

**Warsaw University of Technology**  
**Faculty of Mechatronics**

Microcontrollers

**Lab 04**

Report

Analog to Digital Converters

Made by:

Harshit Verma (302601)

## 1) Theory

In this course of this exercise, we must create an application to analyze different inputs and outputs on ADuC834 by pressing specific buttons, using Keil uVision5 software.

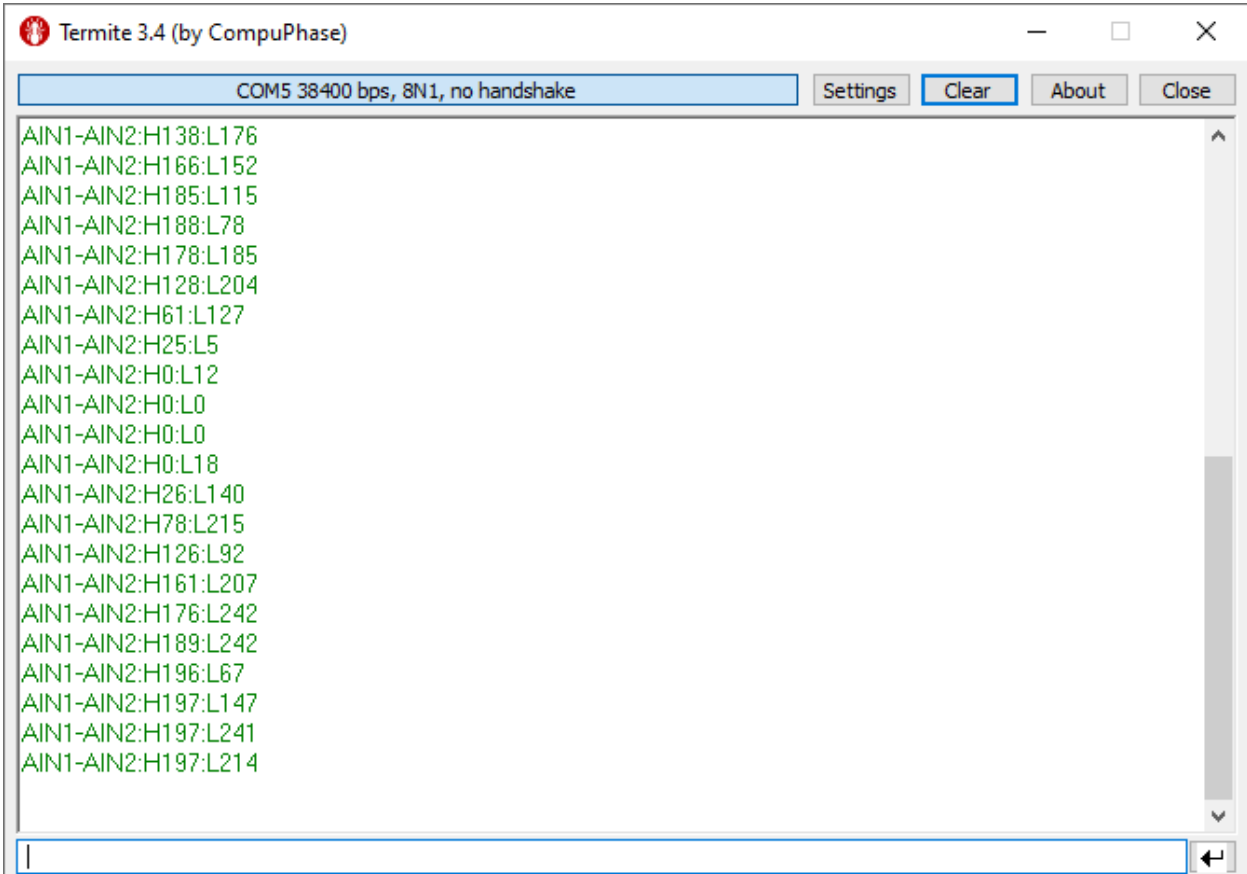
## 2) Result

- Screenshots showing the results obtained on the computer's terminal with different signal sources, with discussion:

➤ to what extent the values changed:

We observe that

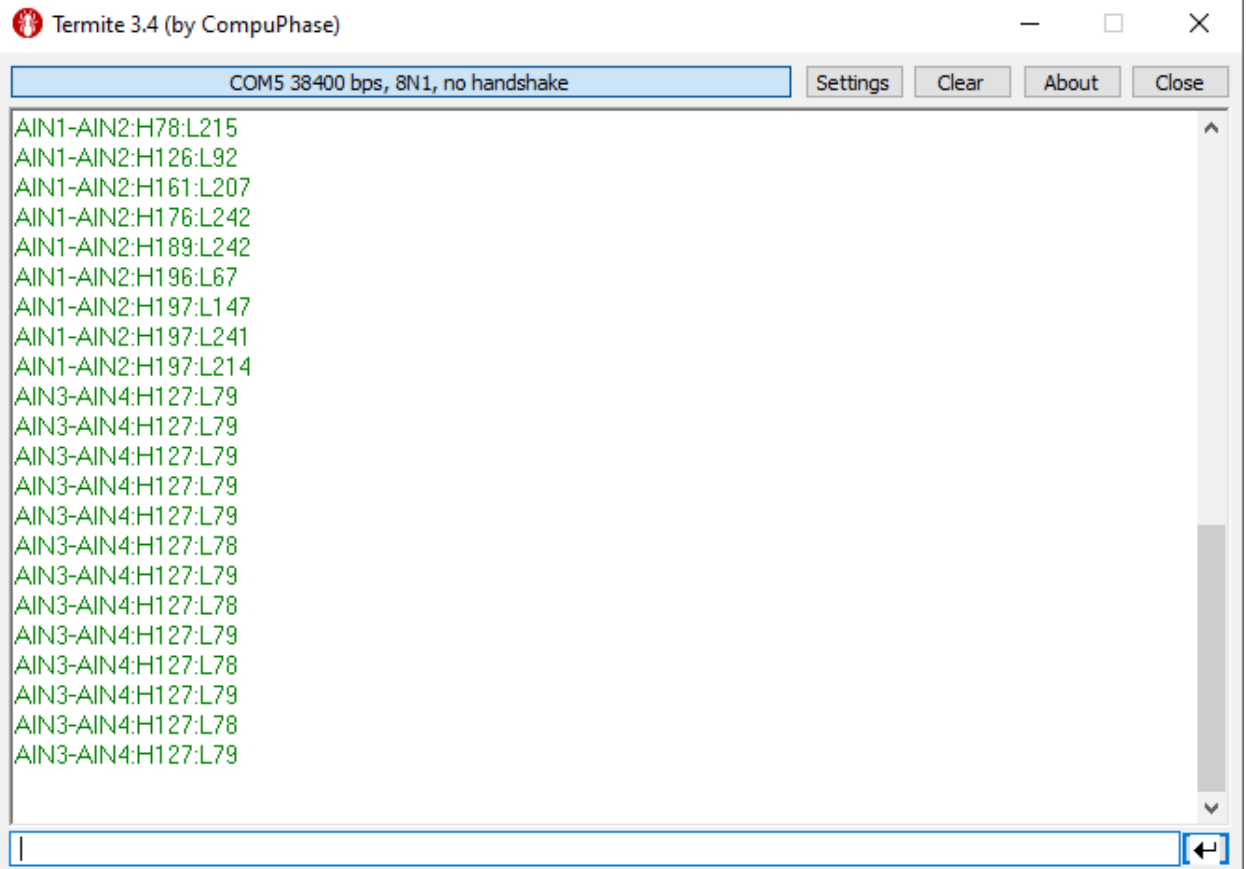
- (1) While the knob is turned clockwise, the values increase and vice-versa; the values change from H:0 L:0 to H:189 L:242; the values for High and Low parts do not change linearly with respect to each other, but rather there is a lag in both of the values. Overall the values increase linearly.
- (2) While the button is pressed, the value changes from H:0 L:0 to H:127 L:79. The increase is more like 1 to 0 rather than linearly increasing.
- (3) While the button is pressed and the thermistor is being manipulated, the value changes from H:228 L:47 to H:228 L:52. The change is the Low count is linear.

