Конструктор робот-манипулятор Hiwonder LOBOT LeArm Single Robot

Инструкция по эксплуатации

Оглавление

3
4
6
7
9
10
10
11

1. Сборка

Для клиентов, которые приобрели робот в виде конструктора, первым делом предстоит его сборка. Подробное видео с процессом сборки вы можете найти у нас на сайте.

Перед сборкой настоятельно рекомендуем убедиться, что в комплекте присутствуют все детали, показанные на картинках ниже. Обратите внимание, что в приведенном списке показаны детали второго поколения.





Сборка делится на 3 этапа: сборка каркаса, крепление сервоприводов, подключение электрических контактов. При сборке каркаса разделите детали на передние, задние, правые, левые, верхние и нижние, внимательно следите за процессом сборки на видео.

При монтаже сервоприводов следите за положением главного и вспомогательного валов, не поворачивайте сервоприводы в процессе сборки. Исходное положение всех приблизительно (90 сервоприводов одинаковое градусов), на видео задействованы муляжи отцентрированных сервоприводов, поэтому, если вы повернете ось сервопривода, будет затруднительно повторить процесс сборки с видео. Сперва отцентрируйте сервопривод с помощью платы управления и лишь затем продолжайте сборку. Подробный алгоритм центрирования сервопривода представлен в справочных материалах. В конце, при подключении электрических контактов и платы управления, строго сверяйтесь указаниями с по подключению. Следите за соответствием маркировки проводов и портов на плате, неправильное подключение грозит порчей оборудования.

2. Калибровка

По окончании сборки манипулятора подключите его к ПК (хосту) через дата-кабель, включите ПК, выберите СОМ-порт, нажмите кнопку «Reset» сервопривода. В нормальном режиме робот-манипулятор должен вести себя так, как показано на рисунке 2, т.е. находиться в вертикальном положении. Если положение отличается, необходимо снять сервопривод, отцентрировать его и повторить все заново. При наличии небольшого отклонения рекомендуется произвести калибровку сверяясь со справочными материалами.

开源机械转	THE R. L. LANS	Same and	-	- Comili	States of	and the second s	
PWM能机 意始的L	論号 하)	Ēl(ms) ID:1	ID:2	ID:3	ID:4	ID:5	ID:6
1500 ID:2							
1500							
					1.12		
1500	动作时间 1000	添加动作	F	制除动作	更新动	# ¹ 1	插入动作
1500							
	Сом:	■ 循环		打开文件		动作组	•
1500	发神车: 9600	•	KZIF	L			下载
				保存文件			全部海除
1500			BC/05			动性组动	新江 停止

Рисунок 1



Рисунок 2

3. Управление

Робот-манипулятор LeArm предусматривает 4 способа управления: от ПК, от смартфона, от джойстика и пассивное управление. Пассивное управление и управление через приложение на смартфоне производится с помощью наборов действий, которые необходимо предварительно загрузить на плату управления. Подключившись к ПК, вы можете писать, редактировать или скачивать наборы действий. Управление через джойстик производится так же через наборы действий или напрямую, каждым сервоприводом по отдельности.

4. Главный компьютер (хост)



На скриншоте показано только то, на что преимущественно следует обратить внимание. Во-первых, в левом верхнем углу находятся две вкладки: ШИМ-сервоприводы и сервоприводы общей шины. На панели расширения LeArm есть порты для подключения сервоприводов общей шины. При покупке шины для сервоприводов нашего производства, вы можете переключиться на вторую выкладку и управлять роботом по общей шине.

Теперь взглянем на 6 голубых боксов, расположенных в левом окне, в каждом из них прописаны ID, обозначающие положение конкретного сервопривода относительно их подключении к плате расширения (на плате расширения имеется маркировка для подключения сервоприводов).

Итак, подключаемся к главному компьютеру. Если к нему в это время подключены другие устройства, они также будут отображаться при выборе СОМ-порта. Рекомендуется устанавливать нужный порт через диспетчер устройств.

СОМ-порт с именем CH340 и есть порт нашего робота-манипулятора, если портов с таким названием несколько, просто выберите один из них. После установки порта передвиньте ползунки сервоприводов. Если рука будет двигаться, это означает, что порт выбран верно, если нет, выберите другой порт и попробуйте снова.

M 能机 总统能制		· -	时间(ms)	ID:1	ID:2	1D:3	1D:4	1D:S	ID:6	
ALCOST AND		1	1000	1500	1500	1500	1500	1580	1500	
IDI										
1500										
10:2										
1500										
10.3										
				F	-					
1500	cJ/fee,	till pit	000	(Starting		HIRA STATE	10 AGE:	*	振入动作	
1000				Standard H	interest of the second s	029-2-4/1 +-	2000		the sector	
104				Sandonina -		6297-921 F			and surface	5
				1000000			228142		and the second s	2
1500								-	0	5
	сом:					打开文体		ಪ್ರಗಳಿತ:	0.	5
	CDM:		•	- 1857 - 1857	547	打开 来#		ಪೆಗಳುಕ್ರ	0 •	5
1500	CDM: 这将查	9600	•	eas.	517	D开文体		र हो%खेः	0 •	5
	сом. Зна	9600	•		\$17	13开交34		ಸ/ಕುಷ್ಣ:	0 	

Каждый набор действий вмещает до 255 действий. Если вы хотите прописать больше действий, воспользуйтесь методом по их добавлению, который представлен в справочных материалах.

