

toweroff's SIMM Flash Progger v3/

User manual v3.2

©toweroff, 2011

Introduction

The programmer is designed for reading, erasing, programming, memory chip 32-bit standard modules SIMM80;

Read, erase, programming chip AT89S52, AT89S53, AT89S8252, AT89S8253 and other incircuit via the SPI interface (ISP - In System Programming);

Reading and programming of the memory chips via the I2C interface

Technical characteristics

- 1. Dimensions 172x77x25mm
- 2. USB communication interface
- 3. No external power supply
- 4. Automatic detection algorithm of the module type and write depending on the chipset used in the module
- 5. Verification of the contents of the modules with a base configuration file for the three hashes MD5, SHA1, AGI CRC, possibly independent add / edit / remove hashes
- 6. ISP programming from 4 to 20 seconds, depending on the type of chip
- 7. Three-color indication of the programmer with the ability to change for each mode of operation

Package included

- 1. Programmer
- 2. I2C interface adapter
- 3. USB cable
- 4. Software and drivers

Software

1. Installation

Run the file setup_FP1_v3.2.exe. Select the installation folder (in the future - [INSLALL_DIR])

Connect the programmer. The system detects the new device. Specify the location of the driver - [INSTALL_DIR] \ Driver

2. Programmer usage

After starting the program opens a window (all figures are approximate and depend on the configuration INI file)

🖗 Flash Programmator (One flash) v3.2	
Flash module programming I2C programming SPI programming	
C Read FLASH @ Auto MD5	Start
Erase FLASH C Manual SHA-1	E <u>x</u> it
Write FLASH AGI CRC	
Verity Byte-to-Byte Check	
	Corporte file HASH
[19:38:07] MDŚ main hash, SHA1 check, AGÍ CRC check	
✓ Word Wrap	Save Log

Flash Module Programming tab

Read Flash

Choosing this item opens a dialog box where you specify where the content of flash module will be stored:



Erase Flash

Selecting this item includes a list of tasks erasing chip SIMM module

Write Flash

Choosing this item opens a dialog box where you specify the contents of the file will be written to the flash module



Verify

After selecting the **Write Flash**, opens the possibility of selecting the **Verify**. If this mode is enabled, after recording will be made byte-to-byte checking contents of the module with the firmware file.

🖗 Flash Programmator (One flash) v3.2		
Flash module programming I2C programming SPI p	programming	,
Read FLASH (Auto MDS		Start
Erase FLASH C Manual SHA-1		Exit
Write FLASH		
Verify		
🔲 Byte-to-Byte Check		
		Compute file HASH
[19:57:22] Program started. Module types 5, vendors [19:57:22] MD5 main hash, SHA1 check, AGI CRC che [20:01:40] Write job deselected [20:01:44] Selected file for write: C:\Work\Gaminator\ [20:01:46] Write verify job selected	: 5, chips 16, dumps hash's 657 in database. :ck \HotSpot\HS3_5425_94_1	
🔽 Word Wrap		Save Log

Byte-to-byte check

Choosing this item opens a dialog box where you specify which file will be done byte-to-byte checking with the contents of the module

Ø	Flash Programm	nator (One flash) v3.2	×
Fla	ash module programm	ning I2C programming SPI programming	
Г	Select file for b	oyte-to-byte check 🔹 💽 🔀	
Γ	Папка:	🔁 HotSpot 💽 🔶 💼 🕈 📰 🗸	
	Недавние	HS3 HotSpot3_5.4-25_94.rar	
V	документы	B H53_5425_94_1 B H53_5425_94_2	
[20 [20	Рабочий стол		ש
	Мои документы		
	П Мой компьютер		
			
	Сетевое окружение	Имя файла: Открыть	
		Тип файлов: Отмена	
V	Word Wrap	Save Log	

Compute HASH

After clicking on this button opens a dialog box where you specify a file for which checksum **MD5**, **SHA1** and **AGI CRC**.

Attention! To execute the transaction programmer must be connected to the USB port!

