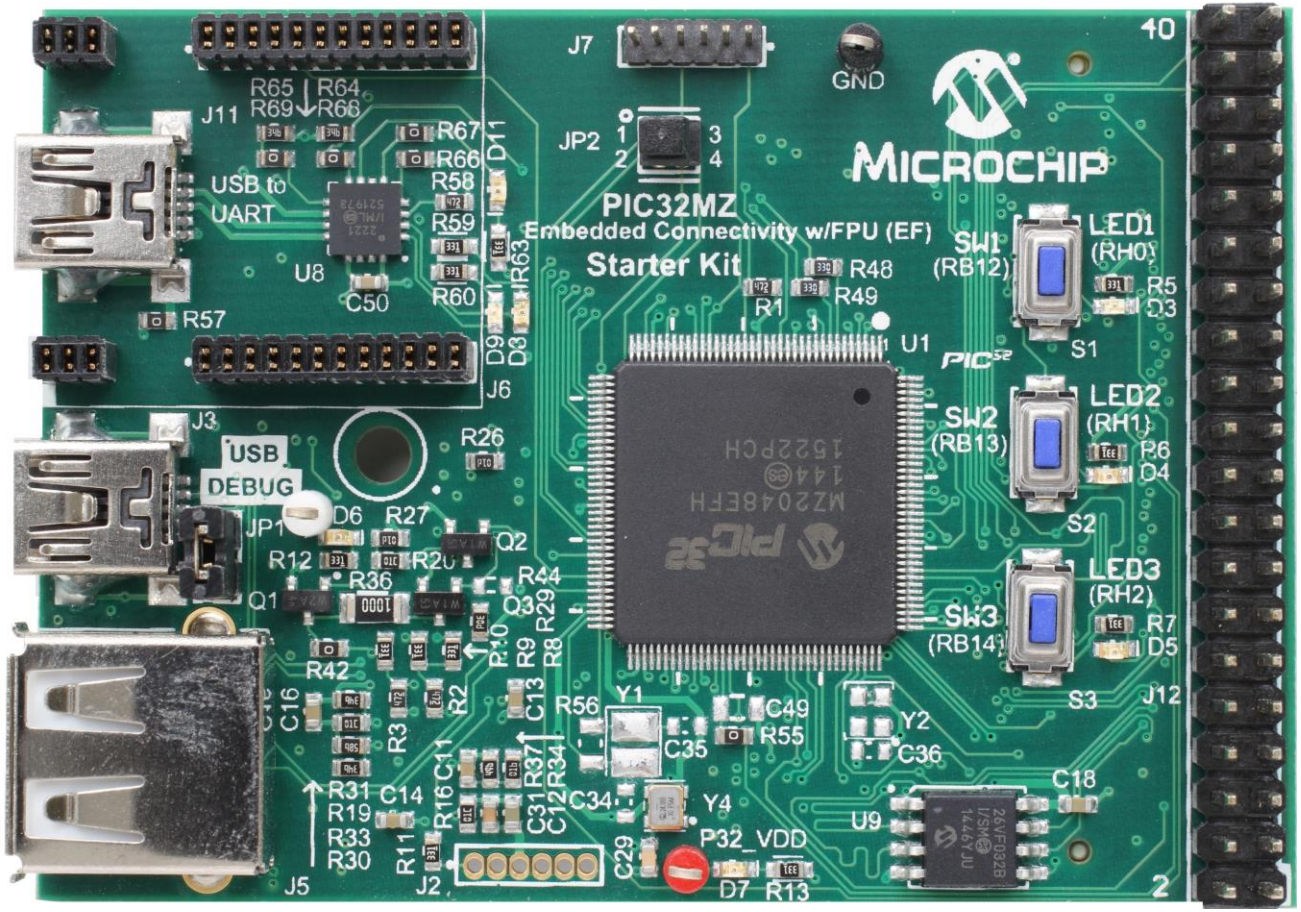
IoT Development Tools for PIC32



Curiosity PIC32MZ EF

January 31, 2018

FRED EADY

Presented by:

DesignNews

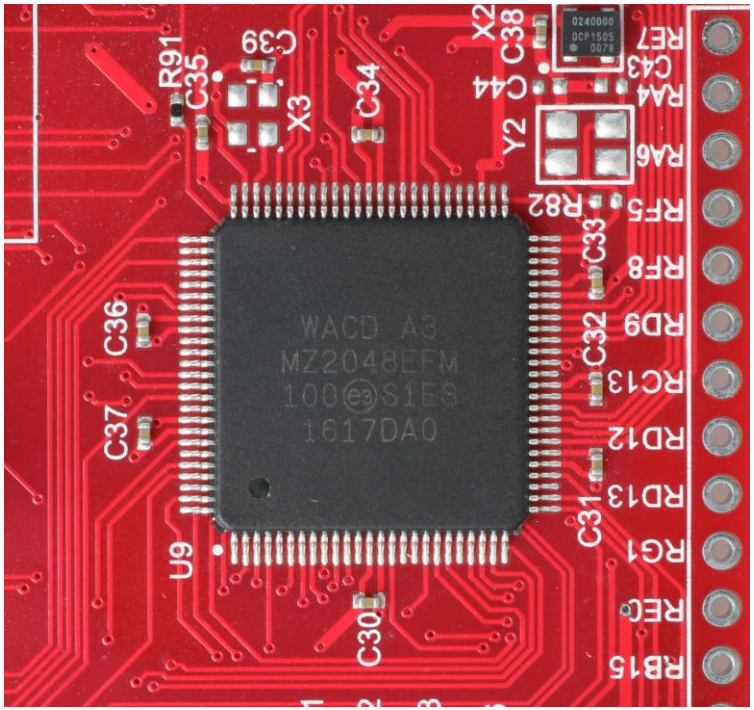
CEC CONTINUING EDUCATION CENTER

Digi-Key ELECTRONICS

IoT Development Tools for PIC32

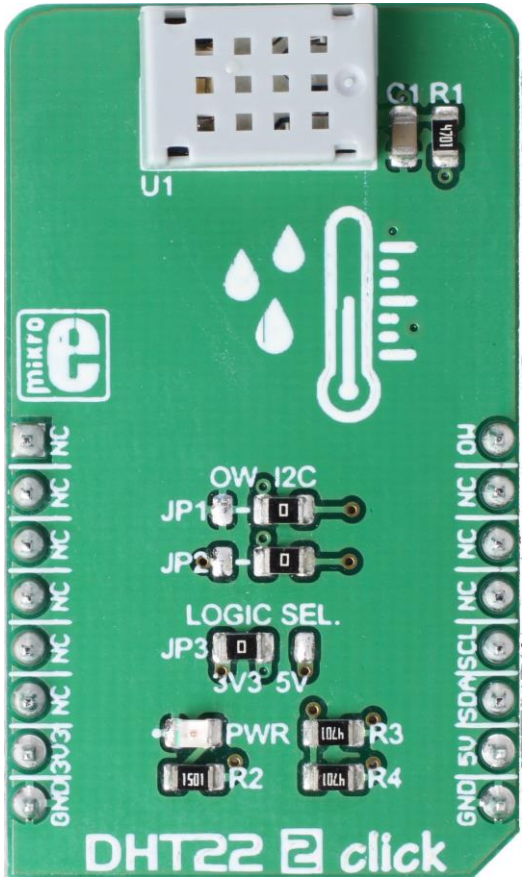
AGENDA

- Curiosity PIC32MZ EF Hardware
- Blinking in Harmony
- Singing Out of Tune
- microSD click Project
- Double click – DHT22-2 click Project
- Buenas Tardes