Обратите внимание, в случае, если вы не отключитесь от главного компьютера, а просто выключите манипулятор или отсоедините USB-кабель, подключение к СОМ-порту на главном компьютере будет отображаться как все еще активное. При последующем включении робота или соединении USB, отключитесь от СОМ-порта и подключитесь снова.

·····································	3
文件(F) 操作(A) 查看(V) 帮助(H)	
● admin-PC ● DVD/CD-ROM 驱动器 ● IDE ATA/ATAPI 控制器 ● WSD 打印提供程序 ● 处理器 ● 建築SERIAL CH340 (COM12) ● USB-SERIAL CH340 (COM12) ● 世現器 ● 世現書 ● 世現書 ● 世現書 ● 世現書 ● 世現書 ● 近日 ● 世現書 ● 通信演員 ● 通信演員 ● 通信 ●	

5. Джойстик

Подробнее об использовании джойстика вы можете узнать из справочных видео. Управление через джойстик делится на 2 типа: через наборы действий и прямое управление одиночным сервоприводом.

При включении LeArm, управление через набор действий выбрано по умолчанию. При нажатии различных клавиш и ввода переменных плата управления реализует загруженные в нее команды и отправляет на исполнение манипулятору. Переключиться на управление одиночным сервоприводом можно посредством комбинации клавиш SELECT+START, при успешном переключении прозвучит сигнал. Если сигнал не прозвучал, повторите попытку. В режиме управления сервоприводом можно осуществлять управление каждым сервоприводом с помощью кнопок или дистанционно. При необходимости переключиться на режим воспроизведения набора действий, зажмите кнопки SELECT+START. При успешном переключении прозвучит сигнал.

IC	Режим выполнения	набора	Режим управления
Клавиши	действий	-	одиночным сервоприводом
STADT	Запустить 1 раз	набор	Установить все
START	действий №0		сервоприводы на 1500
	Запустить 1 раз	набор	Управление вращением
ыперед (левая клавиша)	действий №1	2.5	сервопривода №5
Царад	Запустить 1 раз	набор	Управление вращением
пазад	действий №2		сервопривода №5
Цанара	Запустить 1 раз	набор	Управление вращением
палево	действий №3		сервопривода №6
Направо	Запустить 1 раз	набор	Управление вращением
Паправо	действий №4		сервопривода №6
	Запустить 1 раз	набор	Управление вращением
	действий №5		сервопривода №4
_	Управление вран	цением	Управление вращением
	сервопривода №6		сервопривода №4
- <u>6</u> .C.	Запустить 1 раз	набор	Управление вращением
	действий №7		сервопривода №3
	Запустить 1 раз	набор	Управление вращением
	действий №8		сервопривода №3
	Запустить 1 раз	набор	Управление вращением
	действий №9		сервопривода №2
P1	Запустить 1 раз	набор	Управление вращением
K1	действий №10		сервопривода №2
1.2	Запустить 1 раз	набор	Управление вращением
	действий №11		сервопривода №1
P2	Запустить 1 раз	набор	Управление вращением
K2	действий №12		сервопривода №1
SELECT+START	Переключение в	режим	Переключение в режим
SLLCITSIANI	одиночного сервоприв	вода	выполнения набора команд

Описание кнопок джойстика представлено в таблице ниже.

6. Управление через смартфон

Для управления через смартфон требуется установить специальное приложение. После установки запустите приложение, включить на смартфоне Bluetooth и подключитесь к роботу.

После открытия приложения нажмите на красный значок Bluetooth в верхнем левом углу, в всплывающем списке выберите устройство LOBOT, нажмите "подключиться". При успешном подключении красная иконка Bluetooth станет синей. При нажатии кнопок 1-6 осуществляется вызов соответствующих наборов действий (необходимо предварительно загрузить наборы действий на плату управления).



7. Пассивное управление

В нижнем углу контроллера LeArm есть кнопка пассивного управления 1. Загрузите на плату управления набор, включающий 100 действий, и нажмите эту кнопку. Манипулятор выполнит набор из 100 действий, а при зажатии кнопки на 3 робот будет выполнять цикл действий по кругу. Для остановки цикла следует отключить манипулятор.

8. Создание среды разработки и программирование

8.1 MDK-ARM установка и программирование

В независимости от версии, установка и программирование проходит одинаково. 1. Запустите mdk.exe и подготовьтесь к установке MDK-ARM

Welcome to Keil µ Vision Release 12/2011	
This SETUP program installs:	
MDK-ARM V4.23	
This SETUP program may be used to update a However, you should make a backup copy be	a previous product installation. efore proceeding.
It is recommended that you exit all Windows pr	rograms before continuing with SETUP.
Follow the instructions to complete the product	t installation.

2. Поставьте галочку на соглашение с правилами установки и использования.

