Application Note

AN2272/D
Rev. 0, 3/2002

In-Circuit Programming of FLASH Memory in the MC68HC908LJ12

By Vincent Ko
Systems Engineering
Microcontroller Division
Hong Kong

This application note describes In-Circuit Programming (ICP) of the FLASH memory in the mot out of file MC68HC908LJ12 (LJ12) microcontroller. The LJ12 is a general purpose device based on the HC08 architecture, with 12k-bytes of on-chip FLASH, 512-bytes of RAM, an infrared serial communications interface, a serial peripheral interface, an analog-to-digital converter, a real time clock, and an LCD driver module.

The text is divided into two parts:

- PART 1 — General overview of ICP and techniques that can be applied to the LJ12.
- PART 2 — ICP example on the LJ12.

For detailed specification on MC68HC908LJ12, please refer to the data sheet, Freescale order number: *MC68HC908LJ12/D*.

PART 1: INTRODUCTION

In-circuit programming is a process by which the device is programmed or erased with the device on the final circuit board — the *target system*. This allows the *user code* to be changed without having to remove the device off the target system for reprogramming or initial programming.

On LJ12, the 12k-bytes FLASH memory is allocated for the user code, with an additional 48-bytes for user defined reset and interrupt vectors. A high voltage supply is not required for program or erase operations; as this is generated by an internal charge-pump. This FLASH memory can be programmed or erased using software routines running either in *User mode* or *Monitor mode*, by writing to the FLASH Control Register at address $FE08.

User Mode

In
the FLASH memory
most of the time.

Monitor Mode

In Monitor mode, the LJ12 is running the *Monitor Code* that has been permanently programmed into an area of memory in the LJ12 during fabrication. The monitor code is used for serial communication between the LJ12 and an external host, via LJ12's PTA0 pin. Programming an initially blank LJ12 FLASH memory is executed in monitor mode.

Initial FLASH Programming

The LJ12 operating mode is latched after a power-on-reset (POR), and is determined by the contents of the reset vector ($FFFE–$FFFF) and the logic level on the following pins: IRQ, RST, PTA0, PTA1, PTA2, and PTC1. (For details, please refer to the Monitor ROM section in the data sheet.)

Block Protected FLASH Memory

FLASH memory cannot be erased or programmed when it is *block protected*. The FLASH Block Protect Register at address $FE09 is used to protect a block of FLASH, or the entire FLASH. Writing $FF to the Block Protect Register unblocks the entire FLASH memory.

ROM-Resident Routines

The LJ12's ROM-resident routines are designed to simplify FLASH programming and erasing for the user. These routines are permanent codes stored in a ROM area that can be executed using a subroutine call. **Table 1** shows the four routines for FLASH program and erase operations. (For the full list of routines, please refer to the Monitor ROM section in the data sheet.)

Table 1. LJ12 ROM-Resident Program/Erase Routines

Routine Name	Routine Description	Call Address
PRGRNGE	Program a range of locations.	$FC06
ERARNGE	Erase a page or the entire array.	$FCBE
MON_PRGRNGE	Program a range of locations in monitor mode.	$FF28
MON_ERARNGE	Erase a page or the entire array in monitor mode.	$FF2C

IN-CIRCUIT PROGRAMMING IN USER MODE

ICP in user mode can be implemented so as to maintain target system operation while reprogramming the FLASH memory in the LJ12. Reprogramming the FLASH memory in the LJ12 involves two stages. The first stage is an erase operation to erase the existing data in the FLASH memory. The minimum erase size is 128-bytes, known as a *page*. The MASS bit in the FLASH Control Register provides the option for erasing the entire FLASH array in one operation, known as *MASS erase*. It should be noted that an erased byte of FLASH memory reads as $FF. The second stage is the programming process, which programs the blank FLASH memory with new data. Thus, reprogramming involves: erase and program.

ICP Code

Performing ICP in user mode requires that the *ICP code* to be stored in a part of non-volatile memory that can be called by the user program. This means the ICP code needs to be a section of code that is part of the user code, and programmed into LJ12's FLASH memory. With this in mind, ICP in user mode cannot be performed if the FLASH memory is initially blank (a blank device). Initial blank devices are programmed in monitor mode (see next section for **IN-CIRCUIT PROGRAMMING IN MONITOR MODE**).

With the ICP code programmed into the FLASH memory, ICP can be initiated by software or hardware, and can operate in two ways:

1. The ICP code sets up the LJ12 a communication link with an outside host system via the LJ12 port pins or the SCI interface. The host issues commands to erase the LJ12's FLASH memory and downloads data to program the FLASH memory. In this case, the LJ12 ICP code is acting as a command interpreter.

or

2. The ICP code runs its erase routine to erase the FLASH memory. The ICP code then sets up the LJ12 a communication link with an outside host system via the LJ12 port pins or the SCI interface. New data is downloaded to program into FLASH memory.

In both methods, the ICP code must be loaded into the RAM area of memory, and the code executed in the RAM area. Program or erase operations are not allowed while program is running in the FLASH area. If it was possible for the ICP code to execute in the FLASH area, there is the danger of erasing the ICP code itself.

IN-CIRCUIT PROGRAMMING IN MONI

In monitor mode, the LJ12 is running the *monitor code* that has been permanently programmed into an area of memory ($FC00–$FDFF and $FE10–$FFCF) in the LJ12 during fabrication. First time programming of the LJ12's FLASH memory can only be carried out in monitor mode.

