Trabajo Practico Nº4

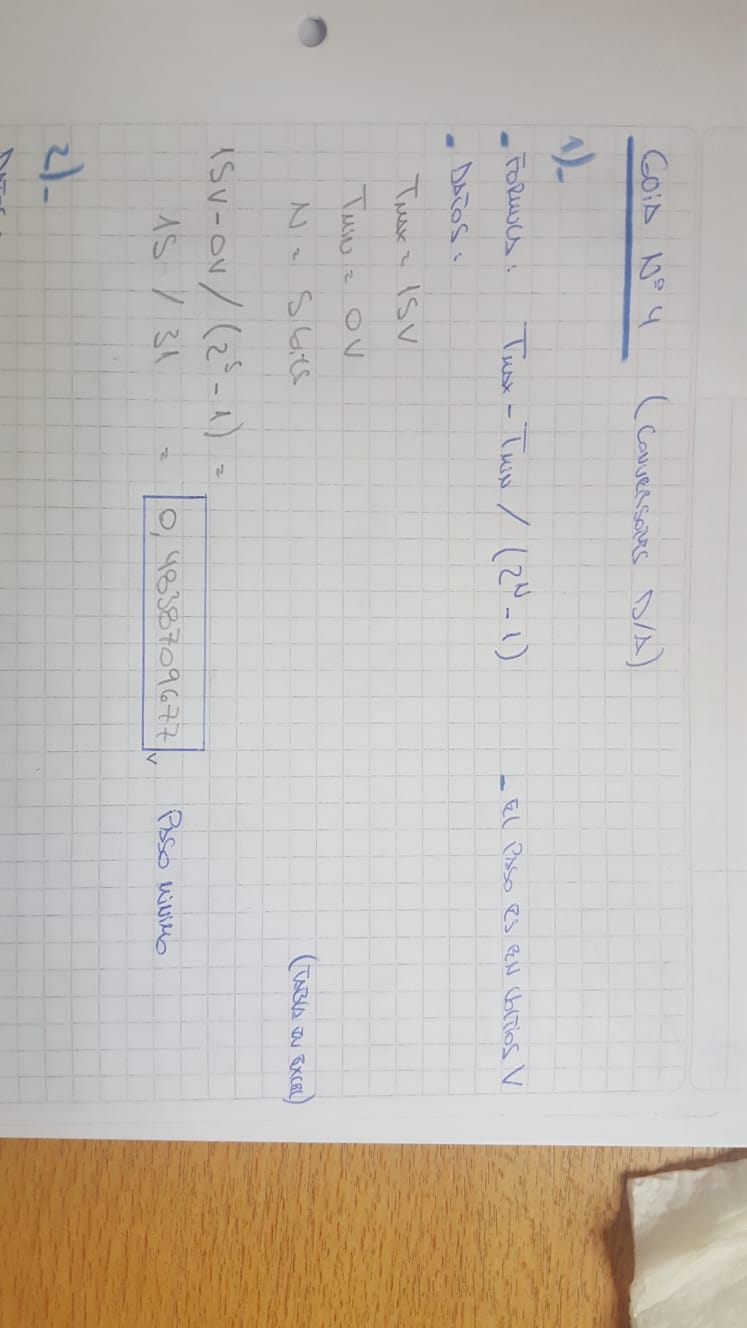
Nombre: Santiago vietto

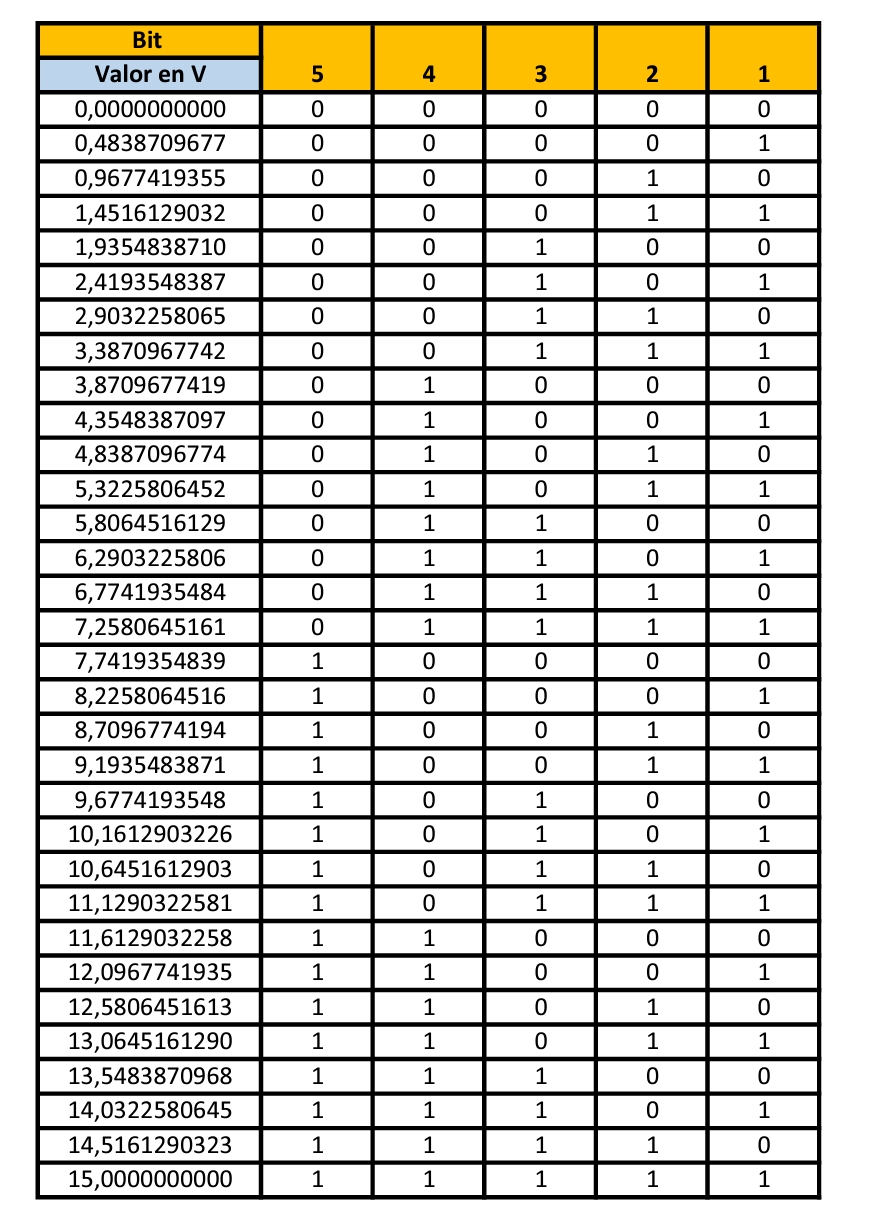
