STM32

STM32 — семейство 32-битных микроконтроллеров производства STMicroelectronics.

https://wiki.st.com/stm32mpu/

Оборудование

STLINK-V3MINI

STLINK-V3MINI — компактный внутрисхемный программатор и отладчик для STM32.

STLINK-V3MINI

Определяется так:

```
0483:374e STMicroelectronics STLINK-V3
```

Имя UART порта: `/dev/ttyACM0`

Product Specification (DB3737), оригинал

UM2502 User manual, оригинал

3D–модель корпуса: STLinkV3 Mini Enclosure by JRad

Адаптер

Разъёмы

STDC14

```
┌──────────────────────┐
│  2  4  6  8  10 12 14 │
│  •  •  •  •  •  •  •  │
│                       │
│  •  •  •  •  •  •  •  │
```

```
│  1  3  5  7  9  11 13 │
└──────────   ──────────┘
```

№ контакта	Назначение
1	—
2	
3	VCC [1]
4	TMS/SWDIO
5	GND
6	TCK/SWCLK
7	GND
8	T_JTDO/T_SWO [2]
9	T_JCLK
10	T_JTDI/NC [3]
11	GNDDetect [4]
12	T_NRST
13	T_VCP_RX
14	T_VCP_TX

ARM20

IDC-20

```
┌──────────────────────────────────┐
│  2  4  6  8  10 12 14 16 18 20   │
│  •  •  •  •  •  •  •  •  •  •    │
│                                  │
│  •  •  •  •  •  •  •  •  •  •    │
│  1  3  5  7  9  11 13 15 17 19   │
└──────────────   ───────────────┘
```

№ контакта	Назначение
1	VCC
2	VSUPPLY
3	nTRST
4	GND
5	TDI
6	GND
7	TMS/SWDIO
8	GND
9	TCK/SWCLK
10	GND
11	RTCK [5]
12	GND
13	TDO/SWO
14	GND
15	nSRST (NRST)
16	GND
17	DBGRQ

№ контакта	Назначение
18	GND
19	DBGACK
20	GND

Соответствие контактов на разъёмах:

№ контакта		Назначение
STDC14	ARM20	
3	1	VTref
4	7	TMS/SWDIO
5	GND	
6	9	TCK/SWCLK
7	GND	
8	13	TDO/SWO
9	11	T_JCLK / RTCK Fix Me!
10	5	T_JTDI/NC / TDI Fix Me!
11	GNDDetect [6]	
12	15	NRST
13	—	RX
14	—	TX

STM32MP157A

STM32MP157A - STMicroelectronics

STM32MP15 resources - stm32mpu

Тип	Код	Название	Версия	Дата	Ссылка
Product Specifications	DS12504	Arm® dual Cortex®-A7 800 MHz + Cortex®-M4 MPU, 3D GPU, TFT/DSI, 37 comm. interfaces, 29 timers, adv. analog	6.0	2021-05-18	оригинал
Reference Manuals	RM0436	STM32MP157 advanced Arm®-based 32-bit MPUs	5.0	2021-01-21	оригинал
Application Notes	AN5031	Getting started with STM32MP151, STM32MP153 and STM32MP157 line hardware development	3.0	2021-01-21	оригинал
	AN5275	USB DFU/USART protocols used in STM32MP1 Series bootloaders	1.0	2019-10-15	оригинал
	AN5168	STM32MP1 series DDR configuration	2.0	2021-10-19	оригинал
Programming Manuals	PM0214	STM32 Cortex®-M4 MCUs and MPUs programming manual	10.0	2020-03-23	оригинал
Errata sheets	ES0438	STM32MP15xx device errata	6.0	2021-02-25	оригинал

Develop on Arm® Cortex®-M4 - stm32mpu

Программы

ST-MCU-FINDER-PC	STM32 and STM8 product finder for desktops

STM32CubeProgrammer

STM32CubeProgrammer — это официальный инструмент от STMicroelectronics для создания разделов на любом накопителе, доступном на платформе STM32.

После создания STM32CubeProgrammer позволяет записывать и обновлять разделы предварительно созданными двоичными файлами.

Соединение между ПК и платой с STM32 может быть установлено через UART или USB.

STM32CubeProg

STM32CubeProgrammer — Wiki

UM2237 STM32CubeProgrammer software description

, оригинал

Возможно понадобится установить пакет `libusb`

```
apt install libusb-1.0-0
```

После установки нужно добавить правила udev:

```
#!/bin/sh
cd ~/STMicroelectronics/STM32Cube/STM32CubeProgrammer/Drivers/rules
sudo cp *.* /etc/udev/rules.d/
```

правила от STM32CubeProgrammer v.2.8.0

Версия 2.8.0 не работает в Debian GNU/Linux — нужно использовать 2.7.0 или 2.6.0.