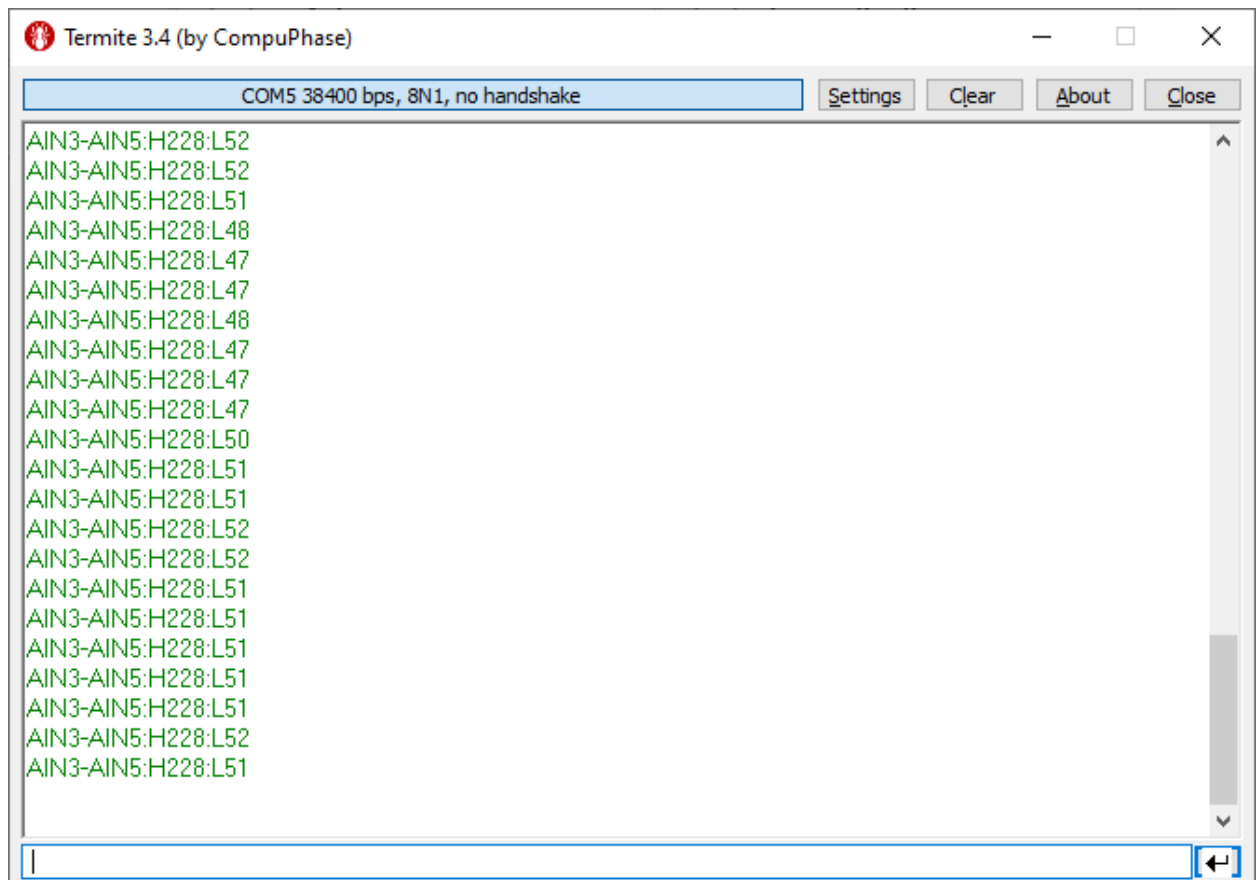


The screenshot shows a terminal window titled "Termit 3.4 (by CompuPhase)". The window has a status bar at the top indicating "COM5 38400 bps, 8N1, no handshake" and buttons for "Settings", "Clear", "About", and "Close". The main area of the terminal displays a list of ADC readings in green text, each consisting of a high (H) and low (L) count separated by a colon. The readings are as follows:

```
AIN1-AIN2:H138:L176  
AIN1-AIN2:H166:L152  
AIN1-AIN2:H185:L115  
AIN1-AIN2:H188:L78  
AIN1-AIN2:H178:L185  
AIN1-AIN2:H128:L204  
AIN1-AIN2:H61:L127  
AIN1-AIN2:H25:L5  
AIN1-AIN2:H0:L12  
AIN1-AIN2:H0:L0  
AIN1-AIN2:H0:L0  
AIN1-AIN2:H0:L18  
AIN1-AIN2:H26:L140  
AIN1-AIN2:H78:L215  
AIN1-AIN2:H126:L92  
AIN1-AIN2:H161:L207  
AIN1-AIN2:H176:L242  
AIN1-AIN2:H189:L242  
AIN1-AIN2:H196:L67  
AIN1-AIN2:H197:L147  
AIN1-AIN2:H197:L241  
AIN1-AIN2:H197:L214
```

At the bottom of the terminal window, there is a text input field with a cursor and a "Send" button (represented by a right-pointing arrow).





### 3) Program code

```
/* a header with definitions of symbols for ADuC834 */  
#include <aduc834.h>  
#include <stdio.h>  
/* optional: declaration of unsigned types of one and two bytes */  
typedef unsigned char uint8_t;  
typedef unsigned int uint16_t;
```

```
char buf[40];
```

```
void uart_transmit(char c)  
{
```

```

    SBUF = c;
    while(TI == 0);
    TI = 0;
}

```

```

void uart_string(char *str)
{
    while (*str != 0)
    {
        uart_transmit(*str);
        str++;
    }
}

```

```

int main(void)
{
    SM0 = 0;
    SM1 = 1;
    REN = 1;
    T3CON = 0x80;
    T3FD = 0x12;

```

```

    ADCMODE = (1<<5)|(1<<4)|(1<<1)|(1<<0);

```

```

    while (1)
    {

```

```

if (!(P1&(1<<1)))
{
    ADC0CON = (1<<2)|(1<<1)|(1<<0)|(1<<3)|(1<<6);

    while (RDY0 != 1){}

    sprintf(buf, "AIN1-AIN2:H%bu:L%bu\n",ADC0H,ADC0L);
    uart_string(buf);
    RDY0 = 0;
}

if (!(P2&(1<<0)))
{
    ADC1CON = (1<<6)|(1<<4)|(1<<5);

    while (RDY1 != 1){}

    sprintf(buf, "AIN3-AIN4:H%bu:L%bu\n",ADC1H,ADC1L);
    uart_string(buf);
    RDY1 = 0;

}

if (!(P3&(1<<5)))
{
    ADC1CON = (1<<6)|(0<<4)|(0<<5);

    while (RDY1 != 1){}

    sprintf(buf, "AIN3-AIN5:H%bu:L%bu\n",ADC1H,ADC1L);
    uart_string(buf);
    RDY1 = 0;
}

}

```

```
    return 0;  
}
```