Materia: Arquitectura de computadoras 2

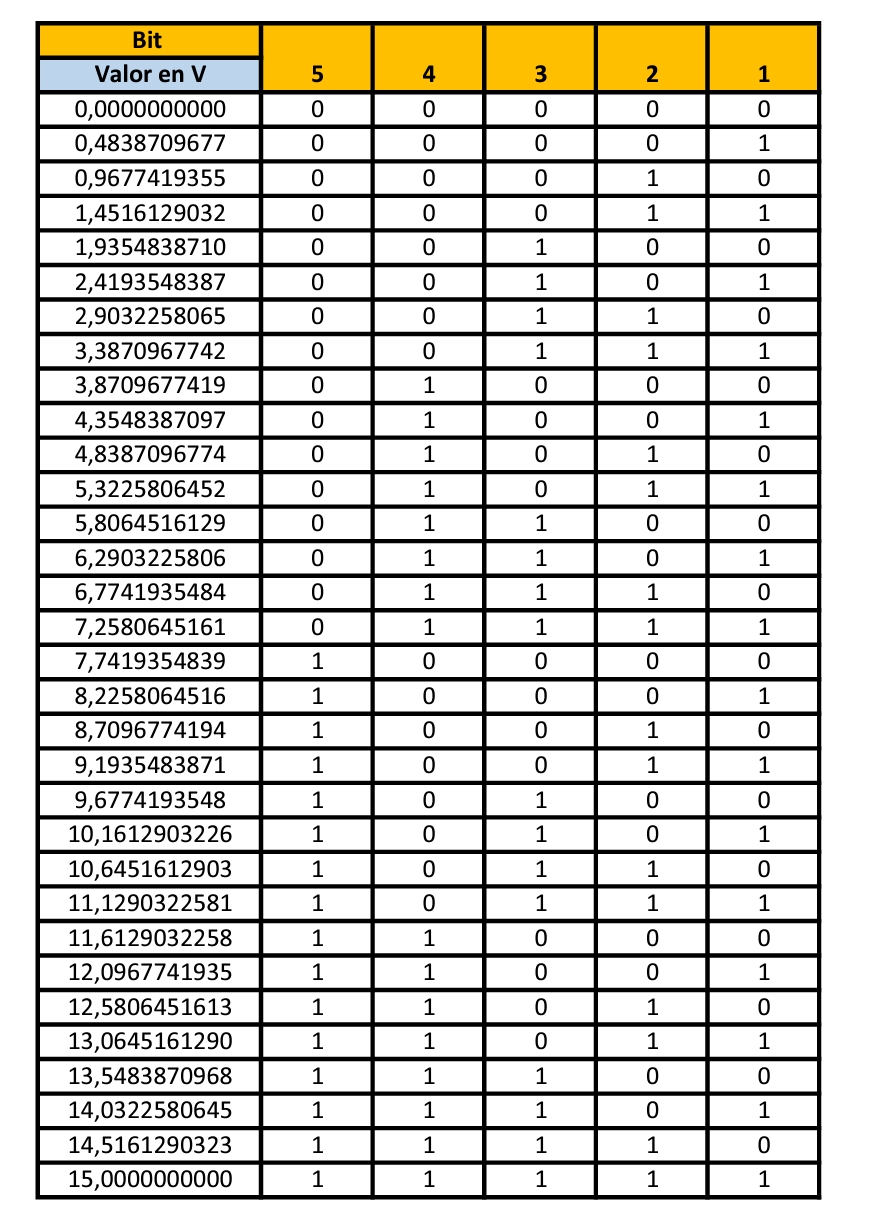
Docente: Ignacio Fazio

