Application Note 50

Title:	**Programming EM68xx 8-bit MCU**
Product Family:	**8-bit Microcontroller**
Part Number:	EM6812
Keywords:	Microcontroller, Programming, ISP
Date:	February 28, 2006

Introduction

All EM68xx flash based can be programmed from the Elnec programmer family:

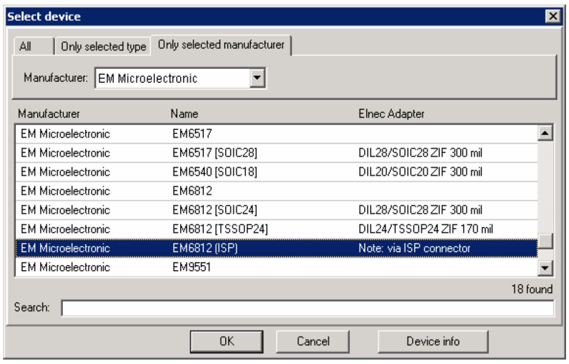
- ❑ LabProg+ (no ISP support)
- ❑ LabProg-48LV (no ISP support)
- ❑ JetProg (DIL48) (no ISP support)
- ❑ SmartProg – SmartProg 2 (ISP support)
- ❑ BeeProg (ISP support)

Be sure to have a software (PG4UW Elnec Software) release 1.60a or later release. Check www.elnec.com website.
For normal operation programming, please refer to the Elnec manual.
EM68xx programming can be done in 2 different modes:

- ❑ Insertion mode Programming
- ❑ In-System Programming with :
  - ▪ ISP Connector
  - ▪ EM Microelectronic Cable

Insertion mode Programming

1. Select Menu device target [Device/Select Device Menu]
2. Choose EM Microelectronic manufacturer and EM68xx microcontroller target
   ⇨ e.g. EM6812 target, then select package.



Select device

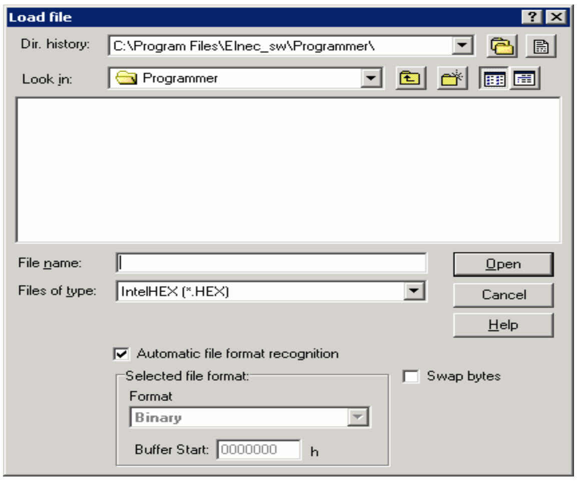
Manufacturer: EM Microelectronic

Manufacturer	Name	Elnec Adapter
EM Microelectronic	EM6517	
EM Microelectronic	EM6517 [SOIC28]	DIL28/SOIC28 ZIF 300 mil
EM Microelectronic	EM6540 [SOIC18]	DIL20/SOIC20 ZIF 300 mil
EM Microelectronic	EM6812	
EM Microelectronic	EM6812 [SOIC24]	DIL28/SOIC28 ZIF 300 mil
EM Microelectronic	EM6812 [TSSOP24]	DIL24/TSSOP24 ZIF 170 mil
EM Microelectronic	EM6812 (ISP)	Note: via ISP connector
EM Microelectronic	EM9551	

18 found

Search:

3. Load your IntelHex format *.hex file *[File/Load Menu].*



Load file

Dir. history: C:\Program Files\Elnec_sw\Programmer\

Look in: Programmer

File name:

Files of type: IntelHEX (*.HEX)

