

Платформа за безпилотен летателен апарат с четири ротора

РАФАЕЛ КАЛЪЧЕВ

ТЕХНИЧЕСКИ УНИВЕРСИТЕТ СОФИЯ

25 ФЕВРУАРИ 2022



Увод

Увод

Цели и задачи - Софтуер

Цели и задачи - Софтуер

- Изграждане на среда за разработка на софтуер с основа *Make* под *Linux*

Цели и задачи - Софтуер

- Изграждане на среда за разработка на софтуер с основа *Make* под *Linux*
- Подбор на хардуерни и софтуерни решения за интеграция с разработената среда

- Изграждане на среда за разработка на софтуер с основа *Make* под *Linux*
- Подбор на хардуерни и софтуерни решения за интеграция с разработената среда
- Създаване на софтуерни модули и драйвъри за работа с вътрешния и външен хардуер

- Изграждане на среда за разработка на софтуер с основа *Make* под *Linux*
- Подбор на хардуерни и софтуерни решения за интеграция с разработената среда
- Създаване на софтуерни модули и драйвъри за работа с вътрешния и външен хардуер
- Подбор на изходни портове, пинове и нужна периферия

- Изграждане на среда за разработка на софтуер с основа *Make* под *Linux*
- Подбор на хардуерни и софтуерни решения за интеграция с разработената среда
- Създаване на софтуерни модули и драйвъри за работа с вътрешния и външен хардуер
- Подбор на изходни портове, пинове и нужна периферия
- Инициализация и конфигурация на микроконтролера и периферията

Увод

Цели и задачи - Хардуер

ЦЕЛИ И ЗАДАЧИ - ХАРДУЕР

- Разположение на хардуера и избор на конфигурация за системата

ЦЕЛИ И ЗАДАЧИ - ХАРДУЕР

- Разположение на хардуера и избор на конфигурация за системата
- Подбор на изходни портове, пинове и нужна периферия

Цели и задачи - Хардуер

- Разположение на хардуера и избор на конфигурация за системата
- Подбор на изходни портове, пинове и нужна периферия
- Съвързване на компонентите и електроснабдяване

- Разположение на хардуера и избор на конфигурация за системата
- Подбор на изходни портове, пинове и нужна периферия
- Свързване на компонентите и електроснабдяване
- Изграждане платформи за опитни постановки

- Разположение на хардуера и избор на конфигурация за системата
- Подбор на изходни портове, пинове и нужна периферия
- Свързване на компонентите и електроснабдяване
- Изграждане платформи за опитни постановки
- Сглобяване на системата

Увод

**Цели и задачи - Идентификация,
моделиране, наблюдение и управление**

Цели и задачи - Идентификация, МОДЕЛИРАНЕ, НАБЛЮДЕНИЕ И УПРАВЛЕНИЕ

Цели и задачи - Идентификация, МОДЕЛИРАНЕ, НАБЛЮДЕНИЕ И УПРАВЛЕНИЕ

■ Моделиране:

Цели и задачи - Идентификация, МОДЕЛИРАНЕ, НАБЛЮДЕНИЕ И УПРАВЛЕНИЕ

- Моделиране:
 - ▶ Ротор с витло

Цели и задачи - Идентификация, МОДЕЛИРАНЕ, НАБЛЮДЕНИЕ И УПРАВЛЕНИЕ

■ Моделиране:

- ▶ Ротор с витло
- ▶ Платформа за управление на ъгъл на завъртане

Цели и задачи - Идентификация, МОДЕЛИРАНЕ, НАБЛЮДЕНИЕ И УПРАВЛЕНИЕ

■ Моделиране:

- ▶ Ротор с витло
- ▶ Платформа за управление на ъгъл на завъртане
- ▶ Платформа за безпилотен летателен апарат с четири ротора

Цели и задачи - Идентификация, МОДЕЛИРАНЕ, НАБЛЮДЕНИЕ И УПРАВЛЕНИЕ

■ Моделиране:

- ▶ Ротор с витло
- ▶ Платформа за управление на ъгъл на завъртане
- ▶ Платформа за безпилотен летателен апарат с четири ротора

■ Идентификация на параметри

Цели и задачи - Идентификация, МОДЕЛИРАНЕ, НАБЛЮДЕНИЕ И УПРАВЛЕНИЕ

- Моделиране:
 - ▶ Ротор с витло
 - ▶ Платформа за управление на ъгъл на завъртане
 - ▶ Платформа за безпилотен летателен апарат с четири ротора
- Идентификация на параметри
- Компенсация на сензорни отмествания

Цели и задачи - Идентификация, МОДЕЛИРАНЕ, НАБЛЮДЕНИЕ И УПРАВЛЕНИЕ

- Моделиране:
 - ▶ Ротор с витло
 - ▶ Платформа за управление на ъгъл на завъртане
 - ▶ Платформа за безпилотен летателен апарат с четири ротора
- Идентификация на параметри
- Компенсация на сензорни отмествания
- Калибриране на сензорите

Цели и задачи - Идентификация, МОДЕЛИРАНЕ, НАБЛЮДЕНИЕ И УПРАВЛЕНИЕ

- Моделиране:
 - ▶ Ротор с витло
 - ▶ Платформа за управление на ъгъл на завъртане
 - ▶ Платформа за безпилотен летателен апарат с четири ротора
- Идентификация на параметри
- Компенсация на сензорни отмествания
- Калибриране на сензорите
- Синтез на наблюдател за оценка на ориентацията на платформата

Цели и задачи - Идентификация, МОДЕЛИРАНЕ, НАБЛЮДЕНИЕ И УПРАВЛЕНИЕ

- Моделиране:
 - ▶ Ротор с витло
 - ▶ Платформа за управление на ъгъл на завъртане
 - ▶ Платформа за безпилотен летателен апарат с четири ротора
- Идентификация на параметри
- Компенсация на сензорни отмествания
- Калибриране на сензорите
- Синтез на наблюдател за оценка на ориентацията на платформата
- Синтез на управление

Софтуерна част

Софтуерна част

Среда за разработка на софтуер с основа
Make под *Linux*

Състав на средата за работа под *Linux*

Състав на средата за работа под *Linux*

- Система за насочено изграждане:
GNU Make

Състав на средата за работа под *Linux*

- Система за насочено изграждане:
GNU Make
- **Компилатор:**
GCC ARM NON-EABI

Състав на средата за работа под *Linux*

- Система за насочено изграждане:

GNU Make

- **Компилатор:**

GCC ARM NON-EABI

- **Връзка с контролера:**

ST-LINK

Състав на средата за работа под *Linux*

- Система за насочено изграждане:

GNU Make

- Компилятор:

GCC ARM NON-EABI

- Връзка с контролера:

ST-LINK

- Текстов редактор:

VIM + Ctags

Състав на средата за работа под *Linux*

- Система за насочено изграждане:

GNU Make

- Компилятор:

GCC ARM NON-EABI

- Връзка с контролера:

ST-LINK

- Текстов редактор:

VIM + Ctags

- Дебъгер:

GDB (GNU Project Debugger)

ОПИСАНИЕ И СЪСТАВНИ ЧАСТИ

Състав на средата за работа под *Linux*

- Система за насочено изграждане:

GNU Make

- Компилятор:

GCC ARM NON-EABI

- Връзка с контролера:

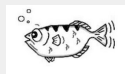
ST-LINK

- Текстов редактор:

VIM + Ctags

- Дебъгер:

GDB (GNU Project Debugger)



Подържани команди

¹В случай, че софтуерът има нужда от (пре)изграждане, системата автоматично го (пре)изгражда.