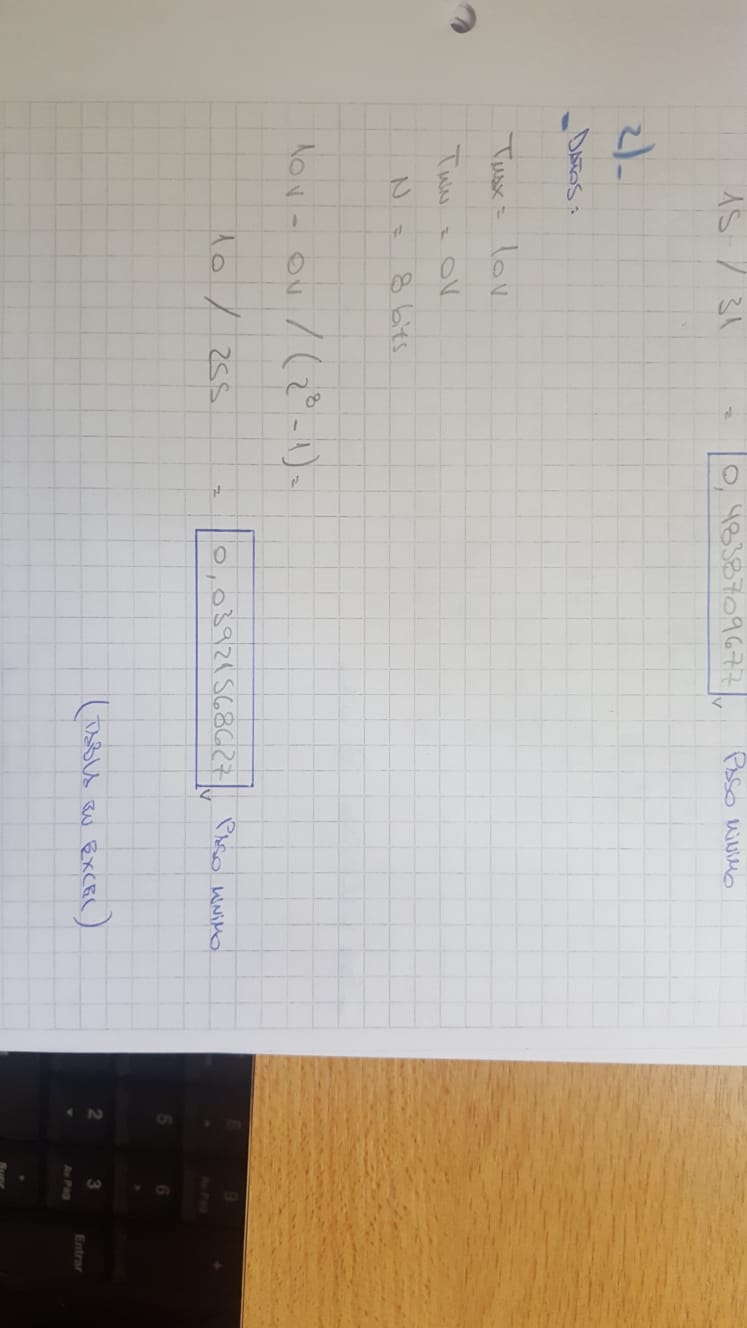
Clave: 1802890

Año: 2020

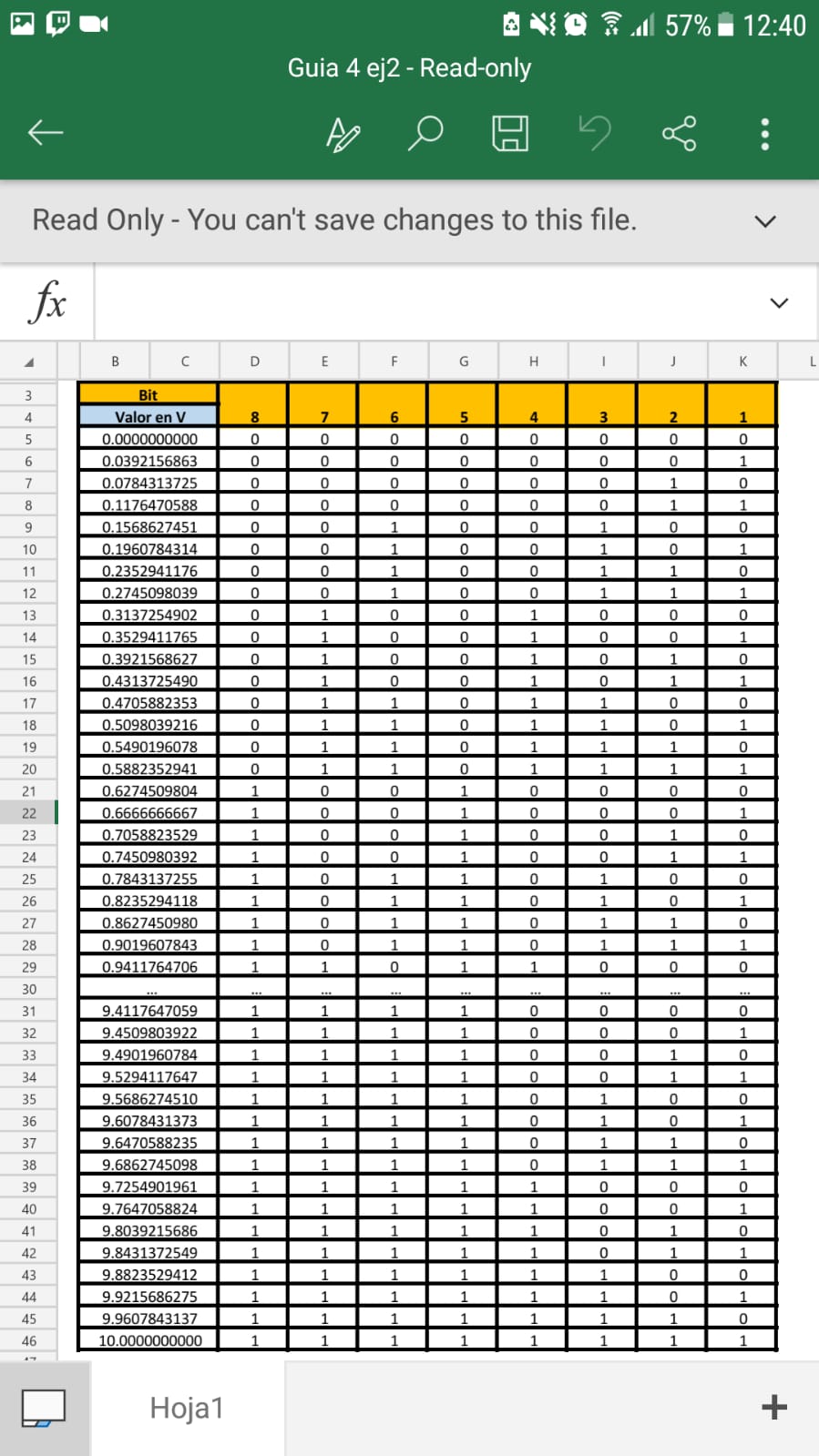
Ejercicio 1:



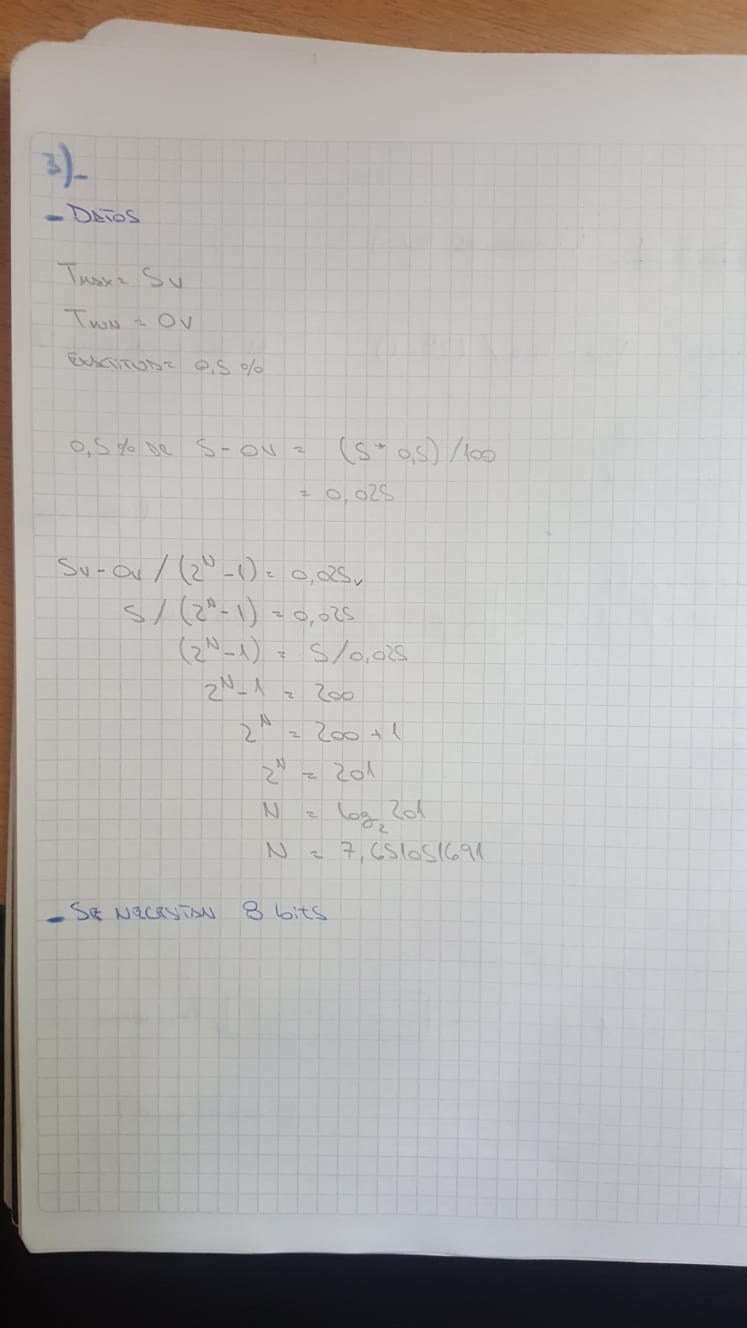




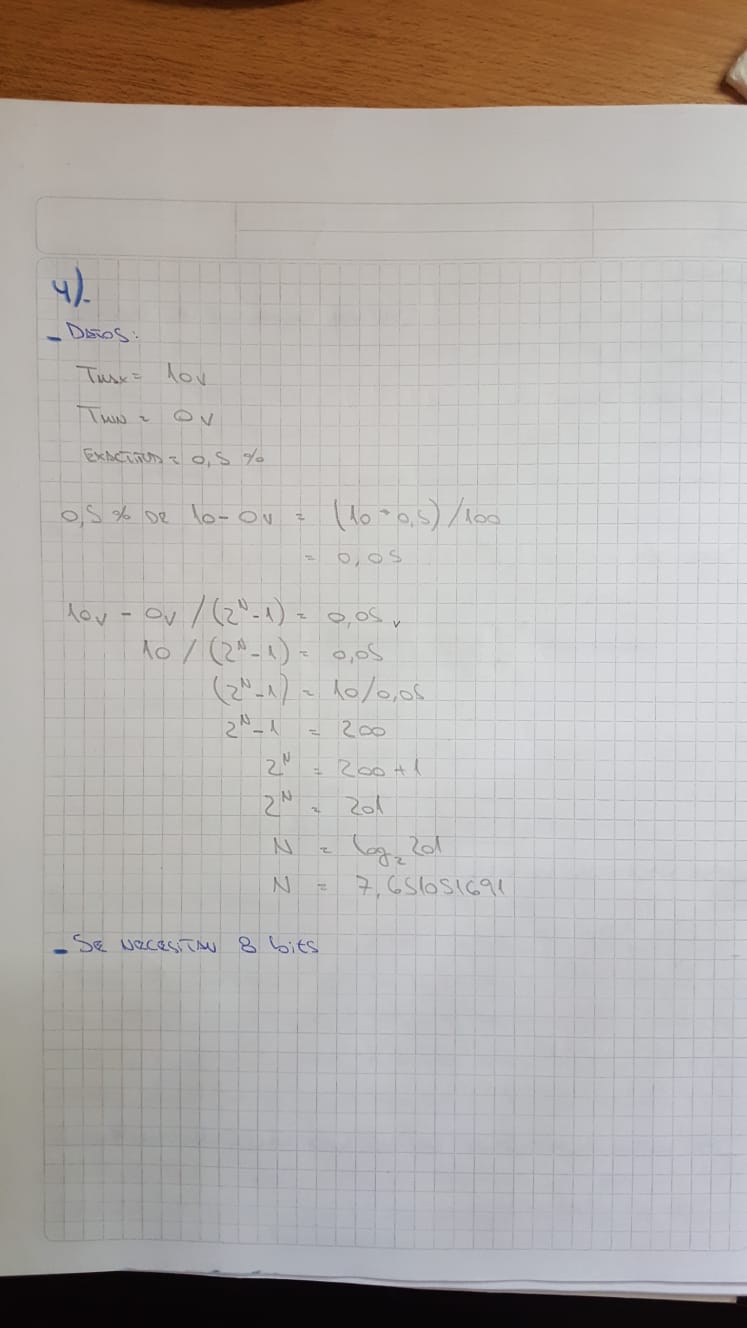
Ejercicio 2:

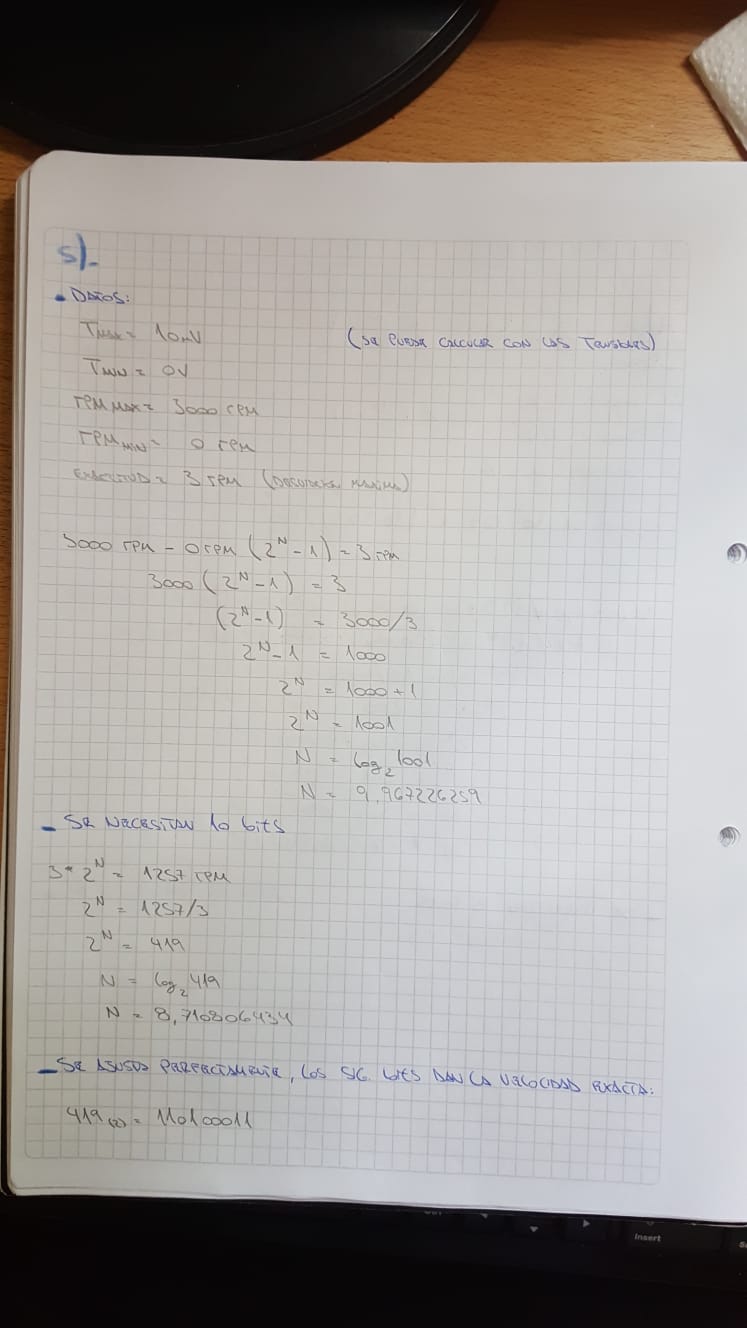


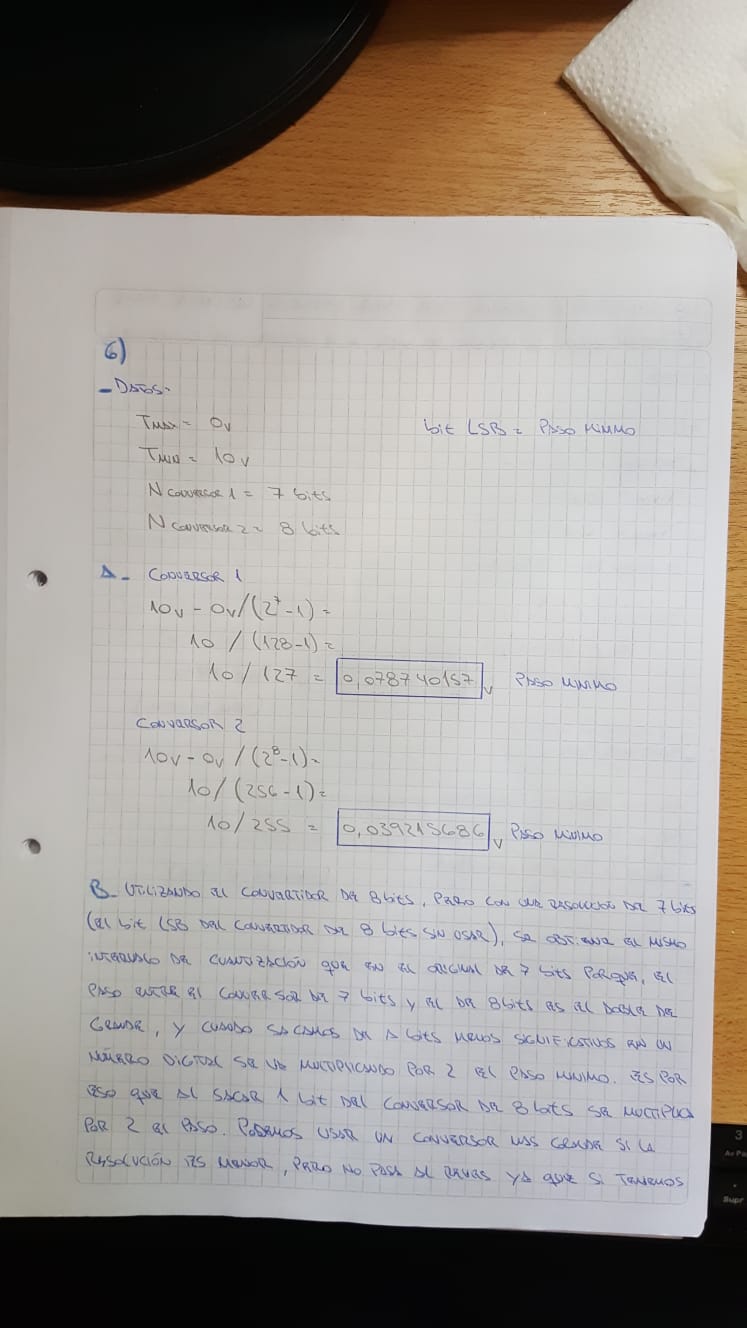
Ejercicio 3:



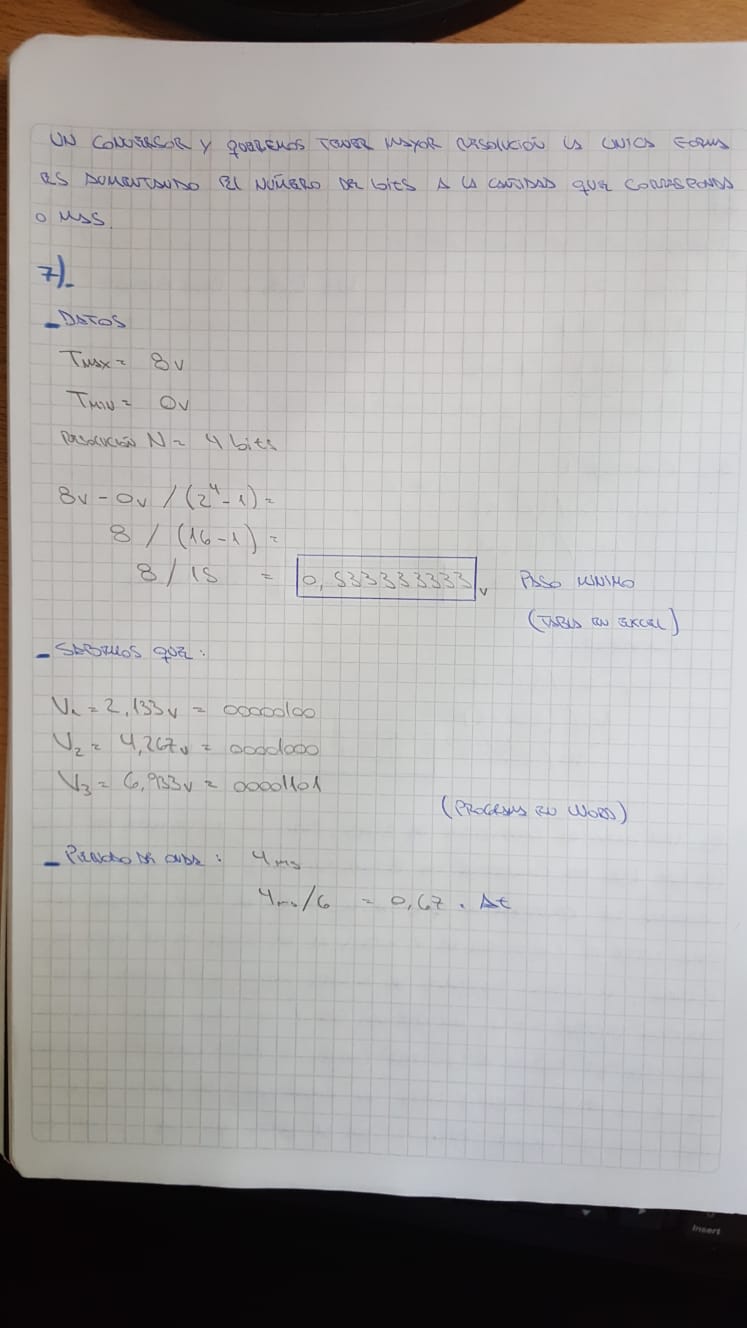
Ejercicio 4:

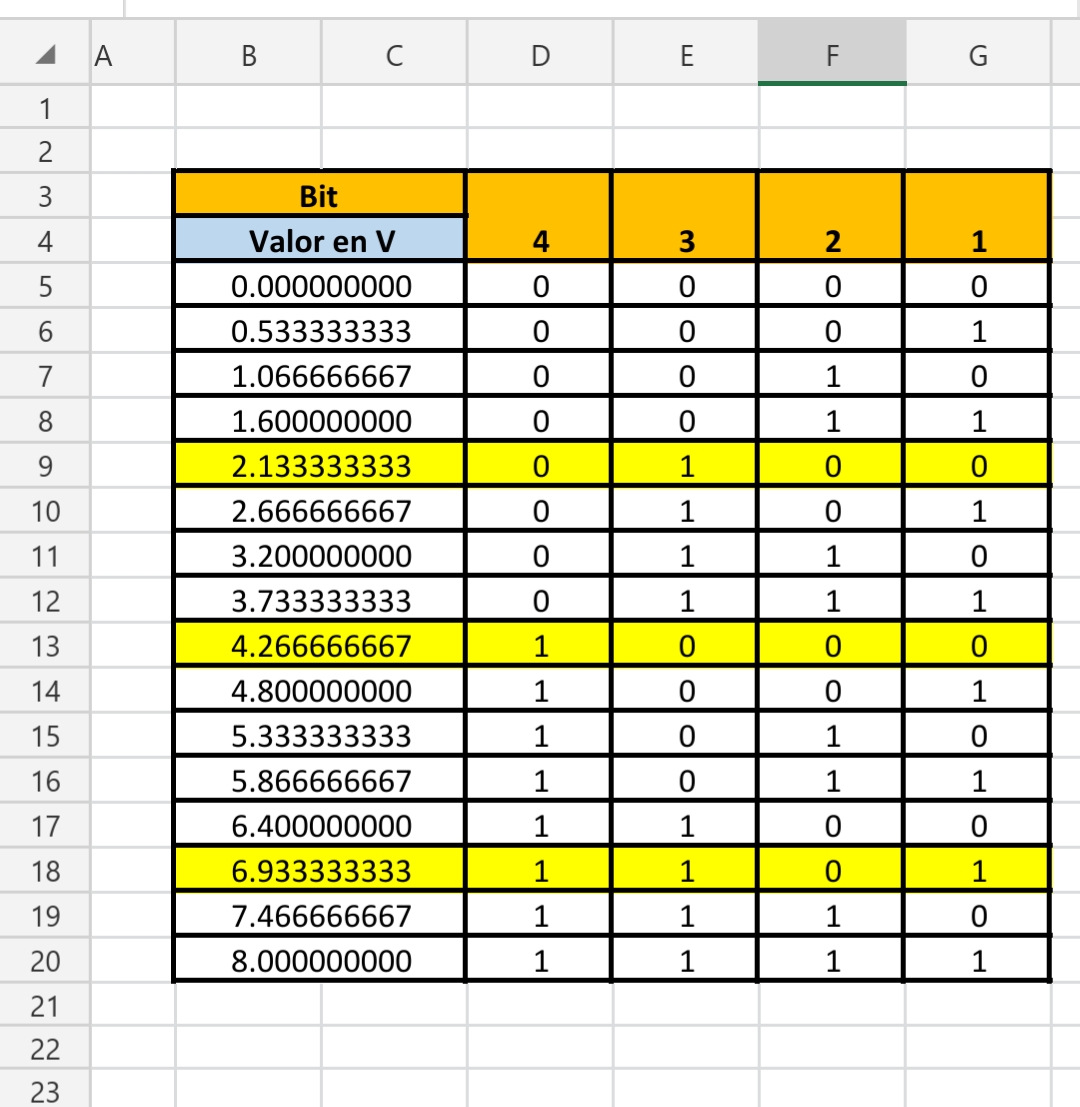


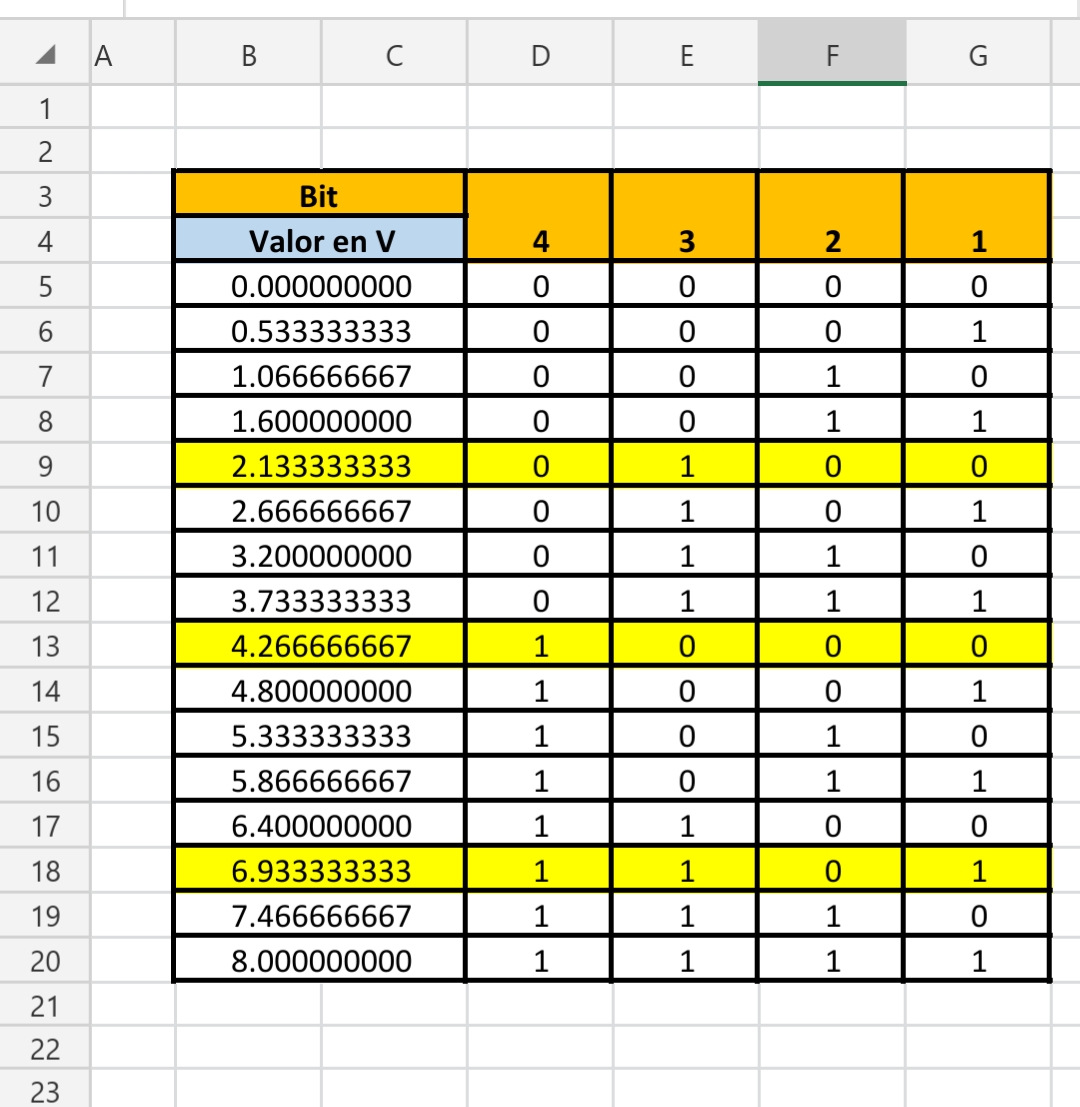
Ejercicio 5:

Ejercicio 6:

**B)** Utilizando el convertidor de 8 bits, pero con una resolución de 7 bits (el bit LSB del convertidor de 8 bits sin usar), se obtiene el mismo intervalo de cuantización que en el original de 7 bits porque, el paso entre el conversor de 7 bits y el de 8 bits es el doble de grande, y cuando sacamos de a bits menos significativos en un numero digital se va multiplicando por 2 el paso minimo. Es por eso que al sacar un bit del conversor de 8 bits de multiplica por 2 el paso. Podemos usar un conversor mas grande si la resolucion es menor, pero no pasa al reves ya que si tenemos un conversor y queremos tener mayor resolucion la unica forma es aumentando el numero de bits a la cantidad que corresponda o mas.

Ejercicio 7:





Codigo:

void crearOnda(){

escribirRegistro(3, 0b00000000);

while(true){

escribirRegistro(3, 0b00000100);

usleep(667);

escribirRegistro(3, 0b00001000);

usleep(667);

escribirRegistro(3, 0b00001101);

usleep(667);

escribirRegistro(3, 0b00001000);

usleep(667);

escribirRegistro(3, 0b00000100);

