Dragonchip

# Software Development Notice for DC6688

AppNote000

Document Revision 4.7

May, 2020

# Table of Content

# 1 Introduction

The Objective of this document is to provide the user a snapshot on some areas in DC6688 Family.

Those customers who have to develop application are obliged to check this document to make sure the application working properly.

# 2 Dragonchip Tools

## 2.1 Troubleshoot

Before software development, it is advised to review the AppNote106.

In addition, the following should pay attention:

1. Avoid using generic pointer
   It was found the Keil C51's Generic Pointer implementation might not be supported by our MCU. Instead, memory-specific pointer is recommended, which fix its referencing memory space during variable declaration [1].
   An example is shown below.

**Memory-specific Pointers**

```
char idata * input_string; // memory pointer to indirect SRAM space
unsigned int xdata * cur_index; // memory pointer to XRAM space
char code * message; // memory pointer to CODE space
```

**Generic Pointers *(Not recommended)***

```
char * buffer; // memory pointer that can points to SRAM, XRAM or CODE space
```

# 3 DC6688FSA

## 3.1 Firmware

| | Incident | Item | Description |
|---|---|---|---|
| A | Abnormal software operation | 'PC' register in XFR | Make sure bit 6 is '0' |
| B | Abnormal PC1 state | PCCON[1,0] in XFR | 01 = prohibited |
| C | High stop mode Idd | Add instructions[1] | Add the following instructions to the beginning of the program<br>MOV R0,#0FH<br>MOV A,#00111111B<br>MOVX @R0,A |
| D | High stop mode Idd | Add instructions[2] | Both bit 1 ~ 0 of 'PCCONL' register should be set to 1 before entering stop mode |
| E | Abnormal Serial Communication | UART 0 and UART 1 Receive | Both UART might have chance to get wrong data. The only solution is to send back the received data to confirm whether IC received correctly [3].<br><br>An alternative solution is to use software UART on receive part. Example refers to Application Note 024. |
| F | Abnormal Timer operation | Timer 0 and Timer 1 setting | The following condition at the same time happened is invalid:<br>i) Timer 1 set to mode 1, and<br>ii) Timer 0 set to mode 2 |
| G | Abnormal Data flash content | Initialization Data for Data Flash memory | To avoid using firmware to initialize Data Flash memory during in production line, it is recommended to also download the image to data flash memory by SL or SPI. |
| H | Abnormal Data flash content | Add 100ms delay at the beginning of the program | This applies to case where factory do initialization on data flash memory in production line by firmware itself. In view of the unstable power supply in production line found at the instant of insertion, add this delay at the beginning of the program to let the power stable before starting to run the firmware in production line.[4] |
| J | Abnormal software operation | Initialization for SRAM | This is a must as on power up, the contents in the SRAM are undetermined. |
| K | Software halt | Counter A reset | Counter A should be reset after exit from stop mode |

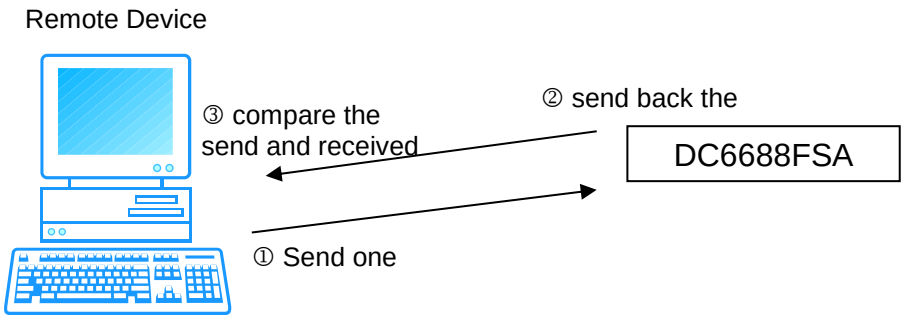
| | | | |
|---|---|---|---|
| **L** | Abnormal Interrupt | Bit 'IT0' in SFR | IT0 should set to 0 for PA interrupt for proper operation |
| **M** | Abnormal Interrupt[5] | Accessing Data Flash memory | 1) CPU suspends operation<br>2) only one interrupt is seen by CPU no matter how many interrupts from a peripheral<br>3) any interrupt will be delayed to serve by CPU until completion of accessing data flash<br>4) Disable watchdog before accessing data flash |
| **N** | High stop mode Idd | I/O port configuration in stop mode | In order to meet the Stop mode Idd current as stated in specification, the I/O port should be carefully configured. The following is prohibited:<br>1) Input without pull high - drive low and high<br>2) Output open drain with pull high - drive low<br>3) Output open drain without pull high - drive high |

Remarks: [1] An example of item c above is shown below in red rectangle.

```
        ORG     0013H                   ;CA INT         IE.2
        SJMP    INITIAL                 ;
        ORG     001BH                   ;T1 OVER ONLY   IE.3
        LJMP    TIMER1_INTERRUPT        ;
        ORG     0023H                   ;SERIAL         IE.4
        SJMP    INITIAL                 ;
        ORG     002BH                   ;T2 INT         IE.5
        LJMP    TIMER2_INTERRUPT        ;
        ORG     0033H                   ;RESERVED       IE.6
        SJMP    INITIAL                 ;
        ORG     003BH                   ;SERIAL2        IE.7
        SJMP    INITIAL                 ;
;=============================================================================
        ORG     0040H                   ;
;=============================================================================

INITIAL:
        ; delay 100ms to let power stable
        ; in case the firmware write data flash at the beginning
        CALL    DELAY

        ; The following instructions to be added
        MOV     R0,#0FH
        MOV     A,#00111111B
        MOVX    @R0,A

INITIAL_1:
;       Customer Application start here
;       .
;       .
;       .
        JMP     INITIAL_1
```

[2] This item applies only to 28-pin package:

[3] The flow is illustrated below:




[4] An example on item b is shown below inside the red rectangle:

```
        ORG     0013H                   ;CA INT         IE.2
        SJMP    INITIAL                 ;
        ORG     001BH                   ;T1 OVER ONLY   IE.3
        LJMP    TIMER1_INTERRUPT        ;
        ORG     0023H                   ;SERIAL         IE.4
        SJMP    INITIAL                 ;
        ORG     002BH                   ;T2 INT         IE.5
        LJMP    TIMER2_INTERRUPT        ;
        ORG     0033H                   ;RESERVED       IE.6
        SJMP    INITIAL                 ;
        ORG     003BH                   ;SERIAL2        IE.7
        SJMP    INITIAL                 ;
;-------------------------------------------------------------------------------
        ORG     0040H                   ;
;-------------------------------------------------------------------------------

INITIAL:
        ; delay 100ms to let power stable
        ; in case the firmware write data flash at the beginning
        CALL    DELAY

        ; The following instructions to be added
        MOV     R0,#0FH
        MOV     A,#00111111B
        MOVX    @R0,A

INITIAL_1:
;       Customer Application start here
;       .
;       .
;       .
        JMP     INITIAL_1
```

[5] It depends on the application.

## 3.2 DEEMAX Emulator

### 3.2.1 Limitation on DC6688FSA

When using emulator, there are some instructions, listed below, that the machine cycle is not identical to that used in our chip.

| | Dragonchip | DEEMAX emulator |
|---|---|---|
| Mnemonic | Machine cycle | Machine cycle |
| RET | 4 | 2 |
| RETI | 4 | 2 |
| JMP @A+DPTR | 3 | 2 |
| MOVC A,@A+DPTR | 3 | 2 |
| MOVC A,@A+PC | 3 | 2 |
| INC DPTR | 3 | 2 |

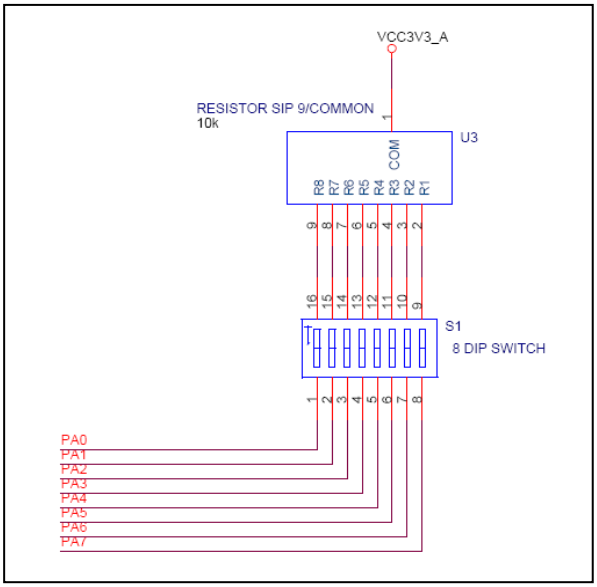
Additional limitation when using “DC6688EMT V4”:
1. Power down mode is not implemented in ICE, don’t use it, otherwise, undetermined result occurs.
2. No ISP select pin on CON4 in the ICE
3. No XOUT pin on CON4 in the ICE
4. No watchdog (basic timer)[1]
5. No backup mode
6. No ISP programming
7. Only operated at 3.3V power
8. No access to ‘T1_PCNTA’ register
9. No T2 output on PC2 by setting bit ‘T2OE’ in ‘T2MOD’ register

Remarks:
[1] Do NOT enable watchdog in emulator mode, otherwise, abnormal operation on emulator may occur.

Additional limitation when using developer III board ver3.0:
1. Power down mode is not implemented in ICE, don’t use it, otherwise, undetermined result occurs.
2. No ISP select pin on CON4 in the ICE
3. No XOUT pin on CON4 in the ICE
4. No pull-up resistors in the ICE's port A, B and C[1]
5. No watchdog (basic timer)[2]
6. No backup mode
7. No ISP programming
8. No UART1
9. Only operated at 3.3V power
10. No access to ‘T1_PCNTA’ register
11. No T2 output on PC2 by setting bit ‘T2OE’ in ‘T2MOD’ register

Remarks:

[1] Item 4 above is suggested to add an optional pull-up resistor (as shown below) on customer's target board.

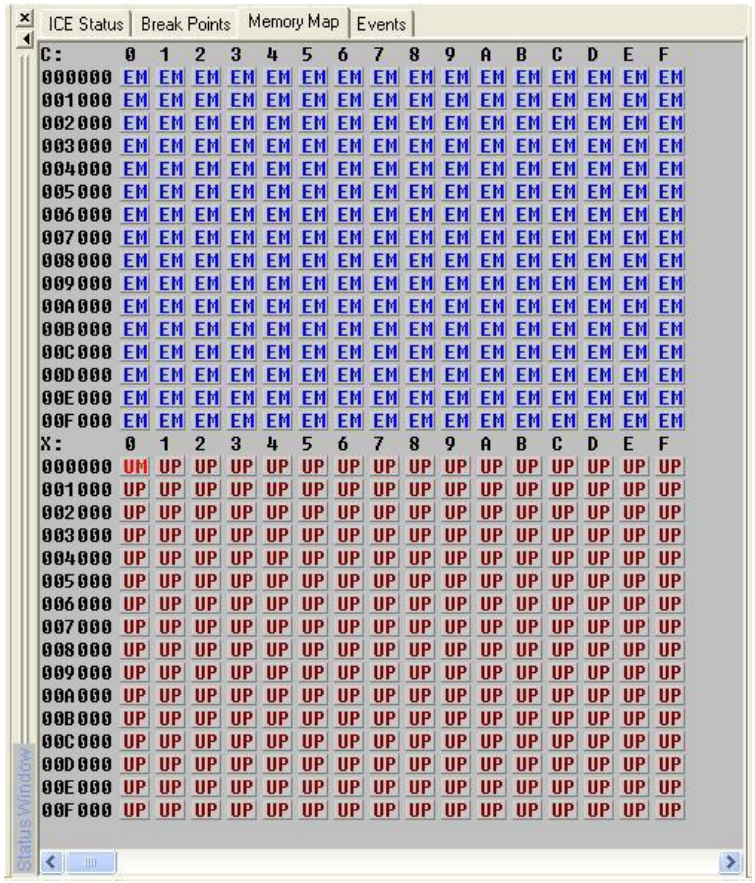



The Dip switch serves as an enable switch when connecting to emulator, and a disable switch when connecting to IC.

[2] Do NOT enable watchdog in emulator mode, otherwise, abnormal operation on emulator may occur.

### 3.2.2 Environment Setting

In the DEEMAX emulator's software environment, the 'Memory map' in 'Status window' has to modify as shown below:




| C: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 000000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 001000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 002000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 003000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 004000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 005000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 006000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 007000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 008000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 009000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00A000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00B000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00C000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00D000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00E000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00F000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| X: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
| 000000 | UM | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 001000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 002000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 003000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 004000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 005000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 006000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 007000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 008000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 009000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00A000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00B000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00C000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00D000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00E000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00F000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |

### 3.2.3 Additional Information

1. When download the code (program flash and/or data flash) to the IC, the whole flash memory will be erased, however, it is not the case in the emulator. The content in Data Flash memory always retained, whenever downloading the program to the emulator.

## 3.3 In-System Programming

During programming, the flash memory has the following steps involved:
1. whole flash memory erase for security reason
2. write data flash and program flash memory
3. read back data flash and program flash memory, and then verify byte by byte
4. lock program/data flash if required

# 4 DC6688FSB

## 4.1 Firmware

| | Incident | Item | Description |
|---|---|---|---|
| A | Abnormal software operation | 'PC' register in XFR | Make sure bit 6 is '0' |
| B | Abnormal PC1 state | PCCON[1,0] in XFR | 01 = prohibited |
| C | High stop mode Idd | Add instructions[1] | Add the following instructions to the beginning of the program<br>MOV R0,#0FH<br>MOV A,#00111111B<br>MOVX @R0,A |
| D | High stop mode Idd | Add instructions[2] | Both bit 1 ~ 0 of 'PCCONL' register should be set to 1 before entering stop mode |
| E | Abnormal Serial Communication | UART 0 and UART 1 Receive | Both UART might have chance to get wrong data. The only solution is to send back the received data to confirm whether IC received correctly [3].<br><br>An alternative solution is to use software UART on receive part. Example refers to Application Note 024. |
| F | Abnormal Timer operation | Timer 0 and Timer 1 setting | The following condition at the same time happened is invalid:<br>i) Timer 1 set to mode 1, and<br>ii) Timer 0 set to mode 2 |
| G | Abnormal Data flash content | Initialization Data for Data Flash memory | To avoid using firmware to initialize Data Flash memory during in production line, it is recommended to also download the image to data flash memory by SL or SPI. |
| H | Abnormal Data flash content | Add 100ms delay at the beginning of the program | This applies to case where factory do initialization on data flash memory in production line by firmware itself. In view of the unstable power supply in production line found at the instant of insertion, add this delay at the beginning of the program to let the power stable before starting to run the firmware in production line.[4] |
| J | Abnormal software operation | Initialization for SRAM | This is a must as on power up, the contents in the SRAM are undetermined. |
| K | Software halt | Counter A reset | Counter A should be reset after exit from stop mode |

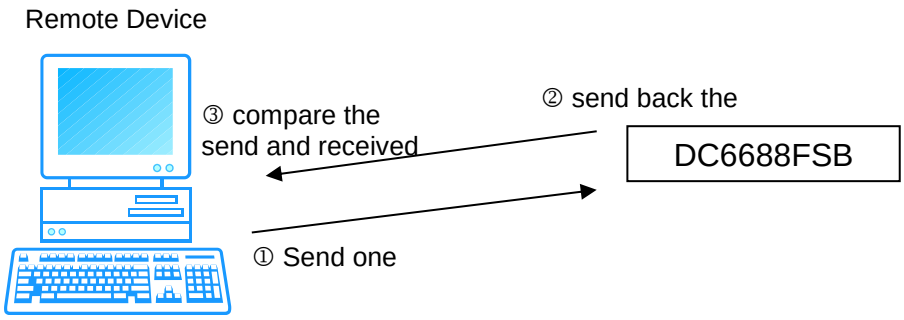
| | | | |
|---|---|---|---|
| **L** | Abnormal Interrupt | Bit 'IT0' in SFR | IT0 should set to 0 for PA interrupt for proper operation |
| **M** | Abnormal Interrupt [5] | Accessing Data Flash memory | 1) CPU suspends operation<br>2) only one interrupt is seen by CPU no matter how many interrupts from a peripheral<br>3) any interrupt will be delayed to serve by CPU until completion of accessing data flash<br>4) Disable watchdog before accessing data flash |
| **N** | High stop mode Idd | I/O port configuration in stop mode | In order to meet the Stop mode Idd current as stated in specification, the I/O port should be carefully configured. The following is prohibited:<br>1) Input without pull high - drive low and high<br>2) Output open drain with pull high - drive low<br>3) Output open drain without pull high - drive high |
| **O** | Abnormal Serial Communication | UART 0 and UART 1 Transmit | Don't modify Port B (byte addressable only) state when data is sending out on TXD0 / TXD1 pins. Otherwise, data will be corrupted. |

Remarks: [1] An example of item c above is shown below in red rectangle.

```
        ORG     0013H                   ;CA INT         IE.2
        SJMP    INITIAL                 ;
        ORG     001BH                   ;T1 OVER ONLY   IE.3
        LJMP    TIMER1_INTERRUPT        ;
        ORG     0023H                   ;SERIAL         IE.4
        SJMP    INITIAL                 ;
        ORG     002BH                   ;T2 INT         IE.5
        LJMP    TIMER2_INTERRUPT        ;
        ORG     0033H                   ;RESERVED       IE.6
        SJMP    INITIAL                 ;
        ORG     003BH                   ;SERIAL2        IE.7
        SJMP    INITIAL                 ;
;=======================================================================
        ORG     0040H                   ;
;=======================================================================

INITIAL:
        ; delay 100ms to let power stable
        ; in case the firmware write data flash at the beginning
        CALL    DELAY

        ; The following instructions to be added
        MOV     R0,#0FH
        MOV     A,#00111111B
        MOVX    @R0,A

INITIAL_1:
;       Customer Application start here
;       .
;       .
;       .
        JMP     INITIAL_1
```

[2] This item applies only to 28-pin package:

[3] The flow is illustrated below:




[4] An example on item b is shown below inside the red rectangle:

```
        ORG     0013H                   ;CA INT         IE.2
        SJMP    INITIAL                 ;
        ORG     001BH                   ;T1 OVER ONLY   IE.3
        LJMP    TIMER1_INTERRUPT        ;
        ORG     0023H                   ;SERIAL         IE.4
        SJMP    INITIAL                 ;
        ORG     002BH                   ;T2 INT         IE.5
        LJMP    TIMER2_INTERRUPT        ;
        ORG     0033H                   ;RESERVED       IE.6
        SJMP    INITIAL                 ;
        ORG     003BH                   ;SERIAL2        IE.7
        SJMP    INITIAL                 ;
;-------------------------------------------------------------------------------
        ORG     0040H                   ;
;-------------------------------------------------------------------------------

INITIAL:
        ; delay 100ms to let power stable
        ; in case the firmware write data flash at the beginning
        CALL    DELAY

        ; The following instructions to be added
        MOV     R0,#0FH
        MOV     A,#00111111B
        MOVX    @R0,A

INITIAL_1:
;       Customer Application start here
;       .
;       .
;       .
        JMP     INITIAL_1
```

[5] It depends on the application.

## 4.2 DEEMAX Emulator

### 4.2.1 Limitation on DC6688FSB

When using emulator, there are some instructions, listed below, that the machine cycle is not identical to that used in our chip.

| | Dragonchip | DEEMAX emulator |
|---|---|---|
| Mnemonic | Machine cycle | Machine cycle |
| RET | 4 | 2 |
| RETI | 4 | 2 |
| JMP @A+DPTR | 3 | 2 |
| MOVC A,@A+DPTR | 3 | 2 |
| MOVC A,@A+PC | 3 | 2 |
| INC DPTR | 3 | 2 |

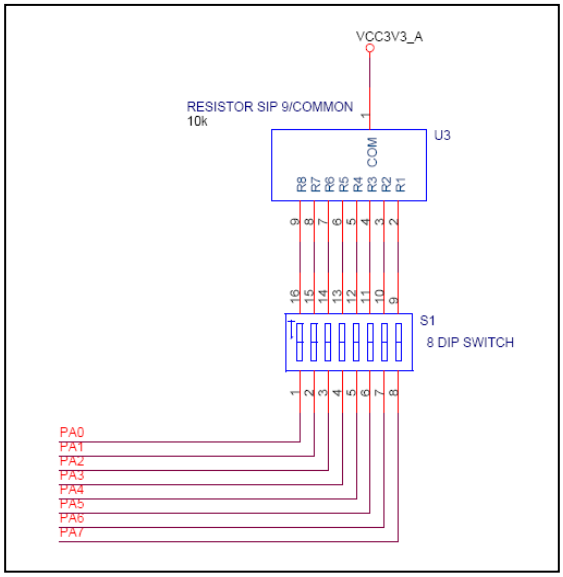
Additional limitation when using “DC6688EMT V4”:
1. Power down mode is not implemented in ICE, don’t use it, otherwise, undetermined result occurs.
2. No ISP select pin on CON4 in the ICE
3. No XOUT pin on CON4 in the ICE
4. No watchdog (basic timer)[1]
5. No backup mode
6. No ISP programming
7. Only operated at 3.3V power
8. No access to ‘T1_PCNTA’ register
9. No T2 output on PC2 by setting bit ‘T2OE’ in ‘T2MOD’ register

Remarks:
[1] Do NOT enable watchdog in emulator mode, otherwise, abnormal operation on emulator may occur.

Additional limitation when using developer III board ver3.0:
1. Power down mode is not implemented in ICE, don’t use it, otherwise, undetermined result occurs.
2. No ISP select pin on CON4 in the ICE
3. No XOUT pin on CON4 in the ICE
4. No pull-up resistors in the ICE's port A, B and C[1]
5. No watchdog (basic timer)[2]
6. No backup mode
7. No ISP programming
8. No UART1
9. Only operated at 3.3V power
10. No access to ‘T1_PCNTA’ register
11. No T2 output on PC2 by setting bit ‘T2OE’ in ‘T2MOD’ register

Remarks:

[1] Item 4 above is suggested to add an optional pull-up resistor (as shown below) on customer's target board.

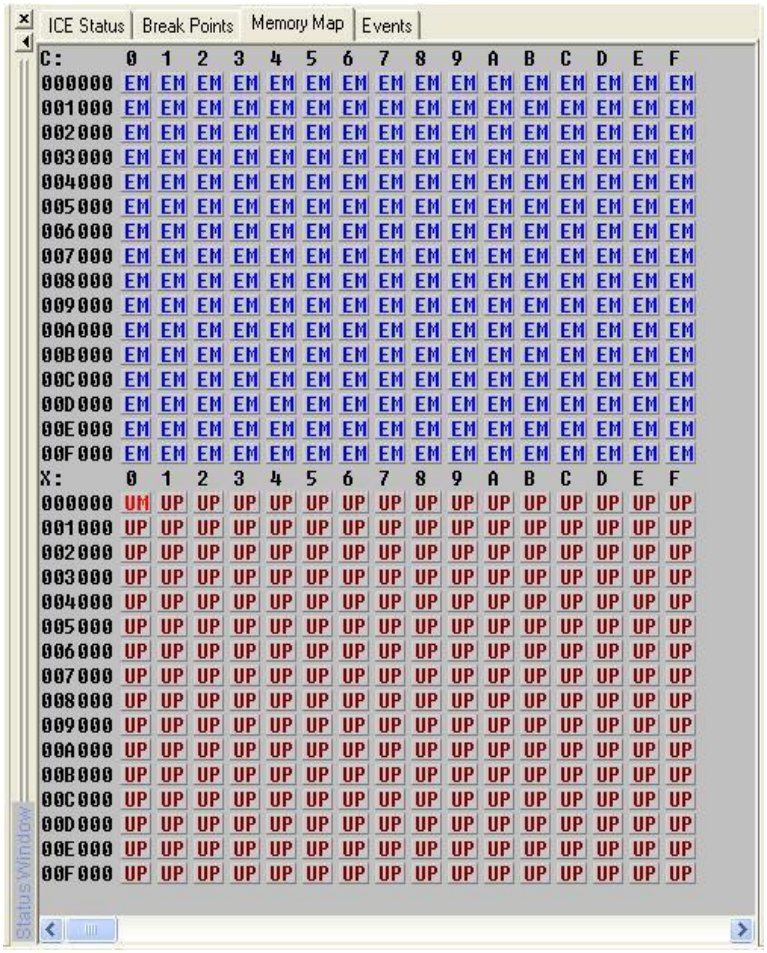



The Dip switch serves as an enable switch when connecting to emulator, and a disable switch when connecting to IC.

[2] Do NOT enable watchdog in emulator mode, otherwise, abnormal operation on emulator may occur.

### 4.2.2 Environment Setting

In the DEEMAX emulator's software environment, the 'Memory map' in 'Status window' has to modify as shown below:




C:

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 000000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 001000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 002000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 003000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 004000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 005000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 006000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 007000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 008000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 009000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00A000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00B000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00C000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00D000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00E000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00F000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |

X:

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 000000 | UM | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 001000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 002000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 003000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 004000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 005000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 006000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 007000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 008000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 009000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00A000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00B000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00C000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00D000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00E000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00F000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |

### 4.2.3 Additional Information

1. When download the code (program flash and/or data flash) to the IC, the whole flash memory will be erased, however, it is not the case in the emulator. The content in Data Flash memory always retained, whenever downloading the program to the emulator.

## 4.3 In-System Programming

### 4.3.1 Programming Steps

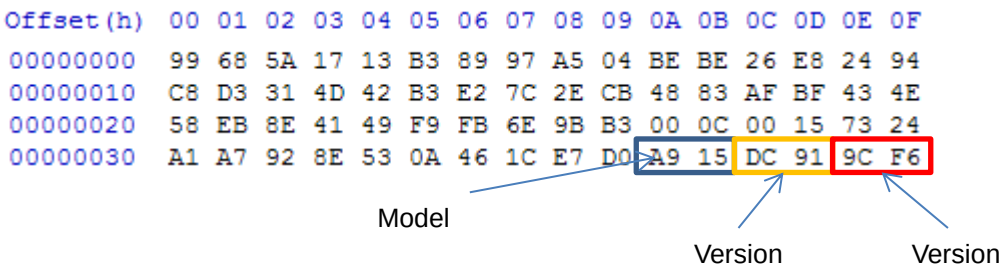
During programming, the flash memory has the following steps involved:

1. whole flash memory erase for security reason
2. write data flash and program flash memory
3. read back data flash and program flash memory, and then verify byte by byte
4. lock program/data flash if required

### 4.3.2 Model/Version/Checksum location

During programming, it is allowed to download customer's model / version / checksum to identify the firmware version. Their location is located in EEPROM.

1) DC6688F4SB / F8SB / F16SB / F24SB / F30SB



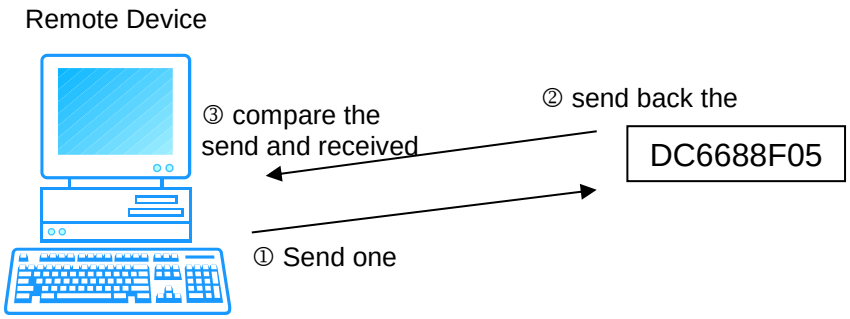
# 5 DC6688F05S

## 5.1 Firmware

| | Incident | Item | Description |
|---|---|---|---|
| A | Abnormal software operation | 'PC' register in XFR | Make sure bit 6 is '0' |
| B | Abnormal PC1 state | PCCON[1,0] in XFR | 01 = prohibited |
| C | Abnormal Serial Communication | UART 0 Receive | It might have chance to get wrong data. The only solution is to send back the received data to confirm whether IC received correctly [1].<br><br>An alternative solution is to use software UART on receive part. Example refers to Application Note 024. |
| D | Abnormal Timer operation | Timer 0 and Timer 1 setting | The following condition at the same time happened is invalid:<br>i) Timer 1 set to mode 1, and<br>ii) Timer 0 set to mode 2 |
| E | Abnormal software operation | Stack pointer | Only increment to 0x3F<br>Make sure no overflow after 0x3F |
| F | Abnormal software operation | Initialization for SRAM | This is a must as on power up, the contents in the SRAM are undetermined. |
| G | Software halt | Counter A reset | Counter A should be reset after exit from stop mode |
| H | Abnormal Interrupt | Bit 'IT0' in SFR | IT0 should set to 0 for PB/PC interrupt for proper operation |
| J | Abnormal software operation | Program Flash size 0x000 ~ 0x1F0 | Any code writing into this region will not be executed by CPU.<br>For Remote control application, to avoid it, use Rom library "ROMCode#1 - v1.2", which is located in folder "AppNote019/ROMCode#1/v1.2". When compiling your source code, if exceeding beyond 0x1F0, Keil compiler will not permit to compile it. |
| K | High stop mode Idd | I/O port configuration in stop mode | In order to meet the Stop mode Idd current as stated in specification, the I/O port should be carefully configured. The following is prohibited:<br>1) Input without pull high - drive low and high<br>2) Output open drain with pull high - drive low<br>3) Output open drain without pull high - drive high |
| L | Abnormal software | Program flash area | Actual area: |

| | | | |
|---|---|---|---|
| | operation | | 0x000 ~ 0x1EF<br>any code writing into the region(0x1F0 ~ 0x1FF) will not be executed by CPU.[2] |

[1] The flow is illustrated below:




[2] For Remote control application, to avoid it, use Rom library “ROMCode#1 - v1.2”, which is located in folder “AppNote019/ROMCode#1/v1.2”. When compiling your source code, if exceeding beyond 0x1F0, Keil compiler will not permit to compile it.