License Agreement Please read the following license agreement carefully.
To continue with SETUP, you must accept the terms of the License Agreement. To accept the agreement, click the check box below.
END USER LICENCE AGREEMENT FOR MDK-ARM THIS END USER LICENCE AGREEMENT ("LICENCE") IS A LEGAL AGREEMENT BETWEEN YOU (EITHER A SINGLE INDIVIDUAL, OR SINGLE LEGAL ENTITY) AND ARM LIMITED ("ARM") FOR THE USE OF THE SOFTWARE ACCOMPANYING THIS LICENCE. ARM IS ONLY WILLING TO LICENSE THE SOFTWARE TO YOU ON CONDITION THAT YOU ACCEPT ALL OF THE TERMS IN THIS LICENCE. BY CLICKING "I AGREE" OR BY INSTALLING OR OTHERWISE USING OR COPYING
I agree to all the terms of the preceding License Agreement
- Keil µVision4 Setup

3. Выберете папку для установки

older Selection Select the folder where SETUP will install files.	Tools by A	RM
SETUP will install µVision4 in the following folder. To install to this folder, press 'Next'. To install to a diffe folder.	rent folder, press 'Browse' and select another	
- Destination Folder		
E:\Embeded\Keil	Browse	\supset
E:\Embeded\Keil Update Installation: Create backup tool folder Backup old files to E:\Embeded\Keil\Backup.00	Browse	\geq
E:\Embeded\Keil Update Installation: Create backup tool folder Backup old files to E:\Embeded\Keil\Backup.00	Browse	

4. Введите данные пользователя (Не обязательно писать подлинные данные)

Customer Information	KEI
Please enter your information.	Tools by AR
Please enter your name, the name of the cor	mpany for whom you work and your E-mail address.
First Name:	
Last Name:	
Company Name:	
Company Name:	
Company Name: E-mail: Keil µVision4 Setup	

5. Установка

Setup Status	
µVision Setup is performing the requeste	ed operations.
Install Files	
Installing LCD_X_SPI_LPC1700.c.	
— Keil µVision4 Setup ————	<< Back Next >> Can
6. Завершение установки	eservi i
File installation and lated	
rile installation completed	
µVision Setup has installed all files succes	ssfully.
µVision Setup has installed all files succes	ssfully. 是否保持当前配置,由于我之前 装讨Kei151
µVision Setup has installed all files succes ▼ Retain current µVision configuration.	ssfully. 是否保持当前配置,由于我之前 装过Kei151
µVision Setup has installed all files succes I ✓ Retain current µVision configuration. ✓ Add example projects to the recently u	ssfully. 是否保持当前配置,由于我之前 装过Kei151 sed project list
µVision Setup has installed all files succes ✓ Retain current µVision configuration. ✓ Add example projects to the recently u ✓ Add example projects to the recently u ✓ Simulated Hardware ✓ 添計	ssfully. 是否保持当前配置,由于我之前 装过Kei151 sed project list

7. Установка драйвера Ulink

		1211-1	
Keil µ Vision4 Setup comp MDK-ARM V4.23	uVision安装完成	To	
µVision Setup has performed a	all requested operations successfully	l.	
🔽 Launch Driver Installation:	"ULINK Pro Driver V1.0"	安装ULINK驱	动
✓ Show Release Notes.			
Keil µVision4 Setup		Rack Finish	Cancel
		Paristi	Cancer
E:\Embeded\Keil\ARM\ULINK	\InstallULINK.exe		
nstalling ULINK Drivers			

Программирование ARM-MDK V4.23

Запустите uVision4, нажмите File>>License Management, скопируйте CID, после чего откройте Keil_Lic, вставьте CID и выберете ARM. Нажмите generate для создания Lic.

Customer In	nformation			Computer 10	
Name:	xtu xttu		1. 复制	CID: CGZJI	E-V29B0
Company:	xtu			Get LI	C via Internet
Email:	xtu				
		1. 15.0 1 4 10			
MDK-ARM	Standard	GMZ8T-RVMLA-8FQIE-5XI	HIM-VLY2E-2RCHY	Expires: Nov 202	0
PK51 Prof.	Developers Kit	CZVFW-QWP9N-UQ80Q-N	E534-4ZUPP-LHWZ4	Expires: Jul 2020	
New Licen	se ID Code (LI	C). WHY74-F5HJY-YV31	Y-19423-07QG4-8CR5	3 Add LIC	Uninstall
LIC Adde	d Sucessfully *	6. 粘贴LIC		7. 添	hulic _
			Close		Help
		(
E	Keil Ge	neric Keygen - E	EDGE		26
			1 - C - C	£	5
	TDOR	Ke	il Embedd		µør ^k
	TO OR	Ke 2. 粘贴CID Details	i.I Embedd		µ¢r ^k
	TD 000	Ke Details	i.I Embedd		HOTK
	TO OR	Ke CGZIE-V2980	i.I Embedd		HOF'R
	TO OR	ке CGZIE-V2980	i.I Embedd		port.
	To day	Ke CGZIE-V2980 Tof. Developers Kit/Re	i.1 Embedd		HOT'R Dench
	Keygen	ке CGZIE-V2980	1.1 Embedd		HOL.
	Keygen	Ke CGZIE-V2980 rof. Developers Kit/Re	i.1 Embedd		Port
	To day	Ke CGZIE-V2980 Tof. Developers Kit/Re	1.1 Embedd		POTR
	Keygen	Ke CGZIE-V2980 rof. Developers Kit/Re	i.I Embedd	ed Horkb	
	Keygen	ке CGZIE-V2980 rof. Developers Kit/Re CHY74-F5HJY-YV3	i.1 Embedd	ed Horkb	

Создание нового проекта

1. Для закрытия старого проекта откройте главную страницу MDK. На главной странице вы увидите список уже созданных проектов. Нажмите на название проекта и в меню выберете: Project->Close Project. Таким способом вы сможете закрыть все ненужные вам проекты, очистив главную страницу MDK. Далее мы рассмотрим, как создать новый проект.

