

# (Technical)

Document No. : MJL-LD-AN-4

Author: [twha@mjl.co.kr]

Version: 1.0

Date: 2000 3 31

Subject: Microprocessor FPGA Passive Serial Configuration.

1. FPGA Configuration data (EPC2, ) FPGA Serial PROM (EPC1441, EPC1, )

2. PCB Board Microprocessor 가 (FPGA configuration data ) System ROM 가  
 , Microprocessor 가  
 Microprocessor FPGA Configuration data 가

3. Altera FPGA (APEX 20K/E, FLEX 10K/E/S/, FLEX 6000) Microprocessor Passive serial configuration 가

4. CPU (Intel 8051 ) FLEX 6000 Series FPGA Configuration data

5. 가 Microprocessor  
 Microprocessor Object Code Compiler  
 Microprocessor FPGA Passive Serial Communication Program

6. Timing information Web site

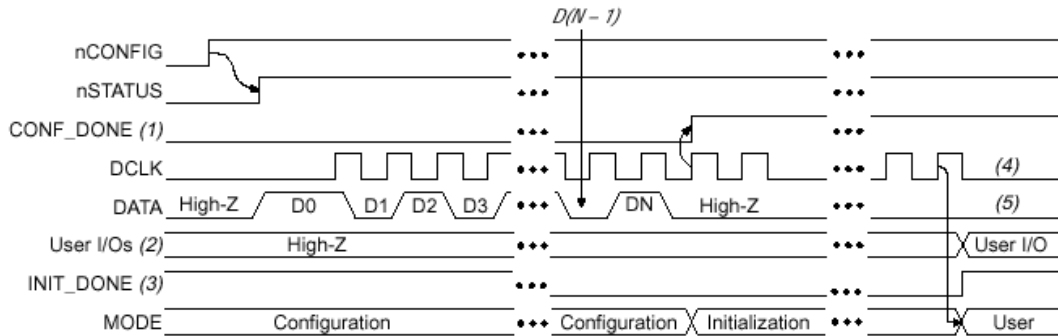
- : AN 116 (Configuring APEX 20K, FLEX 10K & FLEX 6000 Devices), ver. 1.02, December 1999
- : <http://www.altera.com/document/an/an116.pdf>
- : Here is a "altera.PLDWorld.com"... / Tool Manual...
- : <http://www.pldworld.com/@altera/html/technote/toolmanual.htm>

## Device Configuration Overview

가 , APEX 20K, FLEX 10K, FLEX 6000  
 Configuration data SRAM Cell . SRAM Memory

SRAM Cell, APEX 20K, FLEX 10K, Register I/O pin In-system operation, Configuration data, FLEX 6000, (Configuration), User mode, Configuration, Initialization, User mode

**Figure 1. APEX 20K, FLEX 10K & FLEX 6000 Configuration Cycle**



**Notes:**

- (1) During initial power-up and configuration, CONF\_DONE is low. After configuration, CONF\_DONE goes high. If the device is reconfigured, CONF\_DONE goes low after nCONFIG is driven low.
- (2) User I/O pins are tri-stated during configuration. APEX 20K and FLEX 10KE devices also have a weak pull-up resistor on I/O pins during configuration. After initialization, the user I/O pins perform the function assigned in the user's design.
- (3) When used, the optional INIT\_DONE signal is high when nCONFIG is low before configuration and during approximately the first 40 clock cycles of configuration.
- (4) DCLK should not be left floating. It should be driven high or low, whichever is more convenient.
- (5) DATA (FLEX 6000 devices) and DATA0 (APEX 20K and FLEX 10K devices) should not be left floating. It should be driven high or low, whichever is more convenient.

