****

**ОТЧЕТ**

**по производственной практике**

**студента группы И-19-2р**

**специальности 6В06106- «Информатика»**

**Мольганов Андрей Александрович**

**Ф.И.О. студента**

**Место прохождения практики:**

**ТОО «НПО «Группа Компаний «DOSTI»**

**название организации**

**Руководители практики:**

**факультета “ИТ”,**

**Полегенько И.Г.**

**Алматы 2022 г.**

|  |  |
| --- | --- |
| **Дата** | **Количество работы** |
|  | Знакомство с производством (компанией), коллективом. Ознакомление с техникой безопаности и правилами эксплуатации оборудования. |
|  | Определение рабочего места и дальнейшего фронта работы. |
|  | Установка необходимого аппаратного и программного окружения. |
|  | Знакомство с IDE STM32CubeIDE и разработка встраиваемых приложений для «MPS-2.1» на языке программирования C/C++ |
|  | Исследование механизмов аппаратной и программной защиты информации на микроконтроллерной платформе STM32F103 |
|  | Отчет о проделанной работе и сдача руководству |

**Содержание отчета по производственной практике**

**по специальности «Информатика»**

|  |  |  |
| --- | --- | --- |
|  | Введение (цель, задачи практики) ……………………………………… | 4 |
| 1 | Общая характеристика ТОО «НПО «Группа Компаний «DOSTI»» …………………………………………………………............................... | 5 |
| 2 | Роль и функции «Разработчика встраиваемых систем (Безопасность встраиваемых систем)» ………………………………………………… | 6 |
| 3 | Характеристика стенда учебного программируемого «MPS-2.1» …… | 9 |
| 4 | Разработка встраиваемых приложений для «MPS-2.1» ………………. | 12 |
| 5 | Исследование механизмов аппаратной и программной защиты информации……………………………………………………………… | 15 |
|  | Заключение………………………………………………………………. | 19 |
|  | Список использованной литературы…………………………………… | 21 |
|  | Список использованных графических элементов……………………… | 21 |

**ВВЕДЕНИЕ**

Преддипломная практика является составной частью программы подготовки высококвалифицированных кадров. Основным содержанием практики является выполнение практических, творческих, исследовательских, и других проектов, соответствующих характеру будущей профессиональной деятельности, подготовка к выпускной квалификационной работе.

Цель преддипломной практики:

– закрепление теоретических знаний и сбор материала для выполнения выпускной квалификационной работы.

Задачи преддипломной практики:

1. Закрепление знаний и навыков, полученных при теоретическом обучении;
2. Овладение опытом работы с конкретными проектными материалами, изделиями и оборудованием;
3. Сбор необходимых информационных материалов, книг, монографий и научных статей для выполнения выпускной квалификационной работы;
4. Формирование профессиональной позиции обучающегося, его мировоззрения, стиля поведения, освоение профессиональной этики.
5. Приобретение навыков корпоративной, а также совместной работы в одной группе с высококвалифицированными специалистами различных специализаций и компетенций.

Процесс прохождения преддипломной практики направлен на формирование следующих компетенций:

* формирует навыки использовать нормативные правовые документы в профессиональной деятельности;
* стремится к саморазвитию, повышению своей квалификации и мастерства;
* умеет критически оценивать свои достоинства и недостатки, наметить пути и выбрать средства самосовершенствования;
* осознает социальную значимость своей будущей профессии, обладает высокой мотивацией к выполнению профессиональной деятельности;
* способен анализировать и определять требования к проекту;
* способен составлять подробную спецификацию требований к проекту;
* способен синтезировать набор возможных решений задачи или подходов к выполнению проектов различной сложности;
* способен научно и аргументированно обосновать свои предложения, идеи и мысли.

1. **Общая характеристика ТОО «НПО «Группа Компаний «DOSTI»**

Инновационный центр «DOSTI» предоставляет широкий спектр направлений в сфере робототехники для обучения и индивидуального развития. Компания является партнером и спонсором движения WorldSkills Kazakhstan, которое направлено на поднятие престижа и развития рабочих специальностей в стране.

Также совместно с НАО "Talap" при поддержке Министерства Образования и Науки, компания участвует в разработке программы обучения современным компетенциями согласно программе международных чемпионатов WorldSkills в учреждениях средне-специального и высшего образования Республики Казахстан.

Направления, которые охватывает компания:

* Электроника;
* Объектно-ориентированное, низкоуровневое, функциональное программирование;
* Программирование микроконтроллерных платформ (Arduino, STM32, National Instruments myRIO и т.д.);
* Прототипирование;
* Аддитивные технологии;
* Разработка мобильных приложений.

Компания имеет в своей структуре 2 ключевых отдела отвечающих за деятельность компании:

1. **Учебно-методический отдел**

Данный отдел ответственен за организацию и проведение курсов обучения, повышения квалификации, подготовки к чемпионатам WorldSkills, за написание учебных пособий, методических указаний, учебных планов и программ, за проведение учебных и производственных практик для студентов колледжей и университетов. Отдел отвечает за создание новых продуктов (проектов, устройств и т.п.), анализ и подготовку материалов для продуктов, подлежащих промышленному производству.