☑ Automatic file format recognition

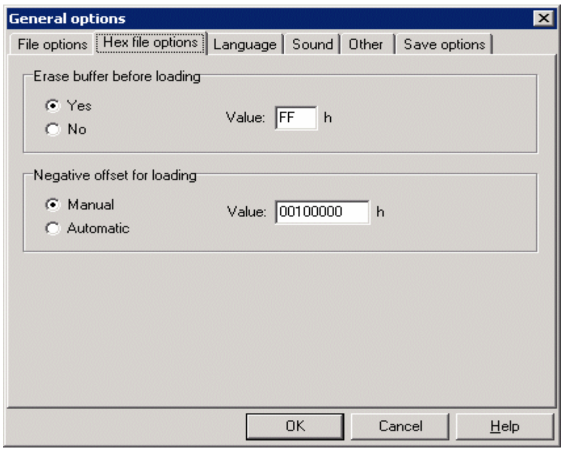
Selected file format:

Format: Binary

Buffer Start: 0000000 h

☐ Swap bytes

4. By default, IntelHex format generated from the CoolRISC Compiler tool chain generate a positive offset (**100000h**). To ensure a significant programming, you **must suppress this offset** with the Hex options Menu *[Options/General Options/Hex File Options Menu].*



General options

File options | Hex file options | Language | Sound | Other | Save options

Erase buffer before loading

● Yes
○ No

Value: FF h

Negative offset for loading

● Manual
○ Automatic

Value: 00100000 h

5. Perform programming.

In-System programming

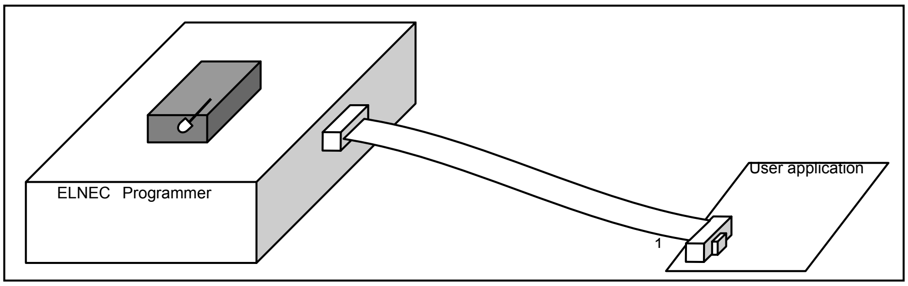


Figure 1: ISP

Introduction

In-System-programming allow EM68xx's programming in your application. It provides flexibility and save time when you need to program or update your Microcontroller's firmware. This feature is useful for debug or in the field for software updates.

This application note gives some information to design your application for In-System-Programming capabilities.

ISP principles

Three entities have to be considered: Programmer (In-System-Programming) connected to Target application which contain the Target EM68xx. Programmer act always as Master.

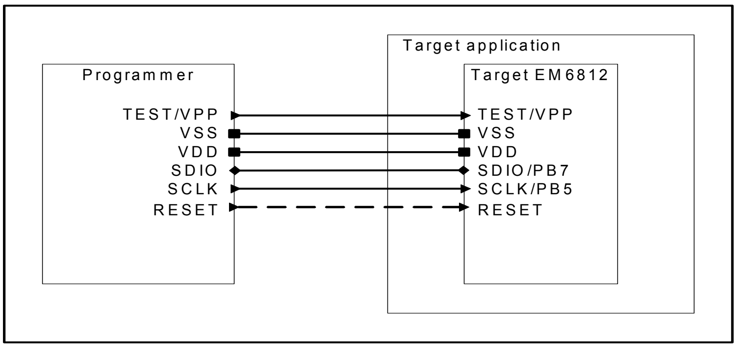


Figure 2: ISP Connections

The ISP interface is based on five wires. Three Power-lines and two serial-lines.

Note 1 : Additionally, reset pin can also be handled by programmer for specific purpose.

Note 2 : Other IO's to be considered :

- ❑ Port A: Becomes an output during programming and the value is 0x00. Default state (after programming) is inputs with pull-downs.
- ❑ Port B [0:4]: Inputs with pull-downs during programming and keep this state as default state.
- ❑ Port B [6]: Output datas during programming when TEST = '1'. These datas must be ignored but state are VSS / +VDD, so take care with your application.

