

## Módulo RTC DS1302. Reloj de tiempo real

(Versión 07-12-17)

### Descripción

Un módulo RTC (*Real Time Clock*) o "Reloj de tiempo real" consiste en un circuito integrado alimentado por una batería el cual, en todo momento, registra la fecha, día de la semana y hora al igual que un reloj digital convencional.



### Reloj de Tiempo Real DS1302.

Cuenta segundos, minutos, horas, mes, día de la semana y año.

#### Descripción

El chip DS1302 contiene un reloj/calendario de tiempo real y 31 bytes de RAM estática. Se comunica con un microprocesador mediante una interfaz serial simple.

Solamente se requieren 3 alambres para comunicarse con el reloj y la RAM: CE (chip enable), I/O (data line) y SCLK (serial clock). Los datos pueden ser transferidos desde y hacia el reloj/RAM 1byte a la vez o 31 bytes de una sola vez. El DS1302 está diseñado para operar con muy baja potencia y retiene los datos y la información del reloj con menos de 1  $\mu$ W.

Para su funcionamiento se necesita un cristal de 32.768 kHz con capacitancia de carga de 6 pF (vendido por separado). **Recuerde que hablamos del Chip, el modulo si tiene el cristal.**

#### Características:

Cuenta segundos, minutos, horas, mes, día de la semana y año

Año bisiesto válido hasta el 2100

Soporta modo de 24 o 12 horas con indicador AM/PM

RAM estática de 31 bytes

Empaque: 8-pin DIP

Especificaciones Clave:

Requerimiento de potencia: 2 a 5 VDC; menos de ~300 nA

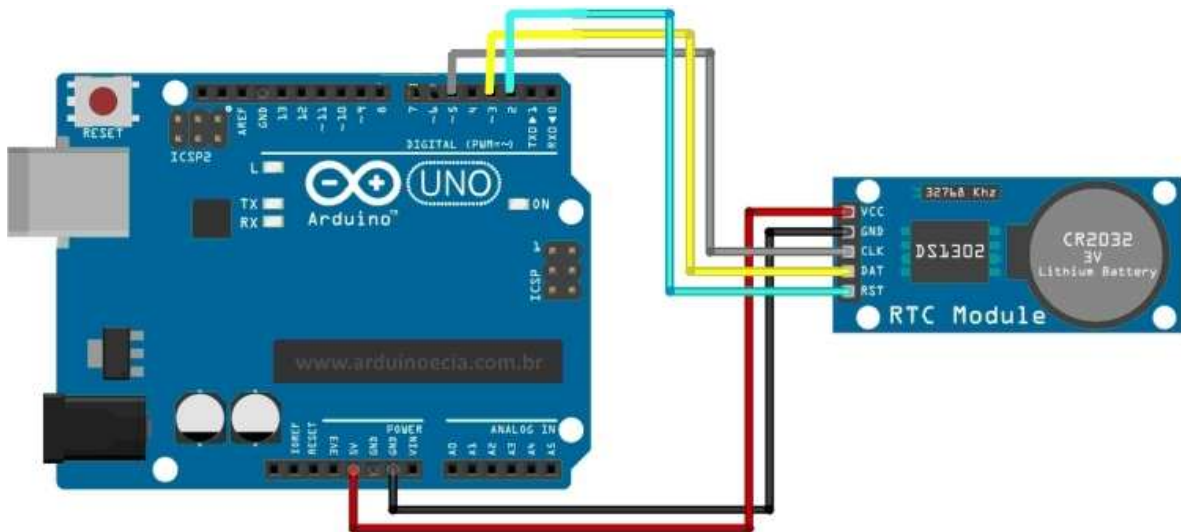
Comunicación: TTL compatible serial sincrónico de 3 líneas

Temperatura de Operación: -32° a +185° F (-0° a + 70° C)

Dimensiones: 0.4 x .025 in (10 x 6.4 mm)

Se le podrá asignar por única vez una fecha, hora y día de la semana, y mediante otro sketch aparte se lo comenzará a utilizar directamente.

Vale la pena conocer las prestaciones que este módulo ofrece, pero no siempre es necesario comprar y emplear un reloj de tiempo real, bien puede registrarse el tiempo mientras corre el programa solamente empleando software y la memoria interna EEPROM, aunque durante el tiempo en que la placa este sin alimentación esto no funcionará. Por lo tanto, si se desea tener en todo momento disponibles y actualizados la fecha, día de la semana y hora independientemente de la alimentación de la placa, este componente es la solución. Para el resto de los casos se lo consideraría innecesario. Visitar la entrada *Reloj por software*.



