

ANALYSIR **Getting Started**

User Guide

Version 0.914.3101.60

User Guide

WARRANTY

While every effort has been made by ANALYSIR to ensure the accuracy of the information contained in this document, we make no warranty of any kind, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose.

ANALYSIR shall not be liable for errors contained herein, or for incidental or consequential damages in connection with the furnishing, performance or use of this material. This publication and features described herein are subject to change without notice.

COPYRIGHT

No part of this publication may be reproduced in any form, without prior written consent of ANALYSIR and shall only be used for the purpose of this publication.

Copyright ANALYSIR © 2012-2014. All rights reserved.

TRADEMARKS

All trademarks, service marks or product names included in this document are recognised as the property of their respective owners.

Table of Contents

CHAPTER 1	5
1. INTRODUCTION	5
OVERVIEW	5
SYSTEM REQUIREMENTS.....	5
ABOUT THIS GUIDE.....	6
ABOUT ANALYSIR	6
CHAPTER 2	7
2. GETTING STARTED	7
INSTALLING ANALYSIR.....	7
OBTAINING YOUR LICENCE KEY.....	7
DEMO MODE.....	8
CONFIGURATION	8
INITIAL SETUP.....	9
NETWORK SETTINGS.....	9
CHAPTER 3	10
3. USING ANALYSIR	10
ANALYSING IR SIGNALS WITH ANALYSIR	11
DUAL CHANNEL CONCEPT	11
CHANNEL 1 & 2 TRACES	11
CHANNEL 1 & 2 SETTINGS	12
CHANNEL 1 & 2 TAB.....	14
RULES TAB.....	15
LOG TAB.....	15
SESSION HISTORY	15
SAVE SCREENSHOT.....	17
EXPLORE	17
PROPERTIES VIEW.....	17
CLEAR BOTH.....	17
CLEAR ALL	17
SERIAL PORT	17
IMPORT.....	18
EXPORT	19
BATCH EXPORT.....	19
PROTOCOL MENU	20
PULSE ANALYTICS	20
SERIAL PORT RESET	21
EXIT.....	21
POWER TOOLS MENU.....	22
HISTORY DECODE	22
SEND IR	22
SEND SELECTED IR	23
SOURCE.....	23
SHOW MODULATION FREQUENCY	25
HELP MENU.....	25
SERIAL PORT	25
CUSTOM IR PROTOCOLS	26
CHAPTER 4	27

4. ARDUINO SETUP	27
CIRCUIT DIAGRAM	27
CHAPTER 5	30
5. RASPBERRY PI SETUP	30
CIRCUIT DIAGRAM	30
CHAPTER 6	33
6. USB IR TOY SETUP	33
CHAPTER 7	33
7. TI LAUNCHPAD TOY SETUP	33
CHAPTER 8	34
8. CONTACTING SUPPORT	34
CHAPTER 9	35
9. APPENDIX A – SAMPLE ARDUINO & RASPBERRY PI CODE	35

1. Introduction

This chapter introduces ANALYSIR. It includes the following sections:

- ❖ *"Overview"*
- ❖ *"System Requirements"*
- ❖ *"About this Guide"*
- ❖ *"About ANALYSIR"*

OVERVIEW

ANALYSIR provides a Windows desktop application to analyse and decode Infra-red (IR) signals recorded via devices such as Arduino, Raspberry Pi, USB IR Toy, Ti LaunchPads etc.

It is also possible to operate ANALYSIR in 'Demo' mode whereby many of the features can be evaluated before purchasing. In this mode the connection to the recording device cannot be used. Instead it is possible to load included sample session histories, for evaluation purposes only.

SYSTEM REQUIREMENTS

You must have the following software installed to use ANALYSIR via the desktop:

- Microsoft Windows PC, with latest updates (including .NET)
- Arduino compatible running at 16MHz (although slower CPU speeds may work) with USB serial interface (Raspberry Pi, USB IR Toy or other MCUs are also possible), plus associated IDE or equivalent.
- Sample code is available for the Arduino IDE and Raspberry Pi. The latest USB IR Toy firmware is available from the vendor's web site.
- A copy of the Licence text and a Valid Licence Key are required at all times.
- An IR emitter such as a TV remote control to generate the signals to be analysed.

