

Контроллеры ARMatura

© Dmitry Ponyatov <dponyatov@gmail.com>, SSAU ASCL

27 февраля 2013 г.

Оглавление

Оглавление	1
I Введение	5
II Железо	7
1 STM32VLDISCOVERY /STM32F100RBT6/	8
2 STM32F4DISCOVERY /SRM32F407VGT6/	10
3 ARMatura /STM32F417IGT/	12
4 PION /STM32F100C4T6B/	13
III Первые шаги	14
5 Установка пакета ПО STlink	15
6 Установка Keil MDK-ARM	21
7 Первый проект: blink	28
7.1 Настройки проекта в Keil	29
7.2 Структура файлов	38
8 Hell Of World	39

IV Средства разработки	40
9 Keil MDK-ARM	41
10 Компиляторы	42
10.1 GCC	42
10.2 KeilCC	42
10.3 IAR	42
11 IDE	43
11.1 Eclipse	43
11.2 Code::Blocks	43
11.3 gVim	43
11.4 Keil uVision	43
11.5 IAR	43
12 Программаторы	44
12.1 STlink	44
12.2 Serial Boot	44
13 Отладчики	45
13.1 JTAG	45
13.2 STM32 SWD	45
13.3 GDB	45
V Основы языка C⁺	46
14 Синтаксис	47
15 Типы данных	48
16 Стандартная библиотека libc	49
VI Отладка	50
17 JTAG	51
18 GDB	52
19 OpenOCD	53
VICMSIS	54
20 Startup	55
21 Стандартная библиотека STM32	56
22 USB client/host	57

<i>Оглавление</i>	3
VII Ядро Cortex-Mx	58
23 Режимы ARM и Thumb	59
24 DMA	60
25 DSP /Cortex-M3/	61
26 FPU /Cortex-M4F/	62
IX Интерфейсы	63
27 USB	64
28 UART	65
29 SPI	66
30 I2C	67
31 CAN	68
X Операционные системы OCPB	69
32 Keil RTX	70
33 FreeRTOS	71
34 eCos	72
35 Linux	73
XI Стек TCP/IP	74
36 Ethernet	75
37 PPP	76
XII Типовые применения	77
38 GPS	78
38.1 Tripod15	78
38.2 WISMO228	78
39 GSM	79
39.1 WISMO228	79
40 шина Dallas 1Wire	80
40.1 RTC	80
40.2 Датчики температуры DS18x20	80

XІІИстраиваемый Linux	81
XІМриложения	82
41 Сводная таблица процессоров	83
41.1 STM32F10x	83

Часть I

Введение

Эта книга – набор методичек по разработке ПО для встраиваемых систем, написанных для Института космического приборостроения СГАУ.

Для применения в реальных проектах научной аппаратуры была разработана линейка унифицированных модулей:

1. ARMatura — модуль на мощном микропроцессоре STM32F417IGT: 1M Flash, 192K SRAM, TQFP176, DSP, FPU,.. [41](#)

предназначен для использования в качестве центрального процессора цифровой системы: обработка данных, сложные алгоритмы управления, ЦОС, вычисления, реализация протоколов передачи данных по интерфейсам USB, Ethernet, RS232/UART, SPI, I2C, CAN,..

2. PION [4](#) — модуль на самом простом и дешевом STM32F100C4T6B: 128K Flash, 8K SRAM, UART, SPI [41.1](#)

периферийный модуль для стыковки с аналоговыми датчиками и исполнительными устройствами, предварительная ЦОС обработка, передача данных на ARMatura-модули для дальнейшей обработки данных.

также модуль применим в качестве самостоятельного простого интерфейса при замене на чип STM32F103 с портом USB или установки внешних интерфейсных микросхем FT232RL (USB Serial), CP1202, MC1551 (CAN).

3. BACKPLANE — коммутационная плата межмодульного интерфейса
4. POWER — модуль импульсного источника питания
5. STEPPER — модуль управления двухфазным шаговым двигателем
6. WISMO — несущая плата для GPS/GSM модуля WISMO 228
7. QVGA — несущая плата для TFT touch-панели

В качестве базового микроконтроллера были выбраны чипы семейства STM32Fxxx с ядрами Cortex-M3, Cortex-M4F (ARM) как самые дешевые, и имеющие хорошую поддержку в виде отладочных плат линейки Discovery.

В общем, линейка модулей ARMatura может рассматриваться в качестве замены устаревшей линейки периферийных контроллеров Arduino на базе МК AVR8.

Проект размещен в репозитории <https://github.com/ponyatov/ARMatura.git> и предоставляется на условиях OpenHardware licence (за исключением прошивок и схем по тематике ИКП СГАУ).

Контакты разработчиков:

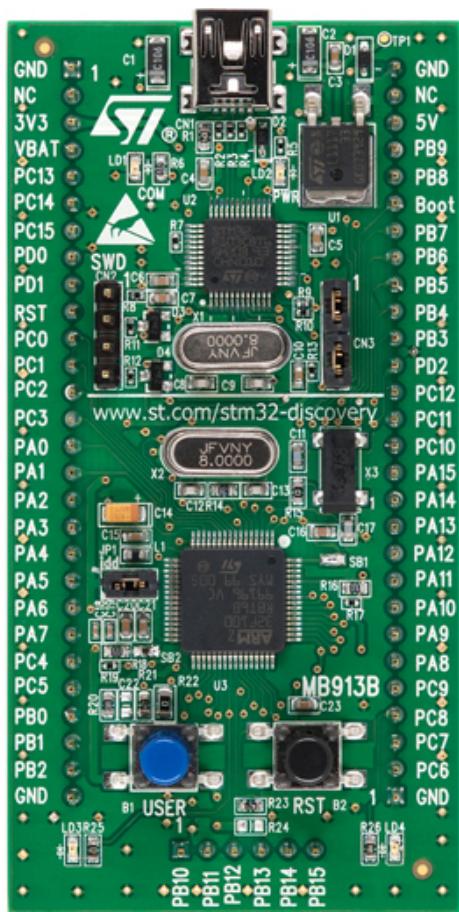
- ИКП СГАУ <semkin@ssau.ru>
- Дмитрий Понятов <dponyatov@gmail.com>

Часть II

Железо

Глава 1

STM32VLDISCOVERY /STM32F100RBT6/



- Микроконтроллер STM32F100RB [41](#), 128 KB Flash, 8 KB RAM in 64-pin LQFP
- Встроенный ST-Link с возможностью использования в режиме внешнего программатора (только с SWD коннектором)
- Разработана для питания как от USB, так и от внешнего источника 3.3 или 5 вольт
- Может обеспечить питание 3 и 5 вольт для внешних устройств
- Два пользовательских светодиода (зеленый и синий)
- Пользовательская кнопка USER

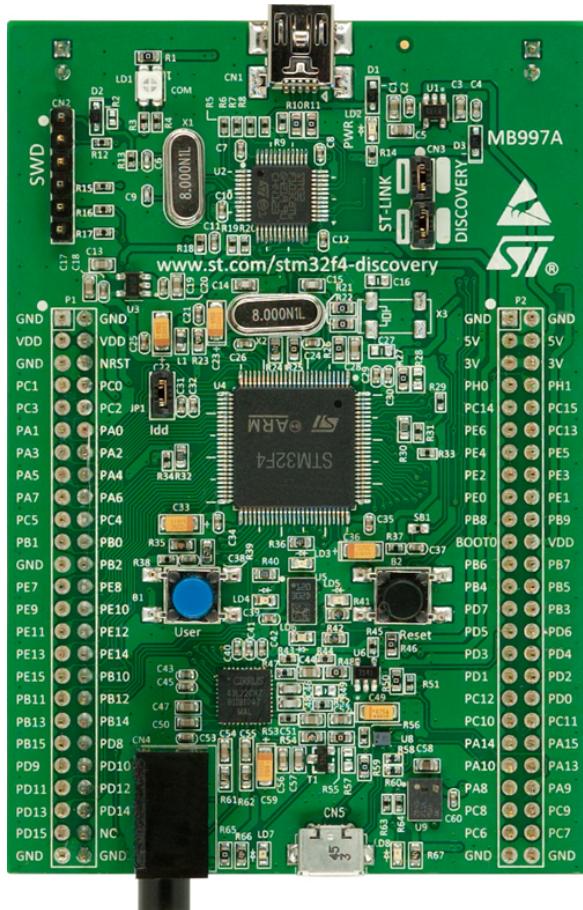
- Кнопка сброса RESET
- Контактные гребенки для всех выводов QFP64 для быстрого подключения и монтажа прототипа

Цена в розницу: 750 р.

<http://www.voltmaster.ru/cgi-bin/qwery.pl?id=127000573571&group=7000046>

Глава 2

STM32F4DISCOVERY /SRM32F407VGT6/



- микроконтроллер STM32F407VGT6 41 на базе 32-битного ядра Cortex-M4F, 1 MB Flash, 192 KB RAM в корпусе LQFP100
- встроенный ST-LINK/V2 с возможностью использования в режиме внешнего программатора (только с SWD коннектором)
- Разработана для питания как от USB, так и от внешнего источника 5 вольт
- Может обеспечить питание 3 и 5 вольт для внешних устройств
- LIS302DL, ST MEMS датчик движения, 3-осевой цифровой гироскоп
- MP45DT02, ST MEMS аудиодатчик, всенаправленный цифровой микрофон

- CS43L22, аудиоЖАП со встроенным аудиоусилителем класса D
- Восемь LEDs: LD1 (red/green) for USB communication LD2 (red) for 3.3 V power on Four user LEDs, LD3 (orange), LD4 (green), LD5 (red) and LD6 (blue) 2 USB OTG LEDs LD7 (green) VBus and LD8 (red) over-current
- Пользовательская кнопка USER
- Кнопка сброса RESET
- USB OTG FS with micro-AB connector
- Контактные гребенки для всех выводов LQFP100 ля быстрого подключения и мон-тажа прототипа

Цена в розницу: 1140 р.

<http://www.voltmaster.ru/cgi-bin/qwery.pl?id=127000854271&group=7000046>

Глава 3

ARMatura /STM32F417IGT/

Глава 4

PION /STM32F100C4T6B/

Модуль PION предназначен для мелких задач управления, первичной обработки данных,стыковки с устройствами измерения и исполнительными устройствами, т.е. для тех задач, для которых ранее использовались микроконтроллеры Atmel AVR8.

процессор	STM32F100C4T6B	41.1
ROM		16K
RAM		4K
шина	AUTObus	
интерфейсы	UART	1
	SPI	1
	AIЦП	10x12b
	ЦАП	2x12b
буфер	Parallel Flash	64K

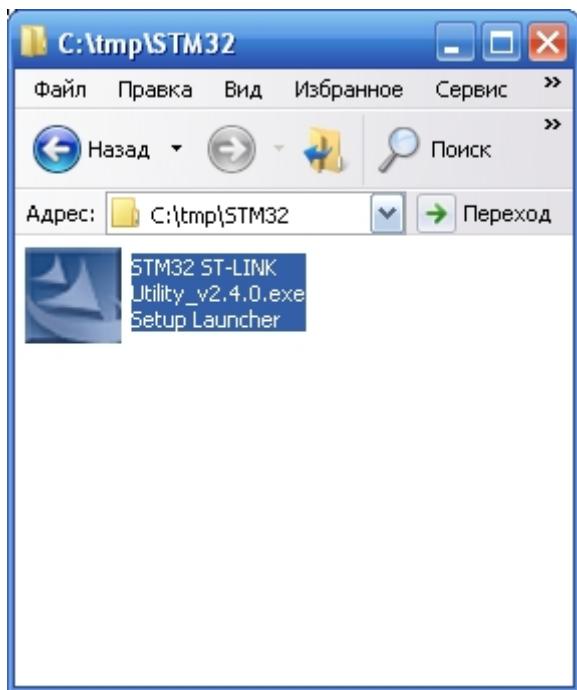
Часть III

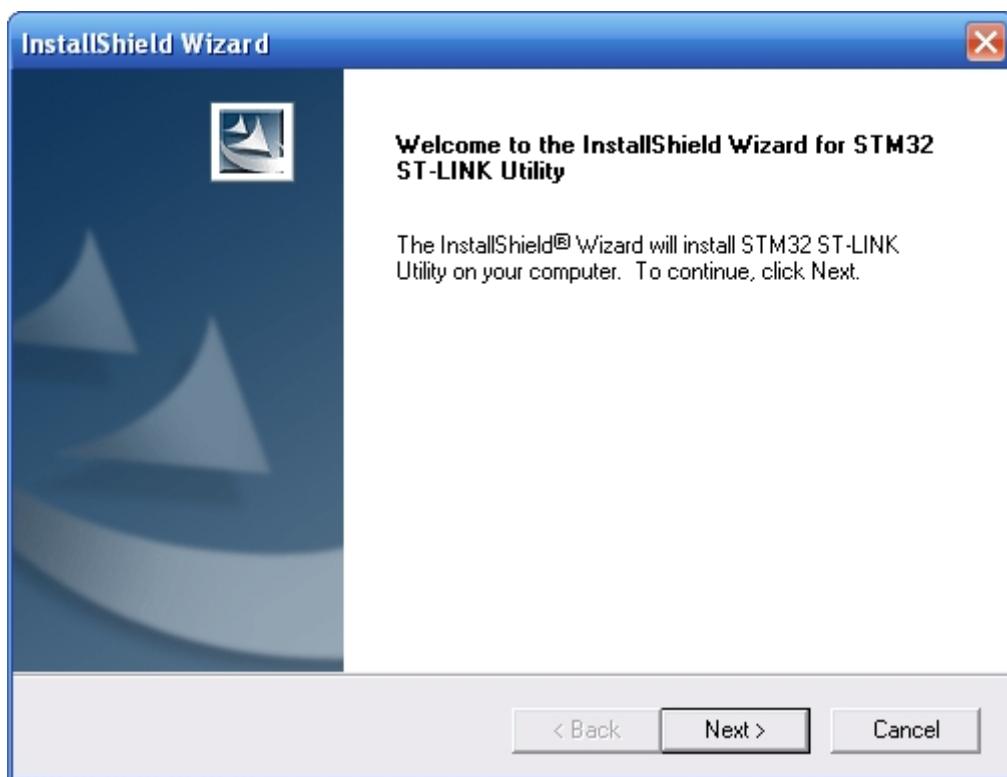
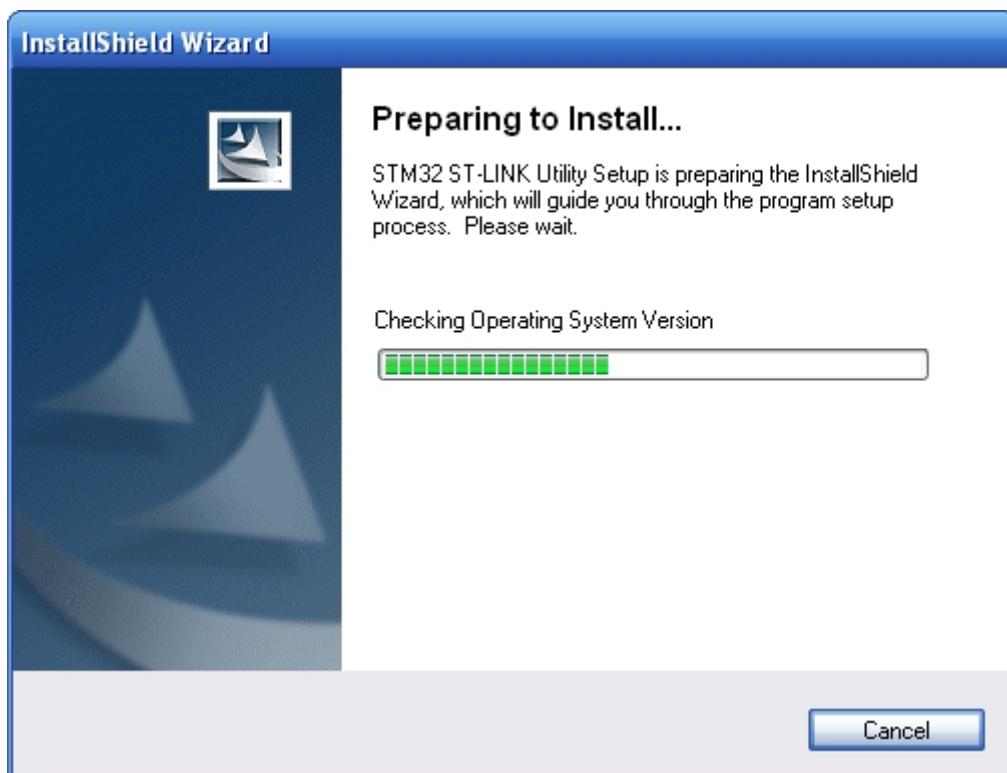
Первые шаги

