

FPGA を用いた自動運転ロボットカーの開発

児島 彰

広島市立大学

1. はじめに

自動運転技術のためのリアルタイムの画像処理を FPGA 搭載システムで実現することを題材とした FPGA 設計コンテストが開催されてきた。本稿は、2020 年に開催される「第 9 回 相磯秀夫杯 FPGA デザインコンテスト⁽¹⁾」のために、我々が開発している小型ロボットカーのシステム構成や制御方式について述べる。

2. 設計コンテストの概要

設計コンテストでは、アルゴリズムは FPGA 内に実装し、FPGA とカメラを搭載し、バッテリー駆動により、単体で自動走行する小型のロボットカーを設計・開発する。カメラからの画像をリアルタイムに画像処理して走行を制御する。道幅約 30cm の道路を模したミニコースを使用し、コース上には信号、障害物、人を模した人形が配置される。交通ルールに沿った規定走行を行わせ、走行を採点する。

3. システム構成

図 1 は、走行コース上に置かれた開発している FPGA ロボットカーの写真で、図 2 は、そのシステム構成である。FPGA ボードは、AVNET Ultra96⁽²⁾ を使用し、広角の USB Web カメラを接続して搭載している。バッテリーは入手性のよい 5V モバイルバッテリーを 2 個使用している。車体はステアリング制御タイプで、TAMIYA「バギー工作基本セット」をベースに改造して、サーボモータでステアリング制御できるようにしている。また、小回りが効くように、ホイールベースを標準的車台より若干短くしている。モータ制御信号のための電圧レベル変換やバッテリー接続などを行うインターフェースボードは小型のユニバーサル基板に実装した。

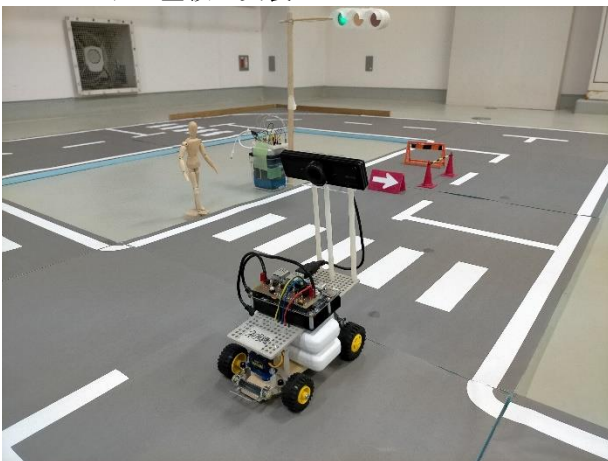


図 1 走行コース上の FPGA ロボットカー

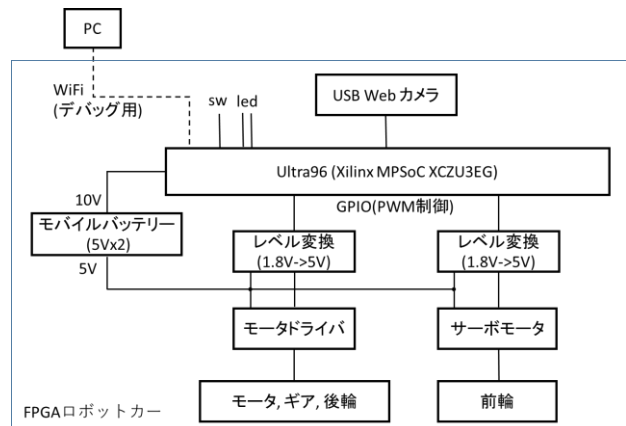


図 2 システム構成図

3. 制御方式

Ultra96 搭載の FPGA である Xilinx Zynq Ultra Scale++ MPSoc ZCU3E は、ARM プロセッサ部と再構成可能なハードウェアロジック部で構成されている。OS は Ubuntu18.04 の PYNQ2.5⁽³⁾ 環境を使用している。全体のシステム制御はプロセッサ部で動作する Linux 上のソフトウェアで実現し、コース上の物体検出はハードウェアロジック回路を使用する。また、走行に必要なモータ制御やサーボモータ制御に使用する PWM 制御回路もハードウェアロジック部に構成している。

コースの基本走行の制御のメインルーチンは Python で記述している。白線検出は OpenCV を使ったソフトウェアで行っている。物体検出には、機械学習させた Yolo v3 の DNN を変換して、Xilinx DPU⁽⁴⁾ を使用している。現時点の物体検出は白線検出に比べてフレームレートが毎秒数フレームと遅いので、白線検出と走行制御を行うメインスレッドとは別に、物体検出を行うサブスレッドとして動作させている。物体検出を除いた、白線検出によるメイン走行制御ループは、カメラ画像に対して、平均 15fps で処理を行っている。FPGA の合成には Vivado 2019.2 を使用している。現在のバージョンの合成結果の資源使用率は LUT 56%、LUTRAM 11%、FF 47%、BRAM 59%、DSP 87% である。

4. まとめ

FPGA 設計コンテストに向けて開発を行っている自動運転ロボットカーについて述べた。今後は物体検出の精度向上や処理の速度向上を行っていきたい。

参考文献

- (1) <https://www.shizuoka.ac.jp/fpt-design-contest/aiso9-committee/> 第 9 回 相磯秀夫杯 FPGA デザインコンテスト, (2020/10/16 最終アクセス).
- (2) <https://www.avnet.com/wps/portal/japan/products/product-highlights/ultra96/> AVNET Ultra96 開発ボード (2020/10/16 最終アクセス).
- (3) <http://www.pynq.io/> PYNQ, (2020/10/16 最終アクセス).
- (4) <https://github.com/Xilinx/DPU-PYNQ> DPU on PYNQ (2020/10/16 最終アクセス).