ABOUT THIS GUIDE

This guide provides information on the following:

- ❖ *"Installing ANALYSIR"*
- ❖ *"Obtaining your Licence Key"*
- ❖ *"Demo Mode"*
- ❖ *"Configuration"*
- ❖ *"Arduino Setup"*
- ❖ *"Using AnalysIR"*
- ❖ *"Arduino setup"*
- ❖ *"Raspberry PI setup"*
- ❖ *"Contacting Support"*
- ❖ *"Sample Code for Arduino & RPi"*

This user guide is intended as a priming guide for users who are new to ANALYSIR, and assumes familiarity with Windows PC applications together with Arduino or similar MCU environments. Some knowledge of IR protocols and digital/binary systems is desirable.

ABOUT ANALYSIR

ANALYSIR is designed to be a comprehensive IR analysis & decoding tool and is targeted at Electronics Engineers, Makers & Hobbyists, Educational Institutions and Maintenance & Repair Engineers.

ANALYSIR is also available for bundling along with 3rd party kits and products. Contact your ANALYSIR representative for further details.

ANALYSIR is a software only product and requires additional 3rd Party equipment such as a Windows PC and Arduino to use the full set of functionality. **It is not intended for use in any situation where loss of life, injury or damage to property could occur.**

AnalysIR supports and can analyse all IR remote control protocols. However, the IR receiver can limit the range of signals available. Our recommended IR Receiver/Learner can support a modulation frequency range from 20 – 60 kHz.

The list of IR protocols decoded by AnalysIR includes:

NEC, JVC, RC6, RC5, SANYO, PANASONIC, DISH, SHARP, XSAT, SONY, KONKA, RECS80, RCA, PIONEER, DENON, DAEWOO, SAMSUNG, LG AC, DAIKIN AC, HELISH3CH – with several of these decoding multiple variants of the base protocol.

2. Getting Started

This chapter describes the initial setup steps and includes the following sections:

- ❖ *"Installing ANALYSIR"*
- ❖ *"Obtaining your Licence Key"*
- ❖ *"Demo Mode"*
- ❖ *"Configuration"*
- ❖ *"Initial Setup"*
- ❖ *"Network Settings"*

INSTALLING ANALYSIR

To install ANALYSIR you must first download the installation file from the website. Once downloaded you must run the installer. During installation you will be provided with normal installation options and will be given an option to select a non-standard install location. However, to complete the installation you must agree to and accept the End User Licence agreement (EULA) for any mode of operation (Demo or Licensed).

Once installed you may proceed in demo mode. To use all features you must obtain a valid licence key.

OBTAINING YOUR LICENCE KEY

New users of the system may obtain their Licence Key credentials as follows:

- Once you install ANALYSIR you will be presented with a popup window looking for your email, and Licence Key.
- Copy the signature field and email it to us along with the email address you want to register with the Licence Key. The registration email address will be provided at time of purchase. Your registration email should be in the following format – [x@y.z](#) . It is a condition of the licence that the licence Key will only be issued to the registered email address.
- As the licence key is always tied to the same machine and email address, you will need to have access to both for updates and support. If you received a discounted Licence (

e.g. educational or hobby club), the email address will require an identifiable domain name such as mynameorID@collegeX.xxx)

- Once you receive an email response with your unique Licence Key, you should start the application again and enter the key and your registered email into the relevant fields in the popup window.
- Then click on the 'Install Licence Key Now' button to proceed.
- While you are waiting to receive your key, you may try out and become familiar with the application in Demo Mode.

DEMO MODE

If you don't have a valid Licence Key, you may use ANALYSIR in Demo Mode, which allows you to evaluate all of the features without interfacing with the (Arduino) Recording device.

To access Demo Mode enter your valid eMail address, click to accept your agreement with the EULA and then click the 'Continue in Demo Mode' button on the initial pop window. This feature is not available if a valid Licence Key has been entered previously.

