

ВСТРОЕННОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ УЗЛА ПЕЧАТНОГО R DCI M ИЗ СОСТАВА ПРОГРАММНОГО КОМПЛЕКСА УПРАВЛЕНИЯ СИСТЕМОЙ ЦИФРОВОЙ МНОГОФУНКЦИОНАЛЬНОЙ ПРОМЫШЛЕННОЙ СВЯЗИ «GIT-COMM»

наименование и индекс изделия

ИНСТРУКЦИЯ ПО УСТАНОВКЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

обозначение документа



www.git-holding.ru Тел: +7 (495) 223-07-25 E-mail: git@git-holding.ru Инструкция по установке программного обеспечения РОФ.ГРЛМ.01.01.05006-01 93 01 Редакция 1.0 R 16 DCI 01 M

ВСТРОЕННОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ УЗЛА ПЕЧАТНОГО R DCI M



ОГЛАВЛЕНИЕ

Введение	3
1 Подготовка к программированию	3
2 Программирование микроконтроллера ATMega128-16AU	4
2.1 Необходимый комплект оборудования и ПО	4
2.2 Загрузка встроенного ПО в программатор E-LAB	4
2.3 Загрузка встроенного ПО в микроконтроллер	14
3 Программирование микроконтроллера STM32F429VGT	15
3.1 Необходимый комплект оборудования и ПО	15
3.2 Подключение программатора к плате	15
3.3 Загрузка программного обеспечения	16
4 Завершение загрузки программного обеспечения	17



ВСТРОЕННОЕ ПРОГРАММНОЕ R 16 DCI 01 М ОБЕСПЕЧЕНИЕ УЗЛА ПЕЧАТНОГО R DCI M



ВВЕДЕНИЕ

В настоящей инструкции приведен состав комплекта необходимого оборудования и программного обеспечения для программирования устройства управления распределением мощности выходной усилителей звука R 16 DCI 01 M (далее – устройство), даны указания и описан процесс загрузки программного обеспечения в микроконтроллеры устройства.

Загрузка встроенного программного обеспечения производится в микроконтроллеры ATMega128-16AU и STM32F429VGT, установленные на платах устройства.

1 ПОДГОТОВКА К ПРОГРАММИРОВАНИЮ

В процессе подготовки к загрузке программного обеспечения в микроконтроллеры плат необходимо выполнить следующие операции:

- 1) отключите усилитель от сети и резервного источника питания;
- 2) отсоедините разъемы Х4 и, если они подключены, Х3, Х2 и Х1;
- 3) выверните винты, обведенные красным кружком (рисунок 1);



Рисунок 1

4) выньте платы из корпуса;

5) подключите плату к источнику питания с рабочим напряжением 60 В постоянного тока в соответствии со схемой, приведенной на рисунке 2;

6) ВНИМАНИЕ!!! Плата теперь включена!

На этом подготовка платы к программированию закончена. Дальнейшие действия по программированию микроконтроллеров ATMega128-16AU и STM32F429VGT приведены в разделах 2 и 3 настоящего руководства по программированию. Загрузка программного обеспечения в платы производится аналогично.



ВСТРОЕННОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ УЗЛА ПЕЧАТНОГО R DCI M







2 ПРОГРАММИРОВАНИЕ МИКРОКОНТРОЛЛЕРА ATMEGA128-16AU

2.1 Необходимый комплект оборудования и ПО

2.1.1 Для программирования микроконтроллера ATMega128-16AU требуется следующий комплект оборудования и программного обеспечения:

- персональный компьютер (ПК) с операционной системой Windows не ниже Windows 7;
- программатор E-LAB UPP-1;
- кабель для подключения программатора E-LAB UPP-1 к порту USB персонального компьютера;
- программное обеспечение «E-LAB ISP-ICP Programmer»;
- актуальные файлы прошивки.

2.1.2 Перед проведением работ по программированию микроконтроллера программное обеспечение «E-LAB ISP-ICP Programmer» должно быть установлено на персональный компьютер.

2.1.3 Работы по программированию микроконтроллера проводятся в два этапа:

1) загрузка встроенного программного обеспечения в программатор E-LAB;

2) загрузка встроенного программного обеспечения из программатора в микроконтроллер.

2.2 Загрузка встроенного ПО в программатор E-LAB

2.2.1 Переместите все файлы обновлений (файлы с расширениями *.ispe и *.hex) в одну папку. Для примера на рисунке 3 используется папка: C:\Firmware\12DDL02\.



ВСТРОЕННОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ УЗЛА ПЕЧАТНОГО **R DCI M**



Size	
1 KB	
123 KB	
	123 KB



2.2.2 Подключите программатор E-LAB к ПК с помощью USB-кабеля. Затем запустите программное приложение E-LAB, дважды щелкнув значок 🙈.

Откроется окно программного обеспечения E-LAB ISP-ICP (рисунок 4).

Tojectionada	none		
Project Name Directory HexFile Name DiserFile Name	F G H I J K GroupA	join Files	CPU Project New Cite Cite Cite Cite Cite Cite Cite Cit

Рисунок 4

R 16 DCI 01 M

	ВСТРОЕННОЕ ПРОГРАММНОЕ	
R 16 DCI 01 M	ОБЕСПЕЧЕНИЕ УЗЛА ПЕЧАТНОГО	ГРУППА ИНДУСТРИАЛЬНЫХ ТЕХНОЛОГИЙ
	R DCI M	

2.2.3 Нажмите на «Search», чтобы найти необходимый файл встроенного ПО (рисунок 5).

E-LAB ISP-UPP Projects		
Project loaded	none	
Project Name Directory HexFile Name UserFile Name UserFile Name ABCDDE	F G H I J	CPU
Test 2313	GroupA	Project New Ti Delete Jelete Load Load

Рисунок 5

2.2.4 Выберите путь к папке с файлами прошивки, например, C:\Firmware\12DDL02\ (рисунок 6).

