

# Курс по STM32

## Лекция #1:

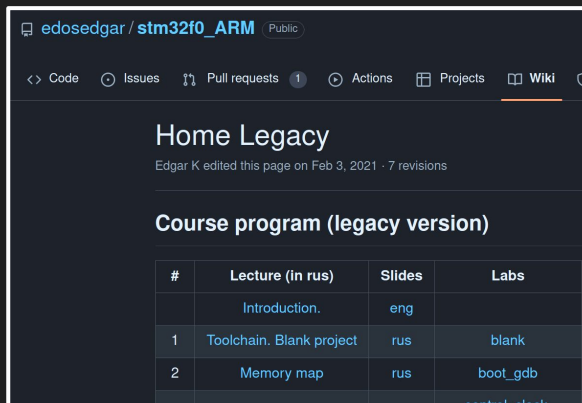
- Что это за курс?
- Программа курса и правила игры.
- Что такое микроконтроллер?
- Зачем и как программировать микроконтроллеры?
- Минимальный набор инструментов.

04.02.2023 / 06.02.2023

# Краткая история студенческого курса по STM32

2018: запуск курса Эдгара Казиахмедова

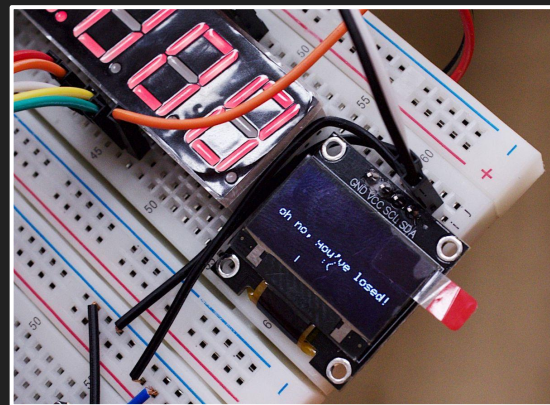
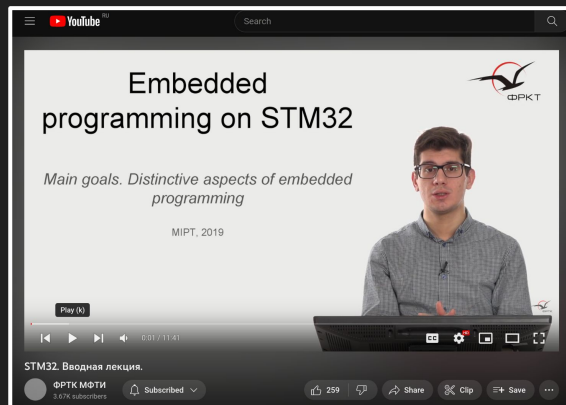
- Грант от Фонда Целевого Капитала.
- Разработка лабораторных работ ([edosedgar/stm32f0\\_ARM](https://github.com/edosedgar/stm32f0_ARM)).
- Закупка оборудования.



# Краткая история студенческого курса по STM32

2019: запуск курса Эдгара Казиахмедова

- Официальный факультатив на кафедре КМТТ, оценка идёт в диплом.
- Запись видеолекций.
- Ответвление курса по Verilog и FPGA ([victor-prutyanyan/drec-fpga-intro](https://victor-prutyanyan.github.io/drec-fpga-intro/)).



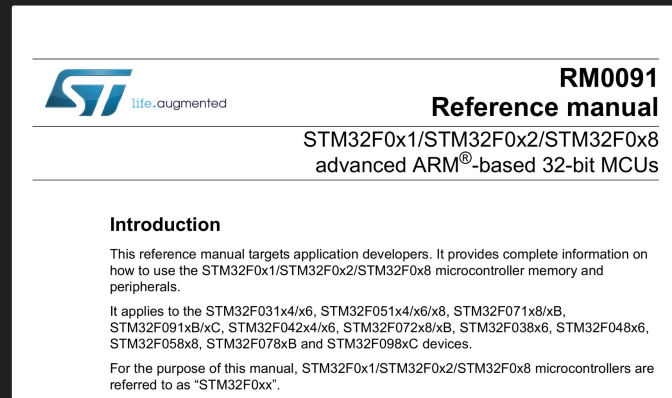
# Краткая история студенческого курса по STM32

2020-2022: поддержание курса

- Помогают Студенческий Совет ФРКТ и кафедра КМТТ.

