

Приложение А

Оглавление

Описание отладочного комплекта	2
Электрическая схема отладочной платы КФЦС.441461.196	4
Установка и настройка Keil IDE	8
Создание и обновление файла конфигурации (hex-файл)	10
Отладка в Keil IDE	11
Настройка отладчика	11
Начало работы с отладкой	12
Возможности отладчика	13
Использование точек останова	16
Просмотр состояния переменных	16
Команда «Flash Download»	17
Установка и настройка Eclipse	19
Настройка окружения	19
Установка Ubuntu 22.04	19
Установка Windows	19
Создание проекта	20
Настройка проекта	23
Удаление/закрытие проекта	25
Описание проекта	26
Руководство по отладке в Eclipse	27
Подготовка к работе с отладочным комплектом	29
Режимы работы микросхемы	30
Программирование микросхемы в режиме отладки (режим «SOFT»)	30
Программирование микросхемы в режим финальной конфигурации (режим «HARD»)	34
Программирование конфигурационной памяти в режиме отладки (режим «SOFT»)	36
Программирование конфигурационной памяти в режим «HARD»	38
Программный сброс	39
Настройка напряжения питания универсальных портов ввода-вывода	40
Обновление программного обеспечения DCSProg	41
Лист регистрации изменений	42

Описание отладочного комплекта

Для демонстрации функциональных возможностей, настройки и программирования микросхемы 5400TP105-003 разработан отладочный комплект КФЦС.441461.197.

Состав отладочного комплекта КФЦС.441461.197:

- Отладочная плата КФЦС.441461.196
- USB-кабель для подключения отладочной платы к ПК
- Блок питания
- ПО DCSProg6 для программирования микросхемы

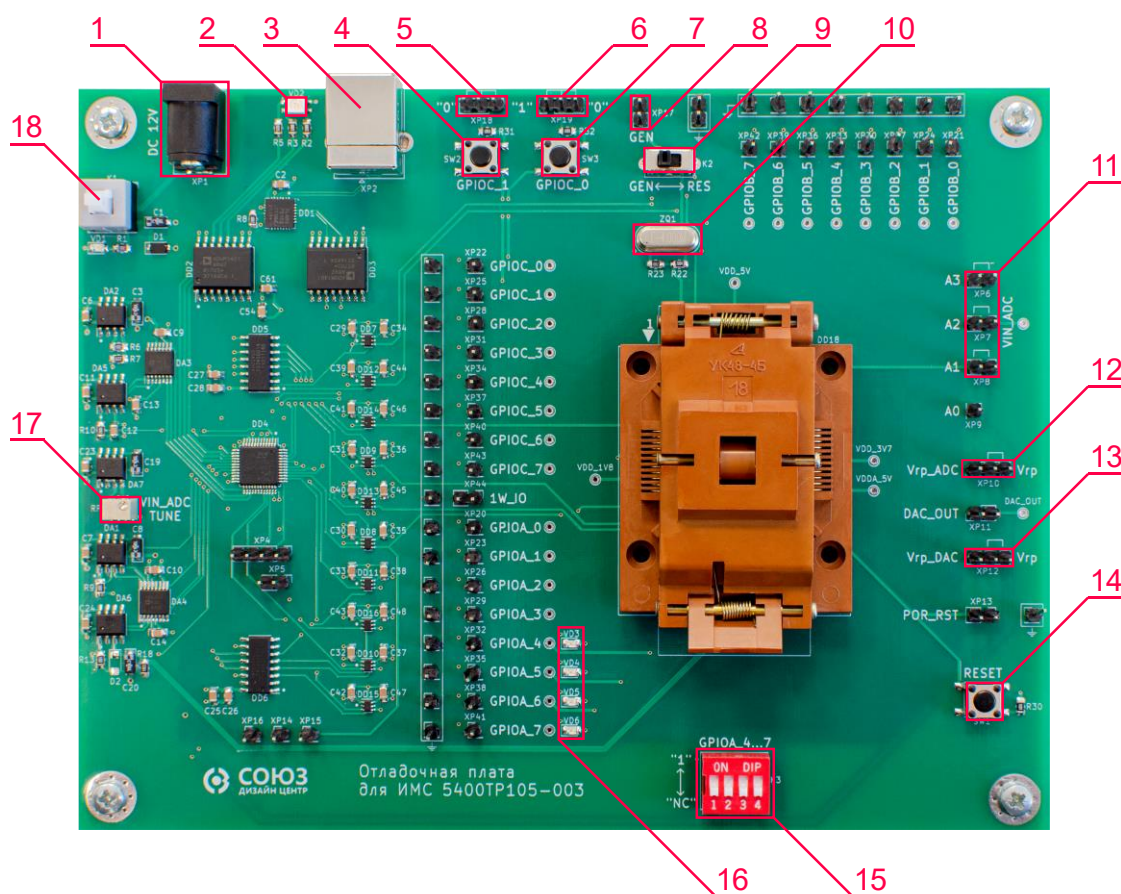


Рисунок 1. Отладочная плата КФЦС.441461.196

- 1 – разъем подключения блока питания;
- 2 – индикация подключения платы к ПК;
- 3 – разъем для подключения платы к ПК (USB Type-B);
- 4-5 – кнопка SW2 (4) для установки лог. «0» или лог. «1» на выводе GPIOC_1 в зависимости от установленного джампера (5);
- 6-7 – кнопка SW3 (7) для установки лог. «0» или лог. «1» на выводе GPIOC_0 в зависимости от установленного джампера (6);
- 8 – выводы для подачи внешней тактовой частоты;
- 9 – движковый переключатель K2 для выбора способа тактирования. В положении «GEN» тактирование микросхемы осуществляется от внешнего генератора, в положении «RES» – от кварцевого резонатора (10);
- 10 – кварцевый резонатор;

11 – джамперы для подачи напряжения на каналы 1-3 мультиплексора перед АЦП. Напряжение настраивается потенциометром (17);

12 – джампер для подачи опорного напряжения АЦП 5,0 В;

13 – джампер для подачи опорного напряжения ЦАП 5,0 В;

14 – кнопка RESET для сброса программы;

15 – DIP-переключатель. Каждый переключатель в положении «1» устанавливает лог. «1» на выводах GPIOA_4 – GPIOA_7 соответственно. В положении «NC» выводы не подключены;

16 – VD3-VD6 – светодиоды для контроля логического уровня на выводах GPIOA_4 – GPIOA_7 соответственно;

17 – потенциометр для настройки напряжения для последующей подачи на входы мультиплексора АЦП (11);

18 – кнопка включения питания отладочной платы.

Электрическая схема отладочной платы КФЦС.441461.196

Таблица 1. Перечень элементов отладочной платы (Рисунок 2)

Обозначение элемента	Название/Тип	Номинал
C1, C3, C8, C19, C20	Конденсатор танталовый SMD A, 16 В, 10%	10 мкФ
C2, C4, C9, C10, C13, C14, C16...C18, C26, C28...C50, C52...C54, C57, C59, C62, C66, C67	Конденсатор керамический 0805, X7R, 50 В, 10%	0,1 мкФ
C5, C15	Конденсатор керамический 0805, X7R, 50 В, 10%	4,7 мкФ
C6, C7, C11, C12, C21, C23...C25, C27, C51, C55, C61, C63	Конденсатор керамический 0805, X7R, 50 В, 10%	1 мкФ
C22, C58	Конденсатор керамический 0805, X7R, 50 В, 10%	10 нФ
C56	Конденсатор керамический 0805, X7R, 50 В, 10%	150 пФ
C60	Конденсатор керамический 0805, X7R, 50 В, 10%	20 нФ
C64, C65	Конденсатор керамический 0805, X7R, 50 В, 10%	64 пФ
D1	Диод SMF12A	—
D2	Диод SMF9.0A	—
D3...D26	Диод BAT54S	—
DA1, DA2, DA5...DA7	Микросхема MIC39102YM	—
DA3, DA4	Микросхема AD5262	—
DA8	Микросхема STG3157CTR	—
DD1	Микросхема CP2102N	—
DD2, DD3	Микросхема ADUM1401ARWZ	—
DD4	Микросхема STM32F302CBT	—
DD5, DD6	Микросхема 74HC595D	—
DD7...DD16	Микросхема SN74LVC1T45DCK	—
DD17	Микросхема 24AA02-OT	—
DD18	Контактирующее устройство УК48-4Б	—
K1	Кнопка KLS7-P8.5x8.5-1	—
K2	Переключатель KLS7-SS03-12D02	—
K3	Переключатель DS1040-04RN	—
R1, R6, R23	Резистор 0805, 5%	2 кОм
R2, R3,	Резистор 0805, 5%	330 Ом
R4, R11, R20, R21, R24	Резистор 0805, 5%	4,7 кОм
R5, R26...R29	Резистор 0805, 5%	510 Ом
R7	Резистор 0805, 5%	1,2 кОм
R8...R10, R12...R17, R25	Резистор 0805, 5%	10 кОм
R18	Резистор 0805, 5%	1 кОм

R19, R33...R56	Резистор 0805, 5%	100 Ом
R22	Резистор 0805, 5%	100 МОм
R30...R32	Резистор 0805, 5%	1 кОм
RP1	Подстроечный резистор	10 кОм
SW1...SW3	Кнопка KLS7-TS6604	–
VD1, VD3...VD6	Светодиод FYLS-0805	–
VD2	Светодиод TO-3228BC-MRPBGHF	–
XP1	Гнездо питания DS-261A	–
XP2	Разъем USB тип В розетка	–
XP4...XP44	Вилка штыревая 2.54мм	–
ZQ1	Кварцевый резонатор HC-49SM	4 МГц

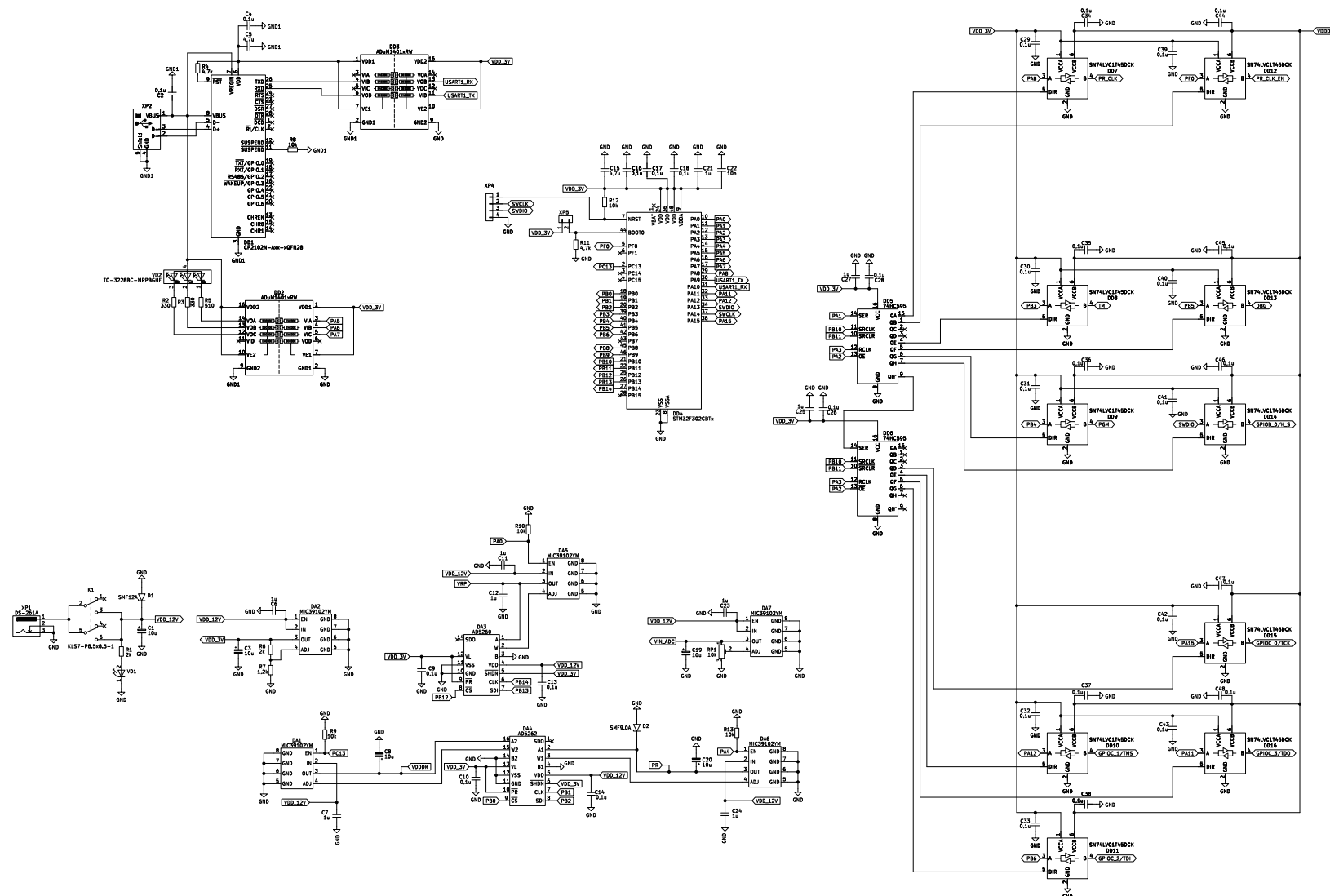


Рисунок 2. Электрическая схема отладочной платы КФЦС.441461.196

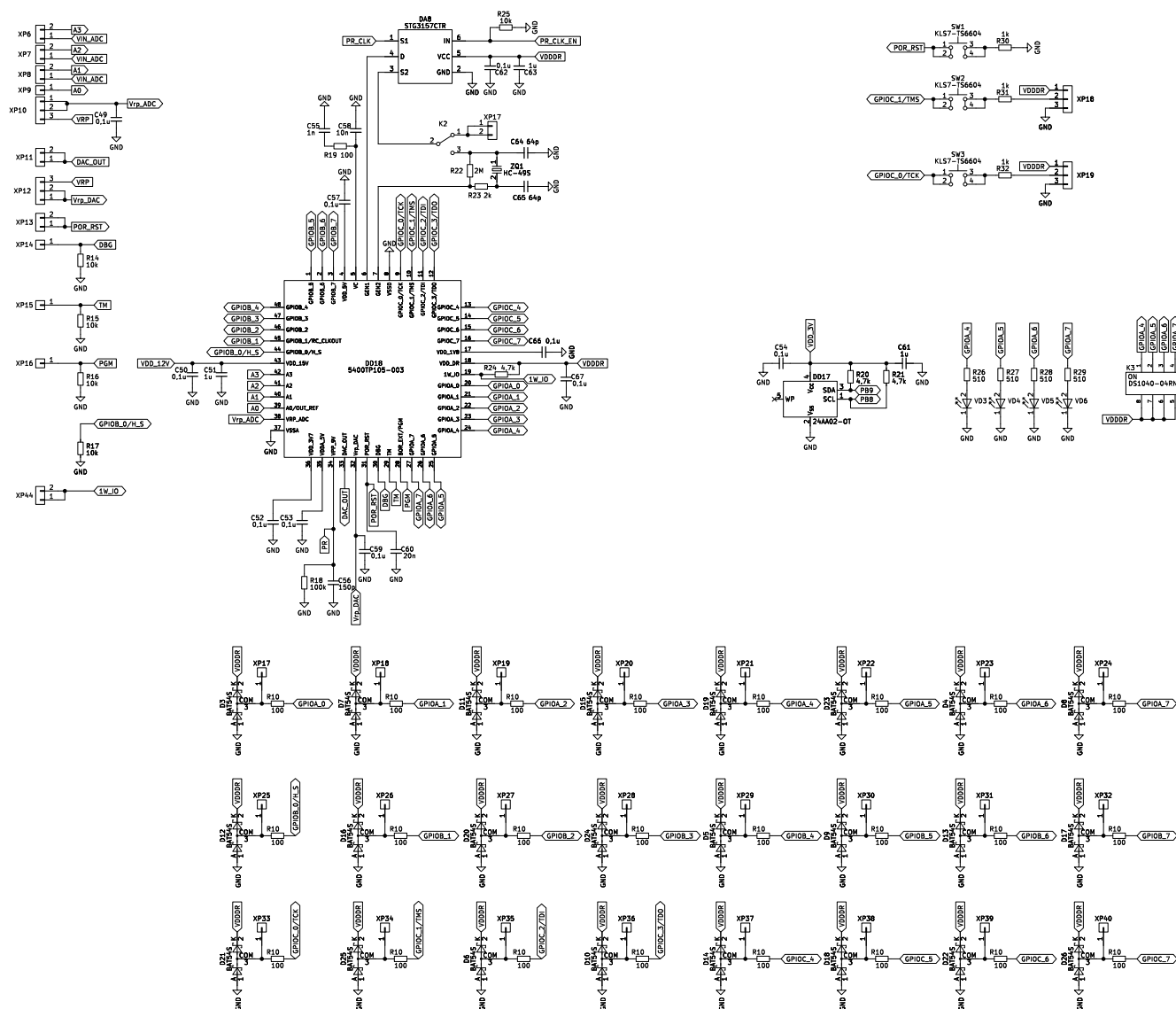


Рисунок 3. Электрическая схема отладочной платы КФС.441461.196 (продолжение)

Установка и настройка Keil IDE

Перед началом работы необходимо загрузить архивы DCSProg6 и projects_keil с сайта компании <https://dcsoyuz.ru> (раздел «Программное обеспечение») и извлечь данные на персональный компьютер. Доступ к разделу «Программное обеспечение» предоставляется по запросу на электронную почту support@dcsoyuz.ru.

Для программирования микросхемы 5400TP105-003 потребуется программа DCSProg6. Создание конфигурационной последовательности выполняется в ПО Keil MVision C51.

Скачать IDE Keil MVision C51 можно с официального сайта разработчика <https://www.keil.com/download/product/>



Рисунок 4. Утилита Keil MVision C51

1. Установить скачанную программу, используя настройки по умолчанию.
2. Открыть проект Example.uvproj.