### Actividad

Para mostrar la mayor cantidad de prestaciones útiles de este módulo se presentará un sketch de prueba mediante el cual se configurarán todos los parámetros por única vez mas un segundo sketch mediante el cual se consultará por separado: año, mes, día de la semana, hora, minuto y segundos para ser mostrados en un display LCD, o Monitor Serial si no se cuenta con un LCD.

El primer paso será descargar la librería de este componente, [DS1302.zip](#)

Recuerde que encontrará una copia en la carpeta RECURSOS del Tutor de Arduino.

Luego abrir Arduino IDE e instalar la librería como ya conocemos. (Si no recuerda vea en Apuntes).

Habiendo hecho esto pueden probarse los ejemplos desde "Menú > Archivo > Ejemplos > DS1302", recuerde que en el IDE de Arduino cada vez que carga librerías también carga nuevos ejemplos.

O de lo contrario cargarse el primer sketch desde este apunte. Este primer sketch permite configurar la fecha y hora del Módulo DS1302.

- Placa: Arduino Uno
- IDE: V1.0.5 o superior
- Módulo: RTC DS1302

RTC ( ) vista frontal (Conexionado)



| (Vcc) | (GND) | (CLK) | (DAT) | (RST) |
|-------|-------|-------|-------|-------|
| [+5V] | [GND] | [D5]  | [D3]  | [D2]  |

### Sketch de prueba, configuración única

(O sea para poner el reloj en hora debemos usar este programa, y cargar en su listado la fecha y hora actual)

```
#include <DS1302.h>

// Inicializacion del modulo
DS1302 rtc(2, 3, 5);
Time t;

void setup(){
  // Desproteger contra escritura
  rtc.halt(false);
  rtc.writeProtect(false);

  rtc.setDOW(TUESDAY); // Configurar dia de la semana: MARTES.
  rtc.setTime(17, 00, 00); // Configurar hora en formato 24hs con min
y seg: 17:00:00 HORAS.
  rtc.setDate(4, 3, 2014); // Configurar fecha en formato
dia/mes/año: 4/3/2014.
}

void loop(){
  digitalWrite(13,HIGH); // El led integrado confirmara la
finalizacion.
}
```

Luego de cargar este programa en el Arduino, nuestro Modulo DS1302 conectado al Arduino, quedará en hora y fecha correcta.

### Sketch de prueba, puerto serie

```
#include <DS1302.h>

// Inicializacion del modulo.
DS1302 rtc(2, 3, 5);
Time t;

void setup() {
  // Inicializacion del puerto serie.
  Serial.begin(9600);
}

void loop() {
  // Obtencion de datos
  t = rtc.getTime();

  // Publicar en LCD
  // Se publicara el dia de la semana
  Serial.print("HOY:");
  if (t.dow == 1) Serial.print("lun"); // La variable t.dow (dia de la
semana) tedra valor de 1 para dia lunes y 7 para domingo.
  if (t.dow == 2) Serial.print("mar");
  if (t.dow == 3) Serial.print("mie");
  if (t.dow == 4) Serial.print("jue");
  if (t.dow == 5) Serial.print("vie");
  if (t.dow == 6) Serial.print("sab");
  if (t.dow == 7) Serial.print("dom");

  Serial.println(); // Linea aparte

  // Se publicaran datos de fecha, en numeros.
  Serial.print("DIA:");
  Serial.print(t.date, DEC); // Dia del mes

  Serial.println();

  // Mes
  Serial.print("MES:");
  Serial.print(t.mon);

  Serial.println();
```

```

// Año
Serial.print("DE:");
Serial.print(t.year, DEC);

Serial.println();

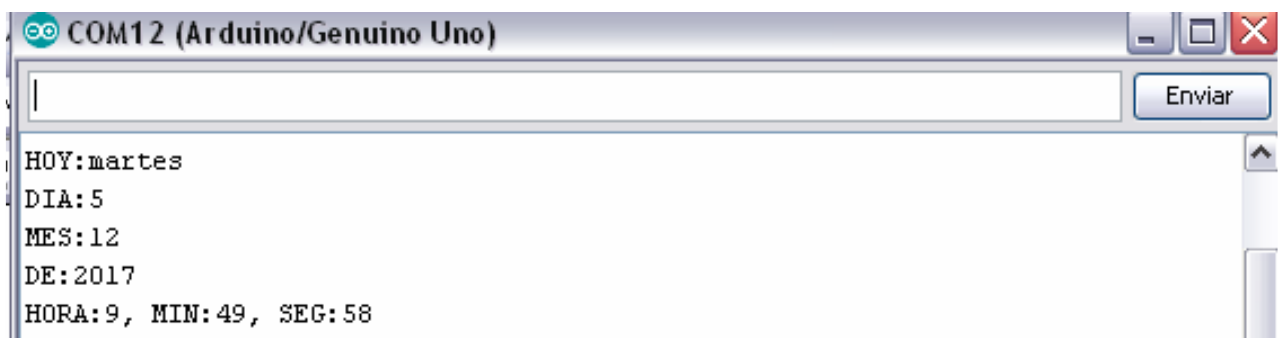
Serial.print("HORA:"); // Hora en formato 0-23.
Serial.print(t.hour, DEC);
Serial.print(", MIN:"); // Minutos.
Serial.print(t.min, DEC);
Serial.print(", SEG:"); // Segundos.
Serial.print(t.sec, DEC);

Serial.println();
Serial.println();
Serial.println();

delay(3000); // Demora para no sobrecargar las comunicaciones con el
modulo.
}

