカードが 2 枚配られた状態
Phase05	プレイヤーのカードが 3 枚配られた状態
Phase06	プレイヤーのカードが決まった状態
Phase07	ディーラーのカードが 3 枚配られた状態
Phase08	ディーラーのカードが決まった状態
Phase09	ゲーム結果を表示する状態

5.3 hand モジュール

図 10 に示す Hand モジュールはフリップフロップからなり、プレイヤー 4 枚、ディーラー 4 枚、計 8 枚のカードを記憶する。

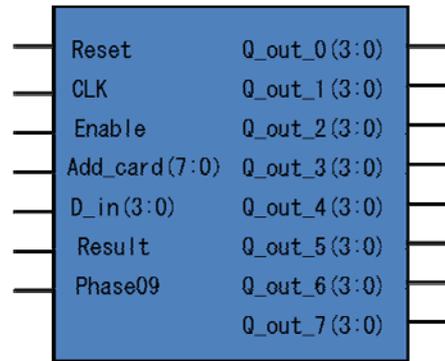


図 10. hand モジュール

5.4 selector モジュールと decide サブモジュール



図 11. selector モジュール

図 11 に、selector モジュールの入出力を示す。組み合わせ回路からなるこのモジュールは、Show 信号、Roulette 信号、Phase 信号の組み合わせで、発光ダイオード及び 7 セグメント LED の表示データを決める。また、下位モジュールとして decide サブモジュールを持つ (図 12)。decide サブモジュールも組み合わせ回路からなり、プレイヤーの手札の合計とディーラーの手札の合計を計算して勝敗を判定する。また、

各々の手札の合計の値を 16 進数から 10 進数に変換する機能を持つ。

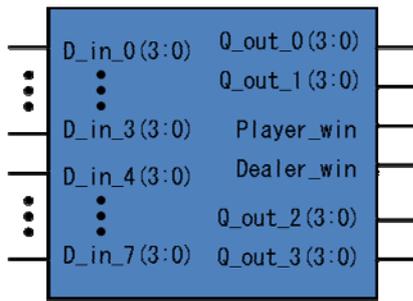


図 12. decide サブモジュール

5.5 stand_judge モジュール

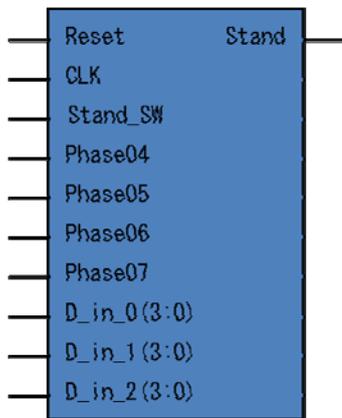


図 13. stand_judge モジュール

図 13 に示す stand_judge モジュールは、プレイヤーの手順で、Stand 入力があった時、1 クロックサイクルの信号を出力する。またディーラーの手順で、手札の合計が 17 以上の時も、同様の信号を出力する。

5.6 switch_ctl モジュール

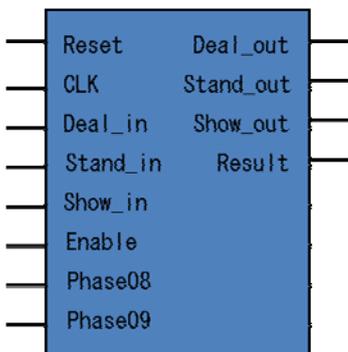


図 14. switch_ctl モジュール

Switch_ctl モジュール (図 14) は、押しボタンスイッチのチャタリングを取り除くと共に、それに対応する制御信号を作るモジュールである。Stand_out 信号と Show_out 信号は、イネーブル付きのフリッ

プフロップで設計する。Deal_out 信号と Result 信号は、同じ Deal_in 入力から作るがその時の Phase 入力によって作る信号を切り替える。それぞれの信号は、図 15、図 16 のようなステートマシンで設計する。

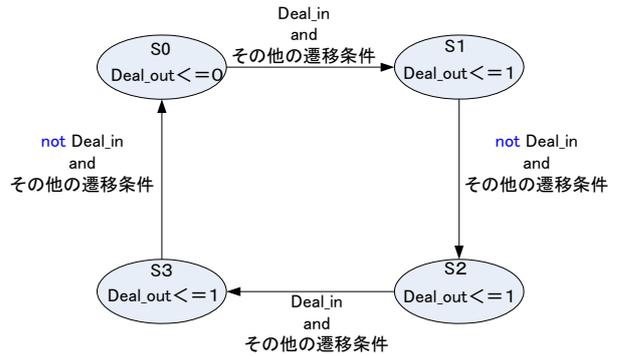


図 15. Deal_out 信号出力

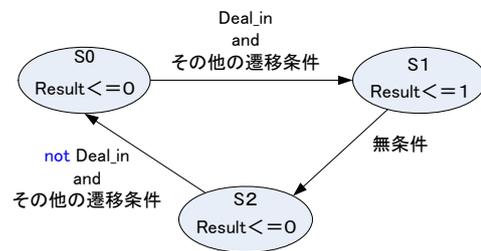


図 16. Result 信号出力

5.7 その他のモジュール

以上が、学生に設計させるモジュールとなる。しかし実際には、これらのモジュールだけでは動作しない。他に、スパルタン3 スタータキット上の 7 セグメント LED を 4 個同時にそれぞれの値を表示させるためのダイナミック点灯モジュールと、スイッチのチャタリングを防止するための Enable 信号を作るモジュールが必要となる。しかし、これらのモジュールまで設計させるのは学生に対して余計な負担になると考え、あらかじめ設計してあるものを提供することにした。