Для открытия проекта в Keil MVision C51 необходимо выбрать меню «Project» → «Open Project».

Путь к файлу проекта в архиве projects_keil «...\\Projects\\Example\\Example.uvproj» (Рисунок 5).

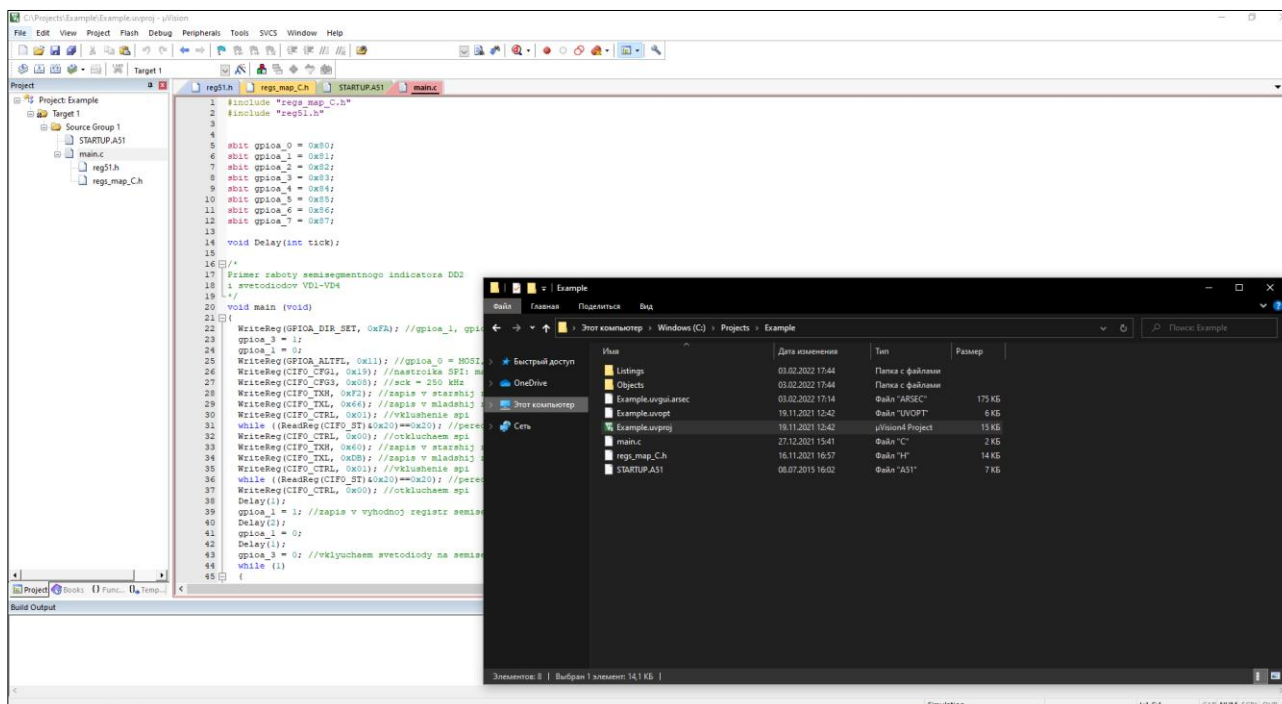


Рисунок 5. Выбор проекта в MVision

3. В настройках проекта («Alt+F7») во вкладке «Target» установить размеры памяти ОЗУ и ПЗУ (Рисунок 6):

- Off-chip Code memory: Start – 0x0000; Size – 0x1000;
- Off-chip Xdata memory: Start – 0x0000; Size – 0x1100.

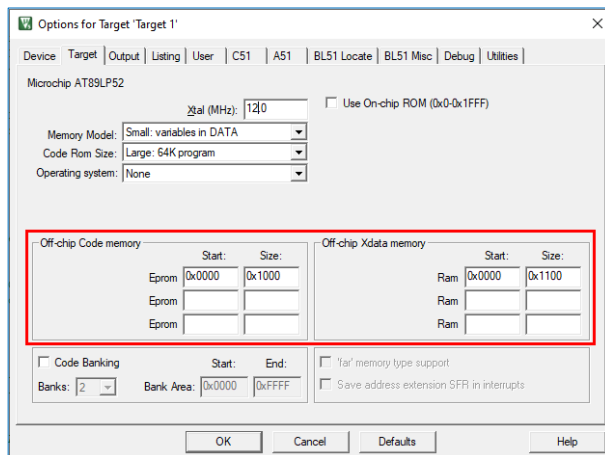


Рисунок 6. Установка размеров памяти ОЗУ и ПЗУ в настройках проекта

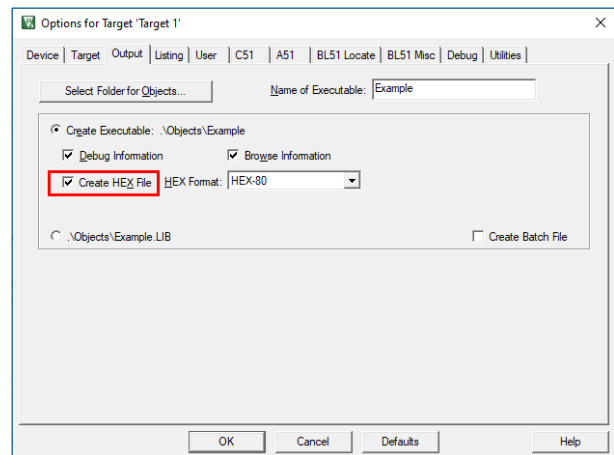


Рисунок 7. Настройка проекта для создания файла с расширением hex

4. В настройках проекта во вкладке «Output» установить галочку «Create HEX File» для создания файла с расширением hex. Нажать «ОК» (Рисунок 7).

Установка и настройка ПО для работы с микросхемой 5400TP105-003 завершена.

В демонстрационной версии проекта Example.uvproj реализована программа с примером работы светодиодов и выводов микросхемы GPIOA_4 – GPIOA_7.

Создание и обновление файла конфигурации (hex-файл)

Для компиляции тестового проекта выполните команду «Project» → «Build Target» или клавиша *F7*. Ошибки компиляции будут выведены в окне Build Output.

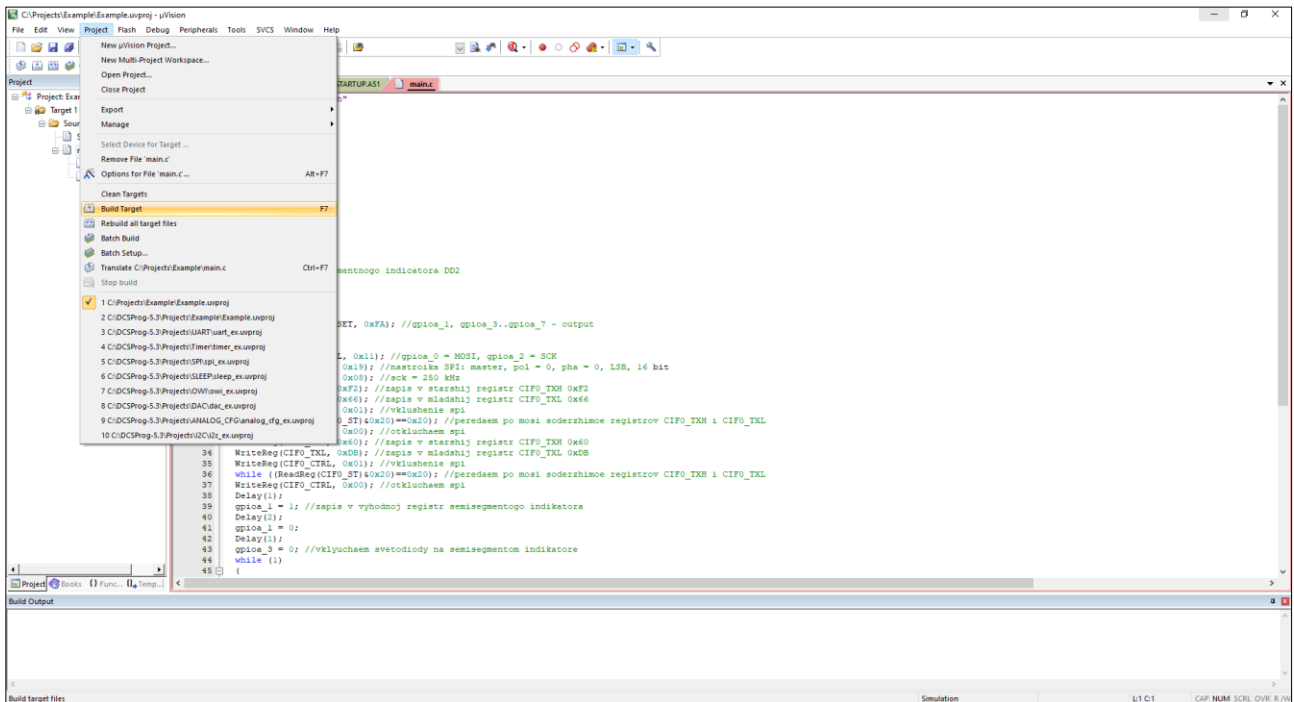


Рисунок 8. Компиляция проекта

Для создания или обновления существующего hex-файла выполните команду «Project» → «Rebuild all target files».

В результате выполнения команды будет сформирован файл с расширением hex.

Путь к hex-файлу «...\\Projects\\Example\\Objects\\Example.hex».

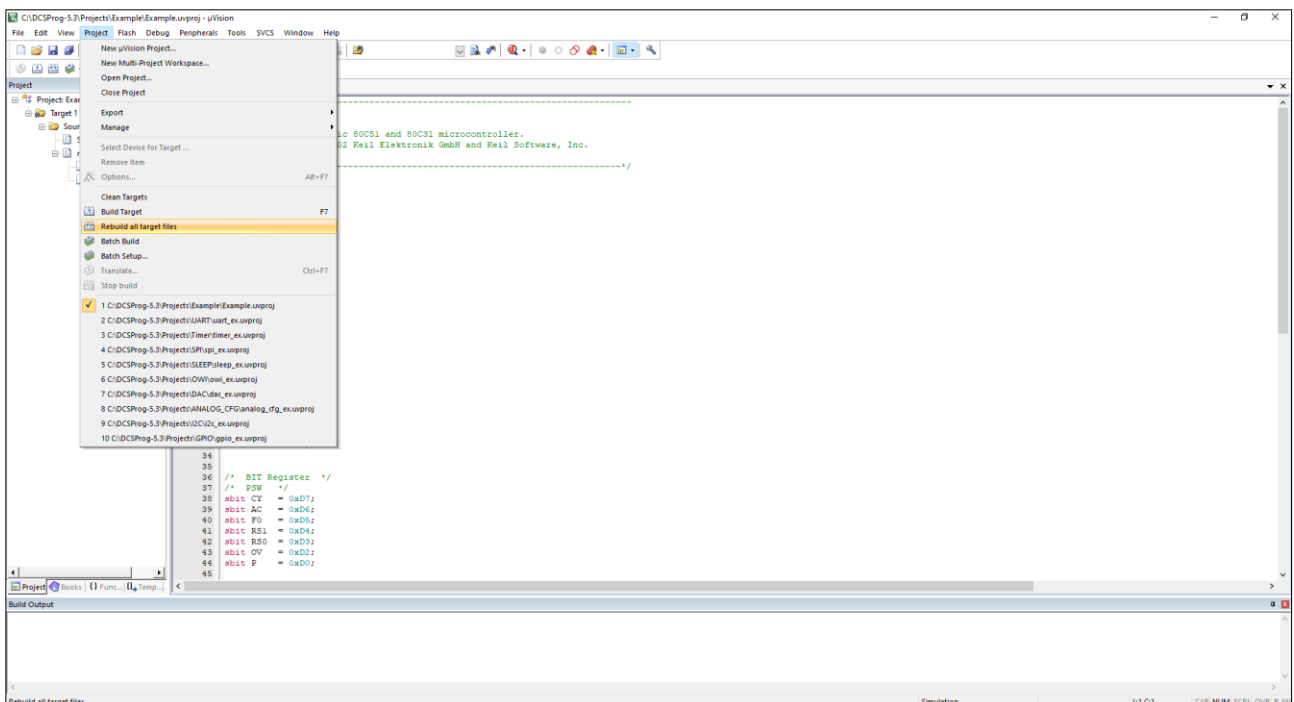


Рисунок 9. Команда для формирования и обновления hex-файла

Отладка в Keil IDE

Настройка отладчика

1. Поместить файл *105-003.dll* из архива DCSProg6 в каталог *C:/Keil_v5/C51/BIN/*.
2. В каталоге *C:/Keil_v5/* в файле *TOOLS.ini* в разделе C51 найти максимальный номер для TDRV и добавить строку *TDRV*=BIN\105-003.dll ("5400TP105-003")*, где * – следующий номер после максимального. Далее необходимо сохранить изменения и закрыть файл.

```

1  [uv2]
2  ORGANIZATION="HP Inc."
3  NAME="HP Inc.", "13"
4  EMAIL="13"
5  TOOL_VARIANT=mdk_lite
6  [c51]
7  PATH="C:\Keil_v5\C51\"
8  VERSION=V9.60
9  BOOK0=HLP\Release_Notes.htm("Release_Notes",GEN)
10 BOOK1=HLP\C51TOOLS.chm("Complete User's Guide Selection",C)
11 TDRV0=BIN\MON51.DLL ("Keil Monitor-51 Driver")
12 TDRV1=BIN\ISD51.DLL ("Keil ISD51 In-System Debugger")
13 TDRV2=BIN\MON390.DLL ("MON390: Dallas Contiguous Mode")
14 TDRV3=BIN\LPC2EMP.DLL ("LPC900 EFM Emulator/Programmer")
15 TDRV4=BIN\UL2UPSD.DLL ("ST-upSD ULINK Driver")
16 TDRV5=BIN\UL2XC800.DLL ("Infineon XC800 ULINK Driver")
17 TDRV6=BIN\MONADI.DLL ("ADI Monitor Driver")
18 TDRV7=BIN\DAS2XC800.DLL ("Infineon DAS Client for XC800")
19 TDRV8=BIN\UL2LPC9.DLL ("NXP LPC95x ULINK Driver")
20 TDRV9=BIN\JLinkEFM8.dll ("J-Link / J-Trace EFM8 Driver")
21 TDRV10=BIN\JLinkIS2083.dll ("J-Link / J-Trace IS2083 Driver")
22 TDRV11=BIN\Nuvoton_8051_Keil_uVision_Driver.dll ("Nuvoton 8051 Keil Driver")
23 TDRV12=BIN\105-003.dll ("5400TP105-003")
24 RTOS0=Dummy.DLL ("Dummy")
25 RTOS1=RTXTINY.DLL ("RTX-51 Tiny")
26 RTOS2=RTX51.DLL ("RTX-51 Full")
27

```

3. Запустить Keil IDE. В настройках проекта (комбинация *Alt+F7* или кнопка «Options for Target» на панели инструментов) во вкладке «Debug» поставить переключатель «Use» и выбрать «Octopus Driver». Ниже установить «Load Application at Startup».

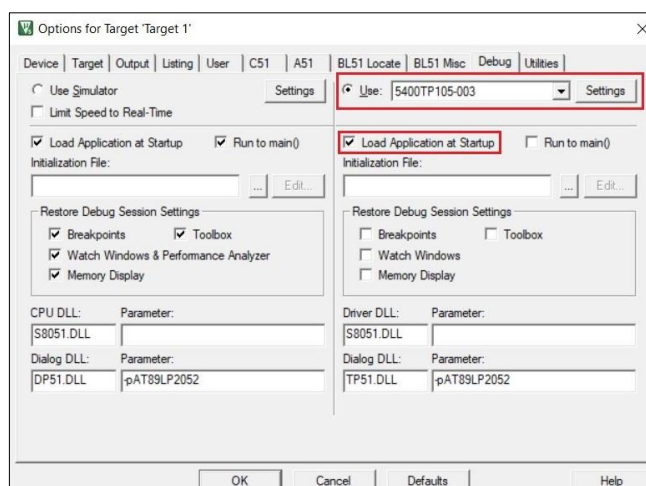


Рисунок 10. Настройка Keil IDE

4. Непосредственно перед запуском отладчика необходимо создать выходной hex-файл отлаживаемой программы («Project» → «Rebuild all target files»), подключить программатор DCSProg6 в USB-порт компьютера, подать на плату питание и частоту.

Начало работы с отладкой

Запуск отладчика производится комбинацией клавиш Ctrl+F5. При успешном подключении программатора к отладчику в окне логов будет выведено следующее сообщение:

```
Инициализация программатора  
Программатор подключен  
Запись байта 0x7 по адресу 0x3200 во внешнюю RAM  
Из внешней RAM по адресу 0x3200 считан байт 0x7  
Reset
```

Рисунок 11. Сообщение при успешном подключении программатора

При неуспешном подключении программатора отладчик будет принудительно отключен, либо в окне логов будет выведено следующее сообщение:

```
Инициализация программатора  
Программатор подключен  
Запись байта 0x7 по адресу 0x3200 во внешнюю RAM  
Из внешней RAM по адресу 0x3200 считан байт 0xc8  
Reset  
Не получен корректный ответ от микросхемы при Reset, проверьте питание и тактирование
```

Рисунок 12. Сообщение при неуспешном подключении программатора

В этом случае необходимо проверить питание и тактирование микросхемы и заново включить отладчик.

Возможности отладчика

После загрузки программы в микросхему, открывается окно с ассемблерным кодом. Если окно не открылось, его можно открыть вручную нажатием «View» → «Disassembly Window». В этом окне можно просматривать ассемблерный код, сопоставленный строкам исходного кода на языке Си (Рисунок 13). Если окно ассемблерных инструкций изначально отображается по-другому, это можно изменить одиночным нажатием ПКМ по окну ассемблерных инструкций и выбором режима отображения «Mixed Mode» (Рисунок 14).