Подържани команди

make | make all Цялостно изграждане чрез компилиране и свързване на всички *нужни* елементи.

¹В случай, че софтуерът има нужда от (пре)изграждане, системата автоматично го (пре)изгражда.

Подържани команди

make | make all Цялостно изграждане чрез компилиране и свързване на всички *нужни* елементи.

make clean Изчистване на средата.

¹В случай, че софтуерът има нужда от (пре)изграждане, системата автоматично го (пре)изгражда.

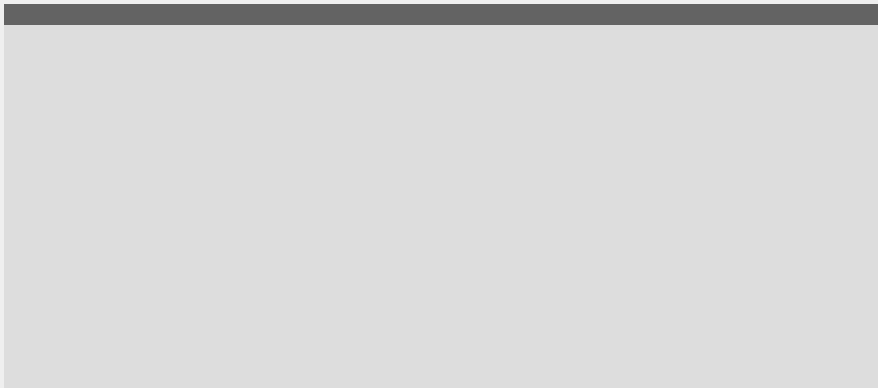
Подържани команди

make | make all Цялостно изграждане чрез компилиране и свързване на всички *нужни* елементи.

make clean Изчистване на средата.

make flash Запис на изградения ¹ софтуер в паметта на микроконтролера.

¹В случай, че софтуерът има нужда от (пре)изграждане, системата автоматично го (пре)изгражда.



²Системата автоматично конфигурира дебъгера да използва дебъг символите в изградения софтуер.

make debug Отваряне на порт за дебъг, стартиране и свързване на дебъгера ²

²Системата автоматично конфигурира дебъгера да използва дебъг символите в изградения софтуер.

make debug Отваряне на порт за дебъг, стартиране и свързване на дебъгера ²

make usart Стартира интерактивен комуникационен интерфейс за връзка (IO) с контролера чрез USART.

²Системата автоматично конфигурира дебъгера да използва дебъг символите в изградения софтуер.

make debug Отваряне на порт за дебъг, стартиране и свързване на дебъгера ²

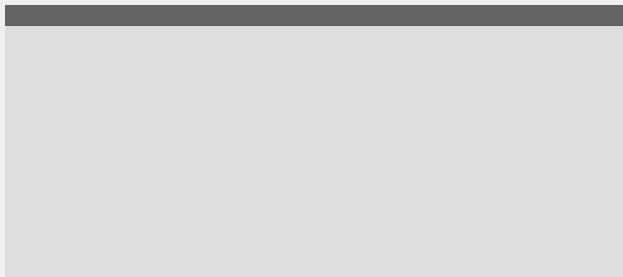
make usart Стартира интерактивен комуникационен интерфейс за връзка (IO) с контролера чрез USART.

make usart_read Създава файл, съдържащ получените данни през USART порта.

²Системата автоматично конфигурира дебъгера да използва дебъг символите в изградения софтуер.

ПОДБРАНИ ХАРДУЕРНИ И СОФТУЕРНИ РЕШЕНИЯ ЗА АНАЛИЗ

ПОДБРАНИ ХАРДУЕРНИ И СОФТУЕРНИ РЕШЕНИЯ ЗА АНАЛИЗ



ПОДБРАНИ ХАРДУЕРНИ И СОФТУЕРНИ РЕШЕНИЯ ЗА АНАЛИЗ

- Логически анализатор и генератор на логически сигнали

ПОДБРАНИ ХАРДУЕРНИ И СОФТУЕРНИ РЕШЕНИЯ ЗА АНАЛИЗ

- Логически анализатор и генератор на логически сигнали
 - ▶ SQ50 - logic analyzer (IKALOGIC S.A.S)

ПОДБРАНИ ХАРДУЕРНИ И СОФТУЕРНИ РЕШЕНИЯ ЗА АНАЛИЗ

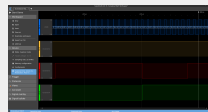
- Логически анализатор и генератор на логически сигнали
 - ▶ SQ50 - logic analyzer (IKALOGIC S.A.S)
 - ▶ ScanaStudio 16.04 (for Linux)

ПОДБРАНИ ХАРДУЕРНИ И СОФТУЕРНИ РЕШЕНИЯ ЗА АНАЛИЗ

- Логически анализатор и генератор на логически сигнали
 - ▶ SQ50 - logic analyzer (IKALOGIC S.A.S)
 - ▶ ScanaStudio 16.04 (for Linux)
- UART / USB Конвертор + picocom

ПОДБРАНИ ХАРДУЕРНИ И СОФТУЕРНИ РЕШЕНИЯ ЗА АНАЛИЗ

- Логически анализатор и генератор на логически сигнали
 - ▶ SQ50 - logic analyzer (IKALOGIC S.A.S)
 - ▶ ScanaStudio 16.04 (for Linux)
- UART / USB Конвертор + picocom



РЕШЕНИ ЗАДАЧИ ПО ОТНОШЕНИЕ НА СРЕДАТА

Конфигурация на компилатор

Конфигурация на компилатор

- Използване на хардуерния модул за числа с плаваща запетая

Конфигурация на компилатор

- Използване на хардуерния модул за числа с плаваща запетая
- Изключване на оптимизациите

Конфигурация на компилатор

- Използване на хардуерния модул за числа с плаваща запетая
- Изключване на оптимизациите
- Генериране на символи за дебъг

Линкерең скрипт

- Очертаване на разположението на отделните секции в паметта

Линкерен скрипт

- Очертаване на разположението на отделните секции в паметта
- Генериране на символи, използвани от буутлоудъра.

Линкерен скрипт

- Очертаване на разположението на отделните секции в паметта
- Генериране на символи, използвани от буутлоудъра.
- Правилно подравняване на отделните секции.

Линкерен скрипт

- Очертаване на разположението на отделните секции в паметта
- Генериране на символи, използвани от буутлоудъра.
- Правилно подравняване на отделните секции.
- Остранване на символи от стнадартните библиотеки.

Линкерен скрипт

- Очертаване на разположението на отделните секции в паметта
- Генериране на символи, използвани от буутлоудъра.
- Правилно подравняване на отделните секции.
- Остранване на символи от стнадартните библиотеки.

Конфигурация на дебъгер

Линкерен скрипт

- Очертаване на разположението на отделните секции в паметта
- Генериране на символи, използвани от буутлоудъра.
- Правилно подравняване на отделните секции.
- Остранване на символи от стнадартните библиотеки.

Конфигурация на дебъгер

- Автоматично отваряне на сесия и подвързване към дебъгерният порт.

Линкерен скрипт

- Очертаване на разположението на отделните секции в паметта
- Генериране на символи, използвани от буутлоудъра.
- Правилно подравняване на отделните секции.
- Остранване на символи от стнадартните библиотеки.

Конфигурация на дебъгер

- Автоматично отваряне на сесия и подвързване към дебъгерният порт.
- Предварително зареждане на дебъг символите от последния обектен файл.

