## Úvod do STM32F407 dosky

- STM32F407VGT6 microcontroller featuring 32-bit ARM<sup>®</sup> Cortex<sup>®</sup> -M4 with FPU core, 1-Mbyte Flash memory, 192-Kbyte RAM in an LQFP100 package
- On-board ST-LINK/V2 on STM32F4DISCOVERY (old reference) or ST-LINK/V2-A on STM32F407G-DISC1 (new order code)
- USB ST-LINK with re-enumeration capability and three different interfaces:
  - Debug port
  - Virtual Com port (with new order code only)
  - Mass storage (with new order code only)
- Board power supply: through USB bus or from an external 5 V supply voltage
- External application power supply: 3 V and 5 V
- LIS302DL or LIS3DSH ST MEMS 3-axis accelerometer
- MP45DT02 ST-MEMS audio sensor omni-directional digital microphone
  - CS43L22 audio DAC with integrated class D speaker driver
- Eight 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
- Two push-buttons (user and reset)
- USB OTG FS with micro-AB connector
- Extension header for all LQFP100 I/Os for quick connection to prototyping board and easy probing
- Comprehensive free software including a variety of examples, part of STM32CubeF4 package or STSW-STM32068 to use legacy standard libraries





## Visual Studio a rozšírenie VisualGDB

1. Spustíme Visual Studio, file, new, project



## 2. VisualGDB, embedded project wizard

New Project									?	×
▷ Recent		.NET Fr	amework 4.5.2	- Sort by:	Default	• #	E	Search Ins	talled Tei	ρ-
▲ Installed			Android Proje	ct Wizard	VisualGDB	Type: Visua	alGDB			
✓ Templates ▷ Visual C# ▷ Visual Period		, T	Custom Proje	ct Wizard	VisualGDB	An Embedd Studio.	An Embedded project I Studio.			al
Visual F#			Embedded Pro	oject Wizard	VisualGDB					
SQL Server JavaScript VisualGDB			Linux Project \ MinGW/Cygw	Wizard vin Project Wiza	VisualGDB ard VisualGDB					
<ul> <li>Fython</li> <li>TypeScript</li> <li>Game</li> <li>Build Accelerat</li> <li>Other Project T</li> </ul>	or vnes v									
▷ Online		1	Click here to go	online and fin	d templates.					
<u>N</u> ame:	STM32F4-LEDBli	nk								
Location: c:\projects\temp		0			<u>B</u> rowse					
Solution na <u>m</u> e:	STM32F4-LEDBli	ink				Create <u>d</u> irect	tory foi ce Con	r solution trol		
								ОК	Cance	2l

3. Vytvárame nový firmvérový súbor



4. Ako robíme firmvér pre STM32F4 dosku, ktorá obsahuje STM32F407VG mikroregulátor, označíme ARM a nainštalujeme ovládače, ak ešte nie sú