18:			gpioa_4 ^= 1;
C:0x08AC	A284	MOV	C, gpioa_4(0x80.4)
C:0x08AE	B3	CPL	C
C:0x08AF	9284	MOV	gpioa_4(0x80.4), C

Рисунок 13. Пример ассемблерного кода в удобочитаемом виде

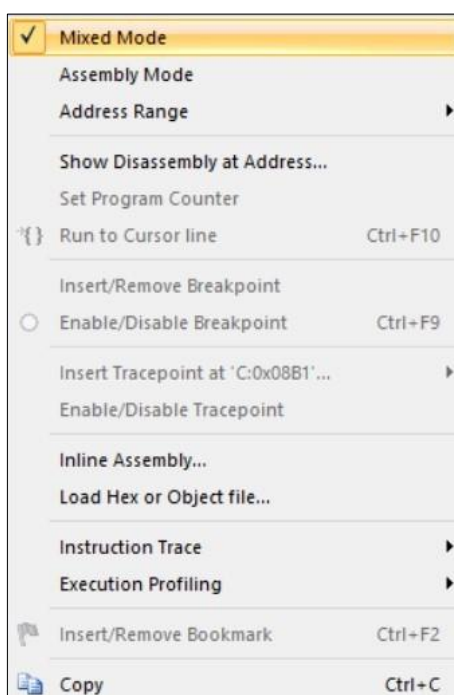







Рисунок 14. Выбор режима отображения ассемблерных инструкций вместе с Си кодом

Примечание! При заполнении окна ассемблерных инструкций отладчик не считывает инструкции из микросхемы, а анализирует входной hex-файл. Верификация записанного кода происходит только при команде «Flash Download» (см. пункт Команда «Flash Download»).

В отладочном окне в левом верхнем углу находится основная панель инструментов для управления выполнением программы:


-  Кнопка «RST» сбрасывает выполнение всех инструкций и возвращает PC («Program Counter») в начало программы;
-  Кнопка «Run» запускает выполнение программы с текущего PC. Выполнение программы остановится по окончании или при встрече брейкпоинта;
-  Кнопка «Stop» прерывает выполнение программы. PC остаётся на последней выполненной инструкции;
-  Кнопка «Step» выполняет одну инструкцию программы, на которую указывает PC;
-  Кнопка «Run to Cursor Line» выполняет программу до той строки, на которую указывает курсор в окне ассемблерного кода (желтая метка) или в окне кода на Си (синяя стрелка);

21:			gpioa_6 ^= 1;
C:0x0816	A286	MOV	C, gpioa_6(0x80.6)
C:0x0818	B3	CPL	C
C:0x0819	9286	MOV	gpioa_6(0x80.6), C

Рисунок 15. Пример поставленного курсора для команды «Run to Cursor Line» в окне ассемблерного кода

▶	21		gpioa_6 ^= 1;
---	----	--	---------------

Рисунок 16. Пример поставленного курсора для команды «Run to Cursor Line» в окне кода на Си

-  Кнопка «Step Over» пропускает одну инструкцию программы, на которую указывает PC.

20:			Delay(10000);
⇒C:0x0813	12084E	LCALL	L?0009(C:084E)

Рисунок 17. Пример вызова функции в ассемблерном окне

При нажатии ПКМ на строку в ассемблерном окне (Рисунок 18) или в окне Си кода (Рисунок 19) можно задать РС на эту инструкцию, используя команду «Set Program Counter». При нажатии кнопки «Run» исполнение программы начнется с инструкции, на которую указывает установленный «РС».

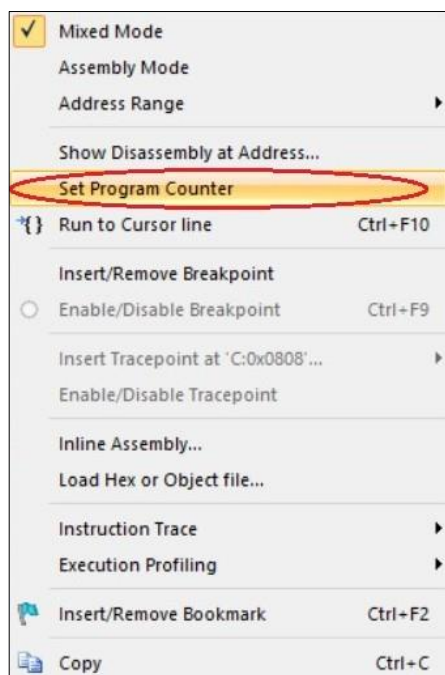


Рисунок 18. Установка РС в окне ассемблерного кода

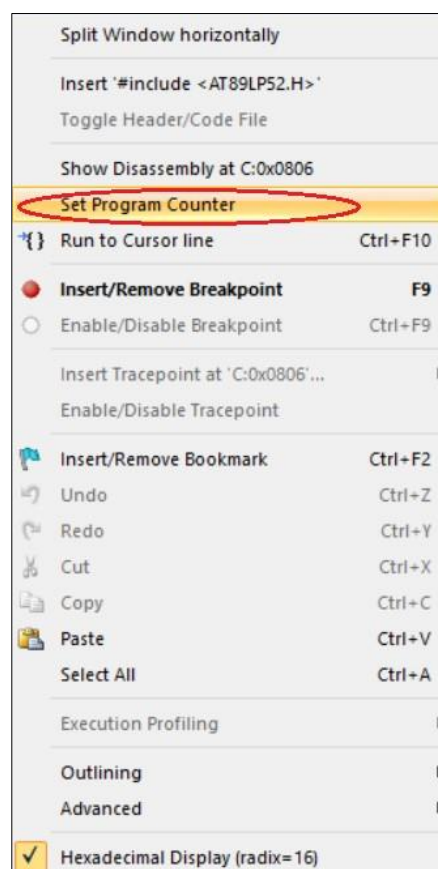


Рисунок 19. Установка РС в окне Си кода

В левой части отладочного окна можно отслеживать состояние SFR регистров ядра 8051 (Рисунок 20). При остановке исполнения программы (принудительно или по брейкпоинту) происходит чтение значений и обновление окна SFR регистров. В остановленном состоянии можно изменять значения SFR регистров, используя данное окно.

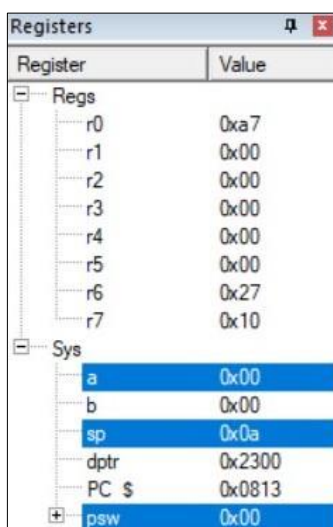
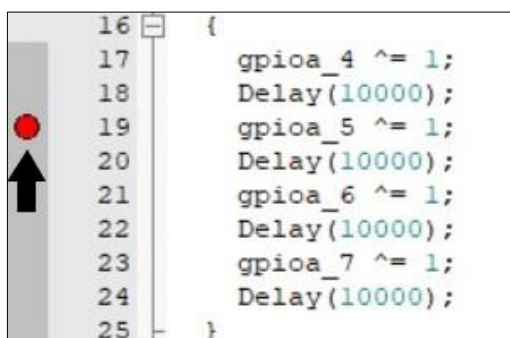


Рисунок 20. Окно SFR регистров ядра 8051

Использование точек останова

Отладчик поддерживает до четырёх брейкпоинтов одновременно. Чтобы установить/удалить брейкпоинт необходимо в области кода слева от нужной строки нажать ЛКМ.



Примечание! Команда «Run to Cursor Line» использует брейкпоинт, поэтому при использовании этой команды необходимо убедиться, что в данный момент установлено максимум три брейкпоинта.

Примечание! Команда «Step Over» при прохождении функции использует брейкпоинт, поэтому при использовании этой команды необходимо убедиться, что в данный момент установлено максимум три брейкпоинта.

Просмотр состояния переменных

Открыть окно с отслеживанием переменных можно по нажатию «View» → «Watch Windows» → «Watch 1». Снизу отладочного окна появится вкладка «Watch 1» (Рисунок 21). В данной вкладке можно ввести название переменной в столбец «Name», состояние которой необходимо отслеживать.

При успешном добавлении переменной будет отображено её значение и тип, при неуспешном добавлении в столбце «Value» будет отображаться «<cannot evaluate>».

Считывание переменной происходит при остановке программы.

Watch 1		
Name	Value	Type
<Enter expression>		

Рисунок 21. Окно для отслеживания переменных

Примечание! При помощи этого окна можно изменять состояние переменных при остановленной программе. Если переменная хранится во внутренней ОЗУ, то состояние меняется, если переменная хранится во внешней ОЗУ (ключевое слово «xdata»), то её состояние не изменяется.

Команда «Flash Download»

Для загрузки программы из Keil IDE в микросхему необходимо в настройках проекта (Alt+F7) во вкладке «Utilities» в меню настроек «Flash Download» указать следующие настройки:

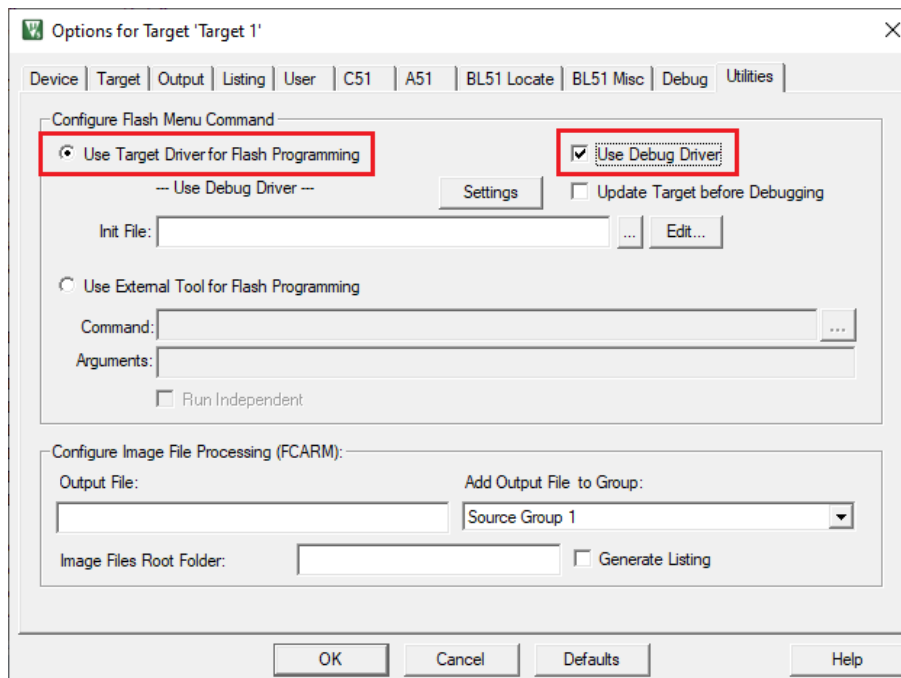


Рисунок 22. Настройки проекта для включения возможности загружать программу сразу из Keil IDE

На панели инструментов окна Keil IDE есть кнопка (Рисунок 23), которая очищает память программ, заполняя ее нулями, затем программа загружается в микросхему, в конце программа считывается из микросхемы и по байтам сравнивается с тем, что должно быть загружено.

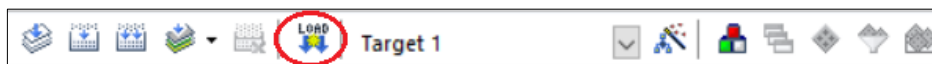


Рисунок 23. Кнопка «Flash Download» на панели инструментов Keil IDE

При успешном подключении к плате, успешной записи и верификации загруженной в микросхему программы в окне логов будет выведено сообщение о том, что все байты загружаемой программы совпали.

```
Программатор подключен
Запись байта 0x7 по адресу 0x3200 во внешнюю RAM
Из внешней RAM по адресу 0x3200 считан байт 0x7
Flash Erase Done.
Flash Write Done: 249 bytes programmed.
Flash Verify Done: 249 bytes verified.
Flash Load finished at 10:39:44
```

Рисунок 24. Информационное сообщение в окне логов о том, что все 249 байт программы были загружены и чтение кода программы это подтвердило

При обнаружении несовпадений байтов программы об этом также будет сообщено в окне логов. В таком случае рекомендуется заново подать питание и тактирование микросхемы и попробовать снова загрузить программу.

Примечание! Кнопка «Flash Download» лишь так называется в контексте Keil IDE. В контексте микросхемы программа будет загружаться в RAM_ROM область в SOFT режиме, так как флеш-память у микросхемы 105-003 отсутствует как таковая.

Примечание! После успешной загрузки и верификации программы для её запуска необходимо нажать на отладочной плате кнопку «RESET».

Примечание! Такая загрузка происходит относительно долго, т.к. перед записью и верификацией программы происходит полное затирание RAM_ROM области (все 4096 байт заполняются нулями).

Установка и настройка Eclipse

Программа для отладки разработана в виде набора плагинов для Eclipse. На текущий момент получить плагины можно только в составе сборки DCS-Eclipse.

В качестве компилятора для 5400TP105-003 предлагается использовать SDCC ([SDCC manual](#)).

Среда разработана и протестирована на операционной системе «Ubuntu 22.04». Поддержка других операционных систем будет добавлена позднее.

Настройка окружения

Предварительно требуется установить SDCC:

Установка Ubuntu 22.04

Загрузить архив eclipse_ubuntu с сайта в разделе «Программное обеспечение» и извлечь данные на персональный компьютер

Установить java-17 командой `sudo apt install openjdk-17-jdk`

Установить компилятор для 8051 командой `sudo apt install sdcc`

Перейти в папку eclipse и из терминала запустить IDE командой `sudo ./eclipse`

Установка Windows

Загрузить архив eclipse_win32 с сайта в разделе «Программное обеспечение» Установить eclipse из архива eclipse_win32.

Установить java-17 с сайта разработчика.

https://download.oracle.com/java/17/archive/jdk-17.0.9_windows-x64_bin.msi

или

https://download.oracle.com/java/17/archive/jdk-17.0.9_windows-x64_bin.exe

Установить SDCC. Скачать файл sdcc-winX, где X – разрядность операционной системы.

<https://sourceforge.net/projects/sdcc/files/>

Перейти в папку Eclipse и запустить eclipse.exe.

Примечание! В случае, если на компьютере были установлены другие версии Java и у вас не запускается программа ниже представлены возможные пути решения проблемы.

1. Папка, в которую установлена Java-17 находится в системных переменных среды операционной системы. Путь к файлу java.exe должен находиться в переменной Path. Для того, чтобы в этом убедиться нажмите ПКМ на «Пуск», выберите «Система» – «Дополнительные параметры системы»–«Переменные среды», выбрать переменную Path и нажать «Изменить».

Если в переменной Path находится одновременно несколько путей до java.exe, то рекомендуется удалить т.к может быть использована некорректная версия java для запуска программы.

2. Системная переменная JAVA_HOME должна ссылаться на папку с установленной java (например C:\Program Files\Java\jdk-17\bin\ или C:\Program Files\Common Files\Oracle\Java\javapath). Если переменная отсутствует, то ее необходимо создать.

Если программа после всех действий не запускается и ваш путь к java содержит C:\Program Files\Common Files\Oracle\Java\javapath, то рекомендуется заменить на явный путь к java C:\Program Files\Java\jdk-17\bin\

Создание проекта

Одновременно допускается использовать только один проект. Если требуется использовать другой проект, остальные необходимо закрыть.

Для создания первого проекта необходимо нажать на строку «Create a new C or C++ project» или «Create new Project...» → «C Project».

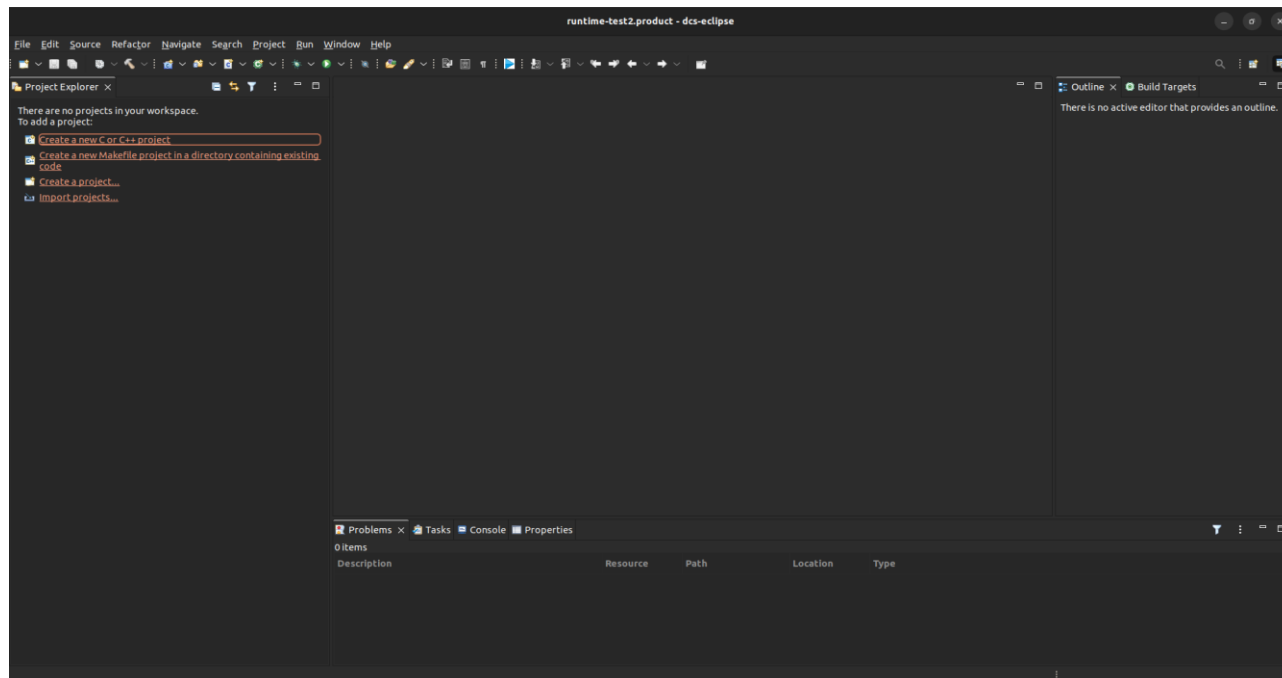


Рисунок 25. Создание проекта

В качестве категории проекта выбрать «C Project» (Рисунок 26). Нажать «Next».