1. **Проектно-производственный отдел**

Данный отдел ответственен за производство продуктов (электронных устройств) компании. Отдел имеет в своем распоряжении лабораторию, укомплектованную всем необходимым оборудованием для производства электронных устройств начиная от печатных плат и заканчивая корпусами устройств. Проектно-производственный отдел принимает студентов на практику в ограниченном количестве и только тех, кого одобрил учебно-методический отдел (после прохождения комплексного тестирования на определенные навыки и умения).

**2. Роль и функции «Разработчика встраиваемых систем (Безопасность встраиваемых систем)»**

Начиная с конца 1950-х годов, крупные промышленные компании, специализирующиеся на производстве специальных инженерных или транспортных систем, например аэрокосмическая компания Boeing или электротехнические компании Westinghouse Electric Company, Toshiba задумались о создании компактных встраиваемых системах в своих продуктах. Впервые такие встраиваемые системы появились в конце 1960-х годов с развитие космической программы США и СССР, где первые встраиваемые системы управляли множеством подсистем космических ракет-носителей и даже помогали людям дистанционно исследовать планеты Марс и Венера с помощью автоматических межпланетных станций.

Встраиваемые системы – информационно-коммуникационная система, состоящая из двух групп компонентов – аппаратного и программного обеспечения, чаще всего такие системы разработаны специально для выполнения конкретного прикладного или фундаментального применения. Например, блок управления навигацией и ориентированием самолета Airbus A350 (см. Рисунок 2.1).



Рисунок 2.1 – Блок управления навигацией и ориентированием самолета Airbus A350

Область применения встраиваемых систем не ограничивается только массивными самолетами и кораблями, встраиваемые системы являясь одной из областей компьютерных наук, постоянно развивается, находя все новые и новые сферы применения (см. Рисунок 2.2). На сегодняшний день, встраиваемые системы используются в следующих сферах:

* Разработка измерительного оборудования (осциллограф, анализатор сигналов, спектрометр, импульсные блоки питания);
* Разработка бортовых управляющих систем (колесный автотранспорт, железнодорожный транспорт, аэрокосмический и водный транспорт);
* Разработка медицинского оборудования (высокоточное хирургическое, лабораторное, терапевтическое, диагностическое, физиотерапевтическое оборудование);
* Разработка телекоммуникационных станций (базовые станции различных телекоммуникационных стандартов, например 4G, 5G, GPS, LoRaWAN);
* Разработка мехатронных и робототехнических станций (станки с числовым программным управлением, автомобили с автономным управлением, высокоточные 3D-принтеры);
* Разработка системы безопасности и сигнализации (блоки управления видеонаблюдением, доступом и полноценные системы управления доступом).



Рисунок 2.2 – Встраиваемая система на основе микроконтроллера ARM промышленного стандарта

Для создания таких специализированных информационно-коммуникационных систем, в системах образования ведущих стран мира была создана специальность «Разработчик встраиваемых систем». Разработчик встраиваемых систем – это уникальный высококвалифицированный специалист в нескольких смежных областях фундаментальной и прикладной науки, область которого заключается в следующем:

1. Архитектура встраиваемых систем – с помощью доступных аппаратных и программных средств разработчик создает встраиваемую систему для конкретного использования на производстве. Например, блок управления асинхронным электродвигателем для создания приемлемых тяговых характеристик электроподвижного состава метрополитена (см. Рисунок 2.3):

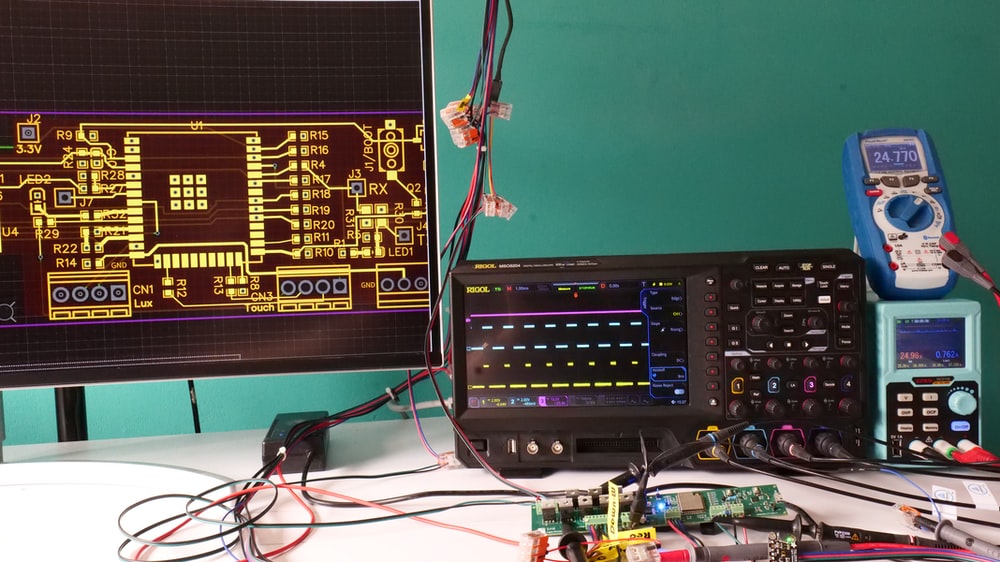


Рисунок 2.3 – Разработка и отладка встраиваемых систем на основе микроконтроллера ARM промышленного стандарта с использованием измерительного оборудования (осциллограф, мультиметр, блок питания) и САПР Altium Designer 22

1. Программирование встраиваемых систем – после создания аппаратного обеспечения встраиваемой системы, разработчик реализует потенциально лучший алгоритм работы на производстве (см. Рисунок 2.4):

