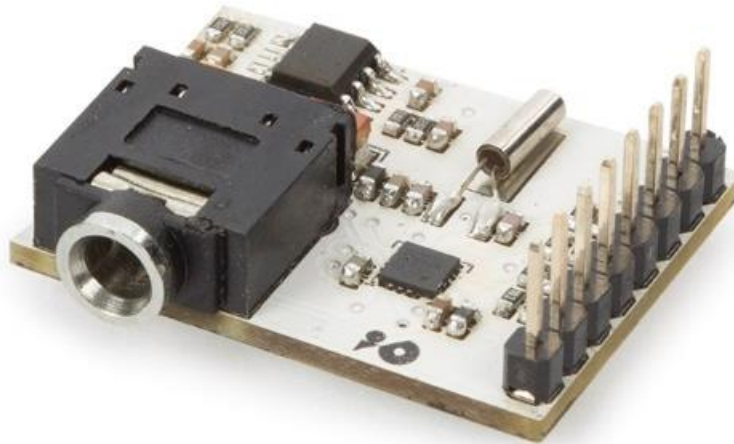


EN SI4703 FM tuner evaluation board

WPB444



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

This board is a radio and RDS receiver combined with an efficient 150 mW audio amplifier. It is ideal if you want to create your own clock radio or if you want to add FM functionality to your projects.

Specifications

- interface: I2C
- power/logic level: 3.3-5 V
- sensitivity: 1.7 μ emf
- max. output power: 150 mW (8 Ω)
- default gain: 18 dB (8 x)(adjustable)
- dimensions: 29 x 22 mm

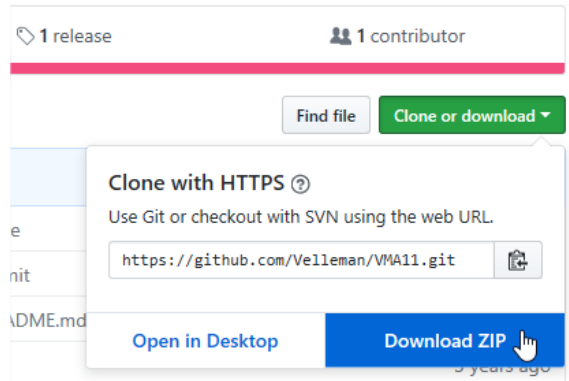
Connection to Arduino® – Pin Assignment

WPB444		WPB100
5V	▶	5V
GND	▶	GND
A4	▶	SDIO
A5	▶	SCLK
D2		RST

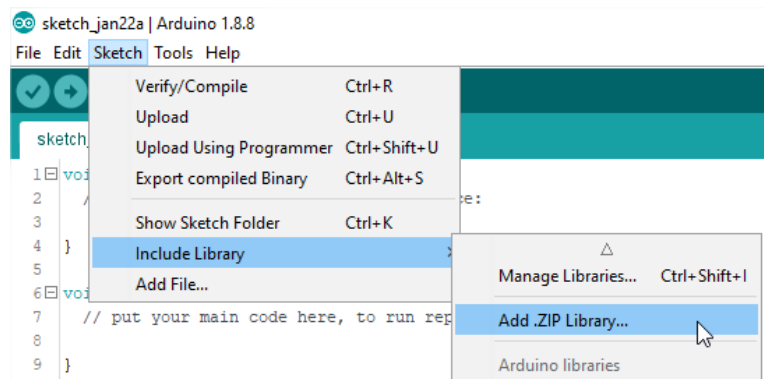
As with any receiver, the WPB444 needs an antenna. The WPB444 antenna is connected to the grounding of your earphone cable, so an earphone or audio cable must be plugged in to operate the WPB444 properly.

Example

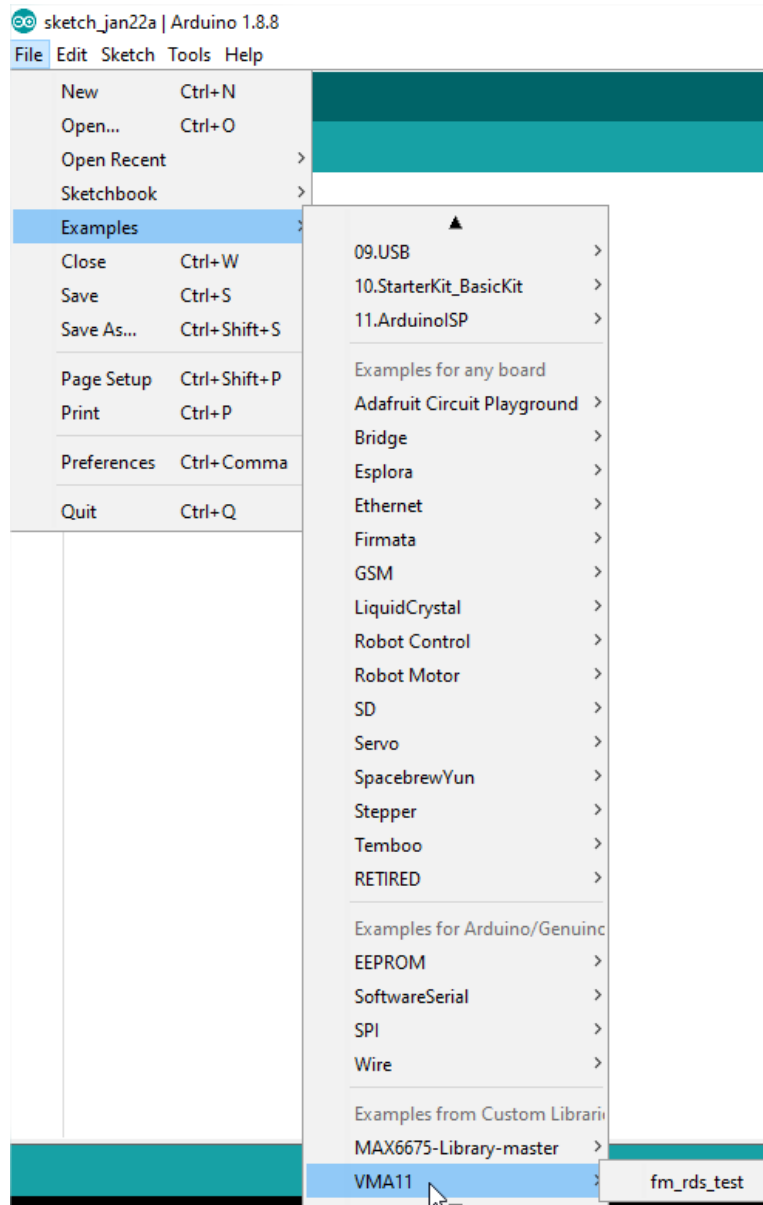
Download the WPB444 library from <https://github.com/Velleman/VMA11>. The easiest way to do this, is using the zip file.