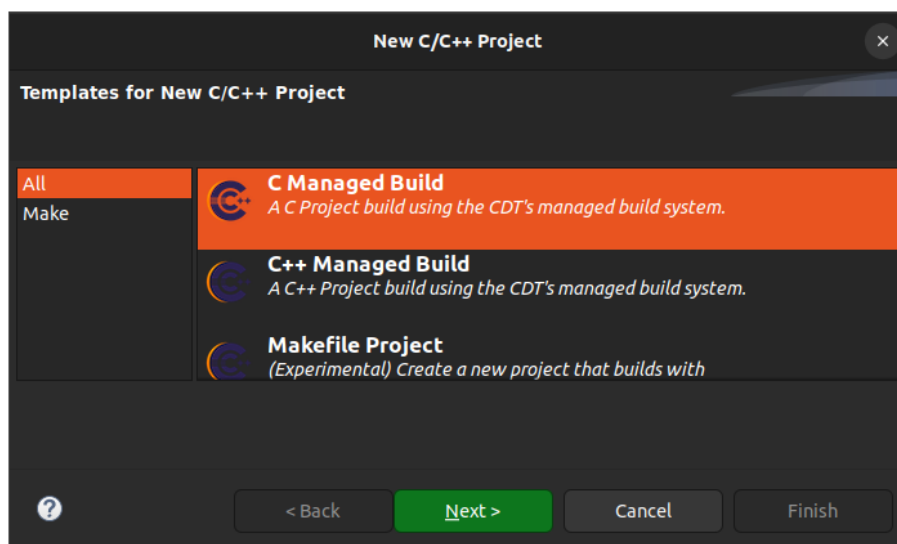


Рисунок 26. Выбор категории проекта

Выбрать тип проекта «Executable/5400TP105-003 C Project» (Рисунок 27).

В окне «Toolchains» выбрать «SDCC Tool Chain», если вариантов более одного, и он не выбран по умолчанию.

Указать название проекта.

Убедиться, что проект создается в ожидаемой директории. При необходимости убрать галочку «*Use default location*» и изменить путь.

Нажать «*Next*».

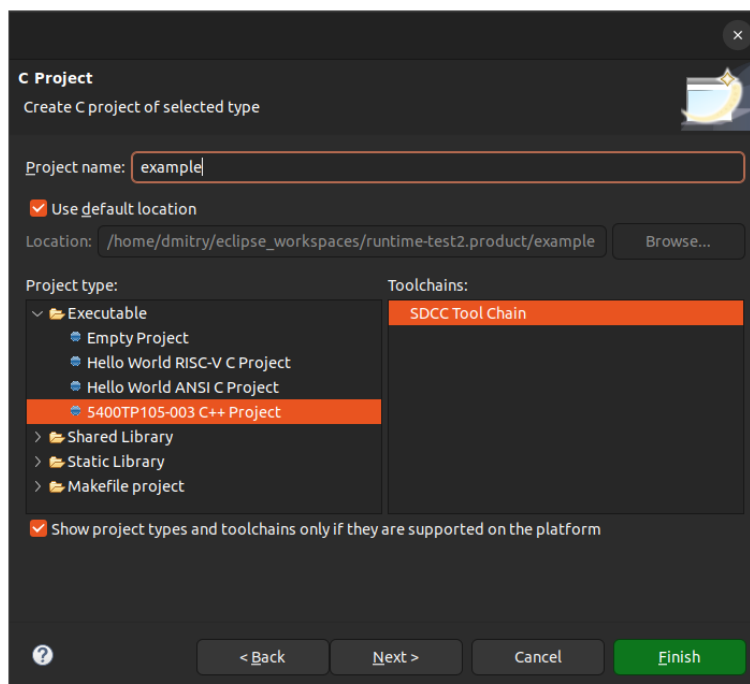


Рисунок 27. Выбор типа проекта

При необходимости изменить свойства проекта (Рисунок 28).

Желательно не трогать поля «*Source*» и «*Linker other options*».

Нажать «*Next*».

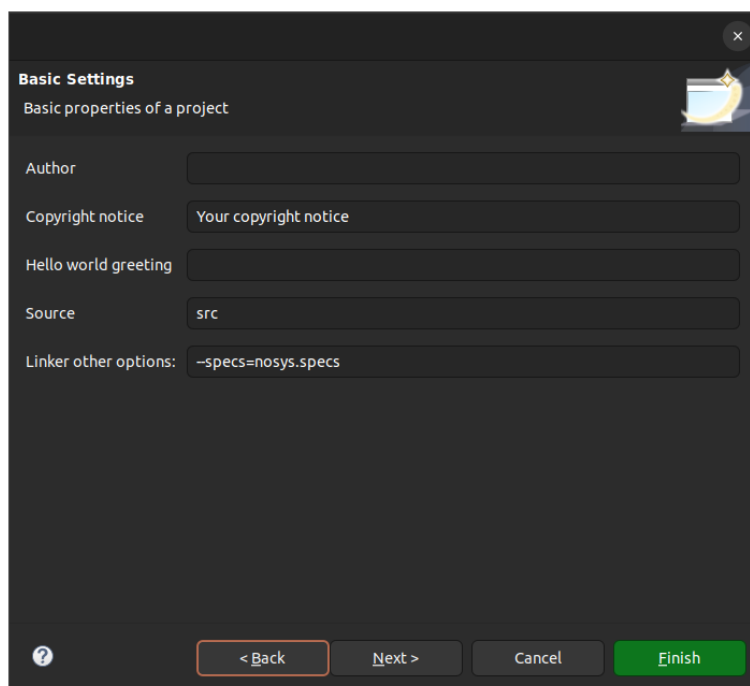


Рисунок 28. Свойства проекта

В следующем окне ничего не изменять (Рисунок 29). Нажать кнопку «*Finish*».

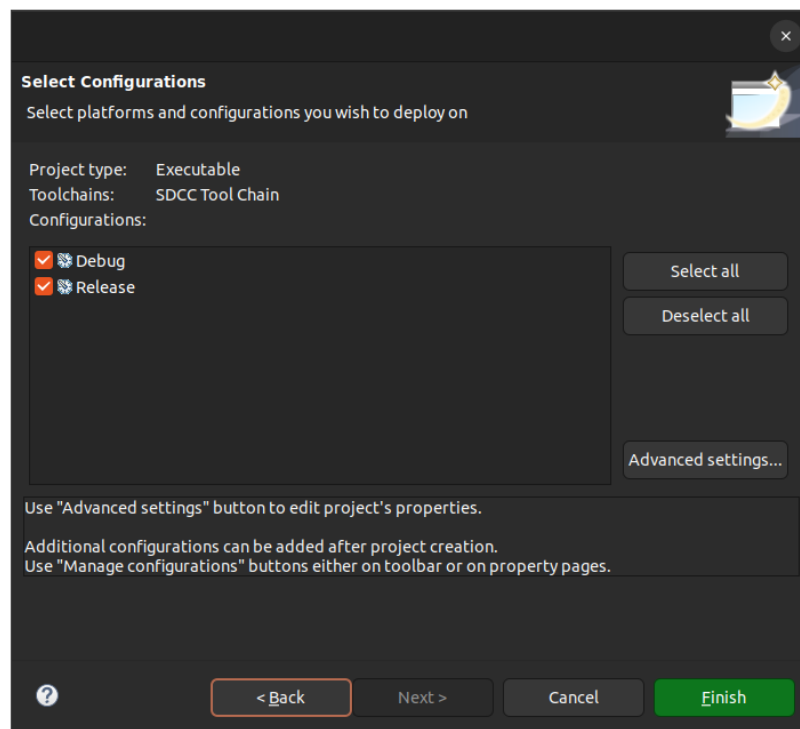


Рисунок 29. Выбор конфигурации проекта

Возможно, IDE предложит сменить перспективу (Рисунок 30). Необходимо согласиться нажатием кнопки «*Open Perspective*».

Перспективу также можно сменить в правом верхнем углу рабочего окна (кнопка «*Open Perspective*»).

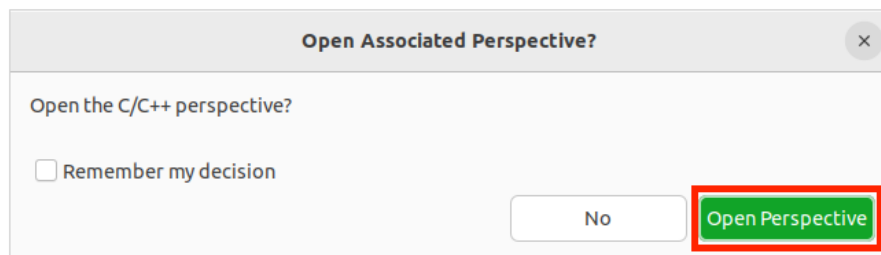


Рисунок 30. Открытие перспективы

Настройка проекта

Для первой сборки проекта нужно выполнить ряд действий:

Настроить проект для использования компилятора. Для этого нужно нажать ПКМ на проект и выбрать «*Properties*» в появившемся контекстном меню.

В категории C/C++ Build нажать кнопку «*Manage Configurations*», выбрать «*Release*» и нажать «*Set Active*».

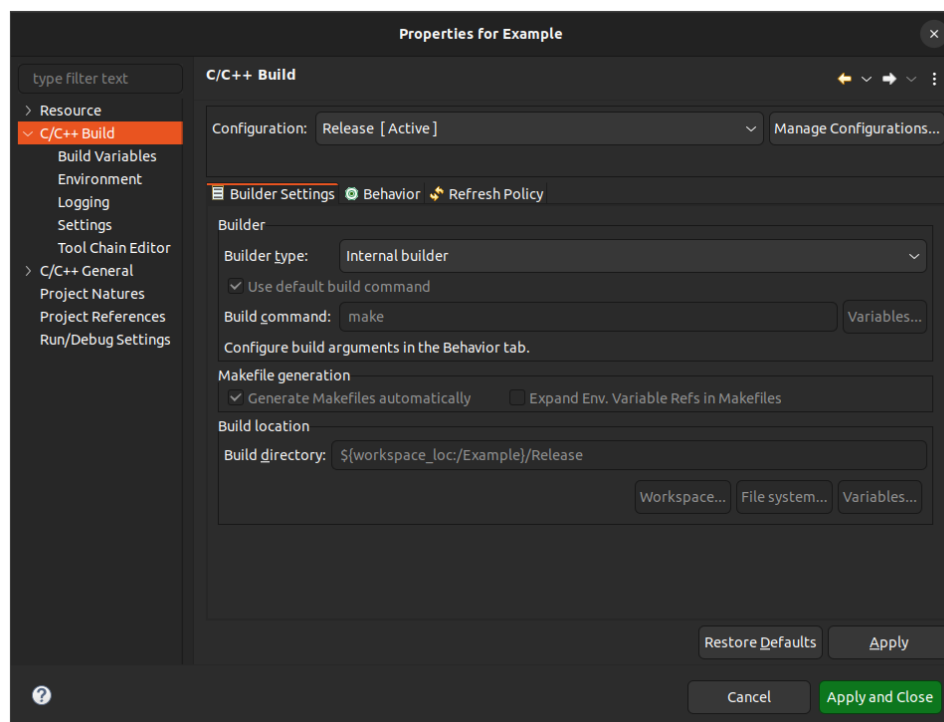


Рисунок 31. Окно настроек проекта

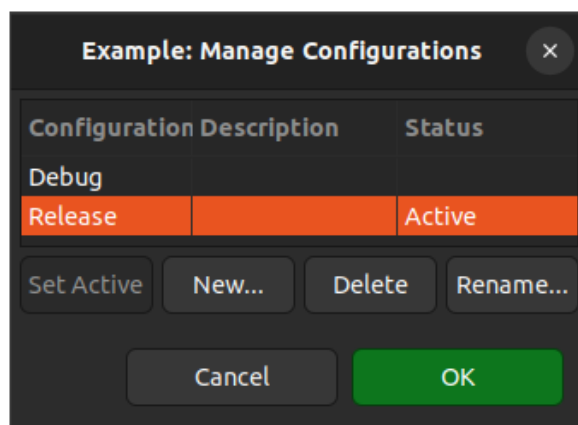


Рисунок 32. Меню управления конфигурацией

В «Configuration» выбрать «Release [Active]» (Рисунок 32).

В категории C/C++ Build выбрать «*Builder type: Internal builder*».

Добавить генерацию hex-файла.

Для этого перейти в подкатегорию «Settings», в окне «Build Artifact» выбрать «Artifact extension: hex».

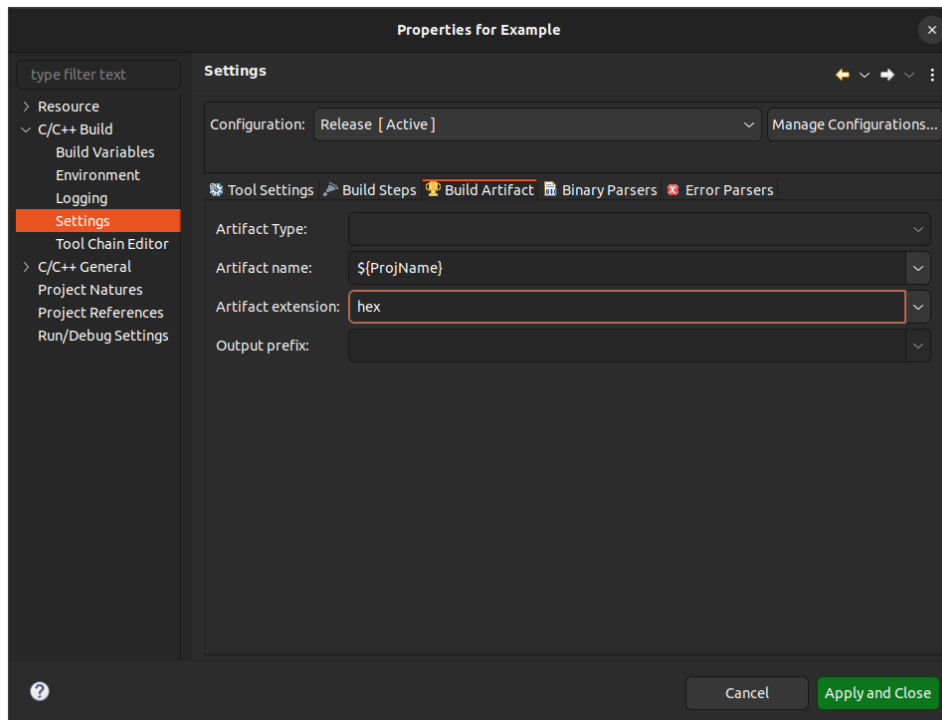


Рисунок 33. Установка генерации hex-файла

Подтвердить изменения нажав «Apply and Close».

Выполнить сборку проекта нажав на молоток на панели инструментов или кнопку «Build all».

Если в консоли возникает ошибка «Error: Program "sdcc" not found in PATH», то необходимо убедиться, что в системной переменной Path среды операционной системы указан путь к sdcc. Для этого, нажмите ПКМ на «Пуск», выберите «Система» – «Дополнительные параметры системы» – «Системные переменные», выбрать переменную Path и нажать «Изменить». Если путь отсутствует, то необходимо нажать кнопку «Создать» и добавить путь (пример C:\Program Files\SDCC\bin).

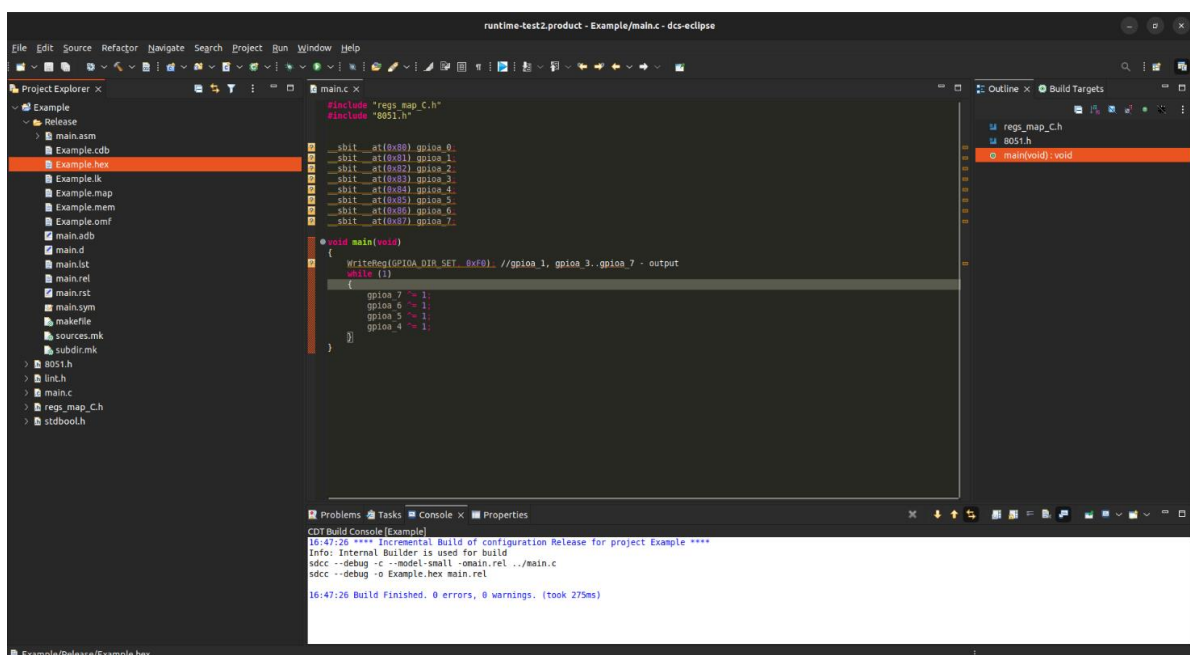


Рисунок 34. Сборка проекта

Удаление/закрытие проекта

Для удаления/закрытия проекта необходимо нажать ПКМ по проекту и в выпадающем меню выбрать пункт «Delete» (Рисунок 35).

