

ДОКУМЕНТАЦИЯ

(описание технических средств хранения исходного текста и объектного кода программного обеспечения, а также технических средств компиляции исходного текста в объектный код программного обеспечения)

Содержание

1 Описание технических средств хранения исходного текста и объектного кода программного обеспечения.....	3
2 Описание технических средств компиляции исходного текста в объектный код программного обеспечения.....	5

1 Описание технических средств хранения исходного текста и объектного кода программного обеспечения

Для хранения исходного текста (на языке ассемблера) и объектного кода программы интеллектуального датчика напряжения используются следующие технические средства:

Локальное хранилище разработчика:

- **Персональный компьютер (ПК)** с установленной операционной системой (Windows/Linux/macOS), обеспечивающий работу среды разработки MPLAB X IDE v5.30.
- **Внутренние накопители:** жёсткий диск (HDD) или твердотельный накопитель (SSD) ПК, на которых хранятся:
 - файлы исходного кода (с расширением `.asm`) – ассемблер для STM32);
 - объектные файлы (`.obj`, `.o`);
 - исполняемые файлы (`.hex`, `.bin`);
 - проектные файлы MPLAB X (`.X`).
- **Внешние накопители** (USB-флешки, внешние HDD/SSD) – для резервного копирования и переноса данных между рабочими станциями.

Системы управления версиями:

- **Локальные репозитории** Git, Mercurial и т. д. – для отслеживания изменений в коде, ведения истории версий и возможности отката к предыдущим состояниям.
- **Удалённые репозитории** (GitHub, GitLab, Bitbucket) – для коллективной разработки, резервного хранения и обеспечения доступа к коду из разных локаций.

Специализированное оборудование:

- **Программатор ChipProg-48** – используется для записи объектного кода в память микроконтроллера STM32. Хранит временные копии прошивок в своей внутренней памяти при процессе программирования.
- **Микроконтроллер STM32** – содержит встроенную флеш-память, в которой хранится окончательный объектный код программы после прошивки.

Организационные меры:

- регулярное резервное копирование на сетевые хранилища или облачные сервисы;
- разграничение прав доступа к файлам и репозиториям;
- ведение журнала изменений (changelog) для учёта версий ПО.

2 Описание технических средств компиляции исходного текста в объектный код программного обеспечения

Процесс компиляции исходного кода на ассемблере в объектный код для микроконтроллера STM32 выполняется с помощью следующего набора инструментов:

Основная среда разработки:

- **MPLAB X IDE v5.30** – интегрированная среда разработки (IDE) от Microchip Technology, обеспечивающая:
 - редактирование исходного кода на ассемблере;
 - вызов компилятора и компоновщика;
 - отладку и симуляцию работы программы;
 - генерацию выходных файлов (HEX, BIN) для прошивки.

Компилятор и сопутствующие утилиты:

- **XC32 Compiler Suite** (или совместимый ассемблер для ARM Cortex-M) – набор инструментов, включающий:
 - ассемблер (преобразует код на ассемблере в объектные файлы `.o`);
 - компоновщик (линковщик) – объединяет объектные файлы и библиотеки в единый загрузочный модуль;
 - утилиту для генерации HEX-файла (готового к прошивке).

Аппаратные средства:

- **Персональный компьютер** с MPLAB X IDE и установленным компилятором XC32 – выполняет процесс компиляции.
- **Программатор ChipProg-48** – служит для:
 - загрузки скомпилированного объектного кода (в формате HEX/BIN) в память микроконтроллера STM32;

- верификации правильности записи (сравнение считанного кода с исходным HEX-файлом).

Целевое устройство:

- **Микроконтроллер семейства STM32** (например, STM32F103C8T6 или аналогичный) – содержит ядро ARM Cortex-M, для которого производится компиляция кода на ассемблере.

Процесс компиляции и прошивки:

1. Разработчик пишет исходный код на ассемблере (файл `.asm`) в MPLAB X.
2. Запускает компиляцию: MPLAB X вызывает ассемблер XC32, который создаёт объектные файлы.
3. Линковщик объединяет объектные модули и библиотеки, формирует выходной HEX-файл.
4. Через интерфейс MPLAB X запускается процесс прошивки: HEX-файл передаётся в программатор ChipProg-48.
5. Программатор записывает код во флеш-память микроконтроллера STM32.
6. Выполняется верификация: ChipProg-48 считывает записанную память и сравнивает с исходным HEX-файлом.

Дополнительные инструменты:

- отладочные платы (например, STM32 Discovery) – для тестирования кода до окончательной интеграции в устройство защиты;
- логический анализатор или осциллограф – для проверки работы алгоритма на реальном железе;
- утилиты командной строки (make, bash-скрипты) – для автоматизации сборки и прошивки.