```

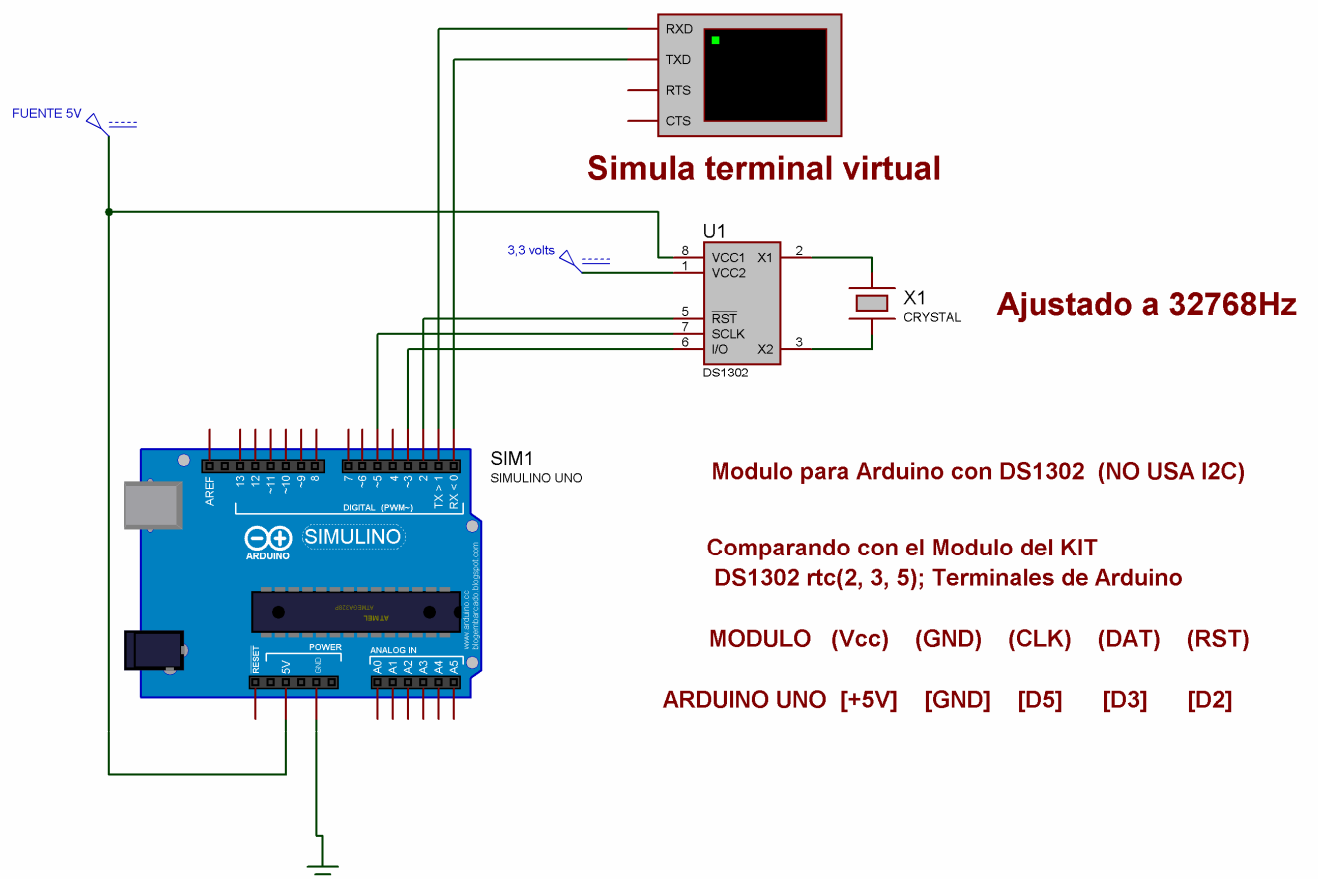
### Visualizando en Monitor Serie:



### En Proteus:

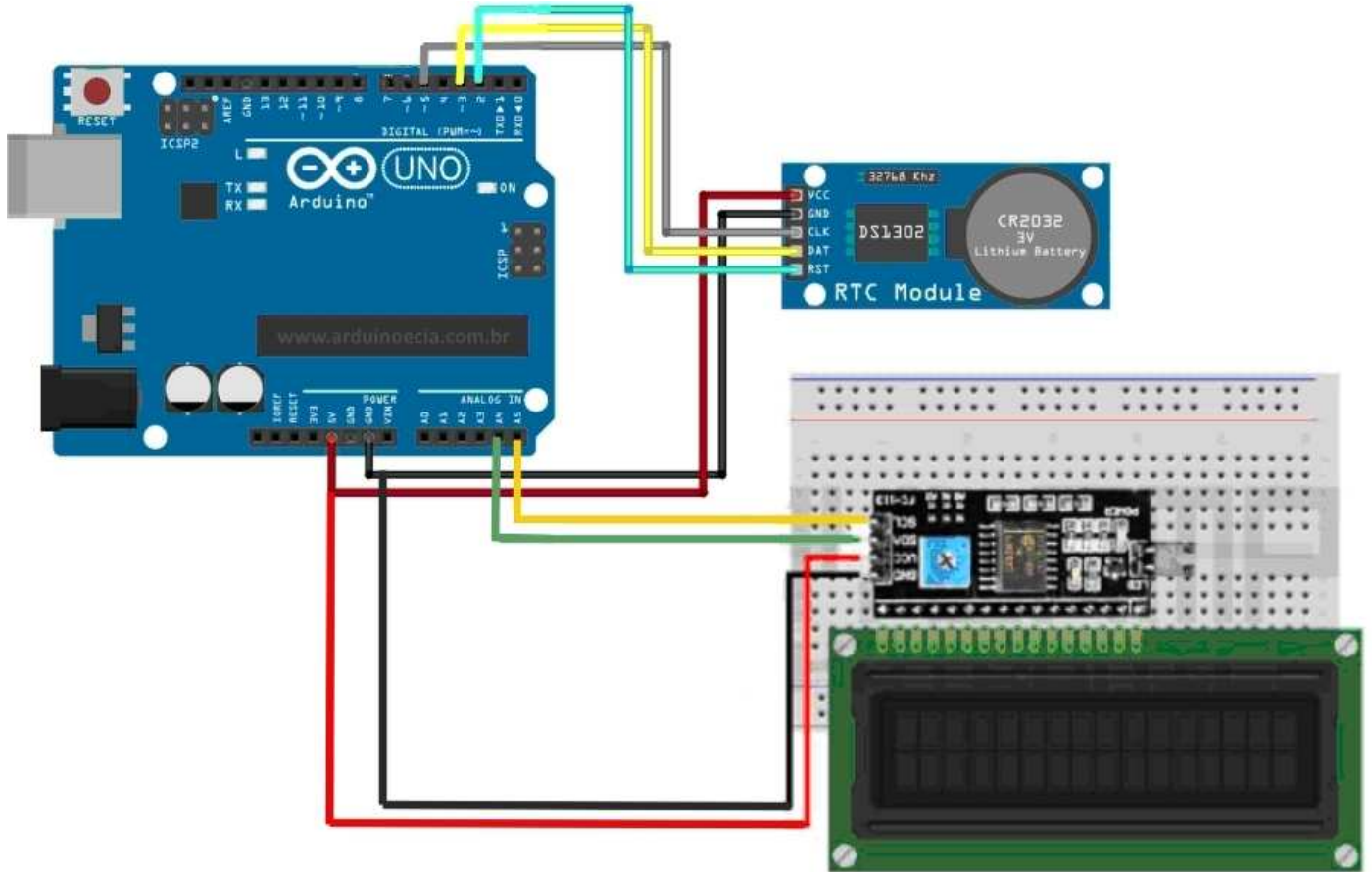
#### Uso del DS1302 Prueba por Monitor Serie

