

# VirtualWire

[Main Page](#)[Files](#)[Examples](#)

## VirtualWire library for Arduino and other boards

This is the VirtualWire library.

### END OF LIFE NOTICE

This VirtualWire library has now been superseded by the RadioHead library <http://www.airspayce.com/mikem/arduino/RadioHead> RadioHead and its RH\_ASK driver provides all the features supported by VirtualWire, and much more besides, including Reliable Datagrams, Addressing, Routing and Meshes. All the platforms that VirtualWire supported are also supported by RadioHead.

This library will no longer be maintained or updated, but we will continue to publish it for the benefit of the the community. Nevertheless we recommend upgrading to RadioHead where possible.

VirtualWire is a library for Arduino, Maple and others that provides features to send short messages, without addressing, retransmit or acknowledgment, a bit like UDP over wireless, using ASK (amplitude shift keying). Supports a number of inexpensive radio transmitters and receivers. All that is required is transmit data, receive data and (for transmitters, optionally) a PTT transmitter enable. Can also be used over various analog connections (not just a data radio), such as the audio channel of an A/V sender

It is intended to be compatible with the RF Monolithics ([www.rfm.com](http://www.rfm.com)) Virtual Wire protocol, but this has not been tested.

Does not use the Arduino UART. Messages are sent with a training preamble, message length and checksum. Messages are sent with 4-to-6 bit encoding for good DC balance, and a CRC checksum for message integrity.

But why not just use a UART connected directly to the transmitter/receiver? As discussed in the RFM documentation, ASK receivers require a burst of training pulses to synchronize the transmitter and receiver, and also requires good balance between 0s and 1s in the message stream in order to maintain the DC balance of the message. UARTs do not provide these. They work a bit with ASK wireless, but not as well as this code.

This library provides classes for

- VirtualWire: unaddressed, unreliable messages

Example Arduino programs are included to show the main modes of use.

The version of the package that this documentation refers to can be downloaded from

<http://www.airspayce.com/mikem/arduino/VirtualWire/VirtualWire-1.27.zip> You can find the latest version at <http://www.airspayce.com/mikem/arduino/VirtualWire>

You can also find online help and disussion at <http://groups.google.com/group/virtualwire> Please use that group for all questions and discussions on this topic. Do not contact the author directly, unless it is to discuss commercial licensing. Before asking a question or reporting a bug, please read <http://www.catb.org/esr/faqs/smart-questions.html>

### Theory of operation

See ASH Transceiver Software Designer's Guide of 2002.08.07 [http://www.rfm.com/products/apnotes/tr\\_swg05.pdf](http://www.rfm.com/products/apnotes/tr_swg05.pdf)

### Supported Hardware

A range of communications hardware is supported. The ones listed below are available in common retail outlets in Australia and other countries for under \$10 per unit. Many other modules may also work with this software.

Runs on a wide range of Arduino processors using Arduino IDE 1.0 or later. Also runs on on Energia with MSP430G2553 / G2452 and Arduino with ATmega328 (courtesy Yannick DEVOS - XV4Y), but untested by us. It also runs on Teensy 3.0 and 3.1 (courtesy of Paul Stoffregen) using the Arduino IDE 1.0.5 and the Teensyduino addon 1.18. Also compiles and runs on ATtiny85 in Arduino environment, courtesy r4z0r7o3. Also compiles on maple-ide-v0.0.12, and runs on Maple, flymaple 1.1 etc. Runs on ATmega8/168 (Arduino Diecimila, Uno etc), ATmega328 and can run on almost any other AVR8 platform, without relying on the Arduino framework, by properly configuring the library using 'VirtualWire\_Config.h' header file for describing the access to IO pins and for setting up the timer. From version 1.22 the library can be compiled by a C compiler (by renaming VirtualWire.cpp into VirtualWire.c) and can be easily integrated with other IDEs like 'Atmel Studio' for example (courtesy of Alexandru Mircescu).

- Receivers
  - RX-B1 (433.92MHz) (also known as ST-RX04-ASK)
- Transmitters:
  - TX-C1 (433.92MHz)
- Transceivers

- DR3100 (433.92MHz)

For testing purposes you can connect 2 VirtualWire instances directly, by connecting pin 12 of one to 11 of the other and vice versa, like this for a duplex connection:

```
Arduino 1      wires      Arduino 1
D11-----D12
D12-----D11
GND-----GND
```

You can also connect 2 VirtualWire instances over a suitable analog transmitter/receiver, such as the audio channel of an A/V transmitter/receiver. You may need buffers at each end of the connection to convert the 0-5V digital output to a suitable analog voltage.

Caution: ATTiny85 has only 2 timers, one (timer 0) usually used for millis() and one (timer 1) for PWM analog outputs. The VirtualWire library, when built for ATTiny85, takes over timer 0, which prevents use of millis() etc but does permit analog outputs.

## Installation

To install, unzip the library into the libraries sub-directory of your Arduino application directory. Then launch the Arduino environment; you should see the library in the Sketch->Import Library menu, and example code in File->Sketchbook->Examples->VirtualWire menu.

