MSP430 Flasher

Documentation

About MSP430 Flasher

MSP430 Flasher is a user-friendly, shell-based interface providing the user with easy access to MSP430 devices through a FET via JTAG or SBW.

It provides the user with the freedom to:

- choose:
  - which port of the computer the FET is connected to
  - which mode to use for accessing the device
  - which device to access
  - which erase method to run on the memory of the device
  - whether to verify the memory of the device
- load a TXT or HEX file into the device
- read memory out and write into a TXT or HEX file
- set hardware breakpoints
- power up the device
- provide a JTAG password (if applicable)
- select the mode of operation for multi-mode devices
- update the firmware of the device
- disable status reports (in Quiet Mode)
- blow the device's JTAG fuse

While keeping the user informed via essential status reports, *MSP430 Flasher* performs all the necessary tasks for accessing a MSP430, including but not limited to:

- Initializing the interface port
- Powering up and down the device
- Checking the firmware and hardware versions
- Writing the TXT or HEX file of choice to the device, including loading the file into external memory for devices which require this.

For convenience status reports are logged into a text file called *log.txt* in the subdirectory Log, which is in line with the *MSP430 Flasher* executor.
NOTE: *New instances are appended to the file so old logs are never overwritten.*

CONTENTS

1. USING MSP430 Flasher

1.1 Introduction

MSP430 Flasher runs from an executable file called *MSP430Flasher.exe*. This file accepts a number of triggers and arguments necessary to give the user access to the full capabilities of the software.

Help information can be accessed by using triggers -H, -h or -?.
Valid *Exit Specifications* (see trigger –z) can be viewed by using trigger -x

1.2 Table of Triggers and Arguments

Trigger	Arguments	Description	Further Info.
-n	Device Name	The name of the device being accessed.	Prompt if missing.
-o	L	Operating mode for L092 device.	L - L092 mode (with external EPROM) C - C092 mode
	C		
-l	Password Length (in words)	For double-checking the entered JTAG password.	Optional. Default: 0
-p	JTAG Password	The password of the device being accessed.	Prompt if incompatible with password length.
-i	(TI)USB	Communication port	Default: TIUSB (=USB)
	LPTn		
	COMn		
-m	JTAG	Communication mode to be used.	Default: AUTO
	SBW2		
	SBW4		
	AUTO		
-r	[Filename,mem_sec]	Memory section to read and filename to write to.	mem_sec can be: MAIN, INFO, RAM or BSL
-w	Filename	TXT or HEX file to be loaded.	Optional.
-b	N/A	Unlock BSL memory for writing	Use only with –w switch
-u	N/A	Unlock locked flash memory (InfoA) for writing	Use only with –w switch
-e	ERASE_ALL	Erase mode before programming.	Use only with –w switch. Default: ERASE_ALL.
	ERASE_MAIN		
	ERASE_SEGMENT		

-v	N/A	Trigger verification after programming.	Use only with –w switch.
-z	[exit_spec,…]	Specifies state of device on exit	
-g	N/A	Disables log.	Default: enabled.
-d	[breakpoint addresses]	Sets hardware breakpoints	Use “,” as delimiter
-t	Timeout_in_ms	Specifies breakpoint timeout in milliseconds	
-f	N/A	Blows the JTAG fuse	
-s	N/A	Suppresses FET firmware update	**Warning:** for error-free code execution FET fw version should always match the DLL version
-q	N/A	Triggers QUIET MODE	No system messages displayed.*
Omitted arguments will be replaced by the default options if possible or the user will be prompted to provide them later. *This excludes errors and prompts.			

Example: In Windows Command Processor, on prompt, the following is entered:

MSP430Flasher.exe -n MSP430C092 -l 4 -p 0x1234ABCD5678 -v -w myfile.txt -i USB

MSP430 Flasher will:

1) Attempt to access the FET connected to the USB port.
2) Decide which access mode to use and use it, since it has been omitted.
3) Attempt to access a device called MSP430C092.
4) Request a password of length 4 (words) from the user, since the password entered is only length 3.
5) Erase the whole memory of the device by default.
6) Program myfile.txt into the device.
7) Verify the content of the memory after the file is programmed

1.3 Table of Exit Specifications

MSP430 Flasher provides the user with the freedom to select a desired state for the device to be set to when *MSP430 Flasher* finishes its operation. This can be done using the trigger -z and passing the argument [exit_spec,…], where exit_spec is a valid exit specification (see table below).

NOTE: the specifications are delimited with the ‘,’ (comma) character and enclosed by square braces ‘[‘ ‘]’.

If the trigger is not called, *MSP430 Flasher* will assume default specifications, listed in the “Further Info.” column of the following table.

Exit Specifications	Description	Further Info.
VCC	Triggers VCC on after programming.	Default: VCC is off

NOTE:

Triggers, arguments and exit specifications are all case insensitive with the exception of *filename.*

1.4 Firmware Update

During runtime, in case *MSP430 Flasher* detects a conflict in the existing firmware versions it will prompt the user to choose whether to let *MSP430 Flasher* update the firmware or not:

>> The firmware of your FET is outdated.
>>Would you like to update it? (Y/N): _

Entering Y will update the firmware of the FET, displaying timely status reports, and on success will end the running instance.