2. Перед созданием проекта рекомендуем создать на вашем компьютере папку, для хранения всех необходимых для нового проекта данных. Для примера мы создали папку под названием STM32-Projects.

3. Откройте меню Keil: Project->NewUvisionProject, после выберете специально созданную папку STM32-Projects, и создайте в ней папку Tempalte. Далее в меню выберете папку Tempalte. Новый проект мы будем сохранять в данной папке. Новый проект назовём Tempalte. Нажмите «Сохранить».

💟 Create New Proje	ect							×
	> 此电脑 > 本地磁曲	t (D:) → Template			~ ð	搜索"Template"		P
组织 • 新建文件	共 央						111 -	0
 ○ 此用题 ◎ 採環 ◎ 照片 ○ 文档 ◆ 下號 ♪ 音乐 ● 盧服 ○ 本地磁盘 (Ci) ○ USB-P (Hi) 	 名称 	~	學改日略	委員件匹配的项。	小大			
文件名(N):	Template							×
保存类型(工);	Project Files (*.uvproj))						~
▲ 隐藏文件夹						保存(5)	取消	

4. После появится окошко «Device», откройте диск где находится STMicroelectronics и выберете STM32F103RB.

Vendor: STMicroelectronics Device: STM32F100RB	
Jata base	Desgription:
STM32F100C4 STM32F100C6 STM32F100C8 STM32F100C8 STM32F100C8 STM32F100R4 STM32F100R6 STM32F100R8 STM32F100R8 STM32F100RB STM32F100RD STM32F100RD STM32F100RB	ARM 32-bit Cortex-M3 Microcontroller, 24MHz, 128kB Flash, 8kB SRAM, PLL, Embedded Internal RC 8MHz and 40kHz, Real-Time Clock, Power Saving Modes, JTAG and SWD, 7-channel DMA, Nested Interrupt Controller, 12-bit 16-ch A/D Converter, 2 12-bit 2-ch D/A converters, 51 Fast I/O Port 2 SPI, 2 I2C, 3 USART, CRC calculation unit, SysTick Timer, 2 Watchdog Timers, 16-bit 6-ch advanced timer with PWM, 6 16-bit Timers with Input Capture, Output Compare and PWM

5. В сплывающем окне «CopySTM32StartupCodetoproject....» появится предложение «добавить код запуска?» выберете «нет». В нашем встроенном программном обеспечении уже встроен документ код запуска.

6. После возвращения в Tempalte мы увидим, что там появились 3 файла:



В Tempalte необходимо создать три папки: CORE, USER, STM32F10x_FWLib.

· 主页 共家	立石						- 0
 (1) 約点要15 (2) 約点 (2) 約 (3) 約 (4) (4) (4) (4) (4) (4) (4) (4) (4) (4)	編 超大田版 100 小数6 121 平峰	日大田标 (2) 列助 内容	中國校 涂粉宫醫 史 東東方式	(9日改重・ 月11日・ 月11日の月日 月11日日月日 月11日日月日	 □ 项目复动幅 ☑ 文件扩展名 ☑ 交件扩展名 ☑ 除血的项目 	<u>ि</u> केस क्राइरहा	
創作		布局		当前视距	显示/除	8-	
	电脑 > 本地磁的	t (D:) > Temp	late			~	
4 下戦	* *						
國 文極	*				-		
副時 10月	1						
FileRecv	+						V4
stm32资料		CORE	STM32F10x FW1	USER	Template camuil	Templata unnot	Template compo
karm1_5		Cont.	b	each.	ucas	remplatements	(endomesiste of
- 本地磁盘 (Gi)							
制成							
2 此电脑							
1 12.00							
三回片							
國 文档							
· martin							
♣ 下數							
◆下数 ♪ 童乐							
● 「「」 「「」 「」 「」 「」 「」 」 「」 「」 「」 」 「」 「」 」 「」 「							
◆ 下数 ♪ 首乐 重度 重型 本地磁差 (C:)							
◆ 千転 〕 宣乐 重 東面 重 本地理是 (C) 二 本地理是 (C)	d.						
◆ 下数 点面 点面 本均磁金(C:) 本均磁金(D) ● 本均磁金(D)	÷						
★ 下数 直 重 重 重 重 重 重 重 数 数 型 差 (C:) 二 本地 数 差 (C:) 二 本地 数 差 (C:) 二 本地 数 差 (C:) 二 本地 数 差 (C:)	ł						

7. Далее необходимо из встроенного программного обеспечения скопировать исходный код и вставить его в наш новый проект. Для этого необходимо открыть папки со встроенным программным обеспечением:

STM32F10x_StdPeriph_Lib_V3.5.0\Libraries\STM32F10x_StdPeriph_Drive. Скопировать src,inc и вставить в STM32F10x_FWLib.

8.