🖉 New Embedded Project						_		×	
R	New	Embedo	ded	Project	t				
Project Type	Please select a toolchain that will be use	ed to build your pro	ject:						
Device Selection	AVR in C:\SysGCC\avr							<u>^</u>	
Device Selection	GCC 5.3.0, GDB 7.11, Revision ESP32 in C:\SysGCC\esp32	7							
Sample Selection	GCC 5.2.0, GDB 7.10, Revision ESP8266 in C:\SvsGCC\esp826	<i>3; ESP-IDF 2.1</i> 56							
Debug Method	GCC 5.2.0, GDB 7.11, Revision	11; IoT SDK 2.0.0	); RTOS .	SDK 1.5.0; 01	TA; NodeMCU				
	MSP430 in C:\SysGCC\msp430-elf GCC 5.9.0, GDB 7.7, Revision 1								
	GCC 6.2.0, GDB 7.12,	)							
	GCC 4.6.3, GDB 7.4,	\ \							
	GCC 4.7.3, GDB 7.7,	)							
	GCC 4.8.0, GDB 7.5.1, (download toolchain for m68k-elf	)							
	GCC 4.8.0, GDB 7.5.1, (download toolchain for msp430)	2							
	GCC 4.6.3, GDB 7.4, (download toolchain for powerpc	-eabi)						- 1	
	♥ GCC 4.9.0, GDB 7.7,	- 						Ý	
	Select a third-party toolchain by lo	ocating gdb.exe							
	Download more toolchains		64K	21	SCK.	dowploadable			
	AT91SAM7X512		128K	5	12K	downloadable			
	AT91SAM7XC128		32K	1	28K	downloadable		- 🗸 📗	
	AT313AM/AC230		0410	2.		downloadable			
	2 Please select a device								
				< <u>P</u> revious	<u>N</u> ext >	<u>F</u> inish	<u>C</u> ar	ncel	
	New	Embedo	ded	Project	t			^	
Project Type	Please select a toolchain that will be use	ed to build your pro	ject:						
	ARM in C:\SysGCC\arm-eabi							•	
Device Selection	Select a device from list O Spece	ify flags manually	<ul> <li>Use</li> </ul>	e ARM mbed		🖶 Download m	iore devi	ces	
Sample Selection	Devices by name Devices by type								
	Supported devices:					Filter: f407			
Debug Method	Device		RAM size	e R	OM size	State		^	
	STM32F407IE STM32F407IG		192K 192K	5	12K 124K	installed		- 11	
	STM32F407VE		192K	5	12K	installed			
	STM32F407VG		192K	1	024K	installed		_	
	STM32F407ZE		192K	5	12K	installed		~	
								^	
	BSP Version	4.3						~	
	Execute from	FLASH						$\sim$	
	Floating point support	Software						$\sim$	
	C Library Type	Newlib-nano						~	
	Reduce the size of C++ binaries								
	Provide default stubs for system calls								
	Shared files location: C:\Lleare\virtual		/isualGDI	B\EmbeddedP	SPs\am_aabi\	COM SVSDroge am	Cha	nge	
	C. Users WITUA	v ppbala (Local (V	naudicipi	o venibedded B	or a varimeably	com.sysprogs.am.	una	ngo	
				< Province	Not 1	Dinich	0	neel	
				< nevious	INCXL 2	1111511	Udi	IUCI	



5. STM32F4 doska obsahuje 4 LED pripojené na piny PD12 až PD15, pozri schéma

LEDs

V tomto projekte rozblikáme zelenú LED, pripojenú na PD12. Nastavíme GPIO a 12 ako port a číslo.

New Embedded Project			_		×
R		New Embedded Project			
Project Type	Show basic samp	les O Show STM32 CubeMX Samples			
Device Selection	Select a sample proje	ect to generate: LEDBlink (HAL)			$\sim$
Device Selection	This is a very basic s	ample project - a classical blinking LED example.			
Sample Selection	LED Port Group	GPIOD			$\sim$
Debug Method	LED Port Number	12			
	Delay (msec)	500			÷
					<u> </u>
		< <u>P</u> revious <u>N</u> ext >		Can	cel

6. STM32F4 doska obsahuje vstavený ST-Link programátor. Ak pripojíme dosku, VisualGDB ju automaticky rozpozná. Vyberieme programátor a spustíme test.



VisualGDB sa pokúsi spustiť OpenOCD a pripojí sa, kde potvrdí, že debugovanie bude možné.

2 New Embedd	ded Project						$\times$
R		New	Embedded Pro	oject			
Project Type	Debug using:	ST-Link v2.1				~	Test
Device Selection	😢 Testing Settings				- 0	×	~
Sample Selection	VisualGDB is testing your del duapter_msrst_delay. At none separate Info : Unable to match Info : Unable to match Info : clock speed 1800	requested speed 2000 kHz, requested speed 2000 kHz, http://www.speed.accounter.org	using 1800 kHz using 1800 kHz			1	(Hz (Hz
🔇 VisualG	DB					×	
	Connection test succeeded. If y	ou encounter problems while de	bugging, please double-ch	eck the gdb stub output fo	or warnings and hints.		
					ОК		
	adapter speed: 4000 kH VisualGDB_OpenOCD_Ready Info : accepting 'gdb' Info : device id = 0x10 Info : flash size = 100 Info : dropped 'gdb' co	connection on tcp/1858 0076413 14kbytes nnection (error -400)				>	/
	Settings tested successfully.				Terminate	Close	
				< Previous Next	> <u>F</u> inish	Cano	el

7. Stlačíme Finish. Vytvorí sa projekt. Ctrl-Shift-B a projekt sa "buildne"



8. Stlačíme F5 a spustíme debuggovanie. Sledujme ako LED bliká. Nastavíme breakpoint na GPIO\_WriteBit() riadok. Breakpoint sa spustí automaticky a ledka sa vypne.