Ø	Flash Programm	ator (One flash) v3.2	2					×
Fla	ash module programm	ing I2C programming SI	PI programming					
	Select file for c	ompute HASH					? 🛛	
	Папка:	🗀 HotSpot		•	+ 🗈 💣	•		
	Недавние документы Рабочий стол Мои документы	HS3 HotSpot3_5.4-25_94.r hotspot4_94_5.6-5.ra HS3_5425_94_1 HS3_5425_94_2	ar r]
	Сетевое Сетевое окружение	Имя файла: Тип файлов:			•		Открыть Отмена	
	Word Wrap						Save Log	

After reading the module, or checksumming a file is checked with a database of checksums stored in the INI file. If no match is found, you can make an entry in the database, simply by specifying the name

ኞ Flash Programmato	or (One flas	sh) v3.2		
Flash module programming	I2C program	nming SPI programming		
🔽 Read FLASH 📀 Au	ito MD5	;		Start
Erase FLASH C Ma	anual SHA-1			E <u>x</u> it
Write FLASH	AGI CRC	: [
Verify				
🔲 Byte-to-Byte Check				
	New Dump			Compute file HASH
[20:53:46] Program started [20:53:50] Selected file for Flash module: 64M (32Mx2) ST, M29W128GH Chip buffer size: 64 bytes [20:53:56] Start jobs [20:53:56] Start read to C:\ [20:55:26] Read done, 0:01 Defailt hash (MD5) check da	MD5 SHA-1 CRC AGI Name Add to	8317cb794bd8dda60669eabb3840eff7 583ae9d13f4ae13fc81b3fa81232e5be40baf3b5 84f747aa EMPTY MODULE database Cancel	ie,	
🔽 Word Wrap				Save Log

These checksums match the empty 64-megabyte module.

🖗 Flash Programmator (One flash) v3.2	
Flash module programming I2C programming SPI programming	
Read FLASH (Auto MD5	Start
Erase FLASH C Manual SHA-1	E <u>x</u> it
Write FLASH AGI CRC	
Verry Ryte-to-Byte Check	
) Byte-to-byte Check	
	Compute file HASH
[20:53:46] MDS main hash, SHA1 check, AGI CRC check [20:53:50] Selected file for read: C:\Work\Gaminator\gfd Flash module: 64M (32Mx2) ST, M29W128GH Chip buffer size: 64 bytes [20:53:56] Start jobs [20:53:56] Start read to C:\Work\Gaminator\gfd [20:55:26] Read done, 0:01:29 Defailt hash (MD5) check database FAIL [20:59:36] Record added to INI file, parameter name: Auto1	
Vord Wrap	Save Log

After adding the new entry is created in section **[HASH]** INI file name parameter is automatically determined (in this case - **Auto1**)

After selecting the required jobs, click on "Start" button

Jobs are executed sequentially, one after the other

🖗 Flash Programmator (One flash) v3.2		
Flash module programming I2C programming SPI programming		
Read FLASH I Auto MD5	Cancel	
Erase FLASH C Manual SHA-1	E <u>x</u> it	
Write FLASH AGI CRC		
Verify		
V Byte-to-Byte Check		
	Compute file HA	4SH
[21:04:39] Selected file for read: C:\Work\Gaminator\gfd [21:04:39] Erase job selected [21:04:49] Selected file for write: C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-10_2 [21:04:50] Write verify job selected [21:04:53] Selected file for byte-to-byte check: C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-1 Flash module: 64M (32Mx2) ST, M29W128GH Chip buffer size: 64 bytes [21:04:59] Start jobs [21:05:00] Start read to C:\Work\Gaminator\gfd	0_2	
V Word Wrap	Save Log	
Reading [Remaining: 0:01:07]		19%

