



User guider of PRGS430multifunctional-offline programmer

项目名称	430multifunctional-offline programmer	项目型号	LSD – PRGS430U-V2.0
编制人	陈冰	编制时间	2010.3.11
修改人	郝强、陈冰	修改时间	2010.6.24

I : overview

PRGS430 multifunctional-offline programmer is applied to MSP430 FLASH MCU of complete series, which can finish programming, blowing electronic fuse and BSL programming after fuse opening to MSP430 FLASH MCU of complete series. Before programming, codes need to be loaded to FLASH memory of programmer, then data stored in memory are extracted and switched into JTAG or UART (BSL) protocols by MCU of PRGS430. Codes can be loaded into target devices, so this programmer can work offline.

PRGS430 multifunctional-offline programmer is fit for programmer either small items during development phase or batch jobs in the course of production.

PRGS430 multifunctional-offline programmer supports three programming modes-JTAG, BSL and SBW.

JTAG programmatic interface is used to program and blow electronic fuse to devices without blowing fuse.

BSL programmatic interface can program to devices without consideration of fuse opening or not.

SBW can program all devices supporting SBW(F20xx,F41x2 and F5xx, slow load speed ,but fewer I/O occupied).

PRGS430 multifunctional-offline programmer supports reading codes from devices online, and password is needed in BSL mode.

Users can switch JTAG, BSL and SBW easily and conveniently, and only need to select mode required on host PC. Same operations are performed for specific function .

More specific functions will be added to meet requirements of different users utmostly.

LSD SCIENCE & TECHNOLOGY CO.,LTD has developed 2 PRGS430 programmers:LSD-PRGS430-IIIA and LSD-PRGS430U.

LSD – PRGS430U host PC software can support both of LSD-PRGS430-IIIA and LSD-PRGS430U programmer. LSD-PRGS430-IIIA uses Serial Port with 9 pins while LSD-PRGS430U adopts USB interface. LSD-PRGS430U can display some information, such as current state, program progress and serial-number, in offline mode on LCD self-owned while LSD-PRGS430-IIIA has not.

II : Hardware connection description

Hardware connection is shown as figure1. LSD – PRGS430U multifunctional-offline programmer can be connected to external via 3 interfaces.



Figure 1

Hardware connection is shown as figure2.LSD – PRGS430-IIIU multifunctional-offline programmer can be connected to external via 4 interfaces.

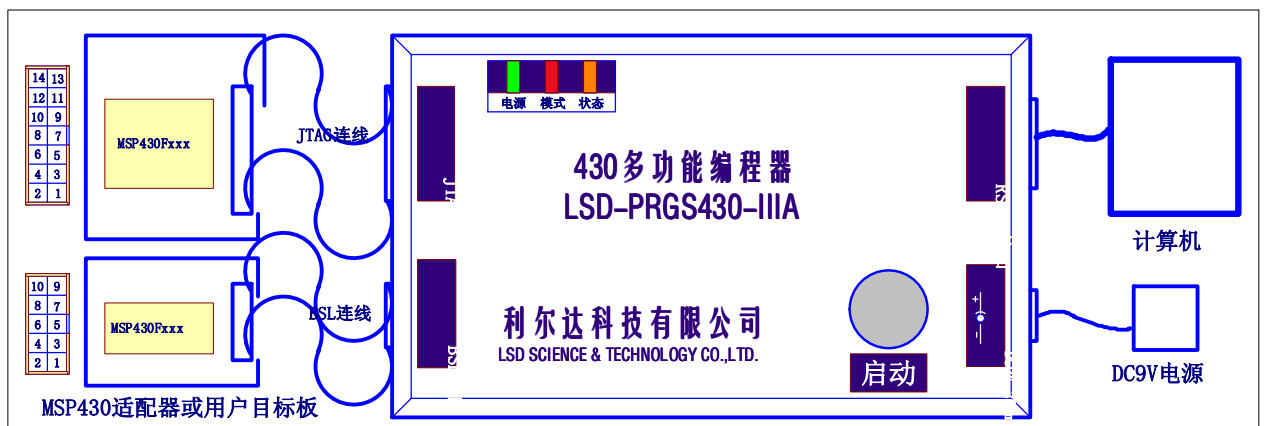


Figure 2

1) Power interface

Power supply with 9V DC/ 200mA.outside of power plug is negative,and inside is positive $+\ominus-$.Connect the plug into the interface as shown.The power may not be used when LSD-PRGS430U programmer works in online mode.

2) Communication interface

USB interface:

LSD – PRGS430U multifunctional-offline programmer has USB interface.Users can set ,update and modify codes using USB interface,and programmer will operate offline according to information from users.There is no need to use power supply of 9V DC/200mA if USB is connected in online mode because programmer can be powered by USB interface.

Serial Port with 9 pins:

LSD – PRGS430-IIIU multifunctional-offline programmer uses serial port with 9 pins. Difference from pro

grammer with USB interface is that it can only be powered by power adapter.

3)JTAG interface

LSD – PRGS430U multifunctional-offline programmer completes program writing and coding(blowning fuse) by JTAG interface. JTAG interface is defined as following:

pin	1	2	3	4	5	6	7	8	9	10	11	12	13	14
definition	TDO	VCC	TDI	VCC_IN	TMS	NC	TCK	TST	GND	NC	RST	NC	TXD*	RXD*

This pin is compatible with BSL interface pin. Pins, such as TXD,RXD*,TEST,RST,TCK,VCC and GND, required in BSL interface, are included in JTAG interface. PRGS programmer with USB interface has only JTAG interface, so a cable will be provided to transform 14pins -JTAG to 10pins-BSL.