A Project Path	? ×
Proj: search a project	E-LAB 🙈
	File contents
C:\Firmware\12DDL02\	Cancel V OK

Рисунок 6



ВСТРОЕННОЕ ПРОГРАММНОЕ R 16 DCI 01 M ОБЕСПЕЧЕНИЕ УЗЛА ПЕЧАТНОГО **СТТ** ИНДИСТРИАЛЬНЫХ R DCI M



A Project Path		? X
Proj: search a project	E-LAB	A
■ :: []	File contents Compute	rs
C:\ - Dell - Drivers - Firmware L [2DDL02 - Hotfix - Intel - ModbusClient - PerfLogs - Program Files - Python27 - Users - Windows	12 DDL 02 R16.ispe	
C:\Firmware\12DDL02\	🗙 Cancel	🗸 ОК

Рисунок 7

2.2.6 Файл встроенного ПО теперь загружен в программное обеспечение E-LAB ISP-ICP (рисунок 8).

A E-LAB ISP-UPP	Projects		? ×
Project load	ded 12 DDL 02 R16		
Project Name Directory HexFile Name EEpFile Name UserFile Name A B C	Test 2313 C:\Program Files\E-LABprog\ S2313.hex S2313.eep D E F G H I J K L GroupA	join Files	90S2313 Project New Colored Project New Colored Project New Colored Project New Colored C

Рисунок 8



www.git-holding.ru Страница 7 / 18

ВСТРОЕННОЕ ПРОГРАММНОЕ R 16 DCI 01 M ОБЕСПЕЧЕНИЕ УЗЛА ПЕЧАТНОГО ССТТ ИНДУСТРИАЛЬНЫХ R DCI M

ВНИМАНИЕ!!! Нажав на кнопку **Delete**, вы удалите файл не только из ПО программатора E-LAB, но и из папки Windows, в которой он хранится! Убедитесь, что у вас есть резервная копия файлов программного обеспечения!

2.2.7 Назначьте файл прошивки и нажмите кнопку «Load» (рисунок 9).

Рисунок 9

Файлы встроенного ПО (*.ispe и *.hex) теперь проверены. Время проверки может составлять от 10 секунд до 2 минут.

2.2.8 После проверки появляется сообщение об ошибке по умолчанию «Target power down!». Нажмите на кнопку «Abort» (рисунок 10).

E-LAB ICP-ISP Programmer [AVR] [12 DDL 02 R16]	x
File Device Program Options Encrypt Tools Help	
Action: check device	4.02
ad Error X3 08 09 0A 0B 0C 0D 0E Farsti 18 95 10 94 10 19 10 10 10 10 10 10 10 10 10 10 10 10 10 <	•
	SPI
adress 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F ascii	
00000 FF F	-
CPU : MEGA128 Clock : 8 MHz 0.00V/0mA UPP-1 Prog Programmer p: 0	

Рисунок 10



www.git-holding.ru Страница 8 / 18 Инструкция по установке программного обеспечения РОФ.ГРЛМ.01.01.05006-01 93 01 Редакция 1.0

ВСТРОЕННОЕ ПРОГРАММНОЕ R 16 DCI 01 М ОБЕСПЕЧЕНИЕ УЗЛА ПЕЧАТНОГО ССТТ ИНДУСТРИАЛЬНЫХ R DCI M

2.2.9 Чтобы загрузить файлы встроенного ПО в программатор E-LAB, нажмите на красный символ телефона на панели значков вверху (рисунок 11).

A	E-LAE	B ICF	P-ISF	Pro	gra	mm	er [A	VR]	[12	2 DD	L 02	2 R1	6]								x
Fi	le D	evice	e P	rog	ram	0	ptio	ns	En	cryp	ť	Tool	s	Help)						
			~			0	1	4	<u>s.</u>	<u>s</u>		-12	51		1		1	Acti	on: none		
	- I	9	¥	-	Ľ.,	₽?			*	T	V	9				:				4.	02
L 1	adr		0.0	01	02	0.2	0.4	05	06	07	0.0	0.9	03	OP	00	0.0	07	05			
L H	looo	000	00	01	5.2	40	09	0.0	00	40	10	05	10	05	10	00	10	02	ascii e		<u>_</u>
	000	010	18	95	18	40	18	95	18	95	18	95	18	95	18	95	18	95		L	- 1
	000	020	18	95	18	95	18	95	18	95	18	95	18	95	18	95	18	95			
	000	030	18	95	18	95	18	95	18	95	18	95	18	95	18	95	18	95			
	000	040	18	95	18	95	18	95	18	95	0C	94	A1	30	18	95	18	95			
	000	050	18	95	18	95	18	95	18	95	18	95	18	95	18	95	18	95			
	000	060	18	95	18	95	18	95	18	95	0C	94	BD	41	0C	94	70	44	ApD		
	000	070	0C	94	95	44	18	95	18	95	0C	94	05	31	18	95	18	95	D1		
	000	080	18	95	18	95	18	95	18	95	18	95	18	95	45	45	57	20	EEW		
1 1	000	090	27	61	64	64	27	20	27	77	6F	72	64	27	20	20	20	20	'add' 'word'		Ŧ
ſ	חחחח	חחח	ппп	ппп		ппп	ппп			חחח	חחר		חחח	ппп	וחחח	חחח	וחחח			S	Ы
											JUU							יוחר			<u>.</u>
	adr	ess	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	OD	0E	OF	ascii		^
	000	000	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	•••••		
	000	010	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	•••••		
	000	020	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	•••••		
	000	030	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	•••••		
	000	040	22	E E	22	22	22	22	22	FF	TT.	22	22	22	22	22	E E	22			
	000	050						TT.		E E	EE EE						E E	TT.			
	000	070	FF	11	FF	FF	TT	FF	FF	FF	FF	FF	FF	FF	FF	FF	11	FF			
	000	080	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF			
	000	090	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF			-
					17																
Fla	sh cor	ntent	ts do	bubl	eclic	k to	edi	t													
СР	U : ME	GA1	28			CI	ock	:81	ИHz			0.00)V/0	mΑ			UPP	-1 P	rog Programmer	p: 0	

Рисунок 11

Через некоторое время, в зависимости от размера файла прошивки, появится всплывающее окно «**UPP-USB File administration**», в котором вам необходимо сохранить файл прошивки для программатора. Появление окна сопровождается звуковым сигналом.