For other application where ROM library is not used, refers to “AppNote021”.

## 5.2 DEEMAX Emulator

### 5.2.1 Limitation on DC6688F05S

When using emulator, there are some instructions, listed below, that the machine cycle is not identical to that used in our chip.

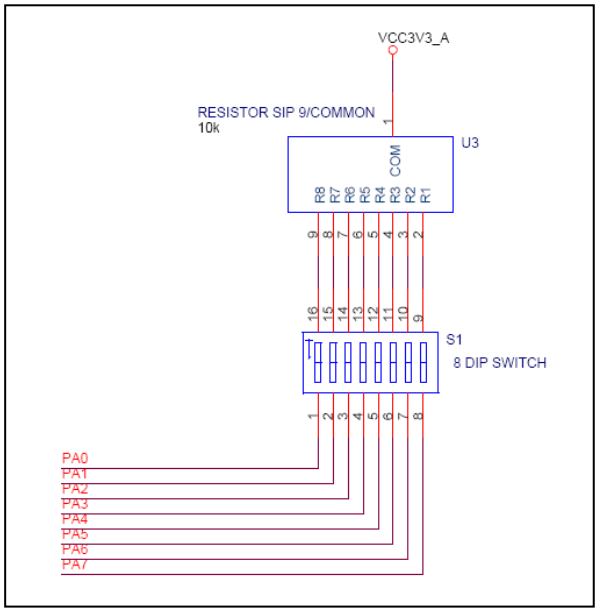
| | Dragonchip | DEEMAX emulator |
|---|---|---|
| Mnemonic | Machine cycle | Machine cycle |
| RET | 4 | 2 |
| RETI | 4 | 2 |
| JMP @A+DPTR | 3 | 2 |
| MOVC A,@A+DPTR | 3 | 2 |
| MOVC A,@A+PC | 3 | 2 |
| INC DPTR | 3 | 2 |

Additional limitation when using “DC6688EMT V4”:
1. Power down mode is not implemented in ICE, don’t use it, otherwise, undetermined result occurs.
2. No ISP select pin on CON4 in the ICE
3. No XOUT pin on CON4 in the ICE
4. No watchdog (basic timer)[1]
5. No backup mode
6. No ISP programming
7. Only operated at 3.3V power
8. No access to ‘T1_PCNTA’ register
9. No T2 output on PC2 by setting bit ‘T2OE’ in ‘T2MOD’ register

Remarks:
[1] Do NOT enable watchdog in emulator mode, otherwise, abnormal operation on emulator may occur.

Additional limitation when using developer III board ver3.0:
1. Power down mode is not implemented in ICE, don’t use it, otherwise, undetermined result occurs.
2. No ISP select pin on CON4 in the ICE
3. No XOUT pin on CON4 in the ICE
4. No pull-up resistors in the ICE's port A, B and C[1]
5. No watchdog (basic timer)[2]
6. No backup mode
7. No ISP programming
8. Only operated at 3.3V power
9. No access to ‘T1_PCNTA’ register
10. No T2 output on PC2 by setting bit ‘T2OE’ in ‘T2MOD’ register

Remarks:
[1] Item 4 above is suggested to add an optional pull-up resistor (as shown below) on customer’s target board.

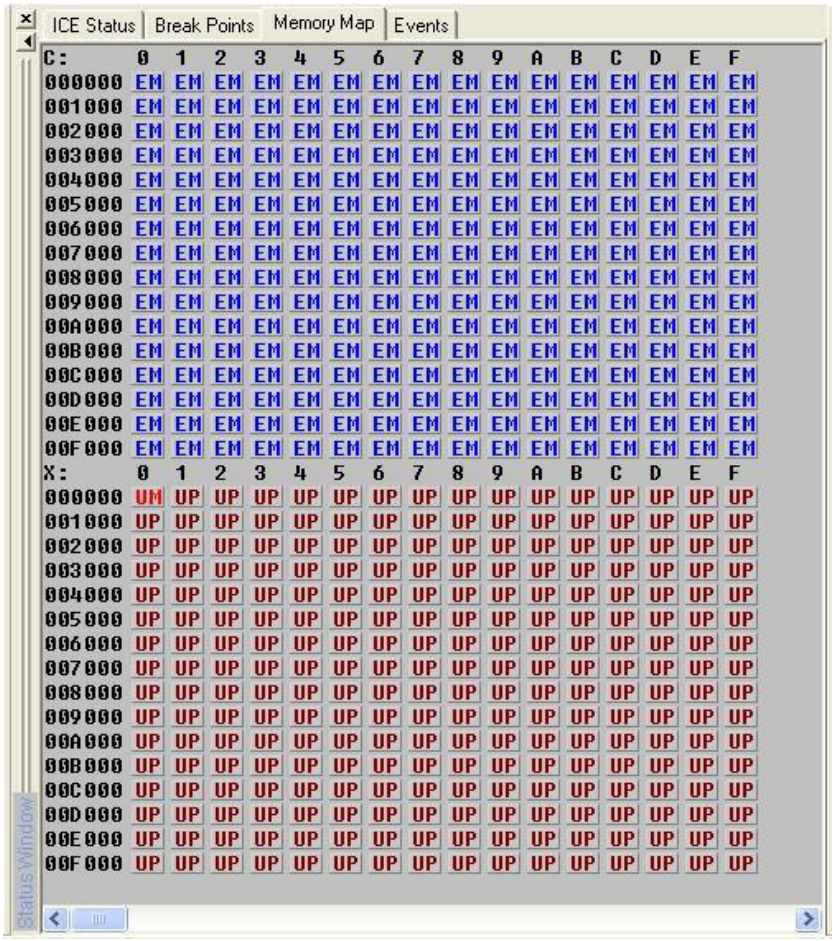



The Dip switch serves as an enable switch when connecting to emulator, and a disable switch when connecting to IC.

[2] Do NOT enable watchdog in emulator mode, otherwise, abnormal operation on emulator may occur.

### 5.2.2 Environment Setting

In the DEEMAX emulator's software environment, the 'Memory map' in 'Status window' has to modify as shown below:




C:

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 000000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 001000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 002000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 003000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 004000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 005000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 006000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 007000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 008000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 009000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00A000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00B000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00C000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00D000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00E000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00F000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |

X:

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 000000 | UM | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 001000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 002000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 003000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 004000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 005000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 006000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 007000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 008000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 009000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00A000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00B000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00C000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00D000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00E000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00F000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |

## 5.3 In-System Programming

During programming, the flash memory has the following steps involved:

1. whole flash memory erase for security reason
2. write program flash memory
3. write customer information(Model/Version/Checksum)
4. read back program flash memory and customer information, and then verify byte by byte
5. lock program flash if required

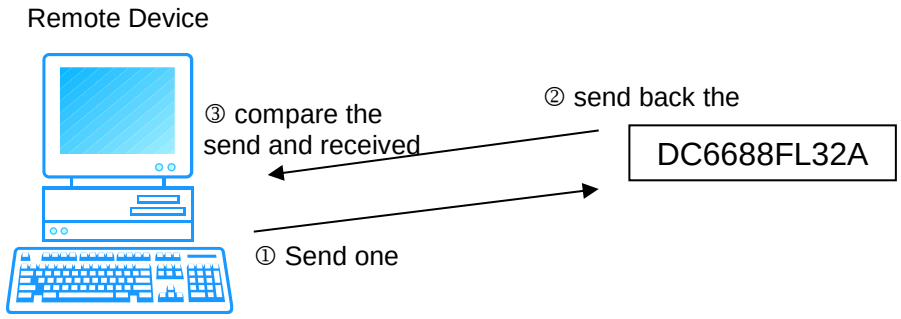
# 6 DC6688FL32A

## 6.1 Firmware

| | Incident | Item | Description |
|---|---|---|---|
| A | Abnormal Serial Communication | UART 0 and UART 1 Receive | Both UART might have chance to get wrong data. The only solution is to send back the received data to confirm whether IC received correctly [1].<br><br>An alternative solution is to use software UART on receive part. Example refers to Application Note 024. |
| B | Abnormal Timer operation | Timer 0 and Timer 1 setting | The following condition at the same time happened is invalid:<br>i) Timer 1 set to mode 1, and<br>ii) Timer 0 set to mode 2 |
| C | Abnormal Data flash content | Initialization Data for Data Flash memory | To avoid using firmware to initialize Data Flash memory during in production line, it is recommended to also download the image to data flash memory by SL or SPI. |
| D | Abnormal Data flash content | Add 100ms delay at the beginning of the program | This applies to case where factory do initialization on data flash memory in production line by firmware itself. In view of the unstable power supply in production line found at the instant of insertion, add this delay at the beginning of the program to let the power stable before starting to run the firmware in production line.[2] |
| E | Abnormal software operation | Initialization for SRAM | This is a must as on power up, the contents in the SRAM are undetermined. |
| F | Software halt | Counter A reset | Counter A should be reset after exit from stop mode |
| G | Abnormal Interrupt | Bit 'IT0' in SFR | IT0 should set to 0 for PA interrupt for proper operation |
| H | Abnormal Interrupt [3] | Accessing Data Flash memory | 1) CPU suspends operation<br>2) only one interrupt is seen by CPU no matter how many interrupts from a peripheral<br>3) any interrupt will be delayed to serve by CPU until completion of accessing data flash<br>4) Disable watchdog before accessing data flash |
| J | High stop mode Idd | I/O port configuration in stop mode | In order to meet the Stop mode Idd current as stated in specification, the I/O port should be carefully configured. The following is prohibited:<br>1) Input without pull high - drive low and high<br>2) Output open drain with pull high - drive low |

| | | | |
|---|---|---|---|
| | | | 3) Output open drain without pull high - drive high |
| K | Abnormal T24 capture | T24 capture reset | To avoid insufficient time for capture reset, T24_CON1 should be configured in the following order in each capture reset::<br>1) T24_CON1 = 0xA0; first, bit 5 set to 1<br>2) T24_CON1 = 0x8E; then, enable pointer x capture as you want, where x = A, B or C. In this example, enable all. |
| L | Abnormal Serial Communication | UART 0 and UART 1 Transmit | Don't modify Port B (byte addressable only) state when data is sending out on TXD0 / TXD1 pins. Otherwise, data will be corrupted. |

Remarks:
[1] The flow is illustrated below:




[2] An example on item b is shown below inside the red rectangle:

```
        ORG     0013H                   ;CA INT         IE.2
        SJMP    INITIAL                 ;
        ORG     001BH                   ;T1 OVER ONLY   IE.3
        LJMP    TIMER1_INTERRUPT        ;
        ORG     0023H                   ;SERIAL         IE.4
        SJMP    INITIAL                 ;
        ORG     002BH                   ;T2 INT         IE.5
        LJMP    TIMER2_INTERRUPT        ;
        ORG     0033H                   ;RESERVED       IE.6
        SJMP    INITIAL                 ;
        ORG     003BH                   ;SERIAL2        IE.7
        SJMP    INITIAL                 ;
;-------------------------------------------------------------------------------
        ORG     0040H                   ;
;-------------------------------------------------------------------------------

INITIAL:
        ; delay 100ms to let power stable
        ; in case the firmware write data flash at the beginning
        CALL    DELAY

        ; The following instructions to be added
        MOV     R0,#0FH
        MOV     A,#00111111B
        MOVX    @R0,A

INITIAL_1:
;       Customer Application start here
;       .
;       .
;       .
        JMP     INITIAL_1
```

[3] It depends on the application.

## 6.2 DEEMAX Emulator

### 6.2.1 Limitation on DC6688FL32A

When using emulator, there are some instructions, listed below, that the machine cycle is not identical to that used in our chip.

| | Dragonchip | DEEMAX emulator |
|---|---|---|
| Mnemonic | Machine cycle | Machine cycle |
| RET | 4 | 2 |
| RETI | 4 | 2 |
| JMP @A+DPTR | 3 | 2 |
| MOVC A,@A+DPTR | 3 | 2 |
| MOVC A,@A+PC | 3 | 2 |
| INC DPTR | 3 | 2 |

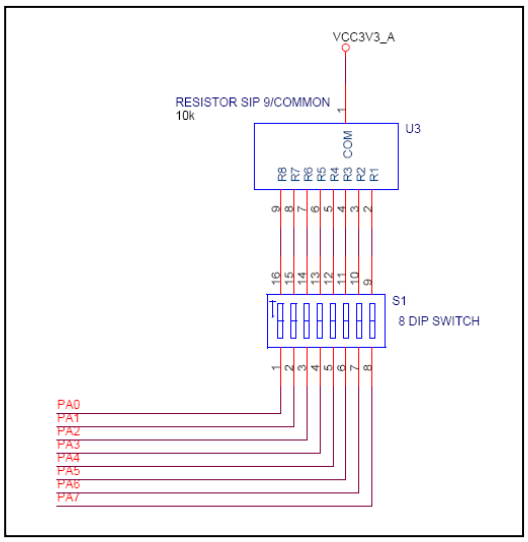
Additional limitation when using “DC6688EMT V4”:
1. Power down mode is not implemented in ICE, don’t use it, otherwise, undetermined result occurs.
2. No ISP select pin on CON4 in the ICE
3. No XOUT pin on CON4 in the ICE
4. No watchdog (basic timer)[1]
5. No backup mode
6. No ISP programming
7. Only operated at 3.3V power
8. No access to ‘T1_PCNTA’ register
9. No T2 output on PC2 by setting bit ‘T2OE’ in ‘T2MOD’ register

Remarks:
[1] Do NOT enable watchdog in emulator mode, otherwise, abnormal operation on emulator may occur.

Additional limitation when using “Developer III board ver3.0”:
1. Power down mode is not implemented in ICE, don’t use it, otherwise, undetermined result occurs.
2. No ISP select pin on CON4 in the ICE
3. No XOUT pin on CON4 in the ICE
4. No pull-up resistors in the ICE's port A, B and C[1]
5. No watchdog (basic timer)[2]
6. No backup mode
7. No ISP programming
8. No UART1
9. Only operated at 3.3V power
10. No access to ‘T1_PCNTA’ register
11. No T2 output on PC2 by setting bit ‘T2OE’ in ‘T2MOD’ register

Remarks:

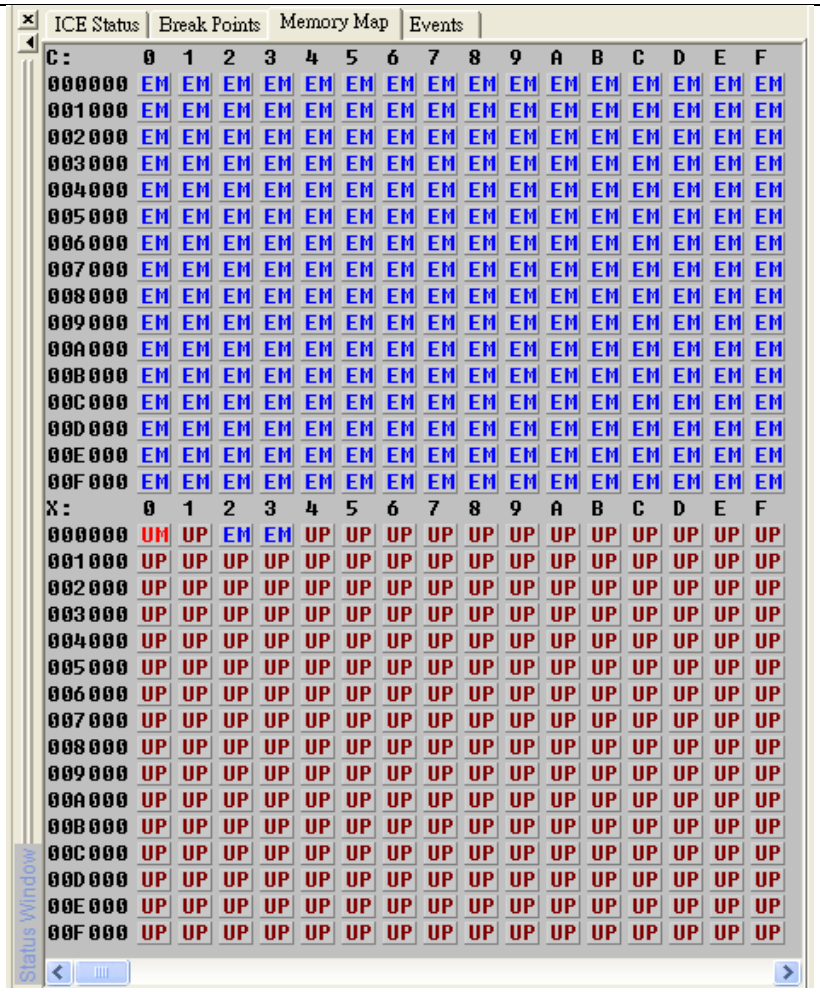
[1] Item 4 above is suggested to add an optional pull-up resistor (as shown below) on customer's target board.




The Dip switch serves as an enable switch when connecting to emulator, and a disable switch when connecting to IC.
[2] Do NOT enable watchdog in emulator mode, otherwise, abnormal operation on emulator may occur.

### 6.2.2 Environment Setting

In the DEEMAX emulator's software environment, the 'Memory map' in 'Status window' has to modify as shown below:



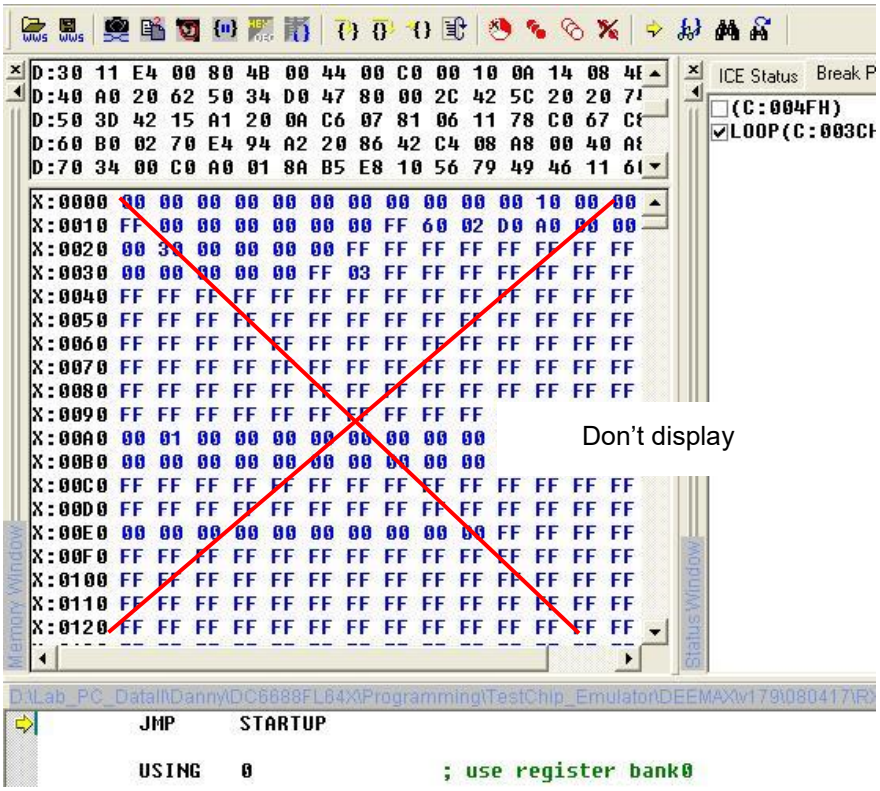

| C: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 000000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 001000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 002000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 003000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 004000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 005000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 006000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 007000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 008000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 009000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00A000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00B000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00C000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00D000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00E000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00F000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| X: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
| 000000 | UM | UP | EM | EM | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 001000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 002000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 003000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 004000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 005000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 006000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 007000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 008000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 009000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00A000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00B000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00C000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00D000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00E000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00F000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |

### 6.2.3 Precaution when debugging

When debugging the program in emulator, in the period of using the following register:

1) “RACTL” = 0x01 or 0x03
2) “RBCTL” = 0x01 or 0x03

Display of the memory region X:0000H ~ 01FFH is prohibited as shown below:




Since when setting “RACTL” to 0x01, whenever reading/writing “RDBA” the pointer “RAPTH/RAPTL” will automatically increment by 1. The emulator itself, in order to update the “Memory window”, will also read the whole XFR table one time whenever the program stops running. This reading will disturb the pointer “RAPTH/RAPTL” and the program to get the wrong data. This principle also applies to “RBCTL”

Display of the memory region X:0200H ~ FFFFH is allowed.

### 6.2.4 Additional Information

1. The ‘IRI’ pin on ‘J1’ connector cannot be used to capture the IR signal. Since the line from IR receiver on target board to ‘IRI’ pin on ‘J1’ connector is rather long, it results in higher noise and affect the receive performance. Therefore, the IR receiver is already made on the “Developer III board ver3.0”.

2. When download the code (program flash and/or data flash) to the IC, the whole flash memory will be erased, however, it is not the case in the emulator. The content in Data Flash memory always retained, whenever downloading the program to the emulator.

## 6.3 In-System Programming

During programming, the flash memory has the following steps involved:
1. whole flash memory erase for security reason
2. write data flash and program flash memory
3. write customer information(Model/Version/Checksum)
4. read back data flash, program flash memory, and customer information, and then verify byte by byte
5. lock program/data flash if required

# 7 DC6688FSX

## 7.1 Firmware

| | Incident | Item | Description |
|---|---|---|---|
| A | Abnormal PC1 state | PCCON[1,0] in XFR | 01 = prohibited |
| B | Abnormal Timer operation | Timer 0 and Timer 1 setting | The following condition at the same time happened is invalid:<br>i) Timer 1 set to mode 1, and<br>ii) Timer 0 set to mode 2 |
| C | Abnormal Data flash content | Initialization Data for Data Flash memory | To avoid using firmware to initialize Data Flash memory during in production line, it is recommended to also download the image to data flash memory by SL or SPI. |
| D | Abnormal Data flash content | Add 100ms delay at the beginning of the program | This applies to case where factory do initialization on data flash memory in production line by firmware itself. In view of the unstable power supply in production line found at the instant of insertion, add this delay at the beginning of the program to let the power stable before starting to run the firmware in production line.[1] |
| E | Abnormal software operation | Initialization for SRAM | This is a must as on power up, the contents in the SRAM are undetermined. |
| F | Software halt | Counter A reset | Counter A should be reset after exit from stop mode |
| G | Abnormal Interrupt | Bit 'IT0' in SFR | IT0 should set to 0 for PA interrupt for proper operation |
| H | Abnormal Interrupt [2] | Accessing Data Flash memory | 1) CPU suspends operation<br>2) only one interrupt is seen by CPU no matter how many interrupts from a peripheral<br>3) any interrupt will be delayed to serve by CPU until completion of accessing data flash<br>4) Disable watchdog before accessing data flash |
| J | High stop mode Idd | I/O port configuration in stop mode | In order to meet the Stop mode Idd current as stated in specification, the I/O port should be carefully configured. The following is prohibited:<br>1) Input without pull high - drive low and high<br>2) Output open drain with pull high - drive low<br>3) Output open drain without pull high - drive high |
| K | Incomplete waveform produced | Counter A operation | Avoid clearing the CAS bit when Counter A is reloading with CADATAL value. Therefore, the clock |

| | | | |
|---|---|---|---|
| | by Counter A even when Stop Carrier Mode (SCM) bit is set to 1 | | cycle between set/clear CAS bit cannot be completely divided by the period in clock cycle of the Counter A waveform. [3] |
| L | Abnormal Serial Communication | UART 0 and UART 1 Transmit | Don't modify Port B (byte addressable only) state when data is sending out on TXD0 / TXD1 pins. Otherwise, data will be corrupted. |

Remarks:

[1] An example on item b is shown below inside the red rectangle:

```
        ORG     0013H                   ;CA INT         IE.2
        SJMP    INITIAL                 ;
        ORG     001BH                   ;T1 OVER ONLY   IE.3
        LJMP    TIMER1_INTERRUPT        ;
        ORG     0023H                   ;SERIAL         IE.4
        SJMP    INITIAL                 ;
        ORG     002BH                   ;T2 INT         IE.5
        LJMP    TIMER2_INTERRUPT        ;
        ORG     0033H                   ;RESERVED       IE.6
        SJMP    INITIAL                 ;
        ORG     003BH                   ;SERIAL2        IE.7
        SJMP    INITIAL                 ;
;-------------------------------------------------------------------------------
        ORG     0040H                   ;
;-------------------------------------------------------------------------------

INITIAL:
        ; delay 100ms to let power stable
        ; in case the firmware write data flash at the beginning
        CALL    DELAY

        ; The following instructions to be added
        MOV     R0,#0FH
        MOV     A,#00111111B
        MOVX    @R0,A

INITIAL_1:
;       Customer Application start here
;       .
;       .
;       .
        JMP     INITIAL_1
```

[2] It depends on the application.

[3] Example:

*Counter A Setting*

```
Clock Divider = 1
CADATAL Value = 0
CADATAH Value = 0
```

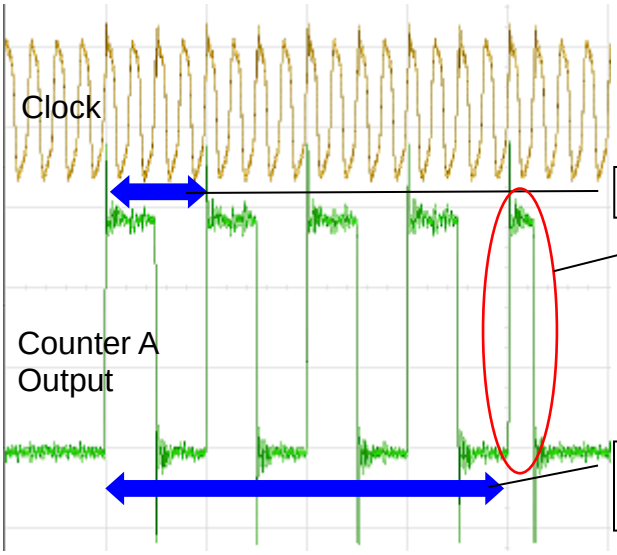
Period of the Counter A waveform in clock cycle:
(CADATAL + CADATAH + 4) × Clock Selection
= (0 + 0 + 4) × 1 = 4

*Error Code Snippet*

```
MOVR0, #CACON
MOVA, #0x07
MOVR1, #0x03
MOVX   @R0, A
MOVA, R1
MOVX   @R0, A
```

Clock Cycle between the set/clear of CAS bit:
Instruction cycle × 4
= ([MOV A,Rn] + [MOVX @Rn,A]) × 4
= (1 + 3) × 4
= 16

*Error Waveform*




Counter A waveform period (4

(Clock between set/clear CAS) modulo (Clock of Counter A waveform) = 16 mod 4 = 0
Hence, partial waveform

Clock cycle between set/clear the CAS bit (16 Clock)

[4] Un-bonded I/O port (orange) is shown below. It should not be configured as "floating".

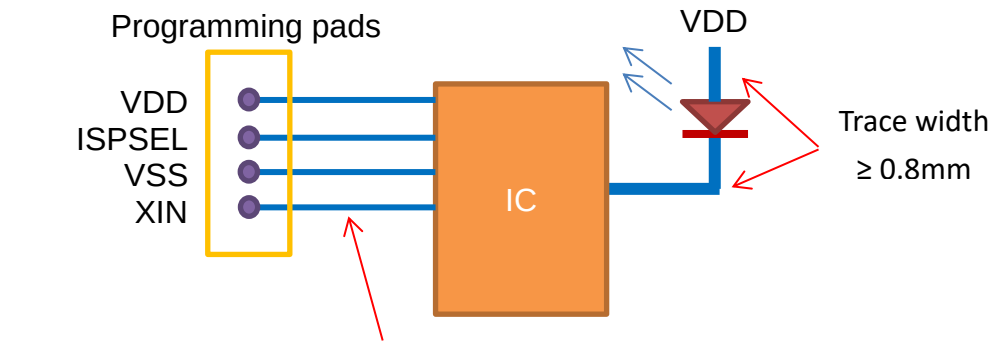
| SOP24 | SOP28 | Pin Name |
|---|---|---|
| 5 | 6 | PA0/INTA |
| 6 | 7 | PA1/INTA/MISO |
| 7 | 8 | PA2/INTA |
| 8 | 9 | PA3/INTA |
| 9 | 10 | PA4/INTA |
| 10 | 11 | PA5/INTA |
| 11 | 12 | PA6/INTA |
| 12 | 13 | PA7/INTA |
| 13 | 16 | PB0/INTB/RxD0/ISPSCK/SDI |
| 14 | 17 | PB1/INTB/TxD0/MOSI/SDO |
| 15 | 18 | PB2/INTB/RxD1/SDI |
| 16 | 19 | PB3/INTB/TxD1/SDO |
| 17 | 20 | PB4/INTB/SCK/SCL1 |
| 18 | 21 | PB5/INTB/PWM0/SDA1 |
| 19 | 22 | PB6/INTB/T2EX/T24EX/PWM1 |
| 20 | 23 | PB7/INTB/SDA0 |

| SOP24 | SOP28 | Pin Name |
|---|---|---|
| 21 | 24 | PC0/T0/ISPSS/SCL0 |
| 22 | 25 | PC1/REM/IRTX/T1/T24_OUT |
| 23 | 26 | PC2/T2/T24_CLK |
| - | 28 | PC3 |
| - | 14 | PC4 |
| - | 15 | PC5 |
| - | 1 | PD2 |

## 7.2 Hardware

### 7.2.1 PCB layout

Below are some guidelines for the layout.