4) SBW interface

SBW interface is defined as following:

pin	1	2	3	4	5	6	7	8	9	10	11	12	13	14
definition	SBWTDIO	VCC	NC	NC	NC	NC	SBWTCK	TST	GND	NC	NC	NC	NC	NC

TX and RX pins of BSL interface differ from device to device,Please see the device-specific data sheet BSL description for further details.

The correct termination of all unused pins of JTAG and BSL, should be open and must not power supplied externally to avoid damage to programmer.

For RST reset pin, users should ensure no reset IC on target board, watchdog for instance, otherwise, malfunction of reset IC may occur and cause failure in program.

Note:

The second pin of JTAG interface on PRGS430 multifunctional-offline programmer,outputs adjustable DC voltage(2.8V-3.6V). Target board or adapter is recommended without external power supply to avoid damage to programmer. VCC output of programmer needs to be off if external power is used.

III: software description

1)Software installation

Run LSD-PRGS430U-V2.10.exe,and install in specified or default installation directory. Shortcut of LSD-PRGS430 multifunctional-offline programmer software can be seen on desktop or start menu after installation.Users can run this program by clicking LSD-PRGS430U.exe on desktop or "start—>program—> LSD – PRGS430U.exe".

2)Software function and operation description

Main operation interface:

Run programmer software and then enter into operation interface of LSD-PRGS430U multifunctional-offli

ne programmer software,version of V2.10, as shown in following figure.

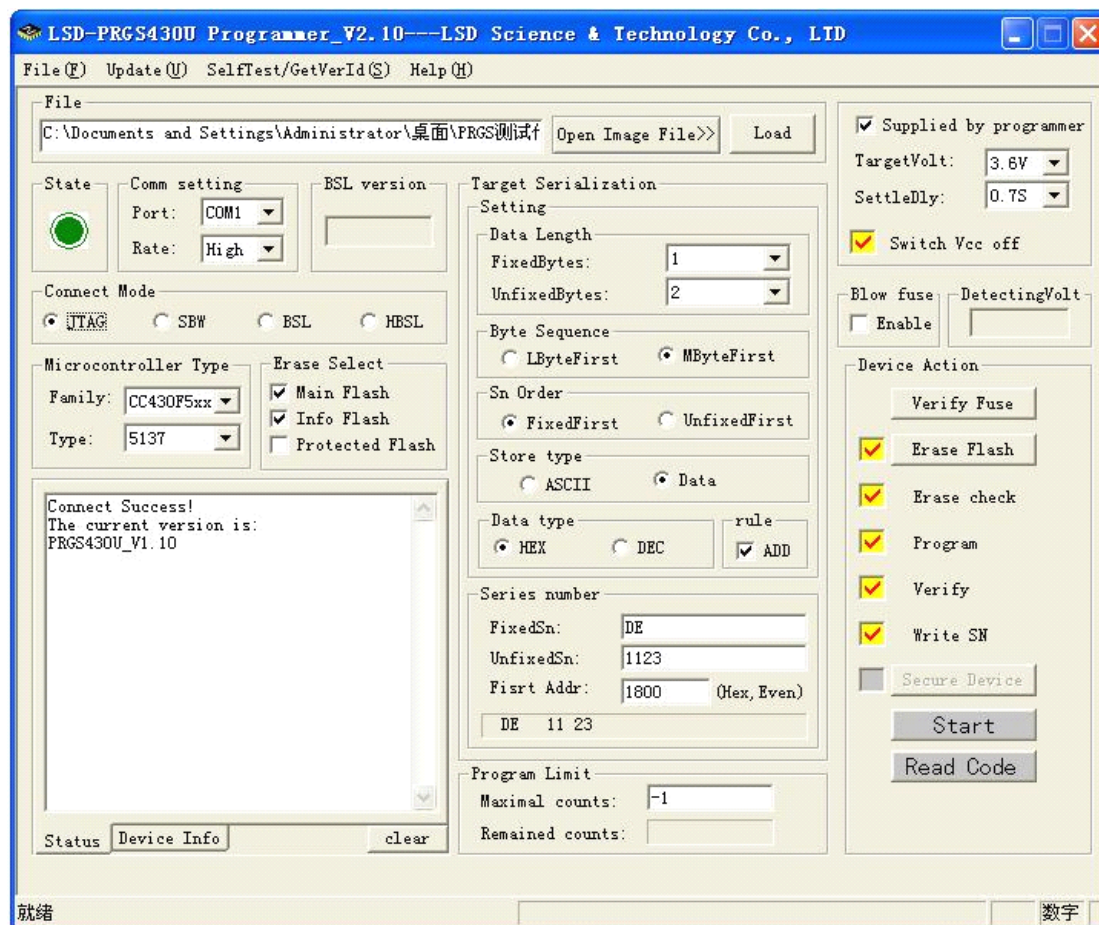


Figure 3

Communication set:

Port and rate are selected automatically by software, and there is no need for users to set. Rate is Low when update programmer , automatically modified to High after update.

Version of BSL:

In BSL mode, programmer can read BSL version information in target device (except F5xx).

Connection mode:

Select mode downloading to target MCU, and users can select from JTAG, SBW, BSL and HBSL. SBW mode can't be selected on default condition for devices on which SBW is not available.