В зависимости от используемой версии программатора E-LAB, программатор может управлять до 10 каналами для файлов встроенного ПО.

Изменять каналы можно на задней панели программатора E-LAB.

Выбранный в данный момент канал будет обозначен символом «=>» в левой части окна «UPP-USB File administration» (рисунок 12).

ВСТРОЕННОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ УЗЛА ПЕЧАТНОГО R DCI M



8 x UPP-USB File administration [12_DDL_02_R16] Maintenance חחח 🔁 reload list | => 0 1 🗙 delete all 2 3 🖉 delete File 4 📔 pack file 5 6 🧠 encrypt file 7 8 C1 check file 9 ŝ 10 file <u>I</u>nfo ш Þ 📇 print erase CHIP prog CHIP verify CHIP <u>c</u>lose file delete ok 953 MB

Рисунок 12

2.2.10 Выберите опцию «pack file», нажав на соответствующую кнопку (рисунок 13).

UPP-USB File administration [12_DDL_02_R16]	2 X
	Maintenance
=> 0	reload list
1	🗙 delete all
3	delete File
4	
5	<u>pack file</u>
7	encrypt file
8	C¹ chec <u>k</u> file
9 10	👌 file Info
	🖺 print
erase CHIP	
file delete o	ok 953 MB

Рисунок 13



R 16 DCI 01 M

www.git-holding.ru Страница 10 / 18

ВСТРОЕННОЕ ПРОГРАММНОЕ R 16 DCI 01 M ОБЕСПЕЧЕНИЕ УЗЛА ПЕЧАТНОГО **СТТ** ИНДУСТРИАЛЬНЫХ R DCI M

2.2.11 Подтвердите появившееся сообщение об ошибке, нажав на «ОК» (рисунок 14).



Рисунок 14

2.2.12 Согласитесь с именем файла, нажав на «ОК» (рисунок 15).

Fi	lename	
m	ax 8 chars	
1	2_DDL_02_R16	T



2.2.13 Во всплывающем окне «ISP3+UPP Packfile Extension», вы можете выбрать предпочитаемый канал, активировав переключатель рядом с ним (рисунок 16). Если выбрано «none», прошивка копируется на следующий свободный канал.



Рисунок 16



ВСТРОЕННОЕ ПРОГРАММНОЕ R 16 DCI 01 M ОБЕСПЕЧЕНИЕ УЗЛА ПЕЧАТНОГО **СТТ** ИНДУСТРИАЛЬНЫХ R DCI M

2.2.14 Подтвердите, нажав кнопку «store file» (рисунок 17).

UPP File extensions	Filename
∩ none ∩	12_DDL_0
C .pk0	Extension
C .pk1	pac
C.pk2	, pao
C .pk3	Dock or DLL
C .pk4	
C .pk5	
C .pk6	
C .pk7	
C .pk8	
C .pk9	store file



2.2.15 Программатор E-LAB приступает к процедуре загрузки. Ход выполнения отображается на индикаторе выполнения в верхней части окна «UPP-USB File administration» (рисунок 18).

	Maintenance
=> 0	Preload jist
1	delete all
2	
3	delete File
4	
5	[h] pack file
6	C encrypt file
7	- ag <u>Encid</u> pende
8	C) check file
9	
10	8 file info
< III	•
	print
🖉 erase CHIP	HIP Sprog CHIP

Рисунок 18

ВСТРОЕННОЕ ПРОГРАММНОЕ R 16 DCI 01 M ОБЕСПЕЧЕНИЕ УЗЛА ПЕЧАТНОГО ССТАТИИ ИНДИСТРИАЛЬНЫХ R DCI M

2.2.16 После успешной загрузки можно увидеть, какой файл встроенного ПО сохранен в программаторе E-LAB. Завершите процесс, нажав кнопку «close» (рисунок 19).

UPP-USB File administration [12_DDL_02_R16]	S X
	Maintenance
=> 0 12_DDL_0 PAC 45056	reload list
2	🗙 delete all
3	delete File
4	pack file
6	encrypt file
8	B check file
9	
10	🧯 file Info
	Print 🗎
🖉 erase CHIP	<u></u> <u>c</u> lose
download o	k 953 MB

Рисунок 19

2.2.17 Закройте программное обеспечение E-LAB ISP-ICP Programmer, нажав на крестик в правом верхнем углу окна (рисунок 20).