APEX 20K, FLEX 10K, FLEX 6000 Configuration data Active Passive configuration scheme Configuration device 가 Active configuration scheme Target device Configuration device Control Synchronization signal 가 Configuration 가 Configuration device Data APEX 20K, FLEX 10K, FLEX 6000

Passive configuration scheme APEX 20K, FLEX 10K, FLEX 6000 Configuration process Intelligent host Microprocessor 가 Host (Hard Disk, RAM, ) Configuration data Passive configuration Reconfigure Target device configuration data ( ) Programming file In-field upgrade

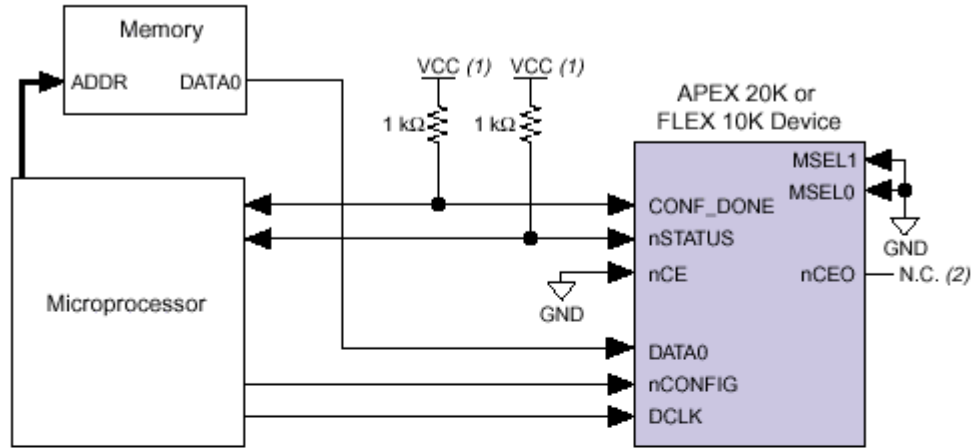
**PS(passive Serial) Configuration with a Microprocessor**

Microprocessor 가 PS Configuration Microprocessor Storage device Data Target APEX 20K, FLEX 10K, FLEX 6000 device Programming hardware Configuration Microprocessor nCONFIG pin Low-to-high transition Target device nSTATUS pin Release

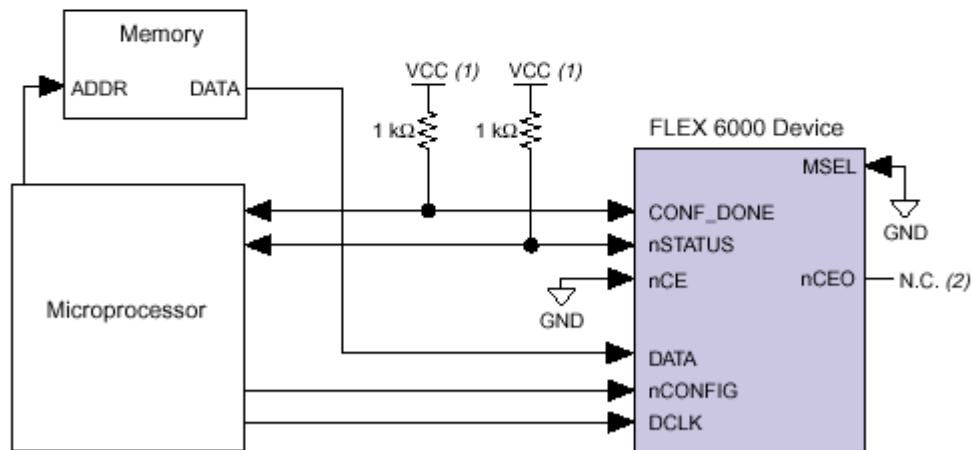
Microprocessor Programming hardware Target device (the DATA0 pin for APEX 20K and FLEX 10K devices, and the DATA pin for FLEX 6000 devices) DATA pin Time Configuration data one bit Data byte Least significant bit(LSB)가 Target device Clocking Data CONF\_DONE 가 High Target device Data 가 , DCLK FLEX 10K FLEX 6000 device 가 Device CONF\_DONE High Device Configuration High pin Device Configuration High pin Max+plus II Quartus software Bits Configuration file 가 Configuration Device DCLK Configuration file Target device Target device Configuration Initialize Configuration Handshaking signal PS Configuration mode Configuration Configuration clock speed Configuration Frequency Configuration DCLK Halt Configuration Target device 가 Configuration Error , Target device Microprocessor Configuration process nSTATUS pin Low Microprocessor nCONFIG pin Low-to-high pulse MAX+plus II Quartus software "Auto-Restart Configuration on Frame Error" option , Target device Reset time-out period nSTATUS Release nSTATUS 가 Release Microprocessor nCONFIG Low Pluse Target device Reconfiguration Microprocessor Configuration CONF\_DONE INIT\_DONE pin Configuration Microprocessor 가 data Initialization clock CONF\_DONE High 가 , Microprocessor Target device Reconfiguration Configuration diagram

