```
#include <1pc214x.h>
          #define LED OFF (IOOSET = 1U << 31)
          #define LED_ON (IOOCLR = 1U << 31)</pre>
  4
 5
          #define PLOCK 0x00000400 // Bit mask for checking PLL lock status
 6
 7
         void systeminit(void);
 8
         void delay_ms(unsigned int t);
 9
10
         int main(){
11
                  unsigned int no clk = 100, no aclk = 100;
12
                  IOODIR |= (1U << 31) | (0xFF << 16) | (1U << 30);
13
                  LED ON;
14
                  delay ms(500);
15
                 LED OFF;
16
                 systeminit();
17
                  do{
18
                      IOOCLR = 0xF << 16; IOOSET |= 1 << 16; delay ms(10); if(--no clk ==0) break;
19
                      IOOCLR = 0 \times F << 16; IOOSET |= 1 << 17; delay ms(10); if(--no clk ==0) break;
2.0
                      IOOCLR = 0xF<<16; IOOSET |= 1<<18; delay_ms(10); if(--no_clk ==0) break;
21
                      IOOCLR = 0 \times F << 16; IOOSET |= 1 << 19; delay ms(10); if(--no clk ==0) break;
22
                  } while (1);
23
                  do{
                      IOOCLR = 0xF << 16; IOOSET \mid = 1 << 19; delay ms(10); if(--no aclk == 0) break;
24
25
                      IOOCLR = 0xF<<16; IOOSET |= 1<<18; delay_ms(10); if(--no_aclk ==0) break;
26
                      IOOCLR = 0xF << 16; IOOSET = 1 << 17; IOOSET = 10; IOOSET 
27
                      IOOCLR = 0xF << 16; IOOSET = 1 << 16; delay ms(10); if(--no aclk == 0) break;
28
                  } while (1);
                  IOOCLR = 0xFF << 16;
29
30
                  while(1)
31
32
33
34
        void delay ms(unsigned int t) {
3.5
              unsigned int i,j;
36
              for(i=0; i<t; i++)</pre>
37
                  for(j=0; j<10000; j++);
38
          }
39
40
41
         void systeminit(void) {
                                                              // Enable the PLL (PLLE = 1)
42
                 PLLOCON = 0x01;
                 PLLOCFG = 0x24;
                                                              // Set the multiplier and divider values (M=5, P=2)
43
44
                 PLLOFEED = OxAA;
                                                              // Sequence to update PLL registers
45
                 PLLOFEED = 0x55;
                 while (!(PLLOSTAT & PLOCK)); // Wait for the PLL to achieve lock
47
48
                  PLLOCON = 0x03;
                                                              // Connect the PLL (PLLE = 1 and PLLC = 1)
49
50
                  PLLOFEED = 0xAA;
                                                              // Sequence to update PLL registers after connecting
51
                  PLLOFEED = 0x55;
52
53
                  VPBDIV = 0x01;
                                                              // Set PCLK = CCLK (PCLK = 60 MHz if CCLK is 60 MHz)
54
          }
```

55