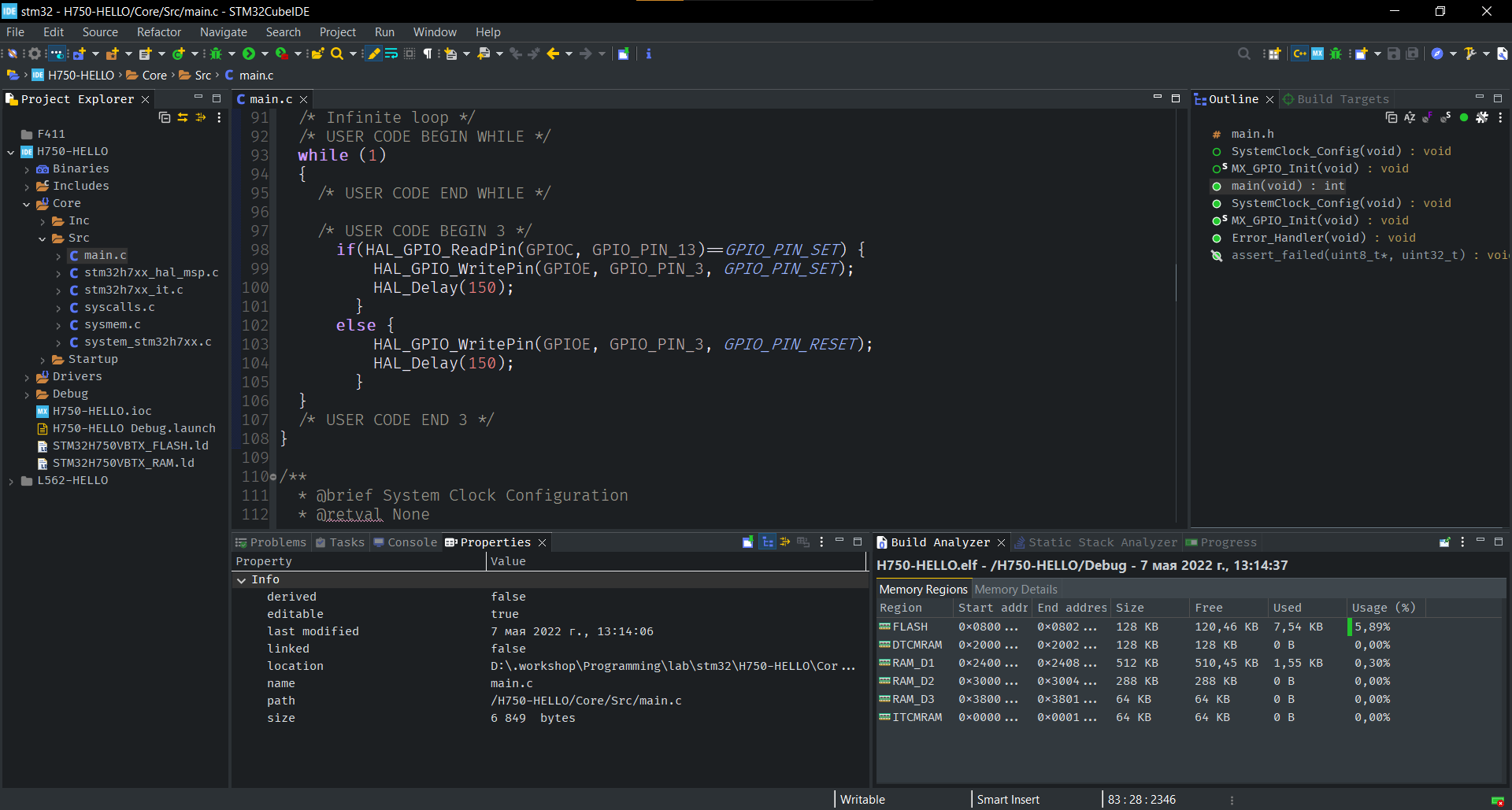


Рисунок 2.4 – Разработка и отладка программного кода встраиваемых систем на основе микроконтроллера ARM промышленного стандарта с использованием IDE STM32CubeIDE

1. Промышленное производство встраиваемых систем – команда разработчиков после опытного производства и создания необходимой инженерной документации для производства и конечного пользователя, внедряет промышленное производство встраиваемых систем и определяет фронт работы для улучшений встраиваемой системы после первого производства в промышленном масштабе.

**3. Характеристика стенда учебного программируемого «MPS-2.1»**

|  |  |
| --- | --- |
|  |  |
| Рисунок 3.1 и 3.2 – Общий и инженерный вид стенда учебного программируемого MPS-2.1 | |

«Стенд учебный программируемый «MPS-2.1» (см. Рисунок 3.1 и 3.2) представляет собой компактное устройство, выполненное из ударопрочного пластика включающее в себя большое количество различных модулей. Устройство моделирует работу основных систем современного легкового автомобиля. Стенд позволяет создавать различные конфигурации программ приближенных к используемым в транспортных, роботизированных и мехатронных системах различной специализации. [9]

Устройство обладает рядом достоинств:

* мобильность;
* возможность использования устройства в автономном режиме и от сети;
* защита от некорректных действий;
* изолированный ударопрочный корпус;
* простота замены отдельных модулей;
* возможность визуального наблюдения за работой модулей.

