

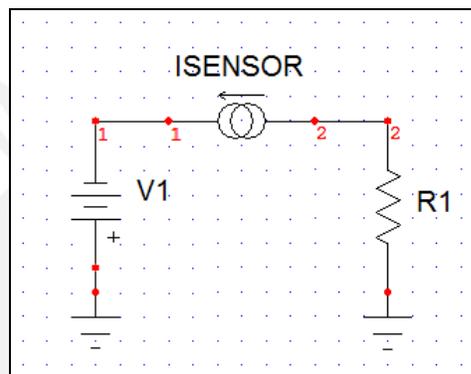
ANEXO N°1

CALCULO DE COMPONENTES DEL CIRCUITO DE SENSADO

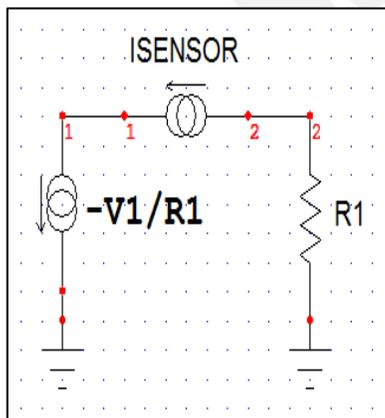
EQUIVALENTE DE THEVENIN

Aplicamos el teorema de Thevenin en el circuito de sensado para simplificar el circuito siendo nuestra carga a partir del condensador **C1**.

Por practicidad consideramos el sensor una fuente de corriente independiente.

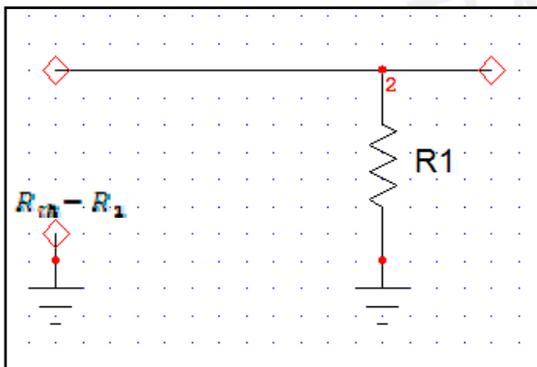


El voltaje de Thevenin se calcula luego de realizar transformación de fuentes:

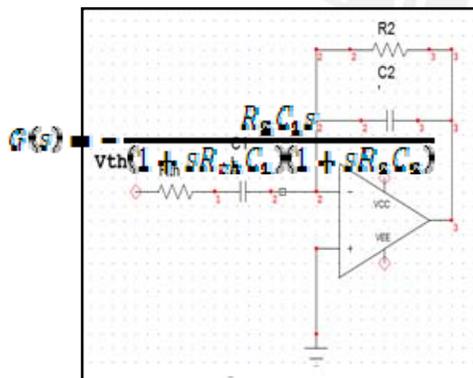


$$V_{th} = \left(-\frac{V_1}{R_1} + I_{SENSOR} \right) \times R_1 \quad , \forall \quad \frac{V_1}{R_1} > I_{SENSOR}$$

La resistencia de Thevenin se obtiene pasivando las fuentes:



El circuito de sensado es prácticamente un integrador derivador con limitación de ganancia, por lo tanto tiene la siguiente función de transferencia:



CÁLCULO DE COMPONENTES

Entonces, con la función de transferencia y las características en frecuencia de mi señal puedo obtener los valores de los componentes.

Atenuo el ruido rosa cuya influencia es más fuerte en las bajas frecuencias y elimino la corriente DC con el primer polo.

$$f_1 = 100\text{Hz} \rightarrow S = 2\pi 100 = \frac{1}{R_1 \times C_1}$$

$$\text{Asumo } R_1 = 10\text{ M} \rightarrow C_1 = 160\text{ pF}$$

Integro la señal de radiación cuya banda de frecuencias es mayor a 10KHz, por lo tanto mi segundo polo estará una década por encima de esa frecuencia.

$$f_2 = 1\text{ KHz} \rightarrow S = 2\pi 1000 = \frac{1}{R_2 C_2}$$

Asumo $C_2 = 33 \text{ pF} \rightarrow R_2 = 4.82 \text{ M}\Omega$

Reajustamos los valores calculados de los condensadores y las resistencias por valores comerciales sin perjudicar los polos obteniendo:

- ✓ $R_1 = 10 \text{ M} , \quad C_1 = 150 \text{ pF} , \quad f_1 = 106.10 \text{ Hz}$
- ✓ $R_2 = 5.1 \text{ M} , \quad C_2 = 33 \text{ pF} , \quad f_2 = 945.662 \text{ Hz}$