MCU type:

Select 430 family and specific device type.

Erase selection:

It determines the erasing field when users are programming .Only the field selected can be erased when eras

ing operation is executed.

Main FLASH: only Main FLASH is erased.

Information FLASH: only InfoB\InfoC\InfoD FLASH is erased. For devices only with 2 information fields, InfoB(e.g.F449) is erased.

Protect Information FLASH:

only InfoA is erased. Pay more attention to F2xx because the calibration setting of DCO is stored in InfoA field of this family.

Note: Program memory should be within the scope of erasing field, because in FLASH memory, programming equals to writing 0 while erasing means writing 1.(Generally, the minimum unit for erasing operation is one single field.). Repeated programming to some address space without erasing operation, will cause programming errors, so program-check function is commendatory selection. When this function is selected, if programming errors occurs, programmer will alarm, stop (programming)operation and request reprogramming operation.

Information bar:

Two info windows,Report and DeviceInfo are included in Information bar. Report displays the current operating state while DeviceInfo displays information about current device selected.

Serial-number:

Serial-number is formed by two parts: fixed value and variation value. Fixed value is fixed and generally users' LOGO information. And also, variation value is variable and numerical order.

Because requirements may vary among users and arrangement mode to serial-number may be distinct, compound modes with certain flexibility are provided.

The image shows a 'Target Serialization' dialog box with the following settings:

- Setting:**
 - Data Length:
 - FixedBytes: 2
 - UnfixedBytes: 3
- Byte Sequence:**
 - ☐ LByteFirst
 - ☒ MByteFirst
- Sn Order:**
 - ☒ FixedFirst
 - ☐ UnfixedFirst
- Store type:**
 - ☐ ASCII
 - ☒ Data
- Data type:**
 - ☒ HEX
 - ☐ DEC
- rule:**
 - ☒ ADD
- Series number:**
 - FixedSn: 1122
 - UnfixedSn: 334455
 - First Addr: 1800 (Hex, Even)

At the bottom, there is a row of numbers: 11 22 33 44 55.

Figure 4

Data-length:

Length of fixed value: 4 bytes can satisfy most users' requirements according to feedback from users and more can be added in particular cases .

Length of variation value: This defines the range of numerical order. It varies from 3 to 6 bytes. Principle of satisfying needs should be kept when users select this length. E.g. if 90000 products are need to be produced , 3 bytes(hexadecimal:0xFFFFFFF,decimal;16777215)can satisfy need and there is no need to set 4 bytes.

Byte order:

Different byte orders can cause different results when 0x112233 is stored in storage location with initial address 0x1800 .

If low bytes are in the front, result is:

Number systems: HEX

Address	1800	1801	1802
Data	33	22	11

This is the default way to storage data for most of MCU.

If high bytes are in the front, result is:

Address	1800	1801	1802
Data	11	22	33

This way is intuitionistic and in keeping with the habit of examining Data.

Serial number order:

This determines the order of fixed value and variation value , i.e.fixed value locates in front of variation val

ue or otherwise.

E.g.fixed value is 0x1122, and variation value is 0x334455, if high bytes are selected in the front and initial memory address is 0x1800:

When fixed value in the front :

Number systems: HEX					
Address	1800	1801	1802	1803	1804
Data	11	22	33	44	55

When variation value in the front :

Number systems: HEX					
Address	1800	1801	1802	1803	1804
Data	33	44	55	11	22

Storage form:

ASCII Code: Fixed value and variation value are stored as ASCII Code, i.e.It's characters that are stored into memory space. Bytes number of fixed value and variation value equals to characters number. Variation value can't change progressively in this mode.

Data: Variation value can progressive increase in this mode , and of course, increment function following some rules should be enabled.

Data mode:

As is shown in Figure 4.

If the data filled in the frames are 1122 and 334455 in selected hex mode, fixed value and variation value are 0x1122 and 0x334455 respectively. Fixed value is 1122(equals to 0x0462)and variation value is 334455(equals to 0x51A77) if decimal system is selected. In one word , hex and decimal system referred to here are just different forms present to users on operation interface of host PC(R&D Engineers prefer hex, and Production Workers prefer decimal system). It's hex data that are loaded into MCU.

Rules:

Check boxes are valid and variation value of serial number can be incremented by one. Changing of numerical order is not allowed in ASCII Code mode.

serial-number:

Fixed value: Input fixed value. The maximum number of characters equals to bytes number of fixed value because one character must occupy memory space of one byte.

Variation value: Input variation value.The maximum number of characters equals to bytes number of variation value.

Serial-number address:

The programmer writes serial-number in word, so FLASH memory address of serial-number should be even and odd address will be error-reported. Serial-number address can't overlap with program address, otherwise failure on writing serial-number will be reported. Valid address of serial-number should be valid even FLASH address of the device. Writing to addresses beyond the lower 64-KB range is supported if target device acts as large capacity FLASH.

Serial-number display field:

The form displayed here is completely similar with form loaded into FLASH of MCU. Users can judge if the compound mode of serial-number is consistent with their supposed way.

Additionally, program-check is default selected when serial-number is written and it ensures the correction of serial-number written. Users can verify by reading target device.