Таблица 3.3 - Технические характеристики MPS-2

|  |  |
| --- | --- |
| **Параметр** | **Значение** |
| Размеры (ШхВхГ) (без учета колес) | 160х83х210 мм |
| Размеры (ШхВхГ) (с учетом колес) | 220х83х210 мм |
| Масса | 800 гр. |
| Рабочая температура | 10-35 С° |
| Рабочее напряжение | 7-10 В |
| Напряжение питания адаптера | 220 В |
| Максимальная скорость вращения колёс | 300 об/мин |
| Максимальный ток адаптера | 3 А |
| Напряжение на выходе адаптера | 9 В |
| Тип аккумуляторов | Li-Ion стандарта 18650 |
| Количество аккумуляторов | 2 шт. |
| Количество символов дисплея LCD2004 | 20х4 |
| Количество символов дисплея TM1637 | 4 |

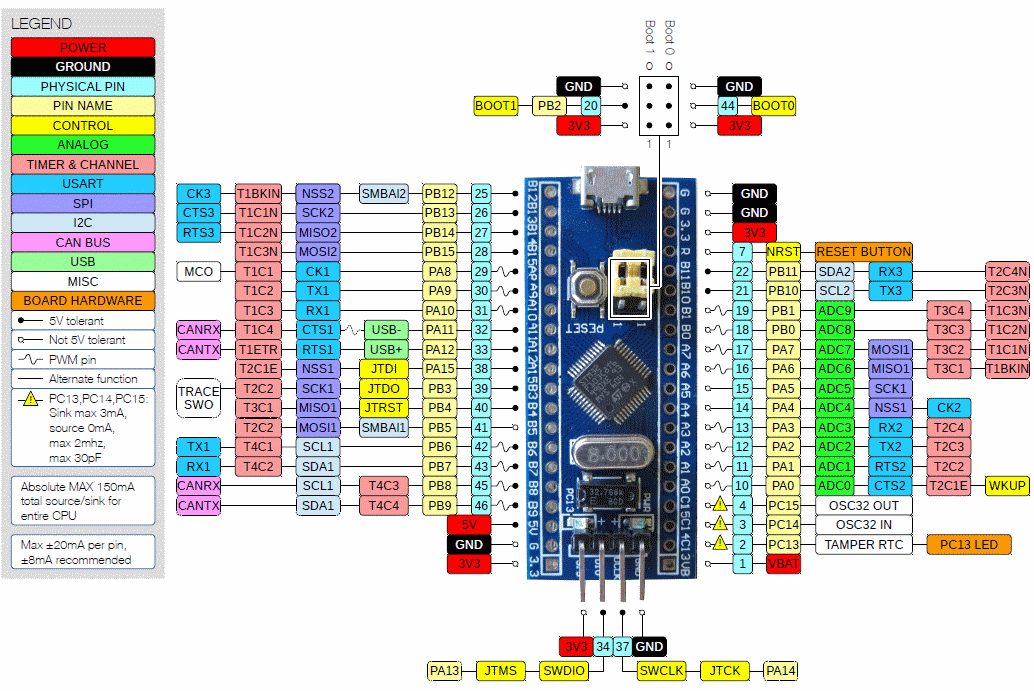


Рисунок 3.4 – Микроконтроллер STM32F103C8T6 и его распиновка

Основой устройства является STM32F103C8T6 (см. Рисунок 3.4) — это микроконтроллер, построенный на ядре ARM Cortex-M3. Данное ядро имеет много преимуществ, его основное преимущество на сегодняшний день — универсальность. За годы существования ядра Cortex-M3, он стал индустриальным стандартом. Об этом говорит количество производителей, присоединившихся к данной архитектуре и количество выпущенных устройств на базе данного микроконтроллера. Все основные производители микроконтроллеров, которые присутствуют в СНГ, кроме Microchip, имеют или развивают решения на основе этой архитектуры. В 32-битном сегменте разработчики редко работают с языком низкого уровня программирования (Ассемблер), в основном используются языки среднего и высокого уровня, например, язык C или C++. [10]

Таблица 3.5 – Технические характеристики микроконтроллера STM32F103C8T6

|  |  |
| --- | --- |
| **Параметр** | **Значение** |
| Разрядность микроконтроллера | 32 бита |
| Максимальная частота | 72 МГц |
| Объем памяти программ (FLASH) | 64 кбайт |
| Объем памяти данных (RAM) | 20 кбайт |
| Общее количество входов/выходов | 37 |
| Количество таймеров общего назначения | 3 |
| Количество расширенных таймер с ШИМ-коррекцией управлением двигателя | 1 |
| Системный таймер | 1 |
| Сторожевые таймеры | 2 |
| Количество интерфейсов UART | 3 |
| Количество интерфейсов SPI | 2 |
| Количество интерфейсов I2C | 2 |
| Количество интерфейсов CAN | 1 |
| Количество интерфейсов USB | 1 |
| Контроллеры прямого доступа к памяти | 7 |
| Наличие АЦП/ЦАП | АЦП 10x12b |
| Часы реального времени | есть |
| Аппаратный модуль расчета CRC | есть |
| Напряжение питания микроконтроллера | от 2 до 3,6 В |
| Напряжение питания платы | 5 В |
| Ток потребления | до 50 мА |
| Размеры платы | 53 x 22,5 мм |

Семейство STM32 отличается от конкурентов хорошим поведением в температурном диапазоне от -40 до 85°С. Производительность ядра и периферии сохраняется полностью. В семействе STM32 есть ряд изделий, сертифицированных на расширенный температурный диапазон от -40 до 105°С.