ANEXO N° 2

CALCULO DE LA POTENCIA RADIANTE

- Se han obtenido los siguientes datos de las hojas técnicas de los diodos:
 - Longitud de Onda de la luz roja (λ) = 644 nm
 - Eficacia Radiante (η) = 85 lm/W
 - Fotosensitividad (S , $\lambda=644\text{nm}$) = 0.46 A/W (Según gráfica)
 - Área Activa (A) = 100 mm²
 - **Intensidad Luminosa (I_v) = 0.8 med = 0.8 $\frac{\text{mlm}}{\text{Sr}}$** (Según gráfica)

Ahora podemos calcular el voltaje V_{OUT} :

Primero, calculamos el flujo radiante:

$$F = \frac{I_v}{\eta} = 0.8 \frac{\text{lm}}{\text{Sr}}$$

$$F = 5.68 \times 10^{-3} \frac{\text{W}}{\text{Sr}}$$

Segundo, asumiendo que el LED está a 1 cm del diodo PIN de silicio obtenemos los estereorradianes con la siguiente aproximación para luego calcular el poder radiante.

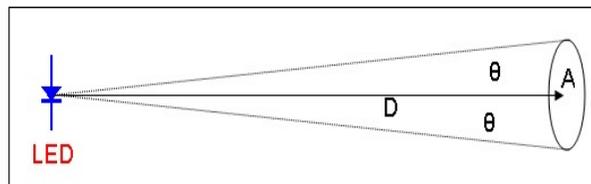


Fig. Geometría de la luz incidente.

Tomado de <http://www.iensign.com/LEDIntensity/index.html>

$$\Omega = A/D_s = \frac{100 \times 10^{-6}}{(1 \times 10^{-2})^2} = 1 \text{ Sr}$$

Entonces, con ambos datos podemos obtener el poder radiante:

$$P_R = F \times \Omega = 3.88 \times 10^{-3} \times 1 = 3.88 \times 10^{-3} \text{ W}$$

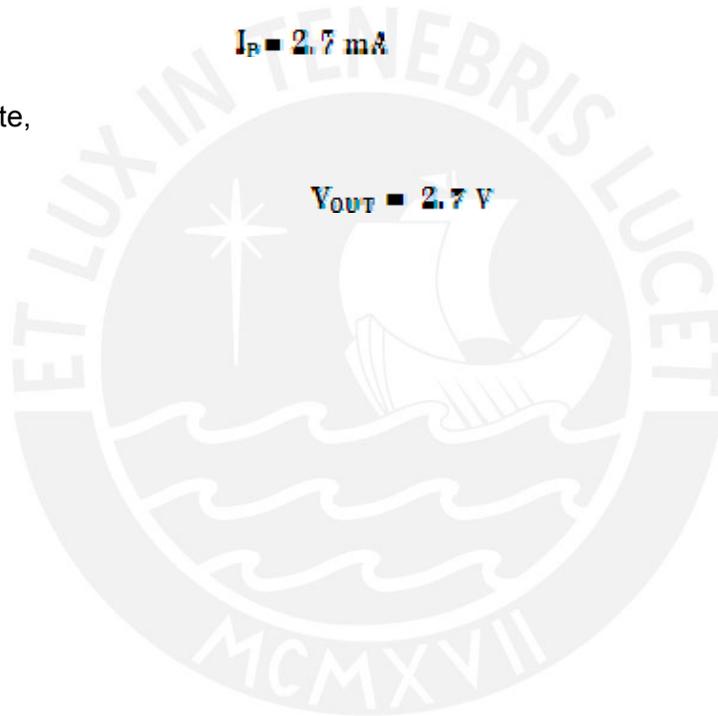
Después, podemos calcular la fotocorriente en el diodo PIN:

$$I_P = S \times P_R = 0.46 \times 3.8 \times 10^{-3}$$

$$I_P = 2.7 \text{ mA}$$

Finalmente,

$$V_{OUT} = 2.7 \text{ V}$$



ANEXO N°3

PROGRAMACIÓN EN MikroC for AVR

Se adjunta el código de prueba utilizado en las simulaciones, para efectos del capítulo 4 se utilizó un límite de alarma de 3mSv/h.

```

/* Tesis: Detector de Rx y Ry para Aplicaciones de Radiografía Industrial *
** Descripción: El siguiente programa se encarga de recibir un valor de *
**             voltaje correspondiente a la dosis de radiación recibida. *
**             Dependiendo del rango de este voltaje realiza diferentes *
**             funciones *
** Microcontrolador: Atmega8 *
** Reloj: 16 MHz *
** Autor: Carlos Hernán Salcedo Soto *
** Asesor: Ing. Willy Eduardo Carrera Soria *
*****/

```

```

char ms[5];
float VRad; float TDosis; float Dosis;
int d;

