

FPGA によるマイクロドローンの自律飛行制御の可能性

程 嘉豪[†]山口佳樹[‡][†]筑波大学大学院システム情報工学研究科コンピュータサイエンス専攻[‡]筑波大学システム情報系

1 はじめに

人間中心社会をキーワードとした次世代 IoT プラットフォームとしてドローンというフレームワークに注目が集まっている。日本再興戦略 2016 にドローンというキーワードが登場し、経産省「平成 30 年度経済産業政策」、文科省「次世代火山研究・人材育成総合プロジェクト」、農水省「G空間プロジェクト」、また官民研究開発投資拡大プログラム (PRISM) など、様々な場所でドローンは注目技術として取り上げられている。ドローンの社会実装を考えたとき、安定飛行や特定問題への最適化に加え、消費電力効率を大きく改善する必要がある。そこで本稿では、屋内とくに工場や施設などパイプや壁面、またビル等の空調システムの検査をする屋内向けドローンについて検討することとした。

このとき、屋内という条件から超小型ドローンが好ましいが、以下の 3 つの問題を抱えていることがわかった。まず、超小型ドローンを初め大きさの小さいドローンの方が検査等には望ましいが連続飛行時間に厳しい制約があること。次に、位置把握において、GPS や気圧計を用いて正確な距離を得るのが難しいこと。最後に、操縦者から見えない範囲にあるドローンを実時間で遠隔操縦することは困難ということ。これらを解決するため、小規模 FPGA を利用した自律飛行について提案および検討を行った。

本稿の構成は、以下の通りである。2 章では、ドローンについて述べ、3 章では FPGA 導入の理由を、4 章では実験環境について述べる。5 章では、FPGA 実装について触れ、最後に、6 章で本稿の結論と今後の課題を述べ、本稿をとじる。

2 ドローンの概要

本稿で議論するドローン (Drone) は、聞き手の背景により、その用語から類推される機能や性能が異なることが多い。そこで本稿では、まずドローンについて大まかに分類し、その分類に基づいて対象としたドローンの議論を行う。

2.1 ドローンの種類

ドローンは、広義には無人潜水艦 (UUV: Unmanned underwater vehicle) [1, 2] や無人船 (USV: Unmanned surface vehicle) [3] など含まれる。しかし、本稿ではドローンは無人航空機 (UAV: Unmanned Aerial Vehicle) として考えるものとする。

次に、無人航空機は浮力を利用する軽航空機と揚力を利用する重航空機に分類される。軽航空機には気球や飛行船などが含まれ、重航空機には飛行機、滑空機、ヘリコプターが含まれる。また、動力という観点で航空機の翼に目を向けると、回転翼 (Multi-rotor) [4, 5] と固定翼 (Fixed Wing) [6] のタイプに分けることができる。固定翼機は、その安定性と比較的長時間の運用が可能なることから、写真測量アプリケーションなどで利用されている [6]。一方、本稿の提案するドローンは、その応用利用する範囲は屋内を想定しており、定点監視や特定箇所の点検作業などへの利用を考えている。このため、前進速度によって浮力を得る固定翼機は、本稿の目的を実現する手段として利用することが難しい。

そこで、本稿では回転翼タイプのドローンについて議論することとし、本章以降においてドローンという単語は回転翼機のドローンを指すものとする。

2.2 連続運用時間

仏 Parrot 社による AR.Drone の販売などをきっかけに、わが国でも 2010 年頃より回転翼タイプのマルチロータードローン (または、マルチコプター) の普及が始まった。マルチコプターは、ホバリングによる空中での停止や運用における柔軟性が非常に高く、幅広いアプリケーションに適用できるという意味で、非常に魅力的なデバイスである。一方、ローターを中心とする動力系で要求される消費電力が大きかったため、運用時間が短いという欠点があった。そこで、本体の軽量化、空力性能の改善、ローターの改良、二次電池の進化、制御アルゴリズムの効率化などについて研究開発が進められている。ここで、市場から入手可能なマルチコプターの重量、飛行時間、電力消費量の関係を図 1 に示す。