Number of times for programming:

Users can set number of times for programming. "-1" or "65535" means unrestricted times. The maximum number that can be set is 65534. Users can't program code in case that it reaches the set programming times. Host PC will automatically read back programming times available after each programming in online mode if neither "-1" nor "65535" is selected. Users should pay attention to coordination of programming times and serial-number. E.g. if variation value of serial-number is 3 bytes, i.e. maximum of serial-number is 0xFF FF FF, and the initial value selected is 0xFF FF 00, programming times selected is 1000 and variation value will overflow after 256 times of programming. Users should pay attention to avoid this, otherwise errors occur in serial-number.

Power supply to target board:

If users mark check box "allowed power supply", programmer will output voltage that can be set by software on 2pin of JTAG. The recommended voltage range is 3.3-3.6V for reliability of system during programming. Please connect external power to 4pin of JTAG to avoid damage to programmer when MCU is powered by external power via users' target board. If 2pin is connected, please cancel "allowed power supply".

setup time:

It means synchronization time between programmer and MCU and is interval time from Programmer's power supply to target board to start of programming operation for a stable working volt on target board. Option "setup time" depends on users' system. System may possess energy storing devices with large capacity (e.g. large capacitor) that may affect programming voltage VCC. If VCC falls below voltage rating 2.7V when energy storing devices are charged, target device will be underpowered and programming fails.

Blown fuse:

This enables blown fuse operation. It's selected only in JTAG or SBW mode.

virtual voltage:

Programmer detects 2pin and 4pin of JTAG. It's 2pin that is detected when target board is powered by programmer. If it's not marked, it means target device is powered by separate external power and programmer detects voltage of 4pin.

Activity:

It includes checking fuse, erasing FLASH, erasion-check, programming, programming check, writing serial-number, blown fuse, power-down after programming, run, read codes, replace-load-run.

checking fuse: judge the fuse is blown or not. Please ensure that current mode is JTAG or SBW before operation (otherwise, invalid button). Select matching device and ensure that power supply is normal.

erasing FLASH: execute erasing FLASH operation. Erasing field is related to selection of users in "Erase selection".

erasion-check: programmer checks if the FLASH of target device is 0xFF.

programming: object code loaded in programmer is switched into JTAG or UART (BSL) protocols and loaded into target devices.

programming check: check if the codes loaded into target device are consistent with codes loaded in programmer.

writing serial-number: write serial-number. Serial-number is loaded when this operation is marked, or programmer will not execute this operation.

blown fuse: in JTAG or SBW mode, users can mark the option when blown fuse is enabled. Target devices can't be accessed in JTAG or SBW mode after fuse is blown, but they can be accessed in BSL mode (password file required).

power-down after programming: users can mark this option when target devices are powered by programmer. This means whether target devices are automatically powered down after loading codes.

Run: click "Run" after all required activities marked and then whole operation starts.

read codes:

in JTAG or SBW mode, click "Read codes" and a dialog box shown in figure 5 will be popped. Users can read codes in target devices using this function and the maximum length is 65535. It's necessary for users to ensure initial address input and byte length are valid for current device selected. In some cases, although the input address is invalid, programmer can also read back data which are completely meaningless.

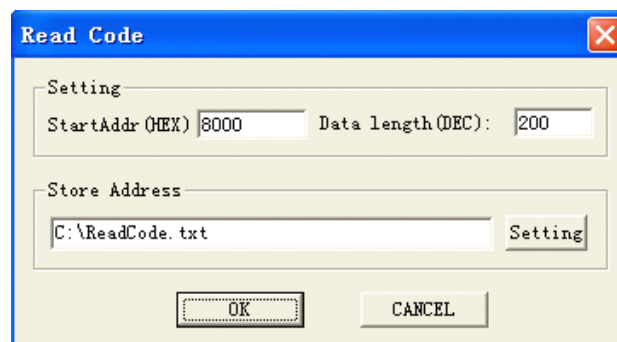


Figure 5

Only BSL mode can be used to read codes after fuse is blown and BSL password is needed. BSL password

points to 32 bytes data addressed from 0xFFE0 to 0xFFFF in target device.

In BSL or HBSL mode, click "read codes" and a dialog box requesting users to input BSL password will be popped, as shown in figure 6. Users can either input 32 bytes password manually or import .txt file including BSL password. Click "open BSL password" , import path of BSL password file and software will extract data from space ranged 0xFFE0-0xFFFF. If users need to change password, please click "Reset" and BSL password restores to 0xFF. Clicking "Confirm" makes host PC to lock BSL password and pops a dialog box as shown in figure 5. Users are asked to input initial address and byte length to read. Host PC holds the initial address of main FLASH in device users select as default initial address , and default length is 200 bytes.

NOTE: only when password is correctly, programmer can return data and sound "BLEEP" after reading. If password input is wrong, programmer will not return data and alarm with "BLEEP, BLEEP". If it works in online mode, host PC can prompt users causes of failure on reading . In some cases of wrong BSL password, failure on reading is one-off and will not impede following reading. But in some other cases, target device starts MassErase operation, erasing all data of FLASH and RAM to prevent unauthorized users acquiring codes . BSL version determines which one of two cases prescribed above will occurs. Generally, F5xx devices will start MassErase operation, see slau265.pdf. Application Report of TI for more detail information.