In case of error the user will be prompted with:

>>Update failed. (R)etry/(C)ancel? _

Entering R will repeat the attempt to update while entering C will resume the running instance with the outdated firmware.

Entering N will resume the running instance with the outdated firmware.

1.5 Example Case: Loading and Executing a TXT file

Details:

- Device: MSP430F5436A
- Interface: USB
- Mode: SBW4
- Password: N/A
- File: file_123.txt
- Erase Type: ERASE_ALL
- Verification: TRUE
- VCC: ON

To load a TI .txt or Intel .a43 file, the user must first ensure that:

- The file to be loaded is in the same directory and level as the executable.

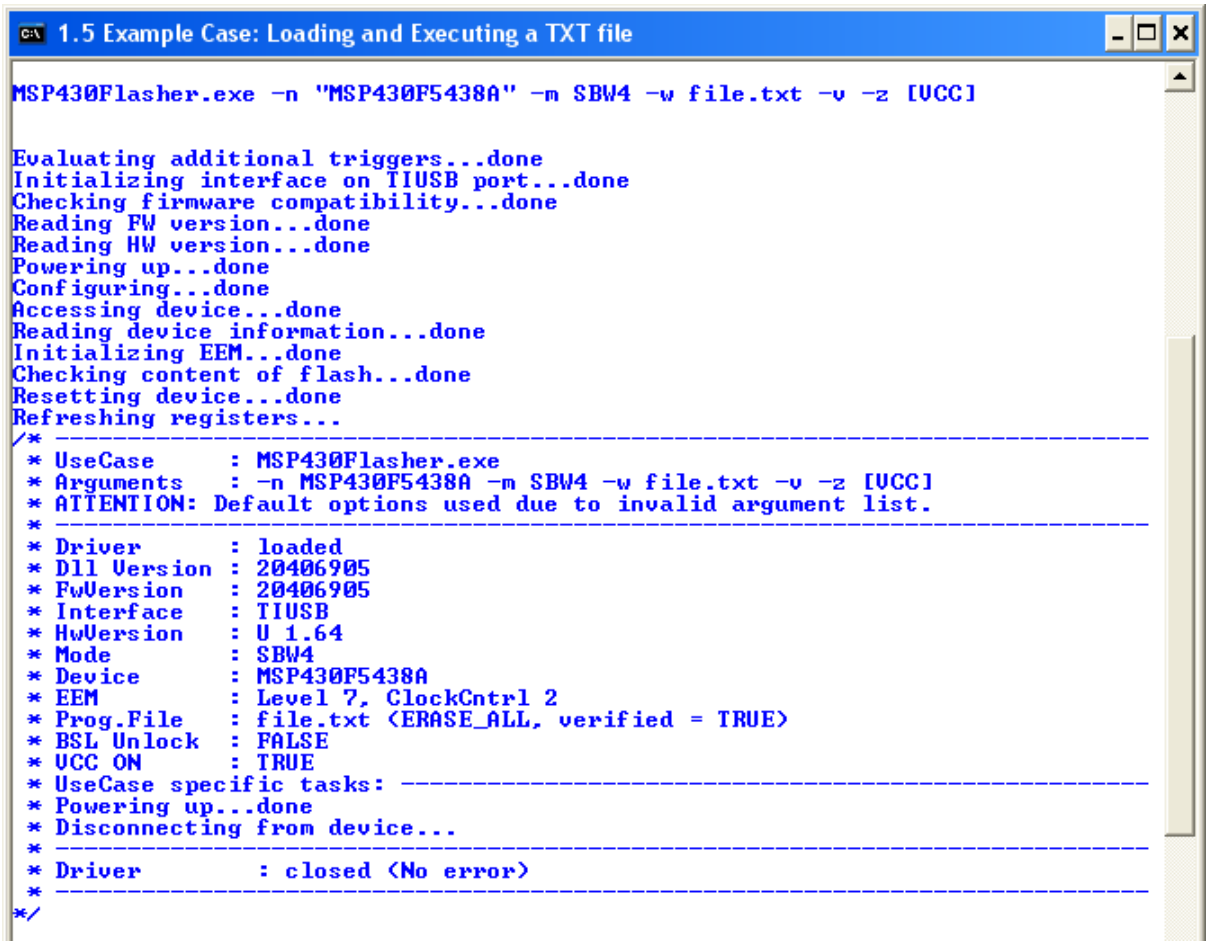
The line to use in this case would be:

MSP430Flasher.exe -n MSP430F5436A -m SBW4 -w file.txt -v -z [VCC] (-i USB)
(-e ERASE_ALL)

NOTE:

- Triggers -p and -l are not used because the device does not require a password.
- Triggers -i and -e may be used but are unnecessary since USB and ERASE_ALL are the default settings for these parameters respectively.