9. Ctrl-Shift-G otvoríme okno registrou.

STM32F4-LEDBlink (Debugging) - Microsoft Visual Studie           File         Edit         View         Project         Build         Debug         Team           ●         •         ●         •         ●         •         ●         •         ●         •         ●         •         ●         •         ●         •         ●         ●         •         ● <th>5 [ools Android Architecture Test Ar [VisualGDB -] ▶ Continue -   戸 = e Events - Thread: [1] 200000 r3 = 0x00000010 r4 = 0x2000002 x000000000 r12 = 0x00000000 sp = 0x20</th> <th>Image: Provide the second s</th> <th>trl+Q P = ₽ × Code Map 8 = ÷ + + ffe0 r8 = 0x00000000 = 0x00000000</th>	5 [ools Android Architecture Test Ar [VisualGDB -] ▶ Continue -   戸 = e Events - Thread: [1] 200000 r3 = 0x00000010 r4 = 0x2000002 x000000000 r12 = 0x00000000 sp = 0x20	Image: Provide the second s	trl+Q P = ₽ × Code Map 8 = ÷ + + ffe0 r8 = 0x00000000 = 0x00000000
▲ LEDBlink.cpp + × () (global scope)	v ∣Q main	• II	Jorer Team Explo
GPIOD_CLK_ENABLE(); GPIO_InitTypeDef_GPIO_InitStruct GPIO_InitStructure.Pin = GPIO_PI GPIO_InitStructure.Mode = GPIO_M GPIO_InitStructure.Speed = GPIO_ GPIO_InitStructure.Pull = GPIO_I HAI_GPIO_INITSTRUCTURE.PULL	ure; N_12; ODE_OUTPUT_PP; SPEED_HIGH; OPULL; toucture}:		÷ ir
100 % - 4			• • •
Real-time watch	<b>→</b> ‡	× openocd	<b>~</b> ₽ ×
Real-time watch is not en You can enable it via VisualGDB Project Pr	nabled in this session operties -> Dynamic Analysis page.	<pre>sing 1800 kHz Info : Unable to match requested sing 1800 kHz adapter speed: 1800 kHz target halted due to debug-reque Thread xPSR: 0x01000000 pc: 0x0800035c Warn : keep_alive() was not invo timelimit. GDB alive packet not around: increase "set remotetime </pre>	speed 2000 kHz, u st, current mode: msp: 0x20020000 ked in the 1000ms sent! (1296). Work out" in GDB
Ready	Ln 18 Col 41 Ch 38	INS	↑ Publish 🔺

**10. Hodnota PC registra** 0x80xxxxx indikuje, že program je spustený z flash pamäte. Stlačme F10 sa posunieme o jeden riadok v kóde. Teraz by LED mala svietiť.



11. Debuggovanie ukončíme pomocou Shift-F5. Pravý click na projektový uzoľ v Solution Explorery a stlač "VisualGDB Project Properties".

