Настройка ПО для ARM

# Введение

Ниже приводится список необходимого ПО под Windows для работы с STM32-Discovery. Это отладочная плата со встроенным программатором/отладчиком, с процессором STM32F100RB. Разумеется, приведенные инструкции касаются и остальных контроллеров ST.

# Программное обеспечение

* Компилятор etc.: YAGARTO <http://yagarto.org/>
* Инструментарий make.exe.
* Среда разработки: Netbeans <http://netbeans.org/>
* Программа для программатора:

<http://www.st.com/internet/com/SOFTWARE_RESOURCES/TOOL/DEVICE_PROGRAMMER/um0892.zip>

* Плагин для связи NetBeans с YAGARTO: net-antistatix-yagarto-flavor.nbm
* Если нужна отладка – придется обзавестись gdb-server с поддержкой SWD.
* Мой makefile.
* Библиотека стандартной периферии – можно взять с st.com, но лучше у меня.

# Установка

* YAGARTO ставим в корень диска, лучше С. Важно, чтобы в пути не было пробелов. C:\YAGARTO – лучший выбор.
* Инструментарий make.exe ставим в корень диска, лучше С. Важно, чтобы в пути не было пробелов. C:\make – лучший выбор.
* Убеждаемся, что в переменной окружения Path присутствуют C:\YAGARTO\bin и C:\make. При отсутствии – добавляем.
* NetBeans ставим куда оно захочет.
* Устанавливаем плагин в NetBeans: Tools-Plugins-Downloaded-Add- указываем файл.
* Устанавливаем ST-Link Utility куда оно захочет.
* Перегружаемся, чтобы заработал Path.

# Настройка NetBeans

Я рекомендую сделать template-проект, настроить его, а потом просто копировать куда нужно и переименовывать по необходимости. Это связано с тем, что нужно указать большое количество всяких файлов и прочего. В сущности, настройка NetBeans сводится к настройке свойств проекта.

1. Создаем проект: New Project – C/C++ - C/C++ Application - Next
2. Указываем имя и рабочую директорию (далее я буду ее называть РД), ToolCollection = Yagarto, жмем Finish.
3. Автоматически созданный makefile удаляем, ничего хорошего там всё равно нет. Копируем в РД мой makefile. Добавляем его в Important Files.
4. Копируем в РД папку ***lib***. Это ST-шная библиотека периферии.
5. Копируем в РД файлы ***system\_kl.h***, ***system\_kl.c***, ***stm32f10x\_conf.h***. Это модифицированные мной стандартные файлы, там указан нужный процессор.
6. В NetBeans добавляем скопированные файлы куда нужно – в sources и headers.
7. Создаем в РД папки ***obj*** (туда будут помещены промежуточные результаты компиляции) и ***output*** (туда будет записан файл прошивки).
8. Настраиваем проект. Либо File-Project properties, либо правой кнопкой по имени и properties.
   1. General
      1. Configuration – all configurations
      2. Source folders: Add – lib
      3. Encoding – Windows 1251
   2. Build-C compiler
      1. Include dirs: lib;lib/CoreSupport;lib/inc;. (точка в конце – должна быть)
      2. Preprocessor definitions: добавить STM32F10X\_MD\_VL и USE\_STDPERIPH\_DRIVER
   3. Build-C++ Compiler
      1. Сделать то же, что для Build-C compiler
   4. Build-Linker
      1. Output: output/output.elf