文件	主页	共享 重着									-
★ 和定到"快 速访问"	2 21	 れは	径 建方式	ष स्रथम स्रथम	no No	新建文件夹	 1 1 1 2 4 4	屋住	目打开。 〕 编辑 ● 历史记录 打开	全部派 計 全部派 ●計 全部派 ● 反向加 通择	封译 [0]補 [5]择
÷ - •	1	🧧 « Libraries)	STM3	2F10x_StdPeriph	Driver >				~ č	搜索*ST	p
 ★ 快速: ● 重面 ◆ 下動 ○ 文相 ○ 面片 ○ File ○ stm ○ xarr ○ 本地 	方问 1 2 32资料 m1_5 城磁盘(4 (Gi)		nc. src			修改日期 2014-09-01 22 2014-09-01 22	1 130 J 130 J	如型 文件夹 文件夹	大小	
 此电 10.5 10.5 10.5 	ă Į	,	e e							F	

9. Сейчас нам необходимо скопировать из встроенного программного обеспечения все необходимые для движения папки в CORE.

Откройте встроенное программное обеспечение:

STM32F10x_StdPeriph_Lib_V3.5.0\Libraries\CMSIS\CM3\CoreSupport, скопируйте core_cm3.c и core_cm3.h и переместите их в CORE. После чего перейдите в STM32F10x_StdPeriph_Lib_V3.5.0\Libraries\CMSIS\CM3\DeviceSupport\ST\STM32F10x\sta rtup\arm скопируйте все находящиеся там элементы и перенесите в CORE.

	布局	当前视圈
> 此电脑 >	本地磁盘 (D:) > Template > COR	E
	Core_cm3.c	
	b core_cm3.h	
*	Startup_stm32f10x_cl.s	
*	₽8 startup_stm32f10x_hd.s	
*	Startup_stm32f10x_hd_vl.s	
*	Startup_stm32f10x_ld.s	
	Startup_stm32f10x_ld_vl.s	
	₽a startup_stm32f10x_md.s	
	Startup_stm32f10x_md_vl.s	
	Startup stm32f10x xl.s	

10. Перемещение в каталог

По данному адресу

STM32F10x_StdPeriph_Lib_V3.5.0\Libraries\CMSIS\CM3\DeviceSupport\ST\STM32F10x находятся три файла: stm32f10x.h, system_stm32f10x.c и system_stm32f10x.h. Их необходимо переместить в наш USER каталог.

После чего с STM32F10x_StdPeriph_Lib_V3.5.0\Project\STM32F10x_StdPeriph_Template необходимо скопировать и вставить в USER 4 файла: main.c, stm32f10x_conf.h, stm32f10x_it.c, stm32f10x_it.h

本	b磁盘 (D:) → Template → USER	
^	名称	修改日期
	CL main.c	2013-08-27 8:56
	h stm32f10x_conf.h	2013-08-27 8:56
	stm32f10x_it.c	2013-08-27 8:56
	stm32f10x_it.h	2013-08-27 8:56
	system_stm32f10x.c	2013-08-27 8:56

11. Все проделанные в основном программном обеспечении 10 этапов нам необходимо скопировать в наш новый проект.

Правой кнопкой мыши нажмите на Target1 и выберете Manage Components. D:\Template\Template.uvproj - µVision4 File Edit View Project Flash Debug Peripherals Tools SVCS Window Help 🛞 🖪 🎬 🥔 🔜 🕌 Target 1 N 📥 🖶 Project **a** Targe Options for Target Target 1'... Alt+F7 Open File Open List File Open Map File Rebuild all target files Build target F7 Translate File Stop build Add Group... Add Files to Group... Remove Item Anage Components... Show Include File Dependencies

12. Project Targets

Переименуйте Target в Template, затем в Groups удалите один столбец и создайте три группы: USER,CORE,FWLIB, нажмите «ок».

the second second second second		
Project 0		
iii 📜 Target 1		
the second se		
	-	-
	Components, Environment and Books	×
	Frojest Components Felders/Estensions Books	
	Project Targets C X 7 + Strouges C X 7 + Price: X 1	
	CORE EWUB	
		_
	get as Current Target	1
		100 C

20

ile Edit View Project	Flash Debug Periphe	rals Tools SVCS	Window He	
	11	[- (A (A (2)	s≓ s≓ //=	
	lemplate			
Template USER CORE FWLIB				SUPERPERIN

13. Добавляем в Group все необходимые нам файлы.

Правой кнопкой мыши нажимаем на Tempate, выбираем Manage Components и добавляем все необходимые нам файлы в Group. Во-первых выбираем FWLIB, нажимаем на Add Files, заходим в STM32F10x_FWLib/src, с помощью Ctrl+A выбираем все файлы и нажимаем Add и после Close.

oject Targets: 🔛 🗙 🕈 🗲	Groups: 📉 🗙 🗲 🗲	Files:	× + +
mplate	USER CORE FWLIB	misc.c stm 32/10x_adc.c stm 32/10x_bkp.c stm 32/10x_can.c stm 32/10x_can.c stm 32/10x_can.c stm 32/10x_can.c stm 32/10x_dac.c stm 32/10x_dbgmcu.c stm 32/10x_dbgmcu.c stm 32/10x_fash.c stm 32/1	~
		stm32/10k_rcc.c stm32/10k_rcc.c stm32/10k_sdio.c	

14. С помощью аналогичного способа добавляем из Group в CORE и USER все необходимые нам файлы. В CORE добавляем core_cm3.c, startup_stm32f10x_md.s, USER, после добавляем main.c, stm32f10x_it.c, system_stm32f10x.c и нажимаем «ок». Components, Environment and Books