Linux - STM32CubeProgrammer 2.8.0 crashes clicking connect

How to find the DEVICE_PORT_LOCATION parameter value for the USB link

STSW-LINK009 - ST-LINK, ST-LINK/V2, ST-LINK/V2-1, STLINK-V3 USB driver signed for Windows7, Windows8, Windows10 - STMicroelectronics

Preparing the USB serial link for flashing

STM32CubeMX

STM32CubeMX is a graphical tool that allows a very easy configuration of STM32 microcontrollers and microprocessors, as well as the generation of the corresponding initialization C code

STM32CubeMX

https://wiki.st.com/stm32mpu/wiki/STM32CubeMX

UM1718 User manual. STM32CubeMX for STM32 configuration and initialization C code generation. оригинал

STM32MP15 - STM32CubeMX generated device tree

STM32CubeIDE

Integrated Development Environment for STM32.
STM32CubeIDE is an advanced C/C++ development platform with peripheral configuration, code generation, code compilation, and debug features for STM32 microcontrollers and microprocessors. It is based on the Eclipse®/CDT framework and GCC toolchain for the development, and GDB for the debugging. It allows the integration of the hundreds of existing plugins that complete the features of the Eclipse® IDE.

STM32CubeIDE

https://wiki.st.com/stm32mpu/wiki/STM32CubeIDE

- OpenSTLinux project support - Cortex®-A

How to install the Yocto Project SDK in STM32CubeIDE - stm32mpu

SW4STM32

Это старое решение — используйте STM32CubeIDE

The System Workbench toolchain, called SW4STM32, is a free multi-OS software development environment based on Eclipse, which supports the full range of STM32 microcontrollers and associated boards. The SW4STM32 toolchain may be obtained from the website www.openstm32.org, which includes forums, blogs, and trainings for technical support. Once registered to this site, users will get installation instructions at the Documentation > System Workbench page to proceed with the download of the free toolchain.

System Workbench for STM32: free IDE on Windows, Linux and OS X

Downloading the System Workbench for STM32 installer

Переход с Keil на SW4STM32 | Radiotech.kz

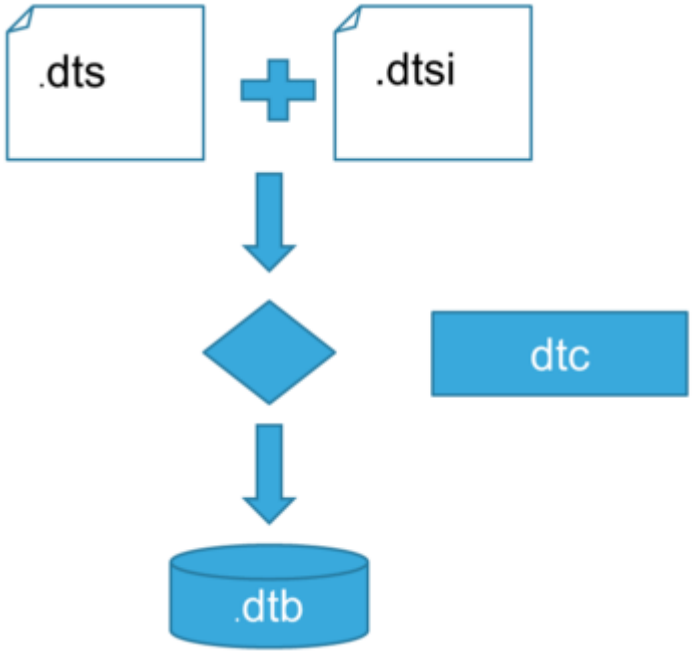
Настройка

Дерево устройств

Расширения файлов:

- `.dts` — источник дерева устройств (DTS). Этот формат представляет собой текстовое представление дерева устройств в форме, которая может быть обработана DTC (компилятором дерева устройств) в двоичный файл, ожидаемый программными компонентами: ядром Linux, U-Boot и TF-A.
- `.dtsi` — исходные файлы, которые могут быть включены из файла DTS
- `.h` — файлы заголовков, которые могут быть включены из файлов DTS и DTSI

Процесс построения дерева устройств:



оригинал
Инструмент под названием DTC (Компилятор дерева устройств) позволяет компилировать исходные файлы DTS в двоичный файл.