The following is what is obtained on entering the line above into Windows Command Prompt if the selected device is in fact connected via the desired COM port:


1.5 Example Case: Loading and Executing a TXT file
MSP430Flasher.exe -n "MSP430F5438A" -m SBW4 -w file.txt -v -z [VCC]
Evaluating additional triggers...done
Initializing interface on TIUSB port...done
Checking firmware compatibility...done
Reading FW version...done
Reading HW version...done
Powering up...done
Configuring...done
Accessing device...done
Reading device information...done
Initializing EEM...done
Checking content of flash...done
Resetting device...done
Refreshing registers...
/* ----------
* UseCase : MSP430Flasher.exe
* Arguments : -n MSP430F5438A -m SBW4 -w file.txt -v -z [VCC]
* ATTENTION: Default options used due to invalid argument list.
* ----------
* Driver : loaded
* Dll Version : 20406905
* FwVersion : 20406905
* Interface : TIUSB
* HwVersion : U 1.64
* Mode : SBW4
* Device : MSP430F5438A
* EEM : Level 7, ClockCntrl 2
* Prog.File : file.txt (ERASE_ALL, verified = TRUE)
* BSL Unlock : FALSE
* VCC ON : TRUE
* UseCase specific tasks: ----------
* Powering up...done
* Disconnecting from device...
* ----------
* Driver : closed (No error)
* ----------
*/

1.6 Example Case: Accessing Device with Device Activation Code

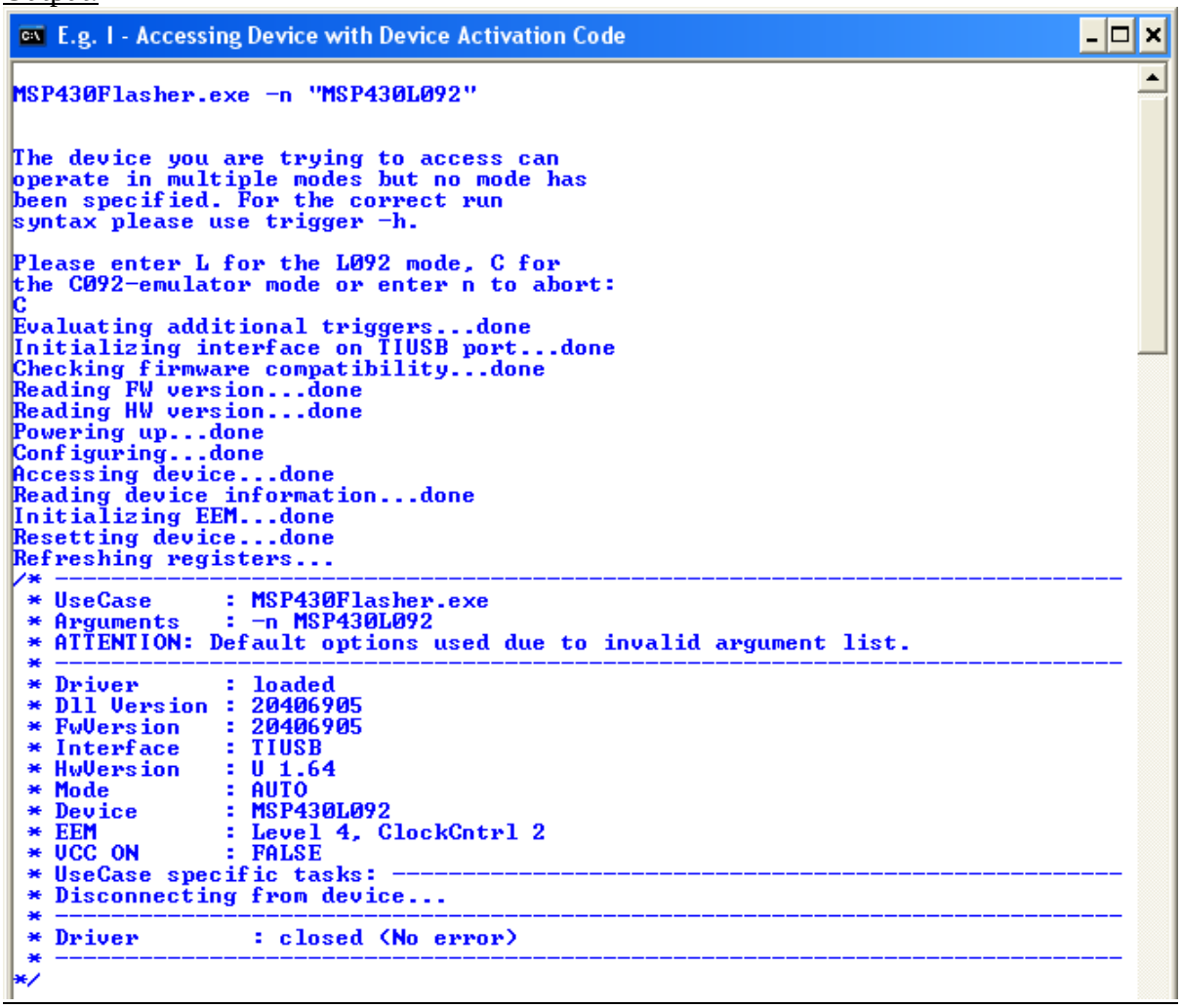
Some devices require a device activation code in order to be operable. Devices of this kind, such as the MSP430L092 will cause an error in *MSP430 Flasher* in case the provided activation code is incorrect or no activation code is provided.

