

## Apéndice E

### Programa para calcular la tabla con valores de inclinación

```
#include <cstdlib>
#include <math.h>
#include <iostream>
#include <stdlib.h>
#include <stdio.h>
#include <string.h>
#include <conio.h>
#include <fstream>

double Voff=2.5, Vout=0, f, Vmax=3.7, sens=1.2; //Voltaje de offset, voltaje del sensor, sensibilidad

int angulo, numero; //Angulo de inclinación

double valores=255; //Las posibles combinaciones dependiendo de cuantos bits para ADC

int i=0,b=8, residuo; //contador y numero de bits del convertidor ADC

double step=0; //Tamaño de cada escalon

int s=0,c,a;

long int n,bin=0;

void binario(int);

main(void)
{
    FILE *TablaADC = fopen("tabla.txt", "w"); //Se genera archivo tipo .txt

    step=(5/valores); //Calculo de la longitud de cada escalon

    printf("\nN= %d \n", b);

    printf("Valores= %f \n", valores);

    printf("Step= %f \n", step);

    fprintf(TablaADC, "          ANGULO          VOUT          BINARIO \n\n"); //Encabezado de la tabla

    for (i=0;i<=valores;i++) //Corre hasta que i= # de escalones posibles

        {
```

## Apéndice E

---

```
f=((Vout/4.16)+2.5-Voff)/sens);

angulo=int((asin(f)*180/M_PI)); //Angulo calculado en grados

numero=i;

printf("    retlw    0x%d    ; %d    %f    ", angulo, i, Vout);

fprintf(TablaADC, "    retlw    0x%d    ; %d    %f    ", angulo, i, Vout); //Se guarda la tabla

binario(numero); //Convirtiendo a binario

printf("\n");

fprintf(TablaADC, "\n");

Vout=(Vout+step); //Siguiete escalon de voltaje

}

fclose(TablaADC); //Se cierra archivo guardado

getch();

}

//Funcion para convertir a binario

void binario(int numero)

{
    int residuo;

    if(numero <= 1)

    {
        printf("%d ", numero);

        return;

    }

    residuo = (numero%2);

    binario(numero >> 1);

    printf("%d ", residuo);

}
```