## 7.3 DC6688EMT Emulator

### 7.3.1 Limitation on DC6688FSX

This applies only to the following version:
1. DC6688EMT-4T Rev2.x

1 When using emulator, the instruction execution time is slightly different from the IC, therefore software delay and processing time will be different. User needs re-adjustment.
2 No backup mode
3 Only operated at 3.3V power
4 Peripherals
  4.1 When the emulator is stopped in debugging platform, all the running peripherals (e.g. timer 2) will still keep running. Hence, the peripherals will be out of synchronization with the code instruction.

5 UART0 cannot use Timer 2 as baud-rate generator
6 No UART1

## 7.4 In-System Programming

### 7.4.1 Programming Steps

During programming, the flash memory has the following steps involved:
1. whole flash memory erase for security reason
2. write data flash and program flash memory
3. write customer information(Model/Version/Checksum)
4. read back data flash, program flash memory, and customer information, and then verify byte by byte
5. lock program/data flash if required

## 7.5 Production highlights in customer factory

It is recommended to review the following items before production in customer site:
1 The prevention and control of Electrostatic Discharge (ESD)

# 8 DC6688FLX

## 8.1 Firmware

| | Incident | Item | Description |
|---|---|---|---|
| A | Abnormal Timer operation | Timer 0 and Timer 1 setting | The following condition at the same time happened is invalid:<br>i) Timer 1 set to mode 1, and<br>ii) Timer 0 set to mode 2 |
| B | Abnormal Data flash content | Initialization Data for Data Flash memory | To avoid using firmware to initialize Data Flash memory during in production line, it is recommended to also download the image to data flash memory by SL or SPI. |
| C | Abnormal Data flash content | Add 100ms delay at the beginning of the program | This applies to case where factory do initialization on data flash memory in production line by firmware itself. In view of the unstable power supply in production line found at the instant of insertion, add this delay at the beginning of the program to let the power stable before starting to run the firmware in production line.[1] |
| D | Abnormal software operation | Initialization for SRAM | This is a must as on power up, the contents in the SRAM are undetermined. |
| E | Software halt | Counter A reset | Counter A should be reset after exit from stop mode |
| F | Abnormal Interrupt | Bit 'IT0' in SFR | IT0 should set to 0 for PA interrupt for proper operation |
| G | Abnormal Interrupt [2] | Accessing Data Flash memory | 1) CPU suspends operation<br>2) only one interrupt is seen by CPU no matter how many interrupts from a peripheral<br>3) any interrupt will be delayed to serve by CPU until completion of accessing data flash<br>4) Disable watchdog before accessing data flash |
| H | High stop mode Idd | I/O port configuration in stop mode | In order to meet the Stop mode Idd current as stated in specification, the I/O port should be carefully configured. The following is prohibited:<br>1) Input without pull high - drive low and high<br>2) Output open drain with pull high - drive low<br>3) Output open drain without pull high - drive high |
| J | Abnormal software operation | Set the program/data flash memory size[3] | Add the following instructions to the beginning of the program<br>MOV R0,#PFS<br>MOV A,#PROG_SIZE |

| | | | |
|---|---|---|---|
| | | | MOVX @R0,A |
| **K** | Abnormal read Data flash | Set RCPTH/RCPTL in Traditional Method | Before reading/writing Data flash with Traditional Method, RCPTH/RCPTL .should be defined.<br><br>There is an exception that the program migrates from DC6688FL32A to DC6688FLX without modification. It is because the default value of RCPTH/RCPTL are '0'. It means pointing to page 0 in expanded SRAM. |
| **L** | Abnormal T24 capture | T24 capture reset | To avoid insufficient time for capture reset, T24_CON1 should be configured in the following order in each capture reset::<br>1) T24_CON1 = 0xA0; first, bit 5 set to 1<br>2) T24_CON1 = 0x8E; then, enable pointer x capture as you want, where x = A, B or C. In this example, enable all. |
| **M** | Abnormal LVI | Bit 'LVI_STATUS' no reset | To reset the LVI function on power up, at the beginning of the program add 'read' instruction of the 'LVI' register.<br>MOV R0,#LVI<br>MOVX A,@R0 ; dummy read<br><br>MOVX A,@R0 ; second time read the bit 'LVI_STATUS'<br><br>The second time or onwards of reading will function properly. |
| **N** | Incomplete waveform produced by Counter A even when Stop Carrier Mode (SCM) bit is set to 1 | Counter A operation | Avoid clearing the CAS bit when Counter A is reloading with CADATAL value. Therefore, the clock cycle between set/clear CAS bit cannot be completely divided by the period in clock cycle of the Counter A waveform. [4] |
| **O** | Abnormal Serial Communication | UART 0 and UART 1 Transmit | Don't modify Port B (byte addressable only) state when data is sending out on TXD0 / TXD1 pins. Otherwise, data will be corrupted. |
| **P** | Abnormal Read Data flash memory to Expanded SRAM | Accessing Data Flash memory | In enhanced method (FAM = 1), if FPTRH2/H/L = 0x yyFF, where yy is odd number, no matter what value in TXZH/L, RCPTH/L,<br>Read data flash memory to Expanded SRAM will always transfer only the first byte. |

Remarks:
[1] An example on item D is shown below inside the red rectangle:

```
        ORG     0013H                   ;CA INT         IE.2
        SJMP    INITIAL                 ;
        ORG     001BH                   ;T1 OVER ONLY   IE.3
        LJMP    TIMER1_INTERRUPT        ;
        ORG     0023H                   ;SERIAL         IE.4
        SJMP    INITIAL                 ;
        ORG     002BH                   ;T2 INT         IE.5
        LJMP    TIMER2_INTERRUPT        ;
        ORG     0033H                   ;RESERVED       IE.6
        SJMP    INITIAL                 ;
        ORG     003BH                   ;SERIAL2        IE.7
        SJMP    INITIAL                 ;
;-------------------------------------------------------------------------------
        ORG     0040H                   ;
;-------------------------------------------------------------------------------

INITIAL:
        ; delay 100ms to let power stable
        ; in case the firmware write data flash at the beginning
        CALL    DELAY

        ; The following instructions to be added
        MOV     R0,#0FH
        MOV     A,#00111111B
        MOVX    @R0,A

INITIAL_1:
;       Customer Application start here
;       .
;       .
;       .
        JMP     INITIAL_1
```

[2] It depends on the application.

[3] For item B, if Program Flash memory required in DC6688FL64X/XE is 24KB, same as DC6688FL32A, then no need to set this ‘PFS’ register in program. In this case, the ratio of Program Flash/Data Flash memory in DC6688FL64X is 24KB/39.5KB after migration.

If the size of Program Flash memory is no longer 24KB, this ‘PFS’ register have to add at the beginning of the program as shown below.

```
        ORG     002BH

        RETI

;--------------------------------------
; Main Program
;--------------------------------------
STARTUP:
;---------------------------------
; define option here
; user is prohibited to modify it
#ifndef CHIP
        MOV     0A0H,#0
#endif
        ;----------------------------
        ; delay 100ms to let power stable
        ; in case the firmware write data flash at the beginning
        CALL    DELAY

        ;----------------------------
        ; initialize the program/data flash size at the very beginning of program,
        ; Warning: these instructions should be located in program memeory within 0x0000 ~ 0x0200
        MOV     R0,#PFS
        MOV     A,#PROG_SIZE
        MOVX    @R0,A
        ;----------------------------
```

[4] Example:

*Counter A Setting*

```
Clock Divider = 1
CADATAL Value = 0
CADATAH Value = 0
```

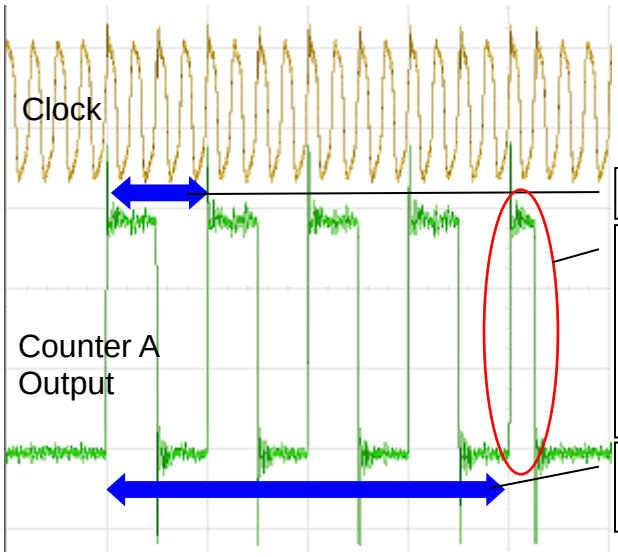
Period of the Counter A waveform in clock cycle:
(CADATAL + CADATAH + 4) × Clock Selection
= (0 + 0 + 4) × 1 = 4

*Error Code Snippet*

```
MOVR0, #CACON
MOVA, #0x07
MOVR1, #0x03
MOVX   @R0, A
MOVA, R1
MOVX   @R0, A
```

Clock Cycle between the set/clear of CAS bit:
Instruction cycle × 4
= ([MOV A,Rn] + [MOVX @Rn,A]) × 4
= (1 + 3) × 4
= 16

*Error Waveform*




Counter A waveform period (4

(Clock between set/clear CAS) modulo
(Clock of Counter A waveform)
= 16 mod 4 = 0
Hence, partial waveform generated

Clock cycle between set/clear the CAS bit (16 Clock)

## 8.2 DEEMAX Emulator

### 8.2.1 Limitation on DC6688FLX

When using emulator, there are some instructions, listed below, that the machine cycle is not identical to that used in our chip.

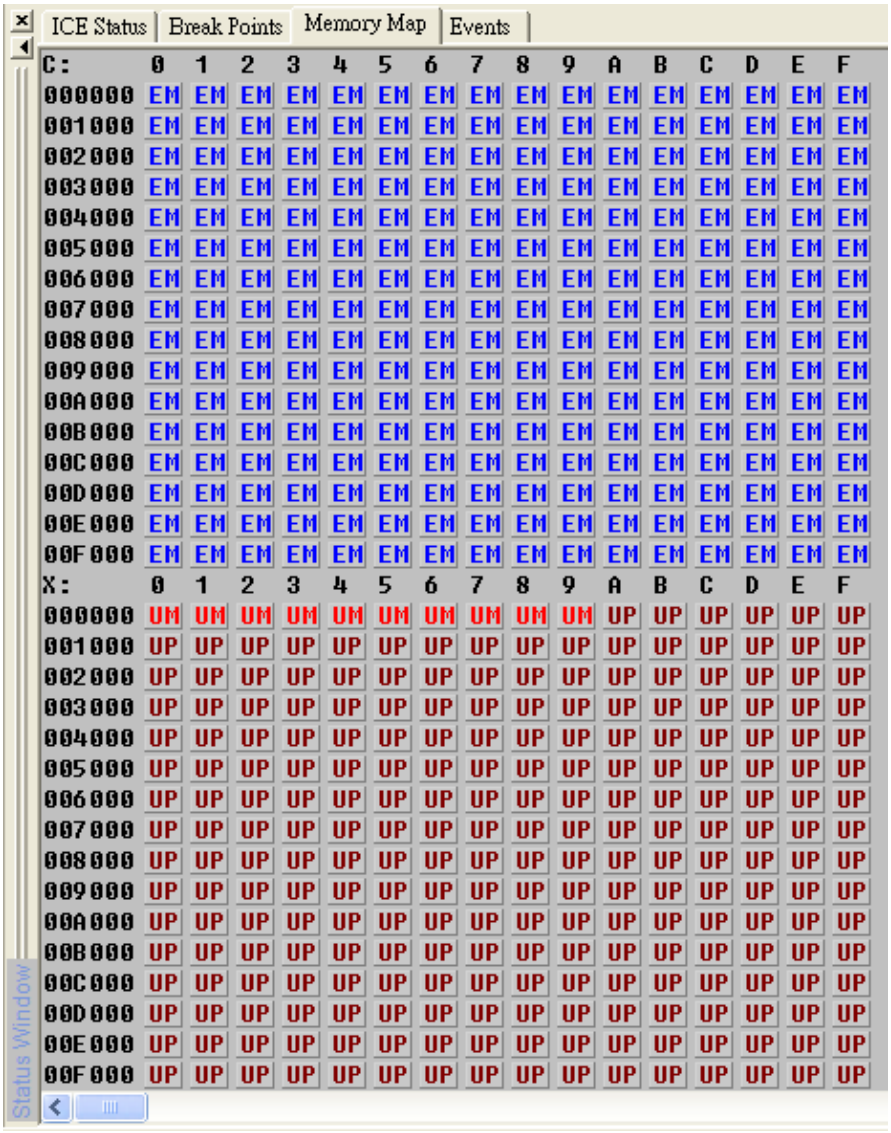
| | Dragonchip | DEEMAX emulator |
|---|---|---|
| Mnemonic | Machine cycle | Machine cycle |
| RET | 4 | 2 |
| RETI | 4 | 2 |
| JMP @A+DPTR | 3 | 2 |
| MOVC A,@A+DPTR | 3 | 2 |
| MOVC A,@A+PC | 3 | 2 |
| INC DPTR | 3 | 2 |

Additional limitation when using “DC6688EMT V4”:
1. Power down mode is not implemented in ICE, don’t use it, otherwise, undetermined result occurs.
2. No ISP select pin on CON4 in the ICE
3. No XOUT pin on CON4 in the ICE
4. No watchdog (basic timer)[1]
5. No backup mode
6. No ISP programming
7. Only operated at 3.3V power
8. No access to ‘T1_PCNTA’ register
9. No T2 output on PC2 by setting bit ‘T2OE’ in ‘T2MOD’ register

Remarks:
[1] Do NOT enable watchdog in emulator mode, otherwise, abnormal operation on emulator may occur.

### 8.2.2 Environment Setting

In the DEEMAX emulator's software environment, the 'Memory map' in 'Status window' has to modify as shown below:




C:

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 000000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 001000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 002000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 003000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 004000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 005000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 006000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 007000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 008000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 009000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00A000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00B000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00C000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00D000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00E000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00F000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |

X:

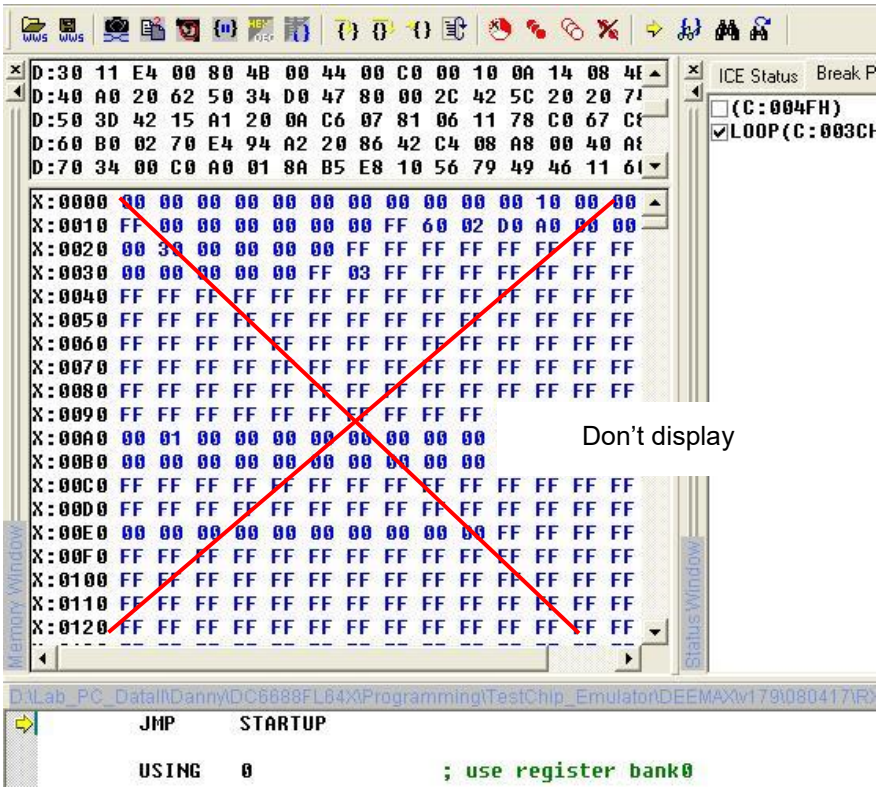
| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 000000 | UM | UM | UM | UM | UM | UM | UM | UM | UM | UM | UP | UP | UP | UP | UP | UP |
| 001000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 002000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 003000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 004000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 005000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 006000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 007000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 008000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 009000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00A000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00B000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00C000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00D000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00E000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00F000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |

### 8.2.3 Precaution when debugging

When debugging the program in emulator, in the period of using the following register:

1) “RACTL” = 0x01 or 0x03
2) “RBCTL” = 0x01 or 0x03

Display of the memory region X:0000H ~ 01FFH is prohibited as shown below:




Since when setting “RACTL” to 0x01, whenever reading/writing “RDBA” the pointer “RAPTH/RAPTL” will automatically increment by 1. The emulator itself, in order to update the “Memory window”, will also read the whole XFR table one time whenever the program stops running. This reading will disturb the pointer “RAPTH/RAPTL” and the program to get the wrong data. This principle also applies to “RBCTL”

Display of the memory region X:0200H ~ FFFFH is allowed.

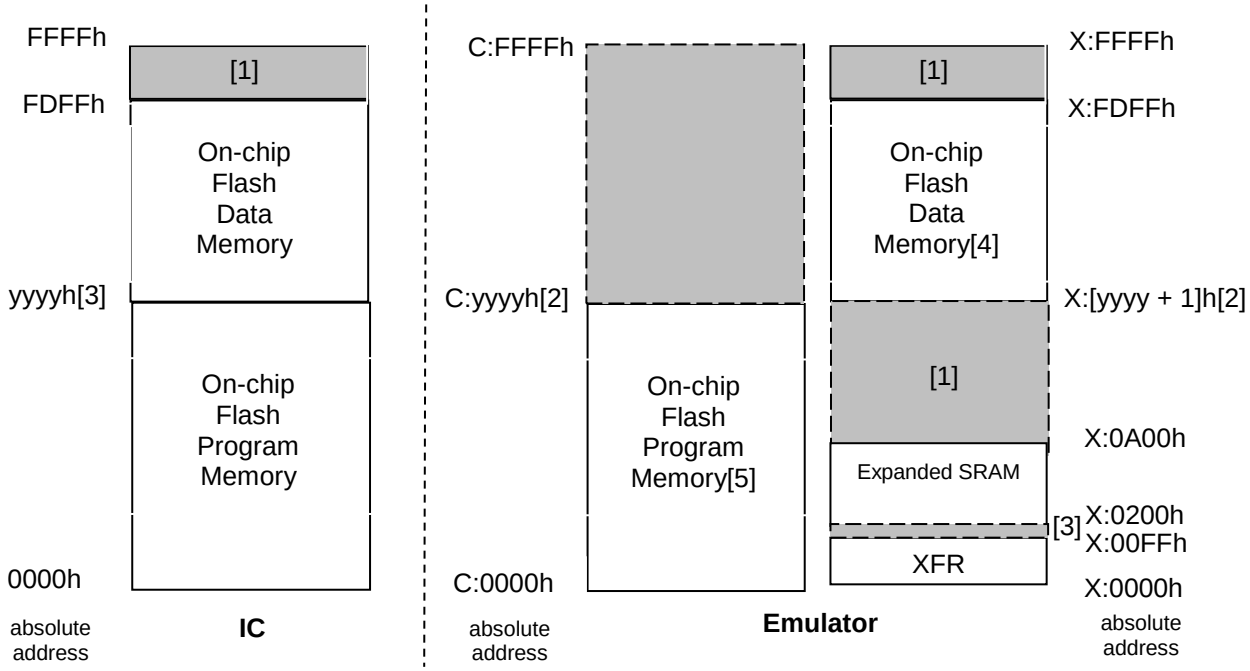
### 8.2.4 Additional Information

1) The 'IRI' pin on 'J1' connector cannot be used to capture the IR signal. Since the line from IR receiver on target board to 'IRI' pin on 'J1' connector is rather long, it results in higher noise and affect the receive performance. Therefore, the IR receiver is already made on the "DC6688EMT V4" or "DC6688EMT dvlp3-0807106".

2) When download the code (program flash and/or data flash) to the IC, the whole flash memory will be erased, however, it is not the case in the emulator. The content in Data Flash memory always retained, whenever downloading the program to the emulator.

3) The structure of memory in IC is different from that in emulator.
The flash memory can have two configurations as described in datasheet:

a) DC6688FL64X
Configuation A(5≤PFS<127):



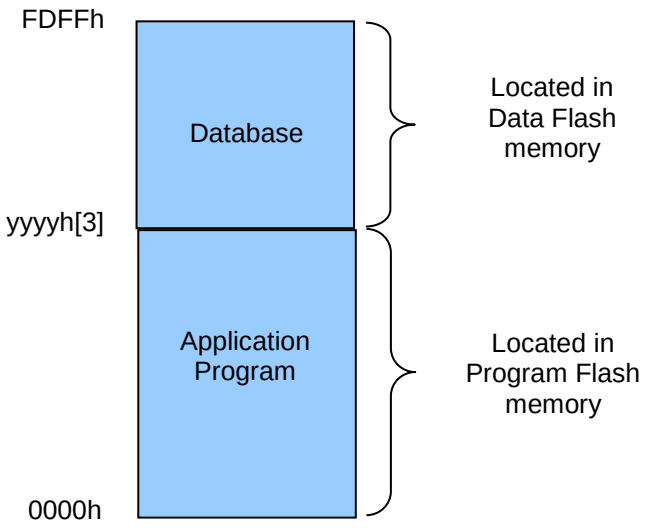

Remarks:
[1] Prohibited by user
[2] Detail refers to datasheet
[3] X:0100h ~ X:01FFh is prohibited by user
[4] To write this data flash memory, CPU can only use data flash memory's related XFR.

In emulator, the structure of DEEMAX emulator have two physical memories:

1) Program memory(refer as "C" region)
   It stores the program code
2) External memory(refer as "X" region)

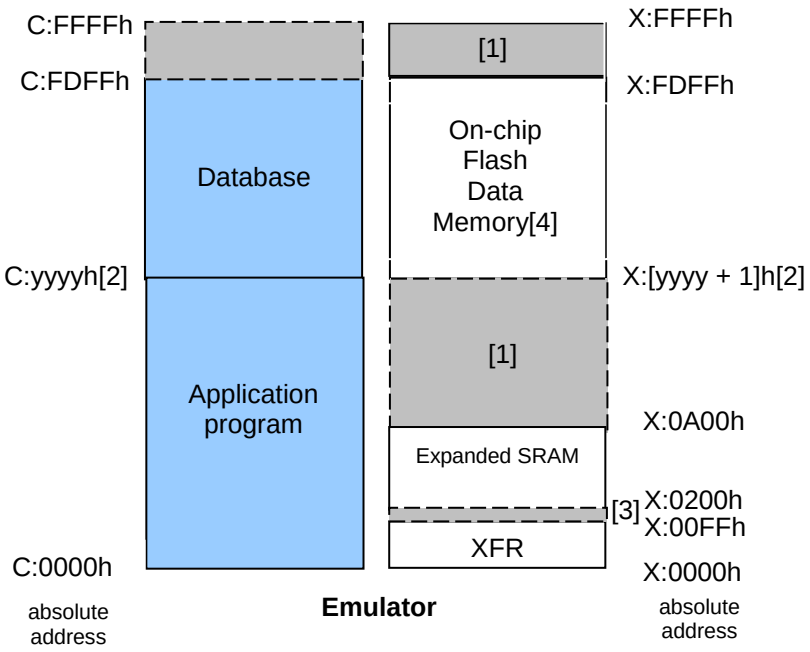
a) Mapped to the Flash Data memory.
The size depends on the value in ‘PFS’ register
It is read only.
b) Mapped to the XFR
c) Mapped to the expanded SRAM

For example, your source code distribution in the memory is as below:



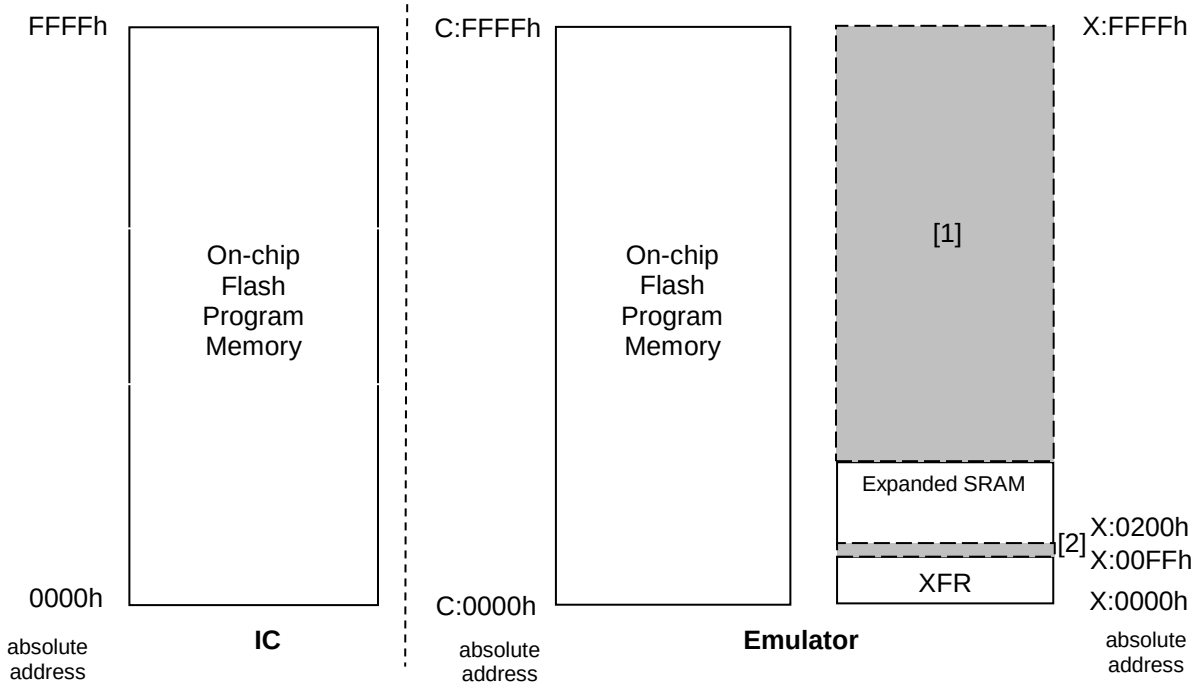

When compile the above code and download the hex file to IC, the structure will be kept the same as above diagram. The program at the beginning setting the register ‘PFS’ to re-structure the program/data flash ratio such that the application program can access the database correctly.

However, when the above code download to the emulator. The distribution is different like this below:




The database is not put in Flash Data memory. Therefore, database cannot be accessed correctly.

