

UNIVERSIDADE FEDERAL DE VIÇOSA  
PÓS-GRADUAÇÃO *LATO SENSU*  
AUTOMAÇÃO E CONTROLE DE PROCESSOS AGRÍCOLAS E INDUSTRIAIS

DANIEL EUGENIO MOREIRA DA COSTA

**Título:** Proposta de Arquitetura para um Sistema de Controle de Voo e Software Embarcado  
de um Veículo Elétrico de Decolagem e Pouso Vertical

Trabalho de Conclusão do Curso de Pós-Graduação  
em Automação e Controle de Processos Agrícolas e  
Industriais do Centro de Pós-Graduação *Lato Sensu*  
da Universidade Federal de Viçosa  
Orientador: Prof. Daniel Khéde Dourado, MSc

Viçosa - MG

2021

## RESUMO

Este trabalho detalha uma proposta de implementação para uma arquitetura de controle de voo e software embarcado aplicada a um veículo elétrico de decolagem e pouso vertical. A implementação é feita em alto nível, e estão descritos quais são os componentes do sistema de controle de voo necessários e como eles se relacionam em termos de interfaces e transmissão de dados. Neste trabalho, também estão detalhadas quais são as funcionalidades que o software embarcado desse sistema deve implementar para que o veículo consiga ser controlado de forma adequada, além de mencionar como o software deve ser embarcado no computador de controle de voo. Assim, o trabalho apresenta, de forma genérica e simplificada, uma solução viável para um dos mais importantes sistemas desse tipo de veículo.

**Palavras-chave:** Sistema de controle de voo. Software embarcado. Lógicas de controle. Arquitetura. Veículo. Computador. Sensores. Atuadores.

## **LISTA DE FIGURAS**

Figura 1: Arquitetura do Sistema de Controle de Voo .....	12
Figura 2: Arquitetura do Software Embarcado .....	16

## **LISTA DE ABREVIATURAS E SIGLAS**

ANAC – Agência Nacional de Aviação Civil  
EASA – European Union Aviation Safety Agency  
eVTOL – Electric Vertical Take-off and Landing Vehicle  
FAA – Federal Aviation Administration  
FCC –Flight Control Computer  
GNSS – Global Navigation Satellite Systems  
GPS – Global Positioning System  
IMU – Inertial Measurement Unit  
MC – Motor Controller  
RADALT – Radio Altimeter  
UAV – Unmanned Aerial Vehicle  
WOW – Weight on Wheels

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO .....</b>	<b>5</b>
<b>2</b>	<b>DESENVOLVIMENTO .....</b>	<b>7</b>
2.1	RESUMO DAS ETAPAS DE DESENVOLVIMENTO .....	7
2.2	SENSORES DO SISTEMA DE CONTROLE DE VOO .....	8
2.3	ATUADORES DO SISTEMA DE CONTROLE DE VOO .....	9
2.4	COMPUTADOR DE CONTROLE DE VOO (FCC).....	9
2.5	ARQUITETURA PARA O SISTEMA DE CONTROLE DE VOO.....	10
2.6	ARQUITETURA PARA O SOFTWARE EMBARCADO.....	13
<b>3</b>	<b>RESULTADOS E DISCUSSÕES.....</b>	<b>16</b>
<b>4</b>	<b>CONCLUSÃO.....</b>	<b>17</b>
<b>5</b>	<b>REFERÊNCIAS.....</b>	<b>18</b>

## 1 INTRODUÇÃO

O contexto atual de mobilidade, principalmente dos grandes centros urbanos, faz com que o mercado se sensibilize por buscar novas soluções para proporcionar à população uma mobilidade cada vez mais eficiente. Nesse intuito, os veículos elétricos de decolagem e pouso vertical (*eVTOLs*) estão prestes a se tornar uma realidade e uma opção a mais para que passageiros possam se locomover a distâncias médias com um tempo menor, e sem precisar enfrentar o trânsito caótico das cidades [MundoGEO, 2021].

Os projetos de concepção e desenvolvimento de *eVTOLs* estão se disseminando ao redor do mundo, com várias empresas já consolidadas e, também, startups anunciando os seus projetos e protótipos. Empresas como *Embraer* [Eve, 2021], *Airbus* [CityAirbus, 2021], *Lilium* [Lilium, 2021], *Volocopter* [Volocopter, 2021], entre várias outras já têm os seus projetos anunciados e vendas já consolidadas no mercado para esse tipo de veículo, mesmo antes dos veículos estarem prontos para serem entregues [CNNBrasil, 2021]. Além disso, também existe a necessidade de se idealizar um ambiente de mobilidade aérea com regras para que esses veículos consigam ser operados com segurança [MundoGEO, 2021].