Для настройки нужно использовать STM32CubeMX.

Готовые файлы удобно редактировать, использую стандартный редактор в Midnight Commander, для этого создан файл подсветки синтаксиса devicetree.syntax. Для включение нужно добавить в настройки (подробнее о настройке здесь):

Syntax

```
file ..\*\\.(dts|dtsi)$ DeviceTree\sFile
include devicetree.syntax
```

Device tree

https://www.devicetree.org/

GPIO

Во время и сразу после сброса альтернативные функции не активны, а большинство портов ввода–вывода настроены на вход (input floating mode).

Когда контакт настроен как выход, значение, записанное в регистр выходных данных (GPIOx_ODR), выводится на контакт ввода–вывода. Выходной драйвер можно использовать в режиме push–pull или в режиме с открытым коллектором (open-drain), работает только низкий уровень, высокий уровень — HI-Z (Z–состояние).

Регистр входных данных (GPIOx_IDR) фиксирует данные, присутствующие на контакте ввода-вывода в каждом такте AHB(Advanced High–performance Bus).

13.3.1 General-purpose I/O (GPIO)

During and just after reset, the alternate functions are not active and most of the I/O ports are configured in input floating mode.

When the pin is configured as output, the value written to the output data register (GPIOx_ODR) is output on the I/O pin. It is possible to use the output driver in push-pull mode or open-drain mode (only the low level is driven, high level is HI-Z).

The input data register (GPIOx_IDR) captures the data present on the I/O pin at every AHB clock cycle

RM0436 — STM32MP157 advanced Arm®-based 32-bit MPUs

GPIO используется для настройки линий ввода–вывода. Каждая группа GPIO состоит из 8 линий (K и Z) или 16 линий (A–J).

Аббревиатуры регистров

В описаниях регистров используются следующие сокращения (x = A до H, K или Z) :

GPIOx_MODER	GPIO port mode register	
GPIOx_OTYPER	GPIO output type register	
GPIOx_OSPEEDR	GPIO output speed register	
GPIOx_PUPDR	GPIO port pull-up / pull-down register	
GPIOx_IDR	GPIO port input data register	чтение состояний (вход)
GPIOx_ODR	GPIO port output data register	запись состояний (выход)
GPIOx_BSRR	GPIO port bit set / reset register	побитная запись в регистр (выход)
GPIOx_LCKR	GPIO port configuration lock register	
GPIOx_AFRL	GPIO alternate function low register	
GPIOx_AFRH	GPIO alternate function high register	
GPIOx_ASCR	GPIO port analog switch control register	