Al simular el modulo en Proteus, toma la hora del sistema (como simulacion)

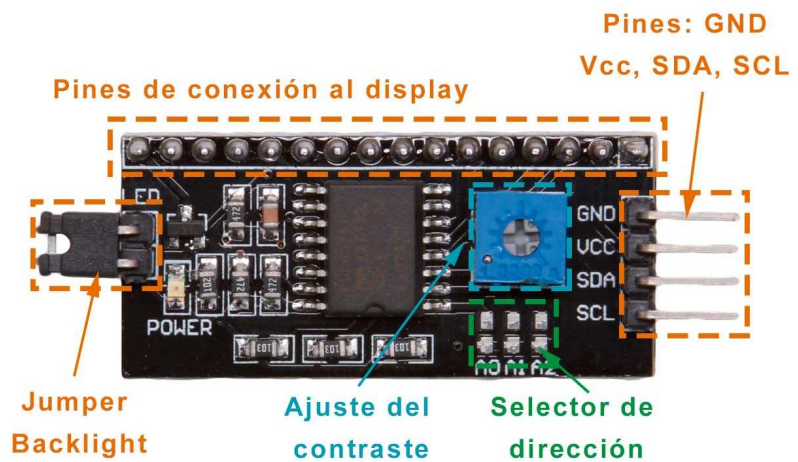


**Conexión del circuito Con LCD Visualizando fecha y hora**

| (Vcc) | (GND) | (CLK) | (DAT) | (RST) |
|-------|-------|-------|-------|-------|
| [+5V] | [GND] | [D5]  | [D3]  | [D2]  |