Софтуерна част

Софтуерни модули за вътрешна и външна периферия

ОПИСАНИЕ НА МОДУЛИТЕ ЗА ВЪТРЕШНА ПЕРИФЕРИЯ

ОПИСАНИЕ НА МОДУЛИТЕ ЗА ВЪТРЕШНА ПЕРИФЕРИЯ

Софтуерните модули са изградени на база описанието на регистрите за вътрешната периферия, посочени в документа за техническа справка на микроконтролери от семейство *STM32F4xxx* [2].

ОПИСАНИЕ НА МОДУЛИТЕ ЗА ВЪТРЕШНА ПЕРИФЕРИЯ

Софтуерните модули са изградени на база описанието на регистрите за вътрешната периферия, посочени в документа за техническа справка на микроконтролери от семейство *STM32F4xxx* [2].

Основите за регистърните блокове на периферията в паметта са описани в дкоумента за съответния модел микроконтролер [3].

ИЗГРАДЕНИ МОДУЛИ ЗА ВЪТРЕШНА ПЕРИФЕРИЯ

RCC (Reset and Clock Control)

RCC (Reset and Clock Control)

Модул за управление на:

RCC (Reset and Clock Control)

Модул за управление на:

- Времеви бази (clock source: HSE, HSI, LSE)

RCC (Reset and Clock Control)

Модул за управление на:

- Времеви бази (clock source: HSE, HSI, LSE)
- PLL честотен множител

RCC (Reset and Clock Control)

Модул за управление на:

- Времеви бази (clock source: HSE, HSI, LSE)
- PLL честотен множител
- Делители на честота

RCC (Reset and Clock Control)

Модул за управление на:

- Времеви бази (clock source: HSE, HSI, LSE)
- PLL честотен множител
- Делители на честота
- Часовници на периферните шини

RCC (Reset and Clock Control)

Модул за управление на:

- Времеви бази (clock source: HSE, HSI, LSE)
- PLL честотен множител
- Делители на честота
- Часовници на периферните шини
- Часовници на периферията

NVIC (Nested Vectored Interrupt Controller)

NVIC (Nested Vectored Interrupt Controller)

Модул за управление на:

NVIC (Nested Vectored Interrupt Controller)

Модул за управление на:

- Конфигурацията на NVIC (контролера).

NVIC (Nested Vectored Interrupt Controller)

Модул за управление на:

- Конфигурацията на NVIC (контролера).
- Разрешаване и забрана на отделни прекъсвания.

NVIC (Nested Vectored Interrupt Controller)

Модул за управление на:

- Конфигурацията на NVIC (контролера).
- Разрешаване и забрана на отделни прекъсвания.
- Маскиране на прекъсвания.

NVIC (Nested Vectored Interrupt Controller)

Модул за управление на:

- Конфигурацията на NVIC (контролера).
- Разрешаване и забрана на отделни прекъсвания.
- Маскиране на прекъсвания.
- Приоритет на прекъсванията

NVIC (Nested Vectored Interrupt Controller)

Модул за управление на:

- Конфигурацията на NVIC (контролера).
- Разрешаване и забрана на отделни прекъсвания.
- Маскиране на прекъсвания.
- Приоритет на прекъсванията
- Локация на таблицата на прекъсванията.

GPIO (General Purpose Input/Output)

GPIO (General Purpose Input/Output)

Модул за управление на:

GPIO (General Purpose Input/Output)

Модул за управление на:

- Отделните GPIO портове.

GPIO (General Purpose Input/Output)

Модул за управление на:

- Отделните GPIO портове.
- Посока (изход/вход) за отделни пинове.

GPIO (General Purpose Input/Output)

Модул за управление на:

- Отделните GPIO портове.
- Посока (изход/вход) за отделни пинове.
- Режим на работа на отделните пинове (Open-drain, Push-Pull, Analog, AF).

GPIO (General Purpose Input/Output)

Модул за управление на:

- Отделните GPIO портове.
- Посока (изход/вход) за отделни пинове.
- Режим на работа на отделните пинове (Open-drain, Push-Pull, Analog, AF).
- Вътрешни (Pull-up/Pull-down/Floating) конфигурации.

GPIO (General Purpose Input/Output)

Модул за управление на:

- Отделните GPIO портове.
- Посока (изход/вход) за отделни пинове.
- Режим на работа на отделните пинове (Open-drain, Push-Pull, Analog, AF).
- Вътрешни (Pull-up/Pull-down/Floating) конфигурации.
- Източник за управление на състоянието на отделни пинове (Регистър/Алтернативна функция)

GPIO (General Purpose Input/Output)

Модул за управление на:

- Отделните GPIO портове.
- Посока (изход/вход) за отделни пинове.
- Режим на работа на отделните пинове (Open-drain, Push-Pull, Analog, AF).
- Вътрешни (Pull-up/Pull-down/Floating) конфигурации.
- Източник за управление на състоянието на отделни пинове (Регистър/Алтернативна функция)
- Работна честота на модула.

TIM (Timers)

TIM (Timers)

Модул за управление на прости таймери, таймери с общо предназначение и специализирани таймери:

TIM (Timers)

Модул за управление на прости таймери, таймери с общо предназначение и специализирани таймери:

- Управление на честотните делители.

TIM (Timers)

Модул за управление на прости таймери, таймери с общо предназначение и специализирани таймери:

- Управление на честотните делители.
- Настройка на период на таймера (ARR).

TIM (Timers)

Модул за управление на прости таймери, таймери с общо предназначение и специализирани таймери:

- Управление на честотните делители.
- Настройка на период на таймера (ARR).
- Настройка на режим на броене.

TIM (Timers)

Модул за управление на прости таймери, таймери с общо предназначение и специализирани таймери:

- Управление на честотните делители.
- Настройка на период на таймера (ARR).
- Настройка на режим на броене.
- Настройка на флагове, стартиращи прекъсвания.

TIM (Timers)

Модул за управление на прости таймери, таймери с общо предназначение и специализирани таймери:

- Управление на честотните делители.
- Настройка на период на таймера (ARR).
- Настройка на режим на броене.
- Настройка на флагове, стартиращи прекъсвания.
- Настройка на режимите на използване на CCR регистрите.

I2C (Inter Integrated Circuit)

I2C (Inter Integrated Circuit)

Модулът предоставя:

I2C (Inter Integrated Circuit)

Модулът предоставя:

- Конфигуриране на отделните I2C периферни модули.

I2C (Inter Integrated Circuit)

Модулът предоставя:

- Конфигуриране на отделните I2C периферни модули.
- Прости команди за отделни операции във връзка с I2C комуникацията.

I2C (Inter Integrated Circuit)

Модулът предоставя:

- Конфигуриране на отделните I2C периферни модули.
- Прости команди за отделни операции във връзка с I2C комуникацията.
- Абстрактни команди за управление на комуникацията изцяло през прекъсването.

I2C (Inter Integrated Circuit)

Модулът предоставя:

- Конфигуриране на отделните I2C периферни модули.
- Прости команди за отделни операции във връзка с I2C комуникацията.
- Абстрактни команди за управление на комуникацията изцяло през прекъсването.
- Динамична поправка на проблеми с комуникацията (при възможност).

USART

Модулът предоставя:

USART

Модулът предоставя:

- Опростен интерфейс за работа с USART периферията.

Модулът предоставя:

- Опростен интерфейс за работа с USART периферията.
- Настройка на честота на предаване/приемане

Модулът предоставя:

- Опростен интерфейс за работа с USART периферията.
- Настройка на честота на предаване/приемане
- Настройка на отделни параметри на комуникацията.