Além do desenvolvimento técnico dos *eVTOLs*, as empresas também devem conseguir certificar esses veículos junto às entidades certificadoras de aviação civil, como *ANAC*, *FAA* e *EASA*. Nesse sentido, as empresas que já desenvolvem aeronaves e que já têm experiência e expertise na certificação delas, podem ter vantagem competitiva nesse quesito, conseguindo operacionalizar seus projetos de *eVTOL* mais rapidamente. Mesmo que essas empresas encontrem dificuldades técnicas de projeto no curto prazo, elas podem conseguir colocar esses veículos em operação mais rapidamente do que outras empresas que não são dessa área de atuação.

Esses veículos estão sendo construídos do zero, sem um histórico de informações técnicas que possam ser reutilizadas. Então, sobre as várias dificuldades técnicas que os projetos de *eVTOL* podem enfrentar, existem algumas áreas de estudo que devem ser levadas em consideração nesses projetos. Uma das áreas mais importantes é a que trata dos sistemas de controle de voo. O sistema de controle de voo é responsável por controlar o funcionamento do veículo durante a operação de voo, de forma que esse sistema recebe as informações de sensores do veículo, processa essas informações e envia os comandos para acionar os motores de propulsão. Lógicas de controle (ou algoritmos de controle) são necessárias para deixar o veículo estável durante o voo, e essas lógicas de controle são implementadas por um software embarcado no computador de controle de voo (*FCC*). No desenvolvimento do sistema de

controle de voo, empresas de aeronaves podem levar vantagem na concepção dessa tecnologia, uma vez que um sistema semelhante é utilizado em aviões.

Analisando os principais aspectos para o desenvolvimento de um sistema de controle de voo para eVTOLs, este trabalho tem como objetivo a concepção de uma arquitetura em alto nível e genérica que descreva de forma simplificada todos os componentes necessários para implementar esse sistema, bem como as interfaces necessárias entre esses componentes. Mais ainda, é também objetivo deste trabalho detalhar a arquitetura do software embarcado (FCC), bem como suas funcionalidades e interações com o restante do sistema.

## 2 DESENVOLVIMENTO

### 2.1 RESUMO DAS ETAPAS DE DESENVOLVIMENTO

Como o objetivo desse trabalho é definir uma arquitetura básica de um sistema de controle de voo para um *eVTOL*, primeiramente é necessário descrever os componentes que compõem esse sistema. Assim, é necessário definir os sensores que vão ser utilizados para reunir as informações de grandezas físicas necessárias para controlar o veículo. Também é necessário definir os atuadores que vão exercer os comandos para que o veículo consiga se movimentar. O computador embarcado (*FCC*) é o responsável por receber todos os sinais dos sensores, processá-los e enviar os comandos corretos para os atuadores. Além disso, é necessário apresentar as arquiteturas básicas que vão constituir a interligação entre esses componentes e implementar o sistema de controle de voo.

Além das definições relacionadas ao sistema, para que seja possível ter uma visão completa de todos os aspectos necessários para a construção de um *eVTOL*, também é necessário mencionar alguns outros componentes que são muito importantes e estão interligados a esse sistema, como as baterias e o corpo do veículo (*frame*). As baterias são uns dos componentes mais importantes desse veículo elétrico, pois é responsável por garantir toda alimentação de energia dos componentes do sistema, bem como de todos os outros sistemas do veículo. As baterias também devem compostas por um sistema de recarga eficiente. O *frame* é o corpo ou a estrutura do veículo, que deve ser capaz de comportar todos os componentes dos sistemas, além do *cockpit* do piloto e o espaço dos passageiros.

No contexto desse trabalho, podemos definir que a construção desse *eVTOL* é limitada para que somente o piloto possa controlar esse veículo internamente. Uma outra possibilidade seria fazer com que o veículo pudesse ser controlado também remotamente. Nesse caso o sistema de controle de voo também seria composto por, como exemplo, uma estação de controle de solo e um sistema de rádio controle. Uma outra possibilidade seria a construção de um veículo autônomo, neste caso, todas as ações de comando seriam implantadas diretamente no software embarcado do sistema de controle de voo.