GPIO internal peripheral

Pinctrl overview

Pinctrl device tree configuration

GPIO device tree configuration

How to control a GPIO in userspace

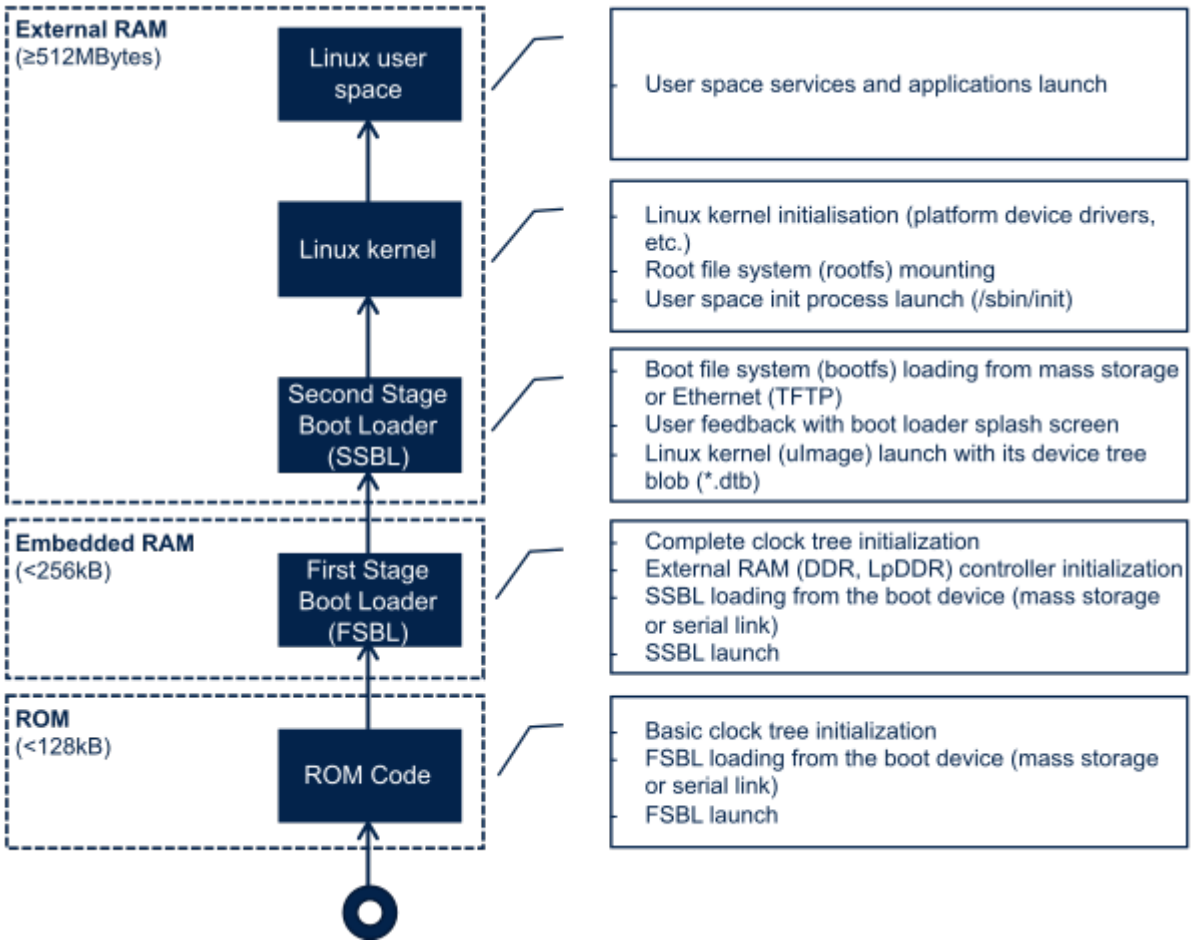
How to control a GPIO in kernel space

AN4899 STM32 microcontroller GPIO configuration for hardware settings and low-power consumption, оригинал

Загрузка

Запуск Linux на процессоре выполняется в несколько этапов, которые постепенно инициализируют периферийные устройства и память платформы.

Общая загрузочная цепочка Linux (указаны типичные размеры памяти для каждого этапа):



оригинал

Boot chain overview - stm32mpu

Код ПЗУ

Код ПЗУ (ROM code) — это часть программного обеспечения, которое хранится в ПЗУ. Он умещается в несколько десятков килобайт и отображает свои данные во встроенной оперативной памяти. Это первый код, выполняемый процессором, и он включает всю логику, необходимую для выбора загрузочного устройства (последовательное соединение или флэш–память), с которого первичный загрузчик (FSBL) загружается во встроенное ОЗУ. Для большинства продуктов требуется доверять приложению, работающему на устройстве, и код ПЗУ является первым звеном в цепочке доверия, которая должна быть установлена для всех запущенных компонентов: это доверие устанавливается путём аутентификации FSBL перед его запуском.

FSBL

First stage boot loader — первичный загрузчик. Он инициализирует (часть) дерева часов и внешнего контроллера ОЗУ, далее он загружает вторичный загрузчик во внешнее ОЗУ и передаёт ему управление.

В качестве такого загрузчика используется:

- TF-A (Trusted Firmware-A)

SSBL

Second-stage boot loader — вторичный загрузчик.

Он работает в большом объёме ОЗУ, поэтому он может работать с USB, Ethernet, дисплей и т. д., которые полезны для более гибкой загрузки ядра Linux (с флэш–устройства, сеть и т. д.) и удобны для пользователя (показывая пользователю экран-заставку).

Обычно d качестве такого загрузчика Linux во встроенных системах. используется:

- U-Boot

TF-A

TF-A (Trusted Firmware-A) разделен на несколько двоичных файлов, каждый из которых выполняет свою главную роль. Для 32-битных процессоров Arm (AArch32) доверенная загрузка разделена на четыре этапа (в порядке выполнения):

