

投擲ロボットにおけるサーボモータの応答速度を 考慮した FPGA コントローラの試作

星野 孝総* 岸 雅基

(受領日：2013 年 5 月 2 日)

高知工科大学システム工学群
〒782-8502 高知県香美市土佐山田町宮ノ口 185

* E-mail: hoshino.yukinobu@kochi-tech.ac.jp

要約：ロボットとの定義とは、現在の私たちからすると産業活動に役立つ機能として捉えている事が多い。今日では、語源である *robota* の「労働」という人間に近い行動や考えを試行させる研究が進んでおり、今回は着地を行うことのできるロボットの実現を目指し、着地を実現させる為のシステムについて論ずる。近い将来、ロボットが歩く動作・走る動作・跳躍する動作を行うようになるであろうと考え、その際に求められる動作は着地であり、ロボットはロボット自身を守る機構が求められる。そして、その行為は人間自身をも守ることに繋がるかと考える。FPGA を使ったコントローラを試作し、サーボモータで駆動するロボットの落下実験を行い、着地時の姿勢の制御について考察する。

1. はじめに

本研究では、着地を行うことのできるロボットの実現を目指し、着地を実現させる為のシステムの一部について論ずる。近い将来、ロボットが歩く動作・走る動作・跳躍する動作を行うようになるであろう。その際に求められる動作は着地である。ロボットはロボット自身を守る機構が求められる。ロボットを守る行為は人間をも守ることに繋がるかと考え研究目的としている。

そこで、着地ロボットに使用されるサーボモータに最適な動作をさせるコントローラが必要である。また、サーボモータを動作させるのにマイコンではなく並列動作に適した FPGA でコントローラを試作する。そして、投擲ロボットにおけるサーボモータの応答速度を考慮した FPGA コントローラの試作し、投擲ロボットの可能性を示す。

2. 着地ロボット

着地ロボットの歴史において、山藤和男氏らの研究¹⁾で二足ロボットによる姿勢検出と着地制御が 90 年代に行われている。研究目的や目標は高い位

置からのロボットの軟着陸であり、ロボットを安全に地上に降ろすことである。猫が高い位置から着地しても無事でいられるように、ロボットにも同様の機能を実装できないか試行していた。機構としては成果を挙げる事ができたようである。ただし、猫が高い位置から着地できるのは猫が自身と地面との位置関係を視覚的に捉えることが前提である。よって内部機構だけでなく外界との関係も考慮しなければならないということも明らかになってきた。歩行に関する研究は多く行われてきたが、図 1 に示すような「着地」に関する研究はあまり行われていないのが現状のようである。

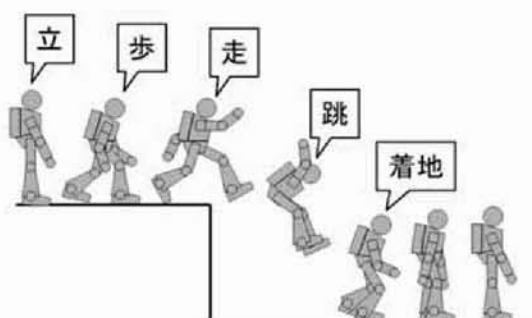


図 1. 将来求められると予想される動作

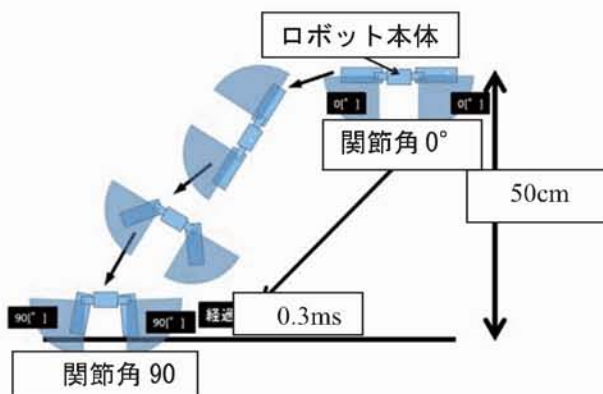


図 2. 求められる性能

本研究では二足歩行ロボット競技大会「ROBO-ONE」主催による「ROBO-ONE 宇宙大会予選競技会」のレギュレーションに沿ってロボットを作成した。

地上 50cm の高さから落下させて 0.3 秒以内に最適な動作を完了させることができるようにロボコン宇宙大会予選競技会でのロボットの動きを動作検証すると次のことが分かった。