Nas seções seguintes, iremos focar nas descrições de tudo que faz parte da implementação de uma arquitetura básica para o sistema e software de controle de voo de um *eVTOL*, restrito para ser controlado por um piloto no *cockpit* do veículo.

## 2.2 SENSORES DO SISTEMA DE CONTROLE DE VOO

Os sensores necessários para construir um *eVTOL* são similares aos sensores também utilizados em drones ou *UAVs* (veículos autônomos não tripulados, chamados também de *VANTs*). Além destes, são utilizados sensores adicionais para que o veículo possa ser controlado por um piloto no *cockpit* da aeronave. Focando nas funcionalidades básicas para o funcionamento de um *eVTOL*, e deixando de lado as considerações específicas que cada fabricante possa ter, os sensores que estão no escopo desse trabalho estão listados abaixo com seus respectivos objetivos para o funcionamento do veículo.

Note que um único tipo de sensor pode ser construído com diferentes princípios de funcionamento e características construtivas. Essa especificação também é decisão de cada projeto.

- *GPS*: sensor de posicionamento que fornece a posição geográfica do veículo por meio de um sistema de navegação por satélite. Esse dispositivo pode utilizar dois sistemas *GNSS* para obter maior exatidão na localização, a rede russa *GLONASS* e a rede norte-americana *GPS*.
- *IMU*: sistema inercial de navegação que integra as acelerações em Norte/Sul, Leste/Oeste para calcular a posição e o deslocamento do veículo. Esse sensor é necessário uma vez que o *GPS* possui uma grande margem de erro. Além disso, o *IMU* não necessita de informações de fora do veículo, como recepção e emissão de sinais, não correndo o risco de sofrer interferência ou oclusão.
- *RADALT*: um tipo de radar que calcula a altitude, ou altura, do veículo por meio da medição do tempo decorrido entre emissão de uma onda de rádio e a recepção desta após ela refletir no solo.
- Magnetômetro: sensor que funciona como uma bússola digital e que auxilia na navegação e orientação do veículo. É parte integrante do *IMU*.
- Acelerômetro e Giroscópio: sensores utilizados para medição da aceleração e inclinação do veículo. É parte integrante do *IMU*.
- *WOW*: sensor que indica se o veículo se encontra em solo ou em voo. Geralmente são usados 4 sensores *WOW* para essa indicação, e com pelo menos 3 desses sensores setados, pode-se considerar que o veículo está em solo.
- *Sidestick*: dispositivo utilizado para que o piloto realize comandos para controlar a aeronave. No caso do *eVTOL*, os movimentos que o piloto realiza no *sidestick*

geram comandos elétricos que são enviados para o computador embarcado (*FCC*), que por sua vez envia comandos para acionar os motores elétricos. No *cockpit* do piloto, também podem existir outros sensores, como botões para ligar/desligar o veículo, entre outras funções mais simples.

### 2.3 ATUADORES DO SISTEMA DE CONTROLE DE VOO

Os atuadores necessários para se construir um *eVTOL* também são similares aos utilizados em um drone ou *UAV*. Basicamente, o sistema de controle de voo vai atuar nos controladores dos motores elétricos, que são responsáveis por realizar a movimentação do veículo, tanto no sentido vertical como no sentido horizontal. Portanto, esses controladores dos motores, também conhecidos como *motor controllers (MC)*, são os atuadores desse sistema. Os *MCs* vão receber os sinais elétricos provenientes do *FCC*, que correspondem a um valor de *RPM* (rotações por minuto), e vão fazer com que os motores elétricos gerem essa rotação desejada. Por sua vez, os motores elétricos vão girar as pás conectadas a eles para movimentar o ar e gerar a propulsão necessária para a nave.

Cada projeto de *eVTOL* pode ter uma quantidade diferente de motores. Essa quantidade vai depender principalmente do projeto aerodinâmico do veículo. Então, em um projeto mais simples podem ter 4 motores no sentido vertical e em um outro projeto mais complexo podem ter 10 motores no sentido vertical e mais 2 motores no sentido horizontal, por exemplo. Um *eVTOL* não necessita de motores na horizontal para se movimentar no sentido horizontal, pois este pode obter propulsão horizontal ao se inclinar de forma semelhante aos drones de pequeno porte. Note que essa inclinação pode ser obtida ao rotacionar as pás dos motores verticais com diferentes velocidades. Outros tipos de projeto também podem considerar motores instalados juntos a eixos móveis, que, a depender do tipo de movimento que se queria realizar, esses eixos móveis podem mudar a direção dos motores. Neste trabalho, em termos de simplificação, vamos utilizar um veículo com apenas 4 motores elétricos de propulsão vertical.