## Donations

This library is offered under a free GPL license for those who want to use it that way. We try hard to keep it up to date, fix bugs and to provide free support. If this library has helped you save time or money, please consider donating at <http://www.airspayce.com> or here:

## Trademarks

VirtualWire is a trademark of AirSpayce Pty Ltd. The VirtualWire mark was first used on April 20 2008 for international trade, and is used only in relation to data communications hardware and software and related services. It is not to be confused with any other similar marks covering other goods and services.

## Copyright

This software is Copyright (C) 2011-2014 Mike McCauley. Use is subject to license conditions. The main licensing options available are GPL V2 or Commercial:

### Open Source Licensing GPL V2

This is the appropriate option if you want to share the source code of your application with everyone you distribute it to, and you also want to give them the right to share who uses it. If you wish to use this software under Open Source Licensing, you must contribute all your source code to the open source community in accordance with the GPL Version 2 when your application is distributed. See <http://www.gnu.org/copyleft/gpl.html>

### Commercial Licensing

This is the appropriate option if you are creating proprietary applications and you are not prepared to distribute and share the source code of your application. Contact [info@airspayce.com](mailto:info@airspayce.com) for details.

## Revision History

### Version

- 1.0 Original release
- 1.1 2008-06-24 Now can compile for atmega8 Reported by creatrope
- 1.2 2009-03-30 Fixed a problem that prevented compiling with arduino-0015 Reported by Jaime Castro
- 1.3 2009-04-01 Fixed a compatibility problem with ATMEGA328 of the new arduino Now use SIGNAL(TIMER1\_COMPA\_vect) instead of ISR(SIG\_OUTPUT\_COMPARE1A) as discussed in <http://www.arduino.cc/cgi-bin/yabb2/YaBB.pl?num=1237714550/11> and reported by Jaime Castro.
- 1.4 2010-01-29 Added vx\_tx\_active(), suggested by Alan Burlison.
- 1.5 2011-09-09 Added vx\_tx\_active() function.
- 1.6 2012-01-10 Fixed a problem where the receiver was always reenabled after transmission. Reported by David Bath
- 1.9 2012-02-07 Documentation updates Documentation updates
- 1.10 Updated CHANGES file with changes since 1.4.
- 1.11 Converted documentation to Doxygen. Moved CHANGES log to this version history. Ensure vw\_rx\_pin is not accessed unless receiver is

enabled

- 1.12 Compiles and runs on on Energia with MSP430G2553 / G2452 and Arduino with ATmega328. Patches contributed by Yannick DEVOS - XV4Y
- 1.13 util/crc16.h needed for compiling on Energia with MSP430G2553 / G2452 was accidentally left out of the distribution
- 1.14 Added support ATtiny85 on Arduino, patch provided by r4z0r7o3.
- 1.15 Updated author and distribution location details to airspayce.com
- 1.16 Added support for Teensy 3.0, contributed by Paul Stoffregen.
- 1.17 Increase default MAX\_MESSAGE\_LENGTH to 80. Added `vw_get_rx_good()` and `vw_get_rx_bad()` functions.
- 1.18 Added support for Maple, Flymaple etc with STM32F103RE processor using timer 1. Tested with Flymaple 1.1 and maple-ide-v0.0.12
- 1.19 Added new function `vw_rx_inverted()`, to allow the incoming RX to be inverted (normally high). Minor improvements to timer setup for Maple. Name `vw_tx_active()` changed from incorrect `vx_tx_active()`
- 1.20 Added support for ATtiny84, patched by Chuck Benedict.
- 1.21 Added support for most AVR8 platforms with proper configuration, without depending on Arduino environment, such as Atmega32u2, Atmega32U4, At90USB162 etc, contributed by Alexandru Daniel Mircescu.
- 1.22 Alexandru Daniel Mircescu fixed some problems with the recently added AtMega32U2 support.
- 1.23 Fixed some errors and inaccuracies in documentation, with the kind assistance of Alexandru Mircescu.
- 1.24 Minor performance improvement to `vw_symbol_6to4()` courtesy Ralph Doncaster.
- 1.25 Removed all use of floating point code, with assistance of a patch from Jim Remington. Result is smaller code size.
- 1.26 Removed util/crc16.h from distribution, since it is now included in arduino IDE since version 1.0. Support for arduino IDE prior to 1.0 is now abandoned.
- 1.27 Reinstated VWutil/crc16.h for the benefit of other platforms such as Teensy. Testing on Teensy 3.1. Added End Of Life notice. This library will no longer be maintained and updated: use RadioHead instead.

#### Implementation Details

See: <http://www.airspayce.com/mikem/arduino/VirtualWire.pdf>

#### Performance

See: <http://www.airspayce.com/mikem/arduino/VirtualWire.pdf>

#### Connections

See: <http://www.airspayce.com/mikem/arduino/VirtualWire.pdf>