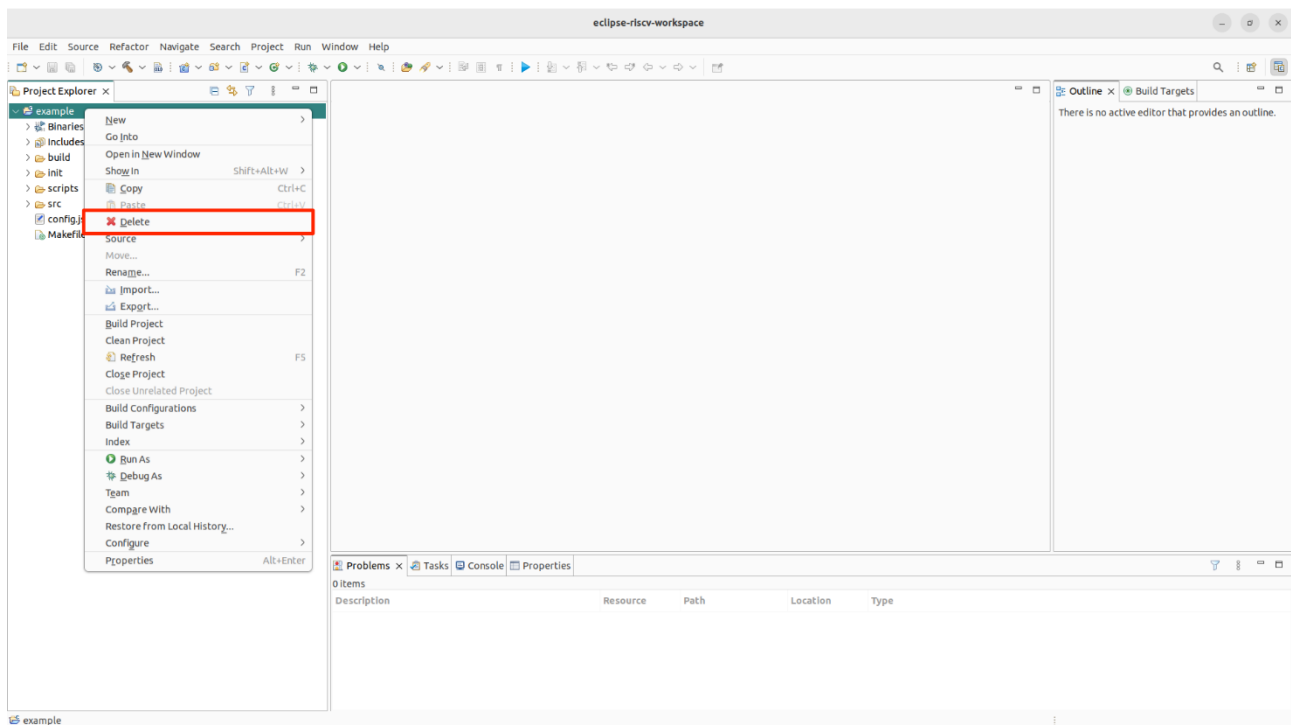


Рисунок 35. Удаление/закрытие проекта

Для удаления нужно нажать на checkbox «Delete project contents on disk (cannot be undone)». Для закрытия проекта галочку ставить не нужно. Подтвердить нажатием кнопки «OK» (Рисунок 36).

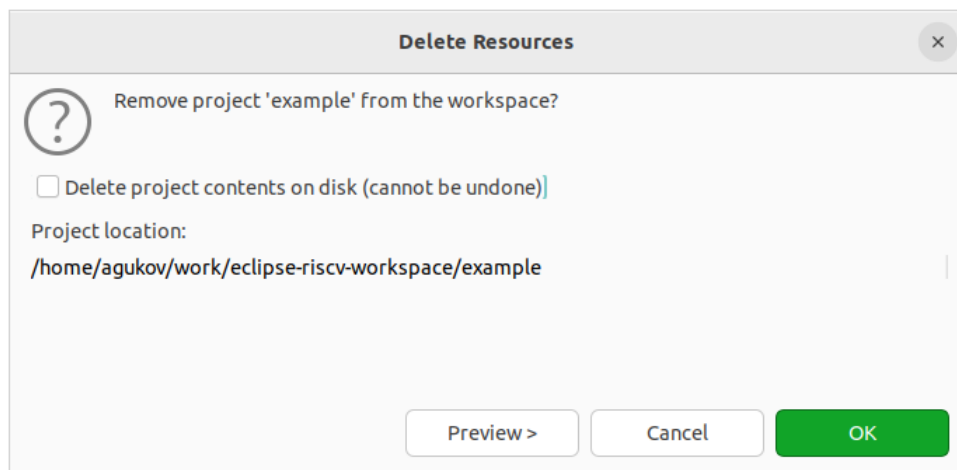


Рисунок 36. Подтверждение удаления/закрытия

Описание проекта

На рисунке ниже показано дерево проекта.

8051.h — библиотека внутренних регистров ядра;

regs_map_C.h — библиотека внешних регистров ядра;

lint.h и **stdbool.h** — файлы для более корректной работы системы проверки синтаксиса;

main.c — функция main.

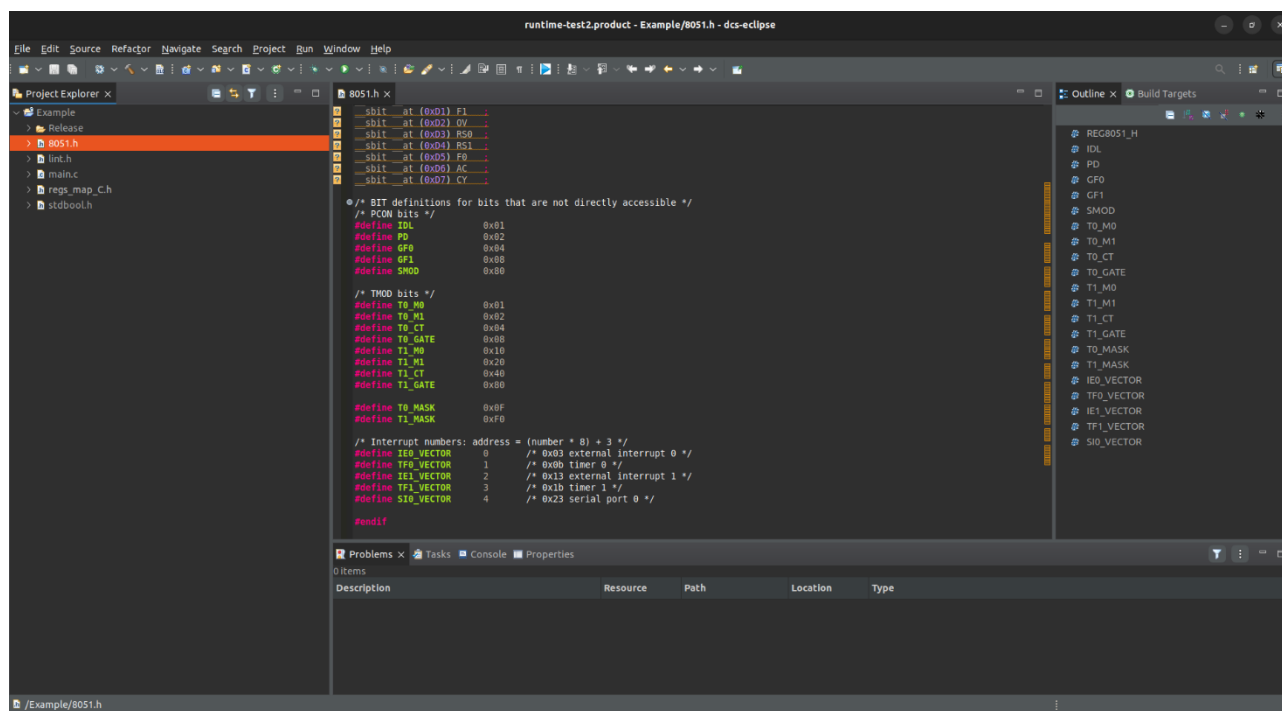


Рисунок 37. Дерево проекта

Руководство по отладке в Eclipse

Для отладки нужно выполнить подключение платы к usb порту устройства. Далее необходимо включить питание и подать частоту.

Примечание (Ubuntu 22.04): в некоторых случаях отладка может не работать из-за системы выделения прав на устройстве. В этом случае следует запустить Eclipse через терминал с дополнительными правами с помощью команды `sudo ./eclipse`

После сборки проекта можно записать программу в микроконтроллер и начать отладку. Для этого следует нажать ПКМ на проекте и в контекстном меню выбрать:

Debug As → Debug Configurations...

Создать новую конфигурацию путём двойного нажатия на «5400TP105-003 Debug» и нажать «Debug».

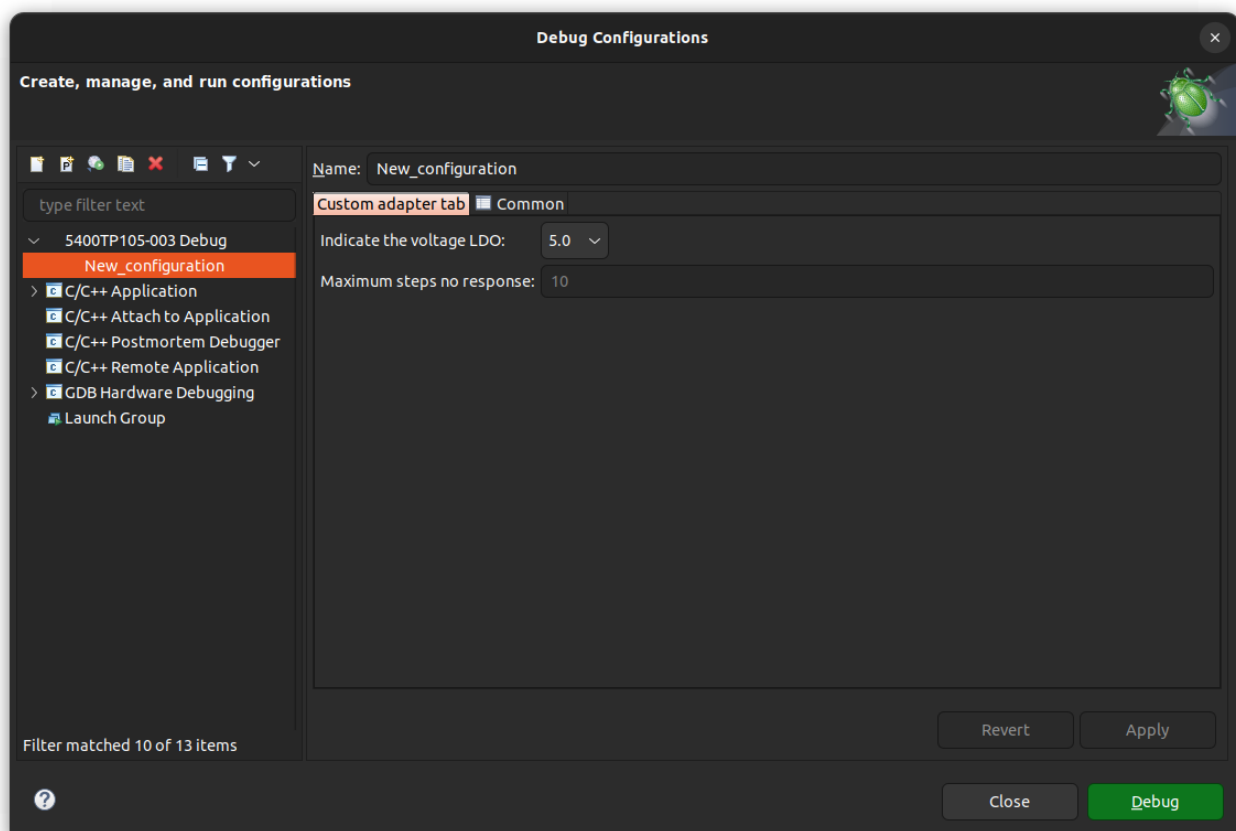


Рисунок 38. Окно запуска отладки

После нажатия кнопки «Debug» будет произведено программирование микросхемы в «SOFT» режим, и она будет переведена в режим отладки.

Примечание: при возникновении ошибок следует проверить и переподключить плату.

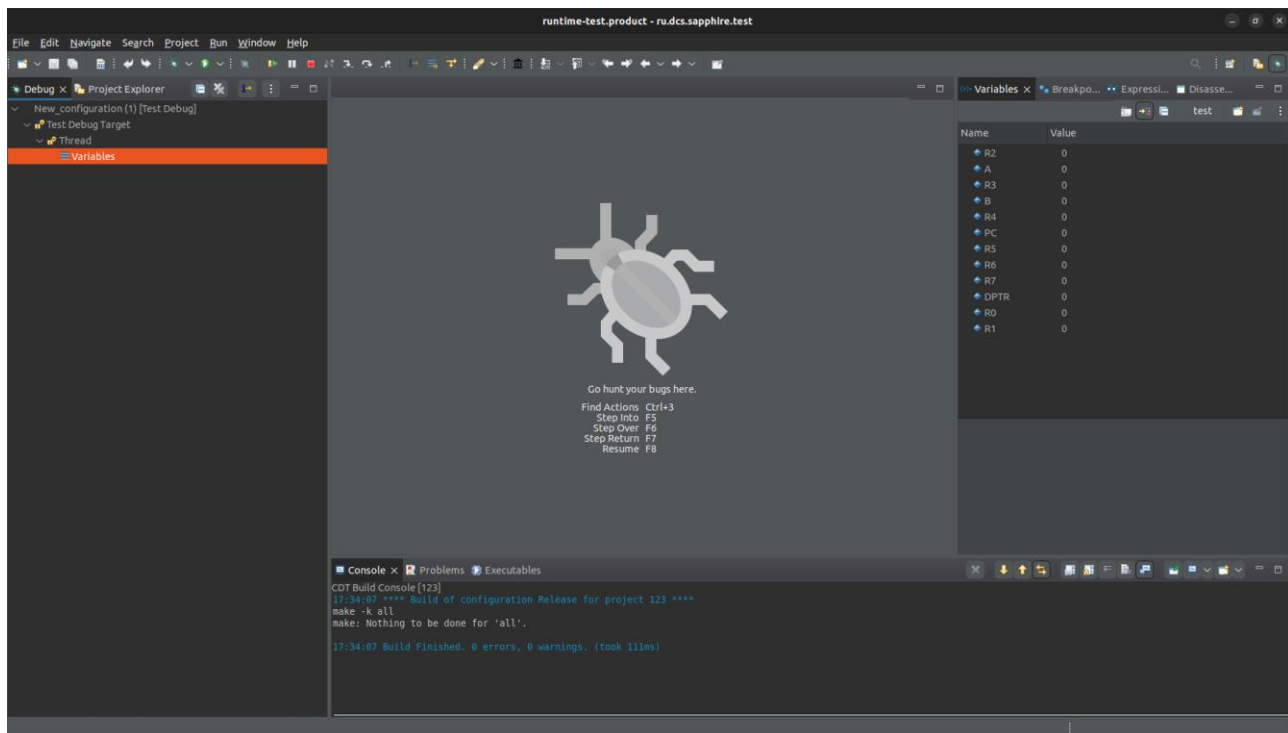


Рисунок 39. Внешний вид при отладке проекта

Справа можно увидеть 4 окна:

Variables: информация об основных переменных и регистрах;

Breakpoints: информация об установленных точках останова;

Expressions: окно с возможностью слежения за выбранными переменными;

Disassembler: окно дизассемблирования.

На панели инструментов есть 3 основных кнопки:

Resume — запуск работы программы;

Suspend — приостановка работы программы;

Terminate — завершение работы программы.

Примечание: при первой загрузке в память выполняется команда Reset для обнуления записанной программы и установки PC в начало.

Точки останова можно добавить с помощью двойного нажатия левой кнопкой мыши в файле с программой.

Примечание: для микроконтроллера 5400TP105-003 доступны только 4 одновременно установленных точки останова.

Подготовка к работе с отладочным комплектом

1. Подключить блок питания к плате
2. Соединить плату с ПК с помощью USB-кабеля

Зайти в диспетчер устройств

Для Windows 10 – нажмите по иконке поиска в панели задач и наберите «диспетчер устройств» в поле ввода, а после того, как нужный элемент будет найден, нажмите по нему ЛКМ для открытия.

Для Windows 7 и 8 – откройте пуск и введите в поле поиска фразу «диспетчер устройств», а после того, как нужный элемент будет найден, нажмите по нему ЛКМ для открытия.

Во вкладке «Порты (COM и LPT)» можно посмотреть какой COM-порт соответствует отладочной плате. В примере плата подключена к порту COM3.

