PLLCON PAGE 1

1 ;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

2 ;

3 ; Author : ADI - Apps www.analog.com/MicroConverter

4 ;

5 ; Date : January 2001

6 ;

7 ; File : PLLcon.asm

8 ;

9 ; Hardware : ADuC816

10 ;

11 ; Description : Demonstrates that the CPU can run at different

12 ; speeds determined by the CD bits in the PLLCON SFR.

13 ; 2 to the power of CD (a 3 bit number), is the divider

14 ; ratio that determines the clock frequency at which

15 ; the CPU will run. (CD=0 =>fcore=12.58MHz,

16 ; CD=7 => fcore=98.3kHz)

17 ;

18 ; The program turns on and off the LED approx every

19 ; 70,000 machine cycles. With the higher frequency

20 ; (CD=0 =>fcore=12.58MHz) the LED toggles at about

21 ; 16Hz. By pressing the INT0 button the CD bit is

22 ; incremented (CD=1 =>fcore=6.3MHz) and the LED will

23 ; toggle at half the frequency as before. At the

24 ; minimum frequency (CD=7, fcore=98kHz) the LED

25 ; toggles at 0.12Hz (toggles every 8.5s). By pressing

26 ; INT0 button again CD rolls over to 0 again and the

27 ; LED toggles at 16Hz again.

28 ;

29 ;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

30

31 $MOD816 ; Use 8052&ADuC816 predefined symbols

32

00B4 33 LED EQU P3.4 ; P3.4 drives red LED on eval board

34

35

36 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

37 ; BEGINNING OF CODE

---- 38 CSEG

39

0000 40 ORG 0000h

41

0000 020060 42 JMP MAIN ; jump to main program

43 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

44 ; INTERRUPT VECTOR SPACE

0003 45 ORG 0003h ; (INT0 ISR)

46

0003 B2B4 47 CPL LED ; complemant LED to indicate INT0

48 ; press.

0005 7A88 49 MOV R2,#136 ; reinitialise R7 and R6 so that

0007 7B00 50 MOV R3,#256 ; after interrupt the full delay

51 ; loop is completed

52

0009 E5D7 53 MOV A, PLLCON ; Only increment CD bits of PLLCON

000B 04 54 INC A ; Rollover to PLLCON = xxxxx000b (fmax)

000C 5407 55 ANL A, #07h ; after PLLCON = xxxxx111b (fmin)

000E F5D7 56 MOV PLLCON, A ; where the x's are 1's and 0's as rqd

57

0010 32 58 RETI

PLLCON PAGE 2

59

60 ;====================================================================

61

0060 62 ORG 0060H ; Start code at address above interrupts

0060 63 MAIN:

0060 75D700 64 MOV PLLCON, #00H

0063 D288 65 SETB IT0

0065 D2A8 66 SETB EX0 ; enable ext int INT0

67 ; (button on eval board)

0067 D2AF 68 SETB EA ; enable interrupts

69

0069 B2B4 70 BLINK: CPL LED

006B 120070 71 CALL DELAY ; wait for 70,000 machine cycles

72 ; =66ms at fmax

73 ; =8.5s at fmin

006E 80F9 74 JMP BLINK

75

76

77 ;====================================================================

0070 78 DELAY: ; This loop delays the program for 70,000

79 ; (approx) machine cycles, corresponding

80 ; to a delay of 66ms at fmax and 8.4s

81 ; at fmin

82

0070 7A88 83 MOV R2,#136 ; 136 \* 256 \* 1.907us = 66ms

0072 7B00 84 DLY1: MOV R3,#256 ;

0074 DBFE 85 DJNZ R3,$ ; sit here for 256 x 2 x machine

86 ; cycle time (=488us @ fmax)

0076 DAFA 87 DJNZ R2,DLY1 ; repeat 136 times (=66ms total @ fmax)

0078 22 88 RET

89

90 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

91

92 END

VERSION 1.2h ASSEMBLY COMPLETE, 0 ERRORS FOUND

PLLCON PAGE 3

BLINK. . . . . . . . . . . . . . C ADDR 0069H

DELAY. . . . . . . . . . . . . . C ADDR 0070H

DLY1 . . . . . . . . . . . . . . C ADDR 0072H

EA . . . . . . . . . . . . . . . B ADDR 00AFH PREDEFINED

EX0. . . . . . . . . . . . . . . B ADDR 00A8H PREDEFINED

IT0. . . . . . . . . . . . . . . B ADDR 0088H PREDEFINED

LED. . . . . . . . . . . . . . . NUMB 00B4H

MAIN . . . . . . . . . . . . . . C ADDR 0060H

P3 . . . . . . . . . . . . . . . D ADDR 00B0H PREDEFINED

PLLCON . . . . . . . . . . . . . D ADDR 00D7H PREDEFINED