Софтуерна част

**Описание на модулите за външна
периферия**

ОПИСАНИЕ НА МОДУЛИТЕ ЗА ВЪНШНА ПЕРИФЕРИЯ

ОПИСАНИЕ НА МОДУЛИТЕ ЗА ВЪНШНА ПЕРИФЕРИЯ

Софтуерните модули за външната периферия са изградени на база описанието и комуникационните интерфейси на отделното използвано устройство.

ОПИСАНИЕ НА МОДУЛИТЕ ЗА ВЪНШНА ПЕРИФЕРИЯ

Софтуерните модули за външната периферия са изградени на база описанието и комуникационните интерфейси на отделното използвано устройство.

Строго проекто специфични, като се възползват от знанията за целите на проекта и са оптимизирани само и единствено за него.

ИЗГРАДЕНИ МОДУЛИ ЗА ВЪНШНА ПЕРИФЕРИЯ

6-канален RS приемник

6-канален RS приемник

Модул за четене и обновяване на данните, постъпващи от 6-канален RS приемник.

6-канален RC приемник

Модул за четене и обновяване на данните, постъпващи от 6-канален RC приемник.

- Сигнал тип PPM 50Hz. ($1000 \rightarrow 2000\mu s$)

6-канален RC приемник

Модул за четене и обновяване на данните, постъпващи от 6-канален RC приемник.

- Сигнал тип PPM 50Hz. ($1000 \rightarrow 2000\mu s$)
- Използват се TIM3 и TIM5 (резолюция $0.2\mu s$), режим Input-Capture.

6-канален RC приемник

Модул за четене и обновяване на данните, постъпващи от 6-канален RC приемник.

- Сигнал тип PPM 50Hz. ($1000 \rightarrow 2000\mu s$)
- Използват се TIM3 и TIM5 (резолюция $0.2\mu s$), режим Input-Capture.
- Сигналите се подновяват в структурата на суровите входове в момента на засичане

BLDC 4 канала

Модул за управление на заданията на четирите ESC за BLDC.

BLDC 4 канала

Модул за управление на заданията на четирите ESC за BLDC.

- Сигнал тип PPM 50Hz. ($1000 \rightarrow 2000\mu s$)

BLDC 4 канала

Модул за управление на заданията на четирите ESC за BLDC.

- Сигнал тип PPM 50Hz. ($1000 \rightarrow 2000\mu s$)
- Използва се TIM2 (резолюция $1\mu s$), режим Output-Compare.

BLDC 4 канала

Модул за управление на заданията на четирите ESC за BLDC.

- Сигнал тип PPM 50Hz. ($1000 \rightarrow 2000\mu s$)
- Използва се TIM2 (резолюция $1\mu s$), режим Output-Compare.
- Сигналите се подновяват след рестартиране на брояча чрез писане върху CCR регистрите.

BLDC 4 канала

Модул за управление на заданията на четирите ESC за BLDC.

- Сигнал тип PPM 50Hz. ($1000 \rightarrow 2000\mu s$)
- Използва се TIM2 (резолюция $1\mu s$), режим Output-Compare.
- Сигналите се подновяват след рестартиране на брояча чрез писане върху CCR регистрите.
- Режим на изходните пинове Open-Drain No-Internal-Pull.

Жироскоп и акселерометър

Жироскоп и акселерометър

Модул за конфигурация и четене на данните от жироскопа и акселерометъра:

Жироскоп и акселерометър

Модул за конфигурация и четене на данните от жироскопа и акселерометъра:

- Сигнал тип I2C.

Жироскоп и акселерометър

Модул за конфигурация и четене на данните от жироскопа и акселерометъра:

- Сигнал тип I2C.
- Използва се I2C3, режим Master.

Жироскоп и акселерометър

Модул за конфигурация и четене на данните от жироскопа и акселерометъра:

- Сигнал тип I2C.
- Използва се I2C3, режим Master.
- Хадуерът е конфигуриран на 52Hz честота на дискретизация.

Жироскоп и акселерометър

Модул за конфигурация и четене на данните от жироскопа и акселерометъра:

- Сигнал тип I2C.
- Използва се I2C3, режим Master.
- Хадуерът е конфигуриран на 52Hz честота на дискретизация.
- Обхват: акселерометър: 8g, жироскоп: 1000DPS.

Магнитометър

Магнитометър

Модул за конфигурация и четене на данните от магнитометъра:

Магнитометър

Модул за конфигурация и четене на данните от магнитометъра:

- Сигнал тип I2C.

Магнитометър

Модул за конфигурация и четене на данните от магнитометъра:

- Сигнал тип I2C.
- Използва се I2C3, режим Master.

Магнитометър

Модул за конфигурация и четене на данните от магнитометъра:

- Сигнал тип I2C.
- Използва се I2C3, режим Master.
- Хадуерът е конфигуриран на 52Hz честота на дискретизация.

Магнитометър

Модул за конфигурация и четене на данните от магнитометъра:

- Сигнал тип I2C.
- Използва се I2C3, режим Master.
- Хадуерът е конфигуриран на 52Hz честота на дискретизация.
- Обхват: 8gaus.

Софтуерна част

Допълнителни библиотеки

Допълнителни библиотеки

Библиотека за работа с матрици (стандарт за именоване CMSIS)

Библиотека за работа с матрици (стандарт за именоване CMSIS)

- Събиране и изваждане

Библиотека за работа с матрици (стандарт за именоване CMSIS)

- Събиране и изваждане
- Умножение

Библиотека за работа с матрици (стандарт за именоване CMSIS)

- Събиране и изваждане
- Умножение
- Транспониране

Библиотека за работа с матрици (стандарт за именоване CMSIS)

- Събиране и изваждане
- Умножение
- Транспониране
- Инверсия (Гаус-Жордан)

Библиотека за работа с матрици (стандарт за именоване CMSIS)

- Събиране и изваждане
- Умножение
- Транспониране
- Инверсия (Гаус-Жордан)
- Псевдоинверсия (Пенроз-Мур) (WIP)

■ Универсален ПИД регулатор

Библиотека за управление

- Универсален ПИД регулатор
- Релеен регулатор

Библиотека за управление

- Универсален ПИД регулатор
- Релеен регулатор
- Ограничители max-min

Софтуерна част

Инициализация на микроконтролера

Инициализация на микроконтролера

Инициализация на микроконтролера

Стартиране

ИНИЦИАЛИЗАЦИЯ НА МИКРОКОНТРОЛЕРА

Стартиране

Започва от `ResetHandler`, където е имплементиран примитивен буутлоудър

Примитивен буутлоудър (ARM assembly)

Примитивен буутлоудър (ARM assembly)

- Инициализира стековия указател (SP) и програмния брояч (PC)

Примитивен буутлоудър (ARM assembly)

- Инициализира стековия указател (SP) и програмния брояч (PC)
- Зарежда статичните променливи в паметта

Примитивен буутлоудър (ARM assembly)

- Инициализира стековия указател (SP) и програмния брояч (PC)
- Зарежда статичните променливи в паметта
- Инициализира .bss секцията с 0.