**Figure 11. PS Configuration Circuit with Microprocessor**

**APEX 20K or FLEX 10K Devices**



**FLEX 6000 Devices**



**Notes:**

- (1) The pull-up resistor should be connected to any  $V_{CC}$  that meets the device high-level input voltage ( $V_{IH}$ ) specification.
- (2) The nCEO pin is left unconnected.

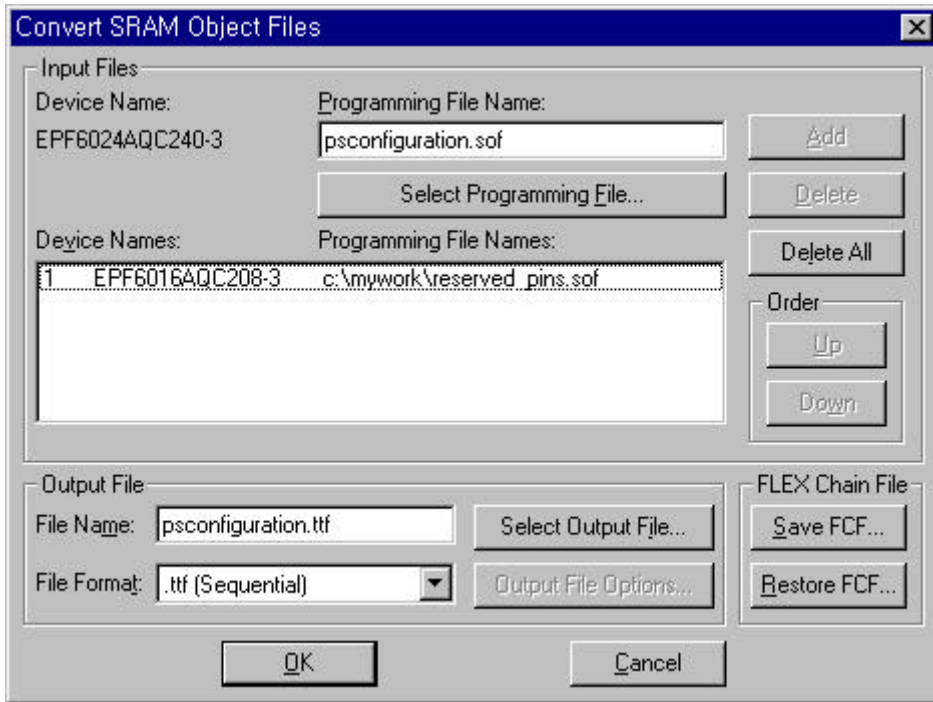
**INTEL 8051 Microprocessor FLEX6000 Passive Serial Configuration**

AN116.PDF

가  
가

1) EPF6016AQC240-3 8051 Microprocessor

- 2) FPGA 가 Microprocessor 가 Configuration data
- 3) Compile \*.sof \*.pof file
- 4) MAX+plus II Pull-down menu File -> Convert SRAM Object Files..