	Pin	Function	Comment
Power-lines	VPP/TEST	Reset / Enable MTP Mode Apply High-Voltage for Write / Erase Flash	Allow programmer to control the MTP mode and apply high-voltage for Flash programming.
	VDD	Target EM68xx power	Target power must be in the correct DC characteristic during programming.
	VSS	Common Ground	Ground must be common to the whole system
Serial-lines	SDIO (PB7)	Serial Bi-directional data line for communication between programmer and EM68xx target.	Generated from the programmer (Master) to target EM68xx for writing datas or configuration. Generated from the target EM68xx to the programmer for reading datas.
	SCLK (PB5)	Serial Clock line for communication between programmer and EM68xx power target.	Generated from the programmer. Also be used by application (Port B - bit 7)

Table 1: Connection table

It is very important to :

- ❑ Connect together grounds. Ground from Target application connected with ground from Target EM68xx and ground from programmer.
- ❑ Let the VPP/TEST pin free in order to be managed by programmer. It is not necessary to add pull-down on your application, an internal weak pull-down is already implemented in the EM68xx target to fix the pin level.
- ❑ Put the Target System in the correct VDD range. First case, programmer supply the Target EM68xx only. Second case, programmer supply the application or part of the application. Third case power supply from Target application is used.
- ❑ Avoid conflicts or significant load on Serial-lines.

Hardware considerations

All pins used for ISP programming must be able to be correctly driven from programming point of view. All application pins should not disturb programming. The recommended operation is to:

- ❑ Disconnect or isolate all applications fields.
- ❑ Supply only the EM68xx target
- ❑ Ensure that serial lines does not interact with application.

Typically, jumpers or switches configuration might be used to establish the direct connection from the programmer to the EM68xx Target.

For some cases, it is also possible to power or keep connected a part of the application during programming. In such cases particular cautions must be respected and some limitations exist due to application connections on the programming lines.

In some cases, Target power-supply can also be used.

VPP / TEST

This pin is used to reset / enable the MTP interface and apply the High-Voltage used during Flash Erasing / Writing. This pin should be **exclusively** driven directly from the programmer. This pin already includes a built-in pull-down and does not require other external handling. This must be kept unconnected on your application PCB except the programmer connection.

VDD

The specified VDD range must be kept during programming operation.

It is recommended to completely disconnect the VDD application from the VDD programmer.

In case, it is not possible to disconnect the application and if the application current consumption is very limited (typically below 20 mA), programming can nevertheless be processed.

Several cases have to be considered:

- ❑ The programmer has not enough drive or can not provide good rise rate timings on VDD (too much load capacitance), a buffer stage can be necessary. Power-supply comes from this buffer stage.
- ❑ The application is limited (limited consumption), the programmer can directly drive the whole application. Power-supply comes from the programmer.
- ❑ Target power-supply from your application is used, it is required to avoid any electrical conflicts and place some security to avoid any damage.

Figure 3 shows a typical example of protection that can be used. The final solution is totally dependant of application requirements and programming flexibility. On this example:

- ❑ Jumpers may be used to configure programming or application mode.
- ❑ Zener may be used to protect the programmer from external (application) over-voltage if programmer power-supply is used and/or limit VDD during programming.
- ❑ Diode may be used if the application delivers the power-supply during programming.

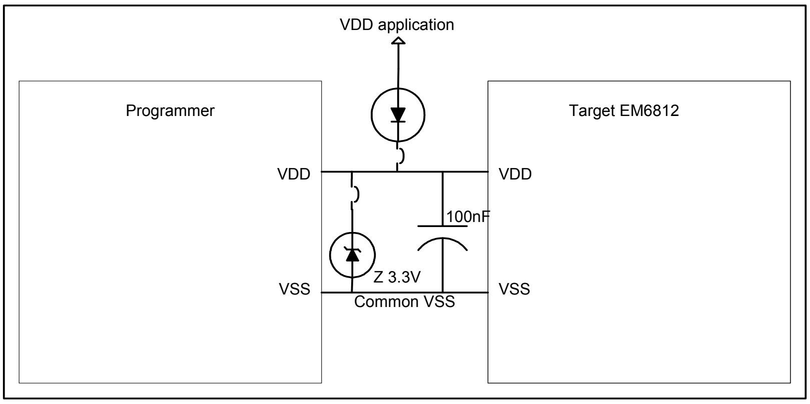


Figure 3: VDD Circuitry example

VSS

All ground must be connected together. Target EM68xx ground connected to Target application and programmer.
Ensure common earth point for Target application, programmer and PC.