STM32F4-LEDBlink - Microsoft Visual Studio         File       Edit       View       Project       Build       Debug       Team       Tools       Android       Architecture       Test         Image: Im	An ugger	같   alyze Window Help ↓   ♬ <sub>♥</sub> 등 남 대   1 1 12 1   ■ 위 위	Quick Launch (Ct 제 책 <sub>무</sub>	trl+C	аран (р. 1997) аран
é LEDBlink.cpp → ×			Solution Explorer		<b>-</b> ∓ X
🛔 () (global scope) 🗸 🛛 main 🗸		VisualGDR Draiest Drapaties		_	5 a a v 2 -
<pre>GPIOD_CLK_ENABLE(); GPIOD_CLK_ENABLE(); GPIO_InitTypeDef GPIO_InitStructure; GPIO_InitStructure.Pin = GPIO_PIN_12; GPIO_InitStructure.Mode = GPIO_MODE_OUTPUT_PP; GPIO_InitStructure.Speed = GPIO_SPEED_HIGH; GPIO_InitStructure.Pull = GPIO_MOPULL; HAL_GPIO_InitGPIOD, &amp; &amp;</pre>	- <mark> </mark>	VisualGDB Project Properties Program and Start Without Debugging Build Rebuild Clean View Analyze Project Only Scope to This New Solution Explorer View Show on Code Map Profile Guided Optimization Build Dependencies Add		>       >       >       >	rer (Ctrl+;) P = P F4-LEDBlink' (1 project) DBlink s Pependencies es files es es files es especific files ink.cpp n_stm32f4xx.c B settings
	₽.	Class Wizard	Ctrl+Shift+X		
100 % - 4	Ĥ	Manage NuGet Packages			xpl Class View Property M
	ø	Set as StartUp Project			<b>-</b> ₽ ×
Share a start from 1600 Decemen Ortext		Debug		•	Project Properties -
Show output from: VisualGDB Program Output		Source Control		۲	
into : dropped gub connection (error 400)	ж	Cut	Ctrl+X		
		Paste	Ctrl+V		STM32F4-LEDBlink
	×	Remove	Del		ties
	X	Rename			c:\projects\temp\STM32
		Unload Project			
		Rescan Solution			name.
Tree Bro Clang I Packag Error List Task List Breakpo Find Sy Preproc SSH Co	9	Open Folder in File Explorer			
Ready Ln 21 Col 1 Ch	×	Properties	Alt+Enter		🋧 Publish 🔺

## 12. Clicknime "Change settings" and nastavme SRAM namiesto of FLASH

VisualGDB Project Properties - STM32F4	-LEDBlink			_	
Configuration: Debug				∼ 🛟 Add	X Delete
Search	Embedded Project This project is built using an embedded o Note that changing options on this page	cross-compiler toolchain. V will affect all configuration	fisualGDB manages the basic is of the project.	build settings automatica	ally.
Embedded Frameworks	Embedded device				
Unit Tests	Change settings	5	Convert to s	stand-alone project	
MSBuild settings	Devices by name Devices by type Supported devices:			Filter:	
Debug settings	Device STM32F405ZG STM32F407JF	RAM si 192K 192K	ize ROM size 1024K 512K	State installed	^
Advanced Debug	STM32F407/G STM32F407/E STM32F407/E STM32F407/G	192K 192K 192K 192K	1024K 512K 1024K	installed installed installed	-11
Dynamic Analysis	STM32F407ZE	192K	512K	installed	~
Custom build steps	BSP Version	4.3			~
Additional Memories	Execute from Floating point support	SRAM FLASH SRAM			~
Custom debug steps	C Library Type	Newlib-nano			~
Custom shortcuts	Reduce the size of C++ binaries Provide default stubs for system calls	$\square$			
Raw terminal	Shared files location: C:\Users\virtual	\AppData\Local\VisualGi	DB\EmbeddedB! Change	e 🥰 Regenerate	MCU files
View VisualGDB build variables for this con	figuration			OK Cancel	Apply

13. Nakoľko predošlí projekt bol nahraný vo Flash pamäti, budeme musieť znovu rebuildnúť projekt. Zopakujeme predošlé kroky, a pozrieme hodnotu v PC registry, ak je v rozsahu 0x20xxxxxx, tak vieme, že program bol nahraný do SRAM.



Chod programu z SRAM nám neskracuje FLASH prepisovacie cykly, ako keď nahrávame pamäť alebo používame breakpointy. Avšak používanie SRAM má svoje nevýhody. Väčšie programy sa nemusia zmestiť do SRAM, takže pre veľké programy s veľa premenami to nemusí byť najvhodnejšie riešenie.

Nakoniec si pozrieme View->Embedded Memory Explorer, aby sme videli, ktoré konkrétne funkcie sú zodpovedné za používanie pamätí.