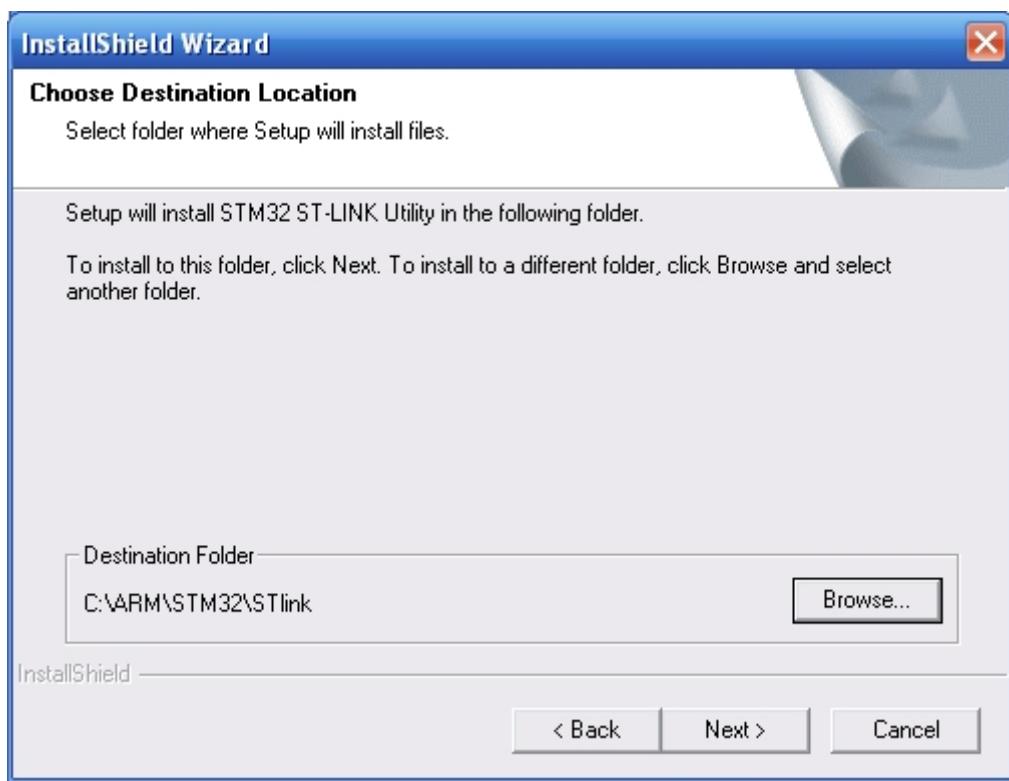
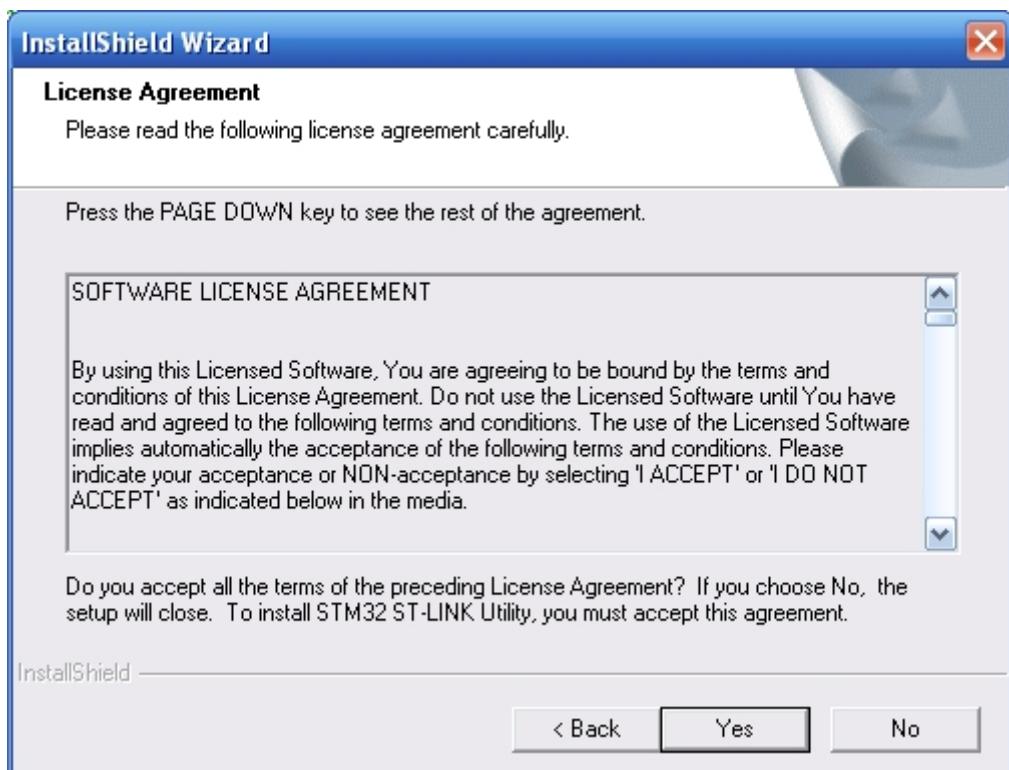
Глава 5

Установка пакета ПО STlink

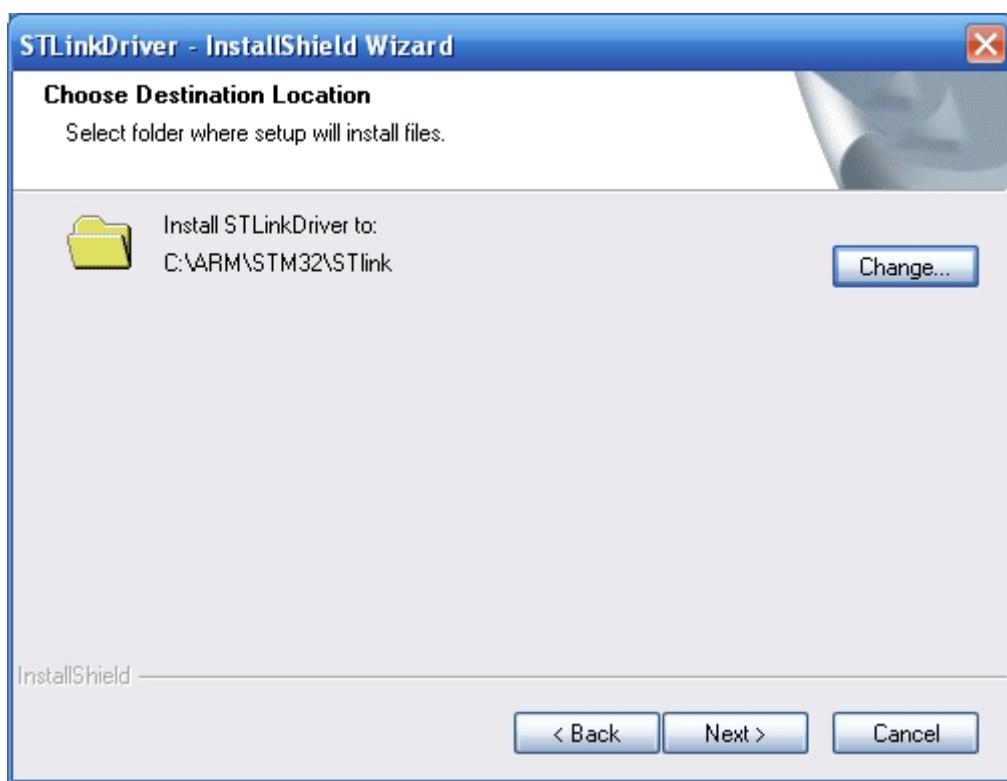
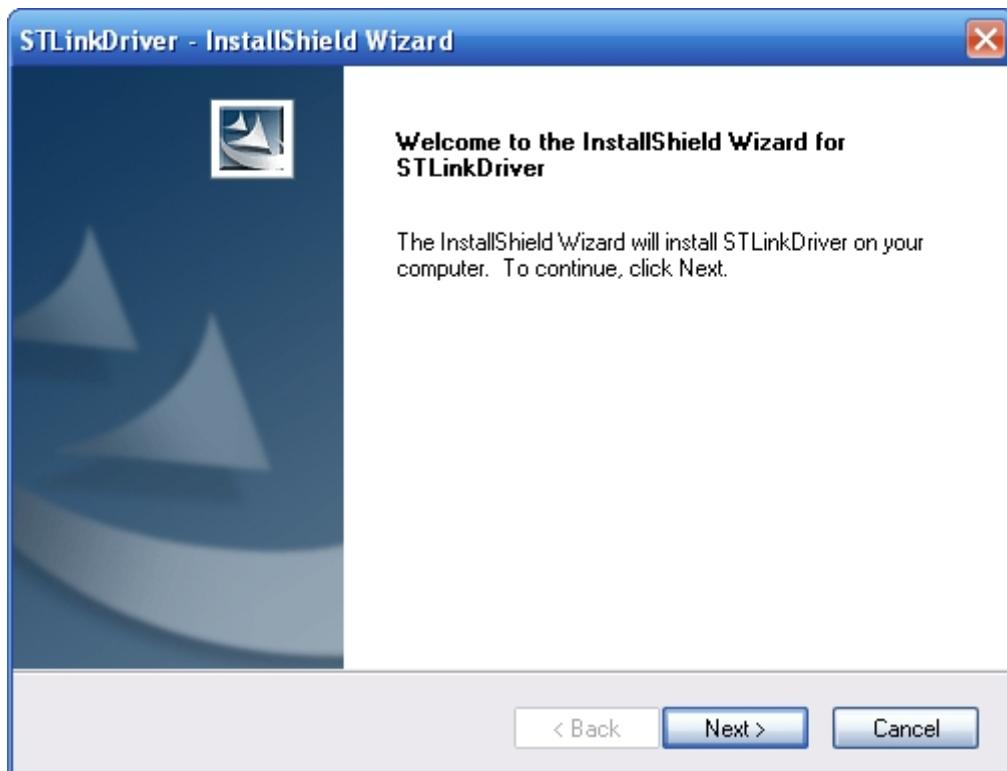
Для работы с чипами STM32 используя программаторы STlink, встроенные на демоплаты Discovery необходимо установить пакет STSW-LINK004 (STM32 ST-link Utility и драйвер), скачав его с <http://www.st.com/web/en/catalog/tools/PF258168>

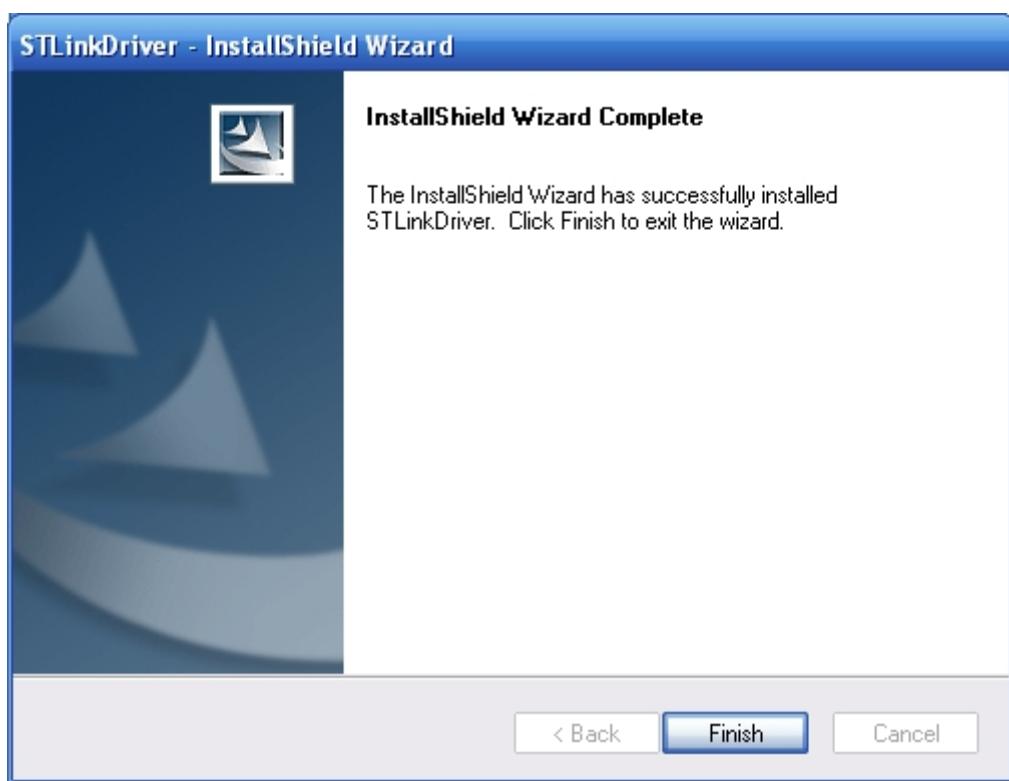
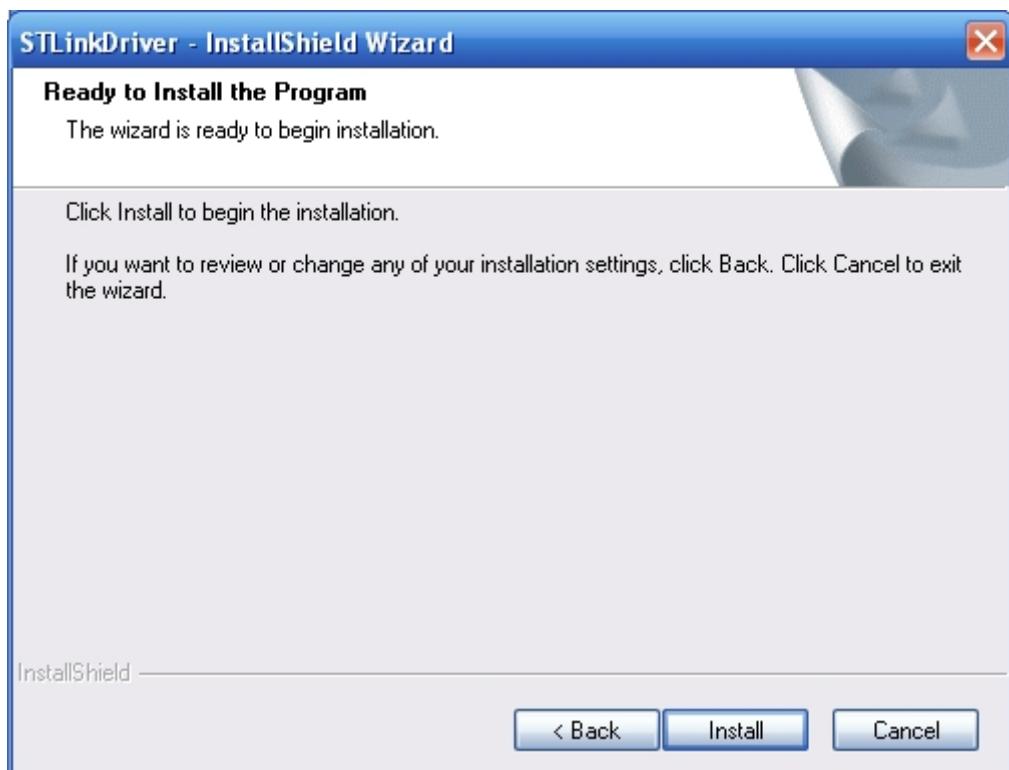


