

BAB IV KESIMPULAN

4.1. Kesimpulan

Berdasarkan hasil pengujian dan analisis terhadap sistem akuisisi data sensor ICM20948 menggunakan SPI berbasis DMA pada lingkungan FreeRTOS, dapat ditarik beberapa kesimpulan sebagai berikut:

- a) Keberhasilan Akuisisi Data Secara Non-Blocking Pemanfaatan SPI DMA serta pemisahan modul menjadi task independent memungkinkan sistem melakukan akuisisi data tanpa mengganggu tugas task lain. Mekanisme inisialisasi *interrupt* tetap responsif dan tidak mengalami kegagalan selama proses pengujian, sehingga efisiensi CPU meningkat.
- b) Performa Sistem Memenuhi Target Kuantitatif Hasil pengujian menunjukkan waktu akuisisi sebesar 65 μ s untuk satu siklus sensor, yang secara signifikan berada di bawah target waktu yang ditentukan 1.060 ms. Hal ini membuktikan bahwa penggunaan DMA dan efisiensi *interrupt* hardware efektif dalam meningkatkan kecepatan transfer data.
- c) Penyederhanaan Kompleksitas melalui Arsitektur Modular Pemisahan modul menjadi SPI MASTER dan SENSOR REGISTER yang berkomunikasi melalui perantara FreeRTOS berhasil mengabstraksikan kompleksitas SPI DMA menjadi fungsi HAL yang lebih sederhana dan mudah dipahami untuk pengembangan selanjutnya.
- d) Kestabilan Sistem di Bawah Beban Kerja Pengujian kontinu selama 4 jam menunjukkan kestabilan operasional tanpa adanya kegagalan transmisi maupun gangguan sistem, meskipun diberikan beban kerja. Hal ini menunjukkan bahwa rancangan sistem cukup andal untuk kebutuhan operasional pada wahana terbang seperti drone.

Secara keseluruhan, penelitian ini telah mencapai tujuannya dalam menghasilkan sistem akuisisi data berbasis SPI DMA yang cepat, stabil, non-blocking, serta memiliki antarmuka HAL yang sederhana di lingkungan RTOS.

4.2.Saran

Untuk pengembangan lebih lanjut, disarankan untuk mengatasi keterbatasan pada jumlah sensor yang dapat di integrasikan dengan menambah fungsi yang melakukan manajemen pada modul sensor sehingga komunikasi dapat bergantian antar modul. Selain itu sistem SPI MASTER masih di efisiensi untuk satu modul sensor sehingga jika ingin menambahkan lebih dari satu sensor wajib melakukan konfigurasi ulang pada beberapa fungsi seperti `xQueue`, ukuran buffer DMA, ukuran besar memory pada `task` RTOS, dan `delay`.

Penggunaan sensor tanpa merek atau sensor *refurbish* wajib memperhatikan kondisi kapasitor, jika kapasitor mengalami kelebihan beban atau rusak, hal tersebut dapat mengakibatkan kerusakan pada modul sensor, dan menyebabkan bit *flip* atau data yang di kirim akan terbalik.

ESP32S3 GPIO *matrix* sering tidak menerima *interrupt* yang di berikan oleh sensor sehingga sistem akan terlihat gagal padahal hal tersebut di akibatkan GPIO yang gagal menerima sinyal, untuk mengatasi hal tersebut dapat menggunakan *millis timeout* sehingga jika tidak ada *interrupt* selama waktu yang di tentukan maka akan tetap terjadi proses akuisisi dan juga inisialisasi *interrupt* ulang.