A E-LAB ICP-ISP Programmer [AVR] [12 DDL 02 R16]	
File Device Program Options Encrypt Tools Help	
Action: none	
	4.02
	^_
000010 18 95 18 95 18 95 18 95 18 95 18 95 18 95 18 95 18 95	
000020 18 95 18 95 18 95 18 95 18 95 18 95 18 95 18 95 18 95	
000030 18 95 18 95 18 95 18 95 18 95 18 95 18 95 18 95 18 95	
000040 18 95 18 95 18 95 18 95 0C 94 A1 30 18 95 18 95	0
000050 18 95 18 95 18 95 18 95 18 95 18 95 18 95 18 95 18 95	1 -D
	1
000080 18 95 18 95 18 95 18 95 18 95 18 95 18 95 45 45 57 20	EEW
000090 27 61 64 64 27 20 27 77 6F 72 64 27 20 20 20 20 'add' 'word	1' +
	JUUUUUU SPI
adress 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F ascii	A
000000 FF	
000010 FF	
000020 FF	
000030 FF	
000050 FF	
000060 FF	
000070 FF	
000080 FF	
000090 FF	•••••
CPU : MEGA128 Clock : 8 MHz 5 Volt UPP-1 Prog Programm	er p: 0

Рисунок 20



ВСТРОЕННОЕ ПРОГРАММНОЕ R 16 DCI 01 М ОБЕСПЕЧЕНИЕ УЗЛА ПЕЧАТНОГО **СТТ** ИНДУСТРИАЛЬНЫХ R DCI M

2.2.18 Отсоедините программатор E-LAB от USB-кабеля.

2.3 Загрузка встроенного ПО в микроконтроллер

2.3.1 Для загрузки встроенного ПО в микроконтроллер, плата устройства должна быть подключена к источнику питания. Напряжение питания должно составлять от 60 В постоянного тока.

Встроенное ПО микроконтроллера должно быть загружено в программатор в соответствии с указаниями п. 2.2 настоящего Руководства по программированию.

2.3.2 Загрузка встроенного ПО в микроконтроллер производится в следующем порядке.

2.3.2.1 Подключите программатор к разъему платы, показанному на рисунке 21.

Как только программатор E-LAB будет подключен к плате, будет слышен короткий звуковой сигнал. Кроме того, светодиод в верхней части программатора E-LAB будет гореть постоянно.



Рисунок 21

2.3.2.2 Удалите предыдущую прошивку микроконтроллера, нажав кнопку «era» на программаторе.

2.3.2.3 Чтобы записать новую прошивку в микроконтроллер, нажмите кнопку «prog» на программаторе.

2.3.2.4 Когда процесс программирования завершится, будет слышен короткий звуковой сигнал. Программирование занимает около 10 секунд. В течение этого времени светодиод в верхней части программатора мигает.

2.3.2.5 После завершения процесса загрузки отключите программатор E-LAB от платы.

ВСТРОЕННОЕ ПРОГРАММНОЕ R 16 DCI 01 М ОБЕСПЕЧЕНИЕ УЗЛА ПЕЧАТНОГО **СТТ** ИНДИСТРИАЛЬНЫХ R DCI M

3 ПРОГРАММИРОВАНИЕ МИКРОКОНТРОЛЛЕРА STM32F429VGT

3.1 Необходимый комплект оборудования и ПО

3.1.1 Для программирования микроконтроллера STM32F429ZIT требуется следующий комплект оборудования и программного обеспечения:

- персональный компьютер (ПК) с операционной системой Windows не ниже Windows 7;
- программатор ST-Link V2;
- программное обеспечение STM32 ST-LINK Utility;
- актуальный файл прошивки.

3.1.2 Перед проведением работ по программированию микроконтроллера программное обеспечение STM32 ST-LINK Utility должно быть установлено на персональный компьютер.

3.2 Подключение программатора к плате

3.2.1 Программирование осуществляется через разъём XP3 платы. Расположение разъёма на плате показано на рисунке 22.



Рисунок 22

3.2.2 Подключите программатор к разъему XP3 платы. Схема подключения приведена на рисунке 23.



www.git-holding.ru Страница 15 / 18

R 16 DCI 01 M

ВСТРОЕННОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ УЗЛА ПЕЧАТНОГО R DCI M





3.3 Загрузка программного обеспечения

3.3.1 Подключите программатор к USB порту ПК и запустите программу STM32 ST-LINK Utility.

3.3.2 В окне программы откройте файл прошивки «File > Open file» (или сочетание клавиш CTRL + O).

3.3.3 В окне программы зайдите в меню «Target > Option Byte» и выберите там настройку «Level 0» (рисунок 24), затем нажмите «Apply».

ВНИМАНИЕ!!! Выбор опции «Level 2» приведёт к выходу из строя микроконтроллера платы.



R 16 DCI 01 M

ВСТРОЕННОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ УЗЛА ПЕЧАТНОГО R DCI M