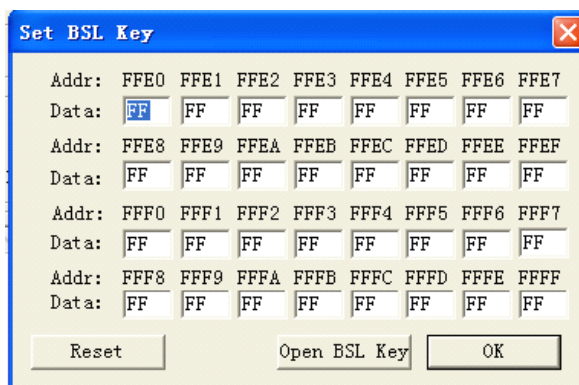


Figure 6

replace-load-run:

special use. There is no need to use this function for most of users(this function has been masked and can be unmasked if users need). If nonblank target devices are reloaded codes and data located in specific address should be holden, users can use this function. The following steps show how to use this function:

1. Click "open source code" and import code file to be reloaded.(users don't need to click "load")
2. Select "update" in menu bar and click "setting for updating target board...". "Hold codes" will appear in popped dialog box and number of data-preservation can reach up to 3. please fill in the initial address and byte number to preserve. Address o f additional 3-k should be cleared as 0 and data length is also filled as 0, if users require preservation while k(=1、 2、 3). otherwise, data in additional 3-k location will be preserved.
3. Mark the activity to be executed(eration-check, programming, programming check....)
4. Click "place-load-run". Note that times of "BLEEP" after programming successfully is related to k. Times of "BLEEP" after programming and placement successful should equals to(k+1). The function in reality is that programmer firstly reads prevented data from target device and integrate them into file which will be loaded and then restart programming. One "BLEEP" follows reading or programming successfully one time, so the times of "BLEEP" should result to (k+1).

IV: operation of PRGS430 multifunctional-offline programmer

Hardware and software about PRGS430 multifunctional-offline programmer has been discussed in previous section and the programming flow and matters need attention will be described in following section.

1. connect hardware:

LSD-PRGS430U programmer:

Connect USB interface of programmer to PC and also connect programmer with target board. After above operation, LCD on programmer will turn back light on and display the LOGO of LSD. Back light will turn off few seconds later.

Connect programmer with target board. Please connect target board with JTAG interface of programmer using 14pin wire in JTAG or SBW mode; please finish this connection using the wire provided by LSD(this wire can be self-made by users) to switch 14pin JTAG to 10pin BSL.

LSD-PRGS430-IIIA programmer:

Connect serial-port with 9pins of programmer to PC and also connect programmer with target board. Plug power adapter and 3 state indicator of programmer will flash 3 times.

Connect programmer with target board according to required programming mode.

2.Run software:

Programming window will appear after LOGO is popped. Buzzer of programmer will sound "BLEEP" and information bar will display version of software if connection is proper. The version is also the handshake signal between pc and programmer. Please relink serial port in programming window or check hardware connection and if programmer is powered.

Naming rule of software version is explained below:

LSD-PRGS430U programmer_V2.10 refers to the name of host pc software, and V2.10 is version number.