2023: педагогическая переработка

- Новый репозиторий ([VladikNeVladik/stm32f051\\_rewind](https://github.com/VladikNeVladik/stm32f051_rewind)).
- Программирование “на голом железе”.
- Работа с документацией (1K+ страниц).
- Активная переработка курса.
- Для студентов 2+ курса.
- Преподают студенты 5-го курса.



# Программа курса

## Программа-минимум:

- 01\_ledblink - минимальная программа для МК, toolchain.
- 02\_gpio - порты ввода-вывода, кнопка и семисегментный индикатор.
- Основы ARMv6 Assembly и timing-perfect delay.
- 03\_systick - обработка исключений и системный таймер.
- 04\_uart - Простейший протокол передачи данных - UART.
- Финальный проект.

# Программа курса

## Программа-максимум:

- Отладка в GDB, использование дизассемблера.
- GPIO: ручная генерация ШИМ, обработка энкодера.
- Таймеры общего назначения: функционал, ШИМ, обработка энкодера.
- Протоколы передачи данных: I2C, SPI.

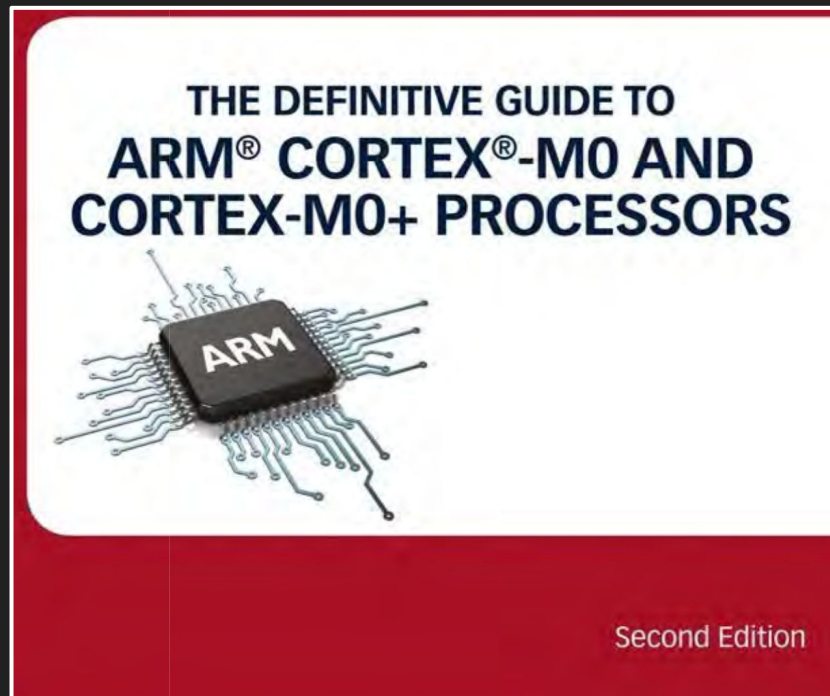
...

```
entry0 (int argc, char **argv, char **envp);  
; arg int argc @ r0  
; arg char **argv @ r1  
; arg char **envp @ r2  
0x00000008    bl      main      ; int main(i  
;-- __halt:  
0x0000000c    b       0xc  
0x0000000e    movs    r0, r0  
board_clocking_init ();  
; var int16_t var_0h @ sp+0x0  
0x00000010    push    {r7, lr}  
0x00000012    add     r7, var_0h  
0x00000014    ldr     r3, [0x000000b0] ; 176
```

```
> 0x120 <main>    push    {r7, lr}  
0x126 <main+2>   add     r7, sp, #0  
0x128 <main+4>   bl      0x10 <board_clocking_init>  
0x12c <main+8>   bl      0x8c <board_gpio_init>  
0x130 <main+12>  ldr     r3, [pc, #32] ; (0x154 <main+48>)  
0x132 <main+14>  ldr     r2, [r3, #0]  
0x134 <main+16>  ldr     r3, [pc, #28] ; (0x154 <main+48>)  
0x136 <main+18>  movs    r1, #128 ; 0x80  
0x138 <main+20>  lsls    r1, r1, #1  
0x13a <main+22>  orrs    r2, r1  
0x13c <main+24>  str     r2, [r3, #0]  
0x13e <main+26>  bl      0xf8 <totally_accurate_quantum_femtosecond_precise>  
0x142 <main+30>  ldr     r3, [pc, #16] ; (0x154 <main+48>)  
0x144 <main+32>  ldr     r2, [r3, #0]
```