MSP430 Flasher provides the necessary *Activation Code* internally but the user must specify the desired operating mode using the -o trigger. As can be seen from the *Table of Triggers and Arguments*, this switch takes the argument 'L' when the L092 operating mode (with external memory) is desired and 'C' when the C092 operating mode (without external memory) is desired.

E.g. I

>>MSP430Flasher.exe -n MSP430L092

Output:



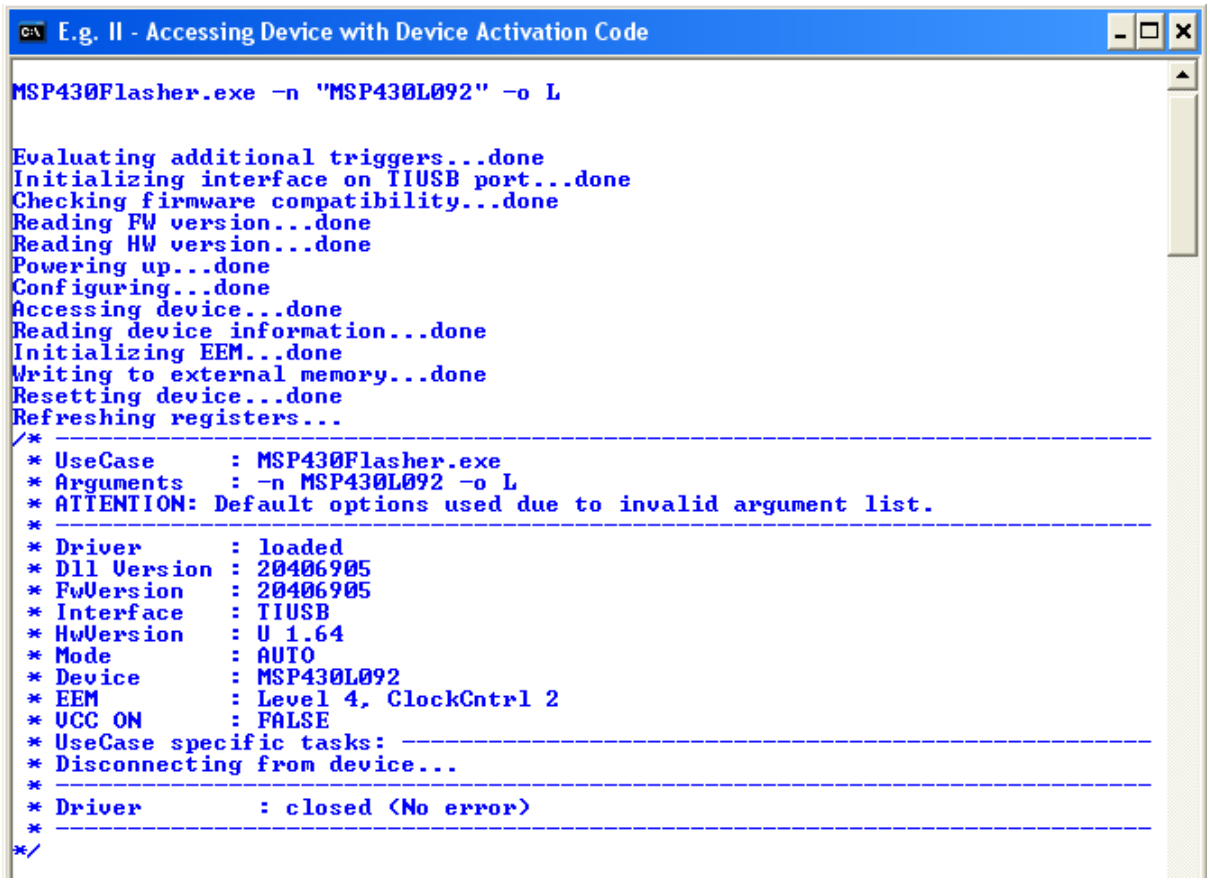
NOTE:

1) The user is prompted to select and operating mode when the device name is found to be MSP430L092 and no mode has been selected.
2) When C is entered, the external memory is not accessed.

E.g. II

>>MSP430Flasher.exe -n MSP430L092 -o L

Output:



NOTE:

1) The L092 mode was selected from the start so the user was not prompted for additional input.
2) Notice that *MSP430 Flasher* wrote to external memory: see system message "Writing to external memory…done"

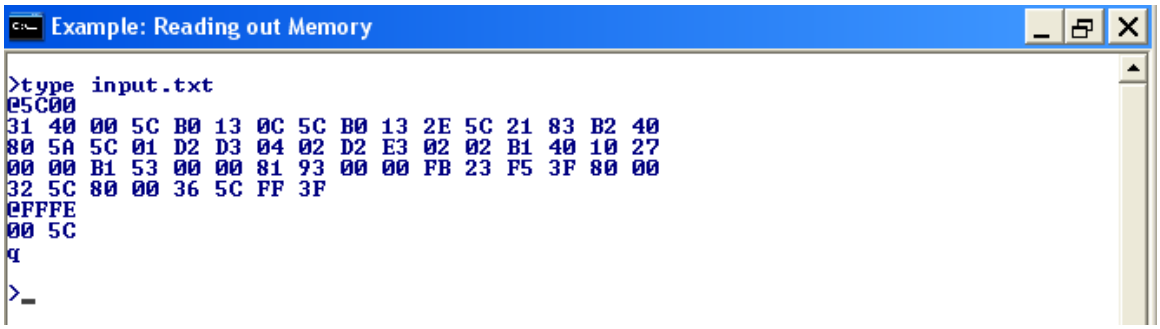
1.7 Example Case: Reading Memory out from Device and into File