図 1 においてマルチコプターの重量について着目すると、極小型 (50g 未満)、小型 (51~200g)、中型 (201~700g)、大型 (701~1500g)、特大型 (1501g 以上)、の大きく 5 つのクラスに分類できる。大型~特大型のマルチコプターは、その積載可能量を大きく取れるため、充電池の大型化により飛行時間を比較的長くすること (20 分以上) が容易である。また、中型の

A Study of FPGA-based Micro-drone Controller

Jiahao Cheng[†] and Yoshiaki YAMAGUCHI[‡][†]Graduate School of Systems and Information Engineering, University of Tsukuba[‡]Faculty of Engineering, Information and Systems University of Tsukuba

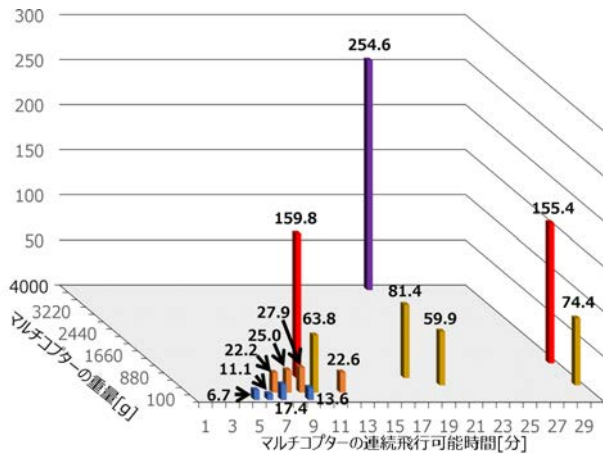


図 1: 重量, 飛行時間, 消費電力 (表中の数字) の関係

マルチコプターは, ここ数年で電力効率を飛躍的に向上させており, 飛行時間の長くすることに成功している. 一方, 超小型~小型のマルチコプターは, 10 分を超える運用時間の実現をどのように解決するかが喫緊の課題となっている. この解決には, 中型や大型~超大型のマルチコプターと同じ解決策を選択できないため, 別の解決提案が要求されている. そこで, この問題を解決するハードウェアシステムについての提案および検証が本稿の貢献である.

2.3 マルチコプターの消費電力

AR.Drone が登場した 2010 年頃, マルチコプターに搭載される撮像素子の解像度は VGA 画像 (約 30 万画素) 程度であった. それが, 2018 年になると, 6K (約 2700 万画素) まで高精細化している. つまり, 約 8 年で約 100 倍の画素数となっており, この解像度の画像に対して複雑な処理を採用した場合, マルチコプター上で 1000 倍以上の演算性能が要求されることとなる. 図 2 に, 単純な画像処理フィルターを適用したときの画素数に対する演算量の変化を示す.

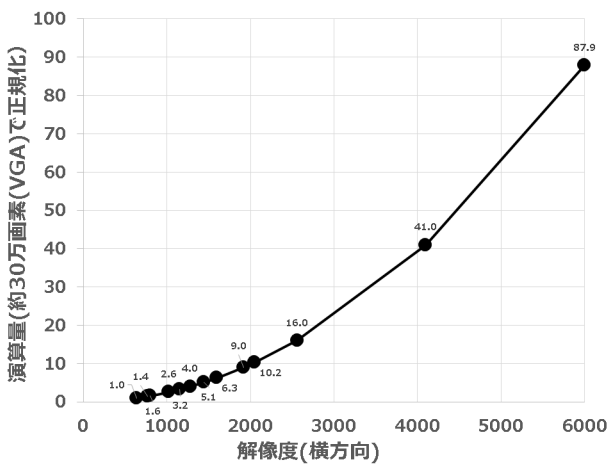


図 2: 画像の解像度に対する演算量の変化の例

ここで, マルチコプターのシステム全体で要求され

るとマルチコプター上の演算に要求とされる消費電力の比率について, 図 1 の値を基に推定された値を図 3 に示す.