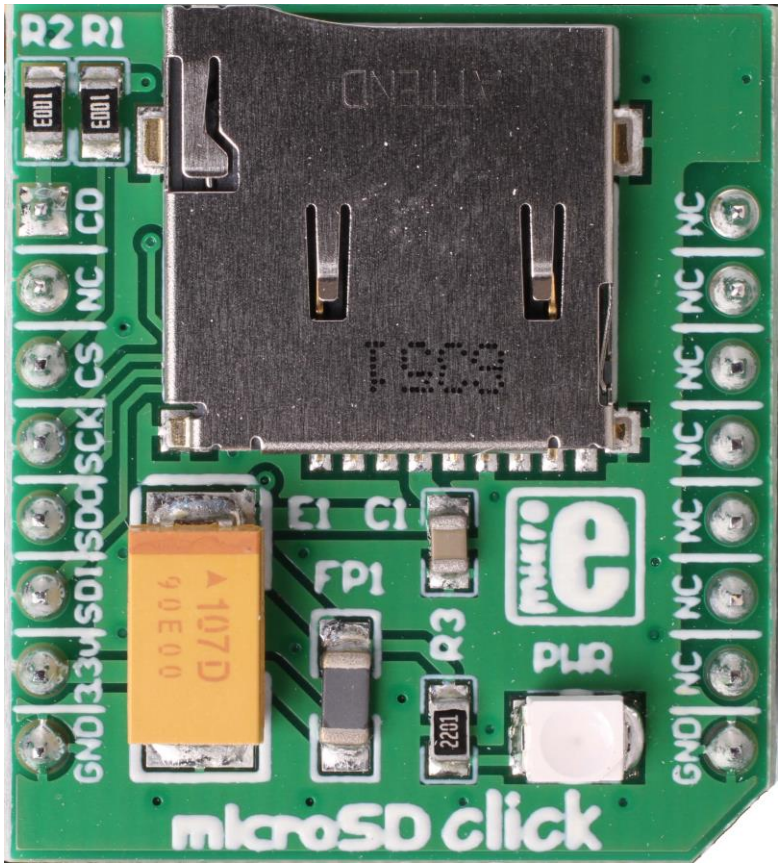


IoT Development Tools for PIC32

Curiosity PIC32MZ EF Hardware – click Part Numbers



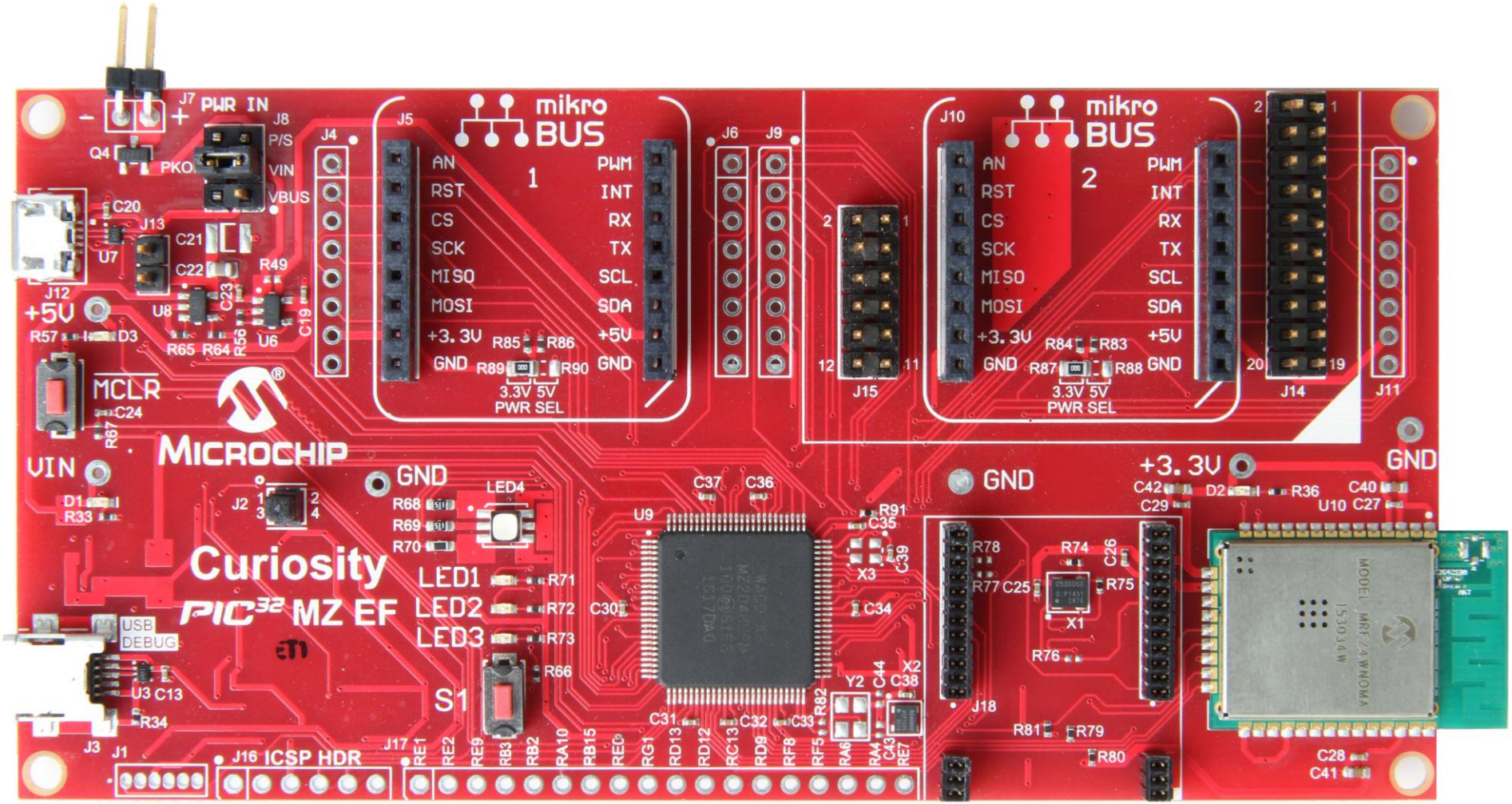
DHT22 2 CLICK
MIKROE-2818
1471-1886-ND



MICROSD CLICK STORAGE
MIKROE-924
1471-1303-ND

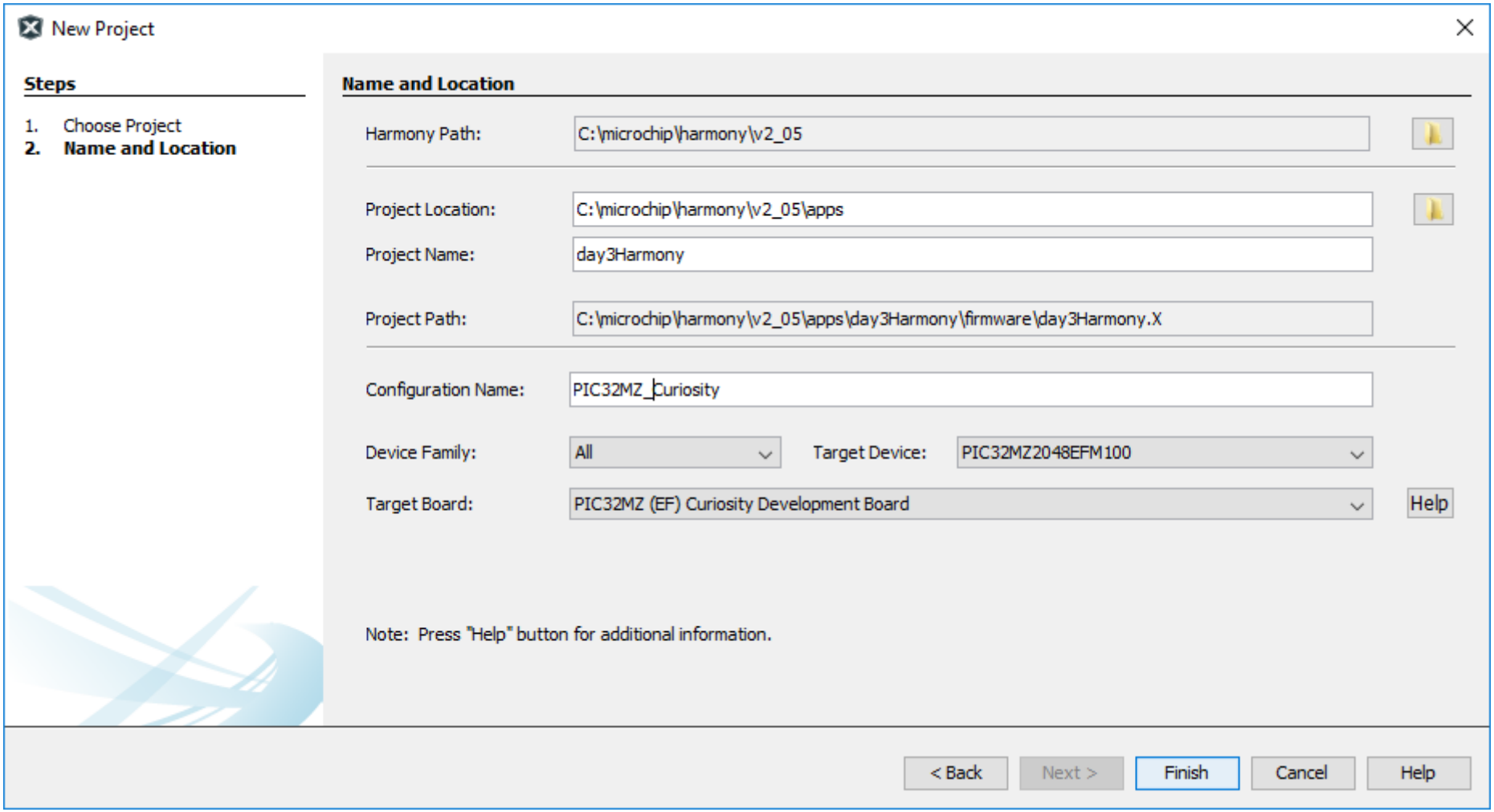
IoT Development Tools for PIC32

Curiosity PIC32MZ EF Hardware



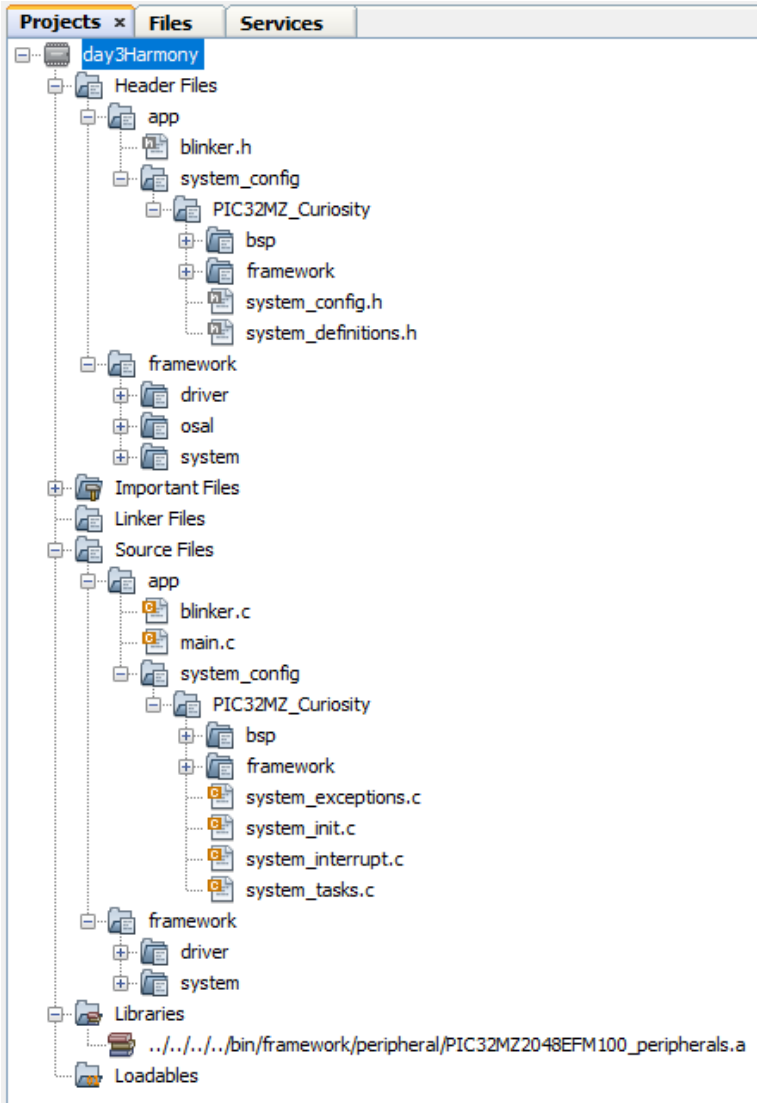
IoT Development Tools for PIC32

Blinking in Harmony



IoT Development Tools for PIC32

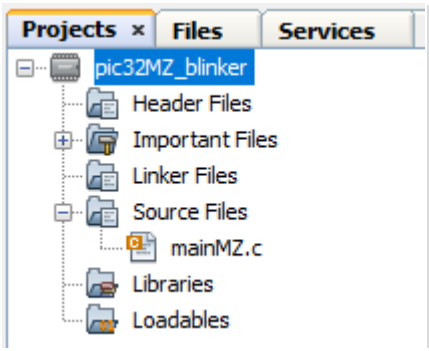
Blinking in Harmony




```
void BLINKER_Tasks ( void )
{

    /* Check the application's current state. */
    switch ( blinkerData.state )
    {
        /* Application's initial state. */
        case BLINKER_STATE_INIT:
        {
            bool appInitialized = true;


            if (appInitialized)
            {
                blinkerData.hDelayTimer = SYS_TMR_DelayMS(BLINKER_DELAY);
                if (blinkerData.hDelayTimer != SYS_TMR_HANDLE_INVALID)
                { // Valid handle returned
                    BSP_LEDOn(BLINKER_LED);
                    blinkerData.state = BLINKER_STATE_SERVICE_TASKS;
                }
                blinkerData.state = BLINKER_STATE_SERVICE_TASKS;
            }
            break;
        }

        case BLINKER_STATE_SERVICE_TASKS:
        {
             if (SYS_TMR_DelayStatusGet(blinkerData.hDelayTimer))
            { // Single shot timer has now timed out.
                BSP_LEDToggle(BLINKER_LED);
                blinkerData.state = BLINKER_RESTART_TIMER;
            }

            break;
        }

        /* TODO: implement your application state machine.*/