### 2.4 COMPUTADOR DE CONTROLE DE VOO (FCC)

O *FCC* é o principal componente de um sistema de controle de voo de uma aeronave, e estes utilizam uma tecnologia chamada *Fly-by-wire* (Voo por Fio). Essa tecnologia também pode ser empregada em *eVTOLs*, e, nesse contexto, o *FCC* fica responsável por receber todas

as informações elétricas de sensores e comandos do piloto, processar essas informações por meio do software embarcado, implementando as lógicas de controle do veículo, e, por fim, gerando os comandos elétricos para serem enviados para os *MCs*.

Portanto, o *FCC* precisa ter uma interface para conseguir se comunicar com todos os outros componentes do sistema, como entradas e saídas digitais e analógicas e, também, protocolos de comunicação. Além disso, o *FCC* precisa ter uma arquitetura computacional de processamento capaz de rodar o software embarcado que vai conter o processamento dessa interface e a lógica de controle que vai controlar o veículo.

Essa arquitetura computacional, como outros tipos de computadores, deve ser composta por processadores, memórias, barramentos, hardware de baixo nível, sistema operacional de tempo real, entre outros. Em geral, o *FCC* é projetado especificamente para as necessidades práticas de voo de um *eVTOL*, pois cada projeto necessita de requisitos de processamento e de interface bastante específicos.

## 2.5 ARQUITETURA PARA O SISTEMA DE CONTROLE DE VOO

A arquitetura genérica deve conter todos os componentes essenciais para que o sistema de controle de voo seja funcional, ou seja, para que ele tenha as funcionalidades básicas que fazem com que o *eVTOL* consiga realizar sua função principal de voo de forma precisa e segura. Nesta seção, serão apresentados todos os componentes necessários e os seus respectivos papéis, bem como eles se comunicam entre si.

Primeiramente, para que todos os sensores do sistema consigam enviar seus respectivos dados elétricos de medições ou indicações para o *FCC*, é necessário que exista uma interface de comunicação entre esses sensores e o computador. Como estamos trabalhando com um sistema do setor aeronáutico, é essencial que tenhamos uma comunicação segura, e que já tenha sido testada e validada pelo mercado.

Nesse contexto, seria adequado utilizar o protocolo de comunicação *ARINC 429* para realizar essa transmissão de dados dos sensores para o *FCC*. O *ARINC 429* já é um dos padrões aeronáuticos de transferência de dados mais utilizados nas aeronaves, e como na aplicação de transmissão de dados sensorais não é necessária uma taxa de atualização muito alta, podemos utilizar esse protocolo, que conta com uma taxa máxima de transmissão de 100 Kbps (100.000 bits por segundo).

No caso das saídas do *FCC*, para que ele consiga enviar os comandos elétricos necessários para acionar os *MCs*, também é necessário o uso de uma interface de comunicação. Nesse caso, essa comunicação necessita de uma taxa de atualização bastante alta, visto que os motores devem ter uma resposta imediata para conseguir fazer com que o veículo seja controlado no voo de forma estável. Considerando esses requisitos, o protocolo de comunicação *CAN Bus* se mostra como uma opção mais aceitável para essa aplicação, já que esse protocolo consegue entregar uma taxa de transmissão de dados de até 1 Mbps (1.000.000 bits por segundo). Considerando que estamos tratando de um sistema aeronáutico, seria adequado que a implementação do *CAN Bus* seguisse a norma aeronáutica *ARINC 825*, que estabelece boas práticas para tornar esse protocolo mais seguro e confiável.

Como a lógica de controle do sistema de voo deve ser implementada em malha fechada, o *FCC* necessita receber o feedback do comando gerado pelos *MCs*, *i.e.*, a velocidade de rotação dos motores, para controlar esses motores elétricos de forma adequada.