SDIO / SCLK

When these 2 pins are used for the application you must keep them available for programming purpose. They must be isolated during programming from the application. The solution to isolate these pins are dependant of your application, the load (programmer output impedance, …) … Simple additional resistors to avoid electrical contention or specific buffers have to be examined.
Figure 4 show a typical example of protection that can be used.

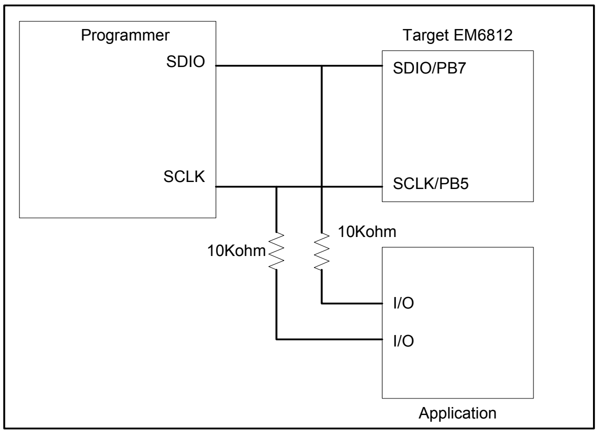


Figure 4: SDIO/SCLK isolation example

Using EM Microelectronic Cable

LabProg+ / LabProg-48LV / JetProg (DIL48) have no ISP support.
If you need to program your EM68xx in your application and if your Elnec programmer does not have ISP support this solution have to be considered.
We recommend to use the dedicated cable that can be provided by EM Microelectronic (contact EM Microelectronic) or use a similar one.
EM Microelectronic Cable : DIL40 male ⇔ HE10 2x8 pole female

N.C.	•1	o	o	2	N.C.
N.C.	3	o	o	4	SCLK
N.C.	5	o	o	6	N.C.
N.C.	7	o	o	8	SDIO
N.C.	9	o	o	10	N.C.
VSS	11	o	o	12	VDD
VPP/TEST	13	o	o	14	N.C.
N.C.	15	o	o	16	N.C.

Figure 5: Pinout of the HE10 2x8 pole female

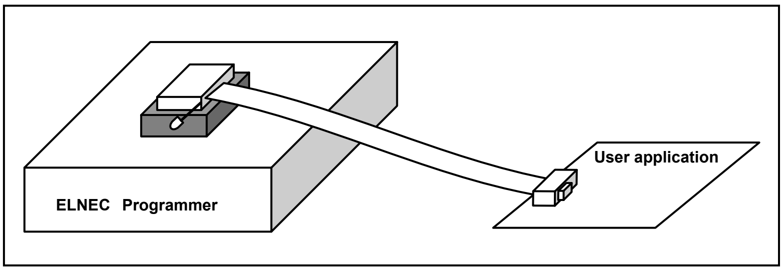


Figure 6: ISP with EM Microelectronic cable

A HE10 male connector has to be placed on the user application.
For programming, select the **EM68xx / DIL24** package in Elnec programmer software Menu, **Disable** the Insertion TEST before programming and **follow the Insertion mode Programming** steps.

Using Elnec ISP connector

Smartprog and Beeprog have ISP capabilities in addition to the on-socket programming mode. They use a specific connector called ISP (In-System-Programming). If you are using an other Elnec programmer without ISP connector and need to program in your application, refer to the Using EM Microelectronic Cable paragraph.
This connector has to be connected to your application through the provided ISP cable. It is recommended to use this cable or a similar one otherwise programming might not be performed correctly due to different cable characteristic.
ISP connector type : Connector HE10 – 2x5 – male

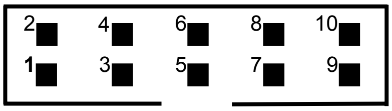


Figure 7: ISP connector layout – Top View

Location	Name	Location	Name
1	**VDD**	6	**SDIO (PB7)**
2	Don't connect	7	**VSS**
3	**Reset - optional**	8	**SCLK (PB5)**
4	Don't connect	9	**VSS**
5	Don't connect	10	**VPP/TEST**

Table 2: ISP connector pinout

During programming when VDD is supplied by programmer VDD is set to 3.75V
During programming a VDD TEST is done. Required VDD range (Min <=> Max) is defined in table 3.
Programmer IDD range is defined in table 4.