- (1) 投擲後着地動作する際にロボットは 1 関節あたり 90° 以下の動作となる。
- (2) 落下までには 0.3 秒～0.4 秒程度の時間が経過する。

よって要求される性能は、図 2 に示すように 0.3 秒以内に関節を 90° 回転し、姿勢を整えなければならないこととなる。

一般的に PWM (Pulse Width Modulation) を実現するにはマイコンを利用することが多く、実際に競技大会に出場している機体はマイコンによる制御がほとんどである。システムを構築する際はフィードバック制御等を用いることがほとんどである。しかし、ロボットが落下するまでの時間が 0.3 秒～0.4 秒とフィードバックを制御する時間は殆どない。このようなミッションをクリアするためにはある一定の時間ごとに動作を行う時間制御を用いるほうが良い。システムの試作に多用される FPGA (Field Programmable Gate Array) は多種多様な PWM を試行するのに適したデバイスと判断でき、本研究ではマイコンは使用せず FPGA を利用する。

3. FPGA による PWM サーボモータコントローラの設計と試作

PWM とはパルス制御法 (Pulse Width Modulation) のことで、周期は一定で、入力信号 (DC レベル) の大きさに応じてパルス幅のデューティ・

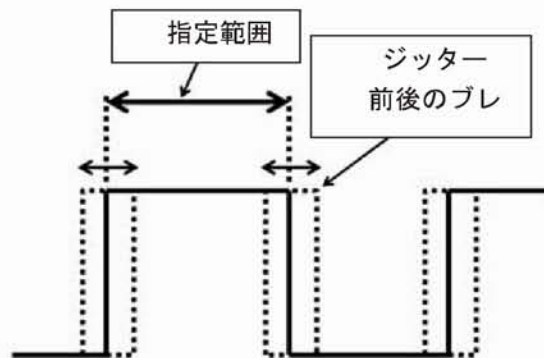


図 3. ジッターの発生

サイクル (パルス幅の H と L の比) を変え、モーターを制御する回路である。従来の制御に対して、これは飽和 (スイッチング) 領域での制御となる。従って、パワー・トランジスタを飽和領域で使用する為、電力ロスが軽減され、トランジスタもそれ程発熱しない。更に必要な時間だけ通電するので、モータ・ドライブ回路全体の効率があがり、電圧の負担も軽くなる。

FPGA を用いるメリットはマイコンと比べて、出力されるパルス (指令) 等を詳細かつ厳密に定義・実装できる点にある。マイコンのソフトウェアによる PWM 実装だとサーボの波形にゆらぎが発生してしまい、割り込みの発生から割り込み応答処理までの時間が一定にならないため、応答時間に誤差が生じる。これがジッターの原因となる。ジッターとは同じ PWM 波形を出力しているつもりでもサーボの波形にゆらぎが発生してしまう問題である。ジッターのある PWM でサーボモータを動作させた場合にサーボモータは振動するような動作を行ってしまう。マイコンのソフトウェア PWM によるジッターの解決は難しい。ジッターが全く無い状態つまり「ジッターフリー」を実現するには PWM を全てハードウェアとして実装してしまえば良い。

FPGA を用いて必要本数分の PWM をハードウェア実装することでジッターフリーの PWM を実装する。マイコンは使用するピンに機能が割り当てられており、ある程度は使用するピンを変更できるが、大幅な変更は難しい。FPGA の場合、電源や GND、クロック等の物理的なピンは決められているが、I/O ピンであればピンの位置を任意に変更することは可能である。PWM 信号を FPGA で出した場合、カウンタ数によってパルス幅を任意に変えるといった動作を容易に行わせることができる。このような動作をマイコンで実装するのは難しい。