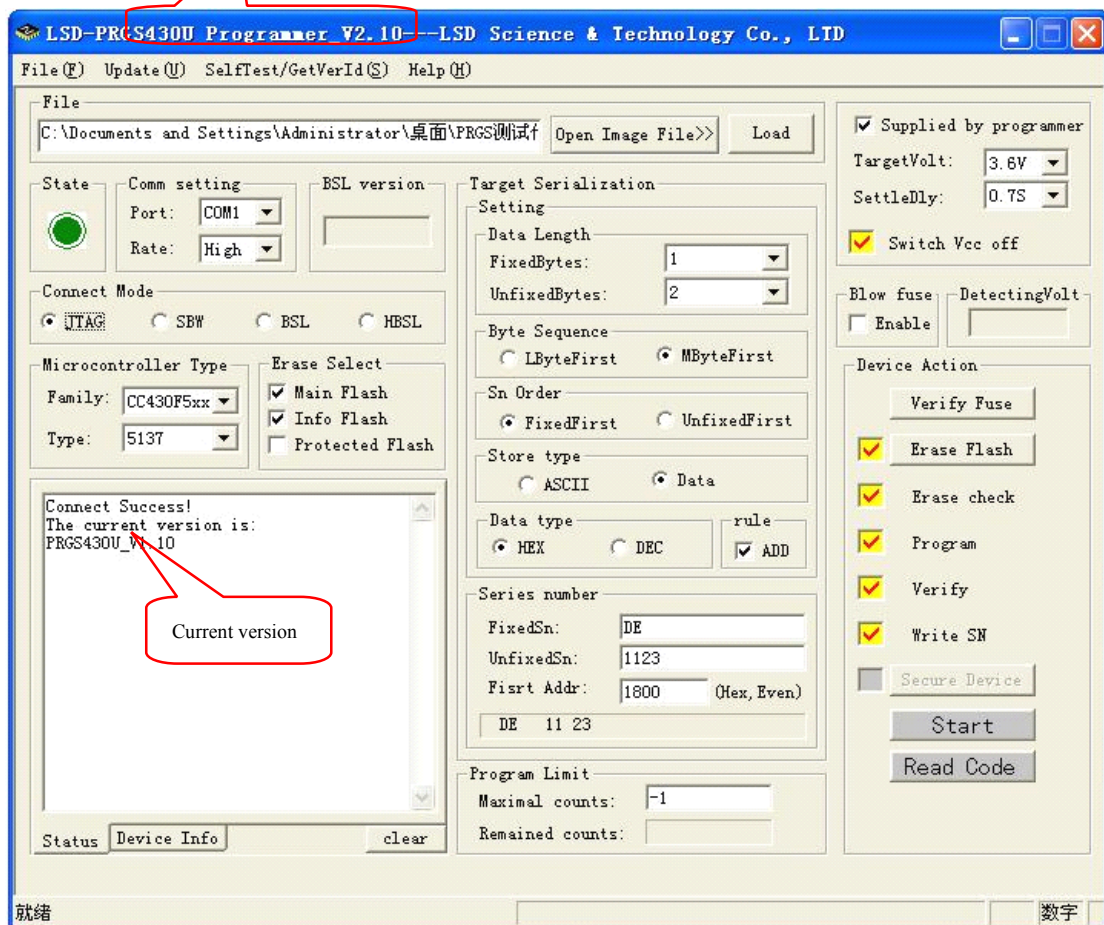


Figure 7

Prompted version number in Reoprt window of host PC is the software version in programmer. It's this software that we update to improve user experience and support more latest 430 devices that TI come out. The form of programming software version released by EDC of LSD is shown as following example:

PRGS430U_V1.10

If TI comes out new 430 devices or corrects certain bug, software version will be updated and become

PRGS430U_V1.11

.....

3.select programmed codes:

Click "open source code" and import path of source code to be programmed.

4.load program:

Click "load" and load codes into programmer. Programmer will self-check loaded codes automatically after load operation to determine whether error codes occur during data transmission. If the self-check is passed, "Sel

fCheck pass!" will appear on LCD and buzzer sound "BLEEP" and then programming operation can be executed. Otherwise, if the self-check is not passed, "SelfCheck Error!" will appear on LCD and buzzer sound "BLEEP, BLEEP" and codes need to load again because old loaded codes are wrong. Time for self-check depends on length of codes. Note: operations of host PC are ignored during self-check time of programmer.

5. Write serial-number :

Users can run this operation according to requirements. Set "Write serial-number" accordingly if it's needed. Skip this step if there is no need.

6.Run:

Initialize functions to be executed firstly in "Activity", such as check, erasing, blown fuse and so on.

Online mode :

If programmer needs working in online mode, click "Run". Operation result will be promoted in "Information display" of programming software. Repeat this operation if it fails.

Offline mode:

Disconnect USB connection and connect to 9VDC/200mA DC power if programmer needs offline mode. Press "Start" button on programmer and programmer will execute operation on MCU according to set function. Note: users should at least program online successfully one time before switch into offline mode to make programmer remember and check the validity of current setting. LCD displays the progress and selected mode during programming. (B) indicates BSL mode, (J) indicates JTAG mode and (S) means SBW mode. Progress bar can display current programming progress.

Buzzer will sound long "BLEEP" and result "success" or "failed" will be displayed on LCD after once operation of programmer according to users' setting. In case of IIIA programmer, green indicator light keeps on and buzzer sound long once if the result is right. And if the result is wrong, buzzer will sound rapidly twice. Complete programming process ends till now. Press "Start" button if users want to write next device.

Offline mode, programmer can remember last setting. If users don't need to change setting, programmer can be used continuously (not affected when powered down) after once initialization and so can be applied to mass production.

Notes for Use:

* for programmer with USB interface, connect USB in online mode and disconnect 9V/200mA DC power.

* close host PC software, power down programmer, power up and open host PC software if errors occur during program loading.

* please place 32 bytes interrupt vectors starting from @FFE0 to @FFFF at the end of program and end program with q if codes are released by IAR software. Codes produced by using Debug are recommended. Normal loading operation may not be completed if users can't follow processing method mentioned above.

V: update PRGS430 multifunctional-offline programmer

PRGS430 multifunctional-offline programmer can be upgraded to improve user experience and support mo

re latest 430 devices that TI come out. Users can download new software updated on our web site <http://www.lierda.com/> to update software online.

Steps to update programmer software are shown as following:

- 1.Setup the latest host PC software.
- 2.Connect programmer with PC using USB wire(serial port wire with 9 pins).
- 3.Open programmer software that has been installed. If "Connect successfully, current version is xxxxxxxx" appears in Report window, please skip to step 4. Otherwise ,check hardware connection, select "self-check/read version number" in menu bar and click "read version number". Repeat step 3 till version number is displayed which shows successful connection.
- 4.Click "Update", select "Update PRGS program" and import the path of latest codes updated(.prgs). Press "Start" button on programmer and update operation of programmer starts while blue progress bar on host PC can show progress.
- 5."Update successfully" will be displayed in "Message display" in case of success . Repeat step 4 if update operation fails.