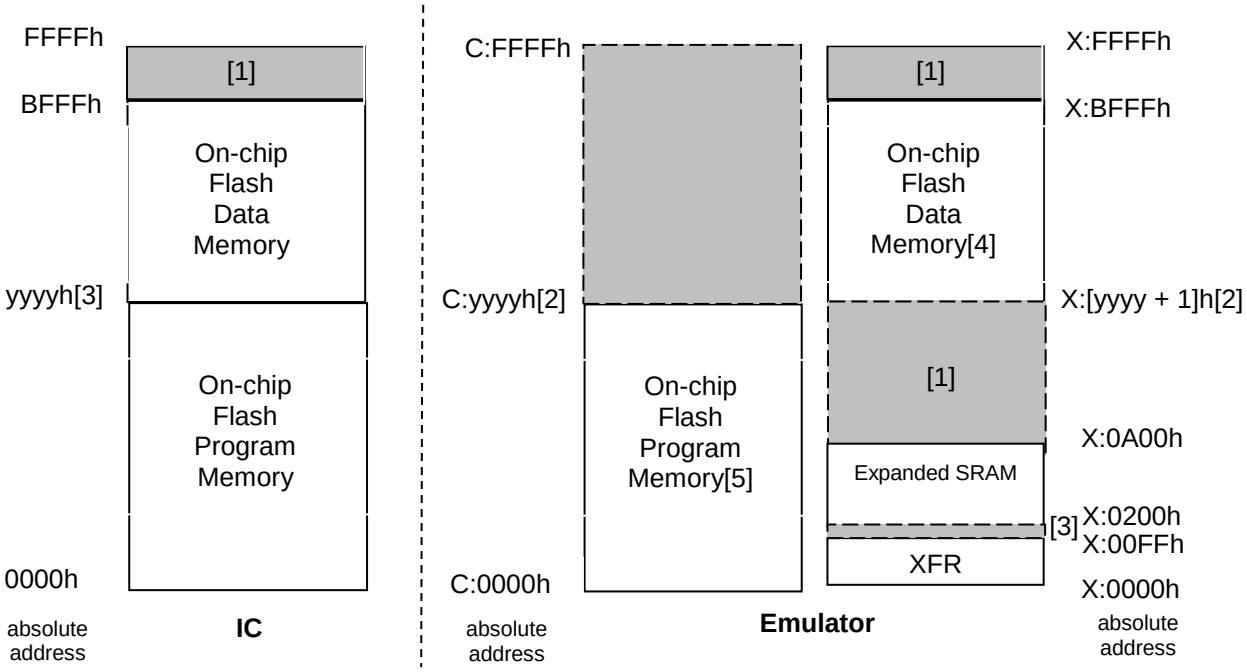
Configuation B(PFS = 127 , 128):




Remarks:
[1] Prohibited by user

[2] X:0100h ~ X:01FFh is prohibited by user

b) DC6688FL48X
Configuation A(5≤PFS<96):




Remarks:
[1] Prohibited by user
[2] Detail refers to datasheet
[3] X:0100h ~ X:01FFh is prohibited by user
[4] To write this data flash memory, CPU can only use data flash memory's related XFR.

In emulator, the structure of DEEMAX emulator have two physical memories:
1 Program memory(refer as "C" region)
  1.1 It stores the program code
2 External memory(refer as "X" region)
3 Mapped to the Flash Data memory.
  3.1 The size depends on the value in 'PFS' register
  3.2 It is read only.
4 Mapped to the XFR
5 Mapped to the expanded SRAM

## 8.3 In-System Programming

During programming, the flash memory has the following steps involved:

1. whole flash memory erase for security reason
2. write program flash and data flash memory
3. write customer information(Model/Version/Checksum)
4. read back program flash, data flash memory, and customer information, and then verify byte by byte
5. lock program/data flash if required

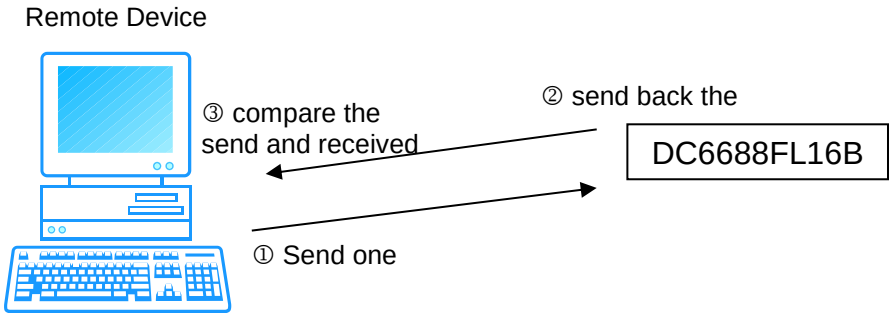
# 9 DC6688FL16B

## 9.1 Firmware

| | Incident | Item | Description |
|---|---|---|---|
| A | Abnormal Serial Communication | UART 0 and UART 1 Receive | Both UART might have chance to get wrong data. The only solution is to send back the received data to confirm whether IC received correctly [1].<br><br>An alternative solution is to use software UART on receive part. Example refers to Application Note 024. |
| B | Abnormal Timer operation | Timer 0 and Timer 1 setting | The following condition at the same time happened is invalid:<br>i) Timer 1 set to mode 1, and<br>ii) Timer 0 set to mode 2 |
| C | Abnormal Data flash content | Initialization Data for Data Flash memory | To avoid using firmware to initialize Data Flash memory during in production line, it is recommended to also download the image to data flash memory by SL or SPI. |
| D | Abnormal Data flash content | Add 100ms delay at the beginning of the program | This applies to case where factory do initialization on data flash memory in production line by firmware itself. In view of the unstable power supply in production line found at the instant of insertion, add this delay at the beginning of the program to let the power stable before starting to run the firmware in production line.[2] |
| E | Abnormal software operation | Initialization for SRAM | This is a must as on power up, the contents in the SRAM are undetermined. |
| F | Software halt | Counter A reset | Counter A should be reset after exit from stop mode |
| G | Abnormal Interrupt | Bit 'IT0' in SFR | IT0 should set to 0 for PA interrupt for proper operation |
| H | Abnormal Interrupt [3] | Accessing Data Flash memory | 1) CPU suspends operation<br>2) only one interrupt is seen by CPU no matter how many interrupts from a peripheral<br>3) any interrupt will be delayed to serve by CPU until completion of accessing data flash<br>4) Disable watchdog before accessing data flash |
| J | High stop mode Idd | I/O port configuration in stop mode | In order to meet the Stop mode Idd current as stated in specification, the I/O port should be carefully configured. The following is prohibited:<br>1) Input without pull high - drive low and high<br>2) Output open drain with pull high - drive low |

| | | | |
|---|---|---|---|
| | | | 3) Output open drain without pull high - drive high |
| **K** | High stop mode Idd | 'PCCONL' register setting | Bit 6 ~ 7 should write to '00' whenever writing to this register |
| **L** | High stop mode Idd | Add instructions | Add the following instructions to the beginning of the program<br>MOV R0,#0FH<br>MOV A,#00111111B<br>MOVX @R0,A |
| **M** | Abnormal T24 capture | T24 capture reset | To avoid insufficient time for capture reset, T24_CON1 should be configured in the following order in each capture reset::<br>1) T24_CON1 = 0xA0; first, bit 5 set to 1<br>2) T24_CON1 = 0x8E; then, enable pointer x capture as you want, where x = A, B or C. In this example, enable all. |
| **N** | Abnormal Serial Communication | UART 0 and UART 1 Transmit | Don't modify Port B (byte addressable only) state when data is sending out on TXD0 / TXD1 pins. Otherwise, data will be corrupted. |

Remarks:
[1] The flow is illustrated below:



[2] An example on item b is shown below inside the red rectangle:

```
        ORG     0013H                   ;CA INT         IE.2
        SJMP    INITIAL                 ;
        ORG     001BH                   ;T1 OVER ONLY   IE.3
        LJMP    TIMER1_INTERRUPT        ;
        ORG     0023H                   ;SERIAL         IE.4
        SJMP    INITIAL                 ;
        ORG     002BH                   ;T2 INT         IE.5
        LJMP    TIMER2_INTERRUPT        ;
        ORG     0033H                   ;RESERVED       IE.6
        SJMP    INITIAL                 ;
        ORG     003BH                   ;SERIAL2        IE.7
        SJMP    INITIAL                 ;
;===============================================================================
        ORG     0040H                   ;
;===============================================================================

INITIAL:
        ; delay 100ms to let power stable
        ; in case the firmware write data flash at the beginning
        CALL    DELAY

        ; The following instructions to be added
        MOV     R0,#0FH
        MOV     A,#00111111B
        MOVX    @R0,A

INITIAL_1:
;       Customer Application start here
;       .
;       .
;       .
        JMP     INITIAL_1
```

[3] It depends on the application.

## 9.2 DEEMAX Emulator

### 9.2.1 Limitation on DC6688FL16B

When using emulator, there are some instructions, listed below, that the machine cycle is not identical to that used in our chip.

| | Dragonchip | DEEMAX emulator |
|---|---|---|
| Mnemonic | Machine cycle | Machine cycle |
| RET | 4 | 2 |
| RETI | 4 | 2 |
| JMP @A+DPTR | 3 | 2 |
| MOVC A,@A+DPTR | 3 | 2 |
| MOVC A,@A+PC | 3 | 2 |
| INC DPTR | 3 | 2 |

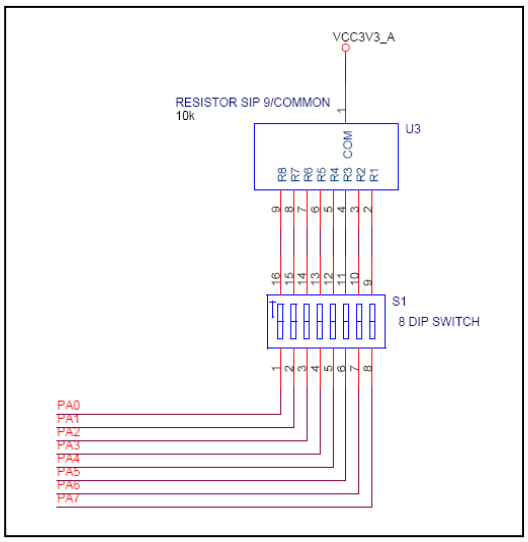
Additional limitation when using “DC6688EMT V4”:
1. Power down mode is not implemented in ICE, don’t use it, otherwise, undetermined result occurs.
2. No ISP select pin on CON4 in the ICE
3. No XOUT pin on CON4 in the ICE
4. No watchdog (basic timer)[1]
5. No backup mode
6. No ISP programming
7. Only operated at 3.3V power
8. No access to ‘T1_PCNTA’ register
9. No T2 output on PC2 by setting bit ‘T2OE’ in ‘T2MOD’ register

Remarks:
[1] Do NOT enable watchdog in emulator mode, otherwise, abnormal operation on emulator may occur.

Additional limitation when using “Developer III board ver3.0”:
1. Power down mode is not implemented in ICE, don’t use it, otherwise, undetermined result occurs.
2. No ISP select pin on CON4 in the ICE
3. No XOUT pin on CON4 in the ICE
4. No pull-up resistors in the ICE's port A, B and C[1]
5. No watchdog (basic timer)[2]
6. No backup mode
7. No ISP programming
8. No UART1
9. Only operated at 3.3V power
10. No access to ‘T1_PCNTA’ register
11. No T2 output on PC2 by setting bit ‘T2OE’ in ‘T2MOD’ register

Remarks:

[1] Item 4 above is suggested to add an optional pull-up resistor (as shown below) on customer's target board.

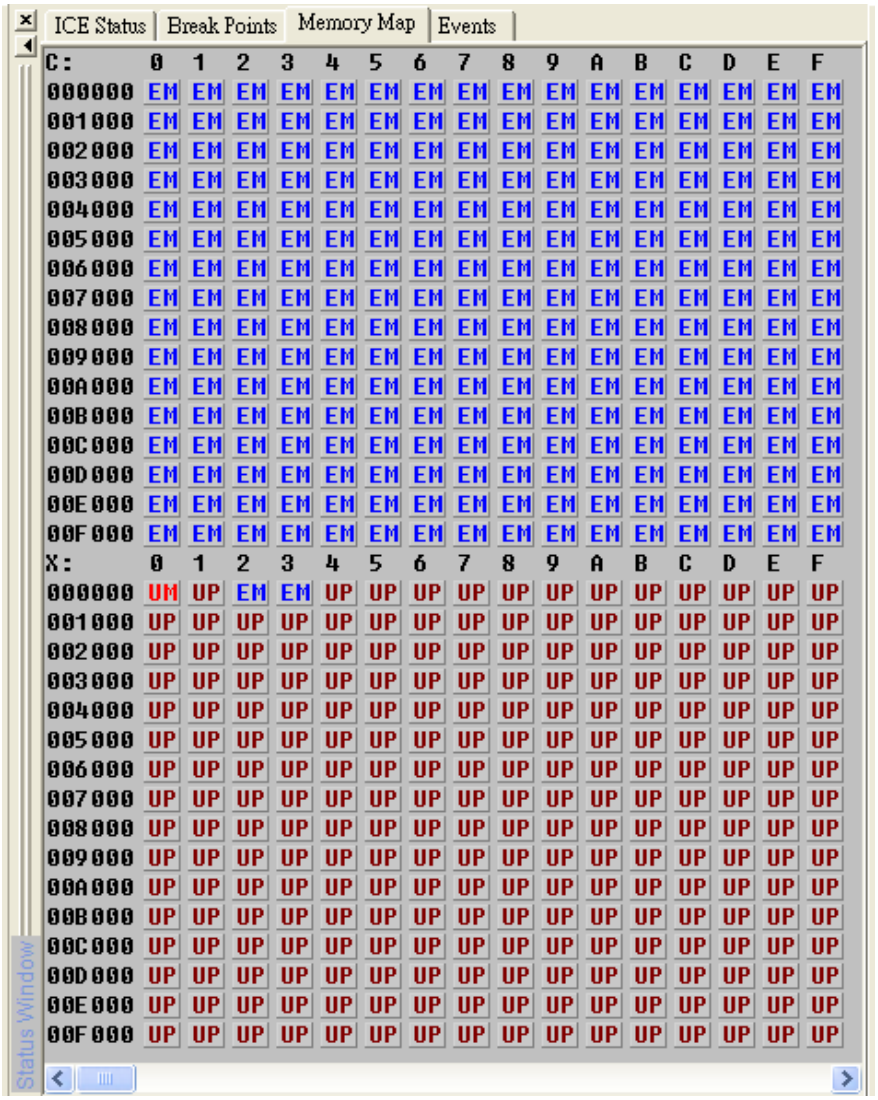



The Dip switch serves as an enable switch when connecting to emulator, and a disable switch when connecting to IC.

[2] Do NOT enable watchdog in emulator mode, otherwise, abnormal operation on emulator may occur.

### 9.2.2 Environment Setting

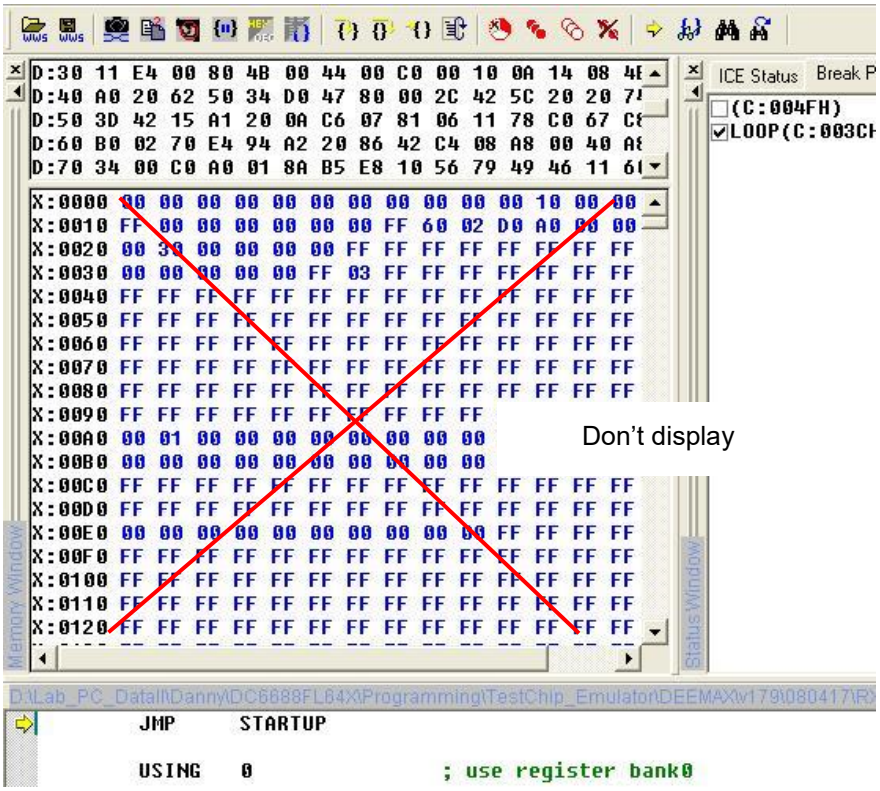
In the DEEMAX emulator's software environment, the 'Memory map' in 'Status window' has to modify as shown below:




| C: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 000000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 001000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 002000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 003000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 004000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 005000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 006000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 007000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 008000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 009000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00A000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00B000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00C000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00D000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00E000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| 00F000 | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM | EM |
| X: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
| 000000 | UM | UP | EM | EM | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 001000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 002000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 003000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 004000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 005000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 006000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 007000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 008000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 009000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00A000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00B000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00C000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00D000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00E000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |
| 00F000 | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP | UP |

### 9.2.3 Precaution when debugging

When debugging the program in emulator, in the period of using the following register:
1) “RACTL” = 0x01 or 0x03
2) “RBCTL” = 0x01 or 0x03
Display of the memory region X:0000H ~ 01FFH is prohibited as shown below:




Since when setting “RACTL” to 0x01, whenever reading/writing “RDBA” the pointer “RAPTH/RAPTL” will automatically increment by 1. The emulator itself, in order to update the “Memory window”, will also read the whole XFR table one time whenever the program stops running. This reading will disturb the pointer “RAPTH/RAPTL” and the program to get the wrong data. This principle also applies to “RBCTL”

Display of the memory region X:0200H ~ FFFFH is allowed.

### 9.2.4 Additional Information

1. The ‘IRI’ pin on ‘J1’ connector cannot be used to capture the IR signal. Since the line from IR receiver on target board to ‘IRI’ pin on ‘J1’ connector is rather long, it results in higher noise and affect the receive performance. Therefore, the IR receiver is already made on the “Developer III board ver3.0”.

2. When download the code (program flash and/or data flash) to the IC, the whole flash memory will be erased, however, it is not the case in the emulator. The content in Data Flash memory always retained, whenever downloading the program to the emulator.

## 9.3 In-System Programming

During programming, the flash memory has the following steps involved:
1. whole flash memory erase for security reason
2. write data flash and program flash memory
3. write customer information(Model/Version/Checksum)
4. read back data flash, program flash memory, and customer information, and then verify byte by byte
5. lock program/data flash if required

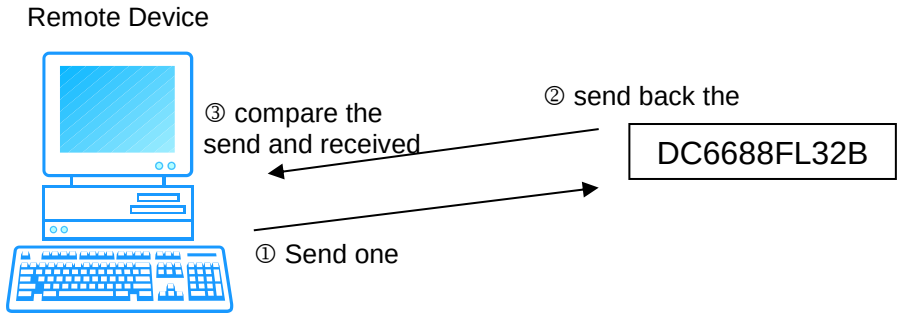
# 10DC6688FL32B

## 10.1 Firmware

| | Incident | Item | Description |
|---|---|---|---|
| A | Abnormal Serial Communication | UART 0 and UART 1 Receive | Both UART might have chance to get wrong data. The only solution is to send back the received data to confirm whether IC received correctly [1].<br><br>An alternative solution is to use software UART on receive part. Example refers to Application Note 024. |
| B | Abnormal Timer operation | Timer 0 and Timer 1 setting | The following condition at the same time happened is invalid:<br>i) Timer 1 set to mode 1, and<br>ii) Timer 0 set to mode 2 |
| C | Abnormal Data flash content | Initialization Data for Data Flash memory | To avoid using firmware to initialize Data Flash memory during in production line, it is recommended to also download the image to data flash memory by SL or SPI. |
| D | Abnormal Data flash content | Add 100ms delay at the beginning of the program | This applies to case where factory do initialization on data flash memory in production line by firmware itself. In view of the unstable power supply in production line found at the instant of insertion, add this delay at the beginning of the program to let the power stable before starting to run the firmware in production line.[2] |
| E | Abnormal software operation | Initialization for SRAM | This is a must as on power up, the contents in the SRAM are undetermined. |
| F | Software halt | Counter A reset | Counter A should be reset after exit from stop mode |
| G | Abnormal Interrupt | Bit 'IT0' in SFR | IT0 should set to 0 for PA interrupt for proper operation |
| H | Abnormal Interrupt [3] | Accessing Data Flash memory | 1) CPU suspends operation<br>2) only one interrupt is seen by CPU no matter how many interrupts from a peripheral<br>3) any interrupt will be delayed to serve by CPU until completion of accessing data flash<br>4) Disable watchdog before accessing data flash |
| J | High stop mode Idd | I/O port configuration in stop mode | In order to meet the Stop mode Idd current as stated in specification, the I/O port should be carefully configured. The following is prohibited:<br>1) Input without pull high - drive low and high<br>2) Output open drain with pull high - drive low |

| | | | |
|---|---|---|---|
| | | | 3) Output open drain without pull high - drive high |
| K | Abnormal Serial Communication | UART 0 and UART 1 Transmit | Don't modify Port B (byte addressable only) state when data is sending out on TXD0 / TXD1 pins. Otherwise, data will be corrupted. |

Remarks:
[1] The flow is illustrated below:




[2] An example on item b is shown below inside the red rectangle:

```
        ORG     0013H                   ;CA INT         IE.2
        SJMP    INITIAL                 ;
        ORG     001BH                   ;T1 OVER ONLY   IE.3
        LJMP    TIMER1_INTERRUPT        ;
        ORG     0023H                   ;SERIAL         IE.4
        SJMP    INITIAL                 ;
        ORG     002BH                   ;T2 INT         IE.5
        LJMP    TIMER2_INTERRUPT        ;
        ORG     0033H                   ;RESERVED       IE.6
        SJMP    INITIAL                 ;
        ORG     003BH                   ;SERIAL2        IE.7
        SJMP    INITIAL                 ;
;-------------------------------------------------------------------------------
        ORG     0040H                   ;
;-------------------------------------------------------------------------------

INITIAL:
        ; delay 100ms to let power stable
        ; in case the firmware write data flash at the beginning
        CALL    DELAY

        ; The following instructions to be added
        MOV     R0,#0FH
        MOV     A,#00111111B
        MOVX    @R0,A

INITIAL_1:
;       Customer Application start here
;       .
;       .
;       .
        JMP     INITIAL_1
```

[3] It depends on the application.

## 10.2 DEEMAX Emulator

### 10.2.1 Limitation on DC6688FL32B

When using emulator, there are some instructions, listed below, that the machine cycle is not identical to that used in our chip.

| | Dragonchip | DEEMAX emulator |
|---|---|---|
| Mnemonic | Machine cycle | Machine cycle |
| RET | 4 | 2 |
| RETI | 4 | 2 |
| JMP @A+DPTR | 3 | 2 |
| MOVC A,@A+DPTR | 3 | 2 |
| MOVC A,@A+PC | 3 | 2 |
| INC DPTR | 3 | 2 |

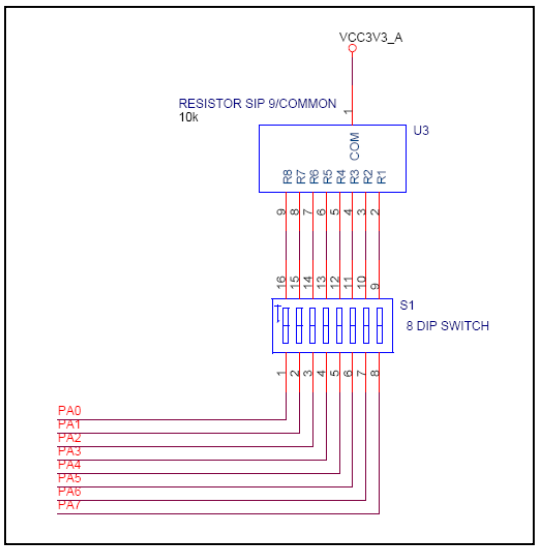
Additional limitation when using “DC6688EMT V4”:
1. Power down mode is not implemented in ICE, don’t use it, otherwise, undetermined result occurs.
2. No ISP select pin on CON4 in the ICE
3. No XOUT pin on CON4 in the ICE
4. No watchdog (basic timer)[1]
5. No backup mode
6. No ISP programming
7. Only operated at 3.3V power
8. No access to ‘T1_PCNTA’ register
9. No T2 output on PC2 by setting bit ‘T2OE’ in ‘T2MOD’ register

Remarks:
[1] Do NOT enable watchdog in emulator mode, otherwise, abnormal operation on emulator may occur.

Additional limitation when using “Developer III board ver3.0”:
1. Power down mode is not implemented in ICE, don’t use it, otherwise, undetermined result occurs.
2. No ISP select pin on CON4 in the ICE
3. No XOUT pin on CON4 in the ICE
4. No pull-up resistors in the ICE's port A, B and C[1]
5. No watchdog (basic timer)[2]
6. No backup mode
7. No ISP programming
8. No UART1
9. Only operated at 3.3V power
10. No access to ‘T1_PCNTA’ register
11. No T2 output on PC2 by setting bit ‘T2OE’ in ‘T2MOD’ register

Remarks:

[1] Item 4 above is suggested to add an optional pull-up resistor (as shown below) on customer's target board.

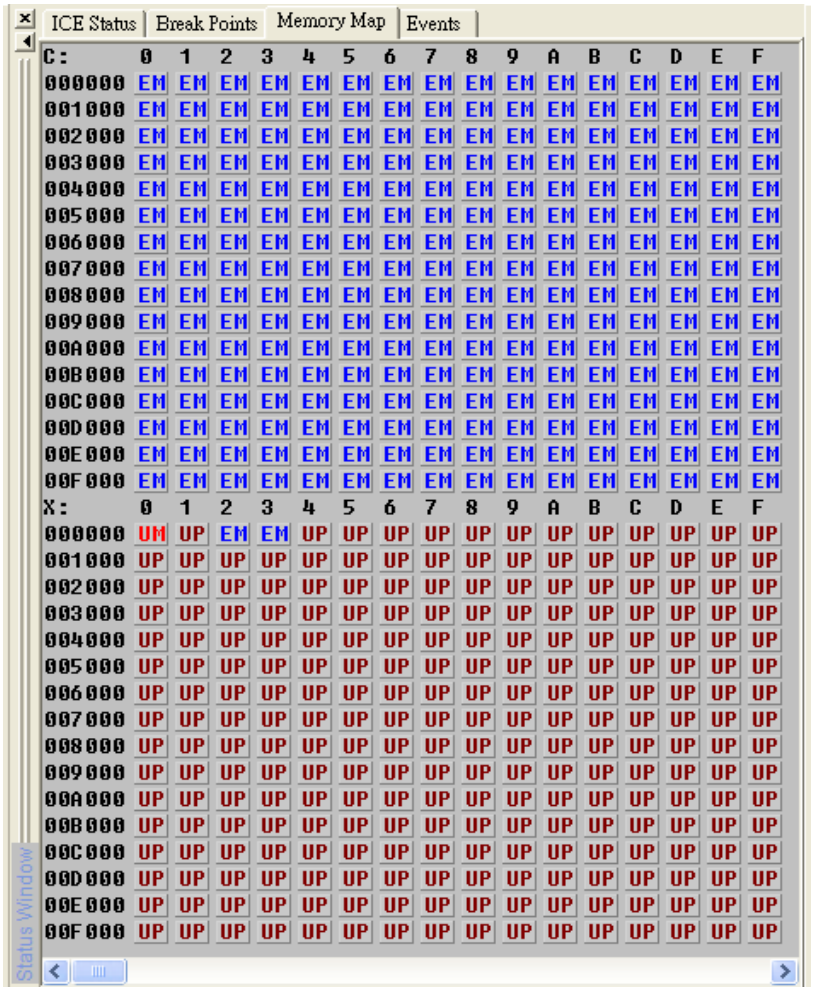



The Dip switch serves as an enable switch when connecting to emulator, and a disable switch when connecting to IC.

[2] Do NOT enable watchdog in emulator mode, otherwise, abnormal operation on emulator may occur.