```

```

//Conexión LCD de 4 bits
sbit LCD_RS at PORTB0_bit;
sbit LCD_EN at PORTB1_bit;
sbit LCD_D4 at PORTB2_bit;
sbit LCD_D5 at PORTB3_bit;
sbit LCD_D6 at PORTB4_bit;
sbit LCD_D7 at PORTB5_bit;

```

```

sbit LCD_RS_Direction at DDB0_bit;
sbit LCD_EN_Direction at DDB1_bit;
sbit LCD_D4_Direction at DDB2_bit;
sbit LCD_D5_Direction at DDB3_bit;
sbit LCD_D6_Direction at DDB4_bit;
sbit LCD_D7_Direction at DDB5_bit;

```

```

// Subrutina de interrupción externa
void Modo_Pulsador() org IVT_ADDR_INT0 {

```

```

    SREG_I_bit = 0; //Deshabilito interrupciones
    delay_ms(1500); //Retardo de 1.5 segundos
    if (PORTD.B2==1){ // Reviso si el pulsador está activado
        if (d==1)

```

```

{d=2;} //Si d=1 (TASA DOSIS) entonces escribo d=2(DOSIS)
else{ //Si d=2 (DOSIS)entonces escribo d=1(TASA DOSIS)
d=1;}
goto finint0 ;}
else //((PORTC.B0==0)
{
PORTD.B5=1;
PORTD.B4=1; // Activación LED y Buzzer
Lcd_Cmd(_LCD_CLEAR);
Lcd_Cmd(_LCD_CURSOR_OFF);
Lcd_out(1,3,"Prueba de"); // Muestra mensaje "Prueba de alarma"
Lcd_out(2,5,"alarma"); // durante 3 segundos
delay_ms(3000);
Lcd_Cmd(_LCD_CLEAR);
Lcd_Cmd(_LCD_CURSOR_OFF);
Lcd_out(1,1,"Prueba Concluida");
PORTD= 0x00; // Desactiva LED y Buzzer
delay_ms(1000);
Lcd_Cmd(_LCD_CLEAR);
Lcd_Cmd(_LCD_CURSOR_OFF);} // Borra LCD
finint0: SREG_I_bit = 1;} //Habilito interrupciones y salgo de la subrutina

//Inicio de Programa Principal
void main() {
DDR = 0x38; // Configuración de Puertos E/S
PORTD = 0x00; // Inicializo las E/S a cero
d=1; // Inicializo variable
GICR = 0x40; // Activo la interrupción externa INT0
MCUCR = 0x02; // Configuro Interrupción INT0 por flanco de subida
SREG_I_bit = 1; // Habilito interrupciones

Lcd_Init(); //Inicializo LCD
Lcd_Cmd(_LCD_CLEAR); //Borro Pantalla
Lcd_Cmd(_LCD_CURSOR_OFF);

wait: if(PORTD6==0){ // Se queda leyendo PD6 hasta que aparezca la señal de
radiación

goto wait;}
else
{
VRad=ADC_Read(0); //Leo y guardo valores en la ADC
TDosis=VRad*5/1024; // Ec. Bits a Tasa (En este ejm Voltaje=Tasa
de dosis)
Dosis=TDosis/3600; // Conversión Tasa de dosis a dosis

```

```

if (d==1){ // Reviso si muestro dosis o tasa
if (TDosis<=3) // Reviso si la tasa está dentro del límite permitido
{
FloatToStr(TDosis, msj); // Transformo valor flotante a carácter ASCII
Lcd_out(1,3,msj); //Muestra la dosis en pantalla
Lcd_out(1,11,"mSv/h"); //Muestra las unidades
Lcd_out(2,1," ");
PORTD=0x00; //Apaga las alarmas
goto wait;}
else
{PORTD=0x30; //Activa las alarmas
FloatToStr(TDosis, msj);
Lcd_out(1,3,msj); //Muestra la dosis excesiva en pantalla
Lcd_out(1,11,"mSv/h"); //Muestra las unidades
Lcd_out(2,1,"TASA EXCESIVA"); //Muestra mensaje de alarma en la 2da fila
goto wait;}} // Vuelvo a esperar radiación

if(Dosis<=3){ // Reviso si la dosis está dentro del límite seguro
FloatToStr(Dosis, msj); //Convierte el valor flotante a carácter ASCII
Lcd_out(1,3,msj); //Muestra la dosis en pantalla
Lcd_out(1,7,"mSv"); //Muestra las unidades
Lcd_out(2,1," ");
PORTD=0x00; //Apaga las alarmas
}
else
{PORTD=0x30; //Activa las alarmas
FloatToStr(Dosis, msj); //Convierte el valor flotante a carácter ASCII
Lcd_out(1,1,msj); //Muestra dosis
Lcd_out(1,7,"mSv"); //Muestra las unidades
Lcd_out(2,1,"DOSIS EXCESIVA!"); //Muestra mensaje de alarma en la 2da fila
goto wait;} } // Vuelvo a esperar radiación */

```