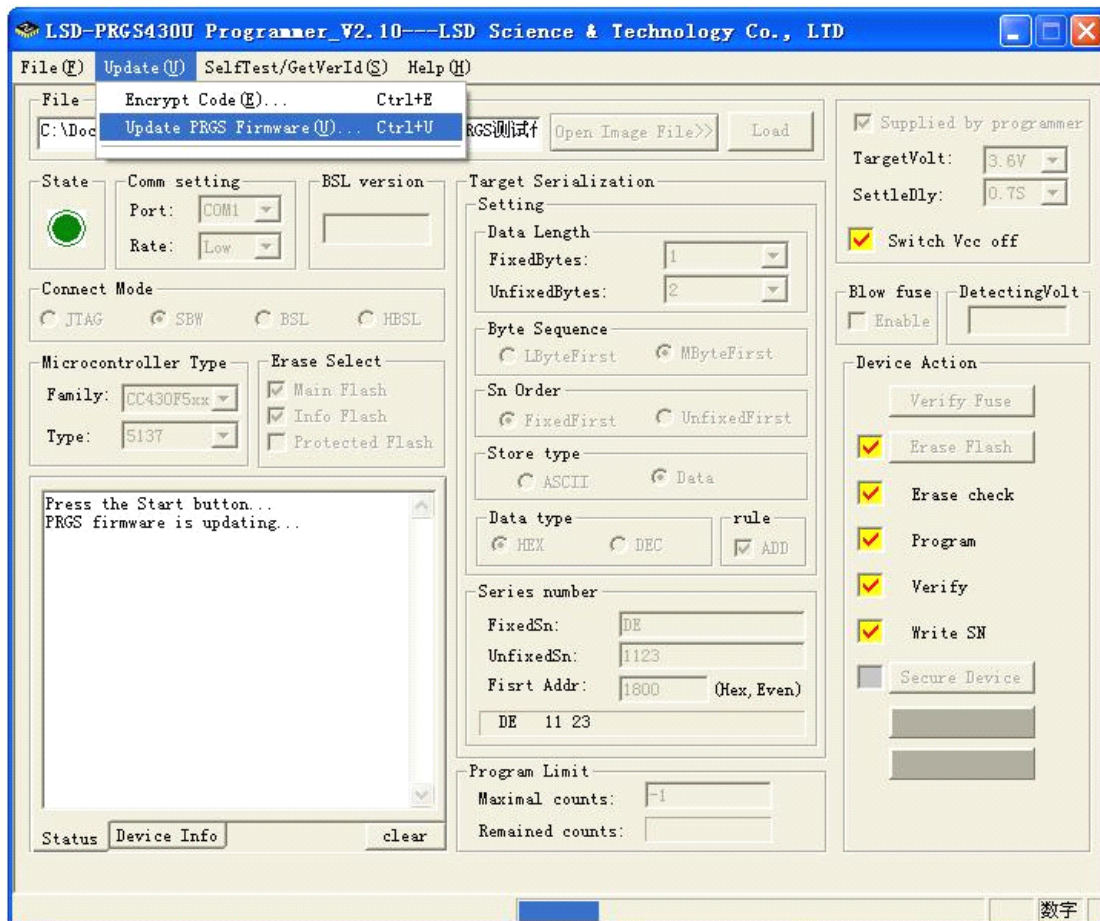


Figure 8

VI: Help

Direction for use of PRGS430 multifunctional-offline programmer is shown in Help section.

VII: Supported devices

VIII: Errors and Notes for Use

1. Device type selected, power supply mode and downloading mode mismatch with hardware connection.
2. Given device type mismatches with address range defined in downloaded files. For example, initial address in downloaded file is 0x1100 while users select F5438, so 0x1100 is illegal address in device F5438.
3. Supply voltage ($\geq 3.0V$ recommended) and setup time are selected incorrectly.
4. It's impossible for users to write codes to InfoA using JTAG mode firstly and then update data of InfoA in BSL mode after fuse blown when they apply non-F5 devices. The reason is that unlocking operation is needed when InfoA is written and unlocking instructions do not exist for non-F5 devices.
5. Pay more attention to unsure correct power supply mode of JTAG interface. Power pin of target device should be connected to 2pin of JTAG when devices are powered by programmer, and 4pin when external power supply.
6. Please ensure no capacitance is connected on RxD pin and TxD pin in HBSL mode. Otherwise, downloading will fail.
7. No matter what programming mode users select, please note that RST pin of target device can't connect to a capacitive load and watchdog chip. Otherwise, connection may fail.
8. Address (start with @) in txt file produced by IAR should be little-endian. If some codes are needed to load into InfoA, please place the codes separately starting with @ to avoid program failure.

11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF 00
11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF 00
11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF 00
11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF 00
11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF 00
11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF 00
11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF 00
11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF 00
11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF 00
11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF 00
11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF 00
11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF 00
11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF 00
11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF 00

@1000

11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF 00
11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF 00
11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF 00
11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF 00
11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF 00
11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF 00
11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF 00
11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF 00
11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF 00
11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF 00
11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF 00

@10C0

11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF 00

11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF 00

11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF 00

11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF 00

q

Annotations:

Original drawing from material of TI about downloading circuit of MSP430 devices is shown as following for users who feel unsure about it. We hope it is useful for users and please note several connection points described in NOTE A\B\C\D\E.

There are two kinds of downloading circuit: four-wire system and two-wire system. Four-wire system refers to usual JTAG connection with 4 signal wires TCK, TDO, TDI and TMS (VCC and GND excluded). Two-wire system refers to SBW connection with 2 signal wires SBWTCK and SBWTDIO as usual mentioned.

Users may meet three questions list as following during use:

- 1) Select value of resistance and capacitance in reset circuit.
- 2) Distinguish pin 2 and pin 4 of JTAG Function of TEST pin.