### 10.2.2 Environment Setting

In the DEEMAX emulator's software environment, the 'Memory map' in 'Status window' has to modify as shown below:

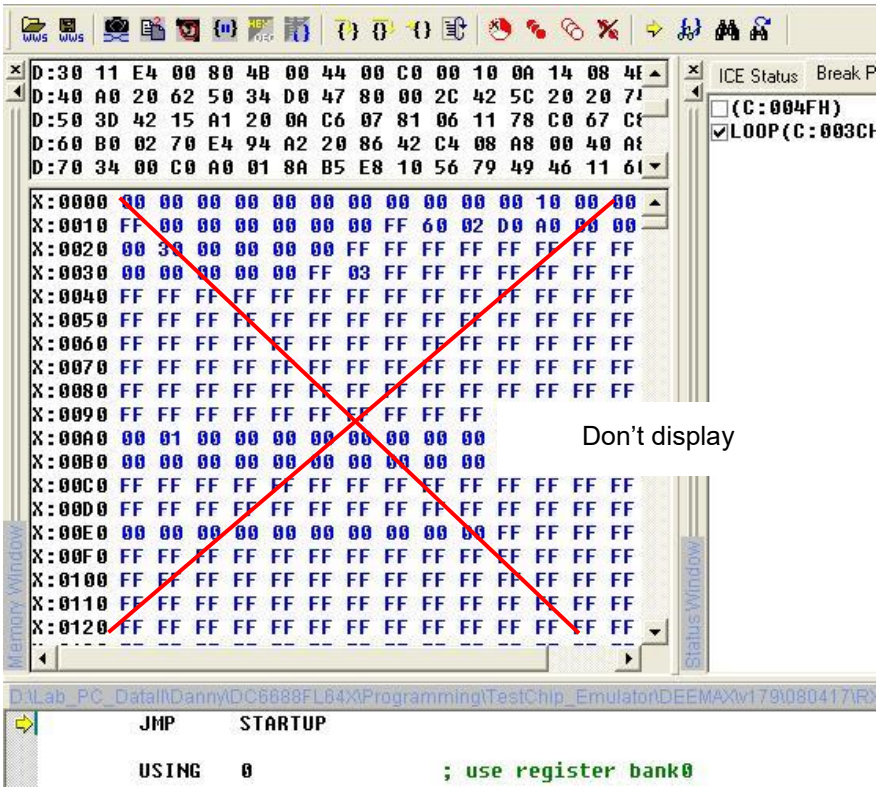


### 10.2.3 Precaution when debugging

When debugging the program in emulator, in the period of using the following register:

1) “RACTL” = 0x01 or 0x03
2) “RBCTL” = 0x01 or 0x03

Display of the memory region X:0000H ~ 01FFH is prohibited as shown below:




Since when setting “RACTL” to 0x01, whenever reading/writing “RDBA” the pointer “RAPTH/RAPTL” will automatically increment by 1. The emulator itself, in order to update the “Memory window”, will also read the whole XFR table one time whenever the program stops running. This reading will disturb the pointer “RAPTH/RAPTL” and the program to get the wrong data. This principle also applies to “RBCTL”

Display of the memory region X:0200H ~ FFFFH is allowed.

### 10.2.4 Additional Information

1. The ‘IRI’ pin on ‘J1’ connector cannot be used to capture the IR signal. Since the line from IR receiver on target board to ‘IRI’ pin on ‘J1’ connector is rather long, it results in higher noise and affect the receive performance. Therefore, the IR receiver is already made on the “Developer III board ver3.0”.

2. When download the code (program flash and/or data flash) to the IC, the whole flash memory will be erased, however, it is not the case in the emulator. The content in Data Flash memory always retained, whenever downloading the program to the emulator.

## 10.3 In-System Programming

During programming, the flash memory has the following steps involved:
1. whole flash memory erase for security reason
2. write data flash and program flash memory
3. write customer information(Model/Version/Checksum)
4. read back data flash, program flash memory, and customer information, and then verify byte by byte
5. lock program/data flash if required

# 11DC6688F2SCN

## 11.1 Firmware

| | Incident | Item | Description |
|---|---|---|---|
| A | Abnormal software operation | 'PORTC' register in XFR | Make sure bit 3~6 is '0' |
| B | Abnormal PC1 state | PCCONL[1,0] in XFR | 01 = prohibited |
| C | Abnormal Timer operation | Timer 0 and Timer 1 setting | The following condition at the same time happened is invalid:<br>i) Timer 1 set to mode 1, and<br>ii) Timer 0 set to mode 2 |
| D | Abnormal software operation | Initialization for SRAM | This is a must as on power up, the contents in the SRAM are undetermined. |
| E | Software halt | Counter A reset | Counter A should be reset after exit from stop mode |
| F | Abnormal Interrupt | Bit 'IT0' in SFR | IT0 should set to 0 for PB/PC interrupt for proper operation |
| G | High stop mode Idd | I/O port configuration in stop mode | In order to meet the Stop mode Idd current as stated in specification, the I/O port should be carefully configured. The following is prohibited:<br>1) Input without pull high - drive low and high<br>2) Output open drain with pull high - drive low<br>3) Output open drain without pull high - drive high |
| H | Incomplete waveform produced by Counter A | Counter A operation | Last pulse is incomplete under a condition even when Stop Carrier Mode (SCM) bit is set to 1<br><br>Avoid clearing the CAS bit when Counter A is reloading with CADATAL value. Therefore, the clock cycle between set/clear CAS bit cannot be completely divided by the period in clock cycle of the Counter A waveform. [1] |
| J | Key detect failure in key scanning | Internal Pull-up 150kΩ | This applies only to part no. **DC6688F2SCN-150**<br>Make sure delay time 150us is inserted just after going back from logic 0 to 1 before reading port PB, PC0 and PC2.[2] |
| K | Abnormal Serial Communication | UART 0 Transmit | Don't modify Port B (byte addressable only) state when data is sending out on TXD0 pins. Otherwise, data will be corrupted. |

Remarks:

[1] Example:

*Counter A Setting*

```
Clock Divider = 1
CADATAL Value = 0
CADATAH Value = 0
```

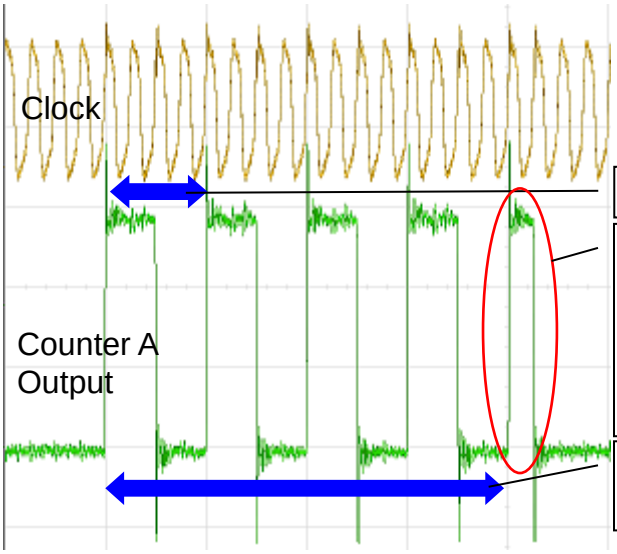
Period of the Counter A waveform in clock cycle:
(CADATAL + CADATAH + 4) × Clock Selection
= (0 + 0 + 4) × 1 = 4

*Error Code Snippet*

```
MOVR0, #CACON
MOVA, #0x07
MOVR1, #0x03
MOVX   @R0, A
MOVA, R1
MOVX   @R0, A
```

Clock Cycle between the set/clear of CAS bit:
Instruction cycle × 4
= ([MOV A,Rn] + [MOVX @Rn,A]) × 4
= (1 + 3) × 4
= 16

*Error Waveform*

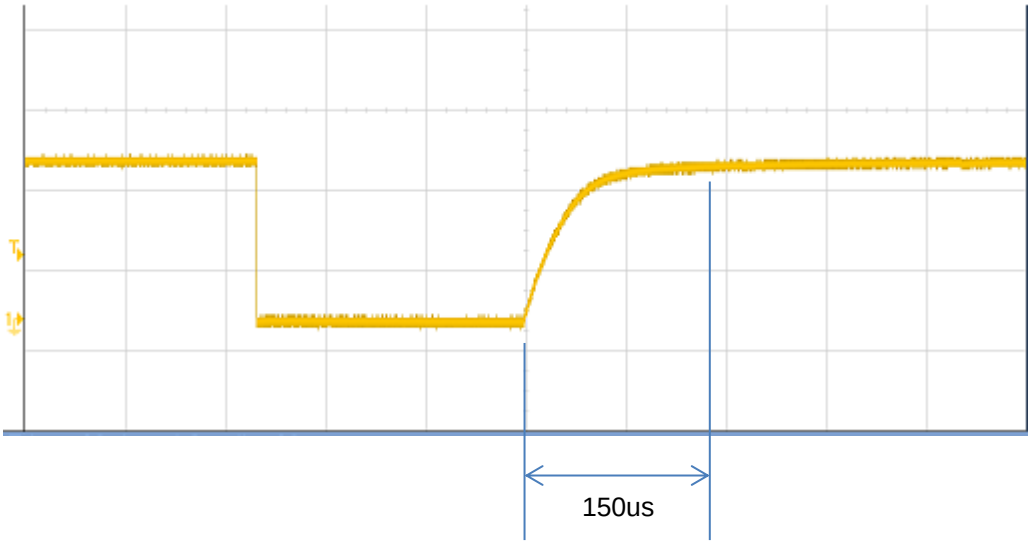



Counter A waveform period (4

(Clock between set/clear CAS) modulo (Clock of Counter A waveform)
= 16 mod 4 = 0
Hence, partial waveform generated

Clock cycle between set/clear the CAS bit (16 Clock)

[2] The timing diagram of PB0 (as an example) rising time due to internal pull-up




PB0 rising curve from 0 to 1 in key scanning

## 11.2 DEEMAX Emulator

### 11.2.1 Limitation on DC6688F2SCN

When using emulator, there are some instructions, listed below, that the machine cycle is not identical to that used in our chip.

| | Dragonchip | DEEMAX emulator |
|---|---|---|
| Mnemonic | Machine cycle | Machine cycle |
| RET | 4 | 2 |
| RETI | 4 | 2 |
| JMP @A+DPTR | 3 | 2 |
| MOVC A,@A+DPTR | 3 | 2 |
| MOVC A,@A+PC | 3 | 2 |
| INC DPTR | 3 | 2 |

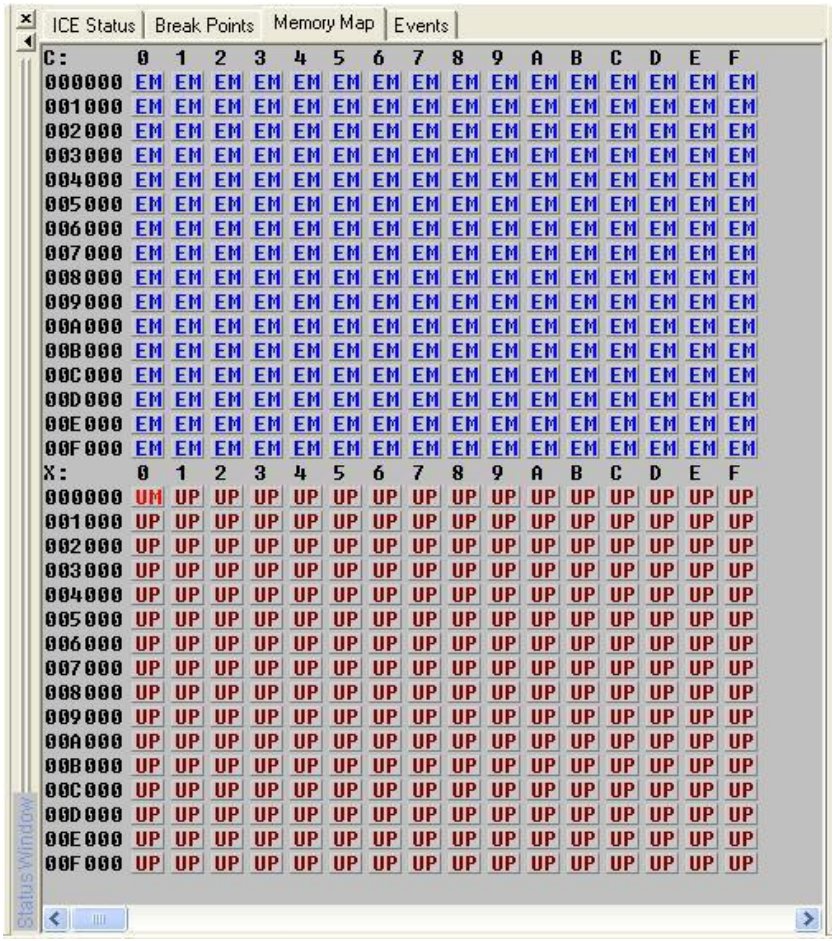
Additional limitation when using “DC6688EMT Rev4.0”:

1. Power down mode is not implemented in ICE, don’t use it, otherwise, undetermined result occurs.
2. No SL pin on J8 in the ICE
3. No XOUT pin on J8 in the ICE
4. No watchdog (basic timer)[1]
5. No backup mode
6. No ISP programming
7. Only operated at 3.3V power
8. No T2 output on PC2 by setting bit ‘T2OE’ in ‘T2MOD’ register
9. Each I/O of PA, PB and PC have series resistor (30 ohm) for protection
10. Each I/O of PA, PB and PC have 62k ohm pull-up resistor.

Remarks:
[1] Do NOT enable watchdog in emulator mode, otherwise, abnormal operation on emulator may occur.

### 11.2.2 Environment Setting

In the DEEMAX emulator's software environment, the 'Memory map' in 'Status window' has to modify as shown below:




## 11.3 In-System Programming

During programming, the flash memory has the following steps involved:

1. whole flash memory erase for security reason
2. write program flash memory
3. write customer information(Model/Version/Checksum)
4. read back program flash memory and customer information, and then verify byte by byte
5. lock program flash if required

# 12DC6688FLE

## 12.1 Firmware

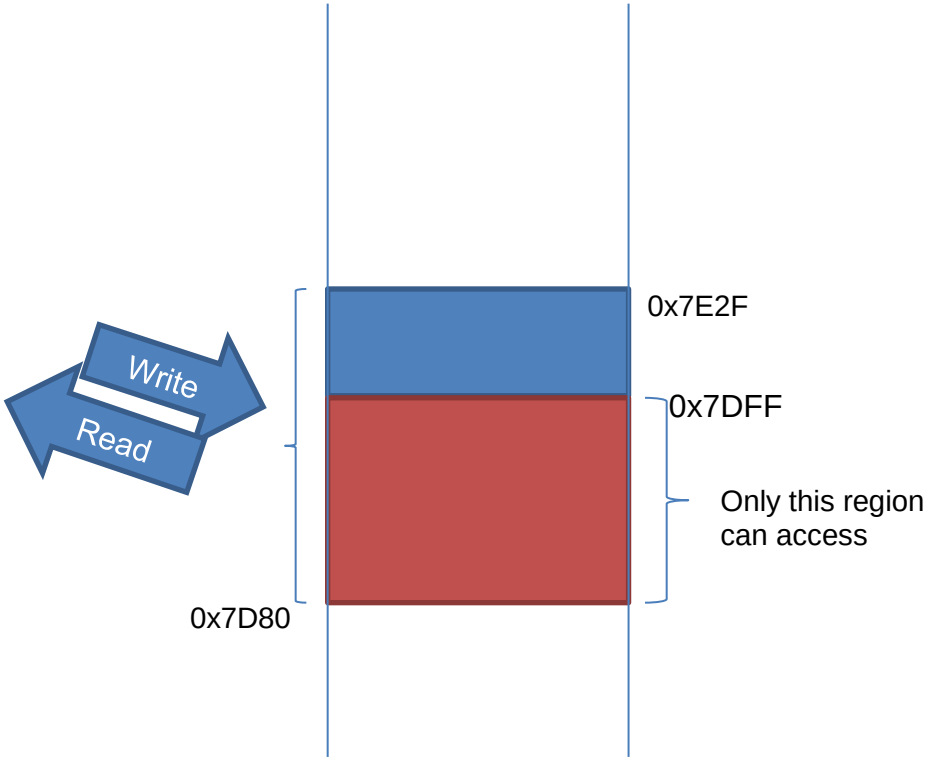
| | Incident | Item | Description |
|---|---|---|---|
| A | Abnormal Timer operation | Timer 0 and Timer 1 setting | The following condition at the same time happened is invalid:<br>i) Timer 1 set to mode 1, and<br>ii) Timer 0 set to mode 2 |
| B | Abnormal Data flash content | Initialization Data for Data Flash memory | To avoid using firmware to initialize Data Flash memory during in production line, it is recommended to also download the image to data flash memory by SL or SPI. |
| C | Abnormal Data flash content | Add 100ms delay at the beginning of the program | This applies to case where factory do initialization on data flash memory in production line by firmware itself. In view of the unstable power supply in production line found at the instant of insertion, add this delay at the beginning of the program to let the power stable before starting to run the firmware in production line.[1] |
| D | Abnormal software operation | Initialization for SRAM | This is a must as on power up, the contents in the SRAM are undetermined. |
| E | Software halt | Counter A reset | Counter A should be reset after exit from stop mode |
| F | Abnormal Interrupt | Bit 'IT0' in SFR | IT0 should set to 0 for PA interrupt for proper operation |
| G | Abnormal Interrupt [2] | Accessing Data Flash memory | 1) CPU suspends operation<br>2) only one interrupt is seen by CPU no matter how many interrupts from a peripheral<br>3) any interrupt will be delayed to serve by CPU until completion of accessing data flash<br>4) Disable watchdog before accessing data flash |
| H | High stop mode Idd | I/O port configuration in stop mode | In order to meet the Stop mode Idd current as stated in specification, the I/O port should be carefully configured. The following is prohibited:<br>1) Input without pull high<br>2) Output open drain with pull high - drive low<br>3) Output open drain without pull high - drive high |
| J | Abnormal read Data flash | Set RCPTH/RCPTL in Traditional Method | Before reading/writing Data flash with Traditional Method, RCPTH/RCPTL should be defined.<br><br>There is an exception that the program migrates from DC6688FL32A to DC6688FLX without modification. It is because the default value of RCPTH/RCPTL are |

| | | | |
|---|---|---|---|
| | | | '0'. It means pointing to page 0 in expanded SRAM. |
| **K** | Abnormal T24 capture | T24 capture reset | To avoid insufficient time for capture reset, T24_CON1 should be configured in the following order in each capture reset::<br>1) T24_CON1 = 0xA0; first, bit 5 set to 1<br>2) T24_CON1 = 0x8E; then, enable pointer x capture as you want, where x = A, B or C. In this example, enable all. |
| **L** | Abnormal LVI | Bit 'LVI_STATUS' no reset | To reset the LVI function on power up, at the beginning of the program, add 'read' instruction of the 'LVI' register.<br>MOV R0,#LVI<br>MOVX A,@R0 ; dummy read<br><br>MOVX A,@R0 ; second time read the bit 'LVI_STATUS'<br><br>The second time or onwards of reading will function properly. |
| **M** | Abnormal Serial Communication | UART 0 and UART 1 Transmit | Don't modify Port B (byte addressable only) state when data is sending out on TXD0 / TXD1 pins. Otherwise, data will be corrupted. |
| **N** | Abnormal Counter A Output | One shot mode in Counter A | In one shot mode (bit 'CAM' = 0), bit 'SCM' have to reset to 0 every time before setting CAS = 1. |
| **O** | Abnormal Read Data flash memory to Expanded SRAM | Accessing Data Flash memory | In enhanced method (FAM = 1), if FPTRH2/H/L = 0x yyFF, where yy is odd number, no matter what value in TXZH/L, RCPTH/L, and bit 1 of FAM,<br>Read data flash memory to Expanded SRAM will always transfer only the first byte. |
| **P** | Abnormal Read/Write Data Flash memory [4] | Accessing Data Flash memory | In enhanced method (FAM = 1), If the read/write flash region cover the location 0x7DFF (physical address), for example the starting address 0x7D80 (physical address) with ending address 0x7E2F (physical address), then only the range 0x7D80 to 0x7DFF can be accessed, no matter what value in TXZH/L, RCPTH/L, and bit 1 of FAM. [3]<br><br>The following location is applicable to this rule:<br>1) 0x7DFF<br>2) 0xFDFF |

Remarks:
[1] An example on item b is shown below inside the red rectangle:

```
        ORG     0013H                   ;CA INT         IE.2
        SJMP    INITIAL                 ;
        ORG     001BH                   ;T1 OVER ONLY   IE.3
        LJMP    TIMER1_INTERRUPT        ;
        ORG     0023H                   ;SERIAL         IE.4
        SJMP    INITIAL                 ;
        ORG     002BH                   ;T2 INT         IE.5
        LJMP    TIMER2_INTERRUPT        ;
        ORG     0033H                   ;RESERVED       IE.6
        SJMP    INITIAL                 ;
        ORG     003BH                   ;SERIAL2        IE.7
        SJMP    INITIAL                 ;
;-------------------------------------------------------------------------------
        ORG     0040H                   ;
;-------------------------------------------------------------------------------

INITIAL:
        ; delay 100ms to let power stable
        ; in case the firmware write data flash at the beginning
        CALL    DELAY

        ; The following instructions to be added
        MOV     R0,#0FH
        MOV     A,#00111111B
        MOVX    @R0,A

INITIAL_1:
;       Customer Application start here
;       .
;       .
;       .
        JMP     INITIAL_1
```

[2] It depends on the application.
[3] Diagram is shown below for illustration

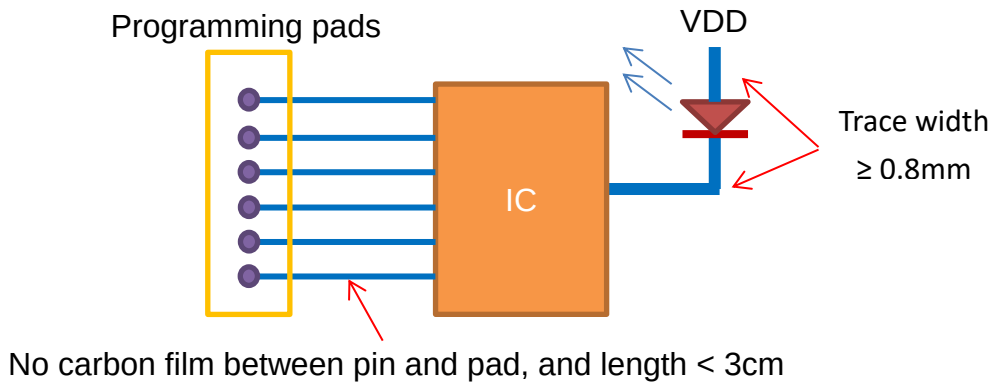



[4] Only applicable to FL96E / FL64E.

## 12.2 Hardware

### 12.2.1 PCB layout

Below are some guidelines for the layout.




## 12.3 DC6688EMT Emulator

### 12.3.1 Limitation on DC6688FLE

This applies only to the following version:
2. DC6688EMT-TX Rev1.x
3. DC6688EMT-4T Rev2.x

7 When using emulator, the instruction execution time is slightly different from the IC, therefore software delay and processing time will be different. User needs re-adjustment.
8 No backup mode
9 Only operated at 3.3V power
10 Peripherals
10.1 When the emulator is stopped in debugging platform, all the running peripherals (e.g. timer 2) will still keep running. Hence, the peripherals will be out of synchronization with the code instruction.
11 UART0 cannot use Timer 2 as baud-rate generator
12 No UART1

## 12.4 In-System Programming

During programming, the flash memory has the following steps involved:

1. whole flash memory erase for security reason
2. write program flash memory
3. write customer information(Model/Version/Checksum)
4. read back program flash memory and customer information, and then verify byte by byte
5. lock program flash if required

## 12.5 Production highlights in customer factory

It is recommended to review the following items before production in customer site:

1 The prevention and control of Electrostatic Discharge (ESD)

# 13DC6688FLT

## 13.1 Firmware

| | Incident | Item | Description |
|---|---|---|---|
| A | Abnormal Timer operation | Timer 0 and Timer 1 setting | The following condition at the same time happened is invalid:<br>i) Timer 1 set to mode 1, and<br>ii) Timer 0 set to mode 2 |
| B | Abnormal Data flash content | Initialization Data for Data Flash memory | To avoid using firmware to initialize Data Flash memory during in production line, it is recommended to also download the image to data flash memory by SL or SPI. |
| C | Abnormal Data flash content | Add 100ms delay at the beginning of the program | This applies to case where factory do initialization on data flash memory in production line by firmware itself. In view of the unstable power supply in production line found at the instant of insertion, add this delay at the beginning of the program to let the power stable before starting to run the firmware in production line.[1] |
| D | Abnormal software operation | Initialization for SRAM | This is a must as on power up, the contents in the SRAM are undetermined. |
| E | Software halt | Counter A reset | Counter A should be reset after exit from stop mode |
| F | Abnormal Interrupt | Bit 'IT0' in SFR | IT0 should set to 0 for PA interrupt for proper operation |
| G | Abnormal Interrupt [2] | Accessing Data Flash memory | 1. CPU suspends operation<br>2. only one interrupt is seen by CPU no matter how many interrupts from a peripheral<br>3. any interrupt will be delayed to serve by CPU until completion of accessing data flash<br>4. Disable watchdog before accessing data flash |
| H | High stop mode Idd | I/O port configuration in stop mode | In order to meet the Stop mode Idd current as stated in specification, the I/O port should be carefully configured. The following example regards as "floating" and is prohibited:<br>1) Input without pull high / down<br>2) Output n-channel open drain with pull high - drive low<br>3) Output n-channel open drain without pull high - drive high<br><br>In addition, the un-used / un-bonded I/O port must also be initialized. [3] |

| | | | |
|---|---|---|---|
| J | Abnormal read Data flash | Set RCPTH/RCPTL in Traditional Method | Before reading/writing Data flash with Traditional Method, RCPTH/RCPTL .should be defined. |
| K | Abnormal T24 capture | T24 capture reset | To avoid insufficient time for capture reset, T24_CON1 should be configured in the following order in each capture reset::<br>1) T24_CON1 = 0xA0; first, bit 5 set to 1<br>2) T24_CON1 = 0x8E; then, enable pointer x capture as you want, where x = A, B or C. In this example, enable all. |
| M | Abnormal Serial Communication | UART 0 and UART 1 Transmit | Don't modify Port B (byte addressable only) state when data is sending out on TXD0 / TXD1 pins. Otherwise, data will be corrupted. |
| N | Abnormal IR Learning | I/O interference | Toggle any I/O during IR learning would interfere the op amp to receive IR signal. Thus, I/O states should keep unchanged. |
| P | Abnormal IR Learning | Interrupted by other tasks | During learning, other tasks' associated interrupts should be disabled. Otherwise, timing will be corrupted. |

Remarks:
[1] An example on item b is shown below inside the red rectangle:

```
        ORG     0013H                   ;CA INT         IE.2
        SJMP    INITIAL                 ;
        ORG     001BH                   ;T1 OVER ONLY   IE.3
        LJMP    TIMER1_INTERRUPT        ;
        ORG     0023H                   ;SERIAL         IE.4
        SJMP    INITIAL                 ;
        ORG     002BH                   ;T2 INT         IE.5
        LJMP    TIMER2_INTERRUPT        ;
        ORG     0033H                   ;RESERVED       IE.6
        SJMP    INITIAL                 ;
        ORG     003BH                   ;SERIAL2        IE.7
        SJMP    INITIAL                 ;
;-------------------------------------------------------------------------------
        ORG     0040H                   ;
;-------------------------------------------------------------------------------

INITIAL:
        ; delay 100ms to let power stable
        ; in case the firmware write data flash at the beginning
        CALL    DELAY

        ; The following instructions to be added
        MOV     R0,#0FH
        MOV     A,#00111111B
        MOVX    @R0,A

INITIAL_1:
;       Customer Application start here
;       .
;       .
;       .
        JMP     INITIAL_1
```

[2] It depends on the application.
[3] Un-bonded I/O port (orange) is shown below. It should not be configured as

"floating".

