

## 4duino Módulo en tiempo real RTC DS1302



### EAN CODE



### Descripción:

El chip de control de tiempo de carga lenta DS1302 contiene un reloj/calendario en tiempo real y 31 bytes de RAM estática. Se comunica con un microprocesador a través de una interfaz serial simple. El reloj/calendario en tiempo real proporciona información de segundos, minutos, horas, días, fechas, meses y años. La fecha de final de mes se ajusta automáticamente por meses con menos de 31 días, incluidas correcciones para el año bisiesto. El reloj funciona en formato de 24h o de 12h con un indicador de AM/PM. La interconexión del DS1302 con un microprocesador se simplifica mediante el uso de comunicación en serie síncrona. Solo se requieren tres cables para comunicarse con el reloj/RAM: CE, I/O (data line), and SCLK (serial clock). Los datos se pueden transferir hacia y desde el reloj/RAM 1 byte a la vez o en una ráfaga de hasta 31 bytes. El DS1302 está diseñado para operar con muy poca potencia y retener información de datos y reloj en menos de 1 $\mu$ W. El DS1302 es el sucesor del DS1202. Además de las funciones básicas de cronometraje del DS1202, el DS1302 tiene características adicionales de dos pines de alimentación para fuentes de alimentación principales y de respaldo, un cargador de goteo programable para VCC1, y siete bytes adicionales de memoria de scratchpad.

### CARACTERÍSTICAS

- El reloj en tiempo real cuenta segundos, minutos, horas, fecha del mes, mes, día de la semana y año con compensación del año bisiesto válido hasta 2100
- 31 x 8 Battery-Backed General-Purpose RAM
- Pines I/O de conteo mínimo seriales
- Full operación de 2.0V hasta 5.5V
- Utiliza menos de 300nA a 2.0V
- Single-Byte o Multiple-Byte (Burst Mode)
- Transferencia de datos para lectura o escritura de reloj o datos RAM
- Interfaz simple de 3 hilos



Número de artículo: 135945  
Número de fabricante: All\_RTC\_DS1302

TTL-Compatible (VCC = 5V)  
Rango de temperatura industrial opcional: -40°C hasta +85°C  
DS1202 Compatible

#### Ejemplo sketch 4duino:

**Code:** /\* FILE: ARD\_1302\_RTC\_Example\_Sketch\_HCMODU0035  
DATE: 03/09/13  
VERSION: 0.1

#### REVISIONS:

20/09/13 Created version 0.1

This is an example of how to use the Hobby Components 1302 Real Time Clock module (HCMODU0035). This example uses the RTC library written by Henning Karlsen (<http://www.henningkarlsen.com/electronics/>). This example sketch demonstrates how to read and write to the RTC module. To connect the module to your Arduino please see the following pinout:

#### PINOUT:

MODULE	ARDUINO
GND	GND
VCC	+5V
SCK	D4
I/O	D3
RST(CS)	D2

You may copy, alter and reuse this code in any way you like, but please leave reference to HobbyComponents.com in your comments if you redistribute this code.  
This software may not be used for the purpose of promoting or selling



products  
that directly compete with Hobby  
Components Ltd's own range of  
products.

THIS SOFTWARE IS PROVIDED  
"AS IS". HOBBY COMPONENTS  
MAKES NO WARRANTIES,  
WHETHER  
EXPRESS, IMPLIED OR  
STATUTORY, INCLUDING, BUT  
NOT LIMITED TO, IMPLIED  
WARRANTIES OF  
MERCHANTABILITY AND FITNESS  
FOR A PARTICULAR PURPOSE,  
ACCURACY OR LACK OF  
NEGLIGENCE.  
HOBBY COMPONENTS SHALL  
NOT, IN ANY CIRCUMSTANCES,  
BE LIABLE FOR ANY DAMAGES,  
INCLUDING, BUT NOT LIMITED TO,  
SPECIAL, INCIDENTAL OR  
CONSEQUENTIAL DAMAGES FOR  
ANY  
REASON WHATSOEVER. \*/

```
/* Define the DIO pins used for the  
RTC module */
```

```
#define SCK_PIN 4  
#define IO_PIN 3  
#define RST_PIN 2
```

```
/* Include the DS1302 library */  
#include <DS1302.h>
```

```
/* Initialise the DS1302 library */  
DS1302 rtc(RST_PIN, IO_PIN,  
SCK_PIN);
```

```
void setup()  
{  
  /* Clear the 1302's halt flag */  
  rtc.halt(false);  
  /* And disable write protection */  
  rtc.writeProtect(false);  
  
  /* Initialise the serial port */  
  Serial.begin(9600);  
}
```



```
/* Main program */  
void loop()  
{  
  
    /* Set the time and date to 16:30 on  
    the 3rd of September 2013 */  
    rtc.setDOW(MONDAY);  
    rtc.setTime(16,30,0);  
    rtc.setDate(3, 9, 2013);  
  
    /* Read the time and date once  
    every second */  
    while(1)  
    {  
        Serial.print("It is ");  
        Serial.print(rtc.getDOWStr());  
        Serial.print(" ");  
        Serial.print(rtc.getDateStr());  
        Serial.print(" ");  
        Serial.print("and the time is: ");  
        Serial.println(rtc.getTimeStr());  
  
        /* Wait before reading again */  
        delay (1000);  
    }  
}
```