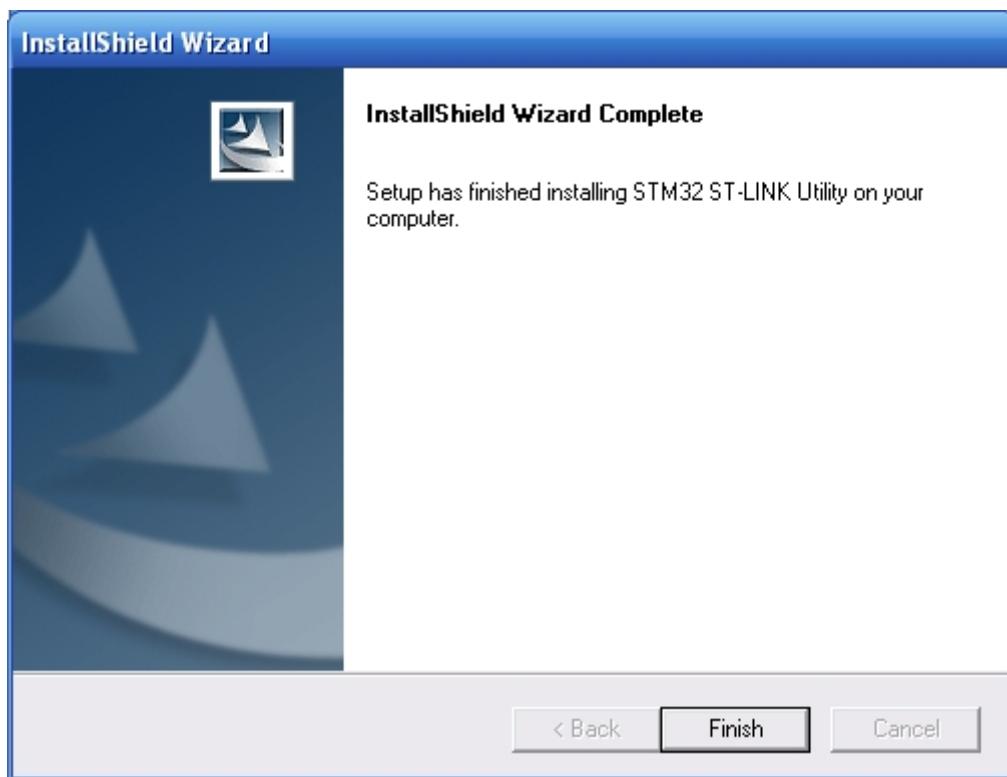




Пакет драйвера входит в пакет STSW-LINK004, и запускается автоматически







Глава 6

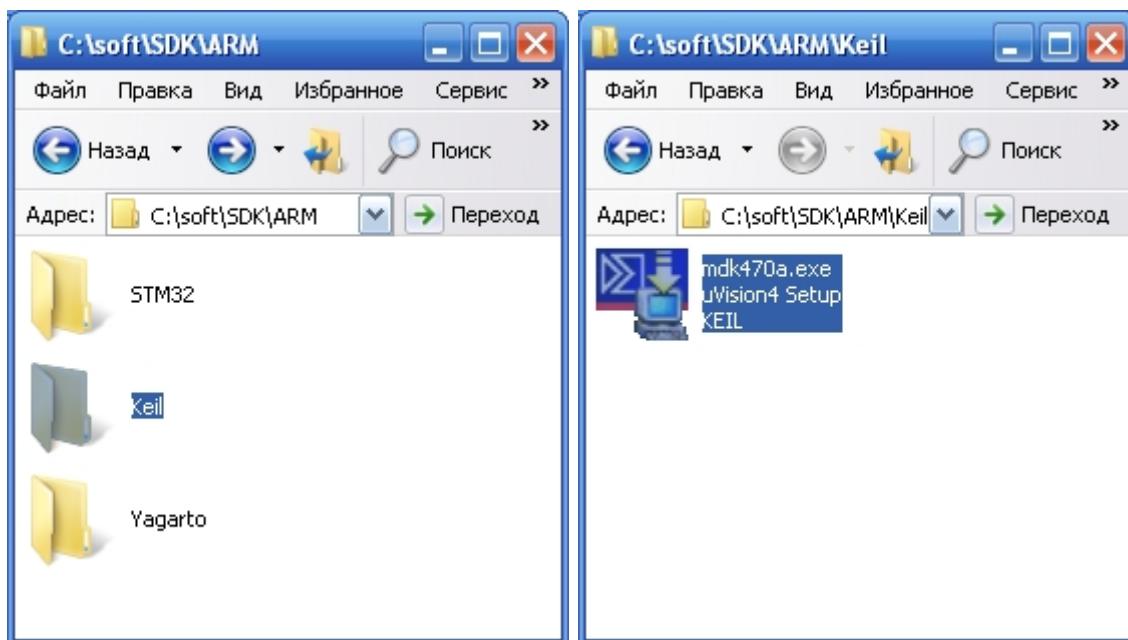
Установка Keil MDK-ARM

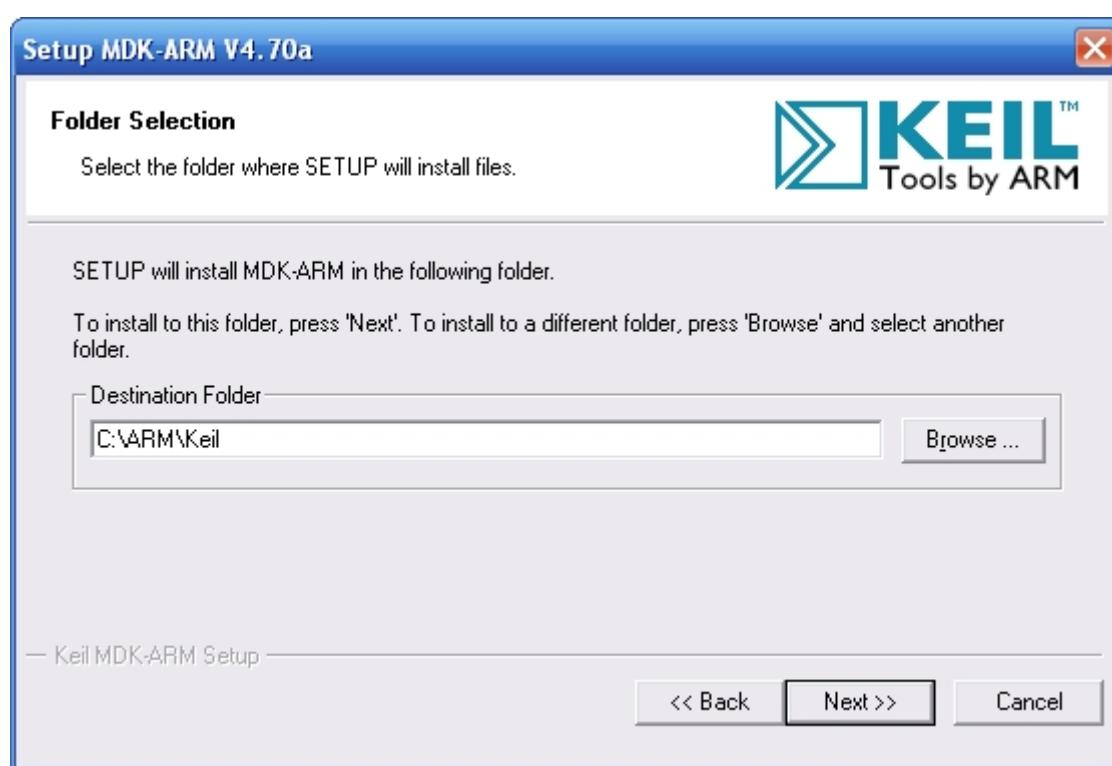
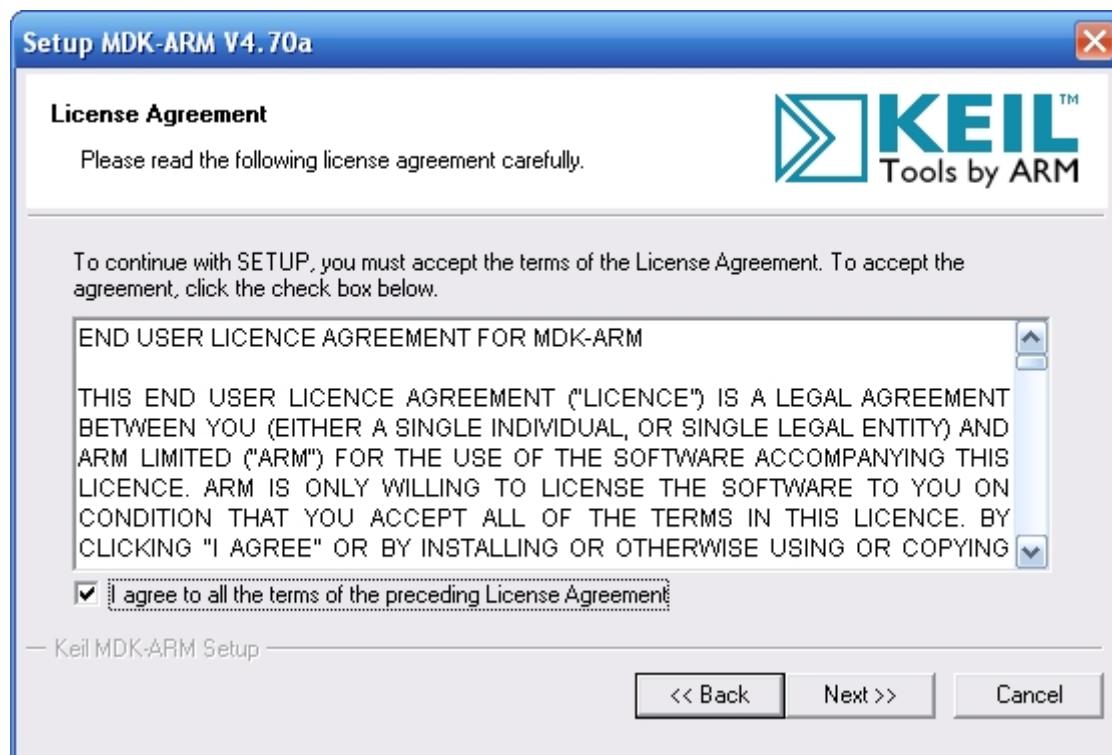


Для начала освоения программирования для ARM рекомендуем использовать бесплатный пакет от Keil: <http://www.keil.com/arm/mdk.asp> — ограничения бесплатной версии в 32К кода вполне достаточно для начального освоения программирования под процессоры семейства Cortex-Mx, а затем уже можно переползать на открытое ПО: GNU toolchain 10.1, Eclipse 11.1 и Linux XIII.

Процесс установки и первоначальной настройки описан в 9. В этом разделе будет рассмотрен только процесс установки и начальной настройки, подробно о пакете Keil MDK-ARM см.

Качаем пакет с официального сайта, заполнив анкету: <https://www.keil.com/demo/eval/arm.htm>.