Once you enter Demo mode you have access to all features except sending & receiving live IR signals from the recording device. Importing of 3rd party IR signals is partially limited. In order to allow you evaluate the system, we have provided a selection of sample recorded sessions, which can be loaded via the File menu (Load Saved Session) option.

When using AnalysIR in 'Demo Mode' you will be redirected to the AnalysIR website at various stages, which will allow you the opportunity to upgrade to a full version.

CONFIGURATION

There is no additional configuration required to operate the system in default mode. However, you should make sure to select the correct port for your recording device (Arduino) in the combo box at the bottom of the screen. The default is set to 'COM3', but is likely to be different on most systems.

In your configuration file (ANALYSIR.ini) which is located in your 'user application directory, you can manually adjust many of the pre-set configurations for your installation. We recommend that you do not alter any of these settings unless requested to do so or unless you consider yourself to be an expert user. As always it is advisable to make a backup copy of files before changing them.

The location of your Data directory and other properties can be found via Menu ⇒ File ⇒ View Properties. You can also navigate to your data directory via the menu: Menu ⇒ File ⇒ Explore.

Detailed instructions for installing the Arduino, RPi, USB IR Toy and LaunchPad firmware are contained in this document or in standalone documentation provided in your installation package.

INITIAL SETUP

Follow the instructions contained within this Guide, the 'README' files provided for the Arduino, Raspberry Pi & other platforms, the comments within the source code provided and also the selection of Getting Started Tips provided alongside your installation package.

Always ensure that you refer to the latest versions of ANALYSIR and associated documentation before contacting support. You will also find a community discussion forum via <http://www.analysIR.com/> .

It is assumed that you are familiar with the Arduino IDE and loading new software onto your Arduino or any other supported platform. For the RPi you will need to be familiar with compiling, running shell scripts and firewall configuration. However, many people will be able to install the RPi version by simply following the instructions available.

If you use a platform other than Arduino or RPi, it should be relatively simple to alter the code to suit your own platform. The important thing to remember is that you transmit the same data over the USB/serial connection to the PC at 2,000,000 bps (or 115,200 if TI LaunchPad is selected as the IR Source).

Please check on the ANALYSIR IRforum for assistance.

NETWORK SETTINGS

Some IR sources may be connected via LAN (Ethernet or Wi-Fi) instead of using a COM port on the PC. In order for this to work the 'Source' must be configured to work with ANALYSIR (See specific instructions for each platform). In addition, ANALYSIR must be informed about the network IP address and TCP/IP port number to connect to, on the 'Source' device. This is achieved by manually entering this information in the ANALYSIR config.ini file located in the APPDATA directory (Menu->File->Explore).

This file should already contain entries for the supported 'Source' device, as follows:

```
[Network]
IPAddress_RPi=000.000.000.000 #Change this to IP address of your RPi
Port_RPi=25 #Change this to Port number used with LIRC & socat on your RPi
#The following is for future reference only, pending support of the Arduino Yún
IPAddress_Yun=000.000.000.000 #Change this to IP address of your Yún
Port_Yun=25
```

Simply change & save the IP address and port number for your 'Source' device in the 'ini' file, restart ANALYSIR and select that source from the 'Source' menu (Menu->Source->{.....})

Currently, only the Raspberry Pi is supported for network connection, with the Arduino Yún to follow.

Note: Connection of RPi via COM port is no longer supported (but may work – if you select Arduino as source)

3. Using AnalysIR

This chapter describes the various features of the system and how to use them. It includes the following sections:

- ❖ *"Analysing IR signals with ANALYSIR"*
- ❖ *"Dual Channel Concept"*
- ❖ *"Channel 1 & 2 Traces"*
- ❖ *"Channel 1 & 2 Settings"*
- ❖ *"Channel 1 & 2 Tab"*
- ❖ *"Rules Tab"*
- ❖ *"Log Tab"*
- ❖ *"Session History"*
- ❖ *"Save Screenshot"*
- ❖ *"Explore"*
- ❖ *"Properties View"*
- ❖ *"Clear Both"*
- ❖ *"Clear All"*
- ❖ *"Serial Port"*
- ❖ *"Import"*
- ❖ *"Custom IR Protocols"*
- ❖ *"Export"*
- ❖ *"Batch Export"*
- ❖ *"Protocol Menu"*
- ❖ *"Pulse Analytics"*
- ❖ *"Serial Port Reset"*
- ❖ *"Send IR"*
- ❖ *"Send Selected IR"*
- ❖ *"Source"*
- ❖ *"Show Modulation Frequency"*
- ❖ *"Exit"*
- ❖ *"Help Menu"*
- ❖ *"Serial Port"*
- ❖ *"Arduino & Raspberry PI setup"*

Before proceeding, please ensure you have installed the system correctly as outlined in the previous chapter and getting started instructions.

ANALYSING IR SIGNALS WITH ANALYSIR

Once your system is installed and configured correctly it is relatively easy to start analysing IR signals. Simply point your IR device (e.g. TV remote control) at the IR receiver attached to your Arduino and press any key. You should then see the signal appear on the PC screen. The signal will be drawn on the trace of the selected Channel (1 or 2). A channel is selected or activated by clicking the Channel 1 or 2 Tab. The currently active or selected channel is always indicated in the status bar at the bottom of the window.

DUAL CHANNEL CONCEPT

ANALYSIR supports 2 'virtual' channels. This means that the Arduino can receive one signal at a time (on one physical channel). However, we have designed ANALYSIR with 2 independent traces or channels. This allows users to record signals on 2 separate traces and compare and contrast them. We found this to be a very useful tool in analysing, comparing and decoding unknown or problem IR signals.

CHANNEL 1 & 2 TRACES

Each Channel trace is made up of 3 areas. The first contains the main signal plot where the IR signal itself is plotted as a 'Square Wave'. As the mouse is moved over the plot area a tooltip is displayed showing the time offset in micro-seconds, the duration and sequence number of any mark or space and the value of the last measurement on that channel in microseconds.

It is also possible to measure the time between 2 points on the trace of either channel by clicking the mouse and then moving the mouse to another position and clicking again. Alternatively, you may drag the mouse. The time between both points will be displayed in the Measure text field for that Channel and the tooltip. In addition, a visualisation of the measurement is drawn on the trace after every click, in the form of a box and the tooltip also displays the value of the last measurement for each channel. This visualisation can be hidden by unchecking the check-box to the left of the 'Measure' control located in either channel panel or by clicking the mouse in the same spot on the signal trace a couple of seconds apart (not a double-click).

The timescale cursor is hidden when this visualisation is displayed (also see following paragraphs).

As you hover the mouse over marks and spaces of the signal, the related row will be highlighted in the Channel 1 or 2 Tab. This allows you 'at-a-glance' to see the duration of that Mark or Space, which can simplify the task of understanding a particular signal.

The second part of the trace is the timescale. The timescale is divided into 1,000 microseconds (minor-ticks) and 10,000 microseconds (major-ticks) units. As you hover over the timescale with your mouse a vertical line (or Cursor) is moved

along the trace for each channel, when enabled (see cursor below). Enabling the cursor hides the measurement visualisation described above.

When a signal is automatically decoded by ANALYSIR a colour coded representation is also displayed in the timeline showing 'Header' & 'Trailer' bits, 'One's and 'Zero' bits. This enhances the user's ability to understand the make-up of a particular signal and protocol. This is only provided for successfully decoded signals.

An image of both traces may be saved to your 'APPDATA' directory via:

Menu ⇒ File ⇒ Save Trace Image

Menu ⇒ Channel 1 ⇒ Save Image

Menu ⇒ Channel 2 ⇒ Save Image

The first option saves the visible portion of both channels and the other 2 options save a copy of the full signal trace for each channel (both visible and not visible)

CHANNEL 1 & 2 SETTINGS

There are 2 Settings panels to be found in the centre of the lower part of the main window (Channel Panel), where various features can be configured for each Channel independently, as follows:

- **Overlay** – when checked the next received or drawn IR signal on this channel will be drawn directly over the existing signal, in a different colour. This continues until the setting is unchecked. (Menu ⇒ Channelx ⇒ Overlay).
Note: if you overlay multiple signals the trace will be become difficult to read.
- **Invert** – When checked the signal plot is inverted. This can be useful when visually comparing signals between Channel 1 & 2 or if the signal was otherwise inverted.
(Menu ⇒ Channelx ⇒ Invert).
- **Lock** – when checked, new signals are ignored on this Channel. This is useful if you want to retain a signal in the display, when you cannot control when IR signals are transmitted in the room or area. Whenever a signal is received on a locked channel the beep is played, if enabled. This can serve as an audible reminder that signals are being received, in case you forget that the channel is locked.
(Menu ⇒ Channelx ⇒ Lock/Unlock)
Another way to achieve a similar outcome is to copy a signal to the other (inactive) Channel.
- **Beep**– When checked, a beep is sounded on the PC whenever an IR signal is received or analysed. When cleared the beep is muted. If you cannot hear the beep, make sure your sound is not muted and the volume is raised. The default value when the application starts is muted and if you un-mute it, the value will not be retained the next time you open the application.
(Menu ⇒ Channelx ⇒ Beep)
- **Discrete** – When checked ANALYSIR treats all received IR signals in the same sequence as individual IR sequences.

Normally, when you press a button on a remote, several signals can be sent quickly as part of one sequence and are typically treated as if just one key was pressed. When 'Discrete' is checked ANALYSIR will attempt to decode each part of the sequence as a discrete signal and split them up into individual signals in the session history. Similarly, if your remote is programmed to send a 'Macro' with a sequence of different commands, using the 'Discrete' mode will help in understanding the various components of the overall sequence.

(Menu ⇒ Channelx ⇒ Discrete)

- **Colour** – You can click on this box to change the colour of the signal plot for new signals.
(Menu ⇒ Channelx ⇒ Change Colour)
- **Scale** – The scale can be adjusted to zoom in and out of the signal plot. The default value is 10 and the range goes from 1 to 100. You can reset the scale via the Menu
(Menu ⇒ Channelx ⇒ Reset Scale) or by double-clicking on the 'Scale' label to the left of the scale control.
Another method for zooming in and out of a signal is to use the mouse wheel when the mouse is over the channel settings panel. If your PC does not have a mouse with a wheel attached or a mouse-pad with wheel emulation, you can still do similar zooming with the up-arrow and down-arrow keys, provided the scale control has focus. Remember to place the cursor inside the Scale box for either channel first when using keys. Up-arrow zooms in and down-arrow zooms out.
- **Start** – This is used to start the IR decoding other than from the start Mark/Space. The number selected corresponds to the row number in the Channel 1(2) tab. You can temporarily achieve the same effect by double clicking on the signal trace at any point in the timeline.
- **Measure** – This feature is enabled by clicking the check-box to the left of the measure control or via the menu.
(Menu ⇒ Channelx ⇒ Measure)
When you click in 2 different positions on the signal plot, this field displays the time difference between the 2 positions in microseconds. Dragging the mouse over the trace also adjusts the measurement, which is shown in the visualisation. The measurement value is displayed in the 'Measure' control for each channel. Also see the description above for measurement visualisation.
The measure feature is disabled automatically when the cursor feature is enabled or when the checkbox is cleared.
- **Cursor** – This feature is enabled by clicking the check-box to the left of the cursor control or via the menu.
(Menu ⇒ Channelx ⇒ Cursor)
As you move the mouse along the timescale, the cursor follows and this field displays the time value along the timescale for the cursor position.
The cursor feature is disabled automatically when the measure feature is enabled or when the checkbox is cleared.
- **Tolerance** – Adjusting this slider allows you to change the tolerance level when decoding signals. For example, if you are receiving weak or poor signals changing the tolerance may increase the chances of successfully decoding a signal. Similarly if ANALYSIR confuses 2 different but similar

protocols, decreasing the tolerance may result in less confusion. For most situations, the default setting should suffice. You can also reset the tolerance to the default value via the Menu or by double-clicking the 'Tolerance' label to the left of the tolerance control.

