Осциллограф С8-53/1

Техническое описание программного обеспечения

ТОПО

Минск, ОАО МНИПИ 2017

Оглавление

[Загрузчик 3](#_Toc473188382)

[Настройки 3](#_Toc473188383)

[Создание прошивки 5](#_Toc473188384)

[Хранение серийного номера и номера версии программного обеспечения 6](#_Toc473188385)

[Исправление бага с неправильным расчётом адреса для сохранения настроек и мёртвым зависанием при включении вследствие этого 7](#_Toc473188386)

# **Загрузчик**

Основная программа начинается с адреса 0x08020000 и занимает сектора 5, 6 и 7 размерами по 128 кб.

В начальных секторах находится загрузчик. Его функции:

- при включении загрузчик проверяет наличие подключённой флешки. Если флешка не обнаружена, происходит передача управления на адрес 0x08020000. Если флешка обнаружена:

- происходит поиск в корневом каталоге файла S8-53.bin. Если файл обнаружен, выводится запрос «На подключенном носителе обнаружена версия программного обеспечения. Установить?». В случае выбора варианта «НЕТ» происходит передача управления на адрес 0x08020000. Если выбрано «ДА»:

- стираются сектора 5, 6 и 7; затем в них записывается содержимое файла S8-53.bin, начиная с адреса 0x08020000. Затем выводится сообщение «Обновление установлено успешно» и происходит переход по адресу 0x08020000.

## Настройки

Проект S8-53

1. Target

Read/Only Memory Areas – on-chip – IROM1 : 0x8000000 0x60000

Code Generation ARM Compiler : Use default compiler version

Read/Write Memory Areas – on-chip – IRAM1 : 0x20000000 0x20000

1. User

After Build/Rebuild – Run#1 :

fromelf –bin –output.\s8-53.bin .\out\s8-53.afx

1. C/C++

Preprocessor Symbols – Define : USE\_HAL\_DRIVER, STM32F207xx

Language/Code Generation – Optimization : Level 0(-O0)

Language/Code Generation – One ELF Section per Function

Language/Code Generation – Warnings : All Warnings

Misc Controls : --C99

1. Linker

Report ‘might fail’ Conditions as Errors

R/O Base : 0x08000000

R/W Base : 0x20000000

Scatter File : .\S8-53.sct

; \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

; \*\*\* Scatter-Loading Description File generated by uVision \*\*\*

; \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

LR\_IROM1 0x08020000 0x00060000 { ; load region size\_region

ER\_IROM1 0x08020000 0x00060000 { ; load address = execution address

\*.o (RESET, +First)

\*(InRoot$$Sections)

.ANY (+RO)

}

RW\_IRAM1 0x20000000 0x00020000 { ; RW data

.ANY (+RW +ZI)

}

}

1. Debug

CPU DLL : SARMCM3.DLL Parameter : -REMAP –MPU

Dialog DLL : DCM.DLL Parameter : -pCM3

Driver DLL : SARMCM3.DLL Parameter : -MPU

Dialog DLL : TCM.DLL Parameter : -pCM3

1. Utilities – Settings

## Создание прошивки

Для того, чтоб получить итоговую прошивку, в её начало нужно записать серийный номер

# Хранение серийного номера и номера версии программного обеспечения

Серийный номер и номер версии программного обеспечения хранятся в секторе 3 размером 16 кБайт по следующим адресам.

0x08010000 – серийный номер в виде НОМЕР (пробел) ГОД, где:

НОМЕР – uint8 (в будущем, возможно, uint16) , где хранится номер прибора;

ГОД – uint32 с годом выпуска.

0x08010010 – версия программного обеспечения в uint32.

Запись этих значений производится через меню ОТЛАДКА-ДАННЫЕ при залитой прошивке.

Также при обновлении через флешку загрузчик перезаписывает версию на актуальную. Актуальная версия прошивки хранится в её первых десяти байтах в виде символьной строки.

# Исправление бага с неправильным расчётом адреса для сохранения настроек и мёртвым зависанием при включении вследствие этого

Данный дефект присутствует в приборах, отгруженных до 26.01.2017 (приблизительно 5 шт.). Возник вследствие неправильного расчёта количества оставшейся памяти в секторе настроек (абсолютные адреса, которые на 0x08000000 больше относительных, сравнивались с относительными, например, с размеров всей флеш-памяти 1МБ).

Для того, чтобы считать последние правильные настройки, нужно просмотреть область памяти массива структур RecordConfig (файл FLASH.c) и найти предппоследнюю заполненную структуру. 26.01.2017 это был адрес 0x080E063C. В строке 75 заменяем строку

RecordConfig \*record = RecordConfigForRead();

на строку

RecordConfig \*record = (RecordConfig\*)0x080E063C;

Далее в функции FLASH\_SaveSettings() следует указать, что места в памяти для сохранения настроек больше нет. Тогда сектор будет стёрт. Это делается заменой строки 105 с

RecordConfig \*record = FindRecordConfigForWrite();

на

RecordConfig \*record = (RecordConfig\*)0;

Затем следует восстановить строки и перезалить.