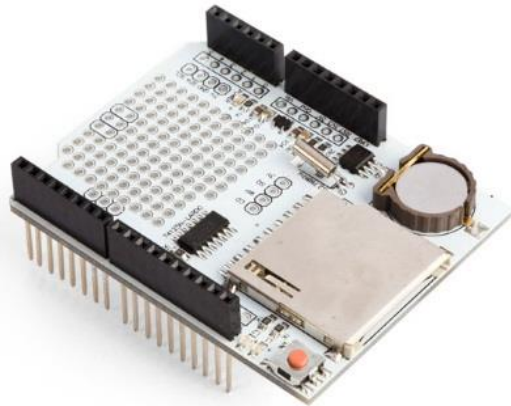
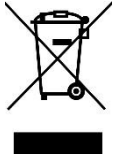


EN Arduino® Compatible Data Logging Shield

WPSH202



Introduction



To all residents of the European Union Important environmental information about this product

This symbol on the device or the package indicates that disposal of the device after its lifecycle could harm the environment. Do not dispose of the unit (or batteries) as unsorted municipal waste; it should be taken to a specialized company for recycling. This device should be returned to your distributor or to a local recycling service. Respect the local environmental rules.

If in doubt, contact your local waste disposal authorities.

Thank you for choosing Whadda! Please read the manual thoroughly before bringing this device into service. If the device was damaged in transit, do not install or use it and contact your dealer.

Safety Instructions



Read and understand this manual and all safety signs before using this appliance.



For indoor use only.

- This device can be used by children aged from 8 years and above, and persons with reduced physical, sensory or mental capabilities or lack of experience and knowledge if they have been given supervision or instruction concerning the use of the device in a safe way and understand the hazards involved. Children shall not play with the device. Cleaning and user maintenance shall not be made by children without supervision.

General Guidelines

- Refer to the Velleman® Service and Quality Warranty on the last pages of this manual.
- All modifications of the device are forbidden for safety reasons. Damage caused by user modifications to the device is not covered by the warranty.
- Only use the device for its intended purpose. Using the device in an unauthorized way will void the warranty.
- Damage caused by disregard of certain guidelines in this manual is not covered by the warranty and the dealer will not accept responsibility for any ensuing defects or problems.
- Nor Velleman Group nv nor its dealers can be held responsible for any damage (extraordinary, incidental or indirect) – of any nature (financial, physical...) arising from the possession, use or failure of this product.
- Keep this manual for future reference.

What is Arduino®

Arduino® is an open-source prototyping platform based on easy-to-use hardware and software. Arduino® boards are able to read inputs – light-on sensor, a finger on a button or a Twitter message – and turn it into an output – activating of a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board. To do so, you use the Arduino programming language (based on Wiring) and the Arduino® software IDE (based on Processing). Additional shields/modules/components are required for reading a twitter message or publishing online. Surf to www.arduino.cc for more information.

Product Overview

A dedicated and well-designed data logging shield for Arduino®.

The SD card interface works with FAT16 or FAT32 formatted cards. The 3.3 V level shifter circuitry prevents damage to your SD card.

The real-time clock (RTC) keeps the time going even when the Arduino® is unplugged. The battery back-up lasts for years.

Works with Arduino® Uno, Leonardo or ADK/Mega R3 or higher. ADK/Mega R2 or lower are not supported.

Specifications

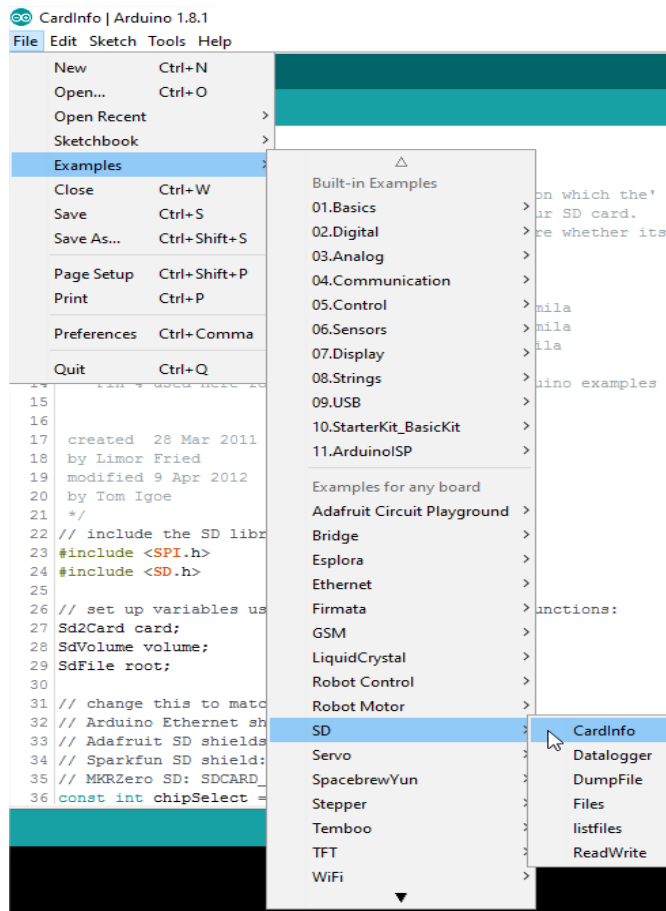
- back-up battery: 1 x CR1220 battery (incl.)
- dimensions: 43 x 17 x 9 mm

