LAB TITLE AND CODE: EMBEDDED COMPUTING LAB (19CLE 283)
EXPERIMENT NUMBER 64
DATE: 10/04/2022, TUESDAY

* AIM S

Perform analog to digital conversion using msr43n by configuring its on-this peripherals.

ALGORITHM & CADO PERIPHERAL PROGRAMMING AND DIGITAL OUTPUT DISPLAY ON LED)

O configure functionality of P2-0, P2-1 and P2-2 as simple GP10 pins.
O configure direction of P2-0, P2-1 and P2-2 as output for this color LEDS

@ Power on should be disabled during configuration.

Sample-and-hald pulse-made select, sysclk. 32 sample clocks and software trigger.

3 set resolution as 12-bit (4 clock cycle conversion time)

1 At input, single-ended and vac - Avec.

O configure functionality of P4-1 for ADC pin At and it's direction as input.

To convert for memory register 5, repeat single channel (sequence sample as active) and enable APC after configuration

(i) start the conversion.

(ii) Wait for ADC conversion to complete

(iv) Puplay bits 10, 9, 8 on the color LEDS.

* CODE: CAPC DERIGHERAL PROGRAMMING AND
DIGITAL OUTPUT DISPLAY ON LED)

include "mp. h"

Mmain Function: int main (vois) int result;

P2 -> SELO k= ~7: 11 Configure functionality of P2.0, P2.1 and P2-2 as simple GP10 pins

P2 -> SEH &= M)

P2 - DIR 1= 7; Il Configure direction of P2.0, P2-1 and P2-2 as output for the color LED.s

ADC 14 -> CTLO = 0x00000010; II power ON should be disabled during configuration

ADC 14 -> CTLO - 0x04080300; // Sample-and-hold pulse-made select. sysclk, 32 sample clocks, software thingger ADC 14 -> CTLI = 0x00000020; // Set resolution as 12-bit (14 clock

yde conversion times

ADC 14 -> MCTL (5) = 6; 11 A6 input, single-ended, vief = A Vcc.

11 Configure functionality of P4.7 for ADC pin A6 and direction as input :-

P4 -> SEL1 /= 0x80;

PY -> SELO 1= Ox80;

11 Convert for memory register 5:5
ADC 14 -> CTLI 1= 0x 00050000: 11 Repeat single channel, sequence sample is active

ADC 14 -> CTLO 1 = 2; Stenable ADC after configuration

I Infinite toop (An embedded phogram does not stop): while (1)

ADC14 -> CTLO 1=1: Il start the conversion now while (! ADC14 - 1FGRO); 11 Wait for ADC conversion to complete result = ADOI4 -> MEM[5]; IT Read ADC data register P2→OUT = result >>8, Noisplay buts 10:8 on this color 4500

DIAGRAM S CIRCUIT Vec (+ SV) MSP432 microcontroller 2 P4.7 in a roman Proba Ground CAND) STEED COME 1 120 121 and 122 an way e 40 carry reagens and reason in surject for the marite throlled during capperation. Who hall you comedo secreto, signis, 32 sappe of s Figure. ADC Pumpheral Programming are tigital output Display on NED 15 Miss of the Tory and in facilities this of the season paint in it I come of the memory willing of finding the formation and it is before I note Europee the explicit established

is it was beauteries for perfection that for early ...

* ALGORITHM: (TEMPERATURE CONTROLLER IMPLEMENTATION WING ADC)

O configure functionality of P2.1 as simple GP10 pins.

3 Configure direction of P2.1 as autput for GREEN-LED

3 Ponor ON should be disabled kuring configuration.

D sample - and - hold pulse-made select, sysclk, 32 sample clocks and software thigger.

@ Set assolution as 8-bit (9 clock cycle conversion time)

@ Set 86 input, single-ended, vneg = Avec

Tonfiguere functionality of P4.7 as ADC pin and it's direction as input.

3 To convert for memory register 5, hepeat single channel (sequence sample is achie) and enable acc after conversion.

(i) start the conversion now

(11) Wait for ACC conversion to complete

(iii) Read ADC data negister (iv) Display bit for GREEN-LED

(v) If temperature of the noom goes above 28° our ON the GROUN-LED)

elx OFF.

CODE: (TEMPERATURE CONTROLLER IMPLEMENTATION USING ADC)

include "msp-R"

I main Function :

int result)

 $P2 \rightarrow SELO V = ~^2$; // Configure functionality of P2+1 as simple GP10 pins $P2 \rightarrow SEL+1 = ~^2$? $P2 \rightarrow DIR 1 > ~^2$; // Configure direction of P2+1 as output for GREEN-LED

ADCIA -> CTLO = 0x00000010; Il Power ON should be disabled during configuration

ADC 14-0 (TO & ax04080300; 11 sample- and-hald pulse-made select,

sysilk, 37 sample clocks, software suigger

ADC 14 -7 CTT i = ax000000000; 11 set aesalution as 8-bit Ca clock cycle conversion time)

ADC 14 -> MCTI [5] = 6: NA6 input, single-ended, Vneg > AVCC

Il Configure functionality of P4.7 as ADI pin and it's direction as input:

M - SEH I= Ox80;

P4 -> SELO 1= Ox50)

I Convert for memory negister 5: -ADC14 -> CTL1 1= 0x00050000; // Repeat single channel, sequence sample is active ADC 14 - 0710 122; Il Enable ADC after conversion

N Infinite Loop (An embedded program does not stop) 8

while (1)

ADC14 -> CTO 101 > 11 start the conversion new while ([ADC14 -> 1FGRO); 11 wait for ADC conversion to complete result - ADCH -> MEM [5]; // Read ADC data negister P2 > OUT = grenelt >> 8) 11 piplay bit for GREEN- LED

Temperature Range - 0 degrees (0x00) to 60 degrees (0x11) If temperature goes above 28 degrees, turn on the HD; else OFF. 60 degrees = 287 bits (2°8 bit -resolution) Therefore, 28 degrees = ??? = (256 + 28) 160 = 119.4667 - 120 bits In hexadecimal nepresentation, (120) 10 = (76) 16

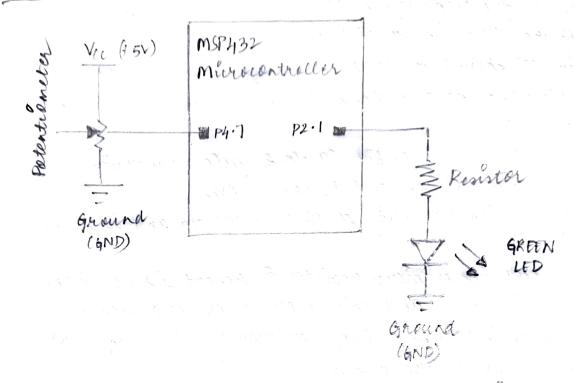


Figure. Temperature Controller Implementation using ADC

ENGREY SENSE WAS BUTTER AND ASSESSED SENSON STREET

```
if (Result >> 0×18)

£

P2→ OUT 1>2; IN TWEN ON P2-N GREEN-LED

else
£

P2→ OUT L> ~2; IN TWEN OFF P2-N GREEN-LED

3
```

* RESULT:

Implemented analog to digital conversion using MSP43 x by configuring its on-chip peripherals and all simulation results were verified successfully