FL32T / FL16T

| SOP16 | WLP16 | TSSOP20 | TSSOP24 | Pin Name |
|---|---|---|---|---|
| - | - | 5 | 5 | PA0/INTA |
| 5 | D2 | 6 | 6 | PA1/INTA/MISO |
| - | - | - | 7 | PA2/INTA |
| - | - | - | 8 | PA3/INTA |
| - | - | 7 | 9 | PA4/INTA |
| 6 | - | 8 | 10 | PA5/INTA |
| 7 | - | 9 | 11 | PA6/INTA |
| - | - | 10 | 12 | PA7/INTA |
| 8 | A1 | 11 | 13 | PB0/INTB/RxD0/ISPSCK/SDI |
| 9 | B2 | 12 | 14 | PB1/INTB/TxD0/MOSI/SDO |
| 10 | - | 13 | 15 | PB2/INTB/RxD1/SDI |
| 11 | - | 14 | 16 | PB3/INTB/TxD1/SDO |
| 12 | A2 | 15 | 17 | PB4/INTB/SCK/SCL1 |
| - | A3 | 16 | 18 | PB5/INTB/PWM0/SDA1 |
| 13 | C3 | 17 | 19 | PB6/INTB/T2EX/T24EX/PWM1 |
| 14 | - | 18 | 20 | PB7/INTB/SDA0 |
| - | D1 | - | 21 | PC0/T0/ISPSS/SCL0 |
| 15 | C1 | 19 | 22 | PC1/REM/IRTX/T1/T24_OUT |
| - | B3 | - | 23 | PC2/T2/T24_CLK |
| 3 | D3 | 3 | 3 | PD0 |
| 2 | D4 | 2 | 2 | PD1 |

FL96T / FL64T

| TSSOP20 (FL96TH4) | TSSOP20 (FL96TH) | TSSOP28 | LQFP32 | Pin Name |
|---|---|---|---|---|
| - | 5 | 6 | 7 | PA0/INTA |
| 6 | 6 | 7 | 8 | PA1/INTA/MISO |
| - | - | 8 | 9 | PA2/INTA |
| - | - | 9 | 10 | PA3/INTA |
| 7 | 7 | 10 | 11 | PA4/INTA |
| 8 | 8 | 11 | 12 | PA5/INTA |
| 9 | 9 | 12 | 13 | PA6/INTA |
| - | 10 | 13 | 14 | PA7/INTA |
| 11 | 11 | 16 | 19 | PB0/INTB/RxD0/ISPSCK/SDI |
| 12 | 12 | 17 | 20 | PB1/INTB/TxD0/MOSI/SDO |
| 13 | 13 | 18 | 21 | PB2/INTB/RxD1/SDI |
| 16 | 14 | 19 | 22 | PB3/INTB/TxD1/SDO |
| - | 15 | 20 | 23 | PB4/INTB/SCK/SCL1 |
| - | 16 | 21 | 24 | PB5/INTB/PWM0/SDA1 |
| 17 | 17 | 22 | 25 | PB6/INTB/T2EX/T24EX/PWM1 |
| - | 18 | 23 | 26 | PB7/INTB/SDA0 |
| - | - | 24 | 27 | PC0/T0/ISPSS/SCL0 |
| 18 | 19 | 25 | 28 | PC1/REM/IRTX/T1/T24_OUT |
| - | - | 26 | 29 | PC2/T2/T24_CLK |
| - | - | 28 | 31 | PC3 |
| - | - | 14 | 15 | PC4 |
| - | - | 15 | 18 | PC5 |
| 3 | 3 | 4 | 5 | PD0 |

| TSSOP20 (FL96TH4) | TSSOP20 (FL96TH) | TSSOP28 | LQFP32 | Pin Name |
|---|---|---|---|---|
| 2 | 2 | 3 | 4 | PD1 |
| - | - | - | 1 | PD2 |
| - | - | - | 32 | PD3 |
| - | - | - | - | PD4 |
| - | - | - | 16 | PD5 |
| - | - | - | 17 | PD6 |

It is suggested that at the beginning of the program, the un-used / un-bonded I/O (orange) should be configured as either:
1 Input with pull up, or
2 Output push-pull

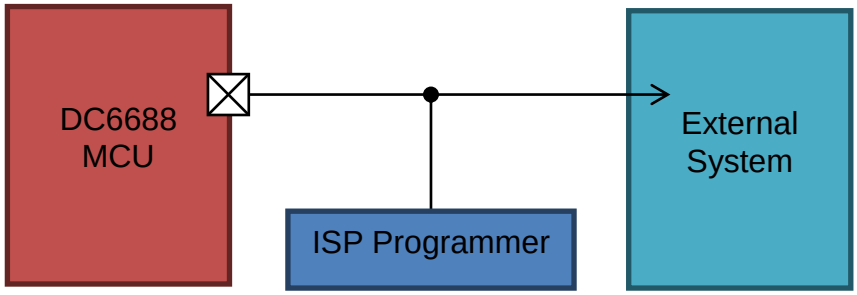
Un-bonded I/O (blue) should be configured as input only.

## 13.2 Hardware

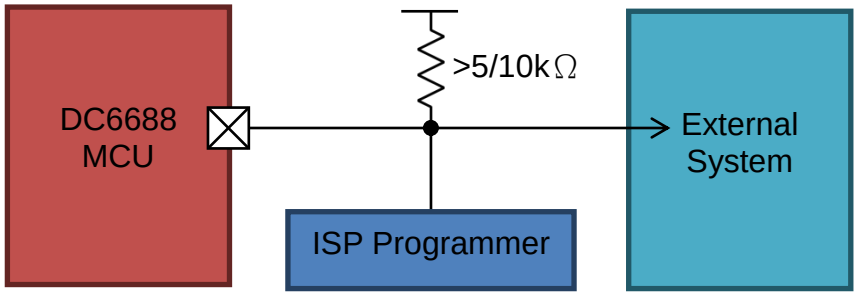
### 13.2.1 Programming pins

Precaution should be taken for the following I/O pins since they are shared with programming pins:
1 PD1
2 PD0
3 PB6

In order to do in-system-programming properly, below is a recommendation:
1 Push-pull output only
If the multiplexed IO is a push-pull output to external system's input, no additional isolation is needed.

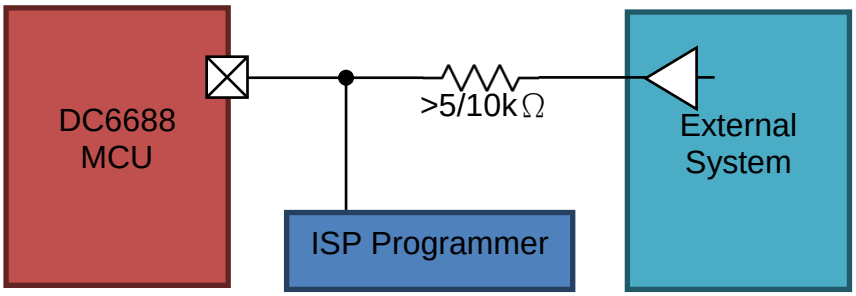



2 Open-drain output only w/ pull-up resistor
If the multiplexed IO is an open-drain output to external system's input with pulling resistor, the pulling resistors must not be under 5k Ohm for ISP of 1 to 2 devices and must not be under 10k Ohm for ISP of 3 or more devices together.




3 Input only or Bi-direction

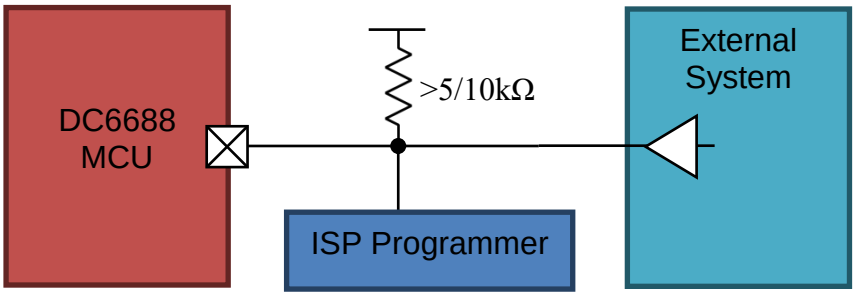
If the multiplexed IO is an input to external system's output, an isolation resistor should be added to block the external system's driving affect the programming signal. The isolation resistors must not be under 5k Ohm for ISP of 1 to 2 devices and must not be under 10k Ohm for ISP of 3 or more devices together.



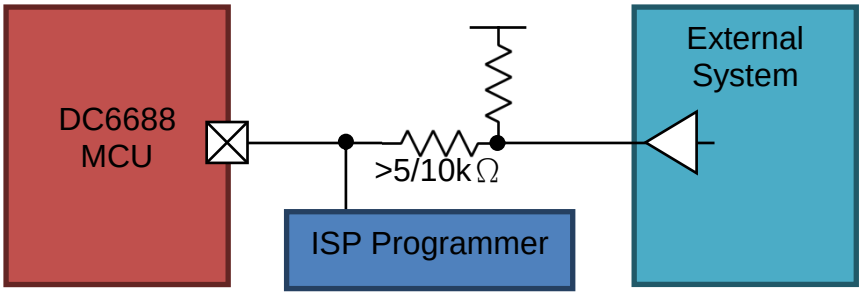

4 Input only or Bi-direction w/ pull-up resistor

If the multiplexed IO is an input to external system's output with pulling resistor, the pulling resistors must not be under 5k Ohm for ISP of 1 to 2 devices and must not be under 10k Ohm for ISP of 3 or more devices together.

If the external system will not drain the signal line during ISP, no additional isolation resistor is needed. (Case 1) On the other hand, if the signal line may be drained by the external system during ISP, an isolation resistor should be added as shown below with resistance value over 5k Ohm for ISP of 1 to 2 devices or over 10k Ohm for ISP of 3 or more devices together. (Case 2)




Case 1: External system will not drain during ISP

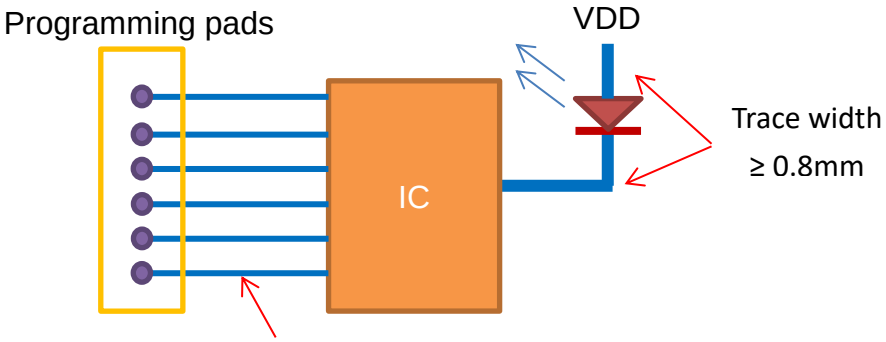



Case 2: External system might drain during ISP

On PD1, a pulse train will output on power up, it should be finished after 200ms

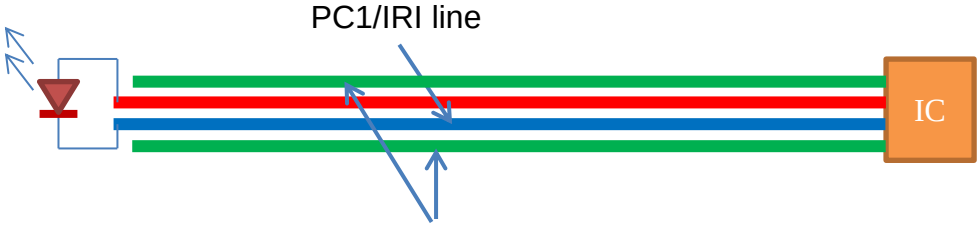
### 13.2.2 PCB layout

Below are some guidelines for the layout.



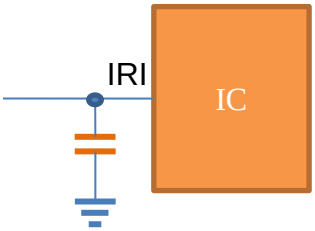

No carbon film between pin and pad, and length < 3cm

One-diode solution (IR Transmit + Learning)




VSS plate/line should be close to and in parallel with PC1/IRI

Maximum of 500pF capacitor on IRI pin can help filter the noise while reducing the sensitivity. The capacitor must be close to IRI pin.



## 13.3 DC6688EMT Emulator

This applies only to the following version:
1 DC6688EMT-FL32T Rev1.x (only for FL16T / FL32T)
2 DC6688EMT-1TS Rev1.x

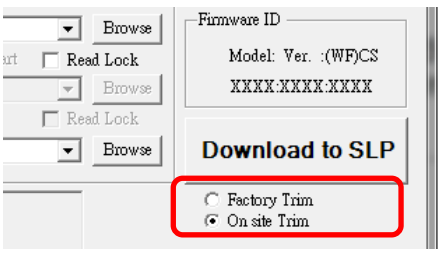
1 No backup mode
2 Only operated at 3.3V power
3 Peripherals
  3.1 When the emulator is stopped in debugging platform, all the running peripherals (e.g. timer 2) will still keep running. Hence, the peripherals will be out of synchronization with the code instruction.
4 Oscillator tolerance within +/-1%

## 13.4 In-System Programming

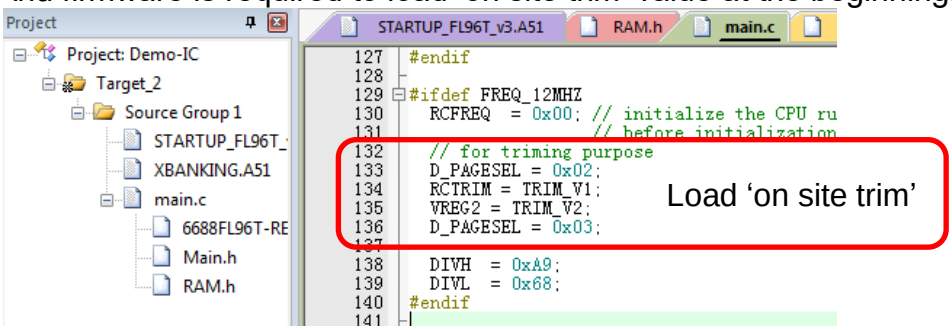
During programming, the flash memory has the following steps involved:
1. whole flash memory erase for security reason
2. write program flash memory
3. write customer information(Model/Version/Checksum)
4. read back program flash memory and customer information, and then verify byte by byte
5. lock program flash if required

## 13.5 Production highlights in customer factory

It is recommended to review the following items before production in customer site:
2 The prevention and control of Electrostatic Discharge (ESD)
3 PCBA level programming/trimming
  3.1 Programming must be done at room temperature (25°C). It is advised to arrange 1 hour cooling between soldering reflow and programming/trimming on PCBA level
  3.2 To obtain the best trimming performance, taking into consideration of PCB factor, it is recommended to do programming/trimming on PCBA level



And firmware is required to load ‘on site trim’ value at the beginning.




| Software SLP setting | Firmware | |
|---|---|---|
| | Load ‘on site trim’ | Load ‘factory trim’[1] |
| Factory Trim | Prohibited | Use ‘factory trim’ |
| On site Trim | Use ‘on site trim’ | Use ‘factory trim’ |

Remarks:
[1] Load ‘factory trim’ is done by disabling to load ‘on site trim’.

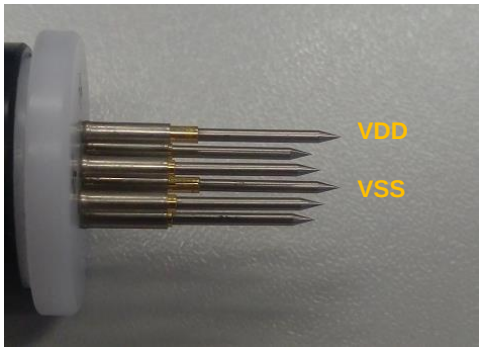
3.3 Trimming will be completed during programming stage

4 Routine check VDD from programmer to IC below 3.8V

5 Use SLP Programmer (DC6688SLP-USB Rev3.2 or higher)

6 Use Software SLP Rev6.9.3 or higher

7 Spring test probe
- 7.1 Add 10k ohm across PB6 and VDD
- 7.2 Add 10k ohm across PROG/ISPSEL and VDD
- 7.3 VDD/VSS pin landing on PCB first as shown below




8 The IC should be powered by our programmer ONLY
- 8.1 The programmer control power off/on to ensure proper programming & verification operations.

## 13.6 UV light on WLP16 package

In view of this kind of incident, an epoxy coating is added on WLP16 to protect the die but also blocking the UV light as much as possible. In general usage where an enclosure is present, it should be sufficient.

It is advised not to put the WLP package next to the UV flash light where the

UV light leakage is present.

# 14DC6688FST

## 14.1 Firmware

| | Incident | Item | Description |
|---|---|---|---|
| A | Abnormal software operation | 'PC' register in XFR | Make sure bit 6 is '0' |
| B | High stop mode Idd | Add instructions | Both bit 1 ~ 0 of 'PCCONL' register should be set to 1 before entering stop mode |
| C | Abnormal Timer operation | Timer 0 and Timer 1 setting | The following condition at the same time happened is invalid:<br>i) Timer 1 set to mode 1, and<br>ii) Timer 0 set to mode 2 |
| D | Abnormal Data flash content | Initialization Data for Data Flash memory | To avoid using firmware to initialize Data Flash memory during in production line, it is recommended to also download the image to data flash memory by SL or SPI. |
| E | Abnormal Data flash content | Add 100ms delay at the beginning of the program | This applies to case where factory do initialization on data flash memory in production line by firmware itself. In view of the unstable power supply in production line found at the instant of insertion, add this delay at the beginning of the program to let the power stable before starting to run the firmware in production line.[1] |
| F | Abnormal software operation | Initialization for SRAM | This is a must as on power up, the contents in the SRAM are undetermined. |
| G | Software halt | Counter A reset | Counter A should be reset after exit from stop mode |
| H | Abnormal Interrupt | Bit 'IT0' in SFR | IT0 should set to 0 for PA interrupt for proper operation |
| J | Abnormal Interrupt [5] | Accessing Data Flash memory | 1) CPU suspends operation<br>2) only one interrupt is seen by CPU no matter how many interrupts from a peripheral<br>3) any interrupt will be delayed to serve by CPU until completion of accessing data flash<br>4) Disable watchdog before accessing data flash |
| K | High stop mode Idd | I/O port configuration in stop mode | In order to meet the Stop mode Idd current as stated in specification, the I/O port should be carefully configured. The following example regards as "floating" and is prohibited:<br>1) Input without pull high / down |

| | | | |
|---|---|---|---|
| | | | 2) Output n-channel open drain with pull high - drive low<br>3) Output n-channel open drain without pull high - drive high<br><br>In addition, the un-used / un-bonded I/O port must also be initialized. [2] |
| M | Abnormal Serial Communication | UART 0 and UART 1 Transmit | Don't modify Port B (byte addressable only) state when data is sending out on TXD0 / TXD1 pins. Otherwise, data will be corrupted. |

[1] An example on item b is shown below inside the red rectangle:

```
        ORG     0013H                   ;CA INT         IE.2
        SJMP    INITIAL                 ;
        ORG     001BH                   ;T1 OVER ONLY   IE.3
        LJMP    TIMER1_INTERRUPT        ;
        ORG     0023H                   ;SERIAL         IE.4
        SJMP    INITIAL                 ;
        ORG     002BH                   ;T2 INT         IE.5
        LJMP    TIMER2_INTERRUPT        ;
        ORG     0033H                   ;RESERVED       IE.6
        SJMP    INITIAL                 ;
        ORG     003BH                   ;SERIAL2        IE.7
        SJMP    INITIAL                 ;
;-------------------------------------------------------------------------------
        ORG     0040H                   ;
;-------------------------------------------------------------------------------

INITIAL:
        ; delay 100ms to let power stable
        ; in case the firmware write data flash at the beginning
        CALL    DELAY

        ; The following instructions to be added
        MOV     R0,#0FH
        MOV     A,#00111111B
        MOVX    @R0,A

INITIAL_1:
;       Customer Application start here
;       .
;       .
;       .
        JMP     INITIAL_1
```

[2] Un-bonded I/O port (orange) is shown below. It should not be configured as "floating".

| SOP16 | TSSOP8 | TSSOP20 | TSSOP24 | Pin Name |
|---|---|---|---|---|
| - | - | 5 | 5 | PA0/INTA |
| 5 | - | 6 | 6 | PA1/INTA/MISO |
| - | - | - | 7 | PA2/INTA |
| - | - | - | 8 | PA3/INTA |
| - | - | 7 | 9 | PA4/INTA |
| 6 | - | 8 | 10 | PA5/INTA |
| 7 | - | 9 | 11 | PA6/INTA |
| - | - | 10 | 12 | PA7/INTA |
| 8 | 6 | 11 | 13 | PB0/INTB/RxD0/ISPSCK/SDI |
| 9 | 7 | 12 | 14 | PB1/INTB/TxD0/MOSI/SDO |
| 10 | - | 13 | 15 | PB2/INTB/RxD1/SDI |

| SOP16 | TSSOP8 | TSSOP20 | TSSOP24 | Pin Name |
|---|---|---|---|---|
| 11 | - | 14 | 16 | PB3/INTB/TxD1/SDO |
| 12 | - | 15 | 17 | PB4/INTB/SCK/SCL1 |
| - | - | 16 | 18 | PB5/INTB/SDA1 |
| 13 | - | 17 | 19 | PB6/INTB |
| 14 | - | 18 | 20 | PB7/INTB/SDA0 |
| - | - | - | 21 | PC0/T0/ISPSS/SCL0 |
| 15 | 8 | 19 | 22 | PC1/REM/T1 |
| - | - | - | 23 | PC2/T2 |
| 3 | 4 | 3 | 3 | PD0 |
| 2 | 3 | 2 | 2 | PD1 |

It is suggested that at the beginning of the program, the un-used / un-bonded I/O (orange) should be configured as either:
1. Input with pull up, or
2. Output push-pull

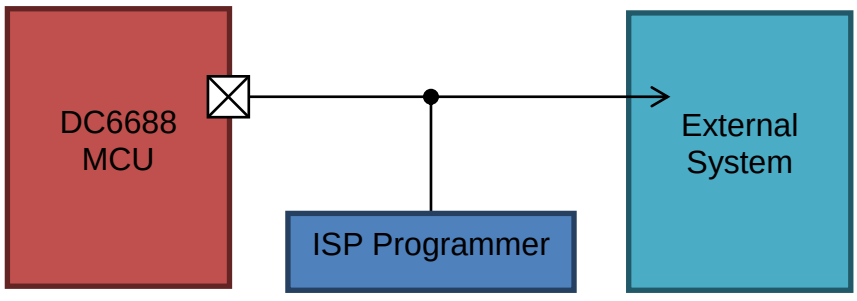
Un-bonded I/O (blue) should be configured as input only.

## 14.2 Hardware

### 14.2.1 Programming pins

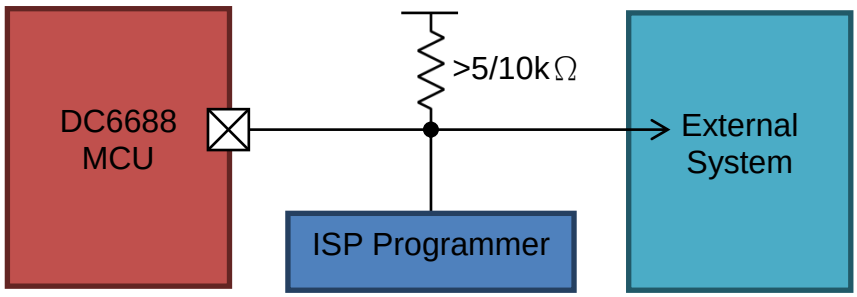
Precaution should be taken for the following I/O pins since they are shared with programming pins:
1 PD1
2 PD0
3 PB6

In order to do in-system-programming properly, below is a recommendation:
1 Push-pull output only
If the multiplexed IO is a push-pull output to external system's input, no additional isolation is needed.



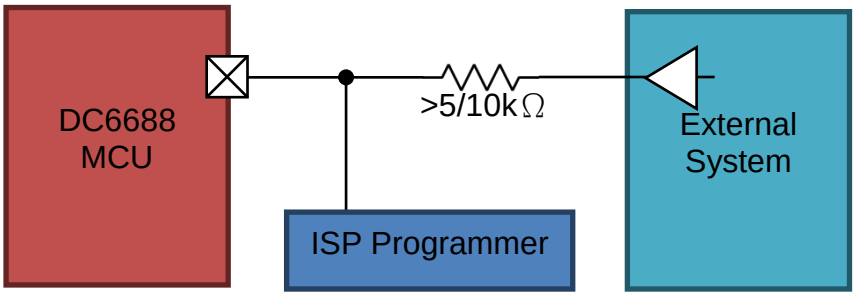

2 Open-drain output only w/ pull-up resistor
If the multiplexed IO is an open-drain output to external system's input with pulling resistor, the pulling resistors must not be under 5k Ohm for ISP of 1

to 2 devices and must not be under 10k Ohm for ISP of 3 or more devices together.




3 Input only or Bi-direction

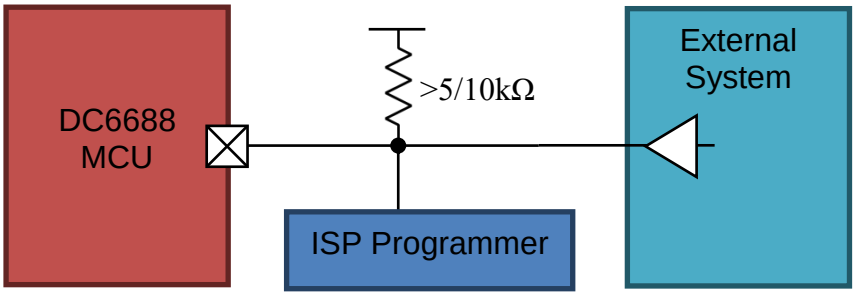
If the multiplexed IO is an input to external system's output, an isolation resistor should be added to block the external system's driving affect the programming signal. The isolation resistors must not be under 5k Ohm for ISP of 1 to 2 devices and must not be under 10k Ohm for ISP of 3 or more devices together.



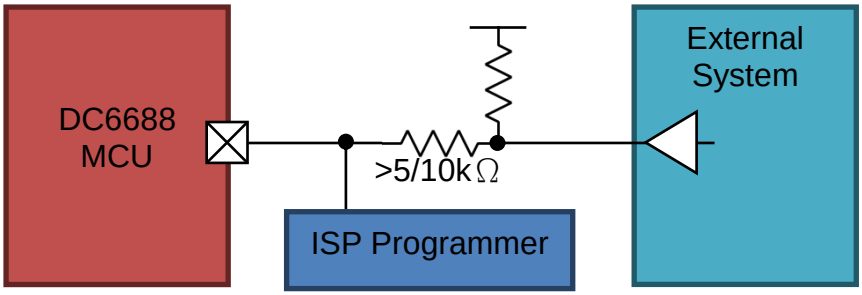

4 Input only or Bi-direction w/ pull-up resistor

If the multiplexed IO is an input to external system's output with pulling resistor, the pulling resistors must not be under 5k Ohm for ISP of 1 to 2 devices and must not be under 10k Ohm for ISP of 3 or more devices together.

If the external system will not drain the signal line during ISP, no additional isolation resistor is needed. (Case 1) On the other hand, if the signal line may be drained by the external system during ISP, an isolation resistor should be added as shown below with resistance value over 5k Ohm for ISP of 1 to 2 devices or over 10k Ohm for ISP of 3 or more devices together. (Case 2)




Case 3: External system will not drain during ISP

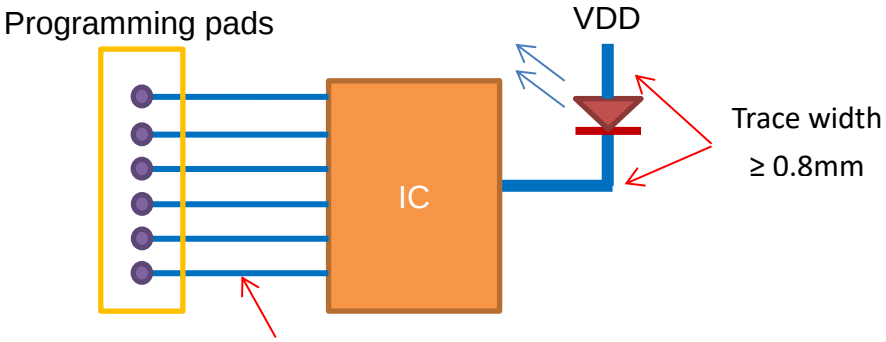



Case 4: External system might drain during ISP

On PD1, a pulse train will output on power up, it should be finished after 200ms

### 14.2.2 PCB layout

Below are some guidelines for the layout.




No carbon film between pin and pad, and length < 3cm

## 14.3 DC6688EMT Emulator

This applies only to the following version:

1 DC6688EMT-FL32T Rev1.x (only for F4ST / F16ST / F30ST)
2 DC6688EMT-1TS Rev1.x

1 No backup mode
2 Only operated at 3.3V power
3 Peripherals
  3.1 When the emulator is stopped in debugging platform, all the running peripherals (e.g. timer 2) will still keep running. Hence, the peripherals will be out of synchronization with the code instruction.
4 Oscillator tolerance within +/-1%

## 14.4 In-System Programming

During programming, the flash memory has the following steps involved:

1. whole flash memory erase for security reason
2. write program flash memory
3. write customer information(Model/Version/Checksum)
4. read back program flash memory and customer information, and then verify byte by byte
5. lock program flash if required

## 14.5 Production highlights in customer factory

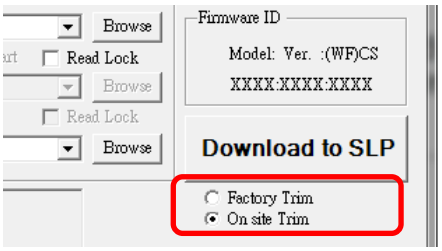
It is recommended to review the following items before production in customer site:

1 The prevention and control of Electrostatic Discharge (ESD)

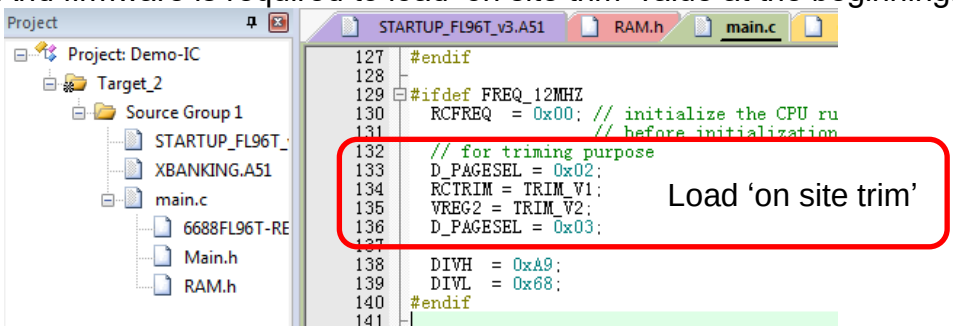
2 PCBA level programming/trimming

2.1 Programming must be done at room temperature (25°C). It is advised to arrange 1 hour cooling between soldering reflow and programming/trimming on PCBA level

2.2 To obtain the best trimming performance, taking into consideration of PCB factor, it is recommended to do programming/trimming on PCBA level




And firmware is required to load ‘on site trim’ value at the beginning.




| Software SLP setting | Firmware | |
|---|---|---|
| | Load ‘on site trim’ | Load ‘factory trim’[1] |

| | Firmware | |
|---|---|---|
| Factory Trim | Prohibited | Use ‘factory trim’ |
| On site Trim | Use ‘on site trim’ | Use ‘factory trim’ |

Remarks:
[1] Load ‘factory trim’ is done by disabling to load ‘on site trim’.