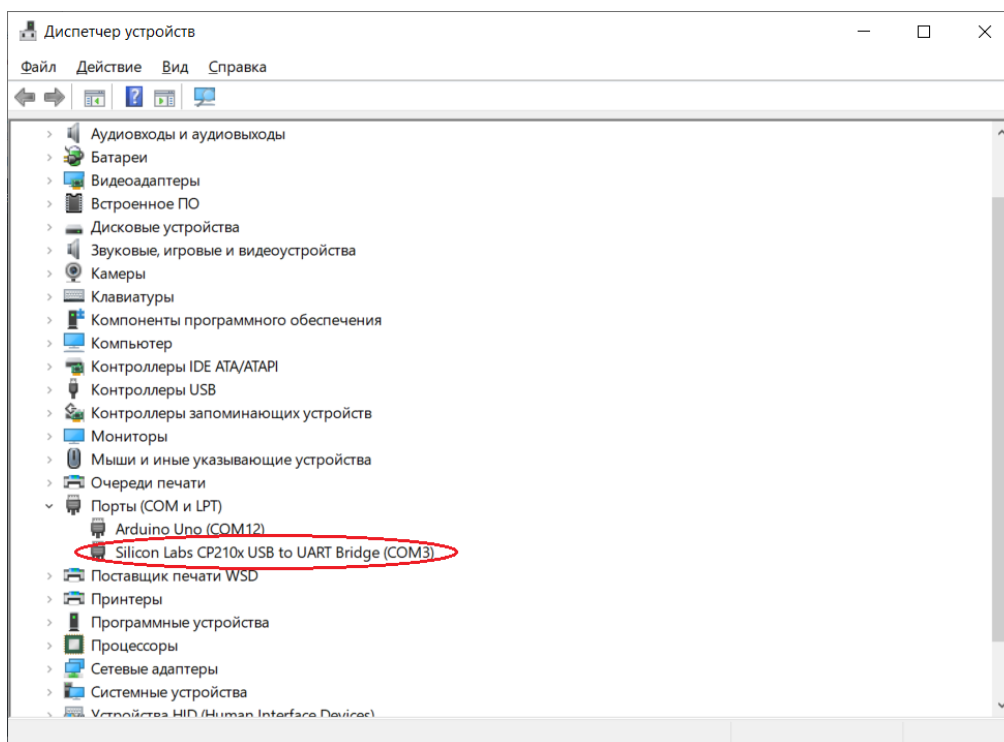


Рисунок 40. Диспетчер устройств

Если компьютер не распознает отладочную плату, то следует установить драйвер CP210x: загрузить архив CP210x_Windows_Drivers с сайта компании <https://dcsoyuz.ru> в разделе «Программное обеспечение», либо с сайта разработчика <https://www.silabs.com/developers/usb-to-uart-bridge-vcp-drivers>.

Режимы работы микросхемы

Микросхема 5400TP105-003 имеет два режима работы:

- режим отладки с возможностью многократного перепрограммирования (режим «SOFT»);
- режим финальной конфигурации с записью в энергонезависимую память (режим «HARD»).

Программирование микросхемы в режиме отладки (режим «SOFT»)

1. Вставьте микросхему в контактирующее устройство на отладочной плате. Направление первого вывода микросхемы указано шелкографией с цифрой «1» над контактирующим устройством (Рисунок 1). Прижмите микросхему крышкой контактирующего устройства.

2. Нажать кнопку включения питания на плате (Рисунок 1, №18).

3. В случае использования внешнего генератора. Перевести переключатель (Рисунок 1, №9) в положение «GEN». Подать прямоугольный сигнал (меандр) от 0 В до 5,0 В с частотой 4 МГц (допустимый диапазон частоты от 500 кГц до 8 МГц) на вывод (Рисунок 1, №8).

В случае использования кварцевого резонатора необходимо перевести переключатель (Рисунок 1, №9) в положение «RES».

4. Запустить приложение DCSProg6.exe из папки DCSProg6.

При подключении платы и включении питания происходит автоматическая идентификация платы в правом нижнем углу программы. Если идентификация не прошла, то приложение опрашивает COM-порт каждые 3 секунды. Без идентификации программирование микросхемы невозможно.

Примечание! Если требуется программирование микросхемы в составе вашего устройства (без отладочной платы КФЦС.441461.196) с помощью программатора DCSProg6, то для завершения идентификации необходимо нажать сочетание клавиш *Ctrl+F12* и в окне выбрать микросхему «5400TP105-003». После этого идентификация будет пройдена и программирование станет доступно (Рисунок 41).

Важно! Если идентификация не проходит, сообщите о проблеме в службу технической поддержки по электронной почте support@dcsoyuz.ru. Проведение идентификации с помощью диалогового окна (сочетание клавиш *Ctrl+F12*) при использовании отладочной платы не является штатным режимом работы.

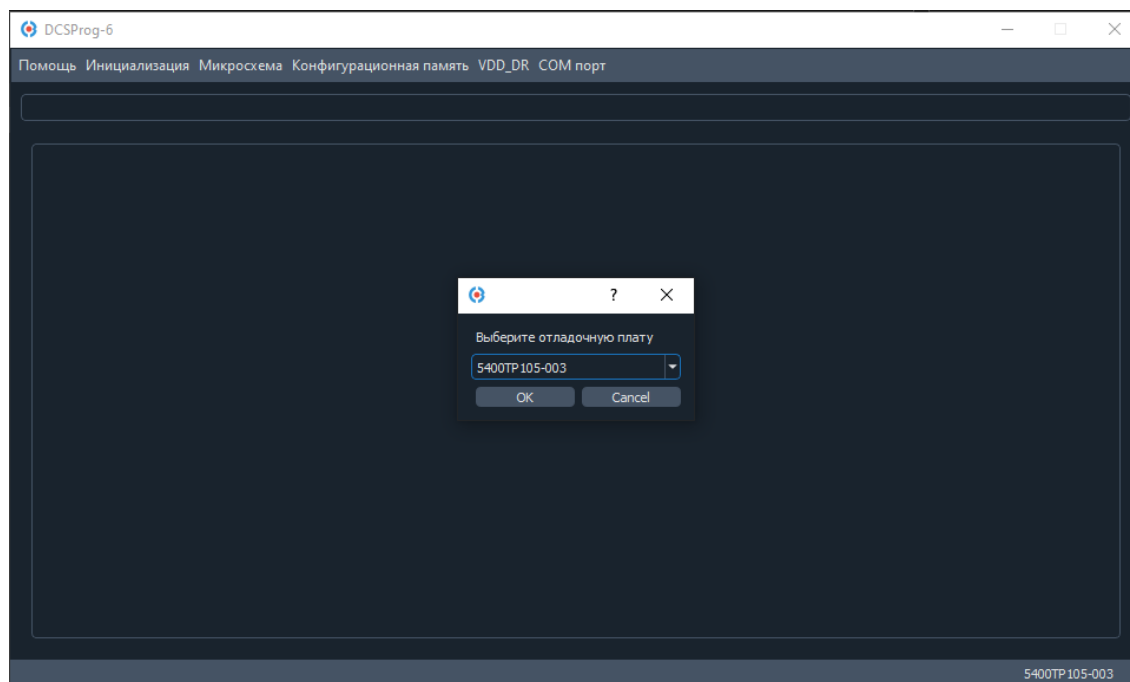


Рисунок 41. Завершение идентификации через диалоговое меню

5. Если подключено больше одного COM-порта к компьютеру, то программа выведет диалоговое окно с выбором COM-порта. В диалоговом окне выберите COM-порт, который соответствует плате, и нажмите «ОК».

Методика определения COM-порта приведена в пункте в пункте «Подготовка к работе с отладочным комплектом» (Рисунок 40). После автоматической идентификации в правом нижнем углу программы появится надпись «5400TP105-003».

6. После идентификации загрузите файл с конфигурацией:

«Микросхема» → «Загрузить hex файл», выбрать ранее скомпилированный hex-файл и нажать кнопку «Открыть» (Рисунок 42) (расположение файла: «...\Projects\Example\Objects»).

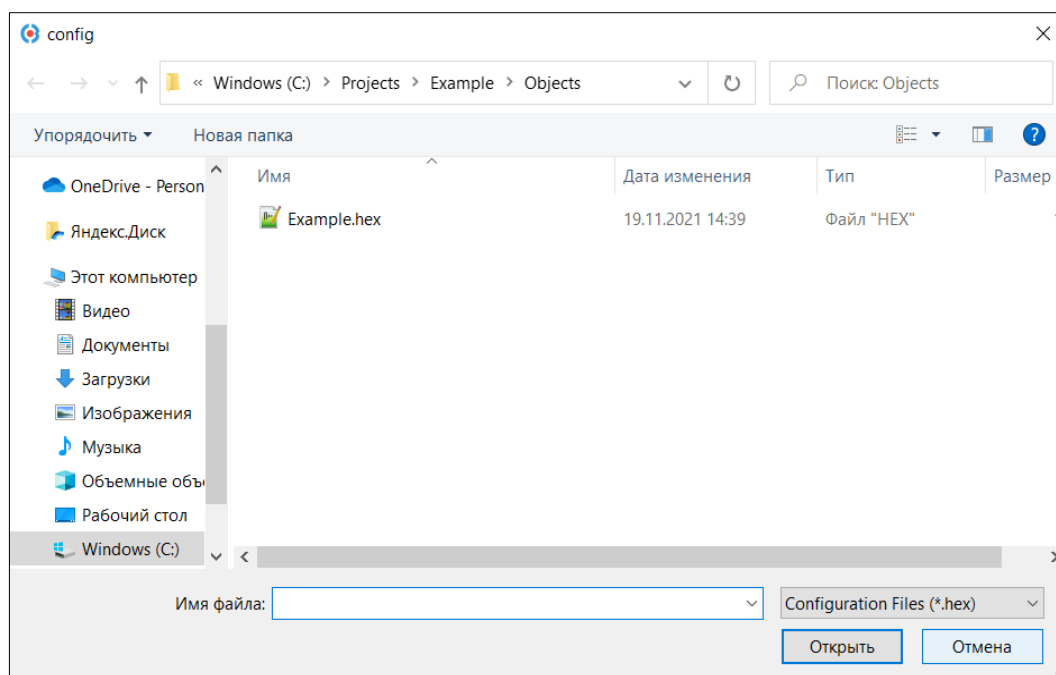


Рисунок 42. Расположение hex-файла

Выбранный hex-файл отобразится в программе DCSProg6 (Рисунок 43);

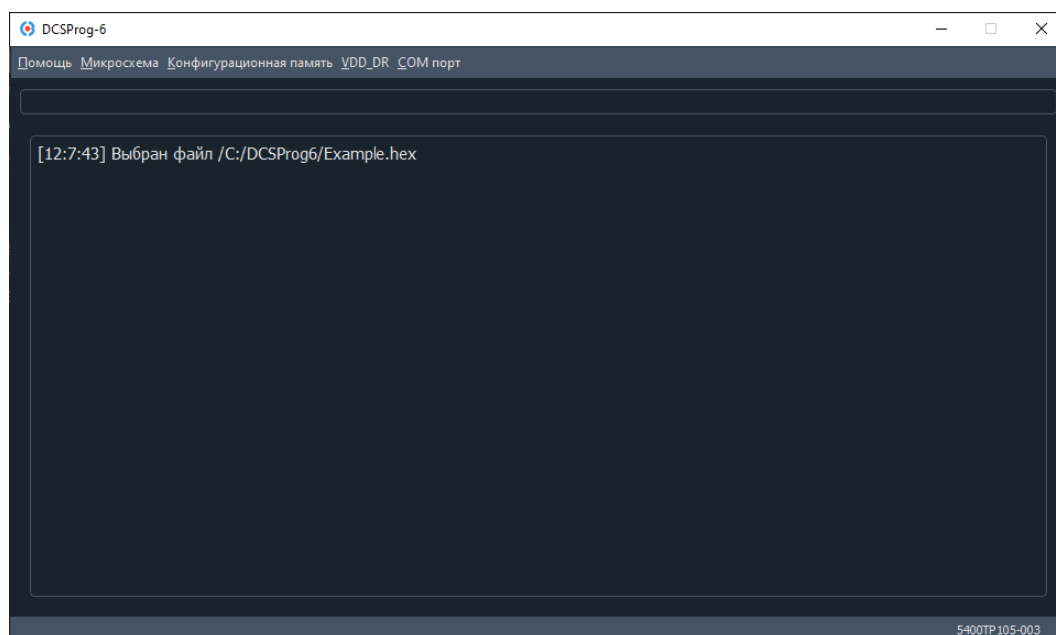


Рисунок 43. Окно программы после загрузки hex-файла

7. Для программирования микросхемы в режиме отладки (режим «SOFT») выполнить:
«Микросхема» → «Прошить ОЗУ» (Рисунок 44);

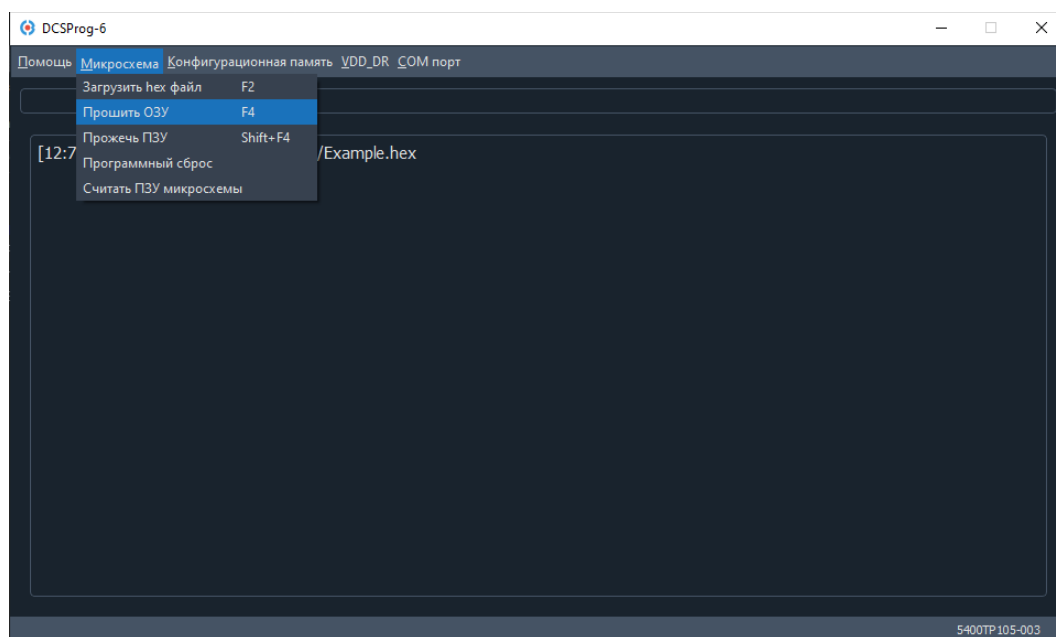


Рисунок 44. Программирование микросхемы в режиме «SOFT»

В случае успешного программирования в окне программы будет выведено «Микросхема запрограммирована в режиме SOFT».

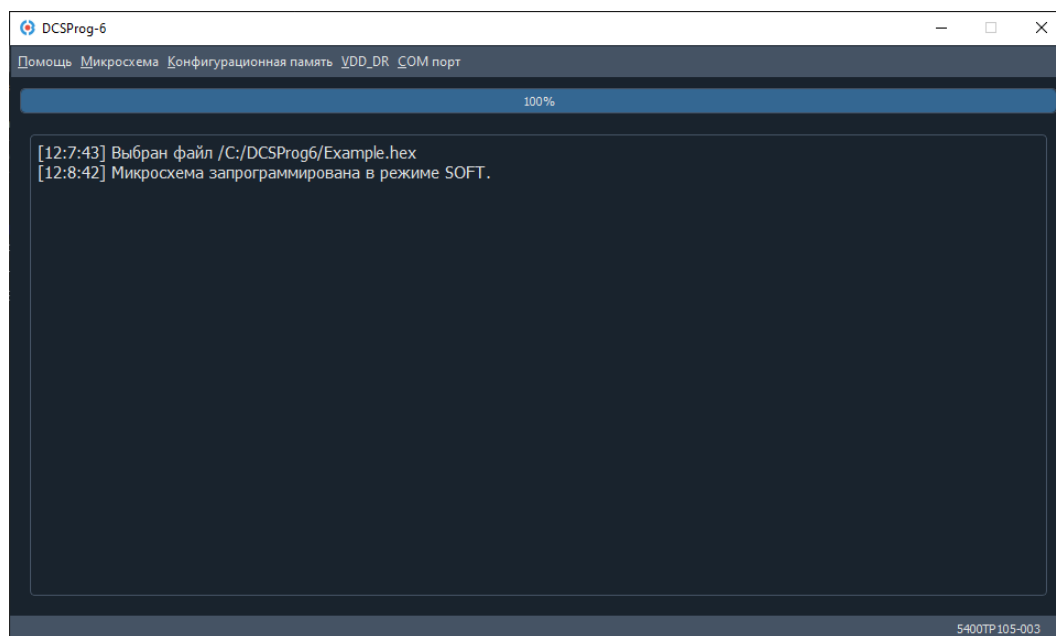


Рисунок 45. Сообщение о успешном программировании в режиме «SOFT»

В случае ошибок при программировании в окне программы будет указан первый адрес, в котором возникла ошибка записи.