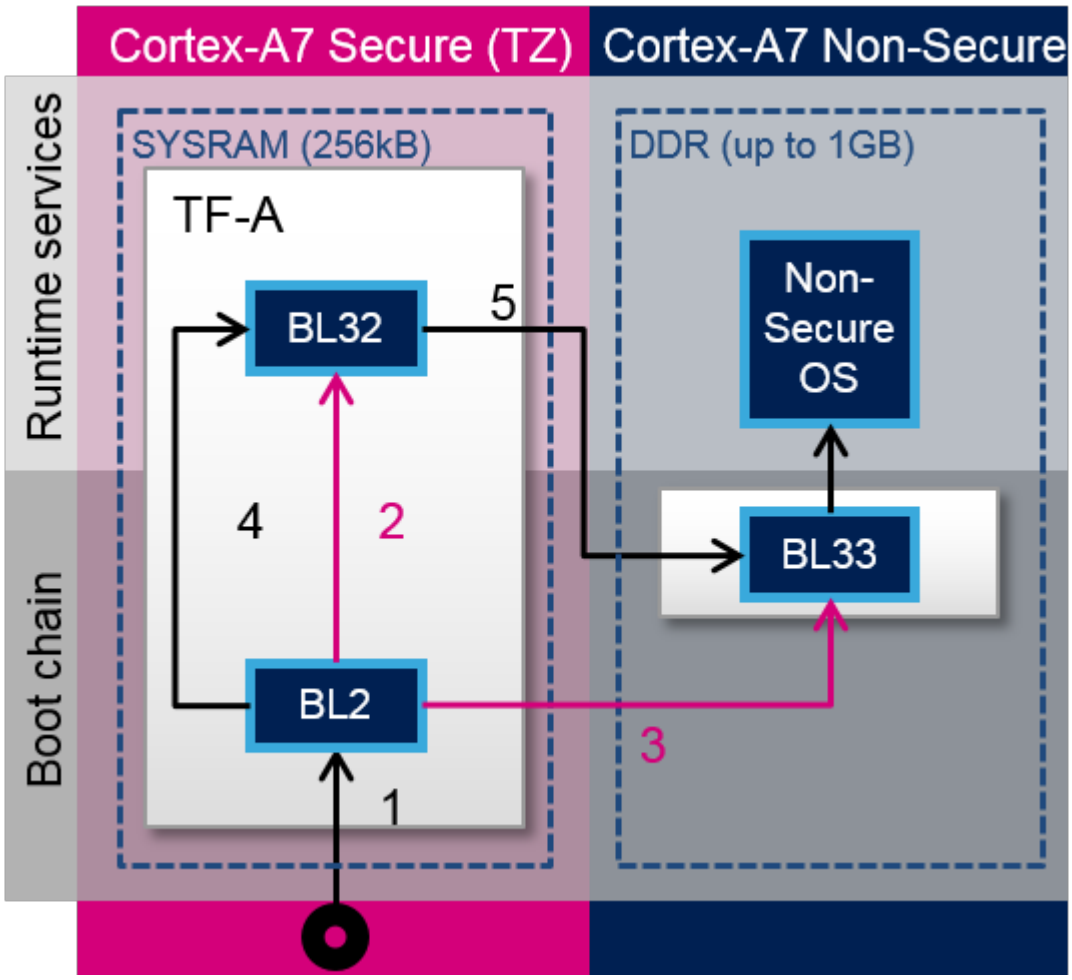
1. 1 (BL1) application processor trusted ROM
2. 2 (BL2) trusted firmware
3. 3-2 (BL32) runtime software
4. 3-3 (BL33) non-trusted firmware

BL1, BL2 и BL32 являются частями TF-A.

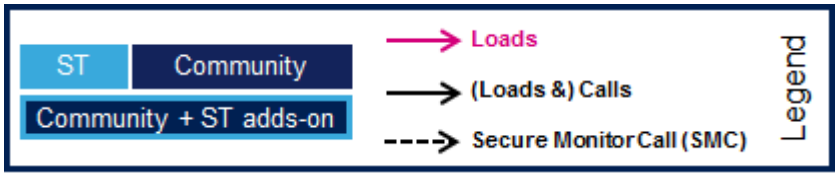
В виду того, что STM32 MPU используют специальный код ПЗУ, этап загрузки BL1 удаляется. Код ПЗУ ожидает, что BL2 будет работать на уровне выполнения EL3. Этот режим выбирается, когда включён флаг сборки BL2_AT_EL3.

BL33 находится за пределами TF-A. Это первый незащищённый код, загруженный TF-A. Во время загрузки это вторичный загрузчик (SSBL). Для платформ MPU STM32 SSBL по умолчанию является U-Boot.

TF-A может управлять своей конфигурацией с помощью дерева устройств. На этапе BL2 это сокращённая версия ядра Linux, в которой во время загрузки используются только необходимые устройства. Его можно настроить с помощью STM32CubeMX.