Одна из причин мировой популярности семейства STM32 — максимальный комфорт разработчика. Если универсальность ядра STM32 позволяет менять производителя c минимальными затратами на программный код, то Pin-to-Pin совместимость внутри семейства STM32 позволяет менять объем памяти (флэш-память и ОЗУ) и периферию (Ethernet, USB, CAN, и т.д.), не трогая печатную плату. «Pin-to-Pin совместимость» означает, что для одного размера корпуса все сигналы сохраняются на тех же самых вводах/выводах для разных вариантов микроконтроллеров семейства. [3]

**4. Разработка встраиваемых приложений для «MPS-2.1»**

Для того, чтобы разрабатывать встраиваемые приложения, необходимо сначала определиться с классификацией встраиваемых систем на основе используемой микропроцессорной/микроконтроллерной платформы (см. Таблицу 4.1)

Таблица 4.1 – Технические характеристики микроконтроллера STM32F103C8T6

|  |  |  |  |
| --- | --- | --- | --- |
| № | Класс | Применение | Фотография |
| 1 | Встраиваемые системы на основе FPGA (Программируемые Логические Интегральные Схемы) | Системы аппаратного искусственного интеллекта  Системы анализа телекоммуникационного трафика  Комплексы аппаратной защиты информации  Системы сбора и анализа данных |  |
| 2 | Встраиваемые системы на основе микропроцессоров (ARMv9, RISC-V) | Системы управления мехатронными и робототехническими системами с искусственным интеллектом  Системы управления энергетическими установками малого и среднего класса  Системы сбора данных с помощью датчиков |  |
| 3 | Встраиваемые системы на основе микроконтроллеров (ARM, RISC-V) | Обучающие системы и комплексы  Системы сбора данных  Системы управления электродвигателями малого класса |  |

Разработка встраиваемых приложений для «MPS-2.1» осуществляется с помощью интегрированной среды разработки STM32CubeIDE (см. Рисунок 4.2) компании STMicroelectornics с бесплатной лицензией.

Рисунок 4.2 – Основные этапы разработки приложения для встраиваемых систем

В составе STM32CubeIDE есть также специальное программное обеспечение STM32CubeMX, служащее для упрощения программирования и проведения первоначальной настройки микроконтроллера. Основным преимуществом данной среды разработки является то, что благодаря специальному хранилищу, при подключенном интернете, пользователь имеет доступ ко множеству библиотек и примеров. Основное пространство окна занимает текстовый редактор кода, в нем пользователь набирает и редактирует свои коды. В текстовом редакторе присутствует указание на ошибки, если код набран неверно, появляется надпись об этом. Также имеется очень удобная система авто дополнения, которая, при написании некоторой последовательности символов, предлагает разработчику дополнить текст чтобы получить необходимые функции в программировании. [5]