Примитивен буутлоудър (ARM assembly)

- Инициализира стековия указател (SP) и програмния брояч (PC)
- Зарежда статичните променливи в паметта
- Инициализира .bss секцията с 0.
- Извиква системната инициализация и отдава управлението към `main()`

Примитивен буутлоудър (ARM assembly)

- Инициализира стековия указател (SP) и програмния брояч (PC)
- Зарежда статичните променливи в паметта
- Инициализира .bss секцията с 0.
- Извиква системната инициализация и отдава управлението към `main()`
- Дефинира символите от таблицата на прекъсванията като `.weak` към `DefaultHandler`

Системна инициализация

- Конфигурация на системния часовник (168MHz)

- Конфигурация на системния часовник (168MHz)
 - Използване на HSE за времева основа

- Конфигурация на системния часовник (168MHz)
 - ▶ Използване на HSE за времева основа
 - ▶ Конфигурация на PLL (PLL_M=8, PLL_N=336, PLL_P=2, PLL_Q=7)

Системна инициализация

- Конфигурация на системния часовник (168MHz)
 - ▶ Използване на HSE за времева основа
 - ▶ Конфигурация на PLL (PLL_M=8, PLL_N=336, PLL_P=2, PLL_Q=7)
- Конфигурация на AHB и APB* (AHB_Prescaler=1, APB1_Prescaler=4, APB2_Prescaler=2).
- Разрешаване на хадруерното устройство за числа с плаваща запетая.

Хардуерна част

Хардуерна част

Свързване на системата

СВЪРЗАЩА ПЛАТКА

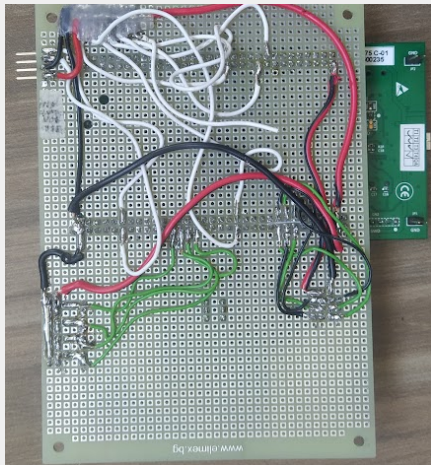


Предна страна

СВЪРЗАЩА ПЛАТКА



Предна страна



Задна страна

ПОДБРАНИ ПОРТОВЕ, ПИНОВЕ И ПЕРИФЕРИЯ

ПОДБРАНИ ПОРТОВЕ, ПИНОВЕ И ПЕРИФЕРИЯ

I2C

ПОДБРАНИ ПОРТОВЕ, ПИНОВЕ И ПЕРИФЕРИЯ

I2C

PA8 (I2C3 SCL) и PA9 (I2C3 SDA). Цялата периферия по шината оперира на 3.3V.

ПОДБРАНИ ПОРТОВЕ, ПИНОВЕ И ПЕРИФЕРИЯ

I2C

PA8 (I2C3 SCL) и PA9 (I2C3 SDA). Цялата периферия по шината оперира на 3.3V.

USART

ПОДБРАНИ ПОРТОВЕ, ПИНОВЕ И ПЕРИФЕРИЯ

I2C

PA8 (I2C3 SCL) и PA9 (I2C3 SDA). Цялата периферия по шината оперира на 3.3V.

USART

PB6(USART1 TX) и PB7 (USART1 RX). Цялата периферия по шината оперира на 3.3V.

ПОДБРАНИ ПОРТОВЕ, ПИНОВЕ И ПЕРИФЕРИЯ

I2C

PA8 (I2C3 SCL) и PA9 (I2C3 SDA). Цялата периферия по шината оперира на 3.3V.

USART

PB6(USART1 TX) и PB7 (USART1 RX). Цялата периферия по шината оперира на 3.3V.

Управляващи сигнали ESC

ПОДБРАНИ ПОРТОВЕ, ПИНОВЕ И ПЕРИФЕРИЯ

I2C

PA8 (I2C3 SCL) и PA9 (I2C3 SDA). Цялата периферия по шината оперира на 3.3V.

USART

PB6(USART1 TX) и PB7 (USART1 RX). Цялата периферия по шината оперира на 3.3V.

Управляващи сигнали ESC

PD12 - PD15. Конфигурация Open-drain, външни Pull-up резистори 10M Ω . Нужно е да се генерира 5V PPM сигнал.

Входове РРМ (приемник, сензор за разстояние)

Входове PPM (приемник, сензор за разстояние)

PA0(TIM5 канал 1), PA3(TIM5 канал 4), PB0(TIM3 канал 3),
PB1(TIM3 канал 4), PB4(TIM3 канал 2), PB5(TIM3 канал 1).

Периферията генерира 5V сигнали на входа. Входовете са 5V толерантни.

БАЛАНСИРАНЕ НА ВИТЛА



Фигура: Небалансирано витло

БАЛАНСИРАНЕ НА ВИТЛА



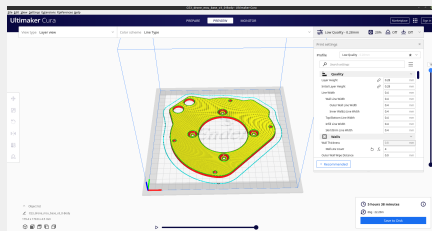
Фигура: Небалансирано витло



Фигура: Балансирано витло

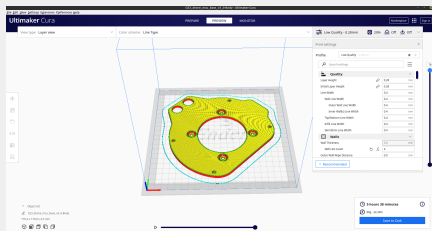
ОСНОВА ЗА МОНТИРАНЕ НА КОНТРОЛЕР И ВЪНШ- НА ПЕРИФЕРИЯ

ОСНОВА ЗА МОНТИРАНЕ НА КОНТРОЛЕР И ВЪНШНА ПЕРИФЕРИЯ

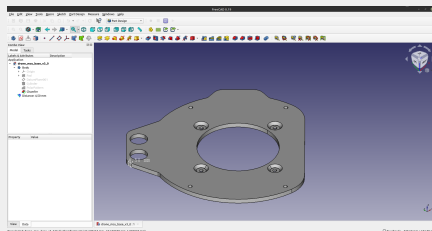


Фигура: Основа за монтиране,
Ultimaker Cura

ОСНОВА ЗА МОНТИРАНЕ НА КОНТРОЛЕР И ВЪНШНА ПЕРИФЕРИЯ



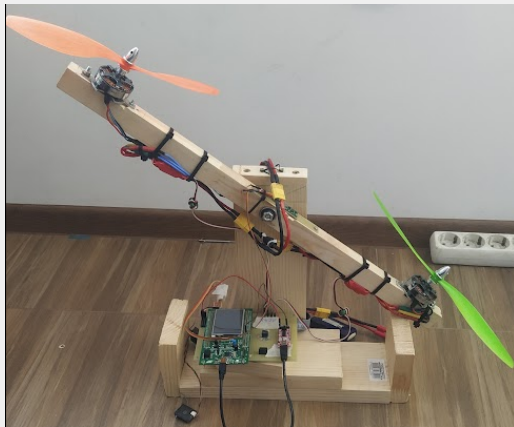
Фигура: Основа за монтиране, Ultimaker Cura



Фигура: Основа за монтиране FreeCad

ПЛАТФОРМА ЗА УПРАВЛЕНИЕ НА ЪГЪЛ НА ЗАВЪРТАНЕ

ПЛАТФОРМА ЗА УПРАВЛЕНИЕ НА ЪГЪЛ НА ЗАВЪРТАНЕ



ПЛАТФОРМА ЗА БЕЗПИЛОТЕН ЛЕТАТЕЛЕН АПАРАТ С ЧЕТИРИ РОТОРА

ПЛАТФОРМА ЗА БЕЗПИЛОТЕН ЛЕТАТЕЛЕН АПАРАТ С ЧЕТИРИ РОТОРА



Моделиране и идентификация

Моделиране и идентификация

Ротор с витло

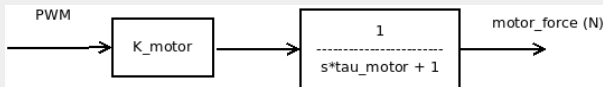
МОДЕЛИРАНЕ НА РОТОР С ВИТЛО

МОДЕЛИРАНЕ НА РОТОР С ВИТЛО

Опростен апроксимиран модел - нискочестотен филтър и коефициент на пропорционалност.