MSP430 Flasher provides the user with the ability to read out any section of the device memory into a file of choice. The file can either be TI TXT or Intel HEX and the three memory sectors are MAIN, INFO, RAM and BSL.

In this example, using device MSP430F5438A, we write file input.txt to memory and then read out MAIN memory into file output.txt.

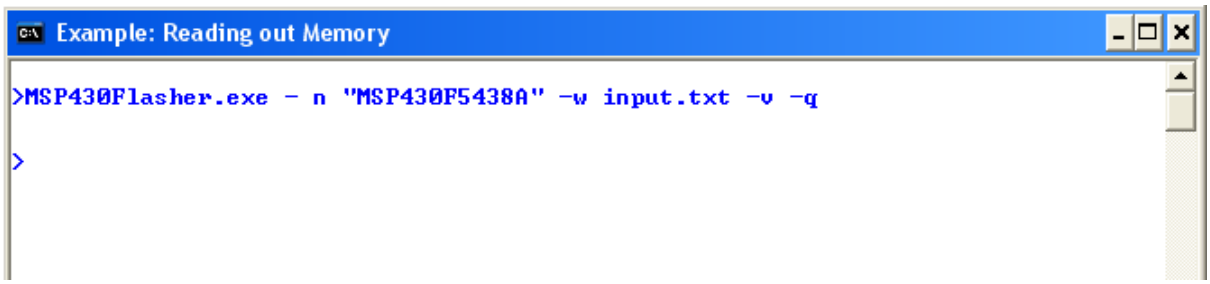
Please note that *MSP430 Flasher* will only read out memory up to a maximum length of 0x1FFFF.

1) Contents of input.txt:



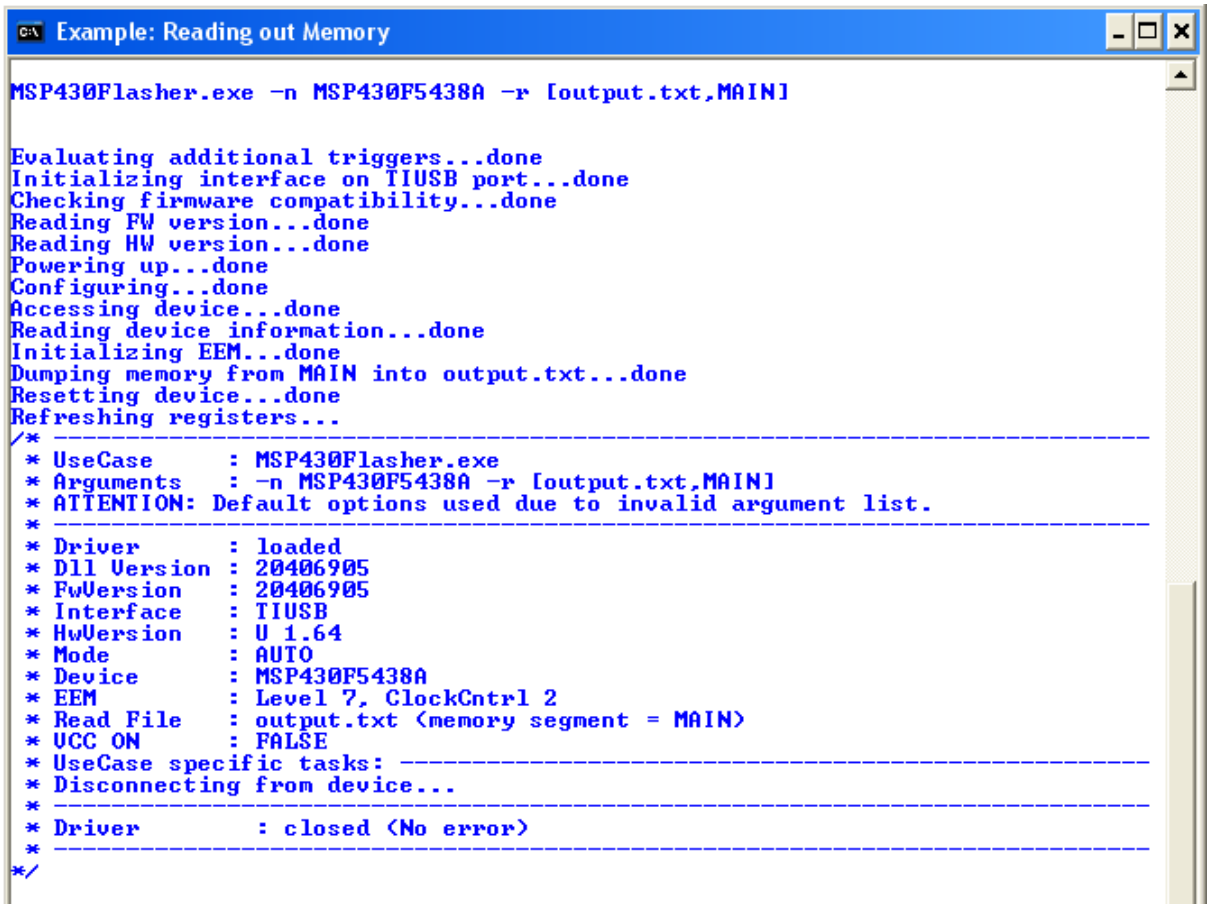
2) Writing to memory (this time using Quiet Mode):