The monitor code consists of routines for communicating to a host connected using a serial link via pin PTA0. Once the link is established, control of the MCU is transferred to the host system. The host controls the MCU by directly writing to the MCU registers. Monitor mode can be entered in two ways:

High Volt Entry to Monitor Mode

Similar to many Freescale 8-bit MCUs, with a high voltage on the $\overline{\text{IRQ}}$ pin ($1.5 \times V_{DD}$, max. 8V, for LJ12) and defined logic levels on several I/O pins (PTA1, PTA2, and PTC1 for LJ12), monitor mode is entered after a power-on reset (POR). With this high voltage entry method, the clock input to the MCU (at OSC1) must be from an external oscillator with a frequency of 4.9152MHz or 9.8304MHz. This clock divides to produce the 9600 baud communication speed on PTA0. The LJ12's PLL is disabled in with this entry method (for details, please refer to the Monitor ROM section in the data sheet).

Blank Vector Entry to Monitor Mode

With the new FLASH memory implementation, there was a need to reduce the number of wire connections to the target system to program the MCU when ICP is required. The other method for entry to monitor mode is a blank (erased) reset vector. The reset vector can only be erased by a mass erase operation. If the $\overline{\text{IRQ}}$ pin was grounded during the mode latch after power-on reset, the PLL is enabled, allowing the use of a 32.768kHz crystal between OSC1 and OSC2 for the input clock, producing a baud rate of 9600. If the $\overline{\text{IRQ}}$ pin was pulled high (to V_{DD}) during the mode latch after power-on reset, the PLL is disabled, a 4.9152MHz input clock (crystal or external oscillator) will produce a baud rate of 4800. This method saves three wires when compared with the high voltage entry method; no connection to PTA1, PTA2, and PTC1.

Implementing ICP in monitor mode has the advantage that no ICP code needs to be written as part of the user code. In addition, the *MCUscribe* program or *ICS Freescale* utilities are available for the PC host system that talks to the MCU via PTA0 serial link.

OTHER ICP CONSIDERAT

Signal Conditioning

To allow an uninterrupted programming process, normal system activities will usually be halted during an ICP operation. Therefore, at the start of the ICP process, the MCU should be configured such that no pin contention or runaway signal will occur during the ICP process. Also note that when the system is first switched-on with a MCU having a blank FLASH memory, the port pins default to their reset states.

Pin Isolation

If the MCU pins used for connecting to the external host are shared with the target system, make sure they are isolated to the proper logic level when the ICP connection is made.

COP Watchdog Operation

When performing ICP in user mode, COP operation must be considered. Once the COP is enabled, it cannot be disabled during user program runtime. Therefore, in the ICP erase and program routines, the COP counter must be periodically cleared to prevent a COP reset. In monitor mode, the COP is disabled.

PART 2: ICP EXAMPLE

The coding below performs the mass erase in user mode. The location RAMFILE is reserved for the data block storage of the ERARNGE subroutine call. When an ICP request is granted, this code needs to be uploaded to RAM before it can be executed by calling RAMSTART. The LJ12 is then reset. In this example, an illegal opcode reset is used.

```
;Reference upload coding for mass erase in user mode
ERARNGE                 EQU       $FCBE
                ORG     RAM
RAMFILE:
                DC.B    $14                     ;Indicates 4*bus (bus=4.9152MHz)
                DC.B    $40                     ;Data size assumed to be 64
                DC.W    $FFFF                   ;$FFFF means mass erase
                DS.B    64                      ;Reserve for data array
RAMSTART:
                LDHX    RAMFILE
                JSR     ERARNGE                 ;Mass erase the FLASH memory
RESET:
                DC.B    $32                     ;Perform illegal opcode reset
```

Af
Wit
following code can be loaded into the LJ12 RAM for FLASH programming.

```
;Reference upload coding for programming in monitor mode
MON_PRGRNGE             EQU       $FF28
                ORG     RAM
RAMFILE:
                DC.B    $14                     ;Indicates 4*bus (bus=4.9152MHz)
                DC.B    $40                     ;Data size assumed to be 64
                DC.W    $C000                   ;Start of FLASH memory
                DS.B    64                      ;Reserve for data array
FLASHPRG:
                LDHX    RAMFILE
                JSR     MON_PRGRNGE
```

The location RAMFILE is reserved for the data block storage of the MON_PRGRNGE subroutine call. User can use the monitor commands, WRITE or IWRITE, to fill up the RAMFILE with data before issuing the RUN command to execute the FLASHPRG routine to program the data into FLASH. When it is done, MON_PRGRNGE will return the control back to the monitor code. This programming procedure is repeated until the entire FLASH memory is programmed. After programming, the LJ12 will enter user mode after a power-on reset.

NOTES:

How to Reach Us:

Home Page:
www.freescale.com

E-mail:
support@freescale.com

USA/Europe or Locations Not Listed:
Freescale Semiconductor
Technical Information Center, CH370
1300 N. Alma School Road
Chandler, Arizona 85224
+1-800-521-6274 or +1-480-768-2130
support@freescale.com

Europe, Middle East, and Africa:
Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
support@freescale.com

Japan:
Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku,
Tokyo 153-0064
Japan
0120 191014 or +81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:
Freescale Semiconductor Hong Kong Ltd.
Technical Information Center
2 Dai King Street
Tai Po Industrial Estate
Tai Po, N.T., Hong Kong
+800 2666 8080
support.asia@freescale.com

For Literature Requests Only:
Freescale Semiconductor Literature Distribution Center
P.O. Box 5405
Denver, Colorado 80217
1-800-441-2447 or 303-675-2140
Fax: 303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com