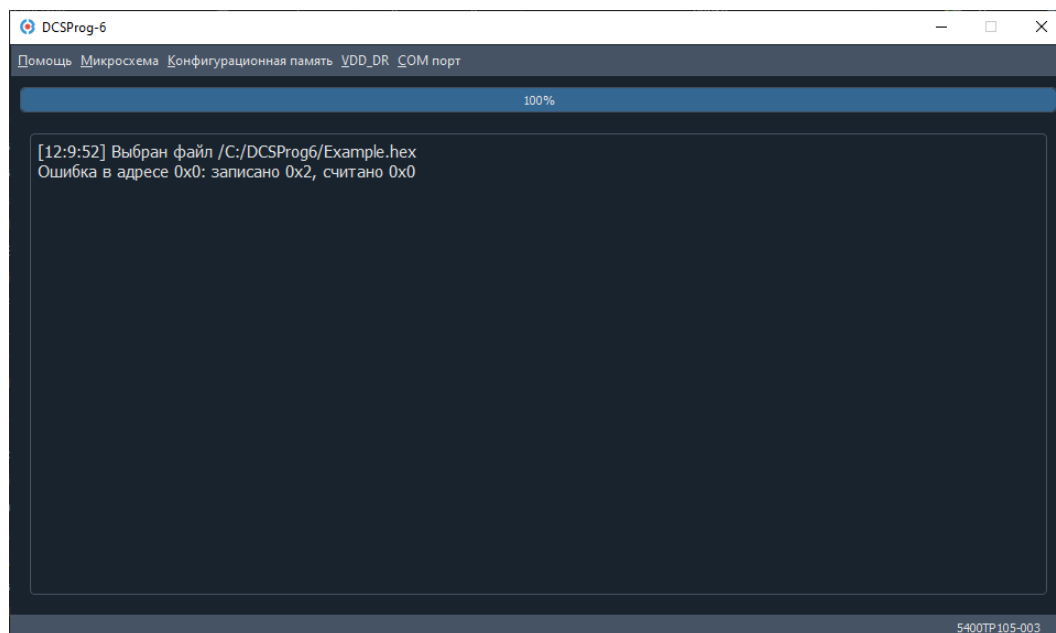


Рисунок 46. Ошибка при программировании в режиме SOFT

8. Микросхема запрограммирована и работает в режиме отладки (режим «SOFT»).

9. Для перепрограммирования микросхемы в режиме «SOFT» (отладка микросхемы) повторить методику с пункта 6.

В случае успешного программирования микросхемы на отладочной плате будут мигать светодиоды VD1-VD4.

Временная диаграмма представлена на рисунке ниже.

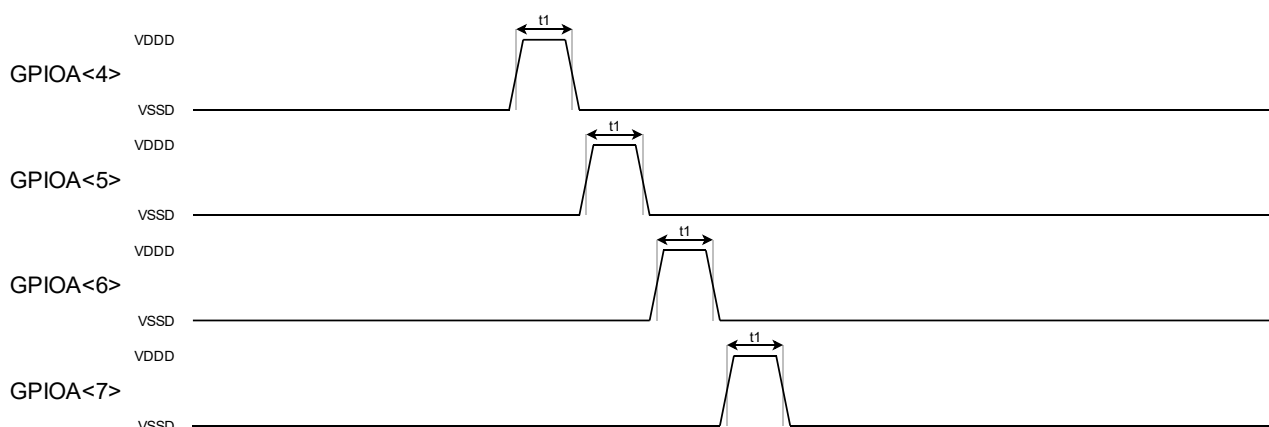


Рисунок 47. Временная диаграмма работы проекта Example.uvproj (t_1 – ~60 мс)

Программирование микросхемы в режим финальной конфигурации (режим «HARD»)

1. Запрограммировать микросхему в режим «SOFT».
2. Выполнить «Микросхема» → «Прожечь ПЗУ». В открывшемся окне подтвердить программирование микросхемы – кнопка «Yes» (Рисунок 48);

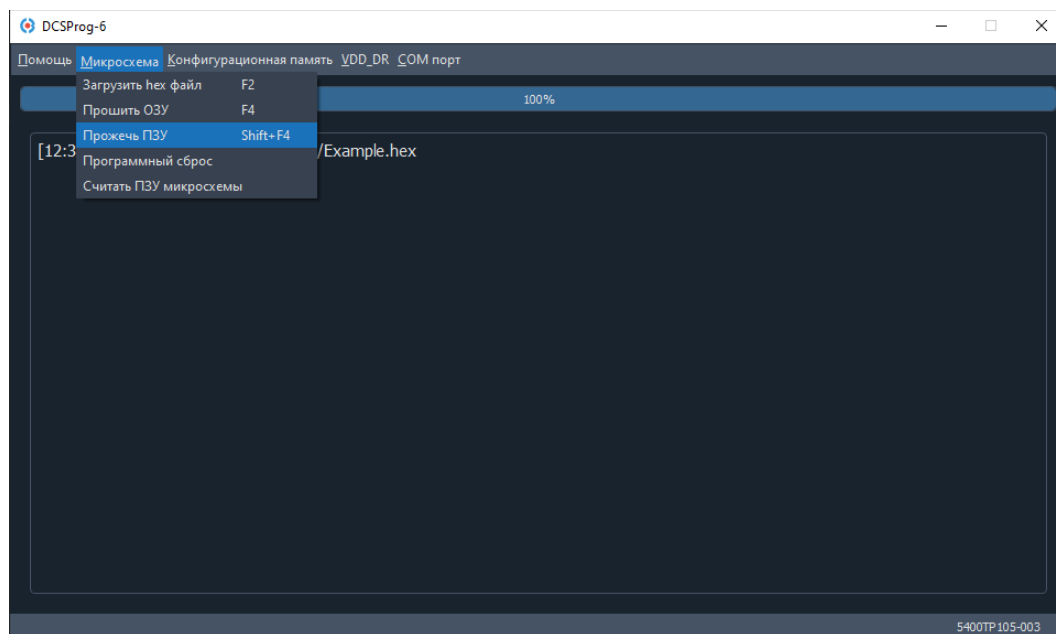


Рисунок 48. Программирование микросхемы в режим «HARD»

Примечание: Прожиг объемной программы занимает длительное время. В момент прожига взаимодействие с программой DCSProg6.exe запрещено.

После завершения программирования микросхемы программа выдаст сообщение «Микросхема запрограммирована в режиме HARD» (Рисунок 49). По завершению программирования произойдет автоматический программный сброс микроконтроллера по JTAG.

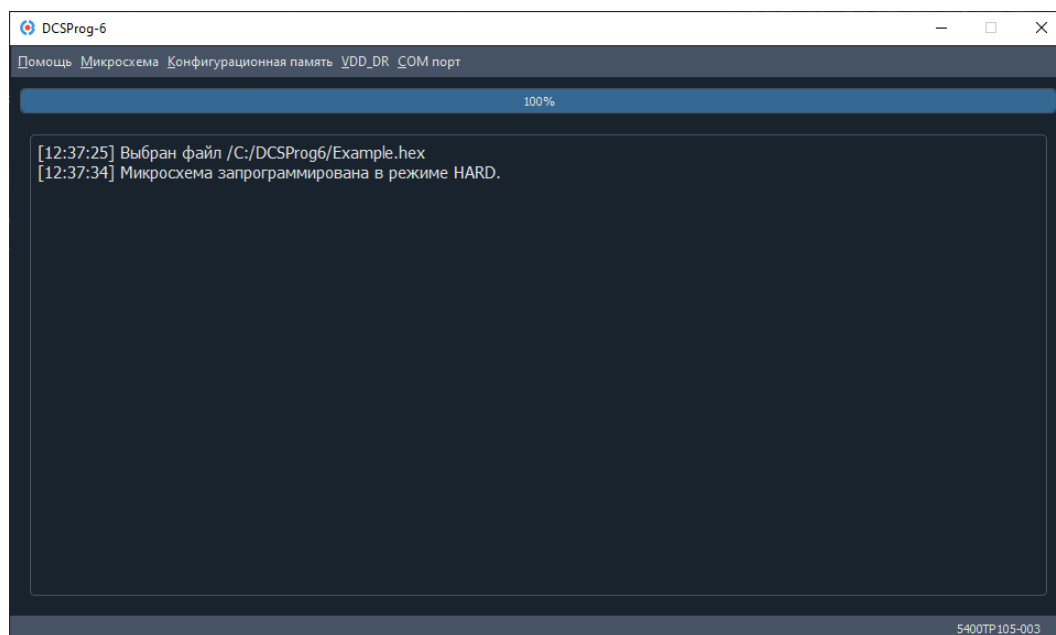


Рисунок 49. Окно окончания программирования

В случае ошибок при программировании в окне программы будет указан первый адрес, в котором возникла ошибка записи.

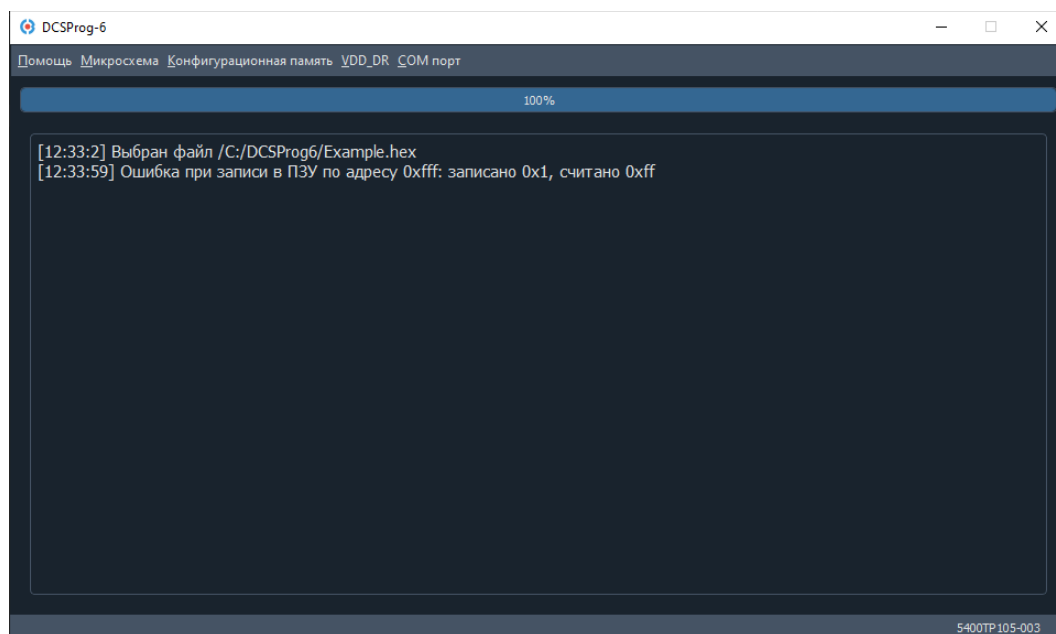


Рисунок 50. Ошибка записи в ПЗУ

После программирования в режиме «HARD» в качестве проверки верной записи данных можно считать их из однократно программируемого ПЗУ. Для этого необходимо выполнить «Микросхема» → «Считать ПЗУ микросхемы». Считанные данные запишутся в файл Read_ROM_time.txt.

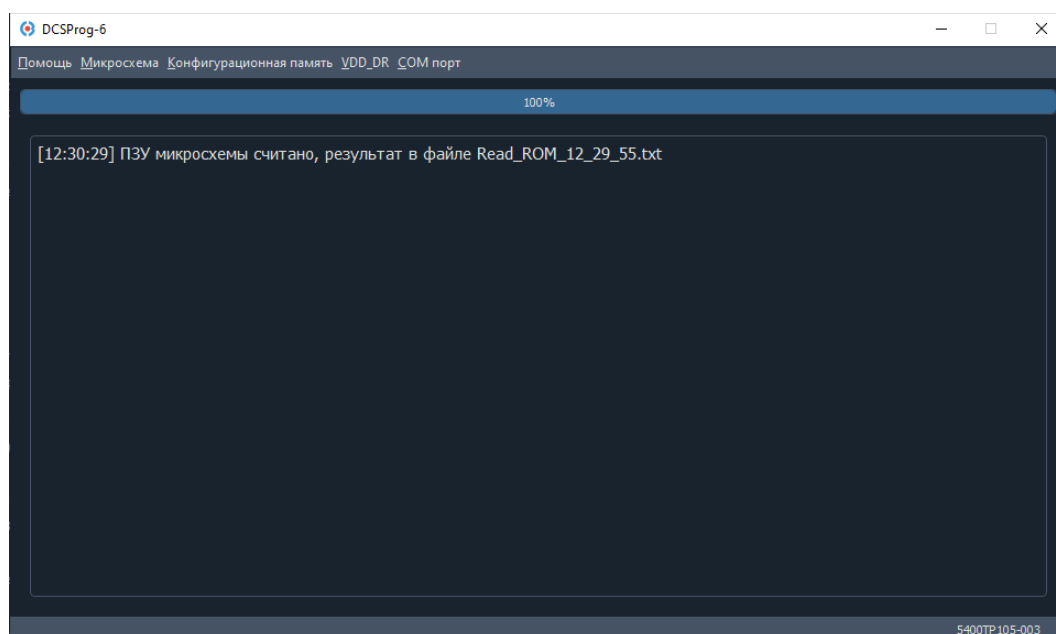


Рисунок 51. Считывание ПЗУ

Программирование конфигурационной памяти в режиме отладки (режим «SOFT»)

В микросхеме реализована конфигурационная память, которая может работать как в «SOFT», так и в «HARD» режиме. Более подробно о регистрах конфигурационной памяти см. в «Спецификация 5400TP105-003» раздел «ANALOG_CFG». Регистры модуля ANALOG_CFG подключены к конфигурационному однократно программируемому ПЗУ.

1. Открыть проект «analog_cfg_ex.uvproj». Путь к файлу проекта в архиве projects_keil «.../Projects/ANALOG_CFG/ analog_cfg_ex.uvproj».

Данный проект настраивает частоту RC-генератора на выводе GPIOB_1.

2. Для настройки конфигурационной памяти необходимо выставить нужные биты в регистрах ANALOG_CFG согласно таблице «Регистры модуля ANALOG_CFG» (см. «Спецификация 5400TP105-003») в проекте analog_cfg_ex.uvproj.

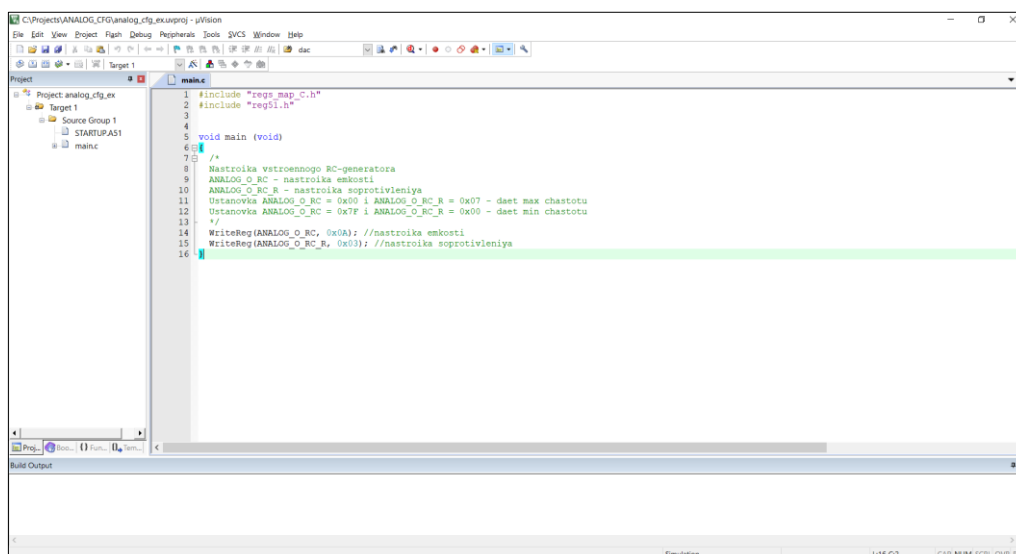


Рисунок 52. Пример настройки конфигурационной памяти

3. Выполнить «Микросхема» → «Загрузить файл», выбрать скомпилированный hex-файл и нажать кнопку «Открыть» (Рисунок 53);

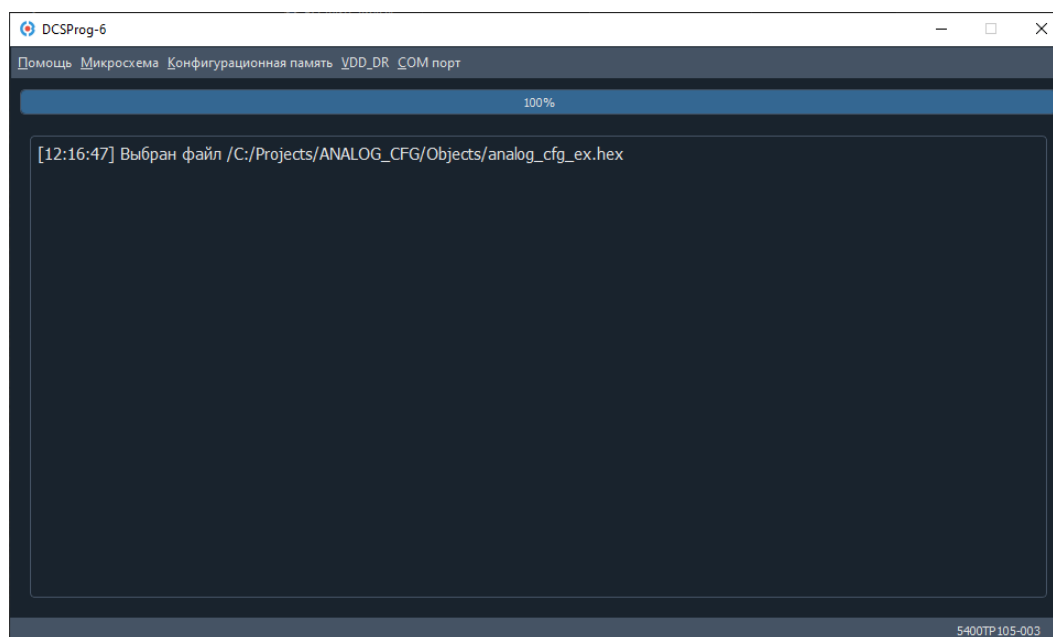


Рисунок 53. Окно программы после выбора hex-файла

4. Выполнить «Конфигурационная память» → «Прошить конфигурационную память» для программирования в режиме отладки (режим «SOFT») (Рисунок 54);