>> MSP430Flasher.exe -n MSP430F5438A -w input.txt -v -q

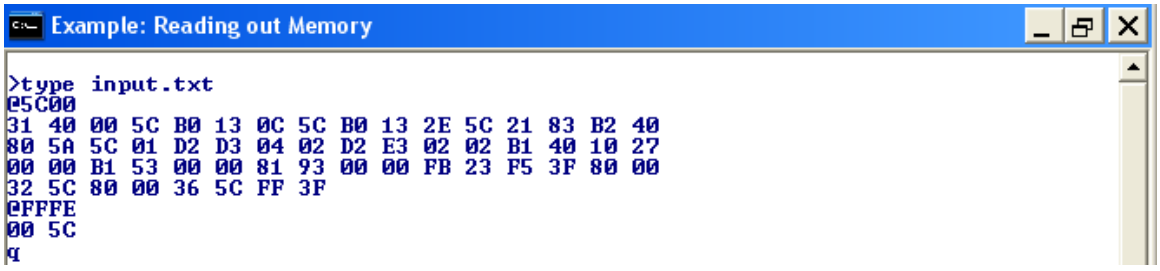


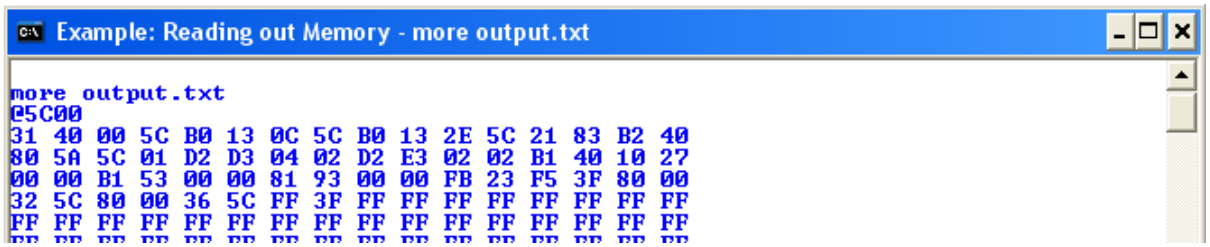
3) Reading from memory and writing to output.txt:

>> MSP430Flasher.exe -n MSP430F5521 -r [output.txt,MAIN]



4) Comparing input.txt and output.txt:





We can see that they are in fact the same.

1.8 Logging of Status Reports

MSP430 Flasher stores all of its display outputs and inputs into a log file called log.txt. This file is stored in the directory Log, which is in line with the executable. New instances are always appended and previous logs are never overwritten. Handling of past logs is left in the user's discretion.

To disable logging, simply append the -g trigger when running the executable.

1.9 Quiet Mode

MSP430 Flasher has a special mode that allows it to operate with minimal interference. In *Quiet Mode* status reports and system messages are not displayed and the user is only addressed in case of an error or an input prompt.

NOTE: All system messages are still stored in the log file for future reference.

1.10 Example Case: Setting Breakpoints and Executing a TXT

MSP430 Flasher offers the user the possibility of setting hardware breakpoints anywhere in the MAIN memory of the device. If the user entered breakpoints using the –d trigger, *MSP430 Flasher* will execute the program code, reading and displaying the CPU registers each time a breakpoint is hit. Once a breakpoint is hit, it will be cleared.

Details:

- Device: MSP430F5438A
- Interface: USB
- File: file.txt
- Erase Type: ERASE_ALL
- Verification: TRUE
- Breakpoints: [0x03F88,0x05c22,0x1F333]
- VCC: ON