そこで、FPGA には Xilinx 製 Spartan3E ファミリー

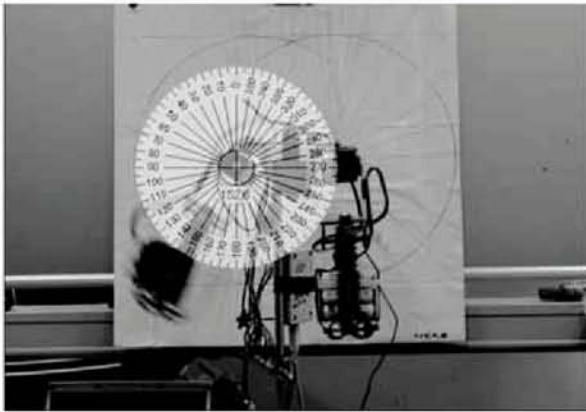


図 4. 計測方法

の XC3S100E と XC3S250E を採用した。サーボの応答速度や PWM モジュールを試作するのに XC3S100E を搭載した FPGA ボードを使用した。(CQ-EVSP3E100) ロボットに実装する際は小型薄型な FPGA ボード (MFPGA-SPAR3E) を使用した。最初は、FPGA は CQ-EVSP3E100 を使用し、PWM モジュールを試作した。また、RC サーボを動作させることができた。ロボットに搭載する際に FPGA ボードの基板面積が大きすぎマウントできない、そこで小型の FPGA ボードの方が望ましい。XC3S100E はロジック数が比較的少ない点から、ほぼ同じ原理で動き、ロジック数の大きい同系列の FPGA を選択した (XC3S250E)。

4. サーボモータ動作実験

FPGA による PWM を利用し、サーボモータを駆動させた。実験の目的は RC サーボモータを実際にロボットに組み込んでサーボモータに負荷 (バッテリー等の重量物を搭載した状態) をかけ、目標時間内 (0.3 秒以内) に 90 度の回転動作を完了できるかを検証・確認することである。検証方法は 60fps のデジタルカメラによる画像の連続撮影を行い、画像からサーボモータの挙動を把握する。クリップのついたハンガーを利用して空中にロボットを固定し、サーボを動かし FPGA から PWM を出力させ、サーボを駆動させた (図 4 参照)。サーボモータに一番負荷のかかるシチュエーションでの実験を行い、動作完了までの時間を動画撮影することで、コマ送りで動作を確認し応答速度を導き出した。デジタルカメラは 60fps のものを使用した。