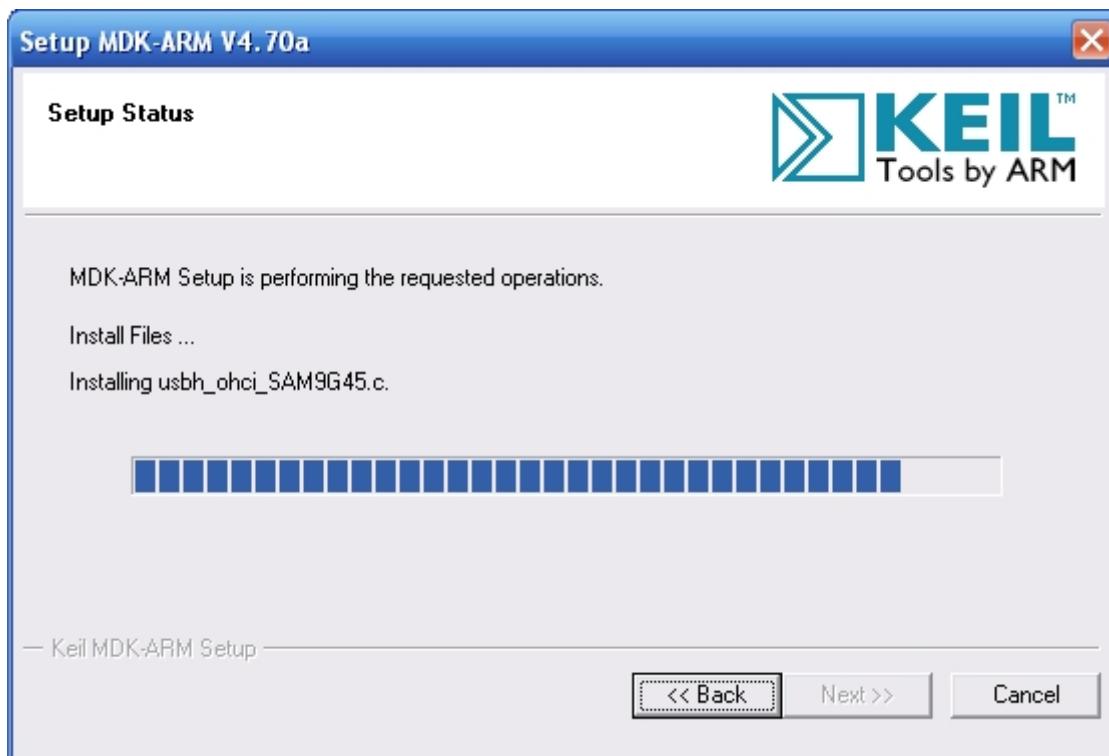


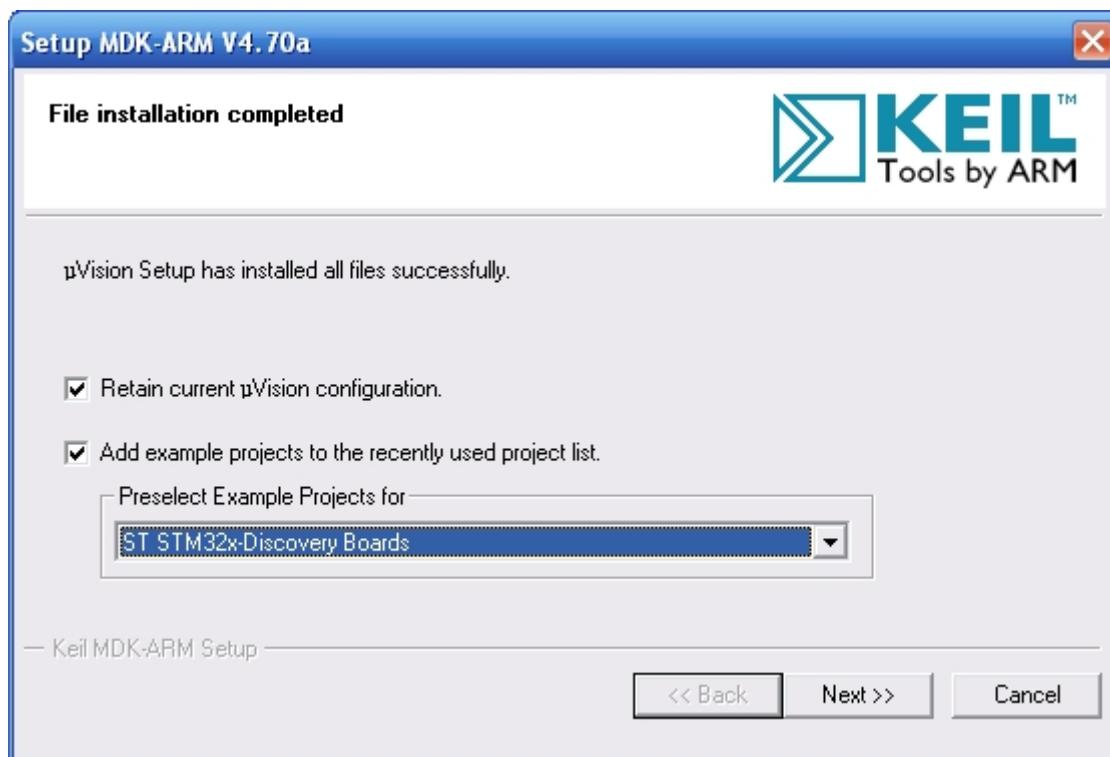


Путь установки пакета

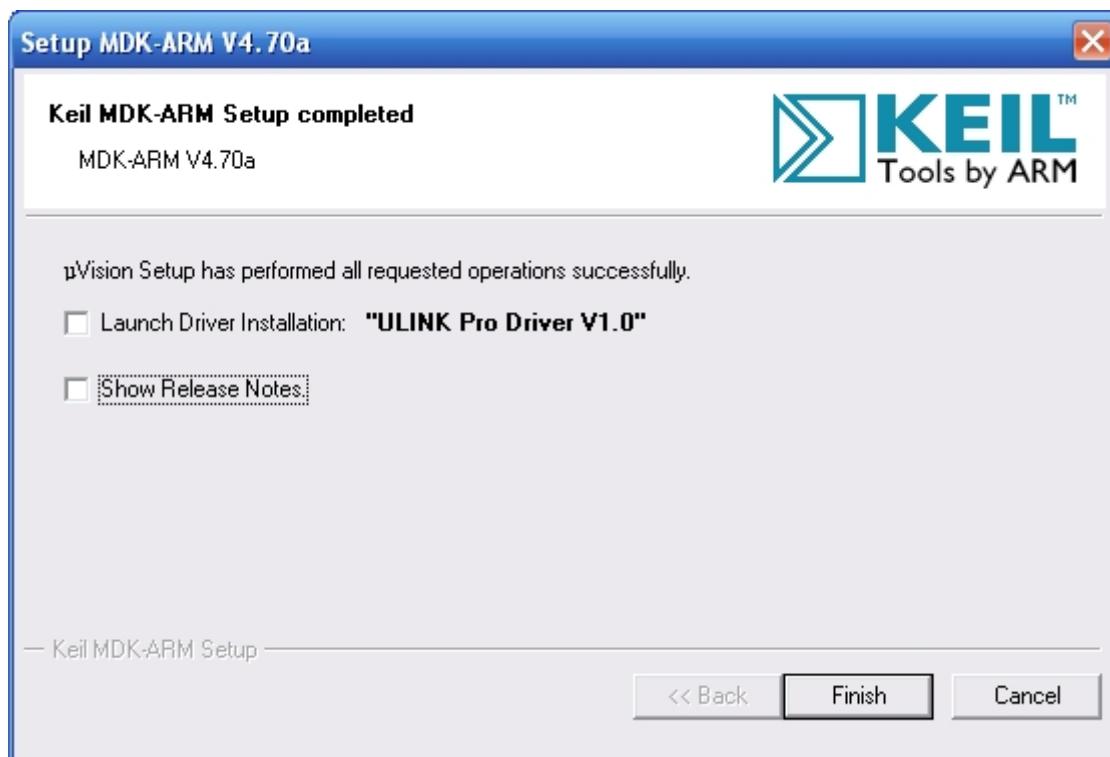


Личные данные: имя, название компании или hobbit, адрес электронной почты.





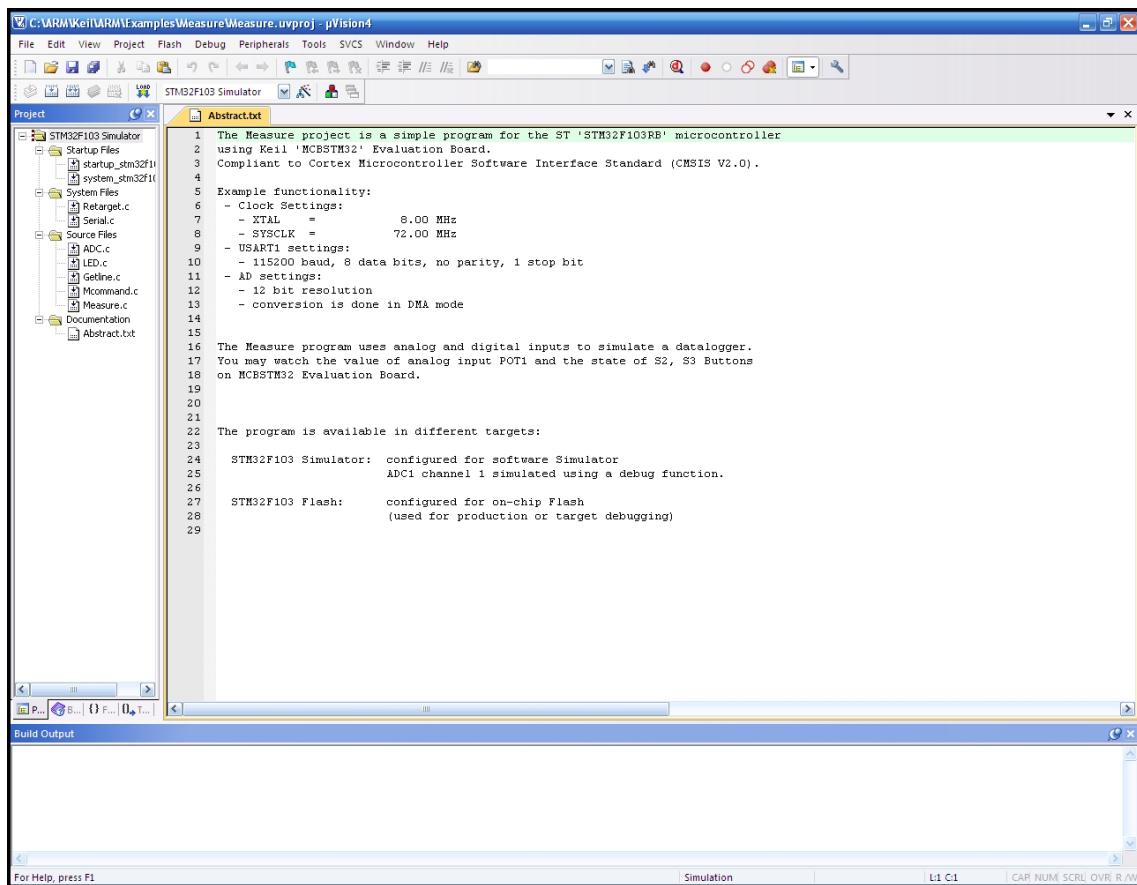
Укажите какие примеры кода добавить в список recently used project list: для работы с другими типами микропроцессоров выберете соответствующий раздел, или оставьте Simulation Hardware по умолчанию.



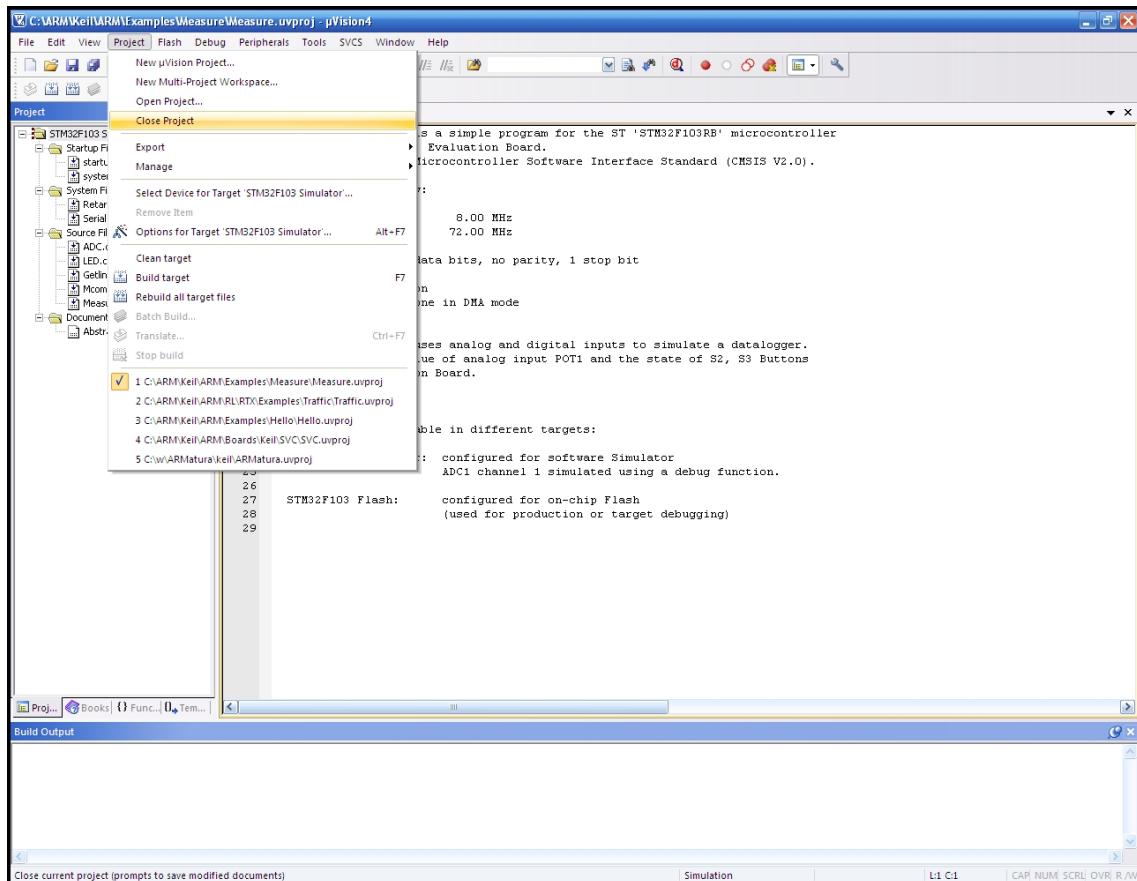
Снять установку драйвера программатора ULINK (если у вас его нет) и вывод текстового файла с последними изменениями Keil.



После запуска открывается проект по умолчанию, настроенный для программного симулятора STM32F103.