Flash Programmator (One flash) v3.2	
Flash module programming I2C programming SPI programming	
Read FLASH & Auto MDS	Cancel
Erase FLASH C Manual SHA-1	Exit
Write FLASH	
Verify	
🔽 Byte-to-Byte Check	
	Compute file HASH
[21:04:39] Selected file for read: C:\Work\Gaminator\gfd [21:04:39] Erase job selected [21:04:39] Erase job selected [21:04:50] Write verify job selected [21:04:50] Write verify job selected [21:04:53] Selected file for byte-to-byte check: C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-1 Flash module: 64M (32Mx2) ST, M29W128GH Chip buffer size: 64 bytes [21:06:24] Read to C:\Work\Gaminator\gfd [21:06:24] Read done, 0:01:24 Defailt hash (MD5) match: HotSpotII3 V5.4-25 94% #1 SHA1 check OK AGI CRC check OK [21:06:24] Start erasing flash	0_2
V Word Wrap	Save Log
Erasing [Bank0: Erasing, Bank1: Erasing]	

🖗 Flash Programmator (One flash) v3.2	
Flash module programming I2C programming SPI programming	
Read FLASH @ Auto MD5	Cancel
Erase FLASH O Manual SHA-1	E <u>x</u> it
Write FLASH AGI CRC	
Verify	
🔽 Byte-to-Byte Check	
	Compute file HASH
[21:04:39] Erase job selected [21:04:39] Erase job selected [21:04:50] Write verify job selected [21:04:50] Write verify job selected [21:04:50] Write verify job selected [21:04:53] Selected file for byte-to-byte check: C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-1 Flash module: 64M (32Mx2) 5T, M29W128GH Chip buffer size: 64 bytes [21:04:59] Start jobs [21:05:00] Start read to C:\Work\Gaminator\gfd [21:06:24] Read done, 0:01:24 Defailt hash (MDS) match: HotSpotII3 V5.4-25 94% #1 SHA1 check OK [21:06:24] Start erasing flash [21:07:03] Flash erase done, 0:00:38 [21:07:03] Start write from C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-10_2	10_2
Vord Wrap	Save Log
Writing [Remaining: 0:00:30]	56%