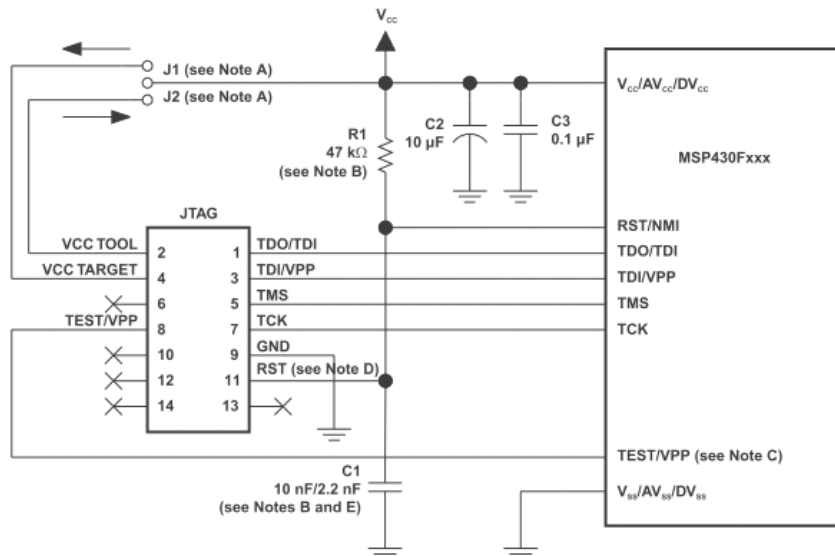
Answers are list in following section:

1) Recommended values by TI are 47K Ω and 10nF or 2.2nF. Users can set according to need in application as long as adequate time is provided to 430 devices to reset. It's not strait in JTAG mode but strait in SBW mode. In SBW mode, because RST pin multiplexes with SBWTDIO pin, SBWTDIO will be smoothed and devices connect unsuccessfully when capacitance value selected is too large. So 2.2nF is recommended by TI. This capacitance is also related to PCB wiring, users can get it off when failure in connection occurs in SBW mode.

2) Pin 2 of JTAG output voltage. Power supplied by programmer or emulator mentioned as usual means that programmer or emulator supplies voltage to pin 2 and MCU is powered via pin 2. Pin 4 feedbacks voltage detection. Pin 4 transfers external voltage to programmer or emulator, which carries out analog-to-digital conversion, and sends information back to host PC (e.g. IAR). Messages, such as alarm message, are given by host PC.

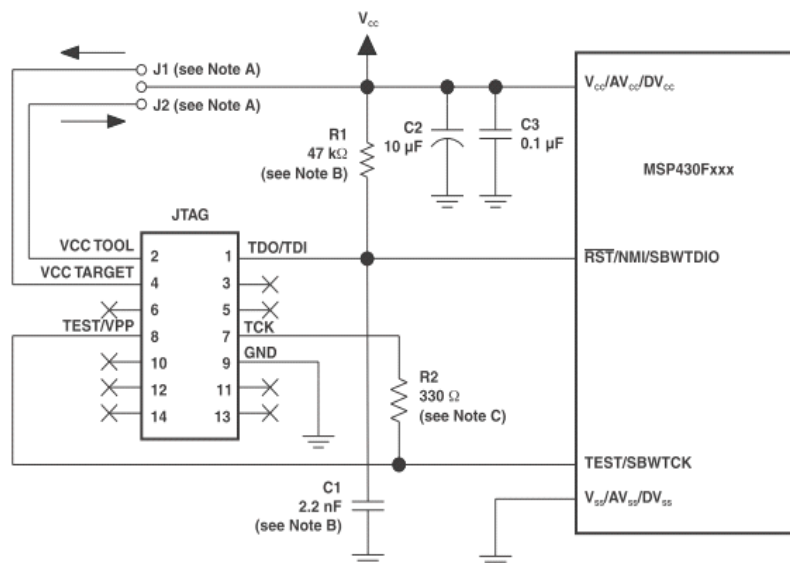
3) Users should firstly know it is necessary that TEST pin must be connected for specific 430 device with TEST pin. TEST pin can determine which mode PJ interface is in, JTAG or SBW mode. See SLAU256f.pdf, Application Report of TI for detail information.

JTAG (Four-wire system)



- A Make either connection J1 (if a local target power supply is used) or connection J2 (if powering the from the debug/programming adapter).
- B The RST/NMI pin R1/C1 configuration is device-family dependent. See the respective MSP430 family user's guide for the recommended configuration.
- C The TEST/VPP pin is available only on MSP430 family members with multiplexed JTAG pins. See the device data sheet to determine if this pin is available.
- D The connection to the JTAG connector RST pin is optional when using 4-wire JTAG communication mode capable-only devices and is not required for device programming or debugging. However, this connection is required when using 2-wire Spy-Bi-Wire communication mode capable devices in 4-wire JTAG mode.
- E When using 2-wire Spy-Bi-Wire communication capable devices in 4-wire JTAG mode, the upper limit for C1 should not exceed 2.2 nF. This applies to both TI FET interface modules (LPT/USB FET).

SBW (Two-wire system)



- A Make either connection J1 (if a local target power supply is used) or connection J2 (if powering the from the debug/programming adapter).
- B The device RST/NMI/SBWDIO pin is used in 2-wire Spy-Bi-Wire mode for bidirectional debug communication with the device and that any capacitance attached to this signal may affect the ability to establish a connection with the device. The upper limit for C1 is 2.2 nF when using current TI FET Interface modules (USB FET).
- C R2 is used to protect the JTAG debug interface TCK signal against the JTAG security fuse blow voltage that is supplied by the TEST/VPP pin during the fuse blow process. In the case that fuse blow functionality is not needed, R2 is not required (becomes 0 Ω), and the connection TEST/VPP must not be made.

Please contact with our local office or send mail to haoqiang@lierda.com if users have any questions during using process.