**Detalle del Adaptador I2C (puede haber diferencias entre distintos fabricantes)**



## Sketch de prueba, LCD I2C

```
//Reloj DS1302 visualizado en LCD 16x2 I2C
#include <LiquidCrystal_I2C.h>
#include <Wire.h>
//Es probable que deba usar una variante de esta libreria
//Se suministra el archivo LiquidCrystal_I2CDJB.ZIP
//para tal variante- lea como instalar librerias
//Crear el objeto lcd dirección 0x27 y 16 columnas x 2 filas
LiquidCrystal_I2C lcd(0x27,16,2); //
//Es probable que deba cambiar la direccion 0x27 por 0x3F
//Dependera de su LCD- Si tiene el PCF8574 o el PCF8574A

//--Referido al RELOJ-----
#include <DS1302.h>
// Inicializacion del modulo.
DS1302 rtc(2, 3, 5);
Time t;

void setup() {
// Inicializar el LCD
  lcd.init();

//Encender la luz de fondo.
  lcd.backlight();

// Escribimos el Mensaje en el LCD.
  lcd.print("Fecha y Hora");
}

void loop() {
  // Obtencion de datos del RELOJ
  t = rtc.getTime();

  // Ubicamos el cursor en la primera posición(columna:0) de la segunda
línea(fila:1)
  lcd.setCursor(0, 1);

// Publicar en LCD
  // Se publicara el dia de la semana
  // La variable t.dow (dia de la semana) 1 lunes y 7 domingo.

  // if (t.dow == 1) lcd.print("lu"); //Suprimido falta de espacio en LCD
  //if (t.dow == 2) lcd.print("ma");
  // if (t.dow == 3) lcd.print("mi");
  // if (t.dow == 4) lcd.print("ju");
  // if (t.dow == 5) lcd.print("vi");
  // if (t.dow == 6) lcd.print("sa");
  // if (t.dow == 7) lcd.print("do");

  //lcd.print(" "); //Separador

  // Se publicaran datos de fecha, en numeros.

  lcd.print(t.date, DEC); // Dia del mes
  lcd.print("/"); //Separador

  // Se publicara Mes

  lcd.print(t.mon);
  lcd.print("/"); //Separador
  // Se publicara Año

  lcd.print(t.year, DEC);
  lcd.print(" "); //Separador

// Hora en formato Hora-Minutos-Segundos

  lcd.print(t.hour, DEC); //Hora
  lcd.print(":"); //Separador

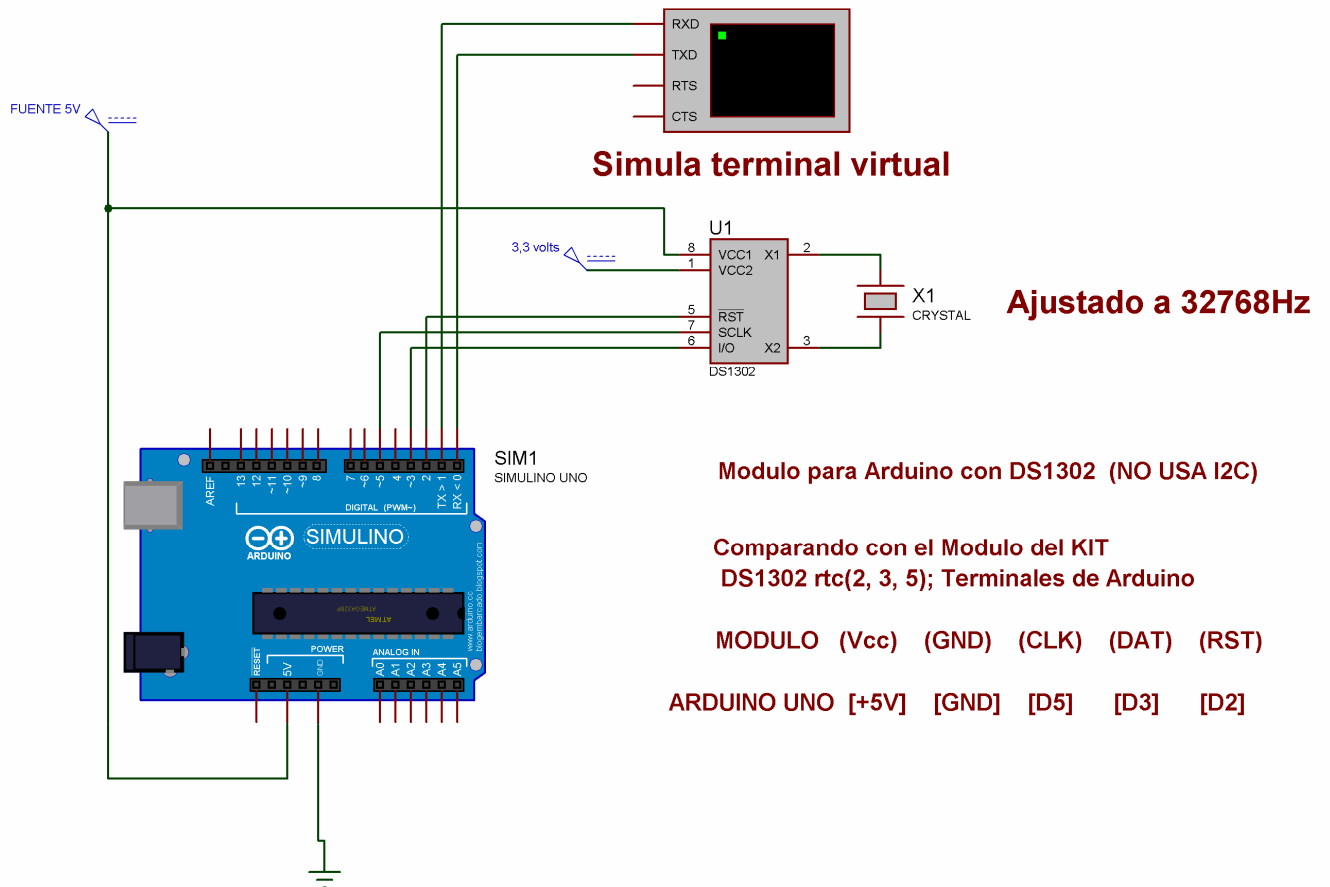
  lcd.print(t.min, DEC); // Minutos.
  //lcd.print(":"); //Separador
  // lcd.print(t.sec, DEC); //Segundos. No por falta de espacio en LCD
  //lcd.print(":"); //Separador - No por falta de espacio en LCD
  delay(1000);
}
```



## Esquema en Proteus del Circuito anterior

### Uso del DS1302 Prueba por Monitor Serie

Al simular el modulo en Proteus, toma la hora del sistema (como simulacion)



En el Tutor de Arduino encontrará esta simulación funcionando, para su observación y prueba.