Comparative Study of Flight Controllers Based on ATmega and ARM STM32 Architectures

Objective

The goal of this project is to conduct a comprehensive review and analysis of flight controllers based on ATmega and ARM STM32 architectures. Students will examine the hardware and embedded software of selected flight controllers, such as the Lynxmotion Quadrino and SpeedyBee F405, and explore the feasibility of programming these controllers using Arduino IDE and Keil software.

Key Components

- ATmega-based Controllers: example Lynxmotion Quadrino (ATmega2560), MultiWii SE (ATmega328P). Known for compatibility with Arduino IDE and ease of programming.
- ARM STM32-based Controllers: example: SpeedyBee F405 (STM32F405), Matek F722-SE (STM32F722) and Holybro Kakute F7 (STM32F745).. High-performance controllers with advanced features and support for complex firmware like Betaflight.

Methodology

1. Literature review: review existing flight controllers based on ATmega and ARM STM32 architectures. Select one ATmega-based and one ARM STM32-based flight controllers for in-depth study.

2. Hardware Analysis: analyze hardware components, including:
   - Microcontroller specifications (CPU, memory, peripherals).
   - Sensor integration (IMU, barometer, GPS).
   - Communication interfaces (UART, I2C, SPI).
   - Document hardware differences between ATmega and STM32-based flight controllers.

3. Embedded Software Study: study pre-installed firmware on selected flight controllers:
   - For ATmega: Explore MultiWii firmware.
   - For STM32: Study Betaflight or INAV firmware.

4. Programming Exploration:

   - ATmega Controllers: use Arduino IDE to create and upload basic flight control programs.

- Explore programming using Keil and STM32Cube to port and modify example firmware.

Deliverables

1. Research Report: comprehensive documentation of hardware and software analysis. Comparative study of ATmega vs. STM32 flight controllers.

2. Code Examples: custom flight control firmware developed using Arduino IDE and Keil. Example code for sensor integration and motor control.

3. Prototyping: basic functional flight tested on selected controllers.