Elash Programmator (One flash) v3-2	
Last module programming 12C programming 5P1 programming	F 1
Read FLASH @ Auto MD5	Lancel
Erase FLASH C Manual SHA-1	E⊻it
Write FLASH AGI CRC	
Verify	
V Byte-to-Byte Check	
	Compute file HASH
[21:04:33] MD5 main hash, SHA1 check, AGI CRC check	^
[21:04:39] Selected file for read: C:\Work\Gaminator\gfd [21:04:39] Erase job selected	
[21:04:49] Selected file for write: C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-10_2 [21:04:50] Write verify job selected	
[21:04:53] Selected file for byte-to-byte check: C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-1	0_2
Flash module: 64M (32MX2) ST, M29W128GH	
Chip buffer size: 64 bytes [21:04:59] Start jobs	
[21:05:00] Start read to C:\Work\Gaminator\gfd [21:06:24] Read done, 0:01:24	
Defailt hash (MD5) match: HotSpotII3 V5.4-25 94% #1	
AGI CRC check OK	
[21:06:24] Start erasing Hash [21:07:03] Flash erase done, 0:00:38	
[21:07:03] Start write from C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-10_2 [21:08:14] Write done. 0:01:10	
[21:08:15] Start verifying with C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-10_2	
	<u></u>
Vord Wrap	Save Log
Checking [Remaining: 0:00:43]	479
Flash Programmator (One flash) v3.2	
Flash Programmator (One flash) v3.2	
Flash Programmator (One flash) v3.2 Flash module programming I2C programming SPI programming Read FLASH C Auto MD5	Cancel
Flash Programmator (One flash) v3.2 Flash module programming I2C programming SPI programming Read FLASH Auto MD5 Erase FLASH Manual SHA-1	Cancel
Flash Programmator (One flash) v3.2 Flash module programming I2C programming SPI programming Read FLASH Auto MD5 Erase FLASH Manual SHA-1 Write FLASH AGI CRC	Cancel Exit
Flash Programmator (One flash) v3.2 Flash module programming I2C programming SPI programming Read FLASH Auto Manual SHA-1 Write FLASH AGI CRC Verify	Cancel
Flash Programmator (One flash) v3.2 Flash module programming I2C programming SPI programming Read FLASH Auto Manual SHA-1 Write FLASH AGI CRC Verify Byte-to-Byte Check	Cancel Exit
Flash Programmator (One flash) v3.2 Flash module programming I2C programming SPI programming Read FLASH I2C mogramming SPI programming Read FLASH Auto MD5 Erase FLASH Manual SHA-1 Write FLASH Manual SHA-1 Verify Verify Byte-to-Byte Check Image: State St	Cancel Exit
Flash Programmator (One flash) v3.2 Flash module programming I2C programming SPI programming Read FLASH Auto MD5 Erase FLASH Manual SHA-1 Write FLASH AGI CRC Verify Byte-to-Byte Check 21:04:49] Selected file for write: C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-10_2	Cancel Exit
Flash Programmator (One flash) v3.2 Flash module programming I2C programming SPI programming Prase FLASH Auto Manual SHA-1 Write FLASH Manual SHA-1 Write FLASH Manual SHA-1 Write FLASH Manual SHA-1 Write FLASH Manual SHA-1 Write FLASH SHA-1 Write FLASH SHA-1 Write FLASH SHA-1 SHA-1 SHA-1 SHA-1 SHA-1 SHA-1 SHA-1 Write FLASH SHA-1 SHA-1 <	Cancel Exit Compute file HASH
 Flash Programmator (One flash) v3.2 Flash module programming I2C programming SPI programming Read FLASH Auto MD5 Erase FLASH Manual SHA-1 Write FLASH AGI CRC Verify Byte-to-Byte Check 21:04:49] Selected file for write: C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-10_2 21:04:50] Write verify job selected 21:04:53] Selected file for byte-to-byte check: C:\Work\Gaminator\G18_94_5-5-10\18_04_5-5-10\18_04_5-5-10\18_04_5-5-10\18_04_5-5-10\18_04_5-5-10\18_04_5-5-10\18_04_5-5-10\18_04_5-5-10\18_04_5-5-10\18_04_5-5-10\18_04_5-5-10\18_04_5-5-10\18_04_5-5-10\18_04_5-5-10\18_04_5-5-10\18_04_5-5-10\18_04_5-5-10\	Cancel Exit Compute file HASH
 Flash Programmator (One flash) v3.2 Flash module programming I2C programming SPI programming Read FLASH Auto MD5 Erase FLASH Manual SHA-1 Write FLASH AGI CRC Verify Byte-to-Byte Check 21:04:49] Selected file for write: C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-10_2 21:04:50] Write verify job selected 21:04:50] Write verify job selected 21:04:50] Write verify job selected 21:04:50] Selected file for byte-to-byte check: C:\Work\Gaminator\G18_94_5-5-10\18_94_	Cancel Exit
Flash Programmator (One flash) v3.2 Flash module programming I2C programming SPI programming I2C programming SPI programming Read FLASH Auto MD5 Erase FLASH Manual SHA-1 Write FLASH Manual SHA-1 Write FLASH AGI CRC Verify Byte-to-Byte Check 21:04:49] Selected file for write: C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-10_2 21:04:50] Write verify job selected 21:04:51] Selected file for byte-to-byte check: C:\Work\Gaminator\G18_94_5-5-10\18_94_	Cancel Exit
 Flash Programmator (One flash) v3.2 Flash module programming I2C programming SPI programming Read FLASH Auto MD5 Erase FLASH Auto MD5 Erase FLASH Auto MD5 Write FLASH AGI CRC Write FLASH AGI CRC Verify Byte-to-Byte Check 21:04:49] Selected file for write: C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-10_2 21:04:50] Write verify job selected 21:04:53] Selected file for byte-to-byte check: C:\Work\Gaminator\G18_94_5-5-10\18	Cancel Exit
Flash Programmator (One flash) v3.2 Flash module programming I2C programming SPI programming I2C programming SPI programming Read FLASH Auto MD5 Erase FLASH Manual SHA-1 Write FLASH Manual SHA-1 Write FLASH AGI CRC Verify Perify Byte-to-Byte Check 21:04:49] Selected file for write: C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-10_2 21:04:50] Write verify job selected 21:04:53] Selected file for byte-to-byte check: C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-	Cancel Exit Compute file HASH
 Flash Programmator (One flash) v3.2 Flash module programming I2C programming SPI programming Read FLASH Auto MD5 Erase FLASH Auto MD5 Write FLASH AGI CRC Verify Byte-to-Byte Check 21:04:49] Selected file for write: C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-10_2 21:04:50] Write verify job selected 21:04:53] Selected file for byte-to-byte check: C:\Work\Gaminator\G18_94_5-5-10\18_9	Cancel Exit Compute file HASH
Flash Programmator (One flash) v3.2 Flash module programming I2C programming SPI programming I2C programming SPI programming I2C programming SPI programming I2C programming SPI programming I2C programming SPI programming I2C programming SPI programming I2C programming SPI programming I2C programming SPI programming I2 Read FLASH Auto I2 Write FLASH AGI CRC I2 Verify Byte-to-Byte Check I21:04:49] Selected file for write: C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-10_2 21:04:53] Selected file for byte-to-byte check: C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10_2 I2:06:24] Read done, 0:01:24 Defailt hash (MD5) match: HotSpotII3 V5.4-25 94% #1 I3HA1 check OK XI:06:24] Start reasing flash I2:07:03] Start write from C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-10_2	Cancel Exit Compute file HASH
Flash Programmator (One flash) v3.2 Flash module programming I2C programming SPI programming Read FLASH Auto MDS Frase FLASH Manual SHA-1 Write FLASH Glack AGI CRC Verify Byte-to-Byte Check Selected file for write: C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-10_2 21:04:49] Selected file for write: C:\Work\Gaminator\G18_94_5-5-10\2 Verify Selected file for byte-to-byte check: C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\2 Verify Selected file for byte-to-byte check: C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-10\2 Start poly Selected file for byte-to-byte check: C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-10\2 Verify Selected file for byte-to-byte check: C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\2 Start poly Selected file for byte-to-byte check: C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-10_2 Start poly Selected file for byte-to-byte check: C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-10_2	Cancel Exit Compute file HASH
Flash Programmator (One flash) v3.2 Flash module programming IZC programming SPI programming Read FLASH Auto MD5 Erase FLASH C Manual SHA-1 Write FLASH C Manual SHA-1 Write FLASH AGI CRC Verify Byte-to-Byte Check Byte-to-Byte Check 21:04:9] Selected file for write: C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-10_2 21:04:50] Write verify job selected 21:04:53] Selected file for byte-to-byte check: C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10_2 21:06:24] Read done, 0:01:24 Defail hash (MD5) match: HotSpotII3 V5.4-25 94% #1 SHA1 check OK XAGI CRC check OK 21:06:24] Start erasing flash 21:07:03] Flash erase done, 0:00:38 21:07:03] Flash erase done, 0:00:38 21:07:03] Start write from C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-10_2 21:08:14] Write done, 0:01:10 21:08:15] Start verifying with C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-10_2 21:09:36] Verify done, 0:01:21 21:09:36] Verify done, 0:01:21	Cancel Exit Compute file HASH
Flash Programmator (One flash) v3.2 Flash module programming I2C programming SPI programming IZC programming SPI programming IZ:04:50] Write rently IZ:04:50] Write verify possion II:04:50] Write for byte-to-byte chec	Cancel Exit Compute file HASH
Flash Programmator (One flash) v3.2 Flash module programming I2C programming SPI programming Read FLASH Auto Manual SHA-1 Write FLASH Manual SHA-1 Write FLASH Manual Verify Byte-to-Byte Check 21:04:49] Selected file for write: C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-10_2 21:04:50] Write verify job selected 21:04:50] Selected file for byte-to-byte check: C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\18_94_5-5-10\21:06:24] Start read to C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-10_2 SHA1 check OK KaI CRC check OK 21:00:24] Start erasing flash 21:07:03] Flash erase done, 0:00:38 21:07:03] Flash erase done, 0:00:38 21:07:03] Flash erase done, 0:01:10 21:08:15] Start verify ing with C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-10_2 21:09:36] Verify done, 0:01:21 21:09:36] Verify done, 0:01:21 21:09:36] Start byte-to-byte check with C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-10_2	Cancel Exit Compute file HASH
Plash Programmator (One flash) v3.2 Flash module programming I2C programming SPI programming Plash module programming IEC programming IEE programming Plash module programming IEE programming IEE programming Plash module programming IEE programming IEE programming Plast programming IEE programming IEE programming IEE programming IEE programming IEE programming Plast programming IEE programming IEE programming IEE programmine IEE programmine <t< td=""><td>Cancel Exit Compute file HASH</td></t<>	Cancel Exit Compute file HASH