6. 回路の完成

今まで述べてきたモジュールを、図 17 の回路図の様に、VHDL で記述すれば回路の完成となる。

7. まとめ

現在、ブラックジャックゲームの設計は、実際に学生実験に取り入れられている。開発ツールの使い方、シミュレーションの方法、VHDL による論理回路の設計などを理解する上で、適当な規模の回路になっていると考える。

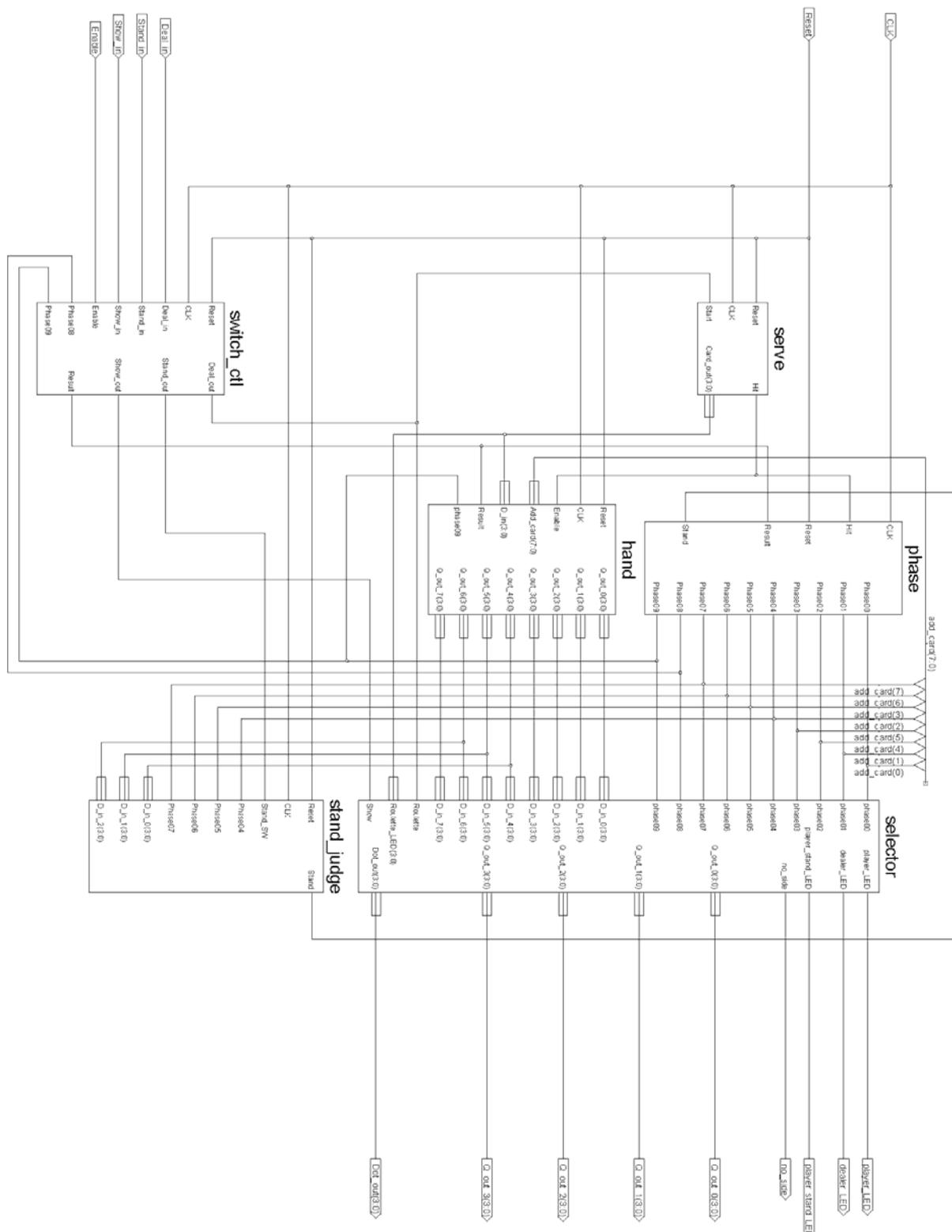


図 17. ブラックジャックゲームの回路図

謝辞

今回の提案を採用してくださった、システム情報工学研究科コンピュータサイエンス専攻の和田耕一教授に感謝いたします。また、学生に提供するモジュールを設計してくださった、システム情報工学等技術室（装置開発班）の小野雅晃技術専門職員に感謝いたします。

参考文献

- [1] 長谷川 裕恭 著, 改訂 VHDL によるハードウェア設計入門, CQ 出版.
- [2] Introduction to Programmable Logic, XILINX.

The design of Blackjack

Masaru Nakayama

Technical Service Office for Systems and Information Engineering, University of Tsukuba,
1-1-1 Tennodai, Tsukuba, Ibaraki, 305-8573 Japan

One assignment among the experiments in the College of Information Science in the University of Tsukuba is to design and develop a processor. Students complete this assignment by using a development tool to implement logic circuits specified in a hardware description language (VHDL) in an FPGA (Field Programmable Gate Array). To do that, students must learn how to use the development tool, and about VHDL.

Thus, it was proposed that students be asked to design a “Black Jack” card game, to make the process of logic circuit design more interesting.

Keywords: VHDL; FPGA; Blackjack