Read Out Protection BOR Level F OFF R User configuration option byte INVDG_STDBY InBoot1 INVDG_STOP INVDG_ULP InBoot1 INSRAM_Parity FZ_INVDG_STOP INDBOOT INSRAM_Parity FZ_INVDG_STOP INDBOOT INSRAM_Parity FZ_INVDG_STOP INDBONT INSRAM_Parity FZ_INVDG_STOP INDBANK INSRAM_Parity FZ_INVDG_STOP INDBANK INSRAM_Parity FZ_INVDG_STOP INDBANK INSRAM_PE PCROP_ROP DB1M DUALBANK INST_STDP NSWBOOTO WMDG_SW INDACL2_VDD_Monitor INST_STDP NSWBOOTO WMDG_SW INDA_MONITO SDADC12_VDD_Monitor INST_MODE Sectority option bytes SEC_SIZE Dx00 BOOT_LOCK Boot address option bytes Data 1 [H] Boot from [H] Boot from [H] User data storage option bytes Data 1 [H] Flash protection Sector 1 Dx0800 16 K No Protection Sector 1 Dx0800 16 K No Protection Sector 3 Dx0800 16 K No Protection Secto	Option Bytes	×				
Image: Construction of the sector of the	Read Out Protection	BOR Level				
User configuration option byte WVDG_STOP I WDG_STDBY I BootI BOOTI SRAM_Parity FZ_IWDG_STOP DBBOTT BFB2 SRAM2_RST FZ_IWDG_STOP DBBNK INBOOT_SEL SRAM2_RST PE_PCROP_RDP DBIM DUALBANK INRST_STOP INSWBOOTO WDG_SW INRST_STOP INSWBOOTO WDG_SW INRST_STOP SEC_SIZE DX00 BOOT_LOCK Boot address option bytes SEC_SIZE DX00 SEC_SIZE2 Dx00 BOOT_LOCK Boot address option bytes BOOT_ADD0 (H) Boot from (H) BOOT_ADD1 (H) Boot from (H) User data storage option bytes Data 0 (H) Data 1 (H) Flash sectors protection Sector 1 0x0800 16 K No Protection Sector 1 0x0800 12 No Protection Sector 10 0x0800 12	Level 0 v	F OFF V R V				
Oser configuration option syste IWVG_STOP IWVG_STOP IWVG_STOP INVDG_STOP INVDG_STOP INVDG_STOP INVDG_STOP INVDG_STOP INVDG_STOP INVDG_STOP INVDG_STOP INRAM_Parity FZ_INDG_STOP INRAM_Parity FZ_INDG_STOP INRAM_Parity FZ_INDG_STOP INRAM_Parity FZ_INDG_STOP INRAM_Parity FZ_INDG_STOP INRST_STOP INTER Boot from [H] Boot from [H]	lies and such a star but					
WWDG_SW WWDG_ULP nBoot1 BOOT1 nSRAM_Parity FZ_IWDG_STOP nDBOOT BFB2 SRAM2_RST FZ_IWDG_STDBY nDBANK nBOOT_SEL SRAM2_RST FZ_IWDG_STDBY nDBANK nBOOT_SEL SRAM2_PE PCROP_RDP DBIM DUALBANK nRST_STDW nBoot1 WWDG_SW MRST_STDP NSWBOOT0 WWDG_SW MRST_STDPY VDDA_Monitor SDADC12_VDD_Monitor NRST_MODE SEC_SIZE2 0x00 BOOT_LOCK Boot address option bytes SEC_SIZE2 0x00 BOOT_LOCK Boot address option bytes Data 1 [H] Boot from [H] BOOT_ADD1 [H] BOOT_ADD1 [H] Boot from [H] Boot from [H] Image (H) Vset data storage option bytes Data 1 [H] Flash protection Image (H) Image (H) Flash protection mode: Write protection Image (H) Image (H) Image (H) Image (H) Image (H) Sector 1 0x0800 16 K No Protection Image (H) Image (H) Image (H) Image (H) Image (H) Image (H) Image (H)<	IWDG STOP	STDBY Bootfl BROOTI				
nSRAM_Parity FZ_WDG_STOP nDBOOT BF82 SRAM2_RST FZ_WDG_STDBY nDBANK nBOOT_SEL SRAM2_PE PCROP_RDP DB1M DUALBANK nRST_STDP nSWB0OT0 WWDG_SW MRST_STDP nSWB00T0 WWDG_SW MRST_STDP nSWB00T0 WWDG_SW MRST_STDBY VDDA_Monitor SDADC12_VDD_Monitor NRST_MODE Security option bytes SEC_SIZE 0x00 Security option bytes Boot from [H] Boot from [H] BOOT_ADD0 [H] Boot from [H] Boot from [H] User data storage option bytes Data 1 [H] Flash sectors protection Flash protection mode: Write protection Image: Sector 1 Sector 1 0x0800 16 K No Protection Sector 2 0x0800 16 K No Protection Sector 3 0x0800 16 K No Protection Sector 4 0x0800 16 K No Protection Sector 5 0x0800 16 K No Protection Sector 7 0x0800 16 K No Protection Sector 7 0x0800 12 No Protection Sector 9<	WWDG SW	ULP nBoot1 BOOT1				
SRAM2_RST FZ_IWDG_STDBY nDBANK nBOOT_SEL SRAM2_PE PCROP_RDP DB1M DUALBANK nRST_STDP nSWBOOT0 WVDG_SW MRST_STDP nSWBOOT0 WVDG_SW MRST_STDP NSWBOOT0 WVDG_SW MRST_STDP NUDA_Monitor SDADC12_VDD_Monitor NRST_MODE SEC_SIZE 0x00 BOOT_LOCK Boot address option bytes Boot from (H) Boot from (H) BOOT_ADD0 (H) Boot from (H) Boot from (H) User data storage option bytes Data 1 (H) Flash sectors protection Flash protection mode: Write protection Sector 1 0x0800 16 K No Protection Sector 2 0x0800 16 K No Protection Sector 3 0x0800 16 K No Protection Sector 4 0x0801 64 K No Protection Sector 7 0x0806 12 No Protection Sector 7 Sector 7 0x0806 12 No Protection Sector 7 Sector 7 0x0808 12 No Protection Sector 7 Sector 7 0x0806 12 No Protection Sector 7<	nSRAM Parity FZ IWI	DG STOP DB00T BFB2				
SRAM2_PE PCROP_RDP DB1M DUALBANK nRST_SHDW nBoot0_SW_Cfg IRHEN BOREN MRST_STOP nSWBOOT0 WDG_SW MRST_STDBY VDDA_Monitor SDADC12_VDD_Monitor NRST_MODE SEC_SIZE 0x00 BOOT_LOCK Boot address option bytes Boot from (H) Boot from (H) BOOT_ADD0 (H) Boot from (H) Boot from (H) BOOT_ADD1 (H) Boot from (H) Data 1 (H) Viser data storage option bytes Data 1 (H) Flash protection Flash protection Flash protection Image: Color (H) Sector 1 0x0800 16 K No Protection Sector 2 0x0800 16 K No Protection Sector 3 0x0800 16 K No Protection Sector 4 0x0801 64 K No Protection Sector 7 0x0808 12 No Protection Sector 7 Sector 7 0x0808 12 No Pro	SRAM2 RST FZ IWI	DG STDBY DBANK DBOOT SEL				