🖗 Flash Programmator (One flash) v3.2	
Flash module programming I2C programming SPI programming	
Read FLASH Auto MD5	Start
Erase FLASH C Manual SHA-1	E <u>x</u> it
Write FLASH AGI CRC	
Verify	
🔽 Byte-to-Byte Check	
	Compute file HASH
Flash module: 64M (32Mx2) ST, M29W128GH Chip buffer size: 64 bytes [21:04:59] Start jobs [21:05:00] Start read to C:\Work\Gaminator\gfd [21:06:24] Read done, 0:01:24 Defailt hash (MD5) match: HotSpotII3 V5.4-25 94% #1 SHA1 check OK AGI CRC check OK [21:06:24] Start erasing flash [21:07:03] Flash erase done, 0:00:38 [21:07:03] Start write from C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-10_2 [21:08:14] Write done, 0:01:10 [21:08:15] Start verifying with C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-10_2 [21:09:36] Verify done, 0:01:21 [21:09:36] Verify done, 0:01:21 [21:09:36] Start byte-to-byte check with C:\Work\Gaminator\G18_94_5-5-10\18_94_5-5-10\18_94_5-5-10_2 [21:10:58] Check done, 0:01:21 [21:10:58] Byte-to-byte check OK [21:10:58] Jobs successful done	
🔽 Word Wrap	Save Log