Template	Groups:	7 +	hiles:	× 7
	FWLIB		stm32f10x_tt.c system_stm32f10x.c	
Set as Current Tarrent			Add Bl	

Project Targets: 🞦 🗙 🗲 🗲	Groups:	🖸 🗙 🗲 🗲	Files:	×	+ +	
Template	CORE FWLIB		core_cm3.c startup_stm32f10x_md.s	5		
						10.2.
						EX.
Set as Current Target			Add Files		L	
			3 and a second s			
	OK	Cancel			Help	

	WA Helb	
	■ //= //= Control	
🧼 🔛 📅 🥔 🔜 🔤 Templa		
oject 🛛 📮 🗾		
Template USER USER System_stm32f10x_it.c System_stm32f10x_c CORE Startup_stm32f10x_r Startup_stm32f10x_r Stm32f10x_adc.c Stm32f10x_dc.c Stm32f10x_cec.c Stm32f10x_cec.c Stm32f10x_dda.c Stm32	There	esit

15. Далее нам необходимо запрограммировать MDK на самостоятельное определение местонахождения необходимых файлов.

Вернитесь в главное меню нового проекта, нажмите на «волшебную палочку» и выберете с/с++, затем нажмите кнопку справа от IncludePaths и добавьте три проекта (как на картинке).

Keil должен находится только в главном каталоге, a path в последнем.



	Device Target Output Listing Use	r C/C++ Asa Linker Debug Utilities	1	
	Define:			
	Ugdefine:			
	Cotinization:	Strict &NSI C Warnings: Enum Container always int Curispective	d> •	
	Optimize for Time Solit Load and Store Multiple One ELF Section per Function	Bein Charlis Signed F Thurb Read-Only Position Independent No Auto Bead-Write Position Independent	Mode .	
	Include Paths Milec			
	Controls Compiler control 4 D:\Kei\ARM\RV31\INC string	ics=interwork	<u>.</u>	
		Cancel Defaults	Help	
lder Setu	p		? ×	
etup Compi	er Include Paths:	<u>m</u>	× + +	
CORE STM32F10	x FWLib\inc			
USER				
JJEN			1	

16. Библиотека ST поддерживает несколько моделей микроконтроллеров

Как и в 15 шаге, нажмите на c/c++ и скопируйте вставьте: STM32F10X_MD,USE_STDPERIPH_DRIVER в Define.

	· · · · ·		
eprocessor Symbols			
Define: STM32F10X_MD,USE_	STDPERIPH_DRIVER		
ndefine:			
nguage / Code Generation			
	T Strict ANSI C	Wamir	igs:
timization: Level 0 (-00) 💌	Enum Container	always int ounsp	ecfied>
Optimize for Time	Plain Char is Sig	ned 🗖 Th	um <u>b</u> Mode
Split Load and Store Multiple	Read-Only Posit	ion Independent 🛛 🗖 No	Auto Includes
One ELF Section per Function	☐ <u>B</u> ead-Write Pos	ition Independent	
nclude CORE:\STM32F10x	FWLib\inc:.\USER		
Paths			
ontrols			
ompiler -c -cpu Cortex-M3 -g -O	0apcs=interwork -I.\CORE -I.	\STM32F10x_FWLib\inc -I.\	USER A
control -I D:\Keil\ARM\RV31\I	NC		

17. Зайдите в USUR проекта и скопируйте и вставьте код в main.c ,

#include"stm32f10x.h"

```
#include"stm32f10x_gpio.h"
```

GPIO_InitTypeDefGPIO_InitStructure;

intmain(void)

{

SystemInit();

```
RCC_APB2PeriphClockCmd(RCC_APB2Periph_GPIOB,ENABLE);
```

```
GPIO_InitStructure.GPIO_Pin=GPIO_Pin_9;
```

GPIO_InitStructure.GPIO_Speed=GPIO_Speed_50MHz;

GPIO_InitStructure.GPIO_Mode=GPIO_Mode_Out_PP;

```
GPIO_Init(GPIOB,&GPIO_InitStructure);
```

while(1)

{

```
/*SetPB9*/
```

GPIO_SetBits(GPIOB,9);

/*ResetPB9*/

```
GPIO_ResetBits(GPIOB,9);
```

18. Для компиляции нажмите на кнопку как на картинке

}

}



На картинке видно, что компиляция прошла успешно.

После успешной компиляции необходимо провести настройку и преобразование файлов в hex файл.

Нажмите на «волшебную палочку», в настройках выберете Output и поставьте галочки на всех вариантах выбора.

Select Folder for Objects	Name of Executable: Template	
Create Executable: .\Template Debug Information Create HEX File Browse Information	☐ Create Batch File	
Create Executable: \Template.lib		- Lesi

8.2 Толкование программы

На основании приведенной выше информации мы можем написать функцию инициализации порта ввода-вывода LED.

```
void InitLED(void)
GPI0_InitTypeDef GPI0_InitStructure;
RCC_APB2PeriphClockCmd(RCC_APB2Periph_GPI0B, ENABLE); //使能PB端口时钟
GPI0_InitStructure.GPI0_Pin = GPI0_Pin_9; //LED2_PB9 端口配置
GPI0_InitStructure.GPI0_Mode = GPI0_Mode_Out_PP; //推挽输出
GPI0_InitStructure.GPI0_Speed = GPI0_Speed SOMHz;
GPI0_Init(GPI0B, &GPI0_InitStructure); //配置I0
}
#define LED_ON____0
```

Сейчас мы можем записать 0 или 1 в IO для управления включением или выключением светодиода.

