```
// Experiment 8 - Setting Analog to Digital Converter to measure openair temperature.

#include <msp430g2553.h> //header file for msp430g2553 developer board.

#ifndef TIMER0_A1_VECTOR              // for other models, check if the vector is not defined.
#define TIMER0_A1_VECTOR TIMERA1_VECTOR  // defining the vector located in headerfile. Just for precaution.
#define TIMER0_A0_VECTOR TIMERA0_VECTOR
#endif

volatile long tempRaw;     // Volatile variable to be used in hardware registers. It will store the measurement result.
void FaultRoutine(void);   // Calibration fault trapper.

void main(void)
{
WDTCTL = WDTPW + WDTHOLD; // Stop watchdog timer

P1DIR = 0x41;            // P1.0 & 1.6 are set as outputs.
P1OUT = 0;               // Initially LEDs off.

if (CALBC1_1MHZ ==0xFF || CALDCO_1MHZ == 0xFF)
{
FaultRoutine(); // If cal data is erased, calibration fails, run FaultRoutine function defined below.
}

BCSCTL1 = CALBC1_1MHZ; // Set range for DCO Clock Source.
DCOCTL = CALDCO_1MHZ;  // Set DCO step + modulation.
                  // BCSCTL3 |= LFXT1S_2; VLO alternative.
BCSCTL2 |= SELM_0 + DIVM_3; // MCLK is set to DCO with a division = DCO/8.

IFG1 &= ~OFIFG; // Clear OSCFault flag.
```

```
while(1)
{

ADC10CTL1 = INCH_10 + ADC10DIV_0;                // Chosing the input channel as on board Temp sensor.The clock division Clock/1.
ADC10CTL0 = SREF_1 + ADC10SHT_3 + REFON + ADC10ON;  // Setting Vref and Vss. x64 Sample and hold time. Referance mode active, ADC is active.
_delay_cycles(5); // Wait for ADC Ref to settle.

ADC10CTL0 |= ENC + ADC10SC;          // Sampling & Conversion start.
P1OUT = 0x40; // green LED on.
_delay_cycles(100); // Wait for ADC Ref to settle.

ADC10CTL0 &= ~ENC;
ADC10CTL0 &= ~(REFON + ADC10ON);    // Stop Sampling & Conversion.

tempRaw = ADC10MEM;                // Measurement stored in tempRaw.
P1OUT = 0; // green LED off.

_delay_cycles(125000); //Measure temperature per 1 sec.
}

}

void FaultRoutine(void)
{
P1OUT = 0x01; // Red LED on.
while(1); // TRAP.
}
```