# Литература

The Definitive Guide to ARM Cortex-M0 and Cortex-M0+ Processors, Joseph Yiu.



# Правила игры

Система оценивания:

- 4 лабы (из репо) + 1 задание (без репо) + проект (техно-творчество)

Оборудование:

- Выдача - после сдачи первой лабы (в эмуляторе QEMU)
- Необходимо вернуть для получения оценки

#	Участник	blinkled	GPIO	delay	SysTick	UART	Проект	Итог	Зачёт (итог + мнение)	Примечания
		10	15	15	20	20	40	120		
1	Студент #NEUD	3	3	3	3	3	3	15	неуд(2)	Проект - отстой
2	Студент #HOR	5	5	5	5	5	5	45	хор(5)	Проект - норм
3	Студент #DESCENT	10	10	3	5	5	10	82	отл(8)	Проект - интересный
4	Студент #GODLIKE	10	10	10	10	10	10	120	отл(10)	Проект - полезный

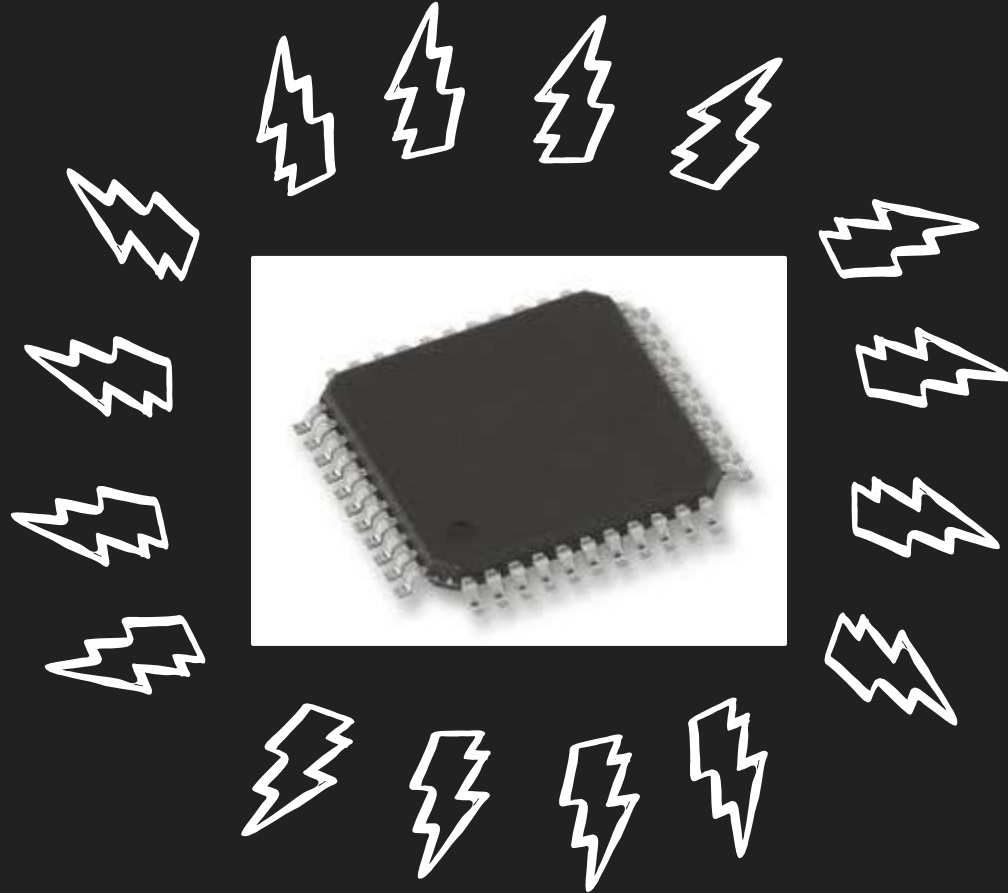


QR-код на добавление в чат



**Организационные  
вопросы?**

# Что такое микроконтроллер?



# Микроконтроллеры повсюду

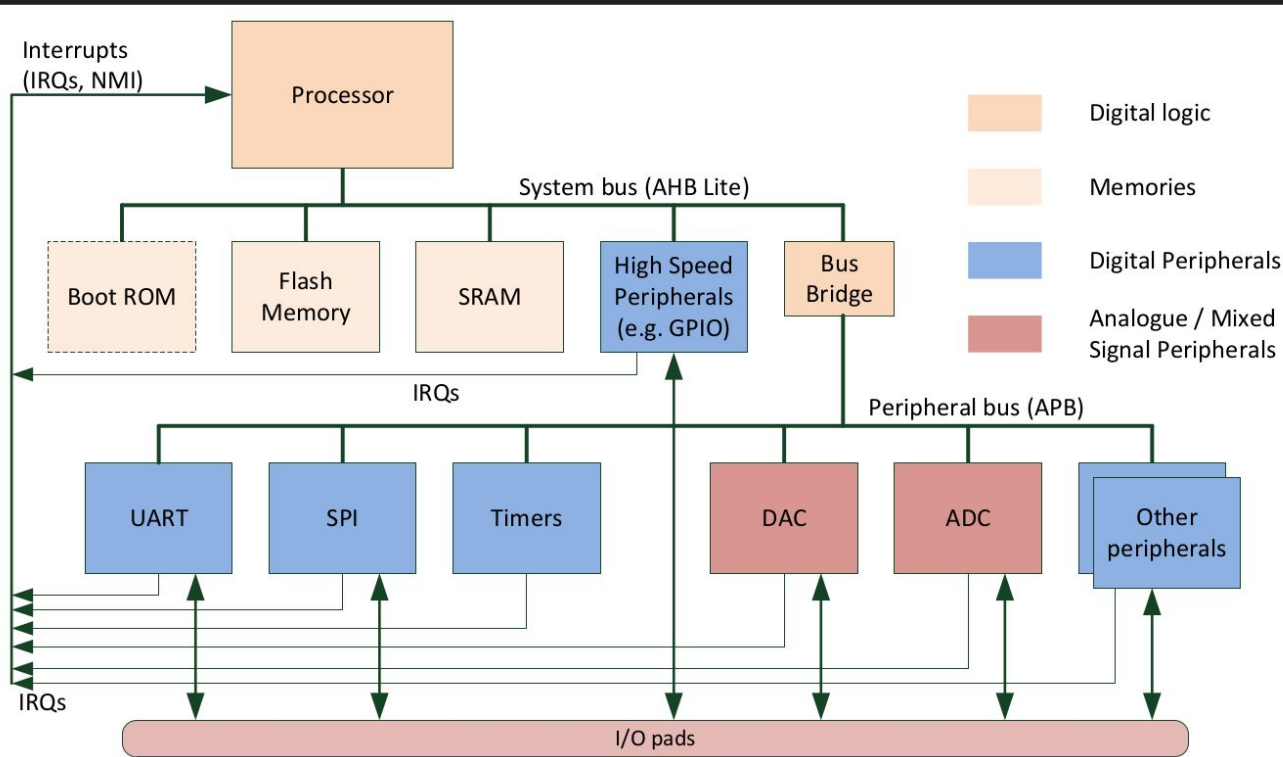
- Бытовая техника: микроволновки, посудомоечные машины, чайники.
- Вычислительная техника: ПК, мобильные устройства, сервера –  
– умные сенсоры и Sensor Hub-ы, системы управления питанием ...
- Транспортные системы: системы коммуникации, навигации, управления двигателями, системы управления ТС, ...
- Медицинская техника, системы автоматического управления производством, ...