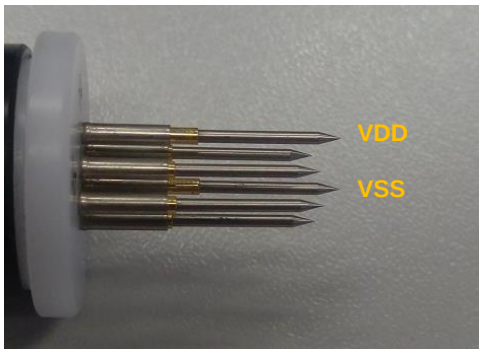
2.3 Trimming will be completed during programming stage

3 Routine check VDD from programmer to IC below 3.8V

4 Use SLP Programmer (DC6688SLP-USB Rev3.2 or higher)

5 Use Software SLP Rev6.9.3 or higher

6 Spring test probe
- 6.1 Add 10k ohm across PB6 and VDD
- 6.2 Add 10k ohm across PROG/ISPSEL and VDD
- 6.3 VDD/VSS pin landing on PCB first as shown below




7 The IC should be powered by our programmer ONLY
- 7.1 The programmer control power off/on to ensure proper programming & verification operations.

# 15DC6688F2SER

## 15.1 Firmware

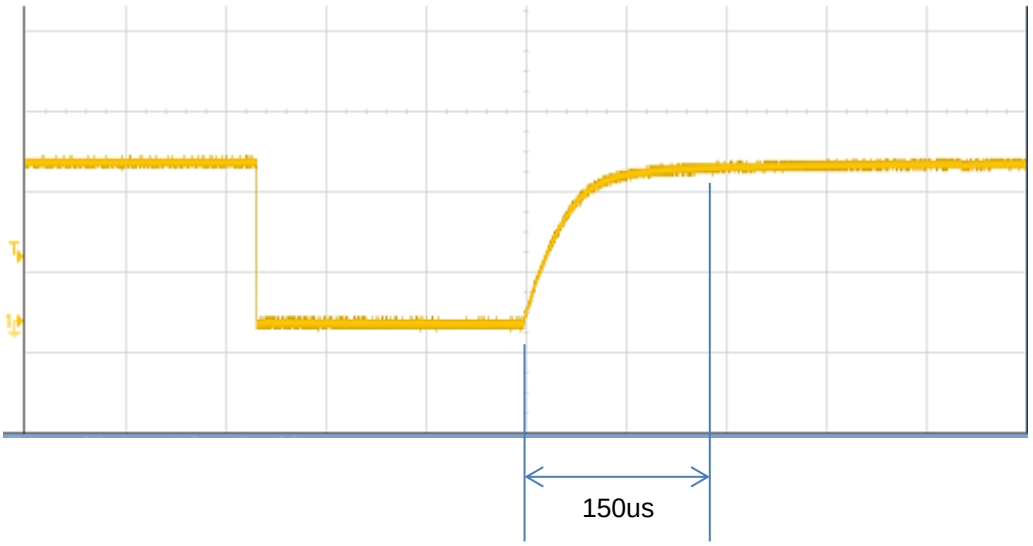
| | Incident | Item | Description |
|---|---|---|---|
| A | Abnormal software operation | 'PORTC' register | Make sure bit 3~6 is '0' |
| B | Abnormal PC1 state | PCCONL[1,0] in XFR | 01 = prohibited |
| C | Abnormal Timer operation | Timer 0 and Timer 1 setting | The following condition at the same time happened is invalid:<br>i) Timer 1 set to mode 1, and<br>ii) Timer 0 set to mode 2 |
| D | Abnormal Data flash content | Initialization Data for Data Flash memory | To avoid using firmware to initialize Data Flash memory during in production line, it is recommended to also download the image to data flash memory by SL or SPI. |
| E | Abnormal Data flash content | Add 100ms delay at the beginning of the program | This applies to case where factory do initialization on data flash memory in production line by firmware itself. In view of the unstable power supply in production line found at the instant of insertion, add this delay at the beginning of the program to let the power stable before starting to run the firmware in production line.[1] |
| F | Abnormal software operation | Initialization for SRAM | This is a must as on power up, the contents in the SRAM are undetermined. |
| G | Software halt | Counter A reset | Counter A should be reset after exit from stop mode |
| H | Abnormal Interrupt | Bit 'IT0' in SFR | IT0 should set to 0 for PB/PC interrupt for proper operation |
| J | High stop mode Idd | I/O port configuration in stop mode | In order to meet the Stop mode Idd current as stated in specification, the I/O port should be carefully configured. The following is prohibited:<br>1) Input without pull high - drive low and high<br>2) Output open drain with pull high - drive low<br>3) Output open drain without pull high - drive high |
| K | Key detect failure in key scanning | Internal Pull-up 150kΩ | Make sure delay time 150us is inserted just after going back from logic 0 to 1 before reading port PB, PC0 and PC2.[2] |

Remarks:

[1] An example on item b is shown below inside the red rectangle:

```
        ORG     0013H                   ;CA INT         IE.2
        SJMP    INITIAL                 ;
        ORG     001BH                   ;T1 OVER ONLY   IE.3
        LJMP    TIMER1_INTERRUPT        ;
        ORG     0023H                   ;SERIAL         IE.4
        SJMP    INITIAL                 ;
        ORG     002BH                   ;T2 INT         IE.5
        LJMP    TIMER2_INTERRUPT        ;
        ORG     0033H                   ;RESERVED       IE.6
        SJMP    INITIAL                 ;
        ORG     003BH                   ;SERIAL2        IE.7
        SJMP    INITIAL                 ;
;-------------------------------------------------------------------------------
        ORG     0040H                   ;
;-------------------------------------------------------------------------------

INITIAL:
        ; delay 100ms to let power stable
        ; in case the firmware write data flash at the beginning
        CALL    DELAY

        ; The following instructions to be added
        MOV     R0,#0FH
        MOV     A,#00111111B
        MOVX    @R0,A

INITIAL_1:
;       Customer Application start here
;       .
;       .
;       .
        JMP     INITIAL_1
```

[2] The timing diagram of PB0 (as an example) rising time due to internal pull-up

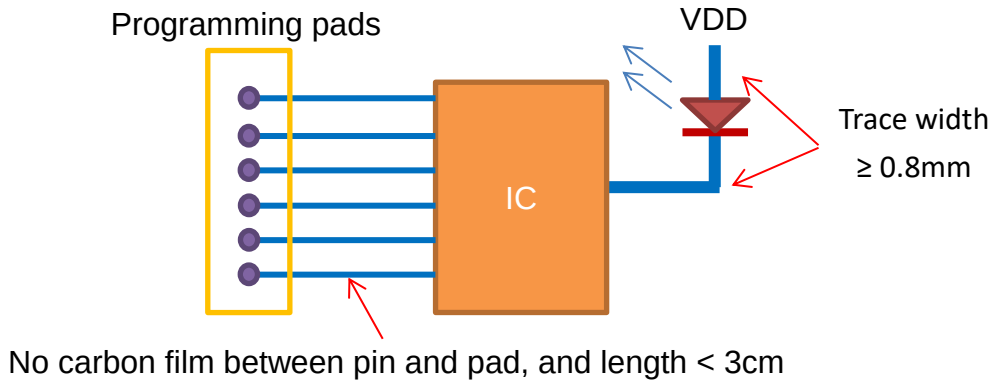



PB0 rising curve from 0 to 1 in key scanning

## 15.2 Hardware

### 15.2.1 PCB layout

Below are some guidelines for the layout.




## 15.3 DC6688EMT Emulator

This applies only to the following version:
DC6688EMT-F2R Rev1.x

1 No backup mode
2 Only operated at 3.3V power
3 Peripherals
  3.1 When the emulator is stopped in debugging platform, all the running peripherals (e.g. timer 2) will still keep running. Hence, the peripherals will be out of synchronization with the code instruction.

## 15.4 In-System Programming

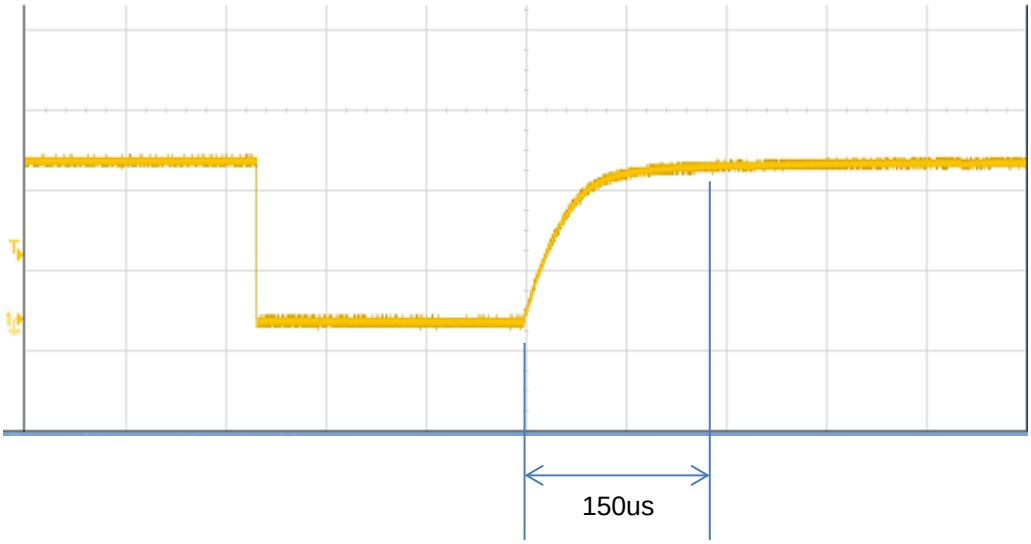
During programming, the flash memory has the following steps involved:
1. whole flash memory erase for security reason
2. write program flash memory
3. write customer information(Model/Version/Checksum)
4. read back program flash memory and customer information, and then verify byte by byte
5. lock program flash if required

# 16DC6688F2SEN

## 16.1 Firmware

| | Incident | Item | Description |
|---|---|---|---|
| A | Abnormal software operation | 'PORTC' register | Make sure bit 3~6 is '0' |
| B | Abnormal PC1 state | PCCONL[1,0] in XFR | 01 = prohibited |
| C | Abnormal Timer operation | Timer 0 and Timer 1 setting | The following condition at the same time happened is invalid:<br>i) Timer 1 set to mode 1, and<br>ii) Timer 0 set to mode 2 |
| D | Abnormal software operation | Initialization for SRAM | This is a must as on power up, the contents in the SRAM are undetermined. |
| E | Software halt | Counter A reset | Counter A should be reset after exit from stop mode |
| F | Abnormal Interrupt | Bit 'IT0' in SFR | IT0 should set to 0 for PB/PC interrupt for proper operation |
| G | High stop mode Idd | I/O port configuration in stop mode | In order to meet the Stop mode Idd current as stated in specification, the I/O port should be carefully configured. The following is prohibited:<br>1) Input without pull high - drive low and high<br>2) Output open drain with pull high - drive low<br>3) Output open drain without pull high - drive high |
| H | Key detect failure in key scanning | Internal Pull-up 150kΩ | Make sure delay time 150us is inserted just after going back from logic 0 to 1 before reading port PB, PC0 and PC2.[1] |

Remarks:
[1] The timing diagram of PB0 (as an example) rising time due to internal pull-up




PB0 rising curve from 0 to 1 in key scanning

## 16.2 DC6688EMT Emulator

This applies only to the following version:
DC6688EMT-F2R Rev1.x

1 No backup mode
2 Only operated at 3.3V power
3 Peripherals
  3.1 When the emulator is stopped in debugging platform, all the running peripherals (e.g. timer 2) will still keep running. Hence, the peripherals will be out of synchronization with the code instruction.

## 16.3 In-System Programming

During programming, the flash memory has the following steps involved:
1. whole flash memory erase for security reason
2. write program flash memory
3. write customer information(Model/Version/Checksum)
4. read back program flash memory and customer information, and then verify byte by byte
5. lock program flash if required

# 17DC6688FLTE

## 17.1 FL96TE / FL64TE

### 17.1.1 Firmware

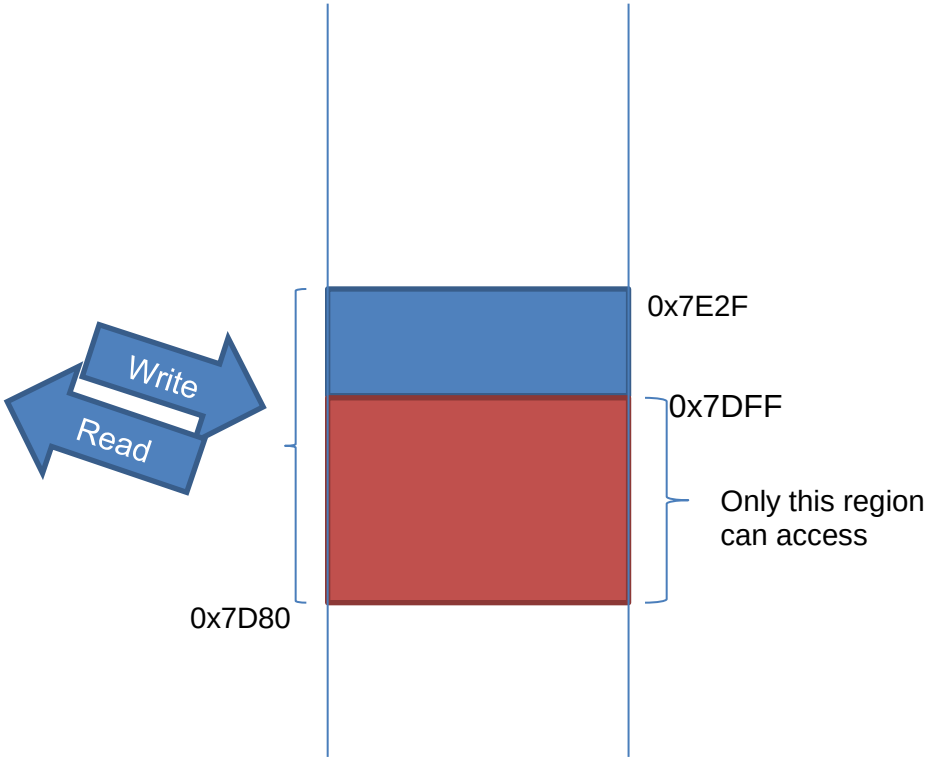
| | Incident | Item | Description |
|---|---|---|---|
| A | Abnormal Timer operation | Timer 0 and Timer 1 setting | The following condition at the same time happened is invalid:<br>i) Timer 1 set to mode 1, and<br>ii) Timer 0 set to mode 2 |
| B | Abnormal Data flash content | Initialization Data for Data Flash memory | To avoid using firmware to initialize Data Flash memory during in production line, it is recommended to also download the image to data flash memory by SL or SPI. |
| C | Abnormal Data flash content | Add 100ms delay at the beginning of the program | This applies to case where factory do initialization on data flash memory in production line by firmware itself. In view of the unstable power supply in production line found at the instant of insertion, add this delay at the beginning of the program to let the power stable before starting to run the firmware in production line.[1] |
| D | Abnormal software operation | Initialization for SRAM | This is a must as on power up, the contents in the SRAM are undetermined. |
| E | Software halt | Counter A reset | Counter A should be reset after exit from stop mode |
| F | Abnormal Interrupt | Bit 'IT0' in SFR | IT0 should set to 0 for PA interrupt for proper operation |
| G | Abnormal Interrupt [2] | Accessing Data Flash memory | 1) CPU suspends operation<br>2) only one interrupt is seen by CPU no matter how many interrupts from a peripheral<br>3) any interrupt will be delayed to serve by CPU until completion of accessing data flash<br>4) Disable watchdog before accessing data flash |
| H | High stop mode Idd | I/O port configuration in stop mode | In order to meet the Stop mode Idd current as stated in specification, the I/O port should be carefully configured. The following is prohibited:<br>1) Input without pull high - drive low and high<br>2) Output open drain with pull high - drive low<br>3) Output open drain without pull high - drive high |
| J | Abnormal read Data flash | Set RCPTH/RCPTL in Traditional Method | Before reading/writing Data flash with Traditional Method, RCPTH/RCPTL .should be defined. |

| | | | |
|---|---|---|---|
| **K** | Abnormal T24 capture | T24 capture reset | To avoid insufficient time for capture reset, T24_CON1 should be configured in the following order in each capture reset::<br>1) T24_CON1 = 0xA0; first, bit 5 set to 1<br>2) T24_CON1 = 0x8E; then, enable pointer x capture as you want, where x = A, B or C. In this example, enable all. |
| **L** | Abnormal LVI | Bit 'LVI_STATUS' no reset | To reset the LVI function on power up, at the beginning of the program add 'read' instruction of the 'LVI' register.<br>MOV R0,#LVI<br>MOVX A,@R0 ; dummy read<br><br>MOVX A,@R0 ; second time read the bit 'LVI_STATUS'<br><br>The second time or onwards of reading will function properly. |
| **M** | Abnormal Counter A Output | One shot mode in Counter A | In one shot mode (bit 'CAM' = 0), bit 'SCM' have to reset to 0 every time before setting CAS = 1. |
| **N** | Abnormal Read Data flash memory to Expanded SRAM | Accessing Data Flash memory | In enhanced method (FAM = 1), if FPTRH2/H/L = 0x yyFF, where yy is odd number, no matter what value in TXZH/L, RCPTH/L, and bit 1 of FAM,<br>Read data flash memory to Expanded SRAM will always transfer only the first byte. |
| **O** | Abnormal Read/Write Data Flash memory | Accessing Data Flash memory | In enhanced method (FAM = 1), If the read/write flash region cover the location 0x7DFF (physical address), for example the starting address 0x7D80 (physical address) with ending address 0x7E2F (physical address), then only the range 0x7D80 to 0x7DFF can be accessed, no matter what value in TXZH/L, RCPTH/L, and bit 1 of FAM. [3]<br><br>The following location is applicable to this rule:<br>1) 0x7DFF<br>2) 0xFDFF |
| **P** | Abnormal Serial Communication | UART 0 and UART 1 Transmit | Don't modify Port B (byte addressable only) state when data is sending out on TXD0 / TXD1 pins. Otherwise, data will be corrupted. |
| | | | |

Remarks:
[1] An example on item b is shown below inside the red rectangle:

```
        ORG     0013H                   ;CA INT         IE.2
        SJMP    INITIAL                 ;
        ORG     001BH                   ;T1 OVER ONLY   IE.3
        LJMP    TIMER1_INTERRUPT        ;
        ORG     0023H                   ;SERIAL         IE.4
        SJMP    INITIAL                 ;
        ORG     002BH                   ;T2 INT         IE.5
        LJMP    TIMER2_INTERRUPT        ;
        ORG     0033H                   ;RESERVED       IE.6
        SJMP    INITIAL                 ;
        ORG     003BH                   ;SERIAL2        IE.7
        SJMP    INITIAL                 ;
;-------------------------------------------------------------------------------
        ORG     0040H                   ;
;-------------------------------------------------------------------------------

INITIAL:
        ; delay 100ms to let power stable
        ; in case the firmware write data flash at the beginning
        CALL    DELAY

        ; The following instructions to be added
        MOV     R0,#0FH
        MOV     A,#00111111B
        MOVX    @R0,A

INITIAL_1:
;       Customer Application start here
;       .
;       .
;       .
        JMP     INITIAL_1
```

[2] It depends on the application.
[3] Diagram is shown below for illustration



### 17.1.2 DC6688EMT Emulator

This applies only to the following version:
1 DC6688EMT-TS Rev1.x
2 DC6688EMT-1T Rev2.x

1 No backup mode
2 Only operated at 3.3V power
3 Peripherals
  3.1 When the emulator is stopped in debugging platform, all the running peripherals (e.g. timer 2) will still keep running. Hence, the peripherals will be out of synchronization with the code instruction.

### 17.1.3 In-System Programming

During programming, the flash memory has the following steps involved:
1. whole flash memory erase for security reason
2. write program flash memory
3. write customer information(Model/Version/Checksum)
4. read back program flash memory and customer information, and then verify byte by byte
5. lock program flash if required

# 18DC6688F2STR

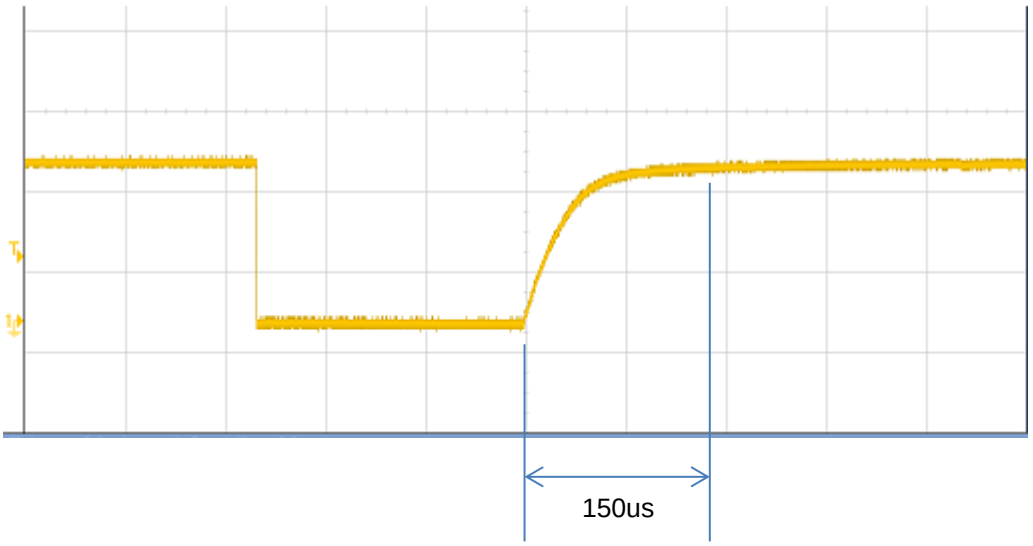
## 18.1 Firmware

| | Incident | Item | Description |
|---|---|---|---|
| A | Abnormal software operation | 'PORTC' register | Make sure bit 3~6 is '0' |
| B | Abnormal PC1 state | PCCONL[1,0] in XFR | 01 = prohibited |
| C | Abnormal Timer operation | Timer 0 and Timer 1 setting | The following condition at the same time happened is invalid:<br>i) Timer 1 set to mode 1, and<br>ii) Timer 0 set to mode 2 |
| D | Abnormal Data flash content | Initialization Data for Data Flash memory | To avoid using firmware to initialize Data Flash memory during in production line, it is recommended to also download the image to data flash memory by SL or SPI. |
| E | Abnormal Data flash content | Add 100ms delay at the beginning of the program | This applies to case where factory do initialization on data flash memory in production line by firmware itself. In view of the unstable power supply in production line found at the instant of insertion, add this delay at the beginning of the program to let the power stable before starting to run the firmware in production line.[1] |
| F | Abnormal software operation | Initialization for SRAM | This is a must as on power up, the contents in the SRAM are undetermined. |
| G | Software halt | Counter A reset | Counter A should be reset after exit from stop mode |
| H | Abnormal Interrupt | Bit 'IT0' in SFR | IT0 should set to 0 for PB/PC interrupt for proper operation |
| J | High stop mode Idd | I/O port configuration in stop mode | In order to meet the Stop mode Idd current as stated in specification, the I/O port should be carefully configured. The following is prohibited:<br>1) Input without pull high - drive low and high<br>2) Output open drain with pull high - drive low<br>3) Output open drain without pull high - drive high |
| K | Key detect failure in key scanning | Internal Pull-up 150kΩ | Make sure delay time 150us is inserted just after going back from logic 0 to 1 before reading port PB, PC0, PC2/3/4.[2] |

Remarks:
[1] An example on item b is shown below inside the red rectangle:

```
        ORG     0013H                   ;CA INT         IE.2
        SJMP    INITIAL                 ;
        ORG     001BH                   ;T1 OVER ONLY   IE.3
        LJMP    TIMER1_INTERRUPT        ;
        ORG     0023H                   ;SERIAL         IE.4
        SJMP    INITIAL                 ;
        ORG     002BH                   ;T2 INT         IE.5
        LJMP    TIMER2_INTERRUPT        ;
        ORG     0033H                   ;RESERVED       IE.6
        SJMP    INITIAL                 ;
        ORG     003BH                   ;SERIAL2        IE.7
        SJMP    INITIAL                 ;
;-------------------------------------------------------------------------------
        ORG     0040H                   ;
;-------------------------------------------------------------------------------

INITIAL:
        ; delay 100ms to let power stable
        ; in case the firmware write data flash at the beginning
        CALL    DELAY

        ; The following instructions to be added
        MOV     R0,#0FH
        MOV     A,#00111111B
        MOVX    @R0,A

INITIAL_1:
;       Customer Application start here
;       .
;       .
;       .
        JMP     INITIAL_1
```

[2] The timing diagram of PB0 (as an example) rising time due to internal pull-up




PB0 rising curve from 0 to 1 in key scanning

## 18.2 Hardware

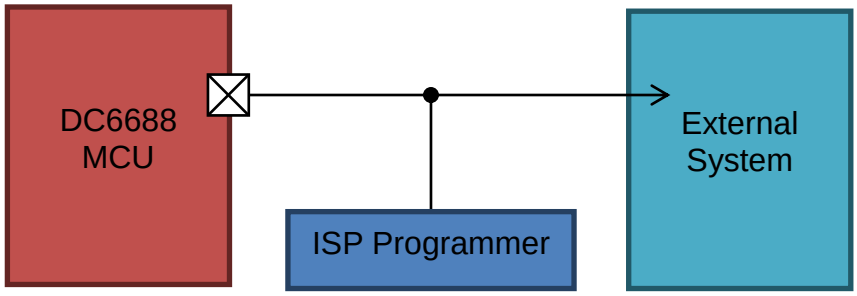
### 18.2.1 Programming pins

Precaution should be taken for the following I/O pins since they are shared with programming pins:

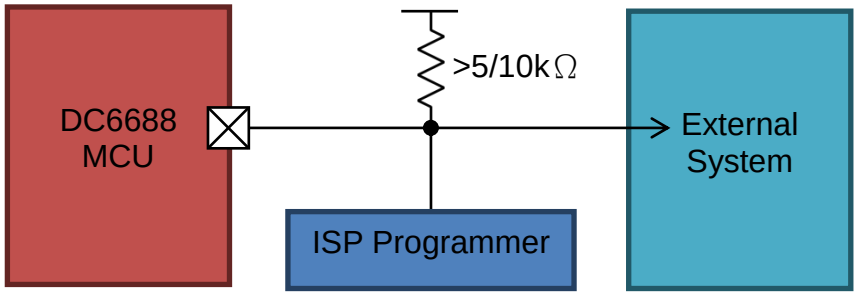
1 PC0
2 PB0

In order to do in-system-programming properly, below is a recommendation:

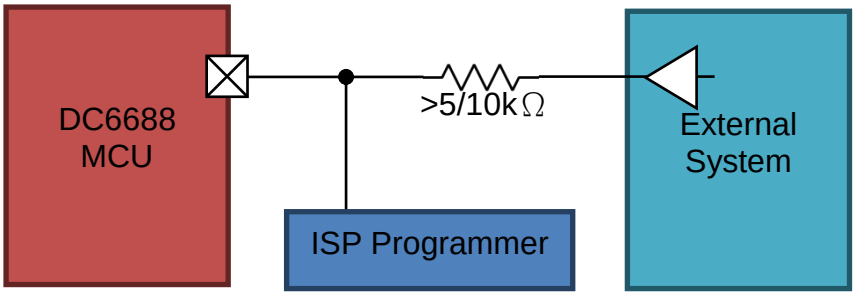
1 Push-pull output only
If the multiplexed IO is a push-pull output to external system's input, no additional isolation is needed.