(Menu ⇒ Channelx ⇒ Reset Tolerance)

- **Clear** – Clicking the clear button erases the signal plot and empties the contents of the associated Channel Tab.
(Menu ⇒ Channelx ⇒ Clear)
- **Copy to CH1 (CH2)** – Clicking this button copies the signal plot from one channel to the other. The contents of the Channel Tab are also copied.
(Menu ⇒ Channelx ⇒ Copy to Channely)
- **Analyse** – Clicking this button runs the decoding analysis on the signal again. Normally this is not required. However, if you load a saved Session History and there is a newer version of ANALYSIR installed, it may be possible for you to decode a signal that wasn't supported in an earlier version. Alternatively, you may want to analyse a signal from a different start position, as sometimes multiple signals will be received in the same sequence. Similarly, you may want to analyse a signal using different settings for tolerance. This option is ignored if the currently selected history row does not match the channel or if the history is empty.
(Menu ⇒ Channelx ⇒ Analyse)

Note: if the signal is successfully decoded, the values in the currently selected history row will be over-written. Also see the 'History Decode' feature of the 'Power Tools' for similar functionality. Remember to save the session if you want to keep a record.

- **Copy to Clipboard** – This menu option allows you to copy the contents of the Channel 1 (or 2) grid to the clipboard, which can then be pasted into other applications such as a spreadsheet for further processing.
(Menu⇒ Channelx⇒ Copy to Clipboard or CTL+1, CTRL+2))

Some of the options above are also available via the Menu. 'Channelx' is used to denote 'Channel1' and/or 'Channel2'.

CHANNEL 1 & 2 TAB

These tabs are located on the lower half of the window to the left. Each one contains detailed information about the signal displayed in the plot, as follows:

- **Seq** – this is the sequence or row number as received from the Arduino (or other MCU). Typically this is numbered from 1 up.
- **Time** – this represents the time offset for each mark or space in the signal when measured from the first space, measured in microseconds.
- **Duration** – This represents the duration of the current mark or space of the signal, measured in microseconds.
- **State** – The state can be '0' or '1'. A space is indicated by '0' and a mark is indicated by '1'

All of these combined allow ANALYSIR to decode a signal, based on a pre-configured rules-engine for each supported protocol.

RULES TAB

The rules tab is located beside the Channel 1 & 2 Tabs and provides details of the various rules used by ANALYSIR to decode supported IR protocols.

When a signal is successfully decoded the rules for that protocol are selected and displayed. Alternatively, you may view the rules for any other protocol by selecting it from the combo box control.

IR protocols can be described in terms of Marks & Spaces, Trailers, Delta (in the case of ANALYSIR this signifies the tolerance of the decoding algorithm). (e.g. when detecting a Mark of duration 1,000 microseconds and a delta of 100 – a valid Mark will be read if the duration is between 900 (1,000-100) and 1,100(1,000+100).

Note: you can also use the tolerance setting to increase or decrease the value of the delta.

'Altname' is used when the 'official' name of the protocol is uncertain or if it can also be known by other name(s). If you are familiar with decoding IR protocols all of the other values should be self-explanatory.

The default values for any protocol can be over-written in the configuration file (AnalysIR.ini), which in theory would allow you to 'fool' the system into decoding other protocols as long as they used a similar encoding scheme. Always make a backup copy before changing any ANALYSIR files.

LOG TAB

When selected, a log of all the recent serial communication messages received from the Arduino is displayed. In addition some debug information from ANALYSIR is also displayed.

If you want to replace the Arduino with another MCU, you can use this information to see and debug the information received from the device. Alternatively, you can copy the contents and carry out your own calculations on the raw data. When zooming and clicking the analyse button you will see additional information being written to this log. Once the size of this log goes beyond a pre-set threshold, it is truncated.

You may clear the on-screen log via:

(Menu ⇒ File ⇒ Clear On-screen Log or CTRL+0)

SESSION HISTORY

The session history is displayed in the middle of the lower part of the display. One row is recorded for each signal stream received, as follows:

CH – Displays which channel the signal was originally recorded

Time – The local timestamp of when the signal was recorded

Type – The name of decoded Protocol. The type will be underlined when a toggle bit is set. If it is not decoded then 'RAW' will be displayed here.