МОДЕЛИРАНЕ НА РОТОР С ВИТЛО

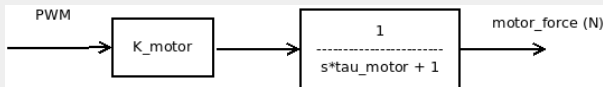
Опростен апроксимиран модел - нискочестотен филтър и коефициент на пропорционалност.



Фигура: Опростен модел на витло и мотор

МОДЕЛИРАНЕ НА РОТОР С ВИТЛО

Опростен апроксимиран модел - нискочестотен филтър и коефициент на пропорционалност.

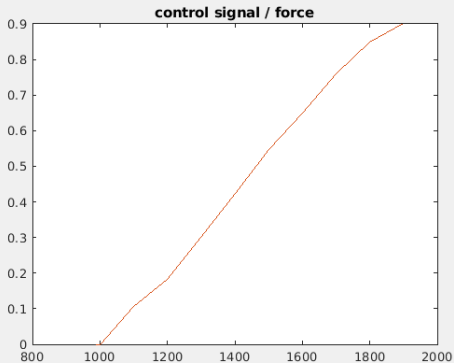


Фигура: Опростен модел на витло и мотор

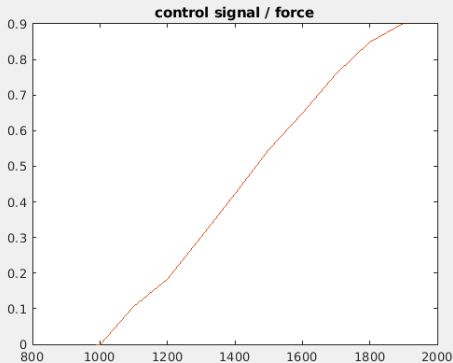
$$G_{motor}(s) = \frac{k_{motor}}{s\tau_{motor} + 1}$$

СЧЕМАНЕ НА ХАРАКТЕРИСТИКИ НА РОТОР С ВИТЛО

СЧЕМАНЕ НА ХАРАКТЕРИСТИКИ НА РОТОР С ВИТЛО

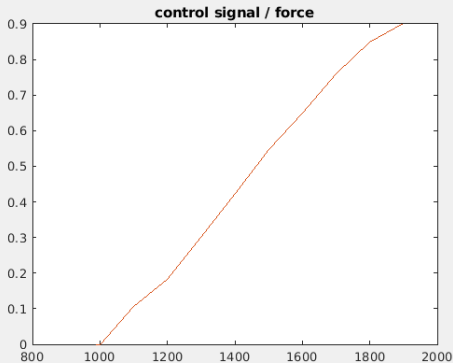


СЧЕМАНЕ НА ХАРАКТЕРИСТИКИ НА РОТОР С ВИТЛО



$k_{motor} = 12.2e - 3$ при сигнал $1300 \rightarrow 1700$.

СЧЕМАНЕ НА ХАРАКТЕРИСТИКИ НА РОТОР С ВИТЛО



$k_{motor} = 12.2e - 3$ при сигнал $1300 \rightarrow 1700$.

Опорната точка при тяга $424g \approx 4.15N$

Моделиране и идентификация

Платформа за управление на ъгъл на
завъртане

МОДЕЛИРАНЕ НА ПЛАТФОРМА ЗА УПРАВЛЕНИЕ НА ЪГЪЛ НА ЗАВЪРТАНЕ

Ако приемем:

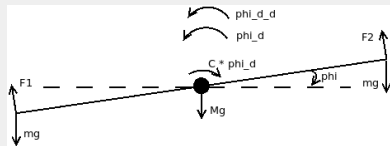
$$F_2 = F_0 + \delta f, F_1 = F_0 - \delta f$$

И апроксимираме:

$$I = \frac{l^2}{12}(M + 6m)$$

Получаваме:

$$G_{sys} = G_{motor}(s)G_{platform}(s) = \frac{k_{motor}l}{s(sl + C)(s\tau_{motor} + 1)}$$



ИДЕНТИФИКАЦИЯ НА ПАРАМЕТРИ НА ПЛАТФОРМА ЗА УПРАВЛЕНИЕ НА ЪГЪЛ НА ЗАВЪРТАНЕ

ИДЕНТИФИКАЦИЯ НА ПАРАМЕТРИ НА ПЛАТФОРМА ЗА УПРАВЛЕНИЕ НА ЪГЪЛ НА ЗАВЪРТАНЕ

След снемане на преходен процес:

ИДЕНТИФИКАЦИЯ НА ПАРАМЕТРИ НА ПЛАТФОРМА ЗА УПРАВЛЕНИЕ НА ЪГЪЛ НА ЗАВЪРТАНЕ

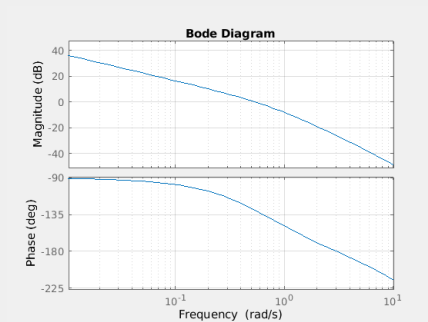
След снемане на преходен процес:

$$I = 0.025, C = 0.018, \tau_{motor} = 0.08$$

ИДЕНТИФИКАЦИЯ НА ПАРАМЕТРИ НА ПЛАТФОРМА ЗА УПРАВЛЕНИЕ НА ЪГЪЛ НА ЗАВЪРТАНЕ

След снемане на преходен процес:

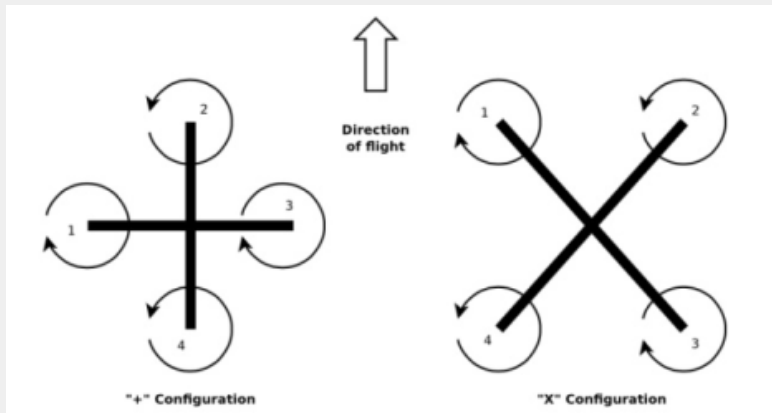
$$I = 0.025, C = 0.018, \tau_{motor} = 0.08$$