Задание:

- Сосчитайте микроконтроллеры в этой комнате.



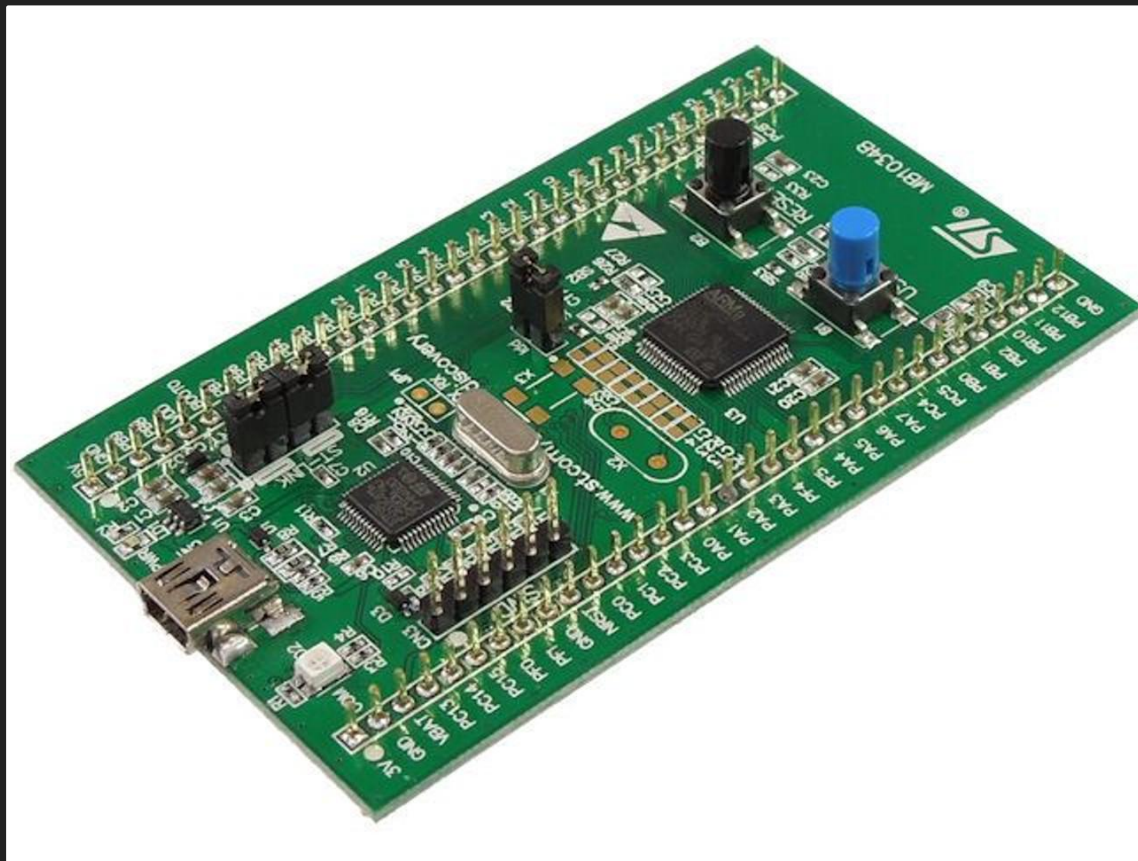
# Что внутри у микроконтроллера?