Em resumo, a interface do *FCC* vai conter os seguintes protocolos:

- *ARINC 492 Rx* (entrada do *FCC*): para receber informações dos sensores.
- *CAN Bus Rx* (entrada do *FCC*): para receber o feedback de velocidade dos motores.
- *CAN Bus Tx* (saída do *FCC*): para enviar comandos aos controladores dos motores.

Em um projeto de *FCC* é sempre importante que ele tenha uma interface de entradas e saída analógicas e digitais, pois sempre vai ser necessário que algum componente específico se comunique dessa forma. Nesse caso, estamos considerando que, antes de chegar ao *FCC*, vai existir um dispositivo de conversão de entradas analógicas e digitais para *ARINC 429*, de forma a converter todos os sinais dos sensores para esse protocolo. Em alguns casos, o próprio sensor já realiza a conversão. De forma análoga, os motores elétricos também devem possuir interface de *CAN Bus*.

É importante mencionar também que todos esses componentes do sistema de controle de voo têm que ser alimentados por uma fonte de tensão. Os sensores e o *FCC* podem ser alimentados por uma fonte normal de baixa tensão, mas os componentes dos motores elétricos devem ser alimentados por uma fonte de alta tensão separadamente. Todo o sistema, como também, o restante dos sistemas do veículo, devem ser alimentados por um conjunto de baterias bastante robusto para que o veículo consiga funcionar pelo maior tempo possível.

A Figura 1 apresenta um esquemático de toda essa arquitetura descrita anteriormente para o sistema de controle de voo hipotético de um *eVTOL*:

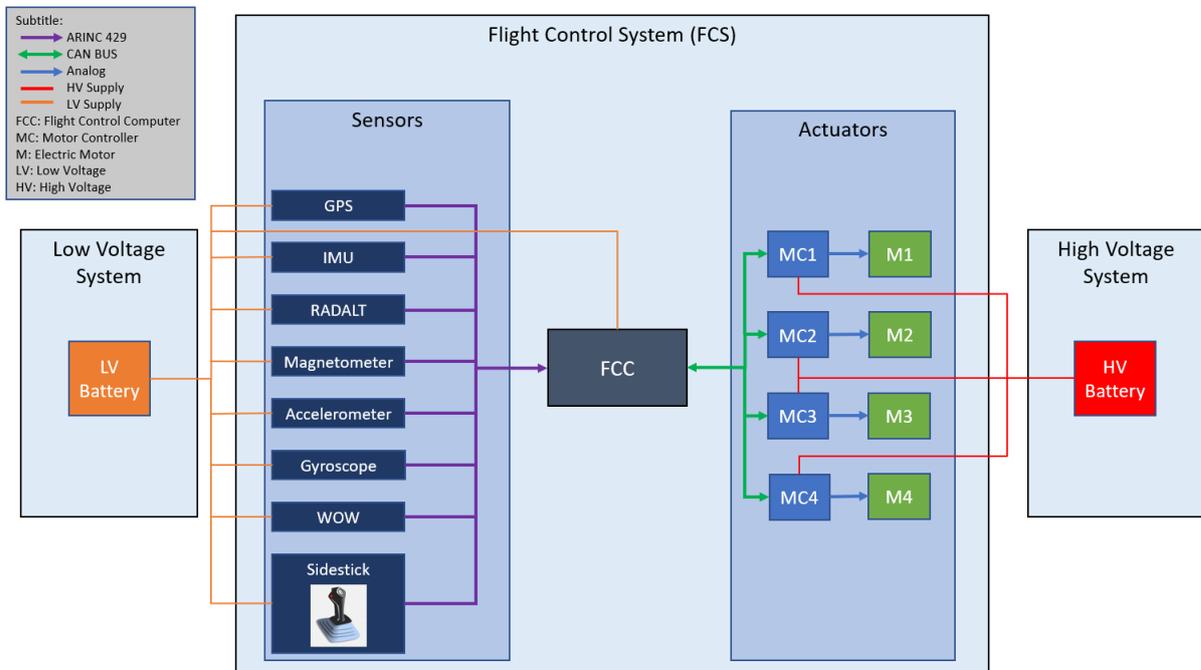


Figura 1: Arquitetura do Sistema de Controle de Voo

Apesar do sistema descrito na arquitetura apresentada ser capaz de implementar todas as funcionalidades necessárias para controlar o veículo, ainda existe a necessidade de que esse sistema tenha a maior confiabilidade possível. No intuito de atribuir uma maior segurança e, também, para atender aos critérios que as agências reguladoras exigem para que o veículo seja certificado, existe a necessidade de implementar uma arquitetura com redundância.