<見つかった課題>

- 図 5 のグラフより 2 つの課題が見つかった。
- (1) オーバーシュート (波形が定常値となる基線を超過する現象のこと) の発生を抑える。

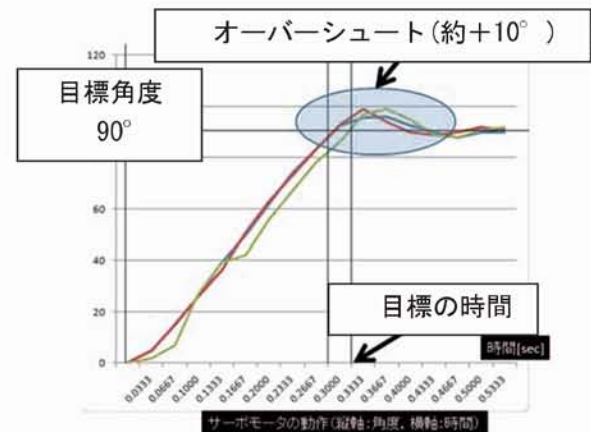


図 5. 動作時間グラフ

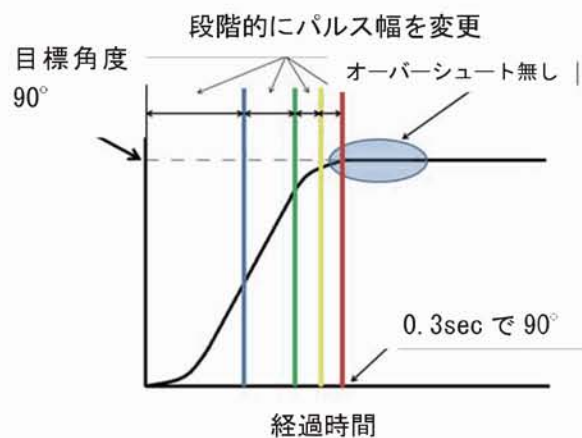


図 6. オーバーシュート防止手法

- (2) PWM の信号幅を経過時間に合わせて変化させ、最適な運用方法を探す (最適値) の以上 2 つである。

<オーバーシュートの防止>

解決手段として PWM の信号幅を経過時間とともに変化させる方法が挙げられる。PWM は 3msec おきに一回パルスを送る。送る回数をカウンタで指定し、PWM の信号幅を変化させ 3~4 段階に信号幅を分ける形をとった。(図 6 参照)

オーバーシュートを抑えるため、指令角度を微妙に変化させて実験を繰り返した。タスク 1 では開始角度を 0° つまり水平状態とし、45° の指令 (経過時間 0~0.1sec) を出す。0.15sec まで 80 度の指令を出し、0.15sec の段階で 90° の指令を行い、4 秒後にカウンタが回りきって 0° に戻る。今回の実験を通して「一回の (サーボの) 移動量によってオーバーシュートが発生するかが決まる」ということがわかってきた。一回の移動量が多いとサーボモータが PWM の指令に追従できず脱力もしくは脱調してし

