```c
#include <lpc214x.h>
#define LED_OFF (IO0SET = 1U << 31)
#define LED_ON (IO0CLR = 1U << 31)

#define PLOCK 0x00000400  // Bit mask for checking PLL lock status

#define COLSEL0 (IO1PIN & (1<<19))
#define COLSEL1 (IO1PIN & (1<<18))
#define COLSEL2 (IO1PIN & (1<<17))
#define COLSEL3 (IO1PIN & (1<<16))

void systeminit(void);
void delay_ms(unsigned int t);
void uartinit(void);

unsigned char rowsel=0, colsel=0;
unsigned char lookup_table[4][4] = {{'1','2','3','4'},{'5','6','7','8'},
                                    {'9','0','A','B'},{'C','D','E','F'}};


int main(){
    IO0DIR |= (1U<<31) |(0xF<<16);
    //IO1DIR |= 0xF<<16;
    systeminit();
    uartinit();
    LED_ON;
    delay_ms(500);
    LED_OFF;
    do{
      delay_ms(50);
      do{
        rowsel=0; IO0SET |= 0xF<<16; IO0CLR = 1<<16;
        if(!COLSEL0){colsel =0; break;} if(!COLSEL1){colsel =1; break;}
        if(!COLSEL2){colsel =2; break;} if(!COLSEL3){colsel =3; break;}

        rowsel=1; IO0SET |= 0xF<<16; IO0CLR = 1<<17;
        if(!COLSEL0){colsel =0; break;} if(!COLSEL1){colsel =1; break;}
        if(!COLSEL2){colsel =2; break;} if(!COLSEL3){colsel =3; break;}

        rowsel=2; IO0SET |= 0xF<<16; IO0CLR = 1<<18;
        if(!COLSEL0){colsel =0; break;} if(!COLSEL1){colsel =1; break;}
        if(!COLSEL2){colsel =2; break;} if(!COLSEL3){colsel =3; break;}

        rowsel=3; IO0SET |= 0xF<<16; IO0CLR = 1<<19;
        if(!COLSEL0){colsel =0; break;} if(!COLSEL1){colsel =1; break;}
        if(!COLSEL2){colsel =2; break;} if(!COLSEL3){colsel =3; break;}
      }while(1);

      delay_ms(50);
      while(!COLSEL0 || !COLSEL1 || !COLSEL2 || !COLSEL3);
      IO0SET |= 0xF<<16;    //Attension
      U0THR = lookup_table[rowsel][colsel]; //Attension
      delay_ms(50);
    }while(1);
}

void delay_ms(unsigned int t){
  unsigned int i,j;
  for(i=0; i<t; i++)
    for(j=0; j<10000; j++);
}


void systeminit(void) {
    PLL0CON = 0x01;      // Enable the PLL (PLLE = 1)
    PLL0CFG = 0x24;      // Set the multiplier and divider values (M=5, P=2)
    PLL0FEED = 0xAA;     // Sequence to update PLL registers
    PLL0FEED = 0x55;

    while (!(PLL0STAT & PLOCK));  // Wait for the PLL to achieve lock

    PLL0CON = 0x03;      // Connect the PLL (PLLE = 1 and PLLC = 1)
```

```
73        PLL0FEED = 0xAA;      // Sequence to update PLL registers after connecting
74        PLL0FEED = 0x55;
75
76        VPBDIV = 0x01;        // Set PCLK = CCLK (PCLK = 60 MHz if CCLK is 60 MHz)
77    }
78
79    void uartinit(void){
80      PINSEL0 |= 0x05;
81
82      U0LCR = 0x83;
83      U0DLM = 0;
84      U0DLL = 32; //BAUD RATE = 115200
85
86      U0LCR = 0x03;
87      U0FCR = 0x07;
88    }
89
```