A seguir, vamos descrever quais são os principais pontos que devem ser destacados para se implementar um sistema de controle de voo com redundância a partir da arquitetura da Figura 1. Note que as decisões podem seguir vários critérios, não existindo apenas um tipo de arquitetura com redundância. O importante é que a arquitetura com redundância apresente níveis de confiabilidade aceitáveis.

- Três sensores de cada tipo: colocando três sensores medindo a mesma grandeza física e instalando eles em pontos diferentes do veículo, podemos garantir com maior segurança que sempre teremos aquela medição, e se perdermos um dos sensores, a medição ainda continua sendo realizada pelos outros dois sensores. A partir dos três valores medidos é possível fazer uma lógica de votação (ou de

filtragem) para obter o valor final, como, por exemplo, fazer a média entre esses três valores ou selecionar o valor intermediário.

- Três computadores de controle de voo: uma arquitetura com três *FCCs* terá um fator de confiabilidade maior. Dessa forma, cada sensor deve enviar suas medições para *FCCs* diferentes. Para isso, deve existir um protocolo de comunicação, como o *ARINC 429*, entre os três *FCCs* para que eles compartilhem esses valores, e cada um obtenha todos os três valores dos sensores para realizar a votação. Além disso, todos os *FCCs* devem ser capazes de enviar comandos para os motores, mas apenas um (1) *FCC* estará ativo em um determinado momento, e se um computador for perdido, um outro *FCC* entra em ação para enviar os comandos aos motores. Por óbvio, todos os *FCCs* executam o mesmo software.
- Duas unidades de processamento em cada computador: dentro do *FCC* existem duas unidades de processamento diferentes, isto é, cada uma com um processador com arquitetura diferente. Então, em cada unidade de processamento é embarcado o mesmo software, que roda a mesma lógica de controle. Uma dessas unidades é chamada de *command*, que é a unidade que realmente comanda os sinais de saída do *FCC*. A outra unidade é chamada de *monitor*, que é uma unidade de monitoramento, verificando se o que foi calculado pelo software da unidade *command* também foi calculado pela unidade *monitor*. Essas duas unidades dentro do *FCC* também podem fazer uma sincronização de execução entre elas.
- Dois controladores dos motores para cada motor: para completar a arquitetura de redundância, podemos utilizar dois *MCs* para cada motor elétrico, um ativo e um inativo. No caso de o *MC* ativo ser perdido, imediatamente o outro *MC* que estava inativo entra em ação. Dessa forma, a probabilidade de que o veículo perca um motor elétrico de propulsão durante a operação diminui consideravelmente.

Em resumo, podemos considerar que essa arquitetura de redundância do sistema de controle de voo apresenta redundância em termos de componentes físicos de sensoriamento e atuação, em termos de computadores e, finalmente, em termos de processamento; visto que o software executa os comandos em processadores diferentes dentro de um mesmo *FCC*.

## 2.6 ARQUITETURA PARA O SOFTWARE EMBARCADO

O software do sistema de controle de voo, que é embarcado no *FCC*, pode ser dividido em basicamente duas partes: o software de aplicação e o software de plataforma. O software de aplicação é o software que vai implementar as funcionalidades do sistema, como as lógicas de controle, a implementação de decodificação e empacotamento dos valores de entrada e saída, a lógica de votação dos sensores, a lógica de cálculo dos comandos para os atuadores, entre outras funcionalidades que o sistema de controle de voo possa exigir. O software de plataforma, complementarmente, faz a função de um tipo de sistema operacional de tempo real de mais baixo nível, que vai realizar o gerenciamento das tarefas do software de aplicação com uma determinada ordem e frequência de execução, implementar os protocolos de comunicação, fazer a interface com o hardware, entre outras questões similares a um sistema operacional.

É importante mencionar a principal funcionalidade do software de aplicação, que são as lógicas de controle. Essas lógicas de controle são implementadas no nível de sistemas em forma de requisitos de sistemas e são entregues para o nível de software para que elas sejam codificadas e embarcadas no *FCC*. As lógicas de controle operam como a inteligência do sistema de controle de voo, pois são elas que contêm as modelagens dos controladores que vão controlar o veículo em todas as fases de voo. As lógicas de controle têm como entradas os valores dos sensores e o feedback dos comandos de rpm enviados aos motores elétricos; e têm como saída os comandos de corrente ou tensão que são enviados os controladores dos motores.