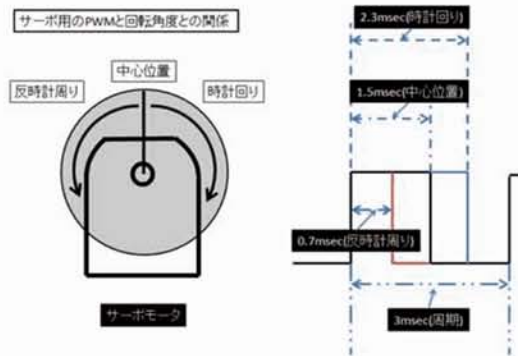


図 7. サーボ用の PWM と回転角度との関係

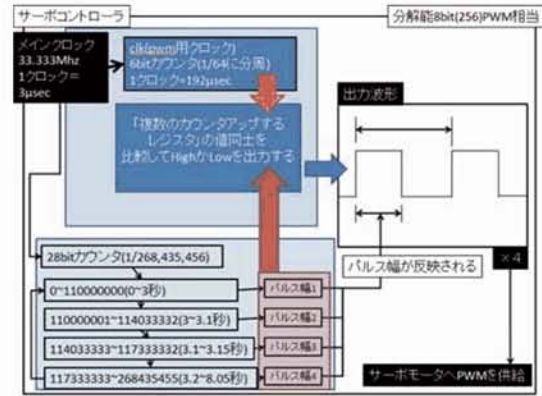


図 8. サーボコントローラの概要

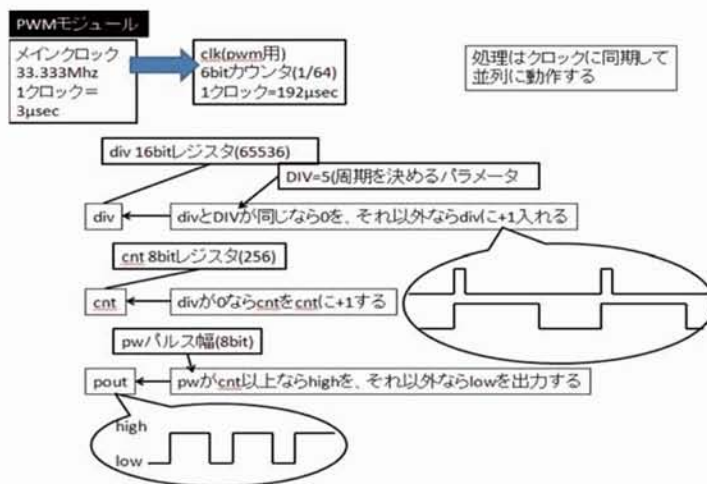


図 9. 使用した各種カウンタの様

まう。おそらく負荷がかけられた状態では応答速度が間に合わないのだろうと考えられる。0.3sec 以内に動作が完了しているので課題はクリアしたと判断した。そこで、この結果を受けて、本格的な FPGA サーボコントローラを設計する。

5. コントローラの作成

今回のコントローラでは RC サーボモータを使用する為、動作に必要な PWM はサーボモータの仕様に沿ったものでなくてはならない。サーボ用の PWM 波形は数ミリ秒の周期である。パルス幅はサーボモータのメーカーや型番によって異なるが 1.5msec をニュートラル位置にしている物が多く、パルスとしての有効な幅は 0.7msec~2.3msec 程度の物が多い。パルス幅とサーボモータの動作角度には線形な関係性があり、パルス幅を変更することでサーボモータを任意の角度に動作させることが可能である。通常、サーボモータ用の PWM を作成す

る場合、マイコンのカウンタや WDT(ウォッチドッグタイマ)を利用することが多く、マイコンの動作クロックからカウンタの回る時間を考慮して I/O 端子から任意の PWM を出力する。しかし、本研究ではマイコンではなく FPGA を利用する。FPGA を用いて PWM を作成する際もカウンタを利用して PWM を出力し、動作メインクロック 33.333Mhz を基にカウンタを数段組み合わせることで任意の PWM 波形を出力する形となる。クロック分周することで PWM のベースとなる信号を作り出す。分周した PWM モジュールのクロックは、図 8 のようになる。

PWM を生成するカウンタを図 9 に示す。(33.333[Mhz])(1 秒間に振動する数)/64 分周 (6 ビットカウンタ) = 520.820[khz]となる。1 クロックの周期は $1/520820[\text{hz}] = 192[\mu\text{sec}]$ 。PWM モジュール内の処理は、複数のカウンタアップするレジスタの値同士を比較し、High を出力するか Low を出力

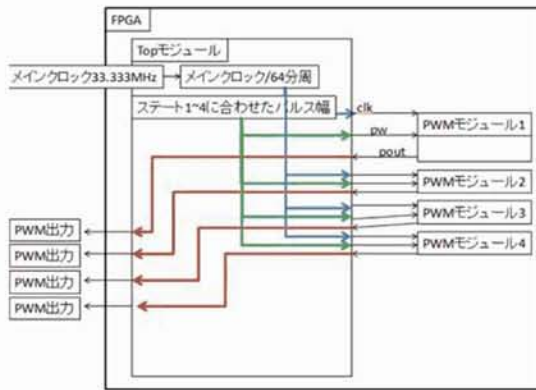


図 10. FPGA 内の信号接続

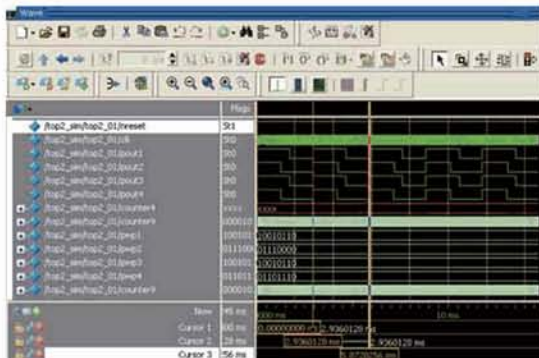


図 11. ModelSim によるシミュレーション

するかで判定している。div カウンタは 16bit カウンタで周期を決める DIV の値と比較し、一緒であれば div の値を 0 にリセットする。(周期を決める) cnt カウンタは div が 0 の場合に cnt を+1 する。Pw (信号幅) と cnt を比較し、pw が cnt 以上なら High を、それ以外なら Low を pout から出力する。