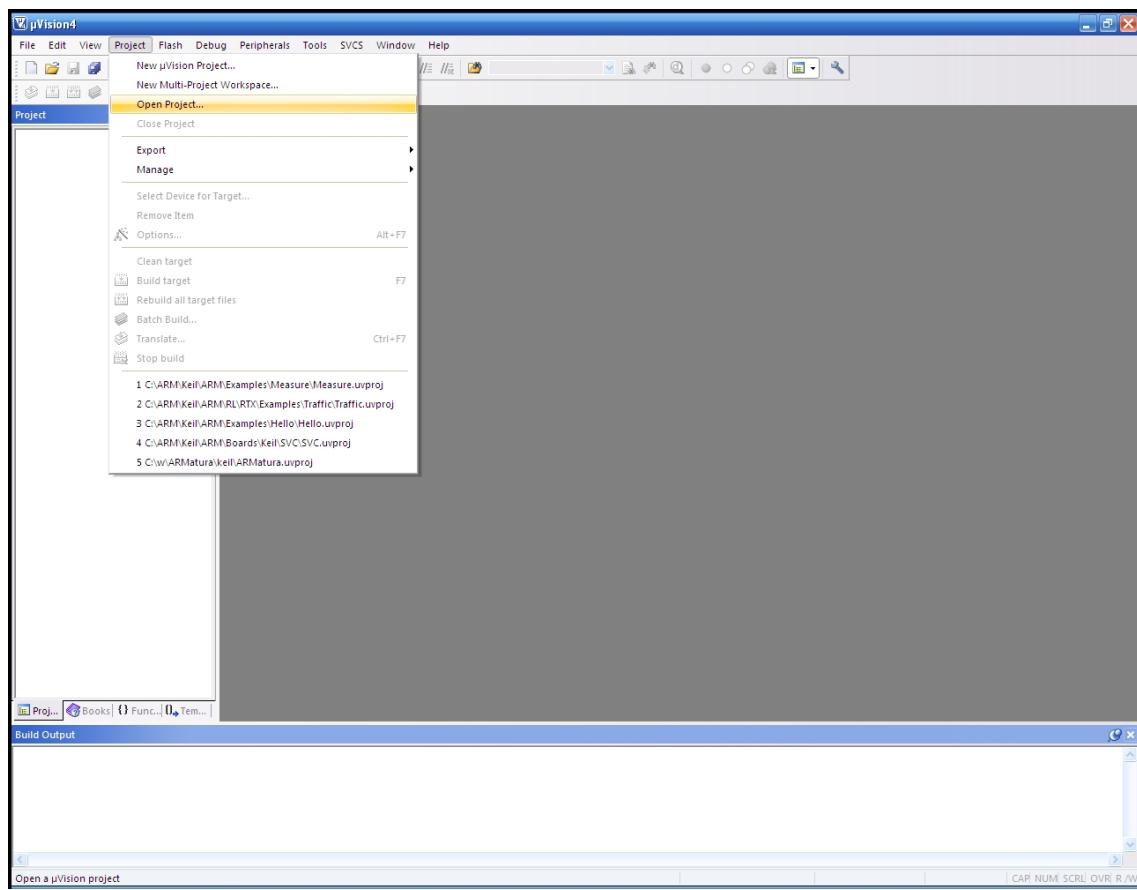
Нужно его закрыть,

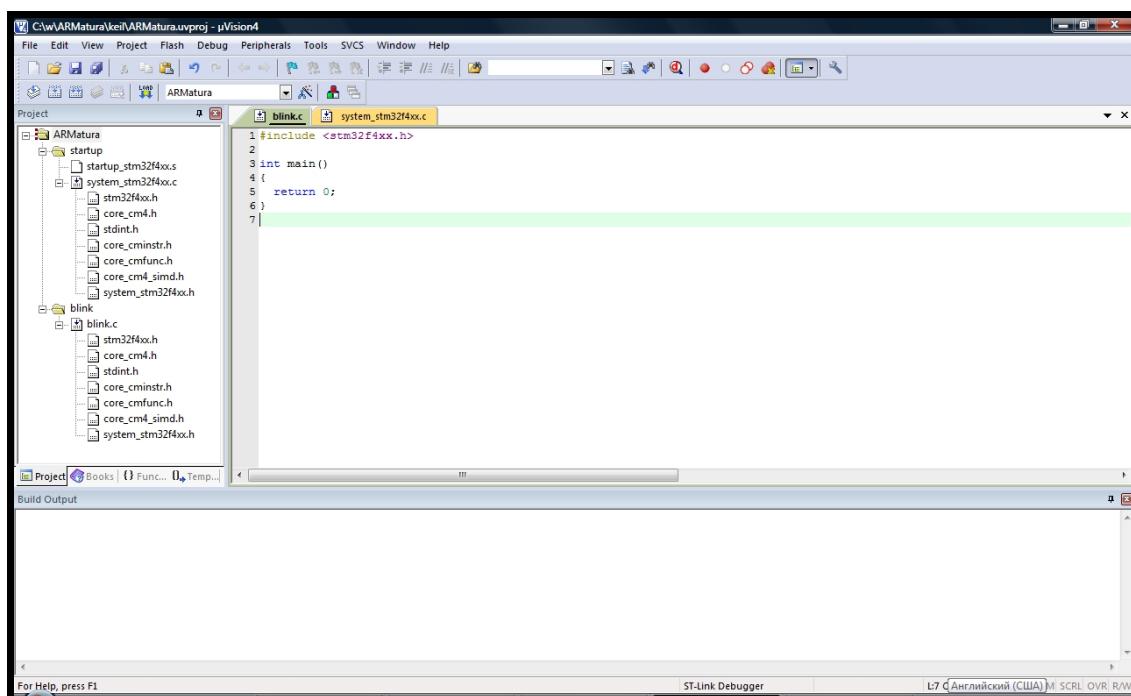
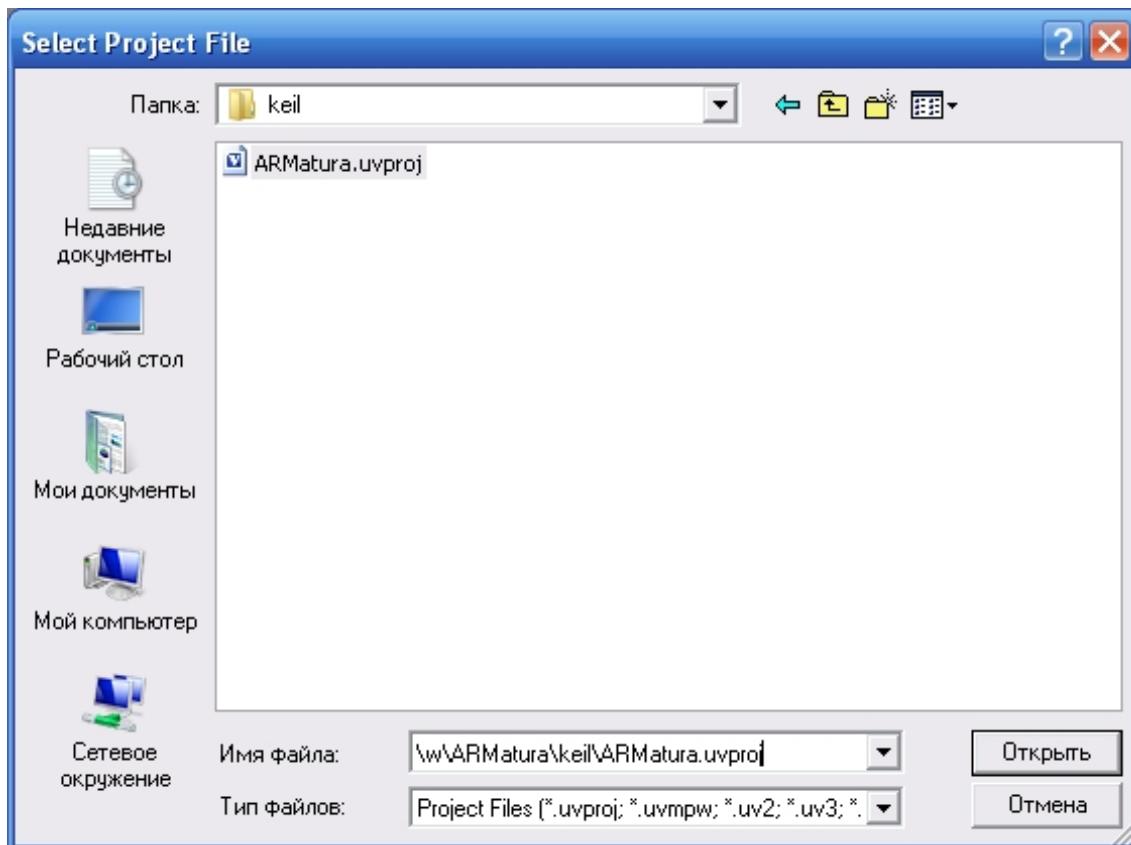


и открыть первый самый простой проект:

Глава 7

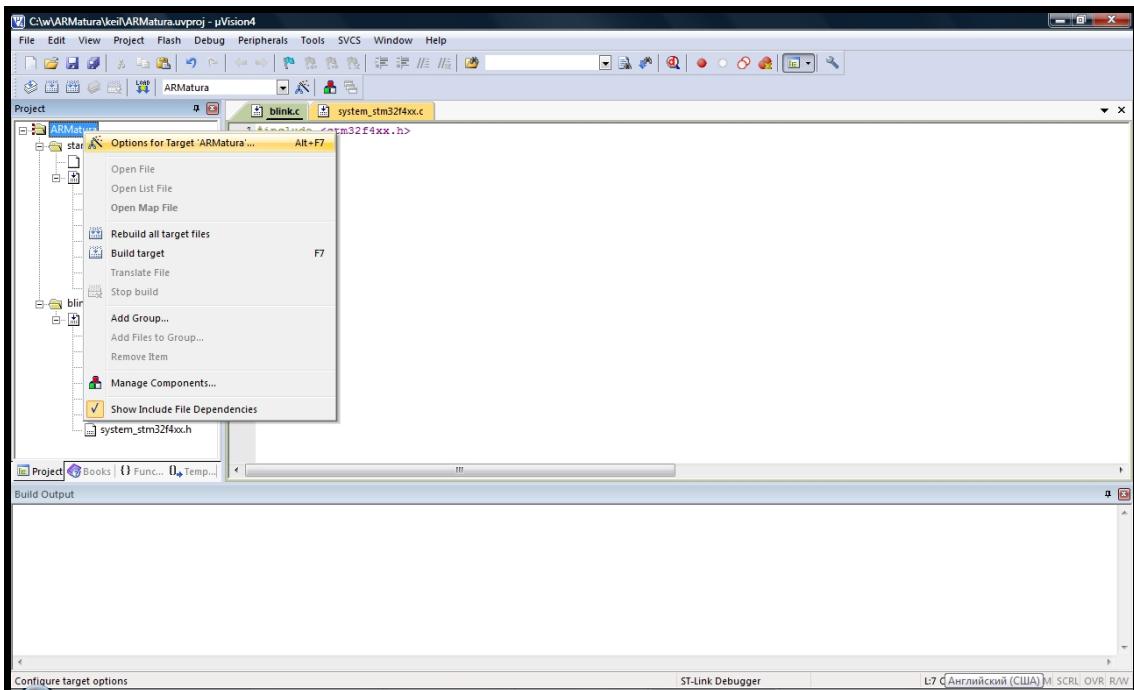
Первый проект: blink



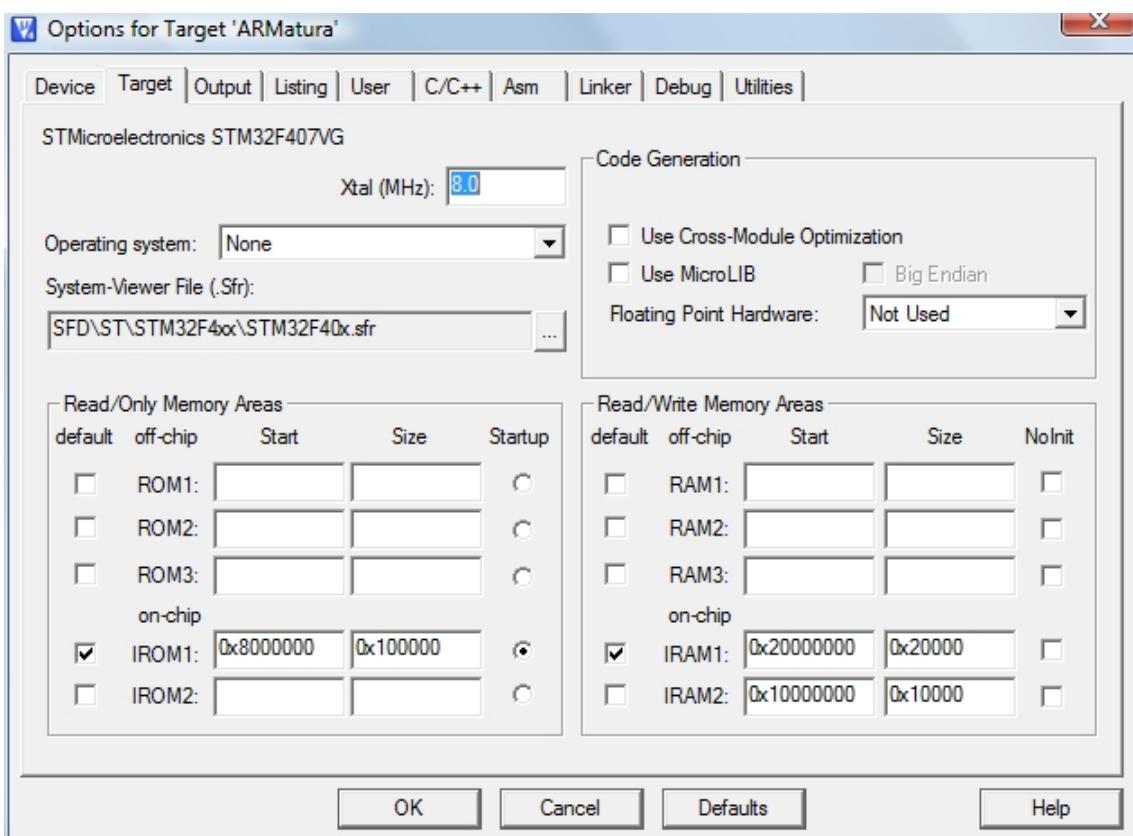


7.1 Настройки проекта в Keil

Настройки проекта вызываются из меню **Project > Options for Target 'ARMatura' ...**, комбинацией клавиш **Alt + F7**, или выбором аналогичной опции из контекстного меню, вызываемого щелчком правой кнопкой мыши на корне дерева проекта **ARMatura** в левом окне **Project**.