	Min.	Typ.	Max.
VDD	3.15V	3.3V … 3.6V	3.7V

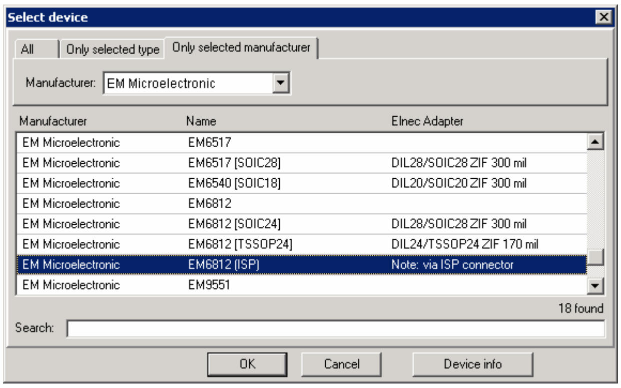
Table 3: Required VDD range

	Beeprog	Smartprog
IDD max.	350 mA	150 mA

Table 4: Programmer IDD drive

ISP steps

1. Select Menu device target *[Device/Select Device Menu]*
2. Choose EM Microelectronic manufacturer and EM68xx microcontroller.
   ⇨ e.g EM6812 target, then select ISP.



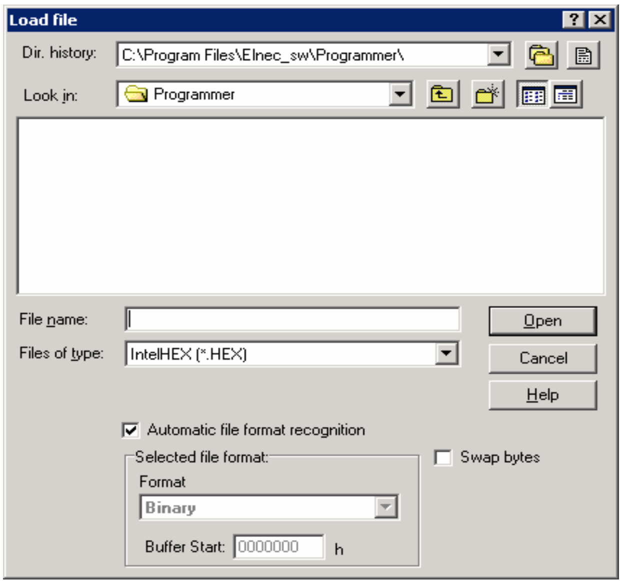
Manufacturer: EM Microelectronic

Manufacturer	Name	Elnec Adapter
EM Microelectronic	EM6517	
EM Microelectronic	EM6517 [SOIC28]	DIL28/SOIC28 ZIF 300 mil
EM Microelectronic	EM6540 [SOIC18]	DIL20/SOIC20 ZIF 300 mil
EM Microelectronic	EM6812	
EM Microelectronic	EM6812 [SOIC24]	DIL28/SOIC28 ZIF 300 mil
EM Microelectronic	EM6812 [TSSOP24]	DIL24/TSSOP24 ZIF 170 mil
EM Microelectronic	EM6812 (ISP)	Note: via ISP connector
EM Microelectronic	EM9551	

18 found

Search:

3. Load your IntelHex format *.hex file *[File/Load Menu].*



Dir. history: C:\Program Files\Elnec_sw\Programmer\

Look in: Programmer

File name:

Files of type: IntelHEX (*.HEX)