マルチコプターの電力効率は年々上昇しており, 将来的には 0.15~0.1[W/g] になると考えられる. このとき, 図 2 で示したように, 画像解像度に対する演算量が増加していると仮定すると演算部の消費電力がより支配的になる可能性がある. これまでは, 画像解像度が低く電力効率も低かったため演算部の消費電力には注目されていなかったが, 今後はマルチコプターをより広い実アプリケーションに適用する上で重要なファクターになると考えられる.

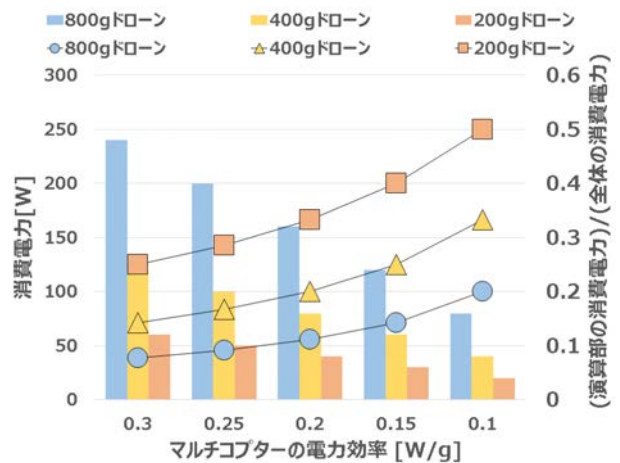


図 3: 全消費電力に占める演算による消費電力の割合

3 マルチコプターの省電力化と FPGA

本稿で想定している屋内向けアプリケーションでは, マルチコプターの座標を GPS (Global Positioning System) または RTK-GPS [9] などの手法により得ることが難しい. 気圧計による高精度なフロア検知も提案されているが [10], 配管の計測などはセンチメートル単位での制御が必要なるため, 本稿のアプリケーションに十分とは言えない. 同様の意味で, 加速度計による位置推定も十分とは言えないため, 画像処理のみ (ないし併用) での位置制御が求められる. このため, 前章で示したように, 撮影画像を実時間で演算処理するのは非常に困難となりつつあるためその工夫が求められる.

組み込み GPU 利用は一つの解決策として提案されている [11]. 例えば, NVIDIA Jetson における 5 [W] という消費電力と 472 GFLOPs の性能 (もしくは, 10 [W] という消費電力と 1.3 TFLOPs の性能) は非常に高い消費電力対性能であることは確かである. これは, 中規模以上のマルチコプターでは確かに有効である. しかし, 超小型~小型のマルチコプターに搭載される充電電池の容量は 500mAh 程度であることが多く, 組み込み GPU を用いて十分な飛行時間を確保するのは非常に難しい.

ここで, 書き換え可能である FPGA (Field Programmable Gate Array) の利用が考えられる [8].

マイクロコプターでできることは限られており、それに合わせて最適化した演算回路を提案できれば、消費電力を大きく削減できる可能性がある。加えて、Lattice 社製 iCE40 FPGA [12] のような低消費電力型 FPGA を利用することで、待機時電力 ($\sim 50\mu\text{W}$) および動作時電力 ($\sim 50\text{mW}$) について、組み込み GPU の消費電力と比較し 1/100 程度もしくはそれ以上に小さくできる可能性がある。

4 実験環境

本章では、本稿で用いたマイクロドローン環境およびその性能について述べる。

4.1 マイクロドローン環境

本稿では、bitcraze 社による crazyflie 2.1 をプラットフォームとして選択した [13]。図 4 に、FPGA ボードを取り付けた状態の crazyflie 2.1 を示す。



図 4: マイクロドローン (crazyflie 2.1)