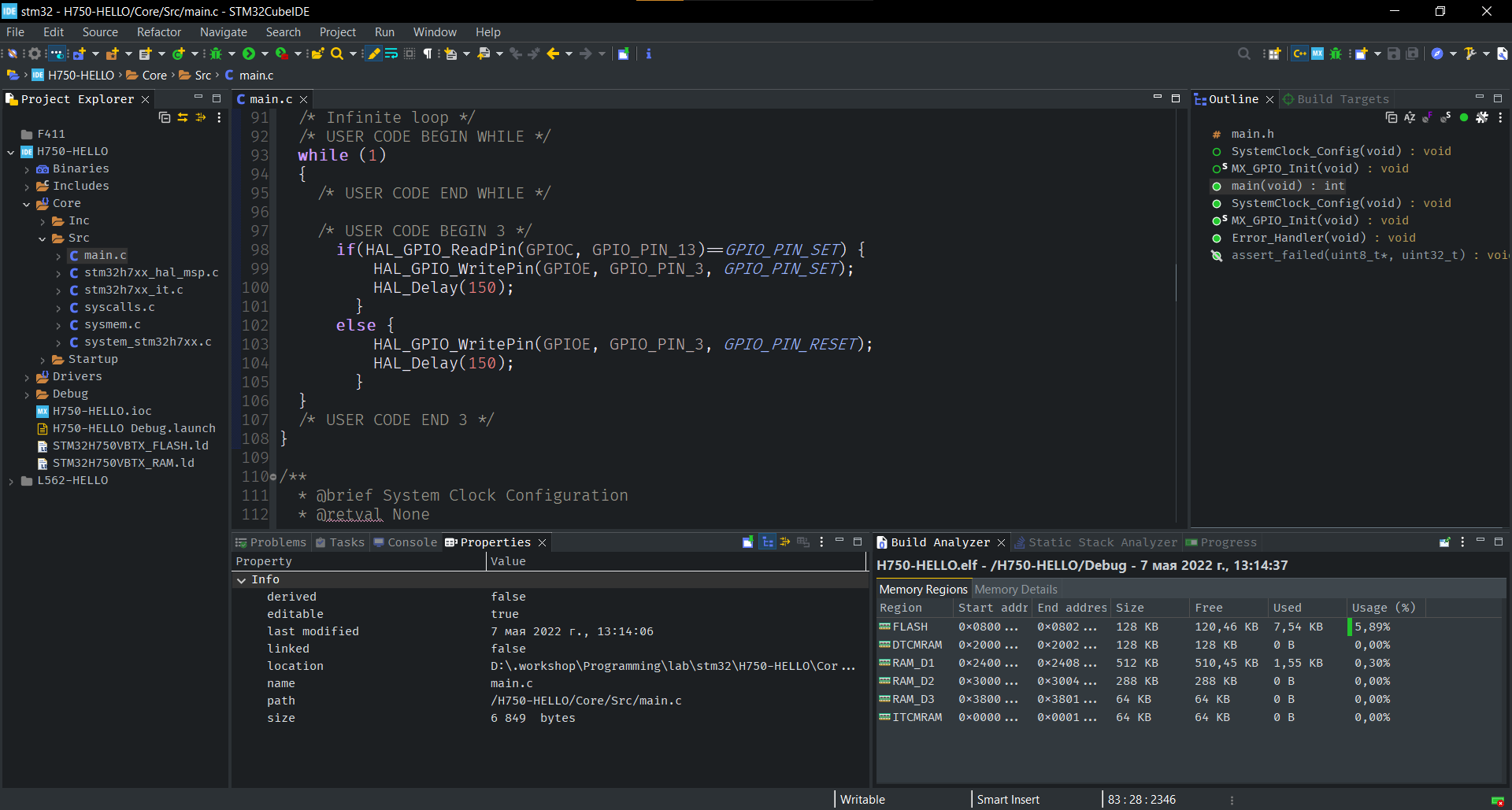


Рисунок 4.2 – Разработка и отладка программного кода встраиваемых систем на основе микроконтроллера ARM промышленного стандарта с использованием IDE STM32CubeIDE

STM32CubeIDE сочетает в себе несколько утилит, необходимых для правильной первоначальной настройки и программирования микроконтроллеров STM32, а именно:

* STM32CubeMX – проводит первоначальную настройку микроконтроллера и дает возможность использовать внешние и внутренние интерфейсы микроконтроллера;
* STM32CubeProgrammer – проводит глубокую настройку микроконтроллера, а также позволяет в реальном времени следить за основными характеристиками микроконтроллера;
* STM32CubeMonitor – проводит анализ всей встраиваемой системы с помощью внешних и внутренних датчиков;

Подытоживая, можно сказать, что разработка приложений для встраиваемых систем, стала намного более удобной, появилось множество дополнительных утилит при работе с такими системами, которые качественно улучшают продуктивность и качество разработки программного кода для встраиваемых систем. [1]

**5. Исследование механизмов аппаратной и программной защиты информации**

Молниеносное развитие сетевых технологий, сподвигло многих компаний и людей задуматься над безопасность передаваемой, отправляемой и хранимой информацией. Существует огромное количество приложений, которые по определению должны обеспечивать высокую степень защиты данных. Это касается Интернета вещей (см. Рисунок 5.1), терминалов оплаты, банкоматов, счетчиков коммунальных услуг, систем безопасности и многих других. При этом в каждом из перечисленных случаев необходимо защитить данные не только от кражи, но и от вредоносного изменения. Это достаточно сложная задача.

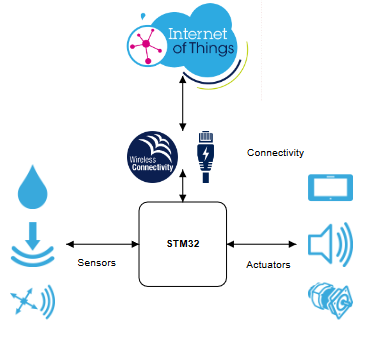


Рисунок 4.2 – Составная схема устройства Интернета вещей на базе микроконтроллера STM32

В июля 2016 года компания STMicroelectronics презентовала свою фирменную библиотеку шифрования для надежного хранения и защиты информация непосредственно на микроконтроллерах. Библиотека была выпущена в качестве программного расширения для фирменной интегрированной среды разработки STM32CubeIDE. Библиотека имеет сертифицированную поддержку следующих алгоритмов:

* Криптографические алгоритмы серии AES с разрядностью 128, 192, и 256 в режимах ECB, CBC, CTR, CFB, OFB, CCM, GCM, CMAC, KEY WRAP, XTS;
* Хеш-функции с поддержкой режима HMAC серии SHA с разрядностью 1, 224, 256, 384, 512;
* Программный генератор случайных чисел на базе DRBG-AES-128;
* Создание и хранение цифровых ключей по стандарту RSA с поддержкой протокола PKCS-1v1.5 с режимами кодирования/декодирования и хранения цифровой подписи;
* Генерация и хранения ключей по стандарту ECC с режимами Scalar multiplication и ECDSA. [2]

Кроме того, библиотека имеет несертифицированную поддержку дополнительных алгоритмов ARC4, DES, TripleDES (ECB (Electronic Codebook Mode) и CBC (Cipher-Block Chaining)), хеш-функции (MD5 и HKDF-SHA-512), ChaCha20, Poly1305, CHaCHA20-POLY1305, ED25519, Curve25519. Схему работы криптографического сопроцессора можно увидеть на Рисунке 4.3.