nRST_SHDW nBoot0_SW_Cfg IRHEN BOREN nRST_STOP nSWBOOT0 WDG_SW NRST_MODE WDDA_Monitor SDADC12_VDD_Monitor NRST_MODE SEC_SIZE 0x00 SEC_SIZE2 0x00 Security option bytes Boot address option bytes Boot from [H] Boot from [H] BOOT_ADD0 [H] Boot from [H] Boot from [H] User data storage option bytes Data 1 [H] Flash sectors protection Flash protection mode: Write protection Sector Start ad Size Protection Sector 1 0x0800 16 K No Protection Sector 2 0x0800 16 K No Protection Sector 3 0x0800 16 K No Protection Sector 4 0x0801 64 K No Protection Sector 5 0x0802 12 No Protection Sector 7 0x0806 12 No Protection Sector 6 0x0804 12 No Protection Sector 7 0x0806 12 No Protection Sector 7 <th>SRAM2 PE PCROP</th> <th>P RDP DB1M DUALBANK</th>	SRAM2 PE PCROP	P RDP DB1M DUALBANK				
Image: Start ad Size Protection Sector 1 0x0800 16 K No Protection Sector 2 0x0800 16 K No Protection Sector 3 0x0800 16 K No Protection Sector 3 0x0800 16 K No Protection Sector 4 0x0800 16 K No Protection Sector 5 0x0800 16 K No Protection Sector 7 0x0800 16 K No Protection Sector 8 0x0800 16 K No Protection Sector 9 0x0800 16 K No Protection Sector 7 0x0800 16 K No Protection Sector 9 0x0800 16 K No Protection Sector 7 0x0800 16 K No Protection Sector 7 0x0800 16 K No Protection Sector 9 0x0800 12 No Protection Sector 1 0x0800 12 No Protection Sector 10 0x0800 12 No Protection Sector 10 0x0800<	nRST_SHDW nBoot0					
✓ nRST_STDBY ∨DDA_Monitor SDADC12_VDD_Monitor NRST_MODE ✓ Security option bytes SEC_SIZE 0x00 SEC_SIZE2 0x00 BOOT_LOCK Boot address option bytes Boot from (H) Boot from (H) BOOT_ADD0 (H) Boot from (H) User data storage option bytes Data 0 (H) Data 1 (H) Flash sectors protection Flash protection mode: Write protection ✓ Sector Start ad Size Protection Sector 0 0x0800 16 K No Protection Sector 1 0x0800 16 K No Protection Sector 5 0x0800 16 K No Protection Sector 5 0x0800 16 K No Protection Sector 5 0x0801 16 K No Protection Sector 5 0x0802 12 No Protection Sector 7 0x0806 12 No Protection Sector 8 0x0808 12 No Protection Sector 9 0x0808 12 No Protection Sector 10 0x0802 12	nRST_STOP nSWB0	OOTO WDG_SW				
NRST_MODE Security option bytes SEC_SIZE 0x00 Boot address option bytes BOOT_ADD0 (H) Boot from (H) BOOT_ADD1 (H) Boot from (H) User data storage option bytes Data 0 (H) Data 1 (H) Flash sectors protection Flash protection mode: Write protection Sector 0 0x0800 16 K No Protection Sector 1 0x0800 16 K No Protection Sector 2 0x0800 16 K No Protection Sector 3 0x0800 16 K No Protection Sector 5 0x0802 12 No Protection Sector 6 0x0804 12 No Protection Sector 7 0x0806 12 No Protection Sector 9 0x0808 12 No Protection Sector 10 0x0800 16 K No Protection Sector 5 0x0802 12 No Protection Sector 7 0x0806 12 No Protection Sector 9 0x0808 12 No Protection Sector 10 0x0800 12 No Protection	✓ nRST_STDBY VDDA_	Monitor SDADC12_VDD_Monitor				
Security option bytes SEC_SIZE 0x00 SEC_SIZE2 0x00 BOOT_LOCK Boot address option bytes Boot from [H] Boot from [H] BOOT_ADD1 [H] Boot from [H] User data storage option bytes Data 1 [H] Data 1 [H] Image: Sector Start ad Size Protection Flash sectors protection Flash protection mode: Write protection Image: Sector 1 0x0800 16 K No Protection Sector 1 0x0800 16 K No Protection Image: Sector 3 0x0800 16 K No Protection Sector 2 0x0800 16 K No Protection Image: Sector 4 0x0801 64 K No Protection Sector 5 0x0802 12 No Protection Image: Sector 7 0x0806 12 No Protection Sector 7 0x0806 12 No Protection Image: Sector 7 0x0806 12 No Protection Sector 9 0x0806 12 No Protection Image: Sector 7 0x0806 Image: Sector 8 0x0808 Image: Sector 8 0x0808 Image: Sector 8 0x0808 Image: Sector 9 0x0806 Image: Sector 9 0x08080 Image: Sector 9	NRST_MODE	~				
SEC_SIZE Dx00 SEC_SIZE2 Dx00 BOOT_LOCK Boot address option bytes Boot from (H) Boot from (H) BOOT_ADD1 (H) Boot from (H) User data storage option bytes Data 1 (H) Data 1 (H) Flash sectors protection Flash sectors protection Flash protection mode: Write protection ~ Sector Start ad Size Protection ~ Sector 0 0x0800 16 K No Protection ~ Sector 1 0x0800 16 K No Protection ~ Sector 2 0x0800 16 K No Protection ~ Sector 3 0x0800 16 K No Protection ~ Sector 4 0x0801 64 K No Protection ~ Sector 5 0x0804 12 No Protection ~ Sector 9 0x0808 12 No Protection ~ Sector 10 0x0806 12 No Protection ~ Sector 10 0x0806 12 No Protection ~ Sector 10 0x0806 12	Security option bytes					