         case BLINKER_RESTART_TIMER:
         { // Create a new timer
            blinkerData.hDelayTimer = SYS_TMR_DelayMS(BLINKER_DELAY);
            if (blinkerData.hDelayTimer != SYS_TMR_HANDLE_INVALID)
            { // Valid handle returned
             blinkerData.state = BLINKER_STATE_SERVICE_TASKS;
            }
         break;
     }
```

IoT Development Tools for PIC32

Singing Out of Tune



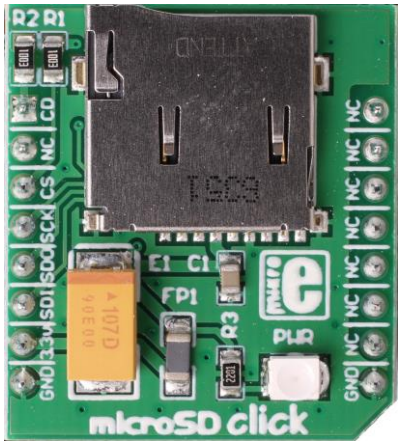

```
130     //Initialize TIMER2
131     T2CONCLR = 0xFFFF;          //Turn timer off - select PBCLK 1:1
132     T2CONSET = 0x0070;          //01110000 1:256 prescale
133     TMR2 = 0;                   //Clear timer register
134     PR2 = 0x0138;               //10mS @ 8MHz
135     //priority 4 sub priority 0 - bits <12:10> sub bits <9:8>
136     IPC2SET = 0x00001000; // 0000 0000 0000 0000 0001 0000 0000 0000
137     IFS0CLR = 0x00000200;
138     IEC0SET = 0x00000200;
139     T2CONSET = 0x8000;
```

```
86  //*****************************************************************************
87  //* SRAM VARIABLES
88  //*****************************************************************************
89  BYTE tmrTicks;
90  //*****************************************************************************
91  //* TIMER2 Interrupt Handler
92  //* it is set at priority level 4 with shadow register support
93  //* subpriority level 0
94  //*****************************************************************************
95  void __ISR_AT_VECTOR (_TIMER_2_VECTOR, IPL4SRS) ms10Handler(void)
96  {
97      if(++tmrTicks == 100)
98      {
99          LATBINV = 0x0001;
100         tmrTicks = 0;
101     }
102     IFS0CLR = 0x00000200;
103 }
```