PWM に使用するクロックはモジュールに入力されている clk では誤差が大きい。そこで div カウンタを用いて任意のスピードのクロックを作る。これにより PWM を構成するカウンタ cnt で駆動させ、PWM のクロックとして採用した。サーボコントローラのシステムは大きく二つの機能がある。サーボ用の PWM を生成するモジュールと経過時間に合わせてサーボの指示角度を指定するモジュールで構成されている。4つのサーボモータを動かすことができるよう実装されている。実装された PWM は周期 3msec である。8bit PWM (分解能 256 段階) に相当する。(図 10 参照)

これらを設計して ModelSim を用いて 3msec 周期の PWM を確認し、パルス幅 0.7~2.3msec のサーボを動作させるのに有効な波形幅を出力することができた。(図 11 参照)

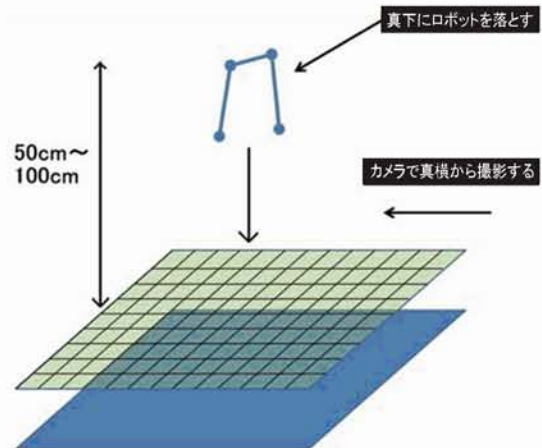


図 12. ロボットを自由落下させる



図 13. 落下実験の様子

6. 落下実験結果と考察

振り下げ動作と同じ動きをさせ、落下実験を行った。実験の目的は落下までに脚部を地面に向けることができているか確認を行う。(図 12 参照)

落下させた結果、固定した状態と同様、振り下げ動作はすべて落下するまでに完了した。落下する時間は 0.3 秒から 0.4 秒であった。(落下位置は網から高さ 60cm の位置) 高速カメラとデジタルカメラの画像を見る限り、動作は完了していることは確認できた。(図 13 参照)

高速カメラで撮影した画像からサーボモータの動き(角度等)をプロットしようと試みた。しかし、60fps では撮影したロボットが撮影ブレにより正確な角度をプロットすることができなかった。また、60fps 以上になると落下のタイミングに合わせて撮影が困難になる。投擲前の写真と落下の直前(仮想の地面)の写真等を確認すると仮想の地面に触れる前に動作が完了していることが分かった。固定した状態での実験と落下実験でのサーボの動きはほぼ同じと考えられる。(図 14 参照)



図 14. 落下実験の検証

7. 今後の課題

システムの一要素であるサーボコントロール部分を実装することはできた。応答速度も落下までの 0.3 秒以内に動作が完了しているので機構として十分である。しかしながら着地ロボットは前述したとおり、センサ（ジャイロ・加速度センサ）からのデータを基に動作を指示する機構を必要とする。これらの問題からセンサからの情報を処理し、コントローラに指令を出すまでの処理時間がどの程度必要なのかを調べるべきである。サーボの応答速度を考慮すると 0.1 秒以内に判断を完了しなければならないことが分かる。

またサーボは同機種のものであっても右回転時と左回転時では動作するスピードにおいて顕著に違いが見られた。PWM 信号の調整を行っても同時に動かすことは困難である。2010 年にはアナログ電圧出力タイプのジャイロセンサや加速度センサを用いて実験したが、ノイズが酷くとても使用に適していなかった。着地ロボットを実現するためには本研究のようなサーボをコントロールする機構だけでなく、センサ類のセンシング技術や処理方法等を深く考慮する必要がある。着地ロボットの実現はハードウェアによる性能に依存するところが大きいと言える。今回の研究では Xilinx を使用したが、Altera の Nios プロセッサを使用したほうがフレキシブルなシステムを効率よく構成できると考える。