LED = LED ON; //亮灯

LED_ON это макрокоманда, которая обозначается 0.

#define LED OFF

Так же мы можем видеть, что просто LED это тоже микрокоманда, обозначающая PBout(9).

#define LED PBout (9)

define BIT ADDR (addr.)	MEN ADDR (BITBAND (addr. bitrum))
//TOULED DEL MADE LEGALY A	A county match match (by a compo (down y wa county))
define GRION ODR Addr	(GPT03 B35F+13) //0x40010800
define GPTOB ODP Addr	(GRTOR RESEATC) (/0×40010/00)
define CDIOC ODD Addr	(CETCO BASETIC) //Artoninooc
define OPTOD ODP Addr	(GPICO_DAGETIC) //OrfOOILAGC
define CPTOD ODK Addr	
define SPICE ODE Addr	(GFIDE BASETIC) //ORTOOIISOC
define GPTOC ODB Addr	(GFIOE DASETIE) //ORTOVIIAUC
FORTING GPICG_ODK_Addr	(GEIDG_DW2E+12) //OE40011EDC
define GPIOA IDR Addr	(GPICA BASE+8) //0x40010808
define GPIOB IDR Addr	(GPIOB BASE+5) //0x40010C08
define GPIOC IDR Addr	(GPIOC BASE+8) //0x40011008
define GPIOD IDR Addr	(GFICD BASE+8) //0x40011408
define GPIOE IDR Addr	(GPICE BASE+8) //0x40011808
define GPIOF IDR Addr	(GPIOF BASE+0) //0x40011A08
Edefine GPIOG IDR Addr	(GPIOG BASE+8) //0x40011E08
//Io口操作,只对单一的Iol	
//确保n的值小于16!	
define PAout (n) BIT 2	DDR (GFICA ODR Addr. n) //輸出
define Dite (a)	

Данные коды мы можем найти в include.h.

Преобразование LED В PBout(9)

PBout(9) B BIT_ADDR(GPIOB_ODR_Addr, 9)

BIT_ADDR(GPIOB_ODR_Addr, 9) B

MEM_ADDR(BITBAND(GPIOB_ODR_Addr,9)

MEM_ADDR(BITBAND(GPIOB_ODR_Addr, 9) B *((volatile

unsigned long*)(BITBAND(GPIOB_ODR_Addr, 9))) B *((volatile unsigned

long*)(((GPIOB_ODR_Addr &

0xF000000)+0x2000000

+((GPIOB_ODR_Addr & 0xFFFFF)<<5)+(9 <<2)))

Здесь использовалась функция битого поля stm32. Регистры Stm32 не могут быть обработаны в битах. Чтобы управлять конкретным портом ввода-вывода, мы должны использовать комбинацию сдвига логических операций. Для удобства использования іо портов, stm32 предоставляет функцию битового поля.

8.3 Порядок записи

1. Откройте программу для записи

1
mcuisp.exe

<mark>≇</mark> mcuisp 系统(Y) 帮 www.mcuis	V0.993 助(Z) p.com :)单片机 Languag 免费软件	在线的 ge 推 脱机	詹程专家www.mcuisp.com 搜索串□(X) <u>P</u> ort:COM1 <u>b</u> ps:115200 /斋线编程器(<u>W</u>)		-	×
联机下载时的 hts\Tencent\	的程序文 QQ\904	<u>7件:</u> 429963\Fil	eRecv	/\STM32\Obj\OpenArmSTM32.hex	☑编程前重装文件		
手持万用编辑 开始	呈器 5	TMESP \$	· 英ST	MIAP LPC2xxxISP 核验 编程后执行 使用RamIsp			-
读器件信则	盟(R)	清除。	5片(2	连续烧录模式 () 读FLASH			
法项字节区 □编程到1	: ASHB1j3	写选项字节	ħ				
读保护: AS	硬件	非选项:	FF	设定选项字节等			
写保护: [7]	FF	FF	FF	用户数据0-1 FF FF			
DTR的高电平	2复位,R	TS高电平	进Boo	otLoader			 ×.
						_	 4

Выберете STMISP, остальные пункты оставьте выбранными по умолчанию.

\Tenc	ent\Q0	2\90442	9963\F	ileRecv	\STM32\Obj\Ope	enArmS	TM32.hex	 ☑编程前	重装文件	
持万用	1编程	STN	MISP :	免费ST	MIAP LPC2xxx	ISP	T			^
ł	开始维	解程(P)			校验 编程后执行 使用RamIsp 连续焕录模式					
读器件	 信息 	(R)	清除	芯片(Z) 读F	LASH				
选项字*]编程	节区: 到FLAS	SH时写j	选项字	节						
卖保护:	A5	硬件》	匙项:	FF	设定选项	字节等	F			
写保护:	FT	FF	FF	FT	用户款据0-1	FF	22			

2. Выберите диск и папку для записи.