☑ Automatic file format recognition

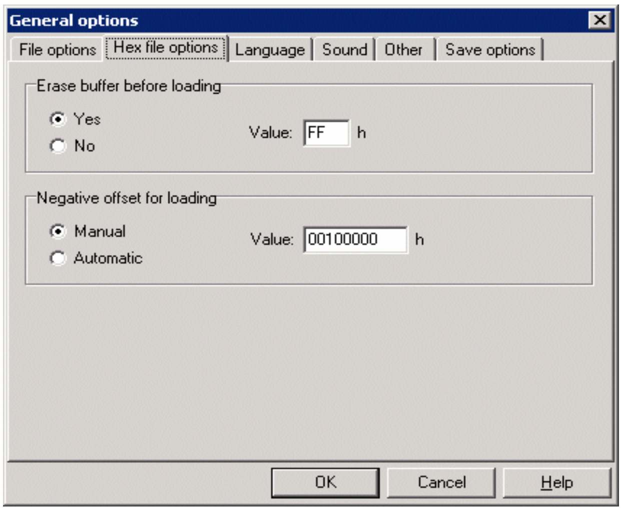
Selected file format:

Format: Binary

Buffer Start: 0000000 h

☐ Swap bytes

4. By default, IntelHex format generated from the CoolRISC Compiler tool chain generate a positive offset (**100000h**). To ensure a significant programming, you **must suppress this offset** with the Hex options Menu *[Options/General Options/Hex File Options Menu].*



Erase buffer before loading

● Yes
○ No

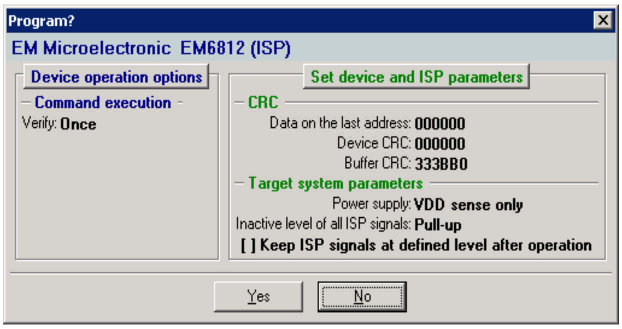
Value: FF h

Negative offset for loading

● Manual
○ Automatic

Value: 00100000 h

5. Perform programming and check the “Set device and ISP parameters”.



EM Microelectronic EM6812 (ISP)

Device operation options

Command execution

Verify: Once

Set device and ISP parameters

CRC

Data on the last address: 000000

Device CRC: 000000

Buffer CRC: 333BB0

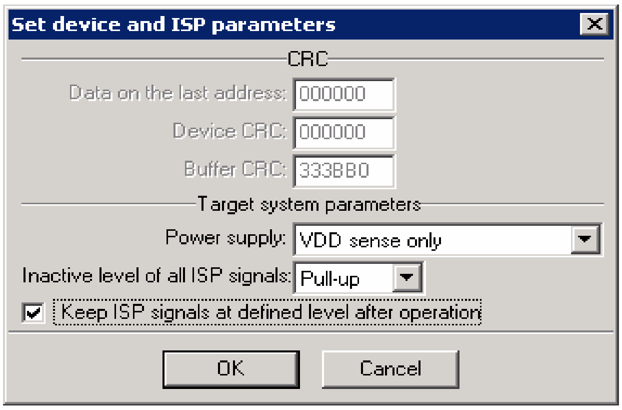
Target system parameters

Power supply: VDD sense only

Inactive level of all ISP signals: Pull-up

[ ] Keep ISP signals at defined level after operation

Recommended options:
Select “Pull-Down” and Enable the “Keep ISP signals at defined level after operation” in the Target system parameters.
In case programmer supply the target, select “VDD sense only” as Power-Supply. In case power-supply is delivered by the application select “VDD for chip from programmer”.



Set device and ISP parameters

CRC

Data on the last address: 000000

Device CRC: 000000

Buffer CRC: 333BB0

Target system parameters

Power supply: VDD sense only

Inactive level of all ISP signals: Pull-up

☑ Keep ISP signals at defined level after operation

© EM Microelectronic-Marin SA, 02/06, Rev. C