По умолчанию открывается вкладка **Target:**



STM32F407VG 41

Xtal (MHz) 8.0

текущий процессор

частота внешнего кварца

на платах Discovery обычно стоит 8

или OCPB Keil RTX

использовать libc от Keil

для Cortex-M4 доступен аппаратный FPU

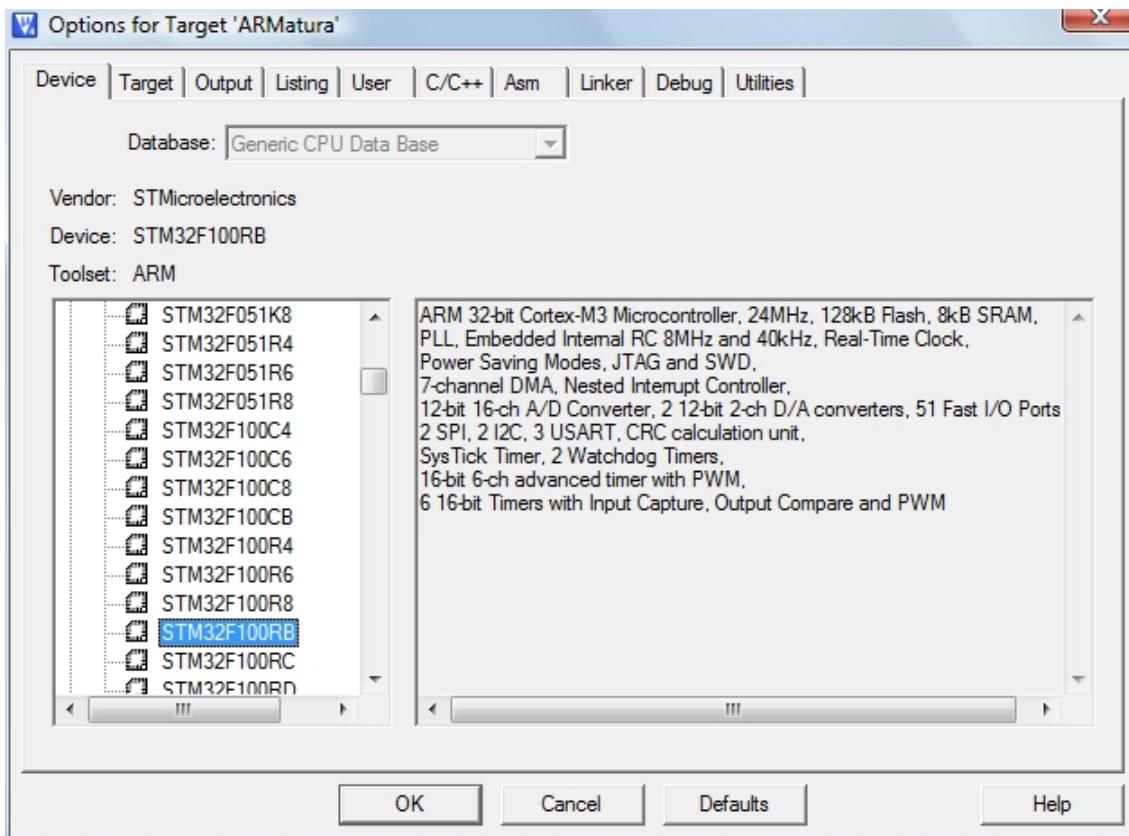
Operating system: None

Use MicroLIB

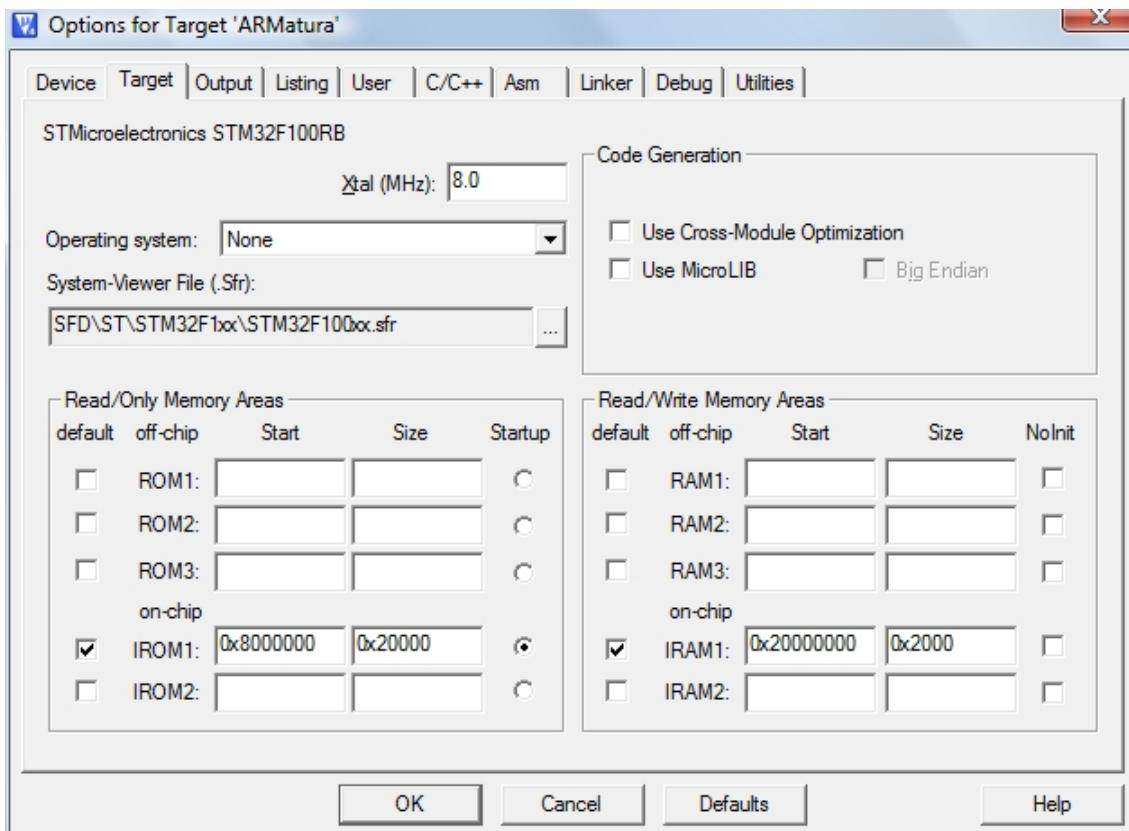
Floating Point Hardware: Not Used

На этой вкладке также прописывается карта встроенной и внешней памяти Flash/SRAM.

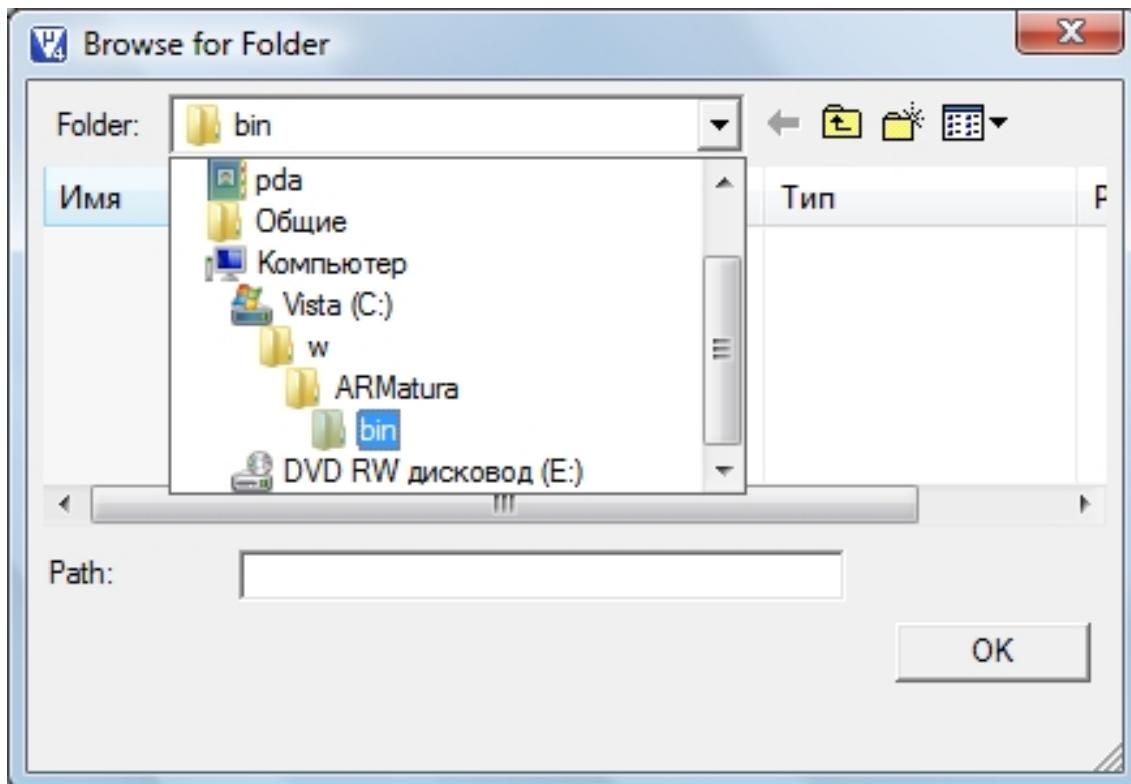
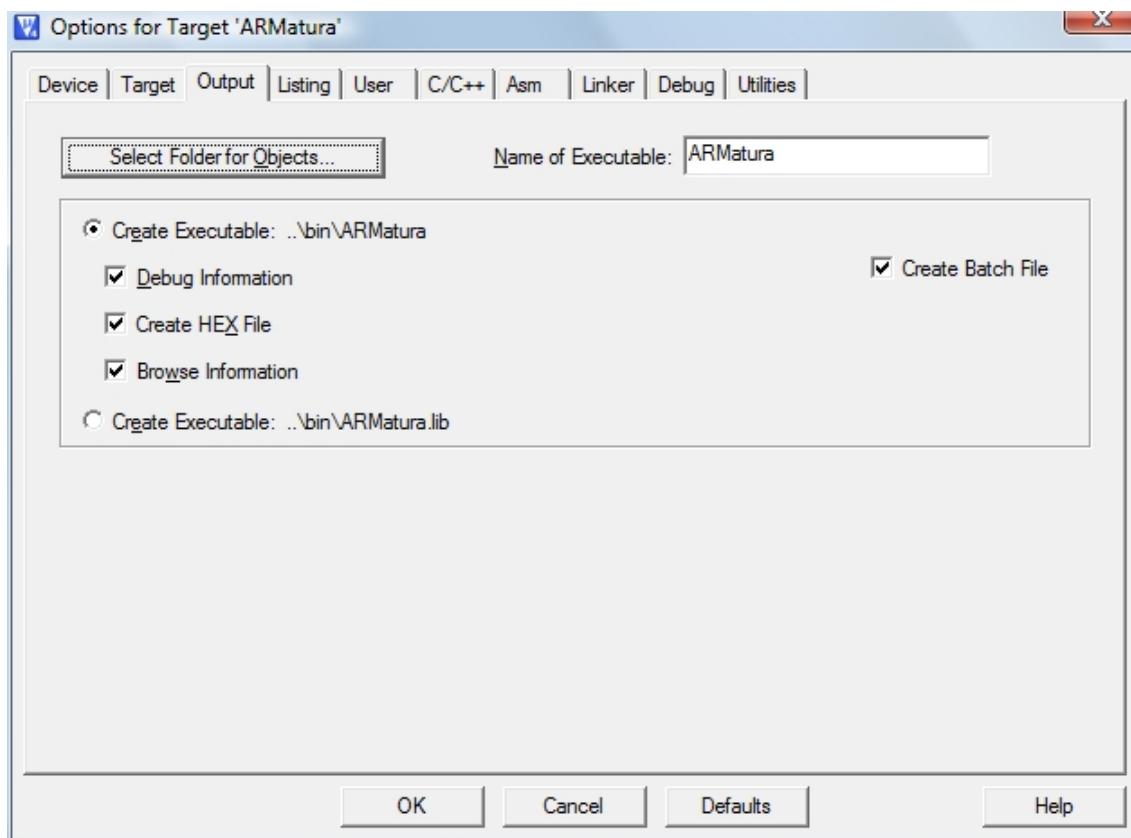
При необходимости собрать проект под другой процессор открываем вкладку **Device**, переконфигурируем проект под другую плату – STM32VLDISCOVERY 1:



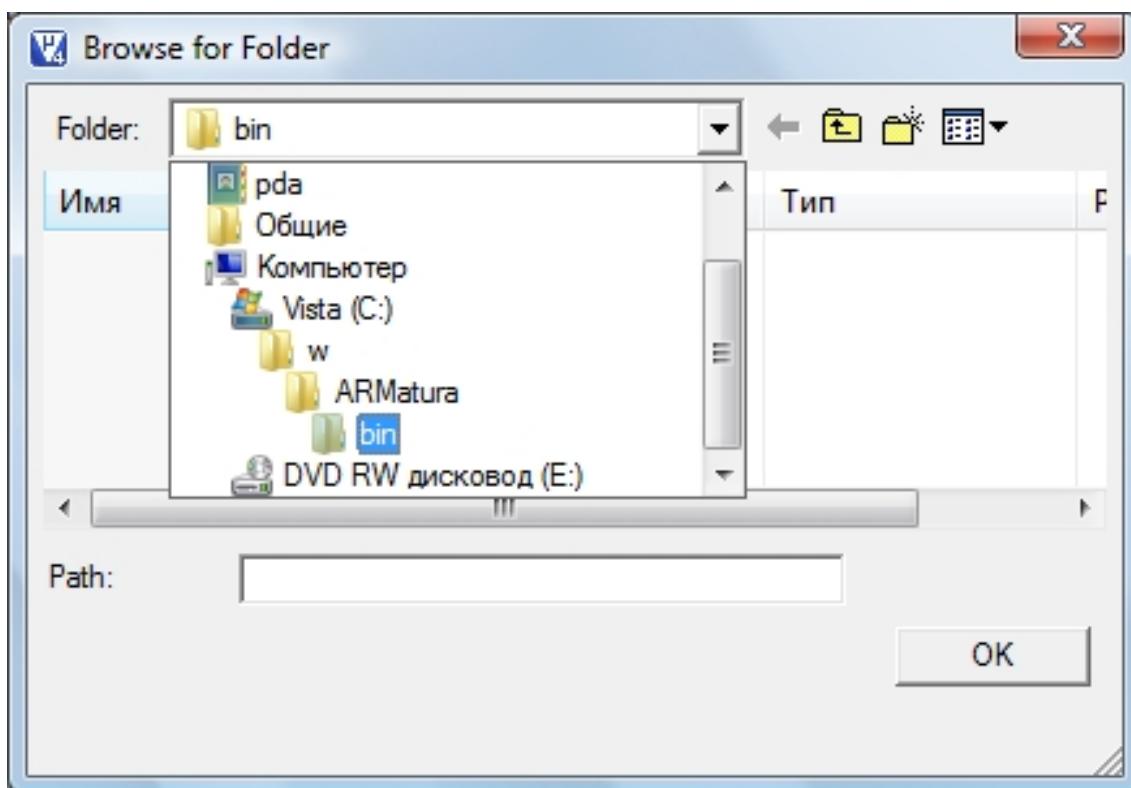
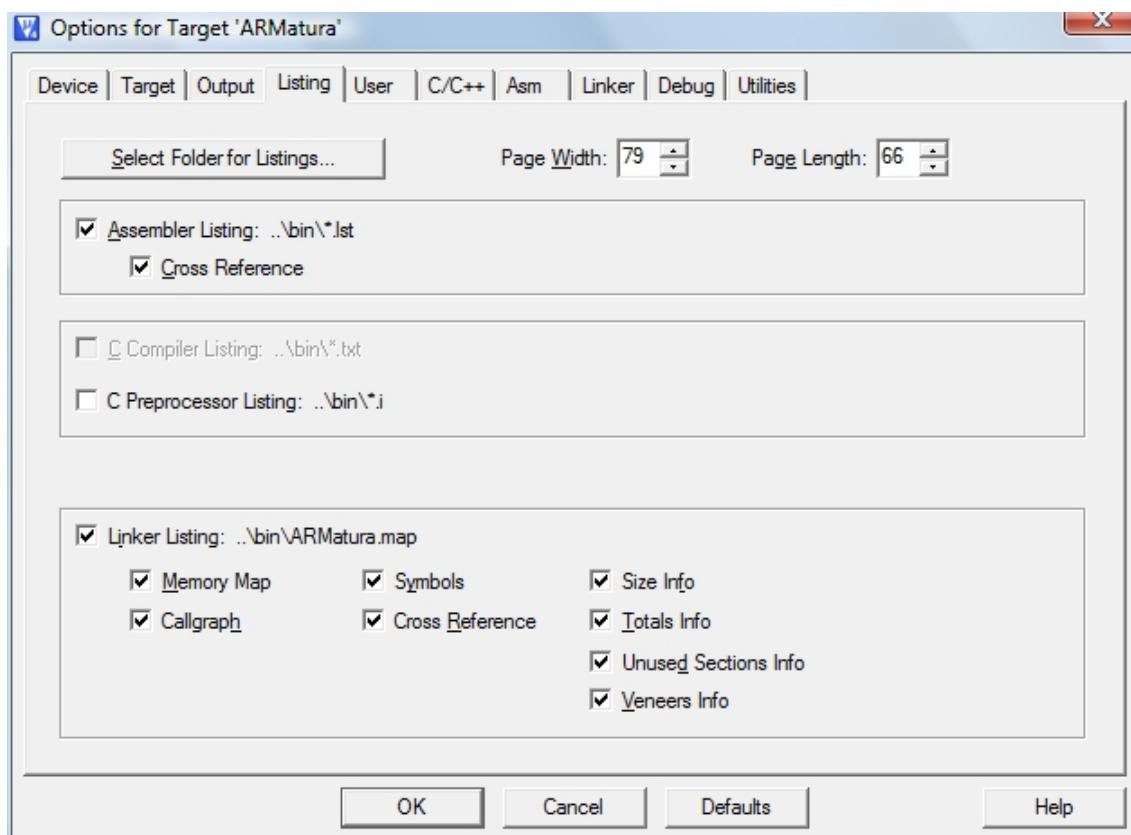
Обратите внимание что на вкладке **Target** изменился чип и настройки памяти:



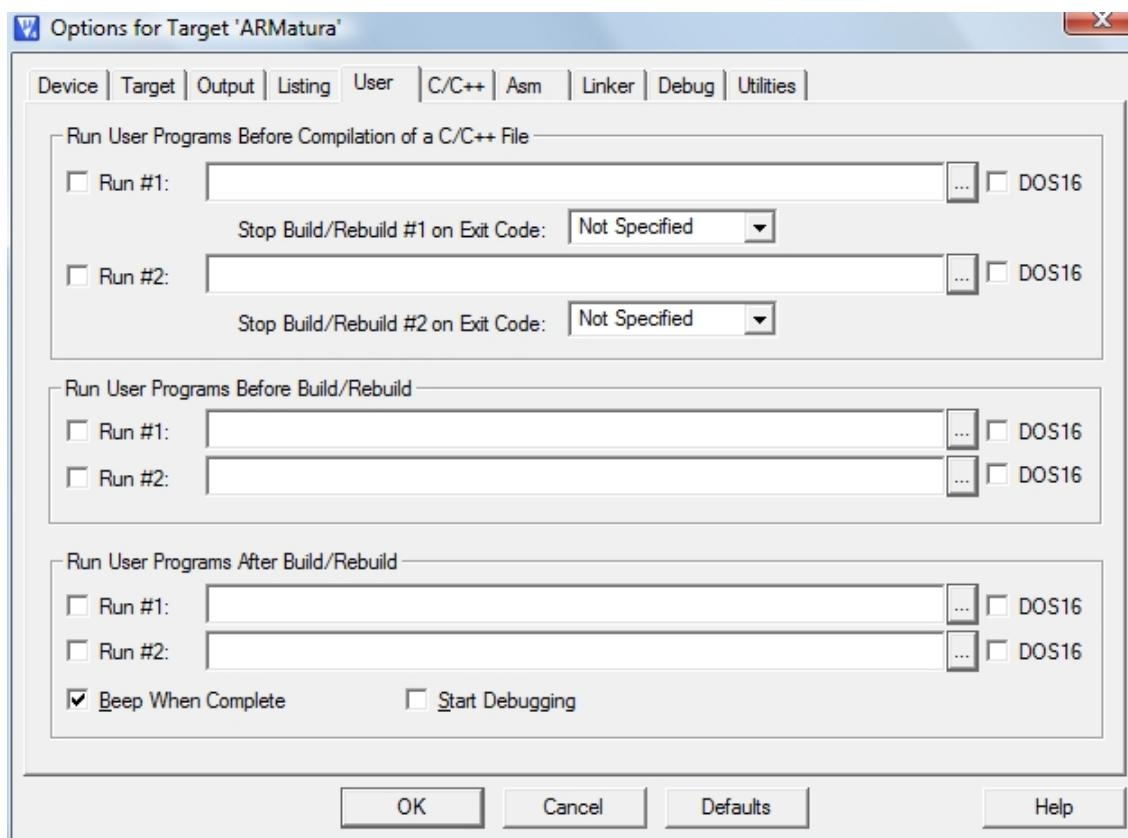
Output Настройки выходных файлов: каталог для бинарных файлов прошивки = firmware, при необходимости можно использовать .hex файл прошивки, имя файла прошивки



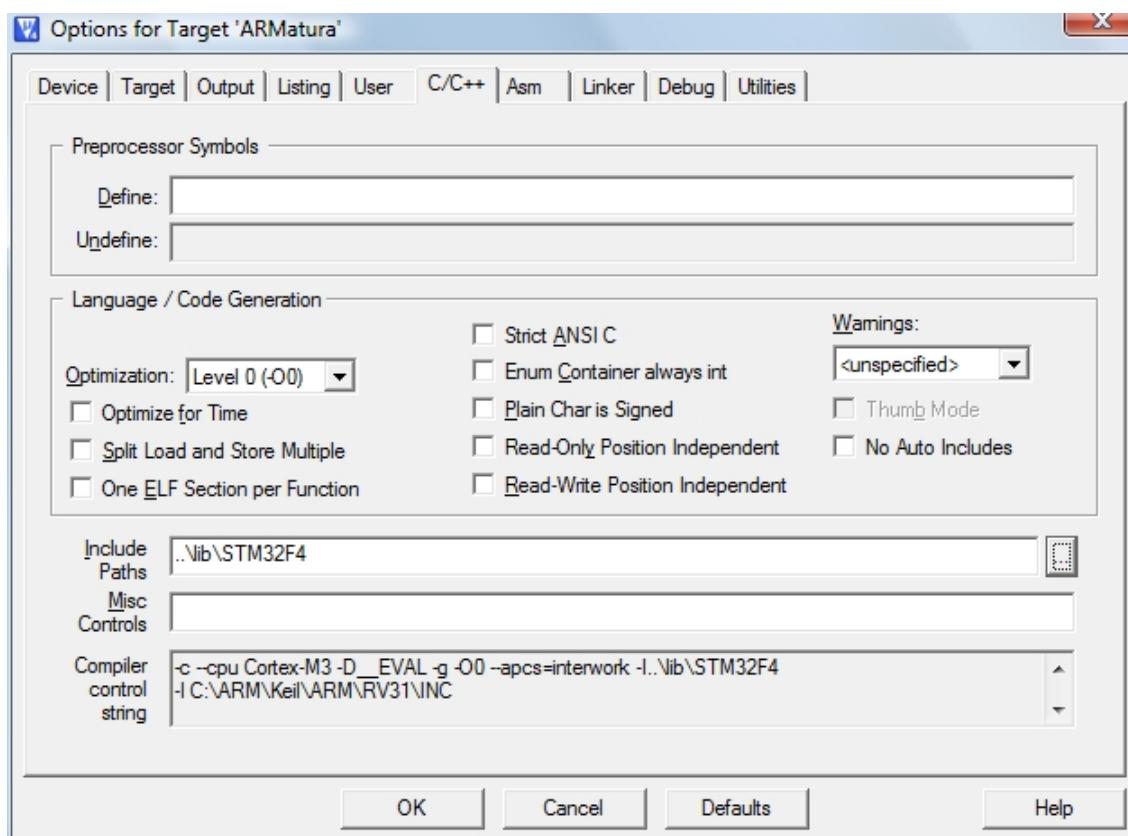
Listing Настройки генерации ассемблерных листингов в тот же каталог с прошивкой



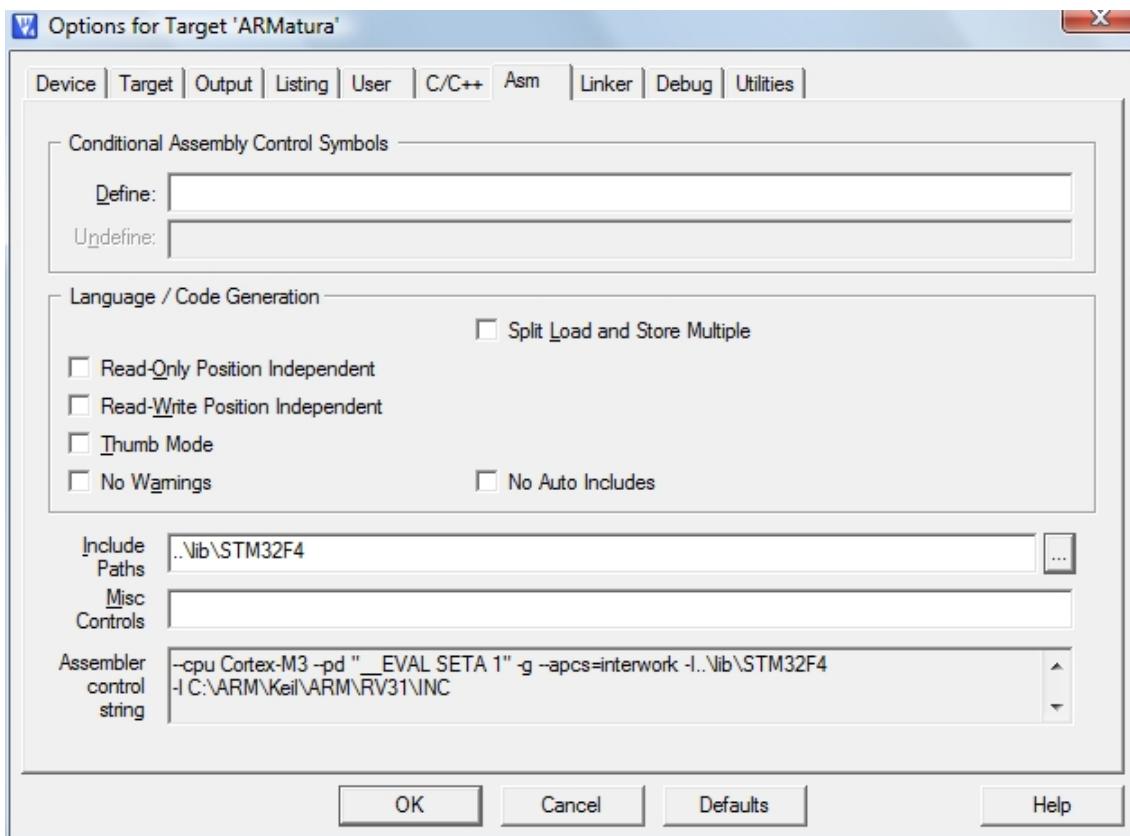
User Запуск пользовательских программ во время сборки проекта: пока не используем



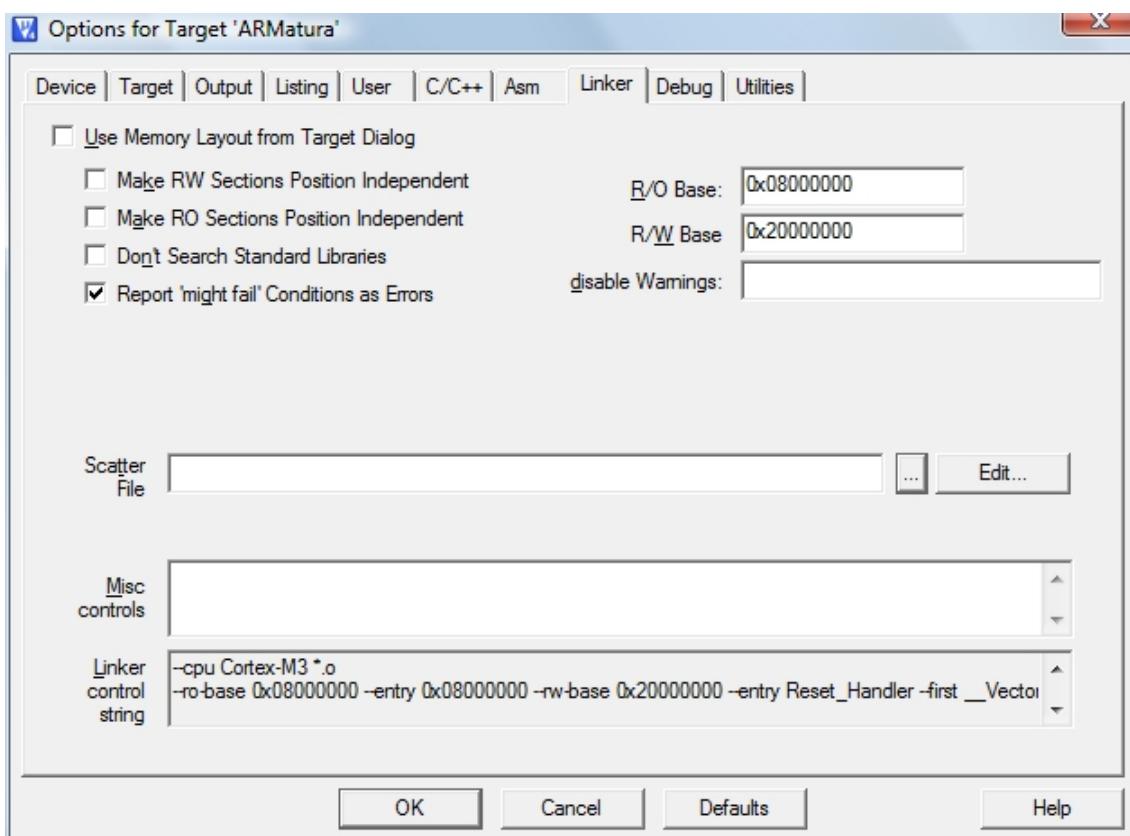
C/C++ Настройки компилятора C^{++} : опции оптимизации, каталоги библиотек и включаемых .h файлов, в нижнем поле прописывается полная команда строка вызова компилятора, которая может вам в дальнейшем понадобится, если потребуется компилировать без использования Keil IDE.