図 4 に示すように、crazyflie 2.1 はクワッドコプターであり、重さは約 27g と超軽量のマイクロドローンである。搭載された充電電池で約 7 分間の飛行が可能であり、積載可能重量は 15g である。飛行時、マルチコプターの姿勢制御は、ST Microelectronics 社のマイコンである STM32F405 が行っている。また、crazyflie はハードウェアも含めてオープンソースという形態をとっており、crazyflie を利用した様々な研究が報告されている [14, 15, 16]

4.2 FPGA 基板

本稿では、前章の crazyflie 2.1 に FPGA ボードを組み合わせている。図 5 に使用した FPGA ボードの拡大写真を示す。

図 5 の FPGA ボードは、産業技術総合研究ナノエレクトロニクス研究部門の小池帆平氏により製作された TIA かけはしボードであり、XILINX 社製 Spartan-6 XC6SLX9 FPGA を搭載したボードとなっている。この FPGA は、5,720 個のスライス LUT, 11,440 個のフリップフロップ, 32 個の 18Kbit ブロック RAM を備えた小規模 FPGA である [17]。2019 年 6 月現在、より

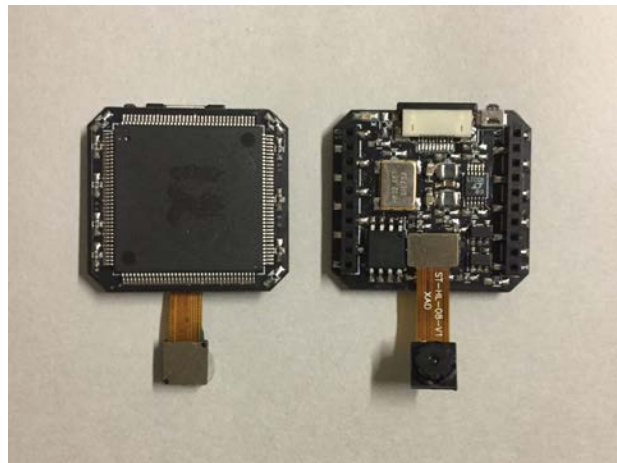


図 5: FPGA ボード

大規模な FPGA を搭載することは可能だが、Lattice 社の低消費電力型 FPGA [12] や今後登場する不揮発性デバイス [18] の容量などを考慮してあえて小規模 FPGA を採用した。また、CMOS センサとして、オムニビジョン社の OV7660 を搭載している [19]。本稿では、このセンサにより撮影された画像情報に演算処理を適用し、自律飛行することを目指した。

5 実装回路

FPGA の実装において 32 bit の加算器を考えたとき、入出力のレジスタとして 96 個のフリップフロップ、演算として 48 個程度のロジックセルを必要とする。このため、本稿で選択した FPGA では大規模な計算は実装できず、特徴抽出や画像認識などを行うことができない。そこで対象アプリケーションを屋内に限定していることから、FPGA 内にいくつかのテンプレートを用意し、そのテンプレートに、ホバリング、直進、旋回、停止などの意味を持たせることとした。FPGA を搭載したマルチコプターは、そのテンプレートを読み取り、ほぼ同時刻に処理を行い、指示をモーターを制御している PIC へと

5.1 RGP 色空間から HSV 色空間の変換

本稿で使用した OV7660 CMOS イメージセンサは RGB 色空間を使用している。RGB 空間において、色は R(赤), G(緑), B(青) の 3 次元ベクトルで表現される。当初、本稿の実験ではこの RGB 空間を用いていたが、光源や影の影響により、安定した色判別を行うことができなかった。そこで、RGB 色空間を HSV 色空間へと変更し演算処理を行うこととした。HSV 空間において色は、色相 (Hue), 彩度 (Saturation), 明度 (Value) の 3 次元からなるベクトルで表現される。

このとき, H , S , V はそれぞれ以下の式で計算される.