Presented by:

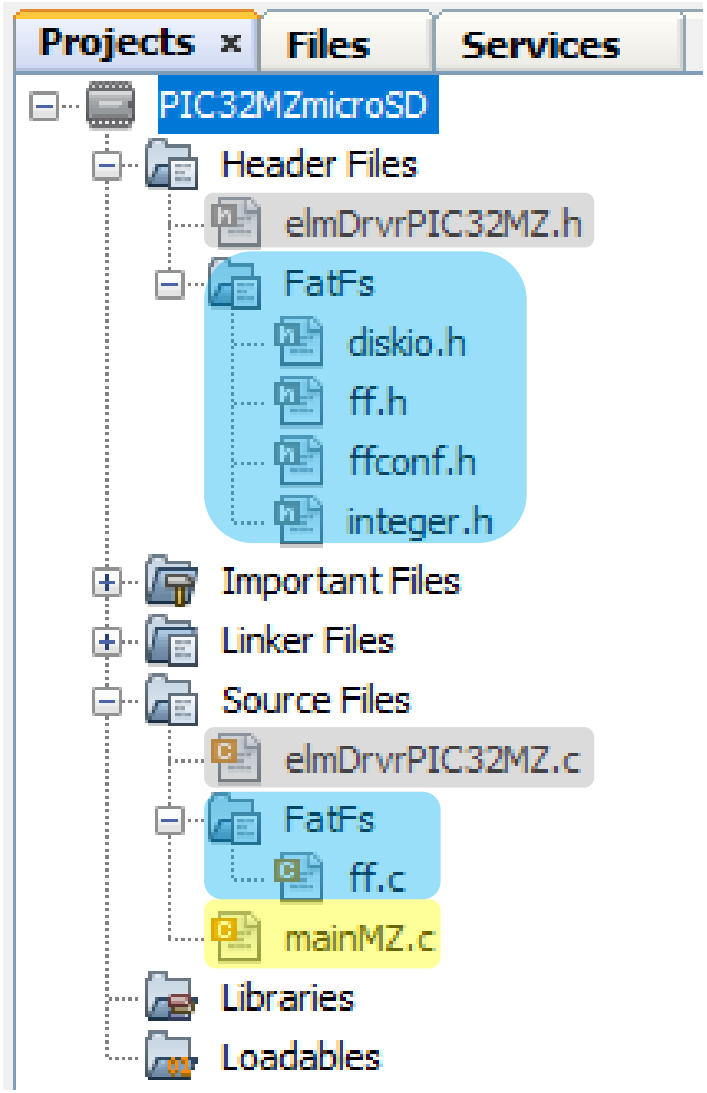
IoT Development Tools for PIC32

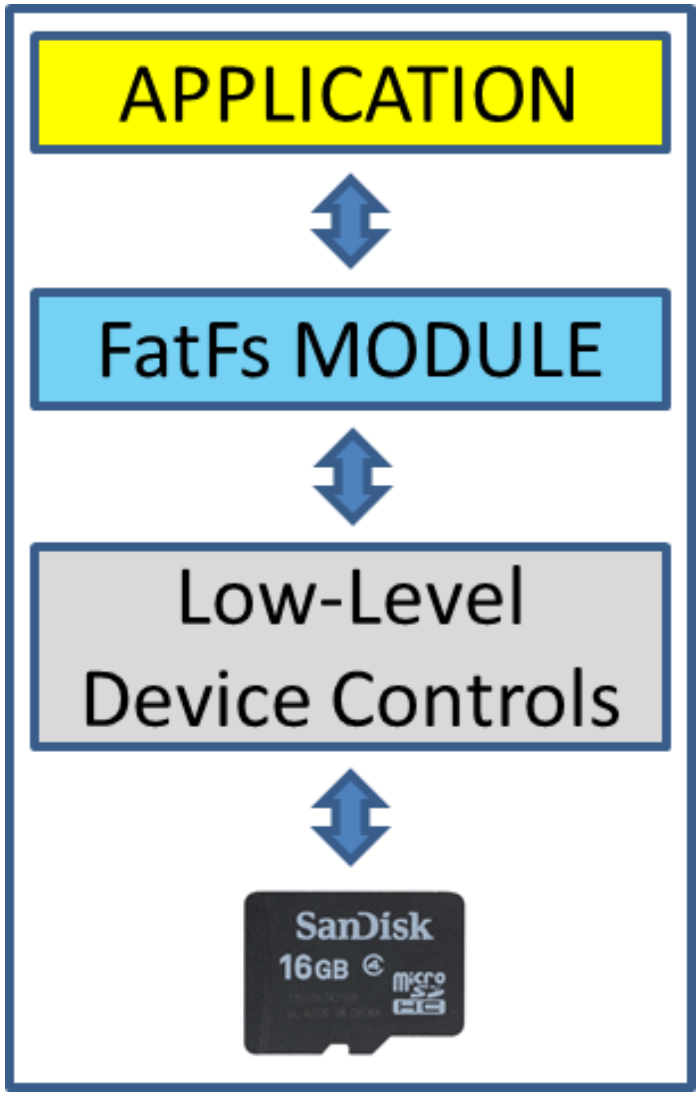
microSD click Project



IoT Development Tools for PIC32

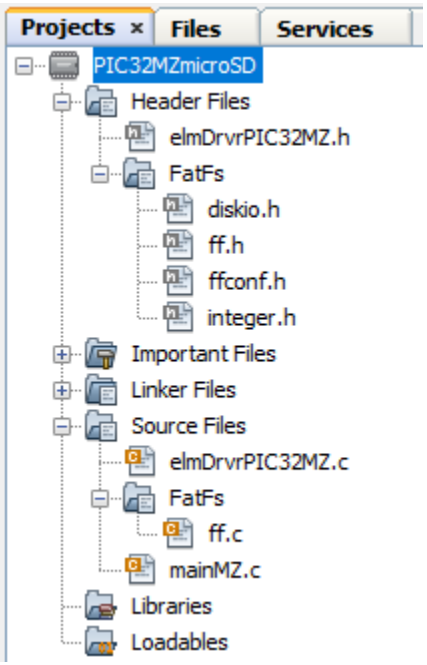
microSD click Project

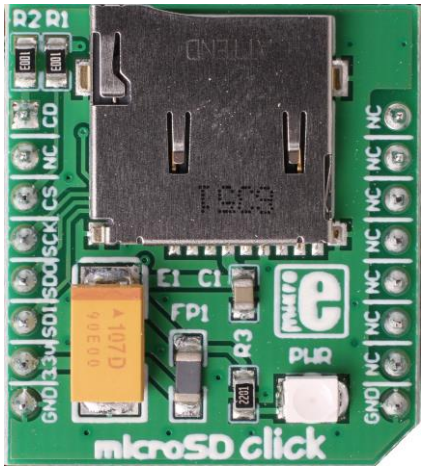




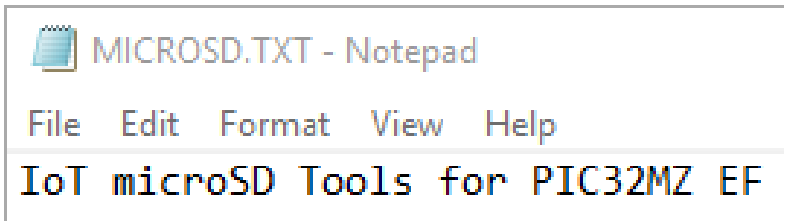
IoT Development Tools for PIC32

microSD click Project



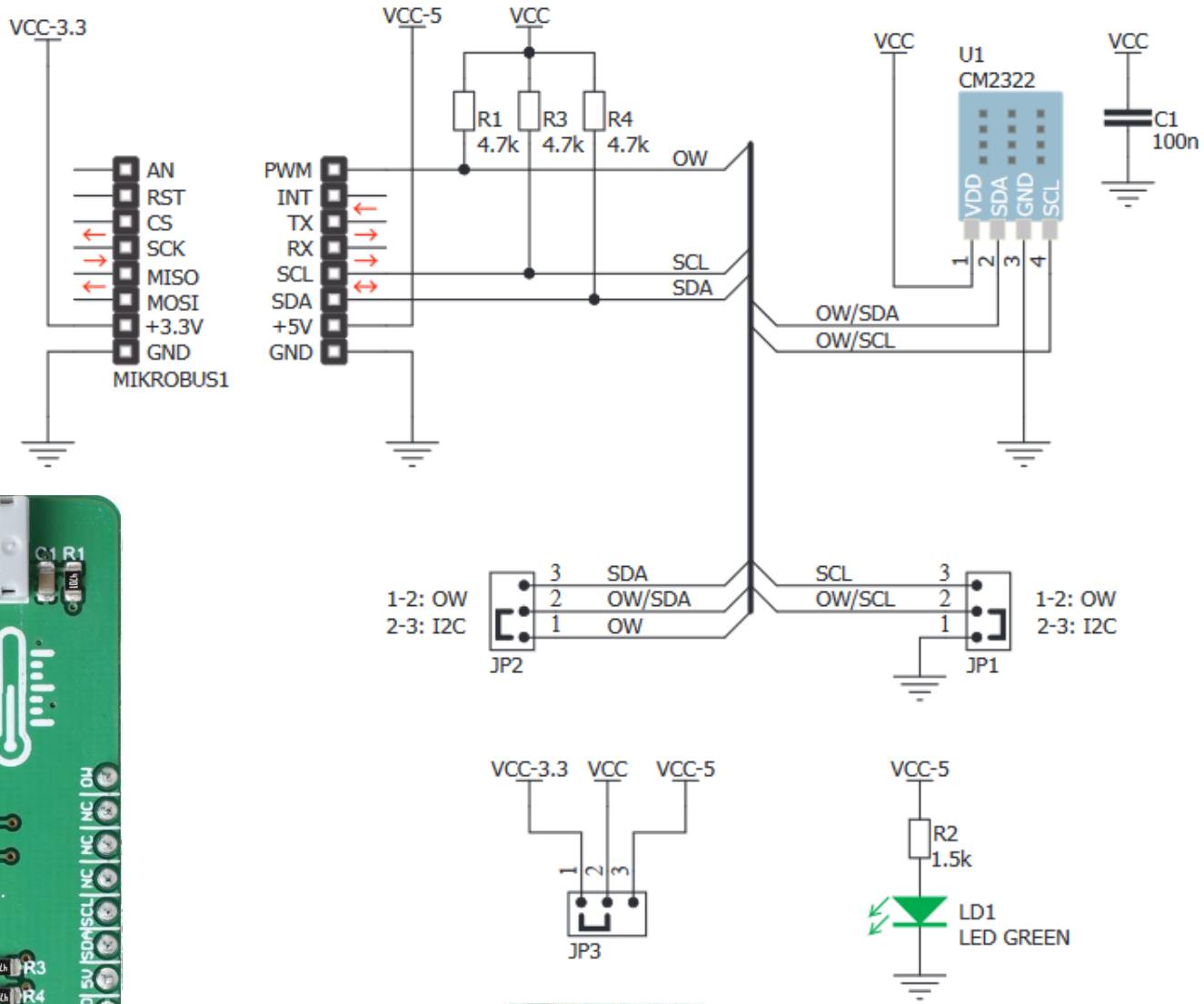


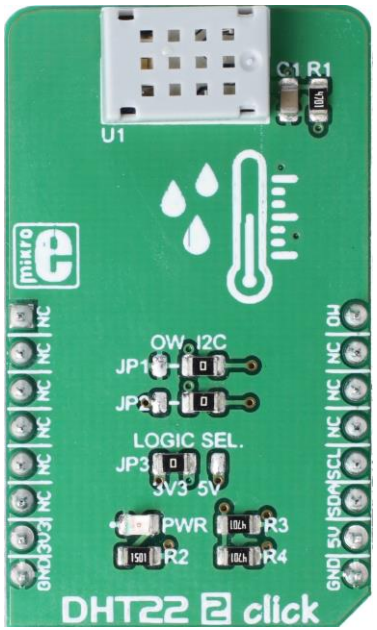

```
297 //*****************************************************************************
298 //* MAIN FUNCTION
299 //*****************************************************************************
300 int main (void)
301 {
302     FRESULT FOpenRes;
303     long accessmode;