оригинал

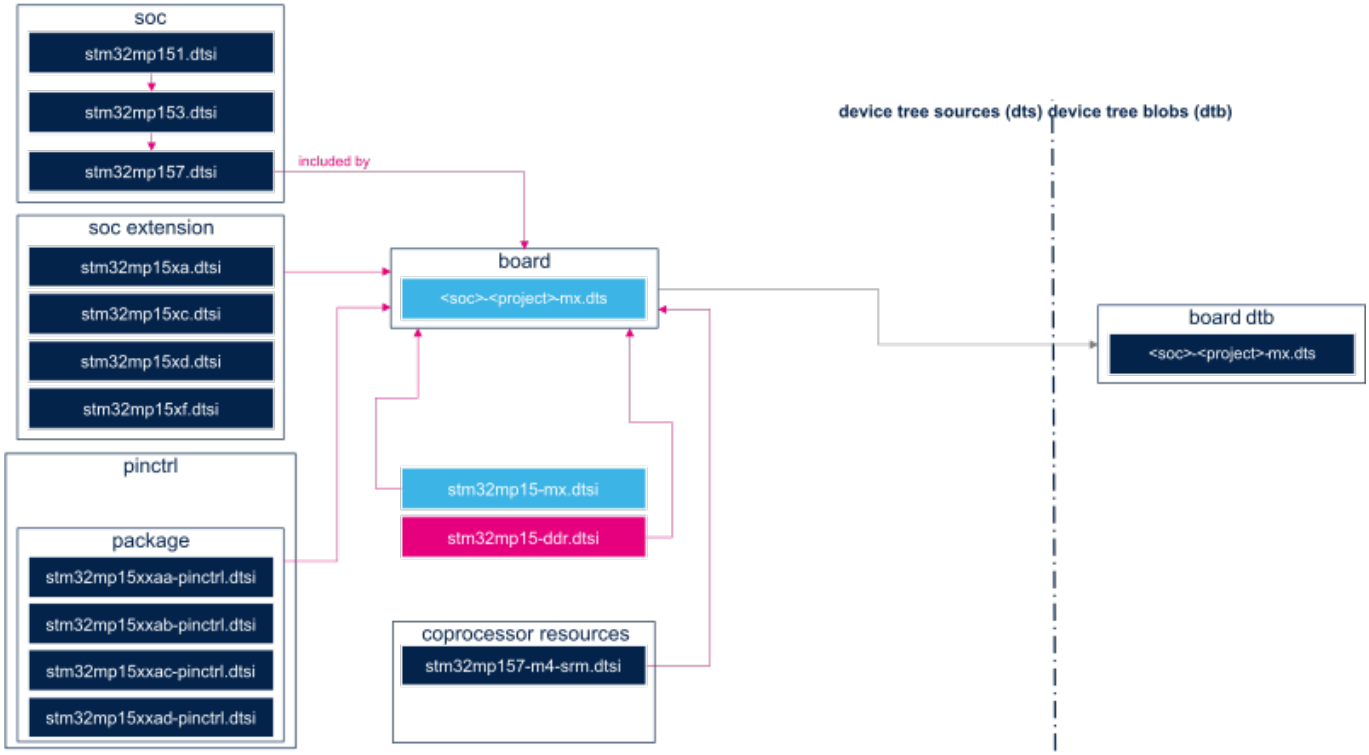


оригинал

Шаги загрузки TF-A:

1. Загрузка кода из ПЗУ и вызов BL2
2. BL2 загружает BL32
3. BL2 загружает BL33
4. BL2 вызывает BL32
5. BL32 вызывает BL33

Дерево устройств в STM32CubeMX:



оригинал

TF-A overview

TF-A device tree

U-Boot

Das U-Boot

U-Boot — это вторичный загрузчик (SSBL) для STM32.