As lógicas de controle geralmente são implementadas no nível de sistemas através de modelos, como por exemplo, modelos *Simulink*, que é uma toolbox do *Matlab*. Nesse caso, é possível utilizar a geração automática de código em diversas linguagens a partir desses modelos. No caso de modelos *Simulink*, pode-se utilizar uma outra toolbox do *Matlab* chamada *Embedded Coder*, que gera o código na linguagem C automaticamente a partir dos modelos. Dessa forma, não seria necessário codificar manualmente todas as funcionalidades da lógica de controle existentes, o que reduz a quantidade de trabalho no projeto do veículo.

Além das lógicas de controle, o software de aplicação deve realizar a decodificação das entradas que chegam no *FCC* e o empacotamento das saídas que são produzidas pelo software e são enviadas do *FCC* para os *MCs*. O método de decodificação adotado irá depender do protocolo pelo qual o valor for recebido, *ARINC 429* no caso dos sensores e *CAN Bus* no caso do feedback dos motores. Para o empacotamento, os valores devem ser codificados para serem enviados pelo *CAN Bus* até os *MCs*.

Sobre as funcionalidades do software de plataforma, a lógica de gerenciamento (*scheduler*) é responsável por gerenciar todas as tarefas do software de aplicação. O *scheduler*

vai executar as unidades de software da aplicação em uma ordem preestabelecida, com cada unidade podendo ser executada em uma determinada frequência. Por exemplo, o *scheduler* é capaz de alocar tarefas vindas de um *FCC* rodando com frequência de execução de 100Hz, ou seja, 10 ms, e tarefas de aplicação sendo executadas a 10 Hz.

No papel de sistema operacional, o software de plataforma também vai implementar os protocolos de comunicação utilizados no controle de voo, que neste trabalho são o *ARINC 429* e o *CAN Bus*. Ao implementar os protocolos, é também definida a estrutura de bits que vai compor cada campo do protocolo, a forma de envio e recebimento dos dados, tratamento de erros, entre outros pontos que são característicos a esses protocolos.

A comunicação do software de plataforma com o hardware realiza os acessos necessários a memórias, processadores, barramentos de comunicação, entre outras funcionalidades de baixo nível.

Para garantir a confiabilidade exigida, os controladores de voo para *eVTOLs* são geralmente desenvolvidos na linguagem de programação C. O motivo dessa escolha se deve ao fato da linguagem C ser bastante disseminada e já ter sido bastante testada nessa área de aplicação. Outra característica favorável é que a linguagem C atua em baixo nível, o que é adequado para esse tipo de aplicação embarcada, que trabalha com sistemas de instrumentação e controle de baixo nível.

### 3 RESULTADOS E DISCUSSÕES

Estão descritas neste trabalho, de forma genérica e simplificada, algumas das principais funcionalidades que o software embarcado de um sistema de controle de voo tem que apresentar para conseguir realizar de forma adequada o controle de *eVTOLs*. A depender do projeto, é natural que funcionalidades adicionais sejam necessárias para suprir as necessidades específicas destes projetos.

Como exemplo de aplicação, a Figura 2 apresenta a arquitetura que foi detalhada na Seção 2, contendo um software embarcado para o sistema de controle de voo.

