

1. それぞれのセンサ値をうまく取得できるようになった件

以前の本ミーティングにて示した、オプティカルフローセンサ及び UWB からの値を得るためのシステム構成を再度 Fig. 1 に示す。

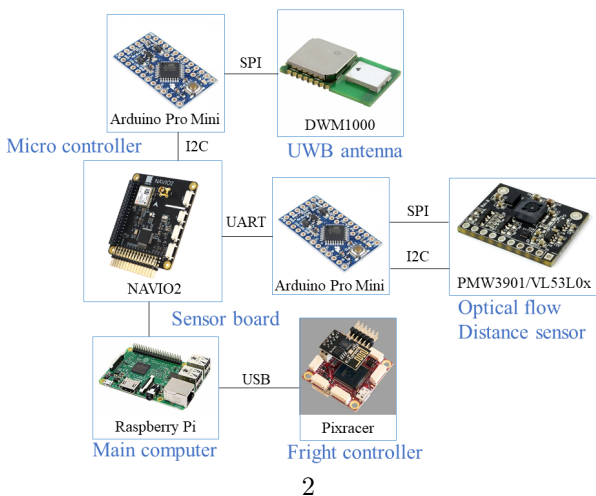


Fig.1: 以前に示した通信方式に I2C を用いたシステム構成

この構成ではドローンに設置した UWB タグの距離データを Arduino にて受け取り、それを I2C 通信にて Raspberry Pi へ送る。また、別の Arduino にて受け取ったオプティカルフローセンサの値を UART 通信にて Raspberry Pi へ送ることで各種センサの全ての値を適切に処理することを考えた。しかし、実際にこの構成にて値を取得しようと試みた所、UWB の値を受け取った Arduino との I2C 通信が不安定になり、安定してデータを受け取ることが不可能であった。具体的には、Raspberry Pi 上のコンソールにて Input/Output エラーという通信エラーを吐き、通信が途中で止まってしまうというものであった。通信エラーの原因として考えられるのは電気ノイズが通信線に乗ってしまったというハード的な問題とプログラムにおけるボーレートの設定や delay などの遅延が適切に管理できておらず、値を受け取る前に次の値を要求してエラーが起きるソフト的な問題などが考えられる。各種対策を試してみたが、根本的な解決には至らなかったため、I2C 通信による通信は諦め、UWB 通信を 2 つ用いる方法を新たに考案した。そのシステムを Fig. 2 に示す。

以前の構成と同様にセンサ値取得用に 2 つのマイコンを使用していることは変わらないが、1 つのマイコンを Teensy という別のマイコンに換装した。このマイコンは今まで用いてきた Arduino とは異なり、同時に

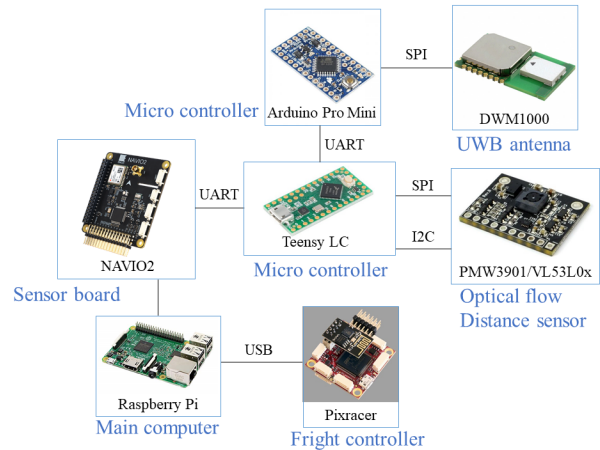


Fig.2: 改良したシステム構成

2 つの UART 通信を処理することが可能である。したがって、以前までは Raspberry Pi の二種類の通信方式 (I2C + UART) を利用し、別々のマイコンを処理していたが、改良後は Raspberry Pi の通信は UART 通信の 1 つに絞り、Teensy にて集めた値を処理した後、まとめて Raspberry Pi に送る方式に変更した。この構成にて各種センサの値をきちんと収集可能かテストした所、通信が途中で途切れることなくデータを得ることが可能であった。

2. 今後の予定

- ・ UWB+オプティカルフローセンサを用いた位置の推定、制御