While doing the sounds reproduced, stored in a folder [INSTALL_DIR]\Sound\

alarm.wav - failure of jobs

ding.wav - transition to the next job

tada.wav - successful completion of jobs

You can save the log by pressing Save Log button

Supported formats: TXT, RTF



I2C programming tab

- Read read chip contents to the buffer
- Write write buffer contents to the chip
- **Open** open file to the buffer. Supported formats: BIN, Intel HEX
- Save save buffer contents to the file

🖗 Flash Programmat	or (One flash) v3.2	2			
Flash module programming	I2C programming S	PI progr	amming		
EEPROM type				1	
© 24C04	Read		Save		
C 24C256					
CLK frequency, kHz	Write		Open		
100		_			
100					
200					
400 started	. Module types 5, vend SNA1 shock ACT CPC -	ors 5, ch	iips 16, dumps hash's 657 in	database.	
[20:33:53] MD5 main nash,	SHAT CHECK, AGI CRU C	леск			
🔽 Word Wrap					Save Log

You can choose the frequency of operation (100, 200, 300 or 400 kHz)

SPI programming tab

MCU type - select of MCU type

Target - working area of the microcontroller

Write Lock Bits - protection of the MCU contents for reading

Verify - verify content after write

Auto - sequential execution of operations marked

🖗 Flash Programmator (One flash) v3.2							
Flash module programming I2C programming SPI programming							
MCU type	CODE	Γ	Erase chip				
C AT8958252 C AT89553	C EEPROM		Read	Save		E <u>x</u> it	
C AT8958253	Use chip buffer	Γ	Write	Open			
ISP crystal, Hz	🔽 Reset active HIGH	Γ	Verify				
11059200		Γ	Write Lock Bits			Auto	
[10000000 [11059200 15000000 20000000 22118400	n started. Module types 5, in hash, SHA1 check, AGI (venda IRC d	ors 5, chips 16, dun heck	nps hash's 657 in da	atabase.		
🔽 Word Wrap						Save Log	

Вы можете выбрать частоту кварцевого резонатора, используемого в схеме. Заметьте, что иногда понижение реальной частоты позволит получить более стабильное программирование

The configuration INI file

Section [Modules] Contains the modules description

[Modules]
module0 = ID,SIZE,TWO_BANKS
...