304     char filelname[]= "microSD.txt";
305     accessmode= (FA_CREATE_ALWAYS | FA_OPEN_ALWAYS | FA_CREATE_NEW | FA_WRITE |FA_READ);
306     char Buff[] = "IoT microSD Tools for PIC32MZ EF";
307     unsigned int bitesWritten=0;
308
309     init();
310     disk_initialize(0);
311     f_mount(&Fatfs,"",0);
312     FOpenRes=f_open(&file1, filelname, (BYTE)accessmode);
313     if(FOpenRes==FR_OK)
314     {
315         LED2ON;
316     }
317     FOpenRes=f_write(&file1, &Buff, sizeof(Buff), &bitesWritten);
318     f_close(&file1);    //close file
319     f_mount(NULL,"",0); //unmount
320     if(bitesWritten)
321     {
322         LED3ON;
323     }
324     do{
325         ++scratch8;
326     }while(1);
327 }
```



IoT Development Tools for PIC32

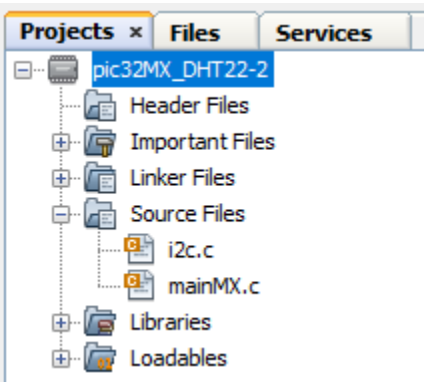
Double click – DHT22-2 click Project





IoT Development Tools for PIC32

Double click – DHT22-2 click Project

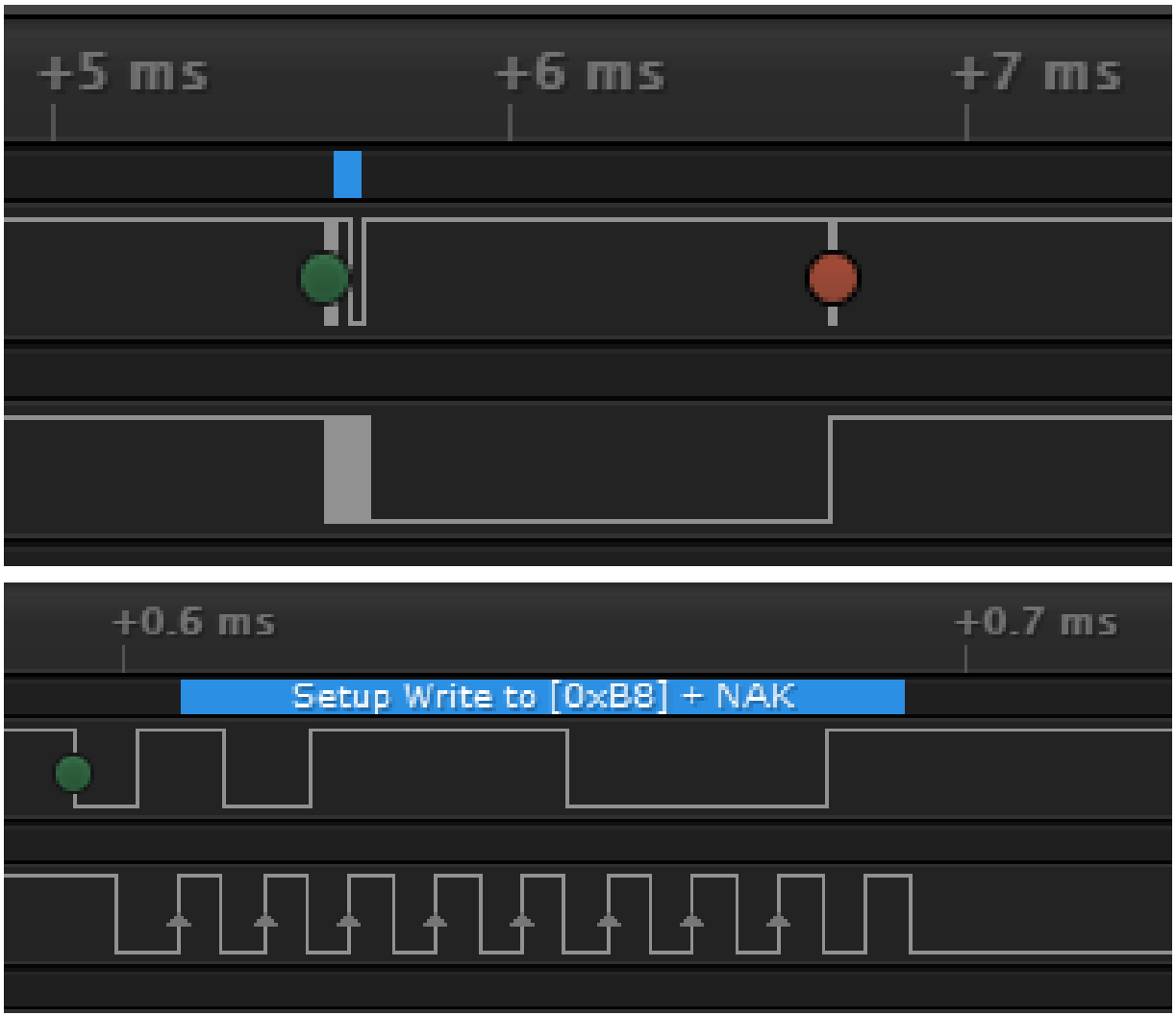



```
//*****************************************
//* MAIN FUNCTION
//*****************************************
void main(void)
{
    initMX();
    txBuf[0] = 0x03;
    txBuf[1] = 0x00;
    txBuf[2] = 0x04;
    clickOFF;
    i2cInit();
    ctDelayms(500);
    clickON;
    do{
    ctDelayms(1000);
    i2cStart();
    i2cSendByte(i2cAddrW);
    ctDelayms(1);
    i2cStop();
    ctDelayms(100);
    i2cWriteDHTRegs(i2cAddrW,txBuf,3);
    ctDelayms(100);
    i2cReadDHTRegs(i2cAddrR,rxBuf,8);
    ctDelayms(1000);
    }while(1);
}
```