$$H = \begin{cases} 60 \times \frac{G-B}{MAX-MIN} & (if MAX = R) \\ 60 \times \frac{B-R}{MAX-MIN} + 120 & (if MAX = G) \\ 60 \times \frac{R-G}{MAX-MIN} + 240 & (if MAX = B) \\ 0 & (if MIN = MAX) \end{cases} \quad (1)$$

$$V = MAX \quad (2)$$

$$(3)$$

$$S = \begin{cases} MAX - MIN & if \text{ cylinder model} \\ \frac{MAX-MIN}{MAX} & if \text{ cone model} \end{cases} \quad (4)$$

上記で紹介した数式には乗算や除算が含まれるため, そのままプログラム化しても FPGA には実装できない. 本稿が FPGA を使用する理由は画像処理からの制御信号生成であり, 色空間の忠実な再現ではない. そこで, 幾らかの式変形を行うことで演算の小型化を図っている.

ここで, 表 1 に回路使用量を示す. RGB から HSV の値変更は複雑だが, 非常にコンパクトな回路で実現できたことがわかる. また, 現在は 4 つのテンプレートパターン (直進, 空中停止, 旋回, 停止 (着陸)) についてのみしか実装されていないが, 今後はより複雑な動作に対応するようテンプレートを増やしていく予定である.

表 1: 回路使用量

回路	使用個数	全体
Register	125	11440
LUT	216	5720
BRAM	0	32
DSP	0	16

6 まとめ

本稿では, 屋内かつ複雑な形状を持つ空間を対象とした調査システムについて検討を行った. マルチコプターによる実装が現実的であったが, (1) 超小型ドローンを初め大きさの小さいドローンの方が検査等には望ましいが連続飛行時間に厳しい制約があり, (2) GPS や気圧計を用いて正確な距離を得るのが難しい, (3) 操縦者から見えない範囲にあるドローンを実時間で遠隔操縦することは困難, という理由から小規模 FPGA を利用した自律飛行について提案および検討を行った. (1) については, 実応用において最低限必要な処理を絞りこむことで, 組み込み CPU や組み込み GPU ではなく小規模 FPGA で実現可能な問題に落とし込むことで開発を図った. (2) については, センチメートルオーダーでの移動を受動型センサのみで行うことは非常に難しいことから, 画像処理による飛行制御を採用

することとした. このとき, FPGA の演算回路の制約から複雑な計算はできないため, 簡単なテンプレート (紙/シール) を利用して飛行ルートを指示することで解決を図った. (3) については, (2) の仕組みを利用することで, 遠隔操縦を不要とすることで解決を図った.

現在, FPGA の開発はほぼ完了し, マルチコプターとの連携 (マルチコプターを動作させるソフトウェアとの連携) 部分について開発を進めているところである. 今後の課題は, FPGA とマルチコプターを機能的に一体化させ, きちんと飛行制御することを示すことである. また, 積載可能量が約 10g 残っているため, これに合わせて充電池を増強し, 飛行時間を 20 分程度まで延長することを考えている.

謝辞

本研究の一部は, 科学研究費補助金 (JP17H01707) および TIA 連携プログラム探索事業「かけはし」(2019 年度) の助成を受けたものである. また, Xilinx 社より「Xilinx University Program」を通じて開発ソフトウェアの支援を受けており, ここに謝意を表す.