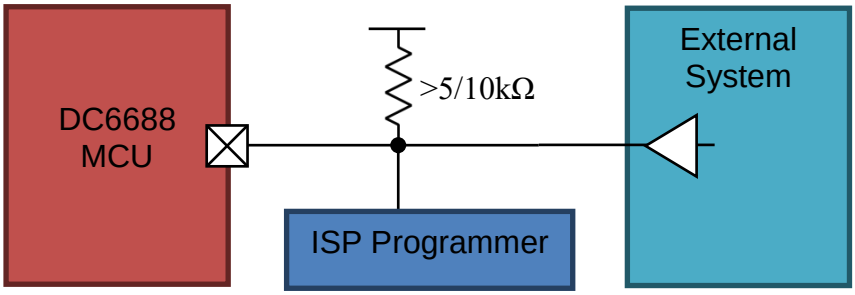

2 Open-drain output only w/ pull-up resistor
If the multiplexed IO is an open-drain output to external system's input with pulling resistor, the pulling resistors must not be under 5k Ohm for ISP of 1 to 2 devices and must not be under 10k Ohm for ISP of 3 or more devices together.



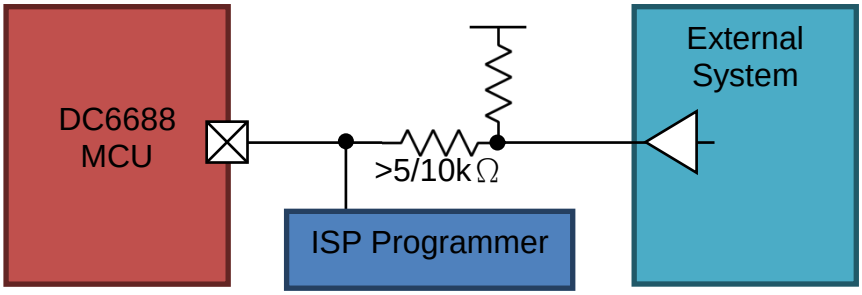

3 Input only or Bi-direction
If the multiplexed IO is an input to external system's output, an isolation resistor should be added to block the external system's driving affect the programming signal. The isolation resistors must not be under 5k Ohm for ISP of 1 to 2 devices and must not be under 10k Ohm for ISP of 3 or more devices together.




4 Input only or Bi-direction w/ pull-up resistor
If the multiplexed IO is an input to external system's output with pulling resistor, the pulling resistors must not be under 5k Ohm for ISP of 1 to 2 devices and must not be under 10k Ohm for ISP of 3 or more devices together.
If the external system will not drain the signal line during ISP, no additional isolation resistor is needed. (Case 1) On the other hand, if the signal line may be drained by the external system during ISP, an isolation resistor should be added as shown below with resistance value over 5k Ohm for ISP of 1 to 2 devices or over 10k Ohm for ISP of 3 or more devices together. (Case 2)




Case 5: External system will not drain during ISP

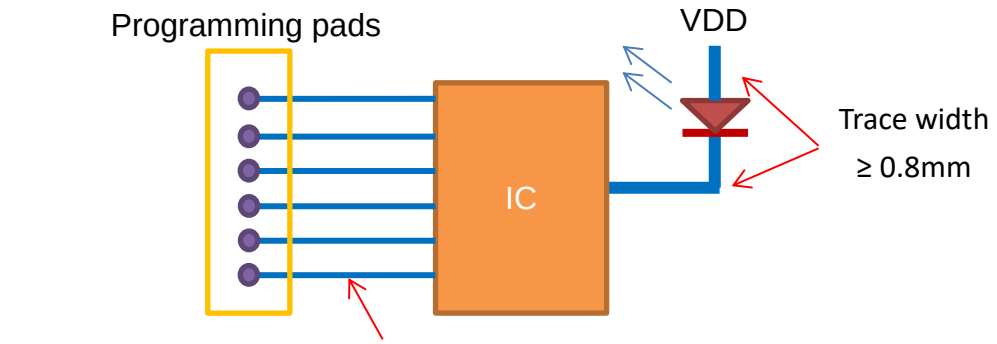



Case 6: External system might drain during ISP

On PC0, a pulse train will output on power up, it should be finished after 200ms

### 18.2.2 PCB layout

Below are some guidelines for the layout.




## 18.3 DC6688EMT Emulator

This applies only to the following version:
DC6688EMT-F2R Rev1.x

1 No backup mode
2 Only operated at 3.3V power
3 Peripherals
  3.1 When the emulator is stopped in debugging platform, all the running peripherals (e.g. timer 2) will still keep running. Hence, the peripherals will be out of synchronization with the code instruction.
4 Oscillator tolerance within +/-1%

## 18.4 In-System Programming

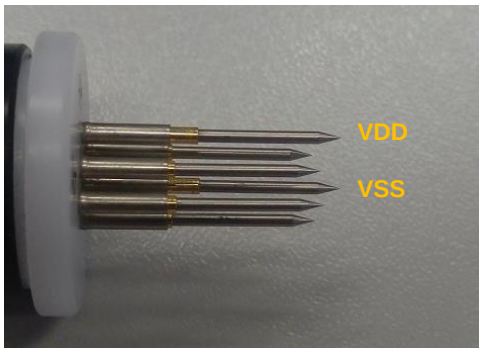
During programming, the flash memory has the following steps involved:
1. whole flash memory erase for security reason
2. write program flash memory
3. write customer information(Model/Version/Checksum)
4. read back program flash memory and customer information, and then verify byte by byte
5. lock program flash if required

## 18.5 Production highlights in customer factory

It is recommended to review the following items before production in customer site:
1 The prevention and control of Electrostatic Discharge (ESD)
2 PCBA level programming/trimming

2.1 Programming must be done at room temperature (25°C). It is advised to arrange 1 hour cooling between soldering reflow and programming/trimming on PCBA level
2.2 To obtain the best trimming performance, taking into consideration of PCB factor, it is recommended to do programming/trimming on PCBA level
2.3 Trimming will be completed during programming stage

3 Routine check VDD from programmer to IC below 3.8V
4 Use SLP Programmer (DC6688SLP-USB Rev3.2 or higher)
5 Use Software SLP Rev6.9.3 or higher
6 Spring test probe
6.1 VDD/VSS pin landing on PCB first as shown below




7 The IC should be powered by our programmer ONLY
7.1 The programmer control power off/on to ensure proper programming & verification operations.

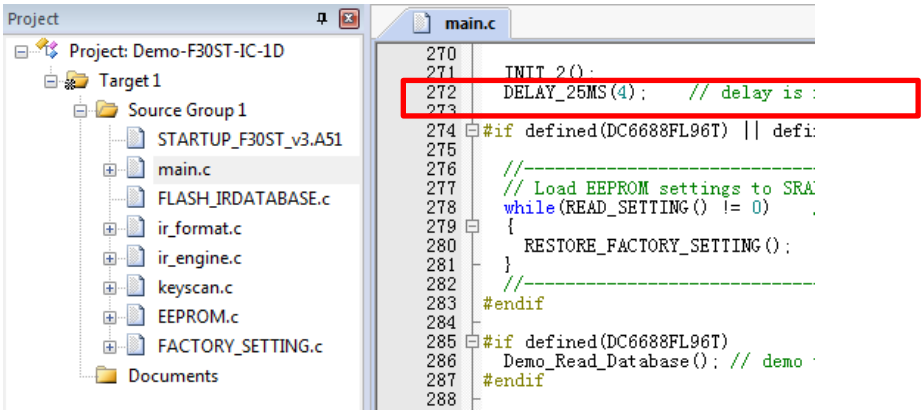
# 19DC6688BT

## 19.1 Firmware

| | Incident | Item | Description |
|---|---|---|---|
| A | Abnormal Timer operation | Timer 0 and Timer 1 setting | The following condition at the same time happened is invalid:<br>1) Timer 1 set to mode 1, and<br>2) Timer 0 set to mode 2 |
| B | Abnormal Data flash content | Initialization Data for Data Flash memory | To avoid using firmware to initialize Data Flash memory during in production line, it is recommended to also download the image to data flash memory by SL or SPI. |
| C | Abnormal Data flash content | Add 100ms delay at the beginning of the program | This applies to case where factory do initialization on data flash memory in production line by firmware itself. In view of the unstable power supply in production line found at the instant of insertion, add this delay at the beginning of the program to let the power stable before starting to run the firmware in production line. |
| D | Abnormal software operation | Initialization for SRAM | This is a must as on power up, the contents in the SRAM are undetermined. |
| E | Software halt | Counter A reset | Counter A should be reset after exit from stop mode |
| F | Abnormal Interrupt | Bit 'IT0' in SFR | IT0 should set to 0 for PA interrupt for proper operation |
| G | Abnormal Interrupt [2] | Accessing Data Flash memory | 1) CPU suspends operation<br>2) only one interrupt is seen by CPU no matter how many interrupts from a peripheral<br>3) any interrupt will be delayed to serve by CPU until completion of accessing data flash<br>4) Disable watchdog before accessing data flash |
| H | High stop mode Idd | I/O port configuration in stop mode | In order to meet the Stop mode Idd current as stated in specification, the I/O port should be carefully configured. The following example regards as "floating" and is prohibited:<br>1) Input without pull high / down<br>2) Output n-channel open drain with pull high - drive low<br>3) Output n-channel open drain without pull high - drive high<br><br>In addition, the un-used / un-bonded I/O port must also be initialized. [3] |

| | | | |
|---|---|---|---|
| J | Abnormal read Data flash | Set RCPTH/RCPTL in Traditional Method | Before reading/writing Data flash with Traditional Method, RCPTH/RCPTL .should be defined. |
| K | Abnormal T24 capture | T24 capture reset | To avoid insufficient time for capture reset, T24_CON1 should be configured in the following order in each capture reset::<br>1) T24_CON1 = 0xA0; first, bit 5 set to 1<br>2) T24_CON1 = 0x8E; then, enable pointer x capture as you want, where x = A, B or C. In this example, enable all. |
| M | Abnormal Serial Communication | UART 0 and UART 1 Transmit | Don't modify Port B (byte addressable only) state when data is sending out on TXD0 / TXD1 pins. Otherwise, data will be corrupted. |
| N | Abnormal IR Learning | I/O interference | Toggle any I/O during IR learning would interfere the op amp to receive IR signal. Thus, I/O states should keep unchanged. |
| P | Abnormal IR Learning | Interrupted by other tasks | During learning, other tasks' associated interrupts should be disabled. Otherwise, timing will be corrupted. |

Remarks:
[1] An example on item b is shown below inside the red rectangle:




[2] It depends on the application.
[3] Un-bonded I/O port (orange) is shown below. It should not be configured as "floating".
BT96

| QFN56 | Pin Name |
|---|---|
| - | PD6 |

BT32

| QFN32 | Pin Name |
|---|---|
| - | PB3 |

It is suggested that at the beginning of the program, the un-used / un-bonded I/O should be configured as either:

1 Input with pull up, or
2 Output push-pull

## 19.2 Hardware

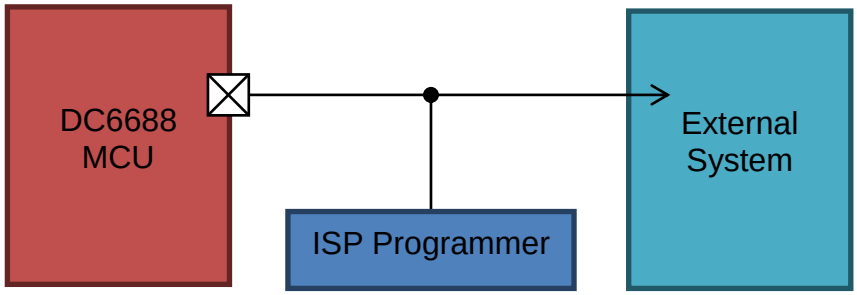
### 19.2.1 Programming pins

Precaution should be taken for the following I/O pins since they are shared with programming pins:

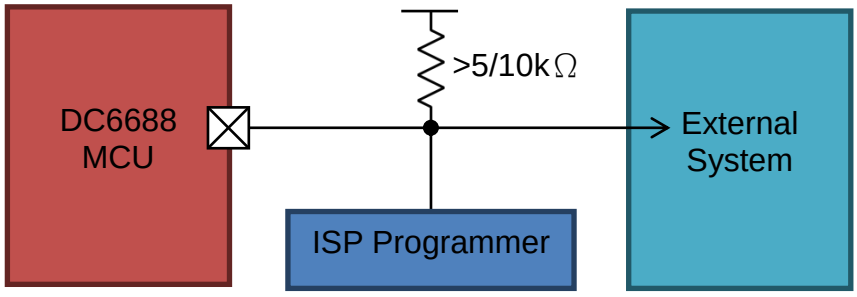
1 PD1
2 PD0
3 PB6

In order to do in-system-programming properly, below is a recommendation:

1 Push-pull output only
If the multiplexed IO is a push-pull output to external system's input, no additional isolation is needed.

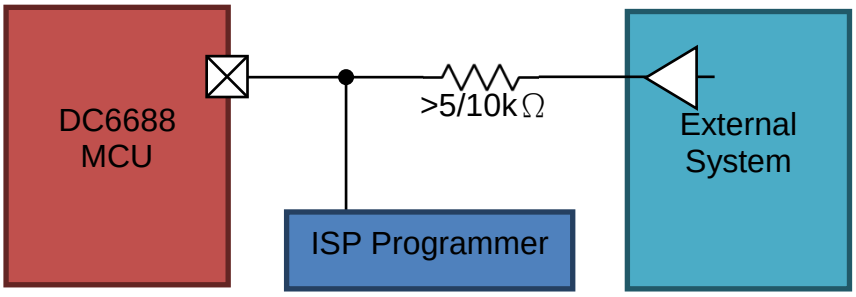



2 Open-drain output only w/ pull-up resistor
If the multiplexed IO is an open-drain output to external system's input with pulling resistor, the pulling resistors must not be under 5k Ohm for ISP of 1 to 2 devices and must not be under 10k Ohm for ISP of 3 or more devices together.




3 Input only or Bi-direction
If the multiplexed IO is an input to external system's output, an isolation

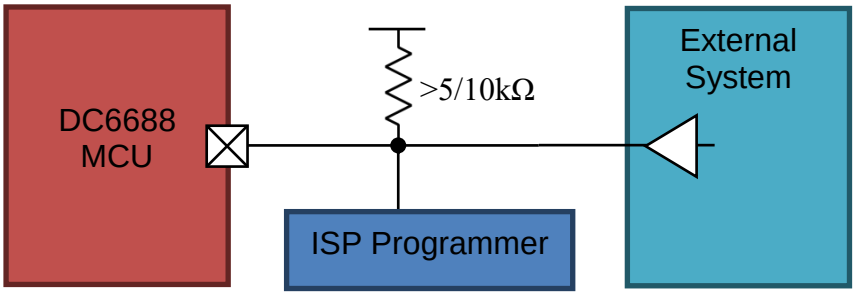
resistor should be added to block the external system's driving affect the programming signal. The isolation resistors must not be under 5k Ohm for ISP of 1 to 2 devices and must not be under 10k Ohm for ISP of 3 or more devices together.



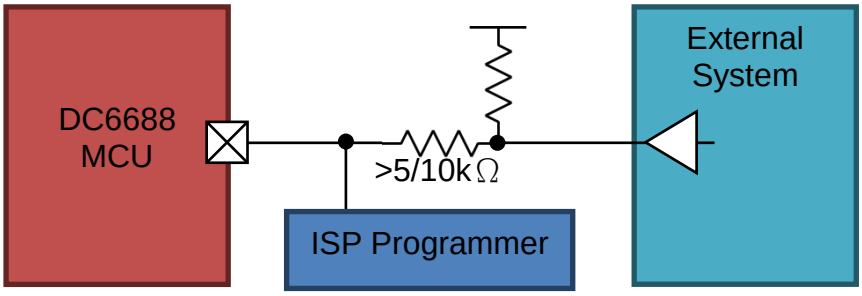

4 Input only or Bi-direction w/ pull-up resistor

If the multiplexed IO is an input to external system's output with pulling resistor, the pulling resistors must not be under 5k Ohm for ISP of 1 to 2 devices and must not be under 10k Ohm for ISP of 3 or more devices together.

If the external system will not drain the signal line during ISP, no additional isolation resistor is needed. (Case 1) On the other hand, if the signal line may be drained by the external system during ISP, an isolation resistor should be added as shown below with resistance value over 5k Ohm for ISP of 1 to 2 devices or over 10k Ohm for ISP of 3 or more devices together. (Case 2)




Case 7: External system will not drain during ISP

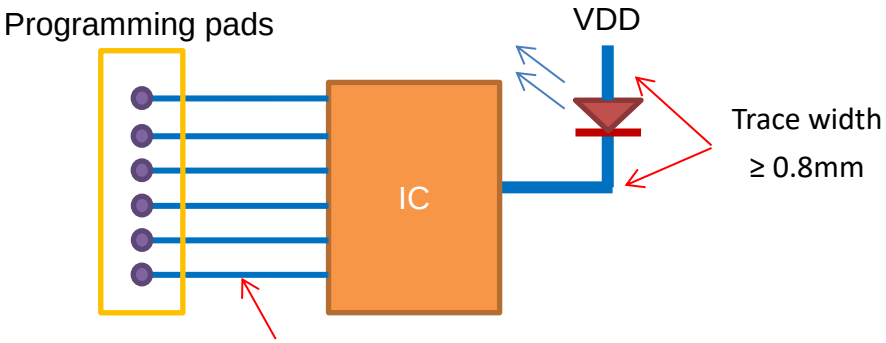



Case 8: External system might drain during ISP

On PD1, a pulse train will output on power up, it should be finished after 200ms

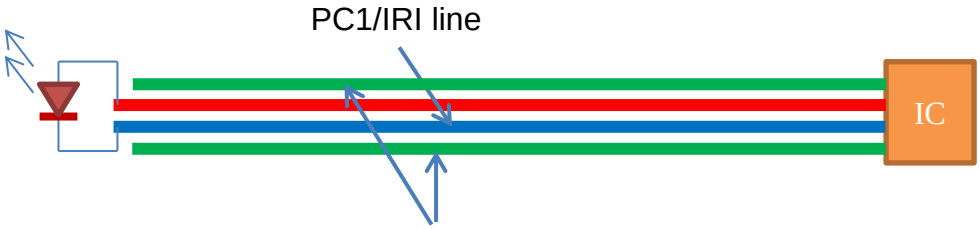
### 19.2.2 PCB layout

Below are some guidelines for the layout.



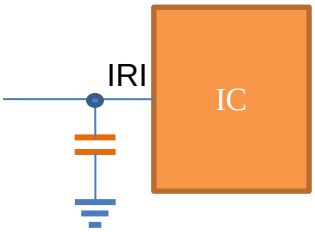

No carbon film between pin and pad, and length < 3cm

One-diode solution (IR Transmit + Learning)




VSS plate/line should be close to and in parallel with PC1/IRI

Maximum of 500pF capacitor on IRI pin can help filter the noise while reducing the sensitivity. The capacitor must be close to IRI pin.




## 19.3 DC6688EMT Emulator

This applies only to the following version:

1 DC6688EMT-BT Rev1.x

1 No backup mode
2 Only operated at 3.3V power
3 Peripherals
  3.1 When the emulator is stopped in debugging platform, all the running peripherals (e.g. timer 2) will still keep running. Hence, the peripherals will be out of synchronization with the code instruction.
4 Oscillator tolerance within +/-1%

## 19.4 In-System Programming

During programming, the flash memory has the following steps involved:
1. whole flash memory erase for security reason
2. write program flash memory
3. write customer information(Model/Version/Checksum)
4. read back program flash memory and customer information, and then verify byte by byte
5. lock program flash if required

## 19.5 Production highlights in customer factory

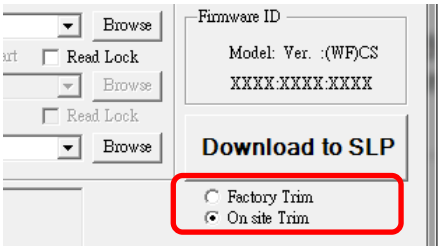
It is recommended to review the following items before production in customer site:

1 The prevention and control of Electrostatic Discharge (ESD)

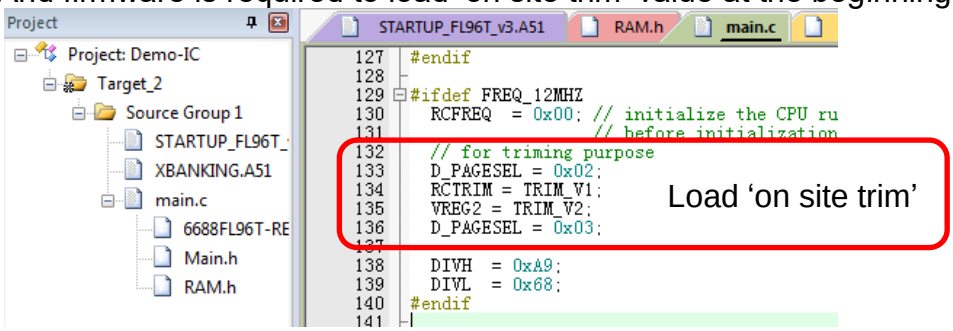
2 PCBA level programming/trimming

2.1 Programming must be done at room temperature (25°C). It is advised to arrange 1 hour cooling between soldering reflow and programming/trimming on PCBA level

2.2 To obtain the best trimming performance, taking into consideration of PCB factor, it is recommended to do programming/trimming on PCBA level




And firmware is required to load 'on site trim' value at the beginning.




| Software SLP setting | Firmware | |
|---|---|---|
| | Load 'on site trim' | Load 'factory trim'[1] |
| Factory Trim | Prohibited | Use 'factory trim' |
| On site Trim | Use 'on site trim' | Use 'factory trim' |

Remarks:
[1] Load 'factory trim' is done by disabling to load 'on site trim'.

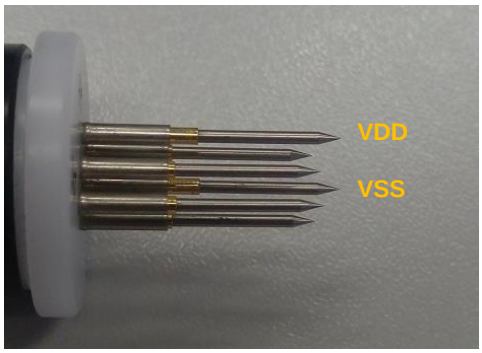
2.3 Trimming will be completed during programming stage

3 Routine check VDD from programmer to IC below 3.8V

4 Use SLP Programmer (DC6688SLP-USB Rev3.2 or higher)

5 Use Software SLP Rev6.9.3 or higher

6 Spring test probe
- 6.1 Add 10k ohm across PB6 and VDD
- 6.2 Add 10k ohm across PROG/ISPSEL and VDD
- 6.3 VDD/VSS pin landing on PCB first as shown below




7 The IC should be powered by our programmer ONLY
- 7.1 The programmer control power off/on to ensure proper programming & verification operations.

# 20Revision History

| Document Rev. No. | Issued Date | Section | Page | Description | Edited By | Reviewed By |
|---|---|---|---|---|---|---|
| 1.0 | Jun, 2008 | | | Preliminary | - | - |
| 1.1 | Jun, 2008 | | | Renumber DC6688FLX | - | - |
| 1.2 | July, 2008 | 7 | | Add DC6688FLB | Danny Ho | Ken Yeung |
| | | 4.1 | | Revised table | | |
| | | 6.1 | | Revised table | | |
| 1.3 | July, 2008 | 6.2.2 | | Add section | Danny Ho | Ken Yeung |
| | | 6.2.3 | | Add section | | |
| | | 4.2.2 | | Add section | | |
| | | 5.2.2 | | Add section | | |
| 1.4 | July, 2008 | 4.2.3 | | Add section “Precaution when debugging” | Danny Ho | Ken Yeung |
| | | 4.2.4 | | Add section “Additional Information” | | |
| | | 6.2.4 | | Add section “Precaution when debugging” | | |
| | | 6.2.5 | | Add section “Additional Information” | | |
| | | 7.2.2 | | Add section “Environment Setting” | | |
| | | 7.2.3 | | Add section “Precaution when debugging” | | |
| | | 7.2.4 | | Add section “Additional Information” | | |
| 1.5 | July, 2008 | All | | Revise Product name | Danny Ho | Kennis To |
| | | 2.2.3 | | Add section “Additional Information” | | |
| | | 4.2.4 | 18 | Add Data flash content in emulator | | |
| | | 5.2.3 | | Add section “Additional Information” | | |
| | | 3.1 | | Revise item C “Abnormal Serial Communication | | |
| | | | | Add item K | | |
| 1.6 | July, 2008 | 2.3, 3.3, 4.3, 5.3, 6.3, 7.3 | | In-System Programming | Danny Ho | Kennis To |
| 1.7 | Aug, 2008 | | | Add DC6688FSB | Kennis To | Ken Yeung |
| 1.8 | Aug, 2008 | 7.1 | | Revise section 7.1 | Kennis To | Danny Ho |
| 1.9 | Aug, 2008 | 7.1 | | Revised section 7.1 item J | Danny Ho | Ken Yeung |
| 2.0 | Sept, 2008 | 8 | | Change FLB to FL16B<br>Add FL32B | Kennis To | Ken Yeung |
| 2.1 | Nov, 2008 | 8.1 | | Add remark for watchdog function | Kennis To | Ken Yeung |
| 2.2 | Jan, 2009 | | | Correct revision history of Rev2.1<br>Add DC6688F2SCN | Ken Yeung | |
| 2.3 | Feb, 2009 | 5.1 | | Add Item J | Kennis To | Danny Ho |
| | | 7.1 | | Add Item K | | |
| | | 8.1 | | Add Item L | | |
| 2.4 | Feb, 2009 | n) 7.2.5 | | Add FL48X description in item3 | Danny Ho | Kennis To |
| | | | | Add more explanation on difference between emulator and IC | | |
| 2.5 | May, 2009 | 7.1 | | Add item L on LVI | Danny Ho | Kennis To |
| 2.6 | Jun, | | | Revise format of the document | Kennis To | Danny Ho |

| | | | | | | |
|---|---|---|---|---|---|---|
| | 2009 | | | | | |
| 2.7 | Jul, 2010 | | | Revise emulator limitation | Kennis To | Danny Ho |
| 2.8 | Apr, 2011 | All | | Update table | Danny Ho | Celia Ki |
| 2.9 | June, 2012 | 11 | | Add DC6688FLE | Danny Ho | Celia Ki |
| 3.0 | Nov, 2012 | All | | Update item in section 'Firmware' | Danny Ho | Celia Ki |
| | | | | Add DC6688FLT | | |
| 3.1 | Apr, 2014 | 11, 12 | | Update item in section 'Firmware' | Danny Ho | Celia Ki |
| | | 12 | | Add FL32T description | | |
| | | 13 | | Add FST description | | |
| 3.2 | June, 2014 | 14 | | Add F2SER description | Danny Ho | Philip Hung |
| | | 15 | | Add F2SEN description | | |
| 3.4 | Oct, 2014 | 12 | | Add FL96T / FL64T | Danny Ho | Philip Hung |
| 3.5 | Apr, 2015 | 17 | | Add F2STR description | Danny Ho | Philip Hung |
| 3.6 | Nov, 2015 | 14.1, 17.1 | | Add Data flash memory related highlights | Danny Ho | Patrick Li |
| 3.7 | May, 2015 | 12, 13 | | Add section "Production highlights in customer factory" | Danny Ho | Patrick Li |
| 3.8 | May 2016 | 12.4, 13.4, 17.4 | | Revise description | Danny Ho | Patrick Li |
| 3.9 | Aug 2016 | 12.5, 13.5, 17.5 | | Add description on IC power<br>Use Software SLP Rev6.9.3 | Danny Ho | Patrick Li |
| | | 12.2<br>13.2<br>17.2 | | Add hardware section for programming pins highlight | | |
| 4.0 | Sept 2017 | 2 | | Insert a section for Dragonchip Tools | Danny Ho | Patrick Li |
| | | 13.5<br>14.5 | | Add description for trimming | | |
| | | | | Update cover page | | |
| 4.1 | Mar, 2018 | 19 | | Add DC6688BT | Danny Ho | Patrick Li |
| 4.2 | Jun, 2018 | 2.1 | | Update link | Danny Ho | Patrick Li |
| 4.3 | Jul, 2018 | 13.1<br>14.1 | | Revise description on TSSOP20, SOP16 | Danny Ho | Patrick Li |
| 4.4 | Dec, 2018 | 12.2, 13.2, 14.2, 15.2, 18.2, 19.2 | | Add description | Danny Ho | Patrick Li |
| 4.5 | Mar, 2019 | 13.2.2<br>19.2.2 | | Add PCB layout | Danny Ho | Patrick Li |
| 4.6 | Apr, 2019 | 7.2, 7.3 | | Add layout guidelines and emulator description | Danny Ho | Patrick Li |
| | | 7.5, 12.5 | | Add production precaution | | |
| 4.7 | May, 2020 | 13.1<br>13.2.2,<br>19.1,<br>19.2.2. | | Add description on IR learning | Danny Ho | Patrick Li |
| | | | | | | |

**Copyright Notice**



**Disclaimer**

Dragonchip Ltd. assumes no responsibility for any errors contained herein.


Dragonchip Ltd.
TEL: (852) 2776-0111
FAX: (852) 2776-0996
http://www.dragonchip.com