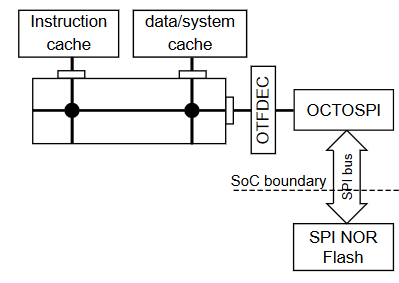


Рисунок 4.3 – Схема работы криптографического сопроцессора на базе микроконтроллера STM32

*Проблему защиты данных можно разделить на несколько задач:*

* **Сохранение целостности информации** подразумевает защиту от несанкционированного или случайного искажения данных. Под искажением понимают удаление, изменение или вставку сторонней информации в защищаемый блок данных. Чтобы обезопасить себя от подобных воздействий, система должна иметь возможности их обнаружения, то есть приемник, получив блок данных, должен убедиться, что они не были изменены.
* **Конфиденциальность** подразумевает, что доступ к закрытым данным могут иметь только авторизованные пользователи. Доступ неавторизованных пользователей или процессов должен быть исключен.
* **Идентификация и аутентификация** позволяют сторонам, участвующим в обмене данными, идентифицировать себя. Если аутентификация проходит успешно, пользователь или процесс получают доступ к информации. Аналогично есть необходимость идентификации источника данных.
* **Безотказность с подтверждением отправителя** подразумевает, что получатель данных обеспечивается проверкой отправителя данных. Это защищает от любой попытки отправителя ложно отрицать передачу данных или их содержимое.
* **Безотказность с подтверждением доставки** гарантирует, что передатчик данных обеспечивается подтверждением доставки данных. Это защищает от любой последующей попытки получателя ложно отрицать получение данных или их содержимое. [3]
* **Шифрование.** Суть этого метода заключается в преобразовании открытых данных в зашифрованную форму. Кодирование производится с помощью специальных алгоритмов и ключа шифрования. Обратный процесс декодирования также требует ключа. При этом возможно два варианта. При симметричном шифровании для кодирования и декодирования используется один и тот же ключ. При асимметричном шифровании для кодирования используется один ключ, а для декодирования — другой.
* **Функции хеширования (свертки)**— это особый вид односторонних функций, которые с помощью определенного алгоритма получают из исходного массива данных битовую строку заданной длины. Исходные данные часто называют ключом (сообщением), а выходную битовую строку «хеш-кодом», «хеш-суммой» или «сводкой сообщения». Используемый алгоритм преобразования должен обеспечивать защиту от возможностей восстановления исходных данных по известному хеш-коду и совпадения хеш-кодов разных сообщений;
* **Код аутентификации сообщений MAC *(****Message Authentication Code****)***. Данный механизм, как и хеш-функции, используется для генерации хеш-кода. Однако для этого требуется не только исходное сообщение, но и секретный ключ, известный только отправителю и получателю. Это позволяет получателю данных проверять целостность информации и идентифицировать отправителя. Если отправитель не имеет секретного ключа — хеш-код будет сформирован неверно, что легко обнаружит получатель;
* **Цифровые подписи.** Этот механизм позволяет производить аутентификацию сообщений, то есть доказывать их подлинность с помощью цифровой подписи. Цифровая подпись работает практически так же, как и обычная подпись на бумаге — по ней всегда можно распознать отправителя. При получении цифровой подписи применяют асимметричное шифрование. Для шифрования сообщения используется закрытый ключ, а для расшифровки — открытый. Закрытый ключ известен только отправителю, в то время как к открытому ключу доступ может иметь множество получателей данных;
* **Генерация случайных чисел.** Защитное кодирование имеет смысл, только если невозможно разгадать ключ шифрования. По этой причине ключ должен быть случайным. Для этого применяется генератор случайных чисел. Он может использовать как программные методы формирования исходной последовательности случайных чисел, так и естественный источник случайного сигнала, например, шум напряжения на диоде. [4]

**Таким образом, библиотека шифрования для микроконтроллерной платформы STM32** представляет собой набор скомпилированных файлов для различных семейств микроконтроллеров и сред разработки. При этом каждый скомпилированный вариант реализации строится по модульному принципу. Такая структура позволяет компилятору использовать только те модули, которые нужны пользователю (AES CTR, AES CCM, HASH SHA, и другие криптографические алгоритмы), а остальные, для экономии памяти, не включать в проект. Это также позволяет при необходимости добавлять модули в любой момент времени, как только это потребуется. Так как библиотеки шифрования представлены в виде скомпилированных библиотечных файлов, то пользователи могут включать их в свои проекты, но не могут вносить в них изменения. По этой причине каждая библиотека поставляется в нескольких экземплярах: для каждой среды разработки (IAR, Keil ARM®, GCC) и с различными опциями, такими как оптимизация по скорости, оптимизация по объему памяти и так далее.

**ЗАКЛЮЧЕНИЕ**

Рассмотренные на практике средства защиты компьютерной  
информации касательно применения таких средств на микроконтроллерных платформах семейства ARM и STM32, конечно же, не закрывают весь спектр угроз безопасности информации.

Кроме этого, всегда следует помнить, что методы и средства  
осуществления угроз безопасности информации постоянно совершенствуются.  
С развитием систем обработки и передачи данных появляются новые виды  
угроз. Поэтому постоянное развитие претерпевают и системы защиты  
информации. Наряду с этим источники угроз ведут систематический поиск новых уязвимостей в существующих пользовательских приложения, информационных службах и системах защиты.

Все это говорит о том, что современному специалисту по защите  
информации необходимо постоянно обновлять и расширять свои знания о  
существующих угрозах, уязвимостях и современных средствах, и методах  
обеспечения информационной безопасности.