8. まとめ

今回は FPGA によるサーボコントローラを実装し、投擲ロボットの着地時の姿勢制御について調査した結果、投擲後の着地ロボットの可能性を示すことができた。使用した RC サーボモータで着地ロボットを構成することは実験からも妥当であると判断できる。FPGA により RC サーボモータを制御することができ、落下までの 0.3 秒間に着地動作を完了させることができ、大きな問題点であった負荷がかかる場合におけるオーバーシュートの解決も PWM の信号幅を変化することで解決できた。今後

は、より軽量化と操作性の向上を目指す。

文献

- 1) 山藤和男, “空中における多関節 2 足ロボットの姿勢検出と着地制御,” 日本機械学会論文集, C 編 59(565), pp.2780-2787, 1993-09-25
- 2) 河村隆, “空中を落下するロボットの姿勢制御と軟着地 (1 報~3 報),” 日本機械学会論文集, C 編 58(545), 151-155, 1992-01-25
- 3) 山藤和男, “空中に投げられたロボットの姿勢検出と着地制御,” 電気通信大学紀要 7(1), 33-39, 1994-06
- 4) 岩田敏彰, “AMBAC[Active Mass Balance Auto Control(System)]-手足の運動を利用した方向制御機能,” 計測と制御, 43(1), 8-9, 2004-01-10
- 5) 西村耀, “二足歩行ロボット格闘 ROBO-ONE のチャレンジャー投げたロボットは立てるのか?,” 計算工学, 13(3), 1878-1881, 2008-07-31, 日本計算工学会
- 6) 山藤 和男, “ロボットに関する独創的な研究開発をめざして ロボット猫の軟着地と 3 次元ひねり,” 機械の研究 (養賢堂), 51(3), 391-397, 1999-03
- 7) 小林 優, “入門 Verilog HDL 記述—ハードウェア記述言語の速習&実践 (Design wave basic),” 出版社: CQ 出版; 改訂版 (2004)
- 8) 一般社団法人日本ロボット学会 (編集), “ロボットテクノロジー,” オーム社 (2011)
- 9) 小林 優, “FPGA ボードで学ぶ組み込みシステム開発入門[Altera 編],” 技術評論社 (2011)
- 10) ROBO-ONE 委員会(編集), “ROBO-ONE で進化する二足歩行ロボットの造り方,” オーム社 (2010)
- 11) 渡辺嘉二郎, 小俣善史, “図解ロボット技術入門シリーズ ロボット入門,” オーム社 (2006)
- 12) 古田勝久, “ロボット・メカトロニクス教科書,” メカトロニクス概論, オーム社 (2007)

Experimental FPGA Controller for Throwing Robots Moved by Servo Motors

Yukinobu Hoshino* **Masanori Kishi**

(Received: May 2nd, 2013)

School of Engineering, Kochi University of Technology
185 Tosayamadacho-Miyanokuchi, Kami, Kochi, 782-8502, JAPAN

* E-mail: hoshino.yukinobu@kochi-tech.ac.jp

Abstract: The definition of robot is considered to be an ordinary function as a machine in industry. Also, the origin of the word “robota” meant physical labor in Czech and Slovakia. Today, many people studies to make robot act or think like human. This paper, we report our challenge for FPGA controller for throwing robot moved by servo motors which can touchdown. We describe about a study of touchdown systems. We expect that robot will be able to walk, run, jump and touchdown in the near future. Then robot needs systems which guards for protects oneself from shock and control balance. We expect the robot systems would share the human life and protect human being. So in this report, we have developed a controller using FPGA. And, experiment is drop the robot and driven by a servo motor to touch down. This paper's report is the result of the attitude control at the time of the landing.