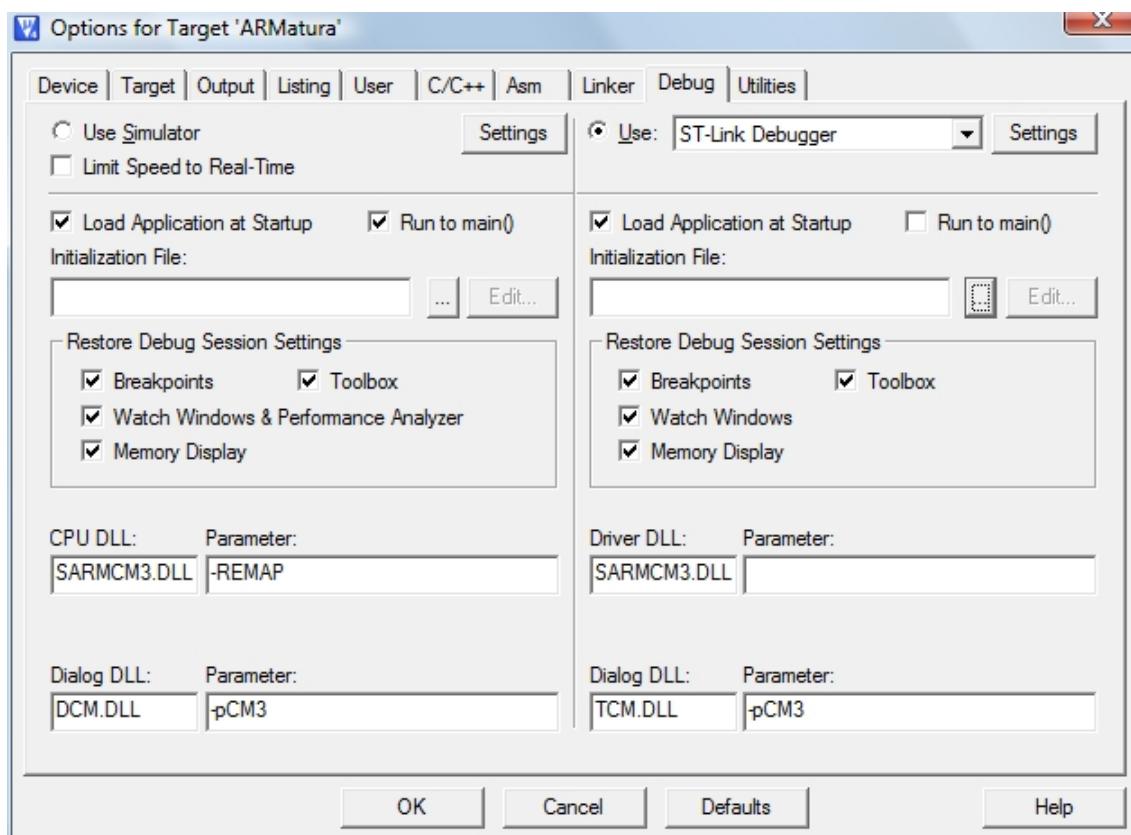
Asm Настройки ассемблера, приблизительно те же что и для C^{++}



Linker Настройки линкера, который собирает объектные файлы .o, в которые компилируются каждый программный файл (модуль) по отдельности, в один готовый файл прошивки (настройки карты памяти контроллера, адреса точки входа программы и т.п.)

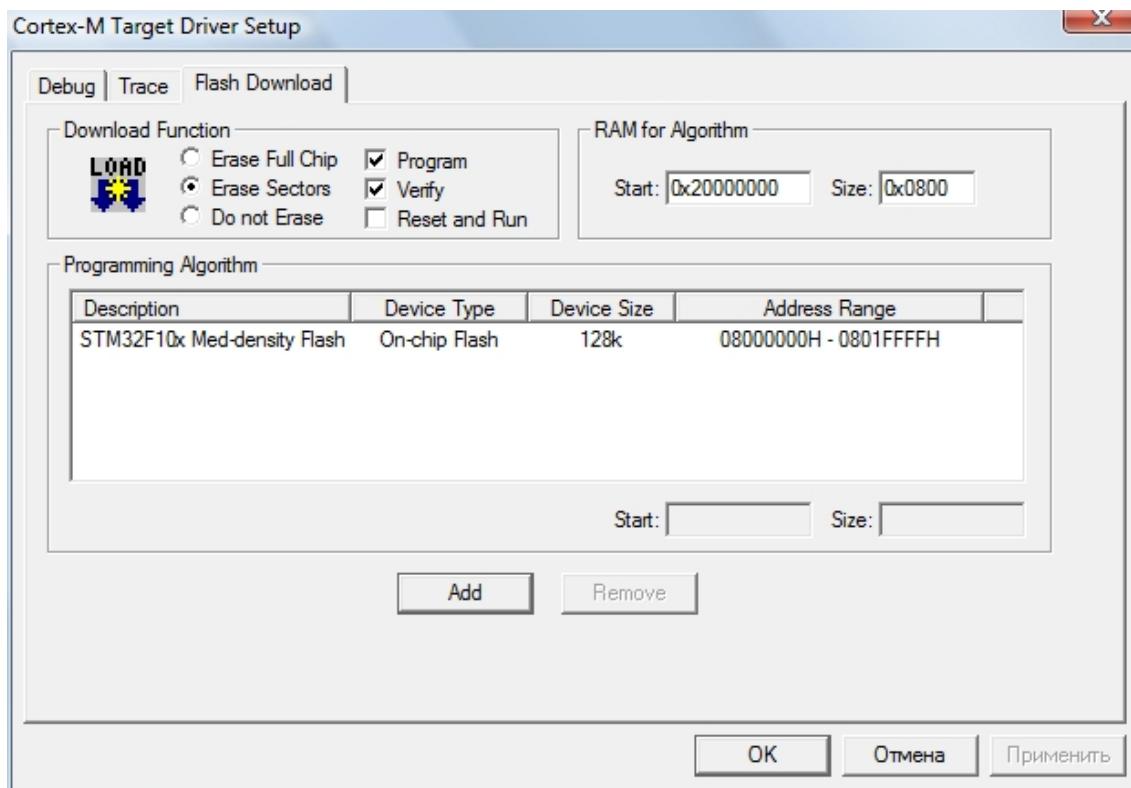


Debug Настройки отладки с помощью адаптера – STlink (внешний или встроенный в оценочную плату), JTAG.

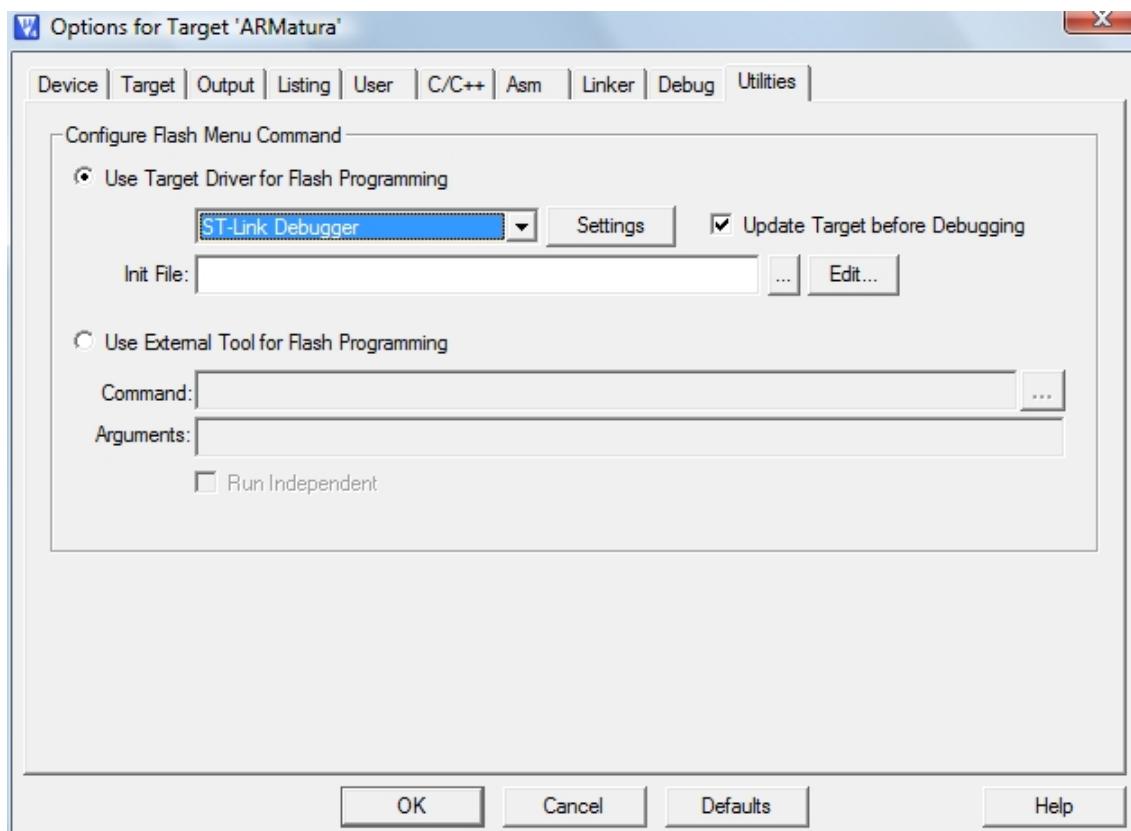


Настройки адаптера STlink – включаем режим SWD для варианта встроенного на оценочную плату, и обнаруживаем что не установили пакет поддержки STlink и драйвера.

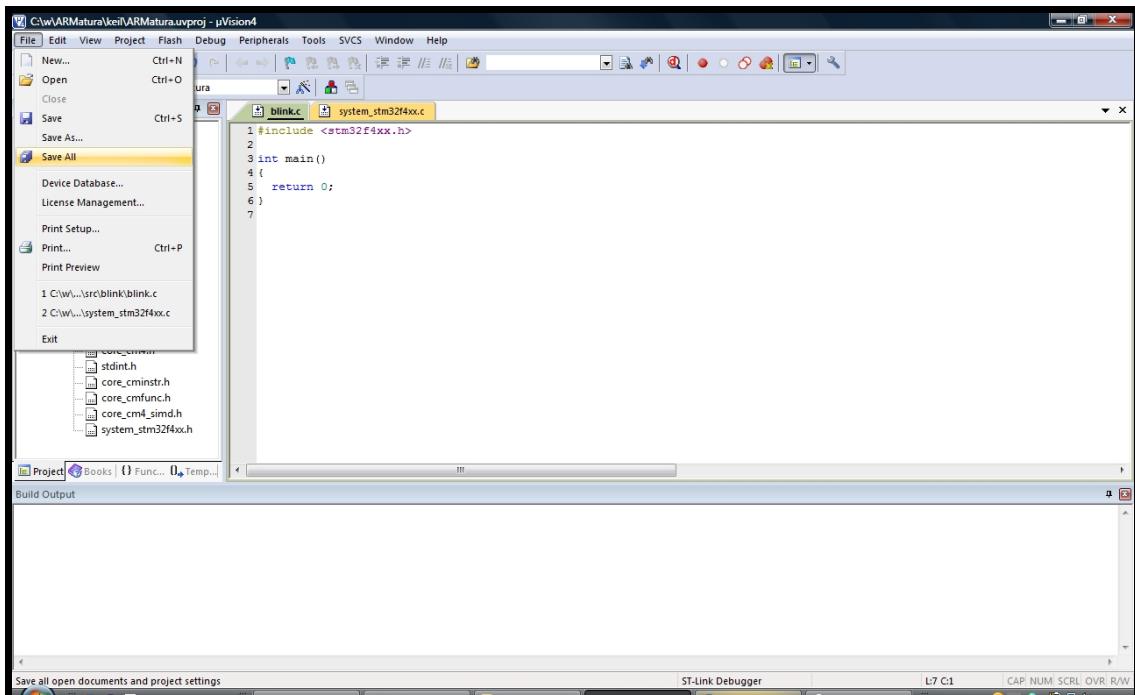




Utilities Настраиваем в качестве программатора для прошивки тот же STlink



Сохраняем настроенный проект



7.2 Структура файлов

ARMatura

- startup код инициализации ядра процессора
 - keil_startup_stm32f4xx.s ассемблерный код инициализации
 - system_stm32f4xx.c синый код инициализации STM32F4
 - system_stm32f4xx.h синый код инициализации STM32F4
 - stm32f4xx.h STM32F4 CMSIS Cortex-M4 Device Peripheral
 - core_cm4.h CMSIS библиотека поддержки ядра Cortex-M4
 - stdint.h стандартные целые типы C⁺⁺
 - core_cmInstr.h CMSIS Core Instruction Interface
 - core_cmFunc.h CMSIS Core Register Access Functions
 - core_cm4_simd.h CMSIS Cortex-M4 расширение SIMD/DSP
- blink
 - blink.c простая программа мигания светодиодом или дрыгания ногой
 - stm32f4xx.h

Глава 8

Hell Of World

Часть IV

Средства разработки

Глава 9

Keil MDK-ARM



Для начала освоения программирования для ARM рекомендуем использовать бесплатный пакет от Keil: <http://www.keil.com/arm/mdk.asp> — ограничения бесплатной версии в 32К кода вполне достаточно для начального освоения программирования под процессоры семейства Cortex-Mx, а затем уже можно переползать на открытое ПО: GNU toolchain 10.1, Eclipse 11.1 и Linux XIII.

Процесс установки и первоначальной настройки описан в 9.

Глава 10

Компиляторы

10.1 GCC

10.2 KeilCC

10.3 IAR

Глава 11

IDE

11.1 Eclipse

11.2 Code::Blocks

11.3 gVim

11.4 Keil uVision

11.5 IAR

Глава 12

Программаторы

12.1 STlink

12.2 Serial Boot

Глава 13

Отладчики

13.1 JTAG

13.2 STM32 SWD

13.3 GDB

STlink gdbserver

OpenOCD

Часть V

Основы языка C^{+^+}

Глава 14

Синтаксис

Глава 15

Типы данных

Глава 16

Стандартная библиотека libc

Часть VI

Отладка

Глава 17

JTAG

Глава 18

GDB

Глава 19

OpenOCD

Часть VII

CMSIS

Глава 20

Startup

Глава 21

Стандартная библиотека STM32

Глава 22

USB client/host

Часть VIII

Ядро Cortex-Mx

Глава 23

Режимы ARM и Thumb

Глава 24

DMA

Глава 25

DSP /Cortex-M3/

Глава 26

FPU /Cortex-M4F/

Часть IX

Интерфейсы

Глава 27

USB

Глава 28

UART

Глава 29

SPI

Глава 30

I2C

Глава 31

CAN

Часть X

Операционные системы ОСРВ

Глава 32

Keil RTX

Глава 33

FreeRTOS

Глава 34

eCos

Глава 35

Linux

подробно рассмотрен в отдельном разделе XIII

Часть XI

Стек TCP/IP

Глава 36

Ethernet

Глава 37

PPP

Часть XII

Типовые применения

Глава 38

GPS

38.1 Tripod15

38.2 WISMO228

Глава 39

GSM

39.1 WISMO228

Глава 40

шина Dallas 1Wire

40.1 RTC

40.2 Датчики температуры DS18x20

Часть XIII

Встраиваемый Linux

Часть XIV

Приложения

Глава 41

Сводная таблица процессоров

	ядро Cortex-	MHz	Flash	SRAM	корпус LQFP	USB	UART	SPI	CAN
STM32F100C4T6B	M3	24	16K	4K	48		2	1	
STM32F100RBT	M3	24	128K	8K	100		1		
STM32F103RBT	M3				100	1	1		
STM32F407VGT	M4F	168	1M	192K	144	2	6		2
STM32F407IGT	M4F	168	1M	192K	176	2	8		2
STM32F427IIT	M4F	168	2M	256K	176	2	8		2

41.1 STM32F10x

STM32F100C4T6B

Ядро	Cortex-M3
Flash	16K
SRAM	4K
16-битные таймеры	6
таймеры ШИМ	3
RTC	да
UART	2
SPI	1
I2C	1
DMA	1 канал
АЦП	10x12 бит
ЦАП	2x12 бит
корпус	LQFP48