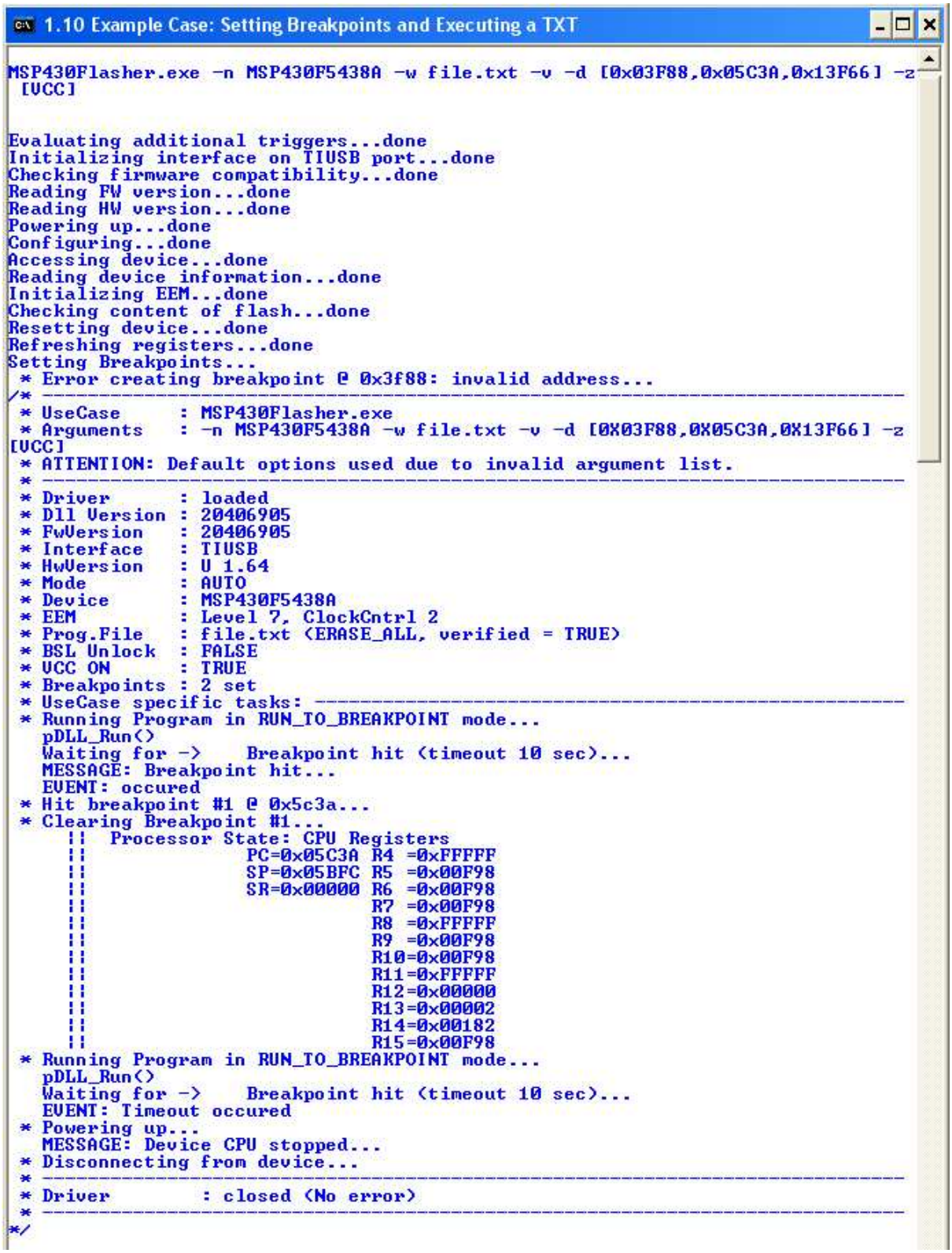
NOTE:

1) BP address #1 (0x03F88) is invalid, as it is not in the device's MAIN memory

2) BP address #3 (0x1F444) will not be reached as it is not placed in the program code

>>MSP430Flasher.exe –n MSP430F5438A –w file.txt –v –d [0x03F88,0x05c22,0x1F333] –z [VCC]

Output:



NOTE:

3) BP address #1 (0x03F88) generates an error message because of its invalid address.

4) The first valid breakpoint entered will be referred to as “Breakpoint #1”.

5) The breakpoints are numbered according to their input order.

6) Breakpoint #1 (0x05C3A) was hit and cleared. The CPU state was displayed.