Boot address option bytes BOOT_ADD0 (H) Boot from (H) BOOT_ADD1 (H) Boot from (H) User data storage option bytes Data 0 (H) Data 1 (H) Flash sectors protection Flash sectors protection mode: Write protection Sector Start ad Size Sector 0 0x0800 16 K Sector 1 0x0800 16 K Sector 2 0x0800 16 K Sector 3 0x0800 16 K Sector 4 0x0800 16 K Sector 5 0x0800 16 K Sector 7 0x0800 16 K Sector 7 0x0800 16 K Sector 7 0x0800 16 K Sector 8 0x0800 16 K Sector 7 0x0806 12 No Protection Sector 8 0x0808 12 No Protection Sector 9 0x0808 12 No Protection Sector 10 0x0802 12 No Protection Sector 10 0x0802 12	SEC_SIZE 0×00	SEC_SIZE2 0x00 BOOT_LOCK				
BOOT_ADD0 (H) Boot from (H) BOOT_ADD1 (H) Boot from (H) User data storage option bytes Data 1 (H) Data 0 (H) Data 1 (H) Flash sectors protection Flash protection mode: Write protection ~ Sector Start ad Sector 0 0x0800 Bector 1 0x0800 Sector 2 0x0800 Sector 3 0x0800 Sector 4 0x0800 Sector 5 0x0800 Sector 5 0x0800 Sector 6 0x0800 Sector 7 0x0800 Sector 7 0x0800 Sector 8 0x0801 Sector 7 0x0802 Sector 7 0x0806 Sector 7 0x0806 Sector 7 0x0806 Sector 9 0x0808 Sector 9 0x0808 Unselect all Select all	Boot address option bytes					
BOOT_ADD1 (H) Boot from (H) User data storage option bytes Data 1 (H) Data 0 (H) Data 1 (H) Flash sectors protection Flash protection mode: Write protection ~ Sector Start ad Sector 0 0x0800 Sector 1 0x0800 Sector 2 0x0800 Sector 3 0x0800 Sector 4 0x0800 Sector 5 0x0802 Sector 5 0x0802 Sector 7 0x0802 Sector 7 0x0802 Sector 7 0x0804 Sector 7 0x0805 Sector 8 0x0808 Sector 9 0x0808 Sector 9 0x0808 Unselect all Select all	BOOT_ADD8 (H)	Boot from (H)				
User data storage option bytes Data 0 [H] Data 1 [H] Flash sectors protection Flash protection mode: Write protection Sector 1 0x0800 16 K No Protection Sector 1 0x0800 16 K No Protection Sector 2 0x0800 16 K No Protection Sector 3 0x0800 16 K No Protection Sector 3 0x0800 16 K No Protection Sector 4 0x0801 64 K No Protection Sector 5 0x0802 12 No Protection Sector 6 0x0804 12 No Protection Sector 7 0x0806 12 No Protection Sector 9 0x0808 12 No Protection Sector 9 0x0808 12 No Protection Sector 10 0x0800 12 No Protection Sector 10 0x0	BOOT_ADD1 (H)	Boot from (H)				
Data 0 (H) Data 1 (H) Flash sectors protection Flash protection mode: Write protection Sector Start ad Size Sector 0 0x0800 16 K No Protection Sector 1 0x0800 16 K No Protection Sector 2 0x0800 16 K No Protection Sector 3 0x0800 16 K No Protection Sector 4 0x0801 6 K No Protection Sector 5 0x0802 12 No Protection Sector 6 0x0804 12 No Protection Sector 7 0x0806 12 No Protection Sector 8 0x0808 12 No Protection Sector 9 0x0804 12 No Protection Sector 8 0x0808 12 No Protection Sector 10 0x0806 12 No Protection Sector 11 0x0806 12 No Protection Sector 10 0x0806 12 No Protection Sector 10 0x0806 12 <th>User data storage option bytes</th> <th></th>	User data storage option bytes					
Flash sectors protection Flash protection mode: Write protection Sector Start ad Sector 0 0x0800 Sector 1 0x0800 Sector 2 0x0800 Sector 3 0x0800 Sector 4 0x0800 Sector 5 0x0801 Sector 5 0x0802 Sector 7 0x0806 Sector 7 0x0806 Sector 7 0x0806 Sector 9 0x0806 Sector 8 0x0808 Unselect all Select all	Data 0 (H)	Data 1 (H)				
Flash protection mode: Write protection Sector Start ad Size Protection Sector 0 0x0800 16 K No Protection Sector 1 0x0800 16 K No Protection Sector 2 0x0800 16 K No Protection Sector 3 0x0800 16 K No Protection Sector 4 0x0801 64 K No Protection Sector 5 0x0802 12 No Protection Sector 7 0x0806 12 No Protection Sector 7 0x0806 12 No Protection Sector 9 0x0808 12 No Protection Sector 9 0x0808 12 No Protection Sector 9 0x0808 12 No Protection Sector 10 0x0806 12 No Protection Unselect all Select all Select all	Flash sectors protection	.,				
Flash protection mode: Write protection Sector Start ad Size Protection Sector 0 0x0800 16 K Sector 1 0x0800 16 K No Protection Sector 2 0x0800 16 K No Protection Sector 3 0x0800 16 K No Protection Sector 4 0x0801 64 K No Protection Sector 5 0x0802 12 No Protection Sector 6 0x0804 12 No Protection Sector 7 0x0806 12 No Protection Sector 9 0x0808 12 No Protection Sector 9 0x0808 12 No Protection Sector 10 0x0806 12 No Protection Sector 10 0x0806 12 No Protection Sector 11 0x0807 12	riash securs protection					