Моделиране и идентификация

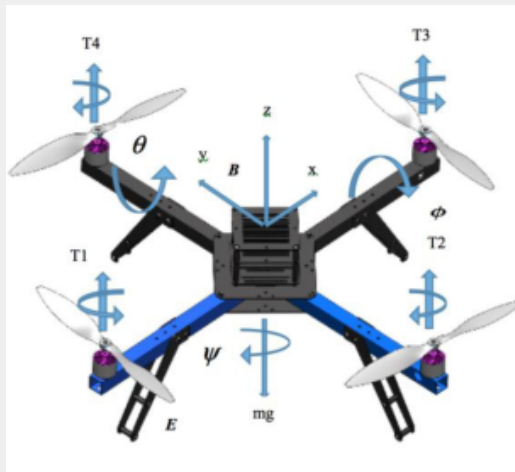
Платформа за безпилотен летателен апарат
с четири ротора

КОНФИГУРАЦИЯ



Сили, моменти и отправни координатни системи

Сили, моменти и отправни координатни системи



За да се носи платформата във въздуха, е нужно:

За да се носи платформата във въздуха, е нужно:

$$\sum_{i=1}^4 T_i = -mg$$

$$\sum_{i=1}^4 M_i = 0$$

$$T_{1,2,3,4} \parallel g$$

$$(\omega_1 + \omega_3) - (\omega_2 + \omega_4) = 0$$

За да се носи платформата във въздуха, е нужно:

$$\sum_{i=1}^4 T_i = -mg$$

$$\sum_{i=1}^4 M_i = 0$$

$$T_{1,2,3,4} \parallel g$$

$$(\omega_1 + \omega_3) - (\omega_2 + \omega_4) = 0$$

от което и следва, че $\phi = 0, \theta = 0, \psi = 0$.

За промяна на ориентацията се налага да:

За промяна на ориентацията се налага да:

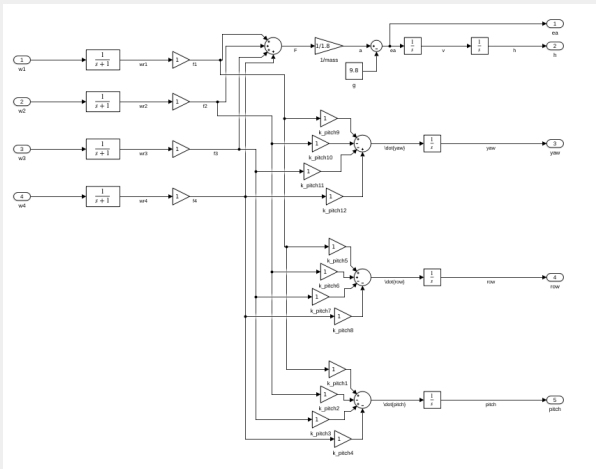
$$\dot{\psi} = k_{\psi}((\omega_1 + \omega_3) - (\omega_2 + \omega_4)), \psi = \int \dot{\psi} dt$$

$$\dot{\phi} = k_{\phi}((\omega_1 + \omega_4) - (\omega_2 + \omega_3)), \phi = \int \dot{\phi} dt$$

$$\dot{\theta} = k_{\theta}((\omega_1 + \omega_2) - (\omega_3 + \omega_4)), \theta = \int \dot{\theta} dt$$

МОДЕЛИРАНЕ НА СИСТЕМАТА ЧРЕЗ БЛОК СХЕМА

МОДЕЛИРАНЕ НА СИСТЕМАТА ЧРЕЗ БЛОК СХЕМА



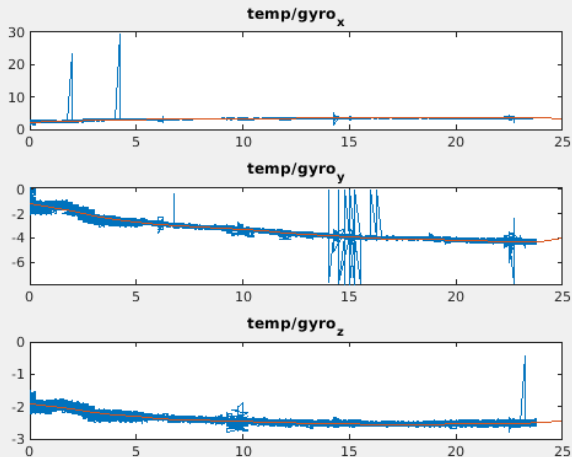
Калибриране и компенсация

Калибриране и компенсация

Жироскопен дрейф

КОМПЕНСАЦИЯ НА ЖИРОСКОПЕН ДРЕЙФ ПО ТЕМПЕРАТУРА

КОМПЕНСАЦИЯ НА ЖИРОСКОПЕН ДРЕЙФ ПО ТЕМПЕРАТУРА



$p_x =$

$$-42.6978e-9 x^7 + 3.7797e-6 x^6 - 131.8066e-006 x^5 + 2.2717e-3 x^4 - 19.5918e-3 x^3 + 67.9744e-3 x^2 + 95.1384e-3 x + 2.2440e+0$$

$p_y =$

$$87.1716e-9 x^7 - 7.6384e-6 x^6 + 263.1019e-006 x^5 - 4.4660e-3 x^4 + 37.8030e-3 x^3 - 129.2365e-3 x^2 - 191.6577e-3 x - 1.1695e+0$$

$p_z =$

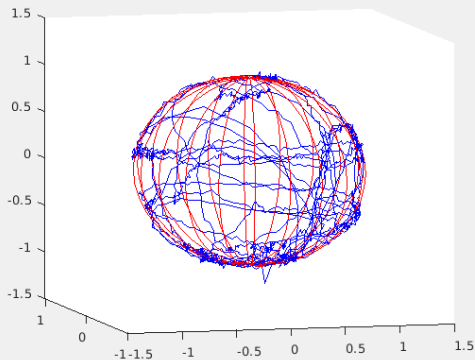
$$15.9591e-9 x^7 - 1.4613e-6 x^6 + 53.0721e-006 x^5 - 961.5439e-6 x^4 + 8.8365e-3 x^3 - 33.3197e-3 x^2 - 44.0584e-003 x - 1.9076e+0$$

Калибриране и компенсация

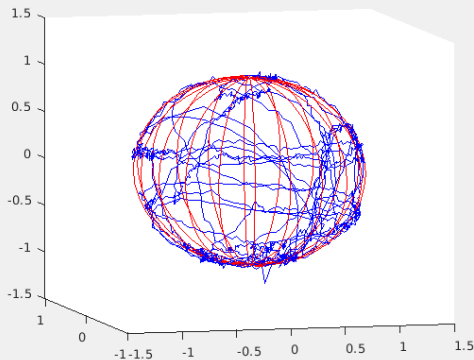
Магнитни отмествания от околната среда

КОМПЕНСАЦИЯ НА МАГНИТНИ ОТМЕСТВАНИЯ ОТ ОКОЛНАТА СРЕДА

КОМПЕНСАЦИЯ НА МАГНИТНИ ОТМЕСТВАНИЯ ОТ ОКОЛНАТА СРЕДА



КОМПЕНСАЦИЯ НА МАГНИТНИ ОТМЕСТВАНИЯ ОТ ОКОЛНАТА СРЕДА



Използвана е техника на Мерайо[1]
(трансформация отместен елипсоид - центрирана сфера).