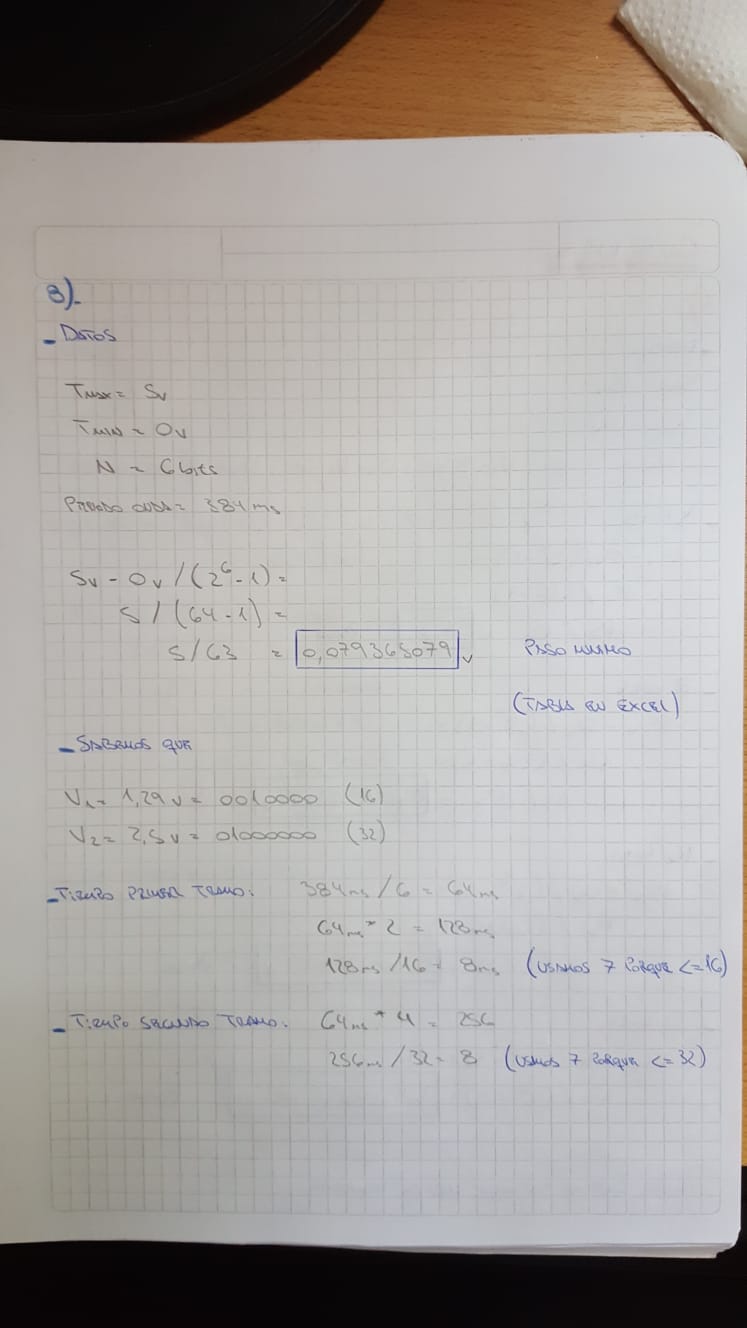
usleep(667);

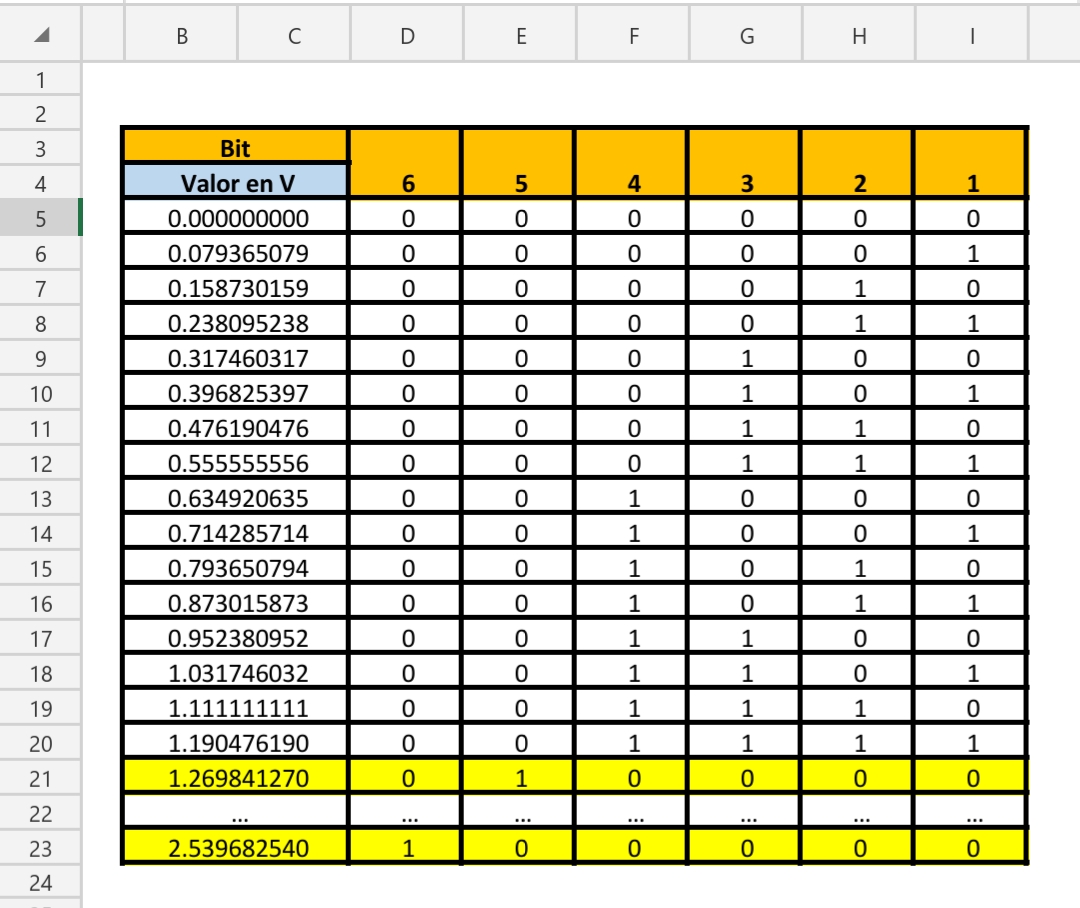
escribirRegistro(3, 0b00000000);

usleep(667);

}

}

Ejercicio 8:



Codigo:

void crearOnda2(){

char aux = 0b00000000;

while(true){

while(aux<=16){

escribirRegistro(3 , aux);

aux++;

usleep(7); }

aux = 0b00000000;

escribirRegistro(3 , aux);

while(aux<=32){

escribirRegistro(3 , aux);

aux++;

usleep(7); }

aux = 0b00000000;

}

}