Sector Start ad Size Protection Sector 0 0x0800 16 K No Protection Sector 1 0x0800 16 K No Protection Sector 2 0x0800 16 K No Protection Sector 3 0x0800 16 K No Protection Sector 4 0x0801 6 K No Protection Sector 5 0x0802 12 No Protection Sector 6 0x0804 12 No Protection Sector 7 0x0806 12 No Protection Sector 9 0x0808 12 No Protection Sector 9 0x0808 12 No Protection Sector 10 0x0806 12 No Protection Sector 10 0x0806 12 No Protection Sector 11 0x0807 12 No Protection Sector 11 0x0807 12 No Protection Sector 11 0x0806 12 No Protection Sector 11 0x0807 12 No Protection Sector 11 <	Flash protection mode:	Write protection ~				
Sector 0 0x0800 16 K No Protection Sector 1 0x0800 16 K No Protection Sector 2 0x0800 16 K No Protection Sector 3 0x0800 16 K No Protection Sector 4 0x0801 64 K No Protection Sector 5 0x0802 12 No Protection Sector 6 0x0804 12 No Protection Sector 7 0x0806 12 No Protection Sector 7 0x0808 12 No Protection Sector 9 0x0808 12 No Protection Sector 10 0x0808 12 No Protection Center 11 0x0807 12 No Protection Unselect all Select all Select all Select all	Sector Start ad Si	ize Protection				
Sector 1 0x0800 16 K No Protection Sector 2 0x0800 16 K No Protection Sector 3 0x0800 16 K No Protection Sector 4 0x0801 64 K No Protection Sector 5 0x0802 12 No Protection Sector 6 0x0804 12 No Protection Sector 7 0x0806 12 No Protection Sector 8 0x0808 12 No Protection Sector 9 0x0808 12 No Protection Sector 10 0x0806 12 No Protection Sector 10 0x0806 12 No Protection Sector 11 0x0007 13 No Protection Vonselect all Select all Select all	□ Sector 0 0×0800 16	6 K No Protection				
Sector 2 0x0800 16 K No Protection Sector 3 0x0800 16 K No Protection Sector 4 0x0801 64 K No Protection Sector 5 0x0802 12 No Protection Sector 6 0x0804 12 No Protection Sector 7 0x0806 12 No Protection Sector 8 0x0808 12 No Protection Sector 9 0x0808 12 No Protection Sector 10 0x0806 12 No Protection Unselect all Select all Select all	Sector 1 0×0800 16	6 K No Protection				
Sector 3 0x0800 16 K No Protection Sector 4 0x0801 64 K No Protection Sector 5 0x0802 12 No Protection Sector 6 0x0804 12 No Protection Sector 7 0x0806 12 No Protection Sector 8 0x0808 12 No Protection Sector 9 0x0808 12 No Protection Sector 10 0x0806 12 No Protection Unselect all Select all Select all	Sector 2 0×0800 16	6 K No Protection				
Sector 4 0x0801 64 K No Protection Sector 5 0x0802 12 No Protection Sector 6 0x0804 12 No Protection Sector 7 0x0806 12 No Protection Sector 8 0x0808 12 No Protection Sector 9 0x0808 12 No Protection Sector 9 0x0808 12 No Protection Sector 10 0x0802 12 No Protection Unselect all Select all	Sector 3 0×0800 16	6 K No Protection				
Sector 5 0x0802 12 No Protection Sector 6 0x0804 12 No Protection Sector 7 0x0806 12 No Protection Sector 8 0x0808 12 No Protection Sector 9 0x0808 12 No Protection Sector 9 0x0808 12 No Protection Sector 10 0x080C 12 No Protection Unselect all Select all Select all	Sector 4 0x0801 64	4 K No Protection				
Sector 6 0x0804 12 No Protection Sector 7 0x0806 12 No Protection Sector 8 0x0808 12 No Protection Sector 9 0x0804 12 No Protection Sector 10 0x0806 12 No Protection Unselect all Select all Select all	Sector 5 0×0802 12	2 No Protection				
Sector 7 0x0806 12 No Protection Sector 8 0x0808 12 No Protection Sector 9 0x0804 12 No Protection Sector 10 0x0806 12 No Protection Unselect all Select all Select all	Sector 6 0×0804 12	2 No Protection				
Sector 8 0x0808 12 No Protection Sector 9 0x080A 12 No Protection Sector 10 0x080C 12 No Protection Unselect all Select all Select all	Sector 7 0x0806 12	2 No Protection				
Unselect all Select all Apply Cancel	□ Sector 8 0×0808 12	2 No Protection				
Unselect all Select all Apply Cancel	□ Sector 9 0×080A 12	Z No Protection				
Unselect all Select all Apply Cancel	□ Sector 10 0x080C 12	Z No Protection				
Apply Cancel	Unselect all Select a	11				
		Apply Cancel				

Рисунок 24

3.3.4 В окне программы зайдите в меню «Target > Connect», затем «Target > Program…». Начнется загрузка программного обеспечения.

3.3.5 По окончании загрузки программного обеспечения отключите программатор от USB порта ПК, отключите программатор от разъёма XP3 платы.

4 ЗАВЕРШЕНИЕ ЗАГРУЗКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

После загрузки встроенного программного обеспечения в микроконтроллеры ATMega128-16AU и STM32F429VGT выполните следующие действия:

1) вставьте платы обратно в корпус устройства;

2) снова затяните винты, обведенные красным кружком на рисунке 2.1;

3) вновь подключите внешние кабели к разъемам Х1 - Х4 усилителя;



www.git-holding.ru Страница 17 / 18

ВСТРОЕННОЕ ПРОГРАММНОЕ R 16 DCI 01 M ОБЕСПЕЧЕНИЕ УЗЛА ПЕЧАТНОГО ССТТ ИНДИСТРИАЛЬНЫХ R DCI M

4) проверьте работоспособность усилителя R 500 DVE 13 М в составе системы многофункциональной промышленной связи «GIT-Comm» в соответствии с его Руководством по эксплуатации.