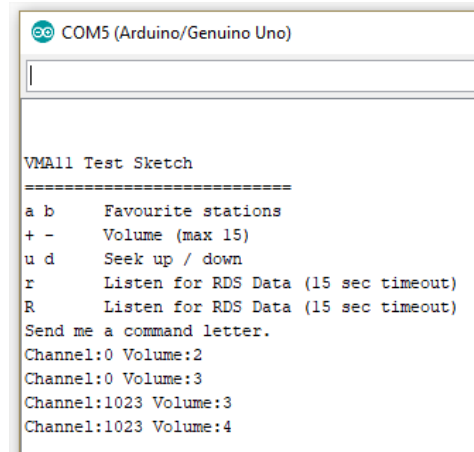
Now, open the Arduino® IDE and add the downloaded master zip file.



Load the example fm_rds_test sketch.



Compile and upload this sketch and listen to the radio! Do not forget the earphones as antenna. The volume, channel and RDS controls can be found by opening the serial monitor. See the example below:



```

COM5 (Arduino/Genuino Uno)

VMA11 Test Sketch
=====
a b   Favourite stations
+ -   Volume (max 15)
u d   Seek up / down
r     Listen for RDS Data (15 sec timeout)
R     Listen for RDS Data (15 sec timeout)
Send me a command letter.
Channel:0 Volume:2
Channel:0 Volume:3
Channel:1023 Volume:3
Channel:1023 Volume:4
  
```

Code

```

#include <VMA11.h>
#include <Wire.h>

int resetPin = 2; // these are the connections between VMA444 and the UNO VMA100
int SDIO = A4;
int SCLK = A5;

VMA11 radio(resetPin, SDIO, SCLK);
int channel;
int volume;
char rdsname[9];
char rdsrt[65];
char previousRadioText[65];
uint8_t lastChar;
void setup()
{
  Serial.begin(9600);
  Serial.println("\n\nVMA11 Test Sketch");
  Serial.println("=====");
  Serial.println("a b   Favourite stations");
  Serial.println("+ -   Volume (max 15)");
  Serial.println("u d   Seek up / down");
  Serial.println("r     Listen for RDS Data (15 sec timeout)");
  Serial.println("R     Listen for RDS Data (15 sec timeout)");
  Serial.println("Send me a command letter.");

  radio.powerOn();
  radio.setVolume(1);
  volume=1;
  
```

```
radio.setChannel(1021);

memset(previousRadioText,0,65);
memset(rdsrt,0,65);
}

void loop()
{
  if(radio.readRDSRadioText(rdsrt))
  {

    if(strcmp(rdsrt,previousRadioText))
    {
      Serial.println(rdsrt);
      strcpy(previousRadioText,rdsrt);
    }

  }
  if (Serial.available())
  {
    char ch = Serial.read();
    if (ch == 'u')
    {
      channel = radio.seekUp();
      displayInfo();
    }
    else if (ch == 'd')
    {
      channel = radio.seekDown();
      displayInfo();
    }
    else if (ch == '+')
    {
      volume ++;
      if (volume >=16) volume = 15;
      radio.setVolume(volume);
      displayInfo();
    }
    else if (ch == '-')
    {
      volume --;
      if (volume < 0) volume = 0;
      radio.setVolume(volume);
      displayInfo();
    }
    else if (ch == 'a')
    {
      channel = 1015; // Rock FM
      radio.setChannel(channel);
      displayInfo();
    }
    else if (ch == 'b')
```

```
{
  channel = 925; // KBCO in Boulder
  radio.setChannel(channel);
  displayInfo();
}
else if (ch == 'r')
{
  // The calling of readRDS and printing of rdsname really need
  // to be looped to catch all of the data...
  // this will just print a snapshot of what is in the Si4703 RDS buffer...
  radio.readRDSRadioStation(rdsname);
  Serial.println(rdsname);
}
else if (ch == 'R')
{
  // The calling of readRDS and printing of rdsrt really need
  // to be looped to catch all of the data...
  // this will just print a snapshot of what is in the Si4703 RDS buffer...
  //Serial.println("RDS listening - screen");

  radio.readRDS(rdsname,rdsrt,&lastChar);
  Serial.println(rdsrt);
}
}
}

void displayInfo()
{
  Serial.print("Channel:"); Serial.print(channel);
  Serial.print(" Volume:"); Serial.println(volume);
}
```




whadda.com



Modifications and typographical errors reserved - © Velleman Group nv. WPB444_v01
Velleman Group nv, Legen Heirweg 33 - 9890 Gavere.