Testing

1. Plug your data logging shield into your Arduino® Uno compatible board (e.g. WPB100).
2. Insert a formatted SD card (FAT16 or FAT32) into the slot.

Testing the SD Card

1. In the Arduino® IDE, open the sample sketch [Cardinfo].



- Your data logging shield uses chipSelect 10 instead of chipSelect 4. Change line 36 in the sketch to:

const int chipSelect = 10;



IMPORTANT



The ATmega2560-based MEGA compatible (e.g. WPB101) and ATmega32u4-based Leonardo compatible (e.g. WPB103) development boards do not use the same hardware SPI pin-out. If you are using one of these boards, please specify the pins used for SPI communication with the SD card. For the VMA202, these are pins 10, 11, 12 and 13.

In the CardInfo sketch, modify line:

while (!card.init(SPI_HALF_SPEED, chipSelect)) {

to:

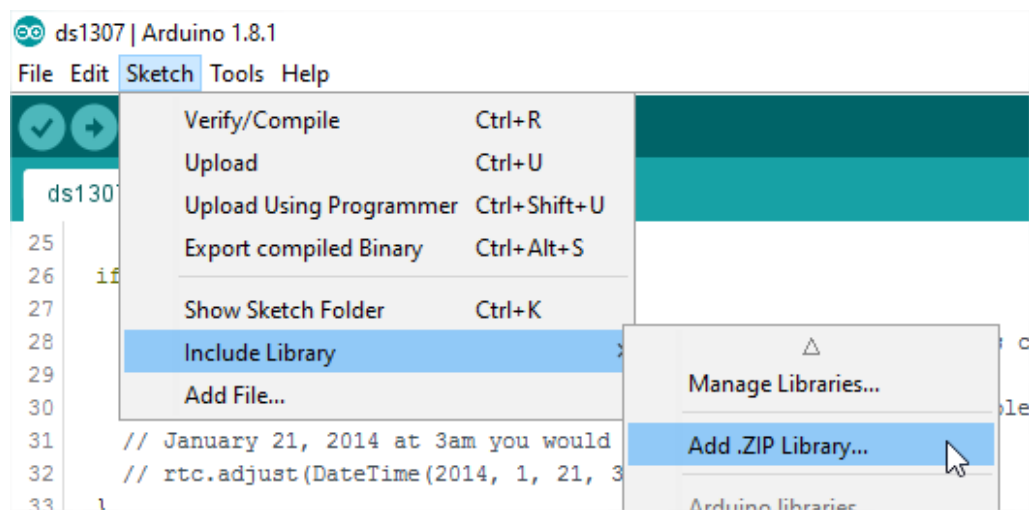
while (!card.init(SPI_HALF_SPEED,1,11,12,13)) {

Also, an updated SD library is needed to avoid error messages. How to replace the SD library:

1. Download the updated SD library from the products page on www.velleman.eu. Make sure the Arduino® IDE is not running.
2. Go to C:\Program Files\Arduino and create a new map, e.g. SD Backup.
3. Go to C:\Program Files\Arduino\libraries\SD and move all files and maps to your newly created map.
4. Extract the downloaded SD library into the now empty SD map. Make sure that the .h and .cpp files are directly under C:\Program Files\Arduino\libraries\SD.
5. Start the Arduino® IDE.

Testing the RTC (Real-Time Clock)

1. Download the RTCLib.zip file from the products page on www.velleman.eu.
2. In the Arduino® IDE select Sketch → Include Library → Add .ZIP Library... Select the RTCLib.zip file you downloaded.





whadda.com