U =

21.4229e+000	837.9000e-003	3.0861e+000
0.0000e+000	21.6201e+000	2.0207e+000
0.0000e+000	0.0000e+000	27.2441e+000

c =

-1.6000e-003
19.9000e-003
45.7000e-003

Синтез на наблюдател

ФИЛТЪР НА КАЛМАН ЗА ОЦЕНКА НА ЪГЪЛ НА ЗАВЪРТАНЕ

ФИЛТЪР НА КАЛМАН ЗА ОЦЕНКА НА ЪГЪЛ НА ЗАВЪРТАНЕ

$$\begin{bmatrix} \phi \\ \psi \\ \omega_{xb} \\ \omega_{yb} \end{bmatrix}_k = \begin{bmatrix} 1 & 0 & -d_t & 0 \\ 0 & 1 & 0 & -d_t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}_k * \begin{bmatrix} \phi \\ \psi \\ \omega_{xb} \\ \omega_{yb} \end{bmatrix}_{k-1} + \begin{bmatrix} d_t & 0 & 0 & 0 \\ 0 & d_t & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}_k * \begin{bmatrix} \omega_x \\ \omega_y \\ 0 \\ 0 \end{bmatrix}_k + N(0, Q_k)$$

$$\begin{bmatrix} A_\phi \\ A_\psi \\ 0 \\ 0 \end{bmatrix}_k = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}_k * \begin{bmatrix} \phi \\ \psi \\ \omega_{xb} \\ \omega_{yb} \end{bmatrix}_k + N(0, R_k)$$

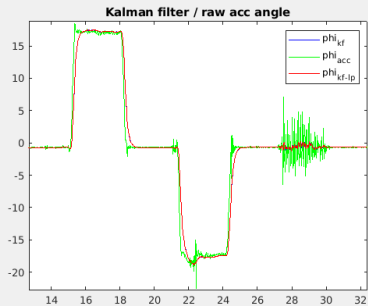
КОВАРИАНТНИ МАТРИЦИ

$$Q = \begin{bmatrix} 33.8 & 0 & 0 & 0 \\ 0 & 33.8 & 0 & 0 \\ 0 & 0 & 0.4 & 0 \\ 0 & 0 & 0 & 0.4 \end{bmatrix},$$

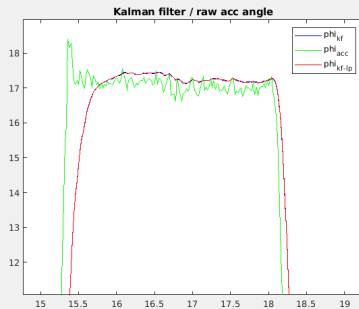
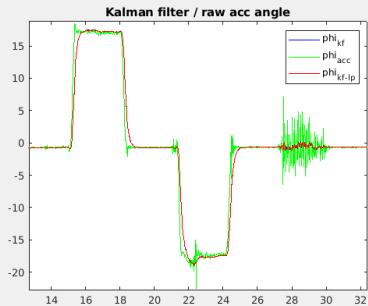
$$R = \begin{bmatrix} 32 & 0 & 0 & 0 \\ 0 & 32 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

ПРОВЕРКА НА РЕЗУЛТАТИТЕ

ПРОВЕРКА НА РЕЗУЛТАТИТЕ



ПРОВЕРКА НА РЕЗУЛТАТИТЕ



Синтез на управление

УПРАВЛЕНИЕ НА ЪГЪЛ НА ЗАВЪРТАНЕ

Опростена схема на управлението:

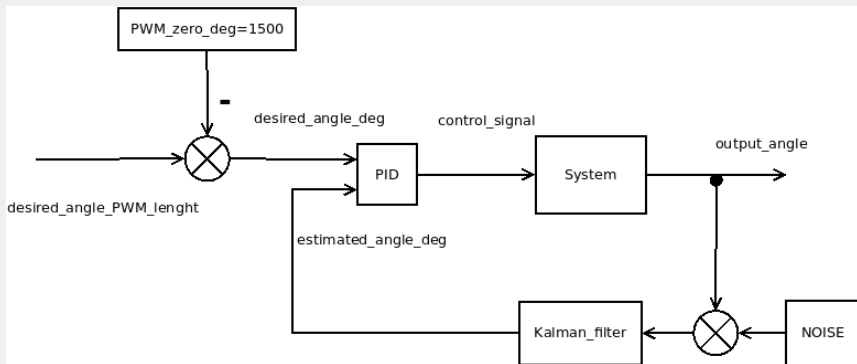
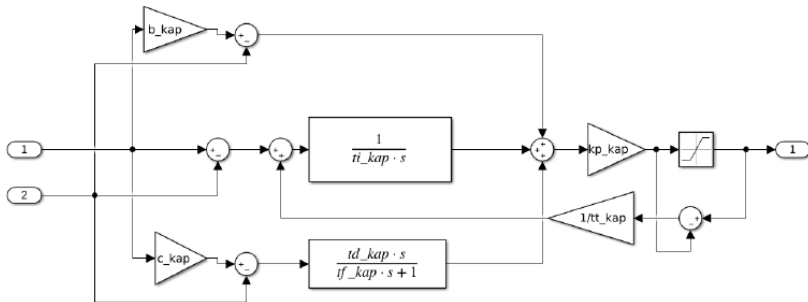


СХЕМА НА ИМПЛЕМЕНТИРАНИЯ УНИВЕРСАЛЕН ПВД

СХЕМА НА ИМПЛЕМЕНТИРАНИЯ УНИВЕРСАЛЕН ПИД



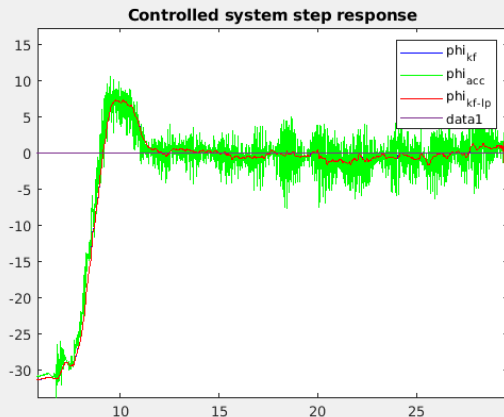
ПАРАМЕТРИ НА РЕГУЛАТОР И ПРЕХОДЕН ПРОЦЕС

ПАРАМЕТРИ НА РЕГУЛЯТОР И ПРЕХОДЕН ПРОЦЕС

$$K_p = 0.4, K_i = 0.003, K_d = 2, \tau_d = 0.26, out_{limit} = \pm 150$$


ПАРАМЕТРИ НА РЕГУЛЯТОР И ПРЕХОДЕН ПРОЦЕС

$$K_p = 0.4, K_i = 0.003, K_d = 2, \tau_d = 0.26, out_{limit} = \pm 150$$



ПРЕДЛОЖЕНИЯ ЗА НАДГРАЖДАНЕ.

ЗАКЛЮЧЕНИЕ.

 JOSÉ MG MERAYO, PETER BRAUER, FRITZ PRIMDAHL, JAN RAAGAARD PETERSEN, AND OTTO V NIELSEN.

Scalar calibration of vector magnetometers.

Measurement science and technology, 11(2):120, 2000.

 ST MICROELECTRONICS.

RM0090 Reference manual STM32F405/415, STM32F407/417, STM32F427/437 and STM32F429/439 advanced ARM-based 32-bit MCUs.

ST Microelectronics, 2020.

 ST MICROELECTRONICS.

STM32F427xx STM32F429xx 32b Arm Cortex-M4 MCU+FPU, 225DMIPS, up to 2MB Flash/256+4KB RAM, USB OTG HS/FS, Ethernet, 17 TIMs, 3 ADCs, 20 com. interfaces, camera and LCD-TFT.

БЛАГОДАРЯ ЗА ВНИМАНИЕТО!