1 Convert SRAM Object Files

- 5) 3 Output File File Format .ttf(Sequential) OK button

- 6) MAX+plus II Software Compile 3 가 Configuration data 가

\*.sof ; ByteBlasterMV cable Configuration Data  
 \*.pof ; Configuration PROM Data  
 \*.ttf ; Passive Configuration with a Microprocessor Data  
 file \*.ttf 4), 5) Sequential type ttf

- 7) \*.ttf file

- 8) Microprocessor Compiler 가 Data file "hex\_conv.exe"  
 Program \*.ttf file DB (Define Byte) file

a. Windows 98 / Windows NT DOS

- b. DOS hex\_conv.exe 가 Directory .
- c. \*.tff file hex\_conv.exe 가 Directory Copy .
- d. DOS prompt hex\_conv.exe . 가  
\*.tff fpga.tff 가 .
- 9) Microprocessor SYSTEM ROM FPGA Configuration  
data 가 .  
"C language Assembler language" Microprocessor  
FPGA .

```

;;;*** altera Configuration interface
pga_dclk    bit        p1.0
pga_data    bit        p1.1
pga_status  bit        p1.2
pga_conf_done bit      p1.3
pga_init_done bit      p1.4
pga_conf    bit        p1.5
;;;;;;;;;;;;;;;;; u-com Emulator Test sub-routine Kevin
;;
;;          jmp      restart
;;
;;;;;;;;;;;;;;;;; u-com Emulator Test sub-routine Kevin
fpga_init:
        mov     r0,#05
        mov     r1,#03
        clr     pga_dclk
        clr     pga_data
fpga_conf_init_loop:
;;      call    pga_clock
;;      djnz   r0,fpga_conf_init_loop
fpga_conf_init_clr:
        clr     pga_conf
        clr     pga_status
fpga_conf_init_clr_loop:
;;      call    pga_clock
;;      djnz   r1,fpga_conf_init_clr_loop
        nop
        nop
        nop
        nop
        nop
        setb    pga_conf
;;      setb    pga_status
fpga_status_wait:
        jnb     pga_status,fpga_status_wait
        mov     r0,#0ffh
fpga_fir_data_wait:
        djnz   r0,fpga_fir_data_wait
        mov     dptr,#fpga_code_data_end
        mov     temp1,dpl
        mov     temp2,dph
fpga_data_loading_begin:
        mov     dptr,#fpga_code_data
        clr     a

```

```

fpga_data_loading_loop:
    mov     r0,#08
    clr     a
    movc    a,@a+dptr
fpga_data_down_loop:
    rrc     a
    mov     pga_data,c
    setb    pga_dclk
    clr     pga_dclk
;;
    call    pga_clock
    djnz    r0,fpga_data_down_loop
    inc     dptr
    mov     a,dpl
    jnb     pga_status,fpga_init
    cjne    a,temp1,fpga_data_loading_loop
    mov     a,dph
    cjne    a,temp2,fpga_data_loading_loop
;;
    jnb     pga_conf_done,fpga_init
    mov     r0,#45
fpga_down_init_done:
    setb    pga_dclk
    clr     pga_dclk
    djnz    r0,fpga_down_init_done
    setb    pga_dclk
;;
    jnb     pga_init_done,fpga_init
restart:
    clr     TR0      ; run tmr 0
    clr     TR1      ; TMR 1 DO NOT RUN
;.....;
;; FPGA.ASM FILE
;; FILE INCLUDE ROUTINE
fpga_code_data:

#include(fpga.ttf)

fpga_code_data_end:

```

iNTEL8051 Assembler language

가            FPGA    Microprocessor  
           가

11            Microprocessor    FPGA

## Revision History

---

- 2000-3-31 - Ver 1.0: Initialize Release...
- 2000-4-14 – Ver 1.1: 가... (Modified by C.W.Yang)