参考文献

- [1] Lin Meng, Takuma Hirayama, and Shigeru Oyanagi, “Underwater-Drone With Panoramic Camera for Automatic Fish Recognition Based on Deep Learning”, in IEEE Access, Vol.6, pp.17880–17886, 2018.
- [2] Bogusław Cyganek and Kazimierz Wiatr, “Design of a Visual Front-End for Parallel Signal Processing on Underwater Search Drone”, 2018 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Ubiquitous Computing & Communications, Big Data & Cloud Computing, Social Computing & Networking, Sustainable Computing & Communications, pp.1046–1047, 2018.
- [3] Ronald O’Rourke, “Navy Large Unmanned Surface and Undersea Vehicles: Background and Issues for Congress”, Congressional Research Service Report R45757, 19 June 2019.
- [4] Rihan Altawy and Amr M. Youssef, “Security, Privacy, and Safety Aspects of Civilian Drones: A Survey”, ACM Transactions on Cyber-Physical Systems, Vol.1, No.2, Article 7, pp.1–25, Feb. 2017.
- [5] Emad Ebeid, Martin Skriver, and Jie Jin, “A Survey on Open-Source Flight Control Platforms of Unmanned Aerial Vehicle”, 2017 Euromicro Conference on Digital System Design, pp.396–402, 2017.
- [6] Alessandro Montanari, Fredrika Kringberg, Alice Valentini, Cecilia Mascolo, and Amanda Prorok, “Surveying Areas in Developing Regions

- Through Context Aware Drone Mobility”, Proc. of the 4th ACM Workshop on Micro Aerial Vehicle Networks, Systems, and Applications, pp.27–32, 2018.
- [7] ジオサーフ株式会社, “eBee RTK パフォーマンス試験レポート –これが固定翼のパフォーマンスだ!”, 2016年9月1日, https://www.geosurf.net/wp/wp-content/uploads/2018/09/ebertk_performance.pdf (最終閲覧日: 2019年6月20日)
- [8] 天野英晴 (編集), “FPGA の原理と構成”, オーム社, 2016年4月22日.
- [9] Hiromune Namie, “The Principle and the Practice of the RTK-GPS”, the GPS International Symposium '01, the Civil GPS Service Interface Committee, the International Information Subcommittee, the 3rd Asia Pacific Rim Meeting Tutorial Lecture, The Proceedings (I) Tutorial Text, pp.21-38, 2001.
- [10] 一刈良介, ルイス カルロス, マンリケ ルイズ, 興梠正克, 蔵田 武志, 北川 朋亮, 吉井 宗太, “気圧変動の要因分解に基づく屋内測位向け高度変化検出手法”, 信学技報, Vol.115, No.245, pp.31-36, 2015.
- [11] NVIDIA Corp., “NVIDIA Jetson”, <https://developer.nvidia.com/embedded/develop/hardware>. (最終閲覧日: 2019年6月20日)
- [12] Lattice Semiconductor, “An FPGA Approach to Implementing Time-Critical Functions in Multi-Sensor Mobile Designs”, A Lattice Semiconductor White Paper, October 2013.
- [13] bitcraze, “Crazyflie 2.1”, <https://store.bitcraze.io/products/crazyflie-2-1>, 2019. (最終閲覧日: 2019年6月20日)
- [14] Luke Kamrath and James Hereford, “Development of Autonomous Quadcopter”, IEEE SoutheastCon 2017, pp.1-6, April 2017.
- [15] Yash Vardhan Pant, Houssam Abbas, Rhudii A. Quaye, Rahul Mangharam, “Fly-by-Logic: Control of Multi-Drone Fleets with Temporal Logic Objectives”, 2018 9th ACM/IEEE International Conference on Cyber-Physical Systems, pp.186–197, August 2018.
- [16] Wojciech Giernacki, Mateusz Skwierczyński, Wojciech Witwicki, Paweł Wroński, Piotr Kozierski, “Crazyflie 2.0 quadrotor as a platform for research and education in robotics and control engineering”, 2017 22nd International Conference on Methods and Models in Automation and Robotics, pp.37-42, 2017.
- [17] Xilinx, “Spartan-6 ファミリー概要”, DS160, Ver.2.0, 2011年10月25日.
- [18] NEC, “Nanobridge 技術搭載 LSI のサンプル製造開始”, https://jpn.nec.com/press/201710/20171019_03.html, 2017年10月19日. (最終閲覧日: 2019年6月20日)
- [19] OmniVision, “OV7670/OV7171 CMOS VGA (640x480) CameraChip with OmniPixel Technology”, Advanced Information Preliminary Datasheet, Ver.1.01, July 8 2005.