Важно понимать, что защита информации не ограничивается  
техническими методами. Большое количество угроз  
несанкционированного доступа к конфиденциальной информации могут быть  
реализованы по причине неправильной работы пользователей компьютерной  
системы вследствие незнания основ сетевой безопасности.  
Обучение пользователей правилам сетевой безопасности может  
предотвратить многие сетевые атаки. Защита информации включает в себя  
кроме технических мер еще и обучение или правильный подбор  
обслуживающего персонала.

Помимо этого, защита должна постоянно совершенствоваться вместе с  
развитием компьютерной сети. Опасно недооценивать и так называемые внутренние ИТ-угрозы, обусловленные действиями легальных пользователей корпоративной системы.

При проектировании, разработке, создании и отладке систем защиты информации следует учитывать возможные негативные последствия злонамеренных действий легальных пользователей.

Поэтому специалист по защите информации пристально внимание  
должен уделять также разработке административного регламента работы  
пользователей с корпоративной информационной системой, информационной  
политике и документальному сопровождению системы безопасности.

В настоящее время обобщенная теория безопасности информации пока не  
создана. Применяемые на практике подходы и средства нередко страдают  
недостатками и не обладают объявленной надежностью. Поэтому необходимо  
обладать достаточной подготовкой и квалифицированно ориентироваться во  
всем спектре вопросов обеспечения информационной безопасности, понимая  
их комплексный и взаимообусловленный характер.

Во время прохождения практики подкрепил полученные во время учебы теоретические навыки практическими. Познакомился с организацией, структурой и принципами принцип работы предприятия. Развил коммуникативные навыки. Получил опыт работы в коллективе.

Во время прохождения практики показал себя, как способного и ответственного сотрудника. Выполнял задания качественно и своевременно.

Научился разрабатывать, проектировать, отлаживать и использовать методы защиты информации в приложения для встраиваемых систем. Во время прохождения практики я узнала многие пункты по техники безопасности, которых я до этого не знал. Эта практика научила меня пунктуальности, приходить вовремя на рабочее место.

Я стал ответственным, терпеливым и так думаю хорошим специалистом, что нельзя было сказать обо мне до этого. Данная практика является хорошим практическим опытом для дальнейшей самостоятельной профессиональной деятельности.

В связи с этим, хотелось бы выразить огромное спасибо моему руководителю по практике – Вафаеву Абдулазизжону Рустамовичу за создание теплой, дружественной и рабочей атмосферы, я освоил несколько аппаратных и программных технологий, а также закрепил знания в нескольких смежных областях науки и производства. Прохождение практики оставило приятные впечатления о производстве, специалистах и людях, которые не покладая рук совершенствуют свой продукт и помогают вырасти молодым кадрам в высококвалифицированных специалистов.

**СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ**

[1] – Пош М., Программирование встроенных систем на C++ 17 / пер. с англ. А.В. Снастина. – М.: ДМК Пресс, 2020. – 394 с.: ил.

[2] – Новелло К., Освоение STM32 / пер. с англ. Д. Карасев. – М.: Leanpub Publishing, 2018. – 826 с.: ил.

[3] – Novello C., Mastering STM32 – Second Edition. - М.: Leanpub Publishing, 2022. – 910 с.: ил.

[4] – AN5156 Application Note – Introduction to STM32 microcontrollers security., 2019. – 55 с.: ил.

[5] – Кудряшов Б.Д., Теория информации: Учебник для вузов. – СПб: Питер, 2009. – 320 с.: ил.

[6] – Бабенко Л.К., Ищукова Е.А., Сидоров И.Д., Параллельные алгоритмы для решения задач защиты информации. – 2-е изд., стереотип. – М.: Горячая линия-Телеком, 2014. – 304 с., ил.

[7] – Владимиров С.М., Габидулин Э.М., Колыбельников А.И., Кшевецкий А.С., Криптографические методы защиты информации. – М.: Издательство Московского Государственного Университета, 2021. – 433 с., ил.

[8] – Романец Ю.В., Тимофеев П.А., Шаньгин В.Ф., Защита информации в компьютерных системах и сетях / под ред. В.Ф. Шаньгина. – 2-е изд., перераб. и доп. – М.: Радио и связь, 2001. – 376 с.: ил.

[9] – Беляков С.Л., Боженюк А.В., Петряева М.В., Основы разработки программы на языке C++ для систем информационной безопасности: учебное пособие. – М.: Издательство Южного Федерального Университета, Ростов-на-Дону, 2020. – 152 с.: ил.

[10] – Варлатая С.К., Шаханова М.В., Аппаратно-программные средства и методы защиты информации. – М.: Издательство Дальневосточного Государственного Технического Университета, 2007. – 318 с.: ил.

**СПИСОК ИСПОЛЬЗОВАННЫХ ГРАФИЧЕСКИХ ЭЛЕМЕНТОВ**

Рисунок 2.1 – <https://www.aviationtoday.com/wp-content/uploads/2019/12/a350-xwb-cockpit-touchscreen_.jpeg>

Рисунок 2.2 – <https://unsplash.com/photos/-7RXjpd_XKw>

Рисунок 2.3 – <https://unsplash.com/photos/Ky0B3cBJGKo>

Рисунок 3.4 – <https://stm32duinoforum.com/forum/download/file_id_1358.jpeg>