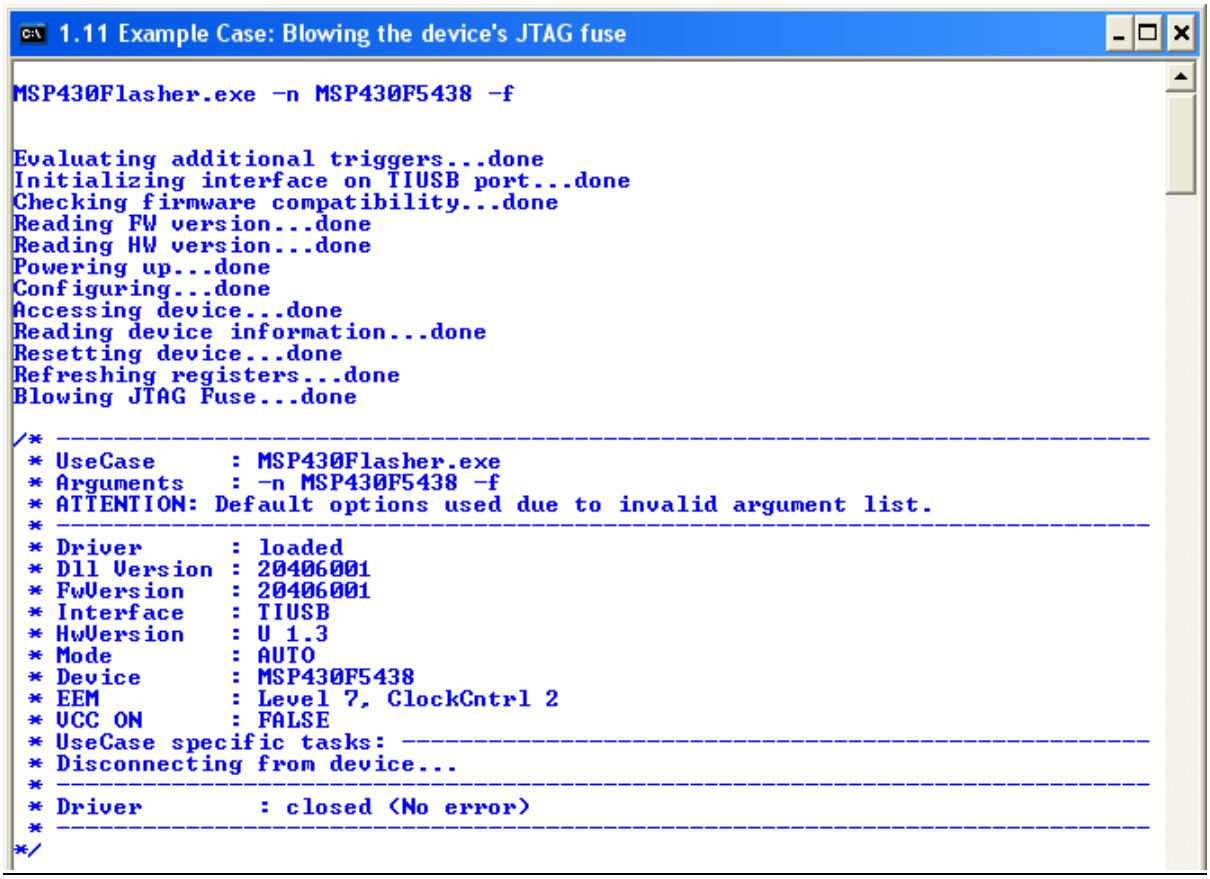
7) Breakpoint #2 (0x13F66) timed out. Despite its valid address, it cannot be reached during program execution.

1.11 Example Case: Blow the Device’s JTAG fuse

MSP430 Flasher lets the user choose whether the target device’s JTAG fuse should be blown. The fuse blow operation is activated by the –f trigger. Once the JTAG fuse is blown, the device is permanently inaccessible via JTAG.

>>MSP430Flasher.exe –n MSP430F5438 –f

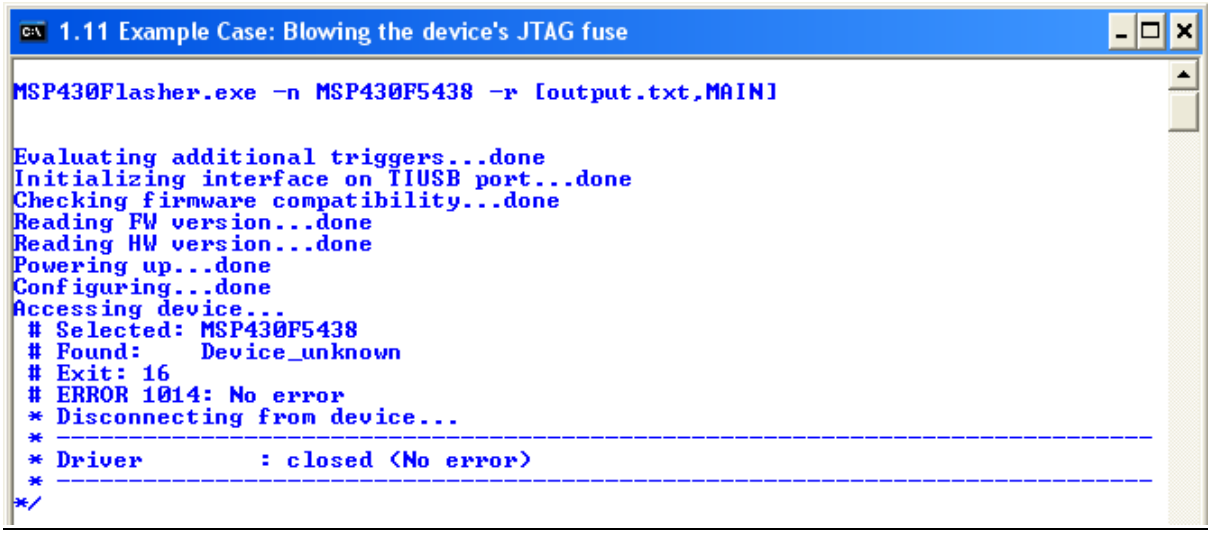
Output:



Trying to read the device's MAIN memory after the JTAG fuse has been blown:

>>MSP430Flasher.exe –n MSP430F5438 –r [out.txt, MAIN]

Output:



NOTE:

1) Breakpoint functionality is disabled when the –f switch is used

2) *MSP430 Flasher* can NOT blow the JTAG fuse of MSP430L092 devices