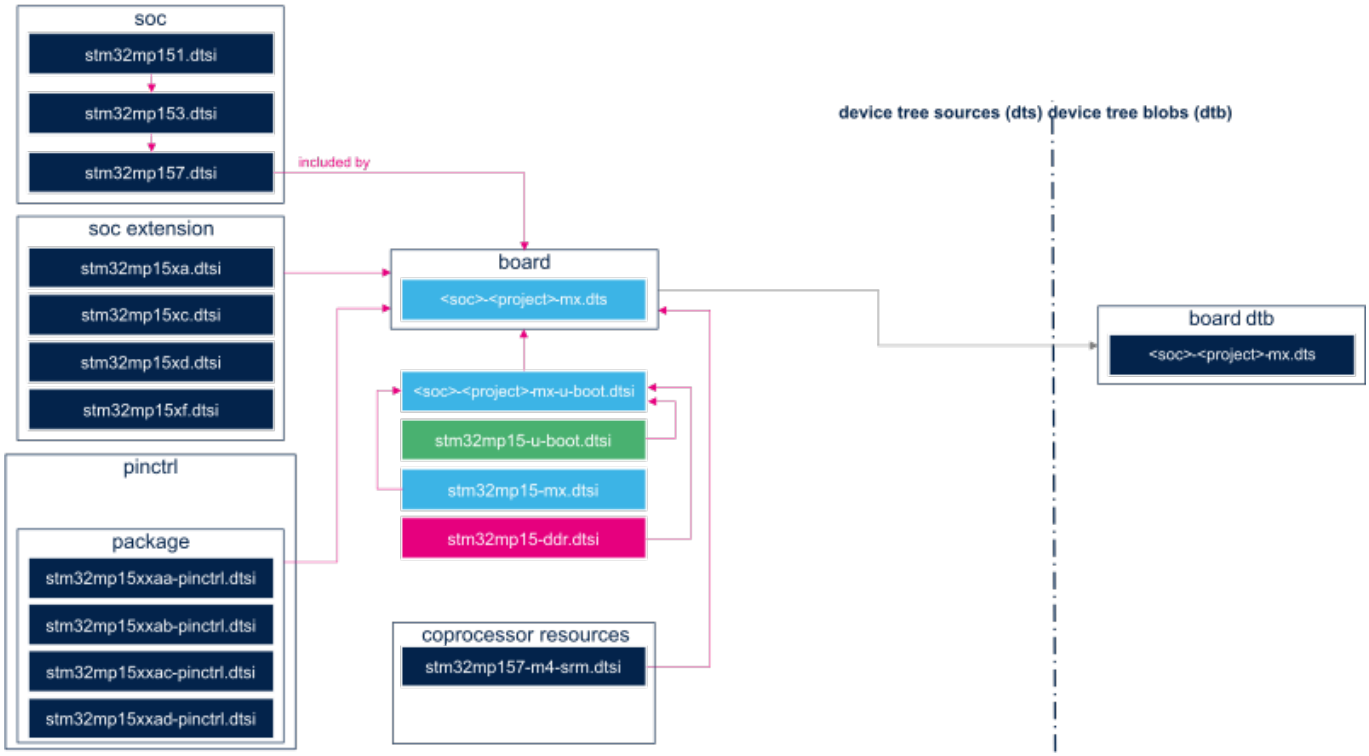
Основные характеристики SSBL:

- Настраиваемый
- Имеет простой интерфейс командной строки (CLI), позволяющий пользователям работать через консоль на последовательном порту
- Возможности создания сценариев
- Он загружает ядро в оперативную память и передаёт управление ядру
- Он управляет несколькими внутренними и внешними устройствами, такими как флэш–память NAND и NOR,

Ethernet и USB

- Он поддерживает следующие функции:
  - Файловые системы: FAT, UBI/UBIFS, JFFS
  - Стек TCP/IP: FTP
  - Дисплей: LCD, HDMI, BMP для заставки
  - USB: хост (запоминающее устройство) или устройство (стек DFU)

Дерево устройств в STM32CubeMX:

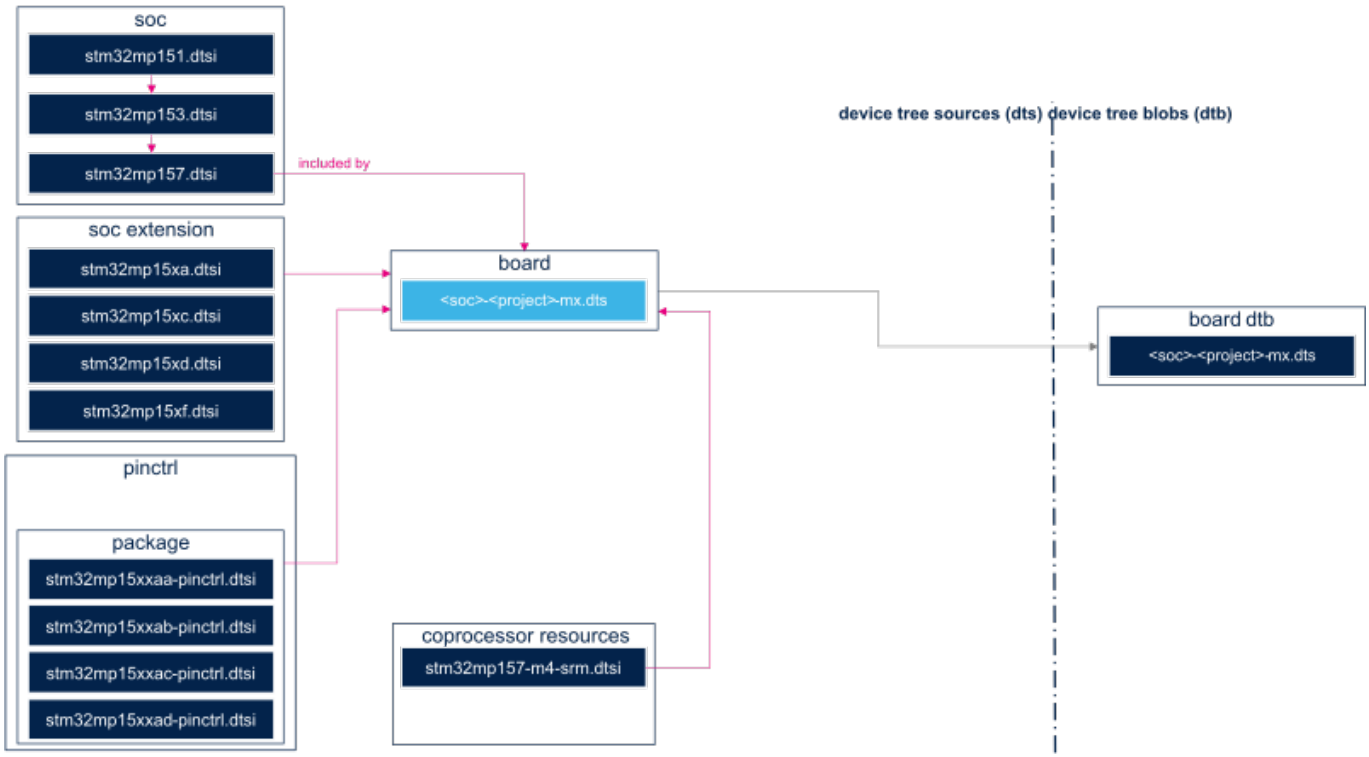


оригинал

- U-Boot overview
- U-Boot device tree
- How to configure U-Boot for your board

Linux

Дерево устройств в STM32CubeMX:



оригинал

/etc/apt/sources.list.d/packages.openstlinux.st.com.list

```
deb http://packages.openstlinux.st.com/3.1 dunfell main updates untested
```

Package repository for OpenSTLinux distribution

Документация

AN5031 Application note: Getting started with STM32MP151, STM32MP153 and STM32MP157 line hardware development, оригинал

STLINK-V3SET debugger/programmer for STM8 and STM32, оригинал

STM32MP1 platform boot, оригинал

STM32MPU Embedded Software architecture overview - stm32mpu

PM0214 Programming manual - STM32 Cortex®-M4 MCUs and MPUs programming manual, оригинал

STLinux

STM32MP1Dev - STM32MP1 OpenSTLinux Developer Package - STMicroelectronics

Wiki archive for STM32 MPU ecosystem-v1 releases

SDK for OpenSTLinux distribution

- https://www.st.com/en/development-tools/stlinux.html
- http://archive.stlinux.com/

- http://archive.stlinux.com/stlinux/2.4/iso/STLinux-2.4-armv7_spear-20121121.iso

Ссылки

W STM32

Как прошить чип STM32 | Многобукфф

Использование экосистемы STMicroelectronics: подключение датчиков к STM32G4

CountZero: выборка по тегу: STM32

Микроконтроллер и Bootloader. Описание и принцип работы.

Собираем и устанавливаем свою Linux-систему на микроконтроллер STM32MP1 / Хабр

STM32MP1: U-Boot, Buildroot, Arch Linux и немного Debian / Хабр

STM32: аббревиатуры и термины | arm | programming

Библиотека HAL для STM32. Перечень функций и ссылок.

Mastering STM32 book

Паяем standalone-версию программатора ST-Link/v2-1 | Записки программиста

1) Input for STLINK-V3MINI

2) SWO is optional, required only for Serial Wire Viewer (SWV) trace

3) NC means not required for SWD connection

4), 6) Connect to GND on target, may be used by STLINK-V3MINI for detection of connection

5) JTAG Return Test ClocK

https://sysadminmosaic.ru/stm32/stm32

2023-03-27 16:38