# 這冊要下數的文件						×
+ 🚺 = De	ocuments + Te	encent > QQ > 904	29963 > FileRecv > STM3	2 > Obj	~ 0 被衆*06	٩.
组织 • 新建文件夹						81 · 🖬 🛛
~ 🛄 出电版 ^	名称	<u>^</u>	伊波日期	9922	大小	
 ○ 圖 初版 ○ 题片 ○ 文档 ○ 大橋 ○ 素坊総直 (C) ○ 本坊総直 (C) 	DpenArn	nSTM32.hex	2017-10-26 14:14	HEX 文体	72 KB	
> 🧝 data (\\LXBUN						
文件名	50 <u>10</u> ;				~ 文件格式	hex, a79. sim, ms _i ~
	COLOR DA				#1开(Q) RUM

3. Откройте соответствующий файл. Снимите заглушку на основной плате stm32. Снятая заглушка обозначает переход с обычного режима в режим записи.



Соедините плату и компьютер с помощью USB кабеля и нажмите «Найти последовательный порт»

≇ mcu 系统(Y) www.mc	isp Vi 春助 uisp.o	0.993- (Z) L	-単片杉 angua 書软件	l在线 ge 非脱机/	浅编程专家www.mcuisp.com 捜索串口(X) Port:COM1 bps:115 和/憲线编程器(W)				200		-	
 秋初, 1) 朝 hts\Tenci	ent\QC	至序又1 2\90442	∓: 29963\F	ileRecv	\STM32\Obj\Ope	en Arms	STM32.h	x 🛄]编程前重装	这件	
手持万用	编程	STI	MISP	免费ST	MIAP LPC2xxx	ISP						
3	开始绑	韻程(P)			校验 编程后执行 使用RamIsp 连续烧录模式							
读器件	信息(R)	清除	芯片(Z) 读F	LASH						
选项字1 □编程	节区: 到FLAS	SH时写	选项字	节								
读保护:	AS	硬件	选项:	FF	设定选项	字节	争					
写保护:	TT	FF	FF	FF	用户数据0-1	FF	FF					
-31410-			1	122] MJ/~8100'1							

4. Выберете необходимый последовательный порт

手持万用编程器 STMISP 免责STMIAP LPC2xxxISP 一 グ 税验 ダ 機強 一 グ 编程后执行 一 使用Ramisp 」 连续娱录模式	
开始编程(P)	
读器件信息(R)	
选项字节区:]编程到FLASH时写选项字节	
卖保护: A5 硬件选项: FF 设定选项字节等	
写保护: FF FF FF 用户数据0-1 FF FF	

5. Нажмите «Начать программировать»

www.mc 联机下载 hts\Tenci	uisp.《 时的 ent\Q(com 免费软(程序文件: 2\904429963\	牛脱机 FileRecv	高线编程器(W) \STM32\Obj\OpenArmSTM32.hex	
手持万用 读器件 违项字节	1编程 开始继 注信息(市区: 2010年3月	器 STMISP 解程(P) R) 清朝 SH11号法面合	免费51	MIAP LPC2xxxISP 校验 编程后执行 使用RamIsp 连续烧录模式) 读FLASH	RTS維持高 开始連接2,接收到:79 在車口COM3連接成功@115200bps,耗时297毫 秒 芯片内BootLoader版本号: 2.2 芯片PID: 0000410 芯片FLASH容量为128KB 芯片SRAM容量为65535KB(此信息仅供参考,新 版本芯片己不包含此信息) 956005芯片唯一序列号。
读保护:	A5	硬件选项:	FF	设定选项字节等	34FF66063446393117532257
写保护:	FF	TF TT	FF	用户数据0-1 77 77	後日的近似テ节: A55AFF00FF00FF00FF00FF00FF00 全片擦除成功 第406毫秒,己准备好 己写入1KB,进度4%,耗时859毫秒

6. После завершения появится надпись «программирование завершено»

ts\rend	ent\Q0	2\90442	9963\F	leRecv	\STM32\Obj\Ope	enArmS	TM32.he	xx 团编程前重装文件
手持万用	编程	STM	ISP 1	免费ST	MIAP LPC2xxx	ISP		90 10 10 10 10 10 10 10 10 10 10 10 10 10 1
	开始维	肩程(P)			校验 编程后执行 使用RamIsp 连续烧录模式			芯片内BootLoader版本号: 2.2 芯片PID: 00000410 芯片FLASH容量为128KB 芯片SRAM容量为5535KB(此信息仅供参考,新 にカリースを含めた良)
读器件	信息((R)	清除	芯片(Z) 读F	LASH		版本心方亡不包含此信息) 96位的芯片唯一序列号:
选项字*	市区; 到FLAS	SH时写)	选项字	节				34FF66063446393117532257 读出的选项字节: A55AFF00FF00FF00FF00FF00FF00FF00
读保护:	A 5	硬件设	悲项:	FF	设定选项	字节制	ş.	全片操除成功
写保护:	FF	FF	FF	FF	用户数据0-1	FF	FF	共写入25KB,进度100%,耗时11016毫秒
								成功从08000000开始运行 <u>Max.sculap.com</u> 愈您报告,命令执行完毕, 一切正常

Вы можете производить программирование удобным Вам способом.

*Скачать все исходные файлы можно по ссылке: https://www.lobot-robot.com/c_detail/15.html

Для скачивания файлов необходимо зарегистрироваться на сайте производителя и войти под своей учётной записью. При скачивании нужно ввести пароль bhn5