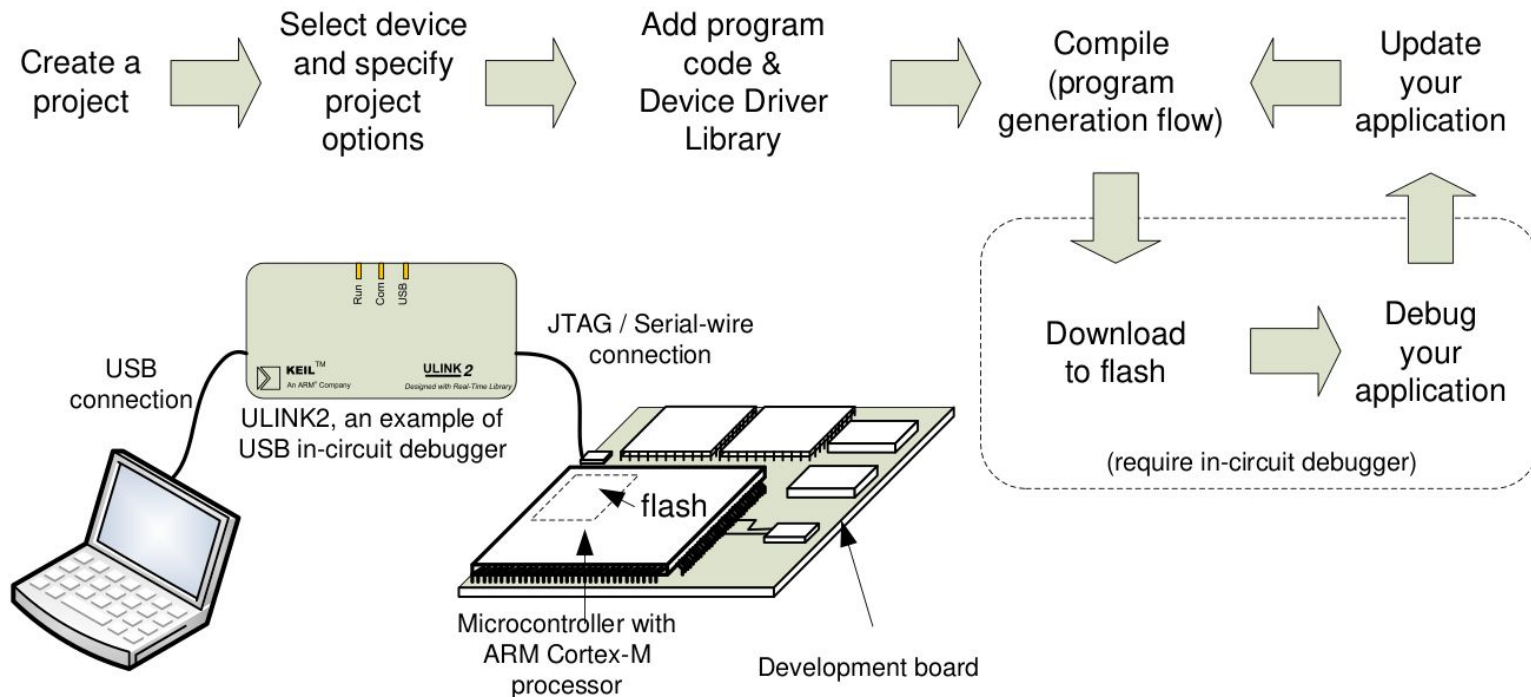
**Figure 2.4**

A simple system with the Cortex<sup>®</sup>-M0 Processor.

# Что такое отладочная плата?



# Как программировать микроконтроллеры?



**Figure 3.14**  
An example of development flow.

# Необходимое программное обеспечение

- 0) Операционная система: желательно – Linux, допустимо – Mac OS
- 1) Кросс-компилятор – GNU Arm Embedded Toolchain  
(`arm-none-eabi-gcc`, ...)
- 2) Эмулятор для первой лабы – QEMU (`qemu-system-gnueabi`)
- 3) Утилита для связи с программатором – ST-Link (`st-util`)

## Домашнее Задание:

- Пройти [руководство по установке ПО](#).
- Установить необходимое ПО.



**Спасибо за внимание!**