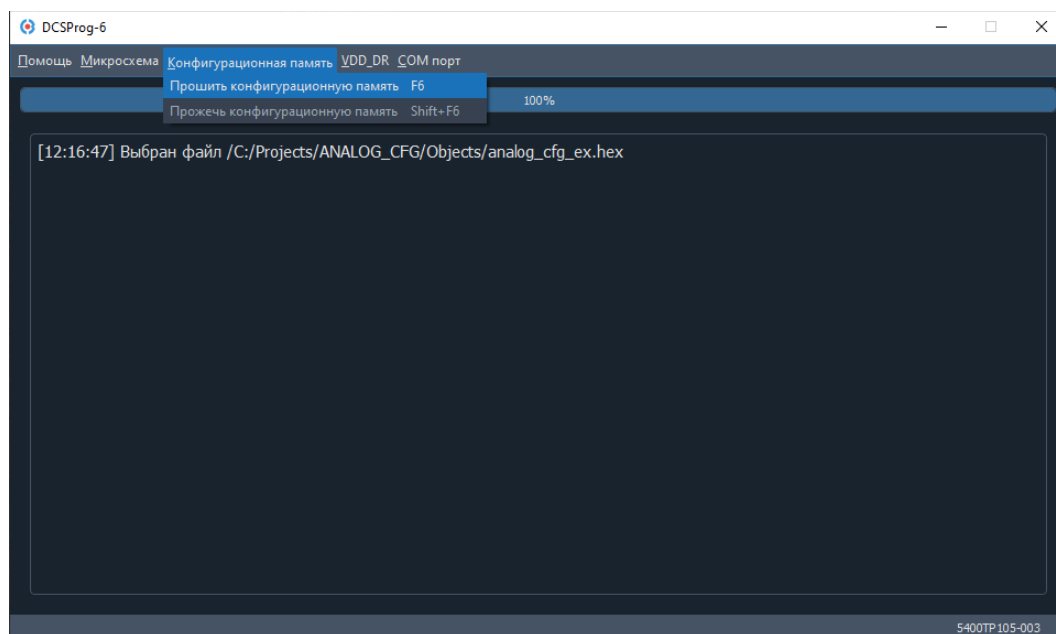


Рисунок 54. Прошивка конфигурационной памяти («SOFT»)

5. После завершения программирования программа выдаст сообщение «Конфигурационная память настроена в режиме «SOFT» (Рисунок 55).

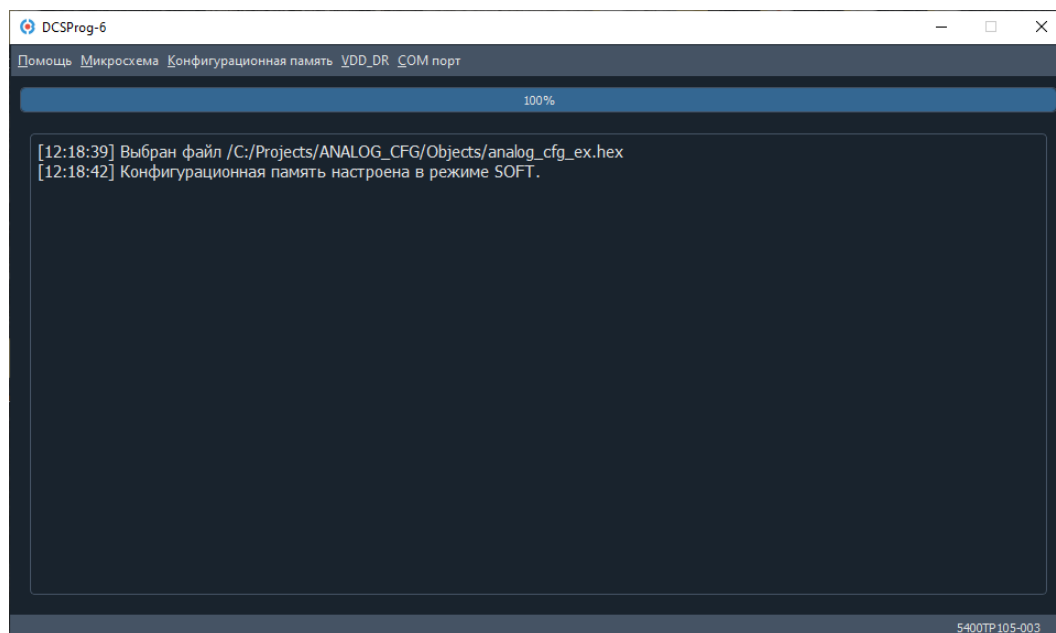


Рисунок 55. Завершение настройки конфигурационной памяти

Результат программирования конфигурационной памяти можно проверить на выводе GPIOB <1> – частота RC-генератора.

Программирование конфигурационной памяти в режим «HARD»

1. Настроить конфигурационную память в режиме «SOFT».
2. Выполнить «Конфигурационная память – Прожечь конфигурационную память» (Рисунок 56).

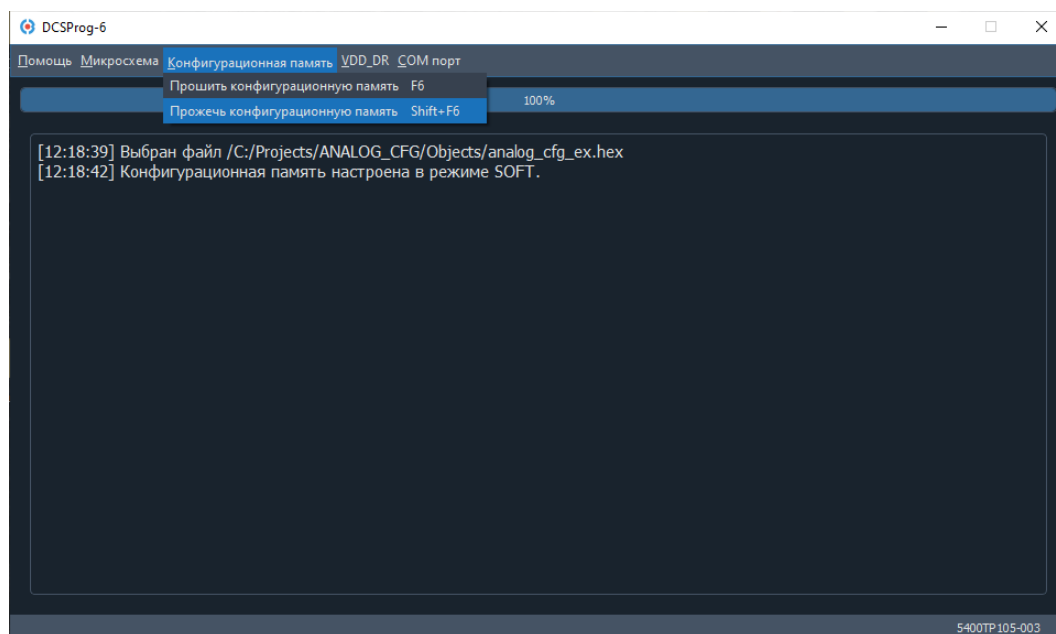


Рисунок 56. Прожиг конфигурационной памяти («HARD»)

Конфигурационная память запрограммирована и работает в режиме финальной конфигурации (режим «HARD»). Результат программирования можно проверить на выводе GPIOB <1> – частота внутреннего RC-генератора.

Программный сброс

Помимо аппаратного сброса (кнопка RESET (Рисунок 1, №14) на отладочной плате) в DCSProg6 реализован программный сброс. Для сброса необходимо выполнить «Микросхема» → «Программный сброс» (Рисунок 57), при этом программа начнет выполняться заново.

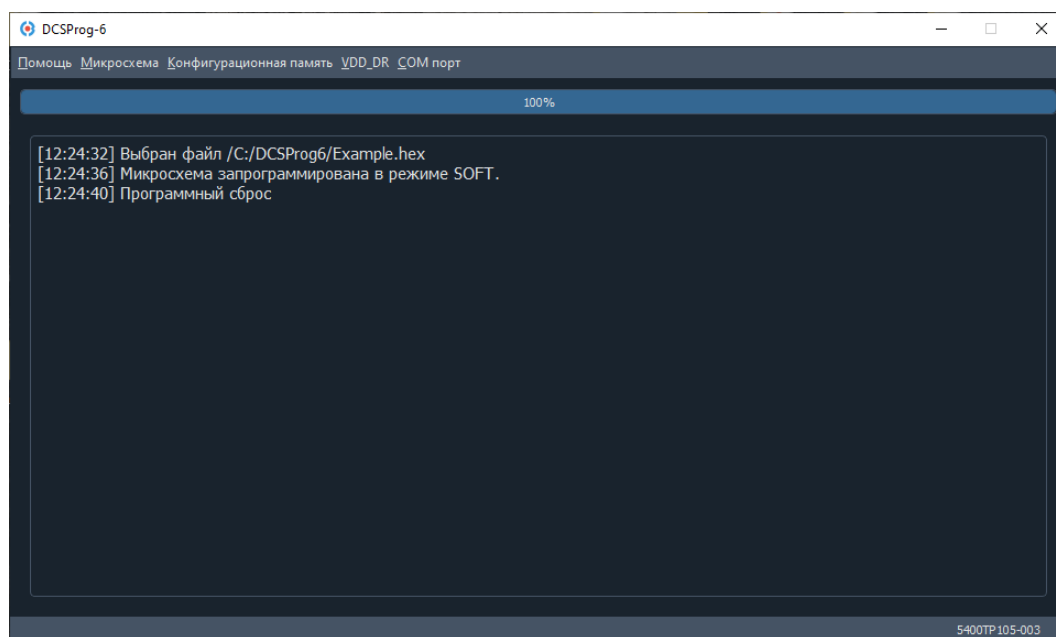


Рисунок 57. Программный сброс

Настройка напряжения питания универсальных портов ввода-вывода

В ПО есть возможность настройки напряжения на выводе VDD_DR (Положительное напряжение питания универсальных портов ввода-вывода микроконтроллера). По умолчанию на выводе VDD_DR установлено напряжение 5,0 В.

В пункте меню «VDD_DR» из выпадающего списка выберите требуемое напряжение (Рисунок 58).

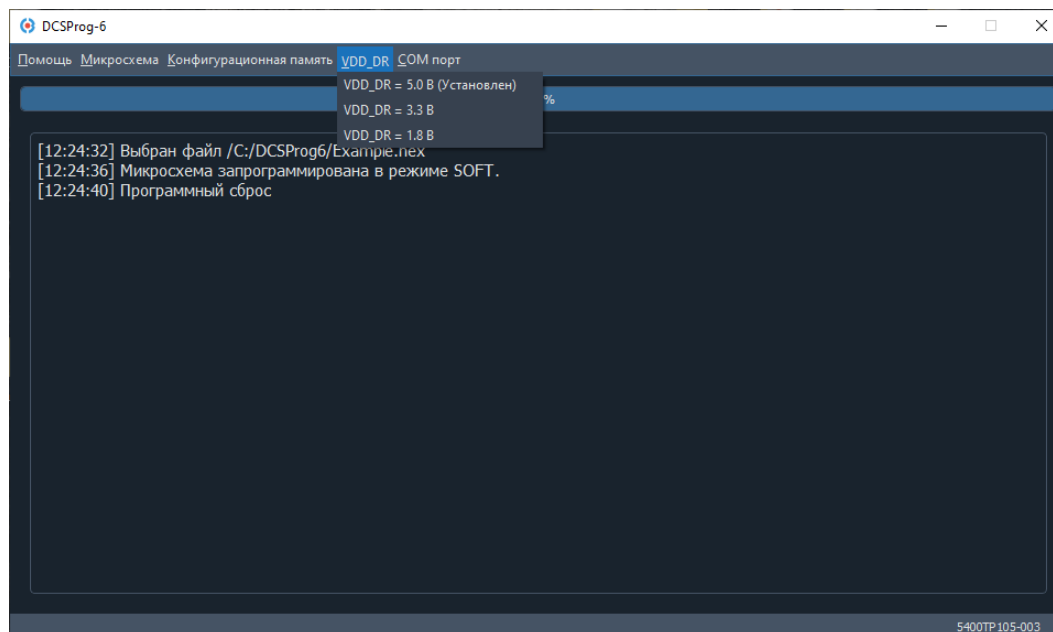


Рисунок 58. Выбор напряжения на выводе VDD_DR

После выбора напряжения программа выдаст информационное сообщение (Рисунок 59).

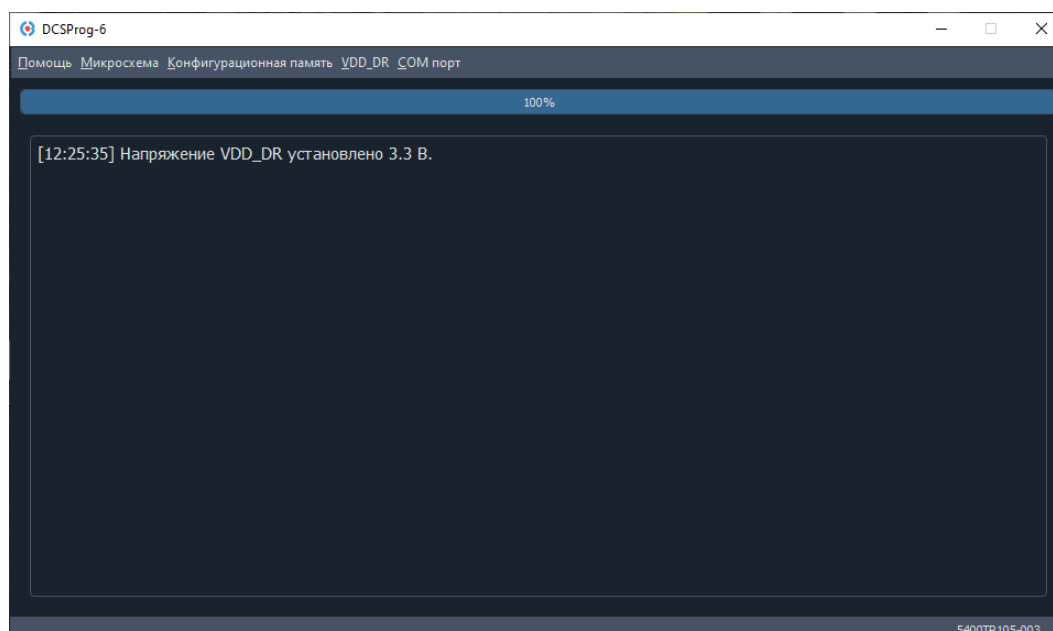


Рисунок 59. Результат успешного переключения напряжения на выводе VDD_DR

При отключении питания и повторном включении, напряжение на выводе VDD_DR установится по умолчанию (5,0 В).

Обновление программного обеспечения DCSProg

Для обновления ПО программатора необходимо:

1. Скачать новый архив DCSProg6 с сайта компании <https://dcsoyuz.ru> (раздел «Программное обеспечение») и извлечь данные на персональный компьютер.

2. Открыть программу DCSProg6.exe из скачанного архива и выполнить «Помощь» → «Обновить ПО» (Рисунок 60).

Программатор обновляет свою прошивку с помощью файла STM32_DCSProg5v3.hex из корневой директории архива.

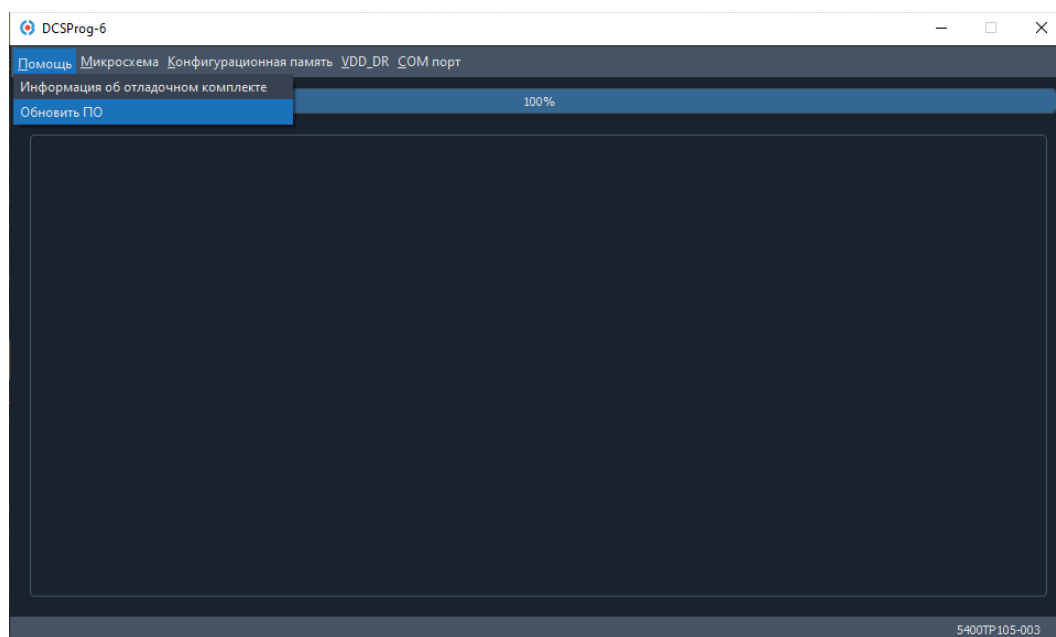


Рисунок 60. Меню обновления DCSProg6

3. Перезапустить приложение DCSProg6.exe.

Лист регистрации изменений

[illegible]