IoT Development Tools for PIC32

Double click – DHT22-2 click Project

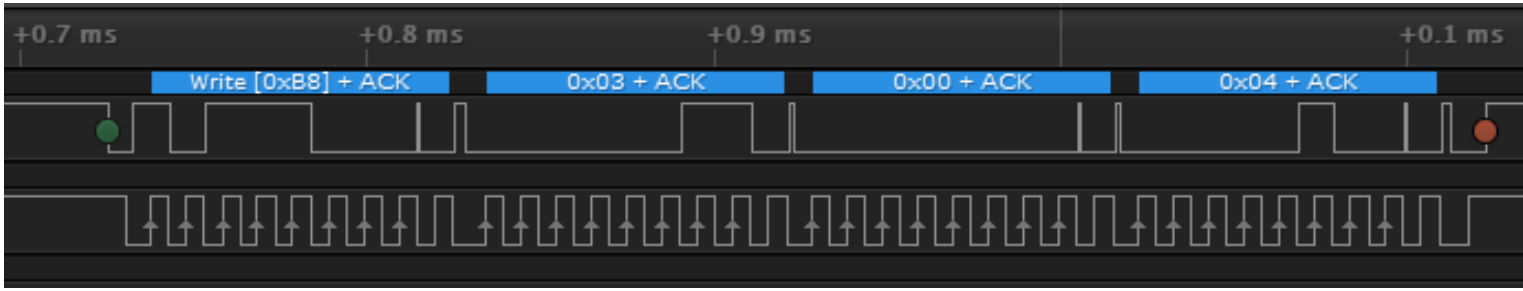
```
114 //****************************************
115 //* MAIN FUNCTION
116 //****************************************
117 void main(void)
118 {
119     initMX();
120     txBuf[0] = 0x03;
121     txBuf[1] = 0x00;
122     txBuf[2] = 0x04;
123     clickOFF;
124     i2cInit();
125     ctDelayms(500);
126     clickON;
127     do{
128     ctDelayms(1000);
129     i2cStart();
130     i2cSendByte(i2cAddrW);
131     ctDelayms(1);
132     i2cStop();
133     ctDelayms(100);
134     i2cWriteDHTRegs(i2cAddrW,txBuf,3);
135     ctDelayms(100);
136     i2cReadDHTRegs(i2cAddrR,rxBuf,8);
137     ctDelayms(1000);
138     }while(1);
139 }
```



IoT Development Tools for PIC32

Double click – DHT22-2 click Project

```
114 //*************************************
115 //* MAIN FUNCTION
116 //*************************************
117 void main(void)
118 {
119     initMX();
120     txBuf[0] = 0x03;
121     txBuf[1] = 0x00;
122     txBuf[2] = 0x04;
123     clickOFF;
124     i2cInit();
125     ctDelayms(500);
126     clickON;
127     do{
128     ctDelayms(1000);
129     i2cStart();
130     i2cSendByte(i2cAddrW);
131     ctDelayms(1);
132     i2cStop();
133     ctDelayms(100);
134     i2cWriteDHTRegs(i2cAddrW,txBuf,3);
135     ctDelayms(100);
136     i2cReadDHTRegs(i2cAddrR,rxBuf,8);
137     ctDelayms(1000);
138     }while(1);
139 }
```

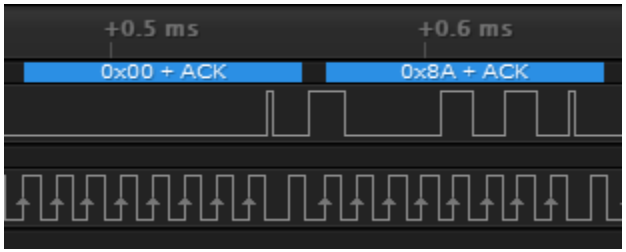


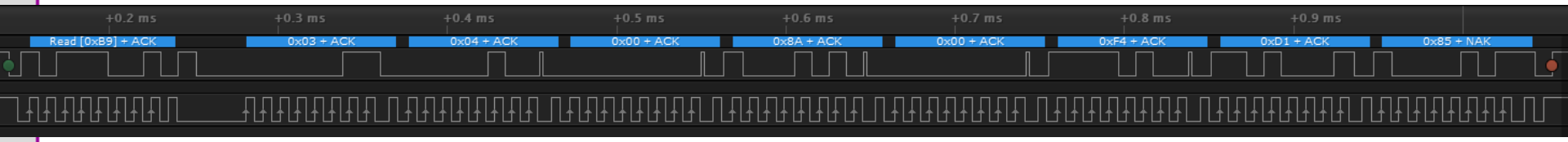
IoT Development Tools for PIC32

Double click – DHT22-2 click Project

```
114 //****************************************
115 //* MAIN FUNCTION
116 //****************************************
117 void main(void)
118 {
119     initMX();
120     txBuf[0] = 0x03;
121     txBuf[1] = 0x00;
122     txBuf[2] = 0x04;
123     clickOFF;
124     i2cInit();
```

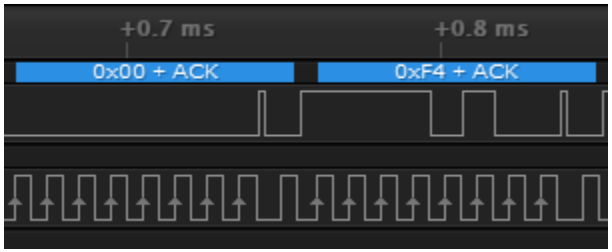
Humidity = 13.8%






```
130     i2cSendByte(i2cAddrW);
131     ctDelayms(1);
132     i2cStop();
133     ctDelayms(100);
134     i2cWriteDHTRegs(i2cAddrW,txBuf,3);
135     ctDelayms(100);
136     i2cReadDHTRegs(i2cAddrR,rxBuf,8);
137     ctDelayms(1000);
138     }while(1);
139 }
```

Temperature = 24.4° C



IoT Development Tools for PIC32
Buenas Tardes