ID - the module ID in HEX SIZE - the module capacity {16|32|64} MBytes TWO_BANKS - two/one banks {1|0}

Section [Vendors]

Vendors description. Used for info only

```
[Vendors]
mnf0 = NAME,ID
```

```
•••
```

NAME - Vendor ID - vendor ID in HEX (ID0 from chip ID's)

Секция [Chips]

Chips description

```
[Chips]
chip0 = NAME,CHIP_ORG,WRITE_METHOD,USING_UBYPASS,ID0,ID1[,ID2,ID3]
...
```

NAME - chip name CHIP_ORG - chip bus width : 0 - 8bit 1 - 16bit 2 - 32bit WRITE_METHOD 0 - word by word 1 - use buffer

- 2 use extended buffer
- 3 use buffer size from CFI

http://toweroff.ru

USING_UBYPASS - chip can use short commands

0 - not using

1 - can used

IDx - chip ID's (started from Vendor ID). Number of bytes (2 <= IDs <= 4)

Section [Hash]

Hashes of dumps database

```
[Hash]
dump = NAME,HASH_MD5,HASH_SHA1,AGI_CRC
...
```

NAME - dump name HASH_MD5 - 32 chars in HEX of MD5 HASH_SHA1 - 40 chars in HEX of SHA1 AGI_CRC - 8 chars in HEX of AGI CRC

The name value (in this case, dump) program is not processed, you can use any name

Section [Setup]

Contains settings check hashes in the database

```
[Setup]
MD5_check = M
SHA1_check = 1
AGI_CRC_check = 1
```

Parameter values can be {M|1|0}

One of the parameters must always be M (Main)

The validation method is that: The database hash sums sought the amount shown as Main. If successful, other types of search are checked against the hash referred to as "1". The hash labeled as "0" is not checked

Section [LED]

Contains a description of the color indication modes

```
[LED]
parameter = time
mode = COLOR,MODE[,COUNT]
...
```

Group frequency settings of flashes

Parameter	time (default) ⁽¹⁾	comment
ShortInterval	10	Time glow and pause mode fast flashing
		(Short)
LongInterval	70	Time glow and pause mode slow
		flashing (Long)
CounterActiveTime	20	Time glow in counting mode (Count)
CounterPassiveTime	20	Time pause in counting mode (Count)
CounterInterval	100	The time intervals between groups of
		flashes in the counting mode (Count)

⁽¹⁾ Real time is calculated as (time * 10) ms

Group of settings for each mode of the glow

COLOR - Green (G), Red (R) or Yellow (Y)

MODE - (L) - slow flashing, (S) - short flashing, (C) - defined number of flashes

COUNT - number of flashes in mode "C"

Default values

Mode	Value
StartReadFlash	G,C,1
FaultReadFlash	R,C,1
StartEraseFlash	G,C,2
FaultEraseFlash	R,C,2
StartWriteFlash	G,C,3
FaultWriteFlash	R,C,3
StartCheckFlash	G,C,4
FaultCheckFlash	R,C,4
StartBTBCheckFlash	G,C,5
FaultBTBCheckFlash	R,C,5
StartReadI2C	Y,S

http://toweroff.ru

FaultReadI2C	R,S
StartWriteI2C	Y,S
FaultWritel2C	R,S
StartReadSPI	Y,S
FaultReadSPI	R,S
StartEraseSPI	Y,S
FaultEraseSPI	R,S
StartWriteSPI	Y,S
FaultWriteSPI	R,S
StartWriteLockbitsSPI	Y,S
FaultWriteLockbitsSPI	R,S

You can not add all the values in section **[LED]**, is sufficient to add only those parameters that specify the new values

<u>Contact us</u>

In all matters pertaining to the operation of hardware and software **toweroff's**, you can contact:

- on the site, using the feedback form <u>http://toweroff.ru/index.php?page=feedback_en</u>
- on forum http://toweroff.ru/forum/index.php

 To access certain sections need to register
 At the forum you can always download the latest database of checksums of firmware,
 lists of supported chips and ask questions
- email <u>toweroff@mail.ru</u>