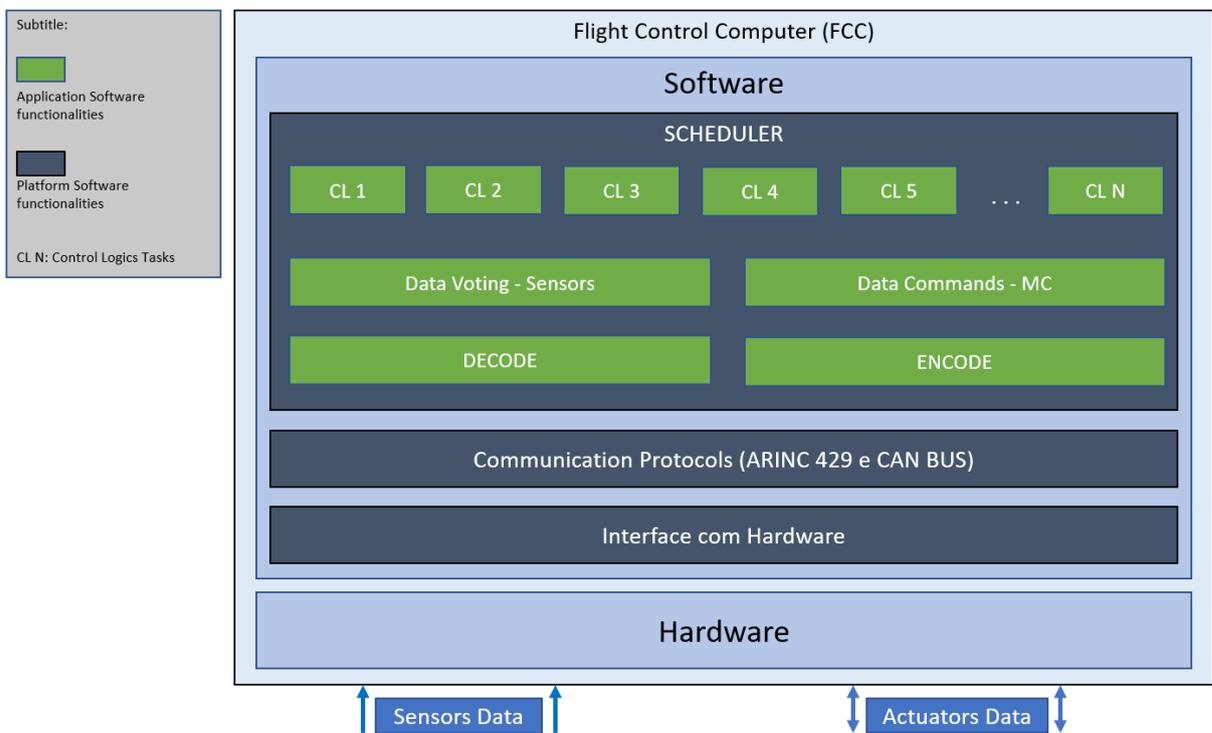


Figura 2: Arquitetura do Software Embarcado

As soluções propostas neste trabalho podem servir como ponto de partida para decisões de quais caminhos devem ser seguidos no intuito de se projetar um sistema de controle de voo e software embarcado aplicado à *eVTOLs*. Além disso, este trabalho apresenta algumas lições que já foram aprendidas com base em projetos anteriores de aeronaves, e que podem ser aproveitadas, uma vez que geram benefícios similares no contexto de *eVTOLs*.

## 4 CONCLUSÃO

Este trabalho cumpriu o objetivo de descrever como seria uma arquitetura adequada para o sistema de controle de voo e software embarcado de um *eVTOL*.

Um importante complemento a este trabalho seria desenvolver e apresentar a lei de controle que está presente no algoritmo de controle em malha fechada, responsável por efetivamente garantir a estabilidade do *eVTOL* durante as tarefas de voo.

Outra sugestão de trabalho futuro seria o detalhamento da certificação de um veículo com esse junto às agências reguladoras nacionais. Este trabalho complementar teria como foco as normas aeronáuticas para sistemas (*ARP4754*) e para software (*DO-178C*), além das cabíveis regras de tráfego para operação desse tipo de veículo aéreo que ainda não foram definidas.

## 5 REFERÊNCIAS

[MundoGEO, 2021] *Mobilidade Aérea para drones e eVTOLs: o futuro está próximo de se tornar realidade no Brasil* (<https://mundogeo.com/2021/08/24/mobilidade-aerea-para-drones-e-evtols-o-futuro-esta-proximo-de-se-tornar-realidade-no-brasil/>)

[Eve, 2021] *Eve Air Mobility* (<https://eveairmobility.com/>)

[CityAirbus, 2021] *CityAirbus NextGen* (<https://www.airbus.com/en/innovation/zero-emission/urban-air-mobility/cityairbus-nextgen>)

[Lilium, 2021] *Lilium* (<https://lilium.com/>)

[Volocopter, 2021] *Volocopter* (<https://www.volocopter.com/>)

[CNNBrasil, 2021] *Corrida dos eVTOLs: conheça os principais projetos de ‘carros voadores’* (<https://www.cnnbrasil.com.br/business/corrida-dos-evtols-conheca-os-principais-projetos-de-carros-voadores/>)