Key – A user editable field to record the identification for the signal (e.g. 1, 2, Vol+, Menu, Power etc.). You may find this useful if you want to record the codes associated with all keys on your remotes controls. You can then save all the keys in one session history file and then later rename the history file with the ID of the remote. The auto complete feature helps in rapid entry of keys. For example if you type lower case 'p' you will be prompted with options like Power and Play etc. Pressing enter will select the default or you can select an alternative with the mouse or arrow keys (Up/Down). If you enter a new key label, the system will include this new label for auto-complete during the current session only. The auto-complete feature will automatically correct any detected capitalisation mistakes.

Value – a Hexadecimal value of the decoded signal. For values smaller than 64 bits this represents the full value. Otherwise, it contains bits 65 to 128.

Value2 – for some protocols a second value is also recorded. For values smaller than 64 bits the value will be zero. Otherwise it will contain bits 1 to 64.

Bits – the number of data bits in the decoded signal

Date – the date when the signal was recorded

Data – The raw data for the IR signal as contained in the Channel tab with row columns separated by ';' and rows separated by '|'.

Toggle – set to 'T' when the toggle bit is set. Note that not all protocols use a toggle bit.

Session histories can be saved to your user data or APPDATA directory (Menu ⇒ File ⇒ Save Session or CTL+S). Additionally, Saved sessions can be reloaded for later review (Menu ⇒ File ⇒ Load Saved Session or CTL+L). The History display can be cleared (Menu ⇒ File ⇒ Clear History or CTRL+H). You can rename any history file using the file system, provided it ends with ...'History.txt'

When saving session history, the user is prompted for a string which describes the session. When this session is re-opened later the description is displayed in the title of the History panel. The session description may be left empty. Clicking 'Cancel' saves the Session History, without a description.

Clicking on the 'Value' or 'Value2' cells will display a pop-up window with a breakdown of both values in both Hex and Binary formats. In addition, the contents of the pop-up window are copied to the clipboard and can be pasted into other applications for further processing.

SAVE SCREENSHOT

This menu option allows you to save an image of the main window to your APPDATA directory. This can be of great assistance when seeking support or for publishing examples of your work. You can locate the saved file via the 'Explore' option below.

Menu ⇒ File ⇒ Save Screenshot or CTL+7

EXPLORE

This menu option allows you to open your ANALYSIR APPDATA folder in 'Windows Explorer'. This provides an easy shortcut to review your session History, configurations and saved image files. All user related files of AnalysIR are contained in this directory.

Menu ⇒ File ⇒ Explore or CTL+F

PROPERTIES VIEW

You may view the values of most properties in ANALYSIR via this menu option, in read only mode (Menu ⇒ File ⇒ View Properties). A window is displayed showing a list of all properties and their value.

Columns can be resized by dragging the dividers of the column headers. The contents can be copied to the clipboard by selecting one or more cells and pressing CTRL+C.

CLEAR BOTH

Selecting this option from the menu (Menu ⇒ File ⇒ Clear Both or CTRL+B) clears both channel traces & data. Each channel can be cleared individually, via their own menu options or 'Clear' button.

CLEAR ALL

Selecting this option from the file menu (Menu ⇒ File ⇒ Clear All or ALT+A) clears both channel traces & data, the History and the on-screen log in one simple step. All of the above can be cleared individually, via their own menu options.

SERIAL PORT

The serial port can be selected by selecting an available PORT from the combo box in the status bar. If no port is selected ANALYSIR will attempt to open the default port COM3. Please check and confirm which port your Arduino (or other recording device) is assigned to before starting ANALYSIR.

If you change the port, your selection will be remembered the next time you start the application. COM port details are stored in the configuration (ANALYSIR.ini) file, which is located in the application data directory.

The default settings for the serial port are as follows:

Baud: 2,000,000
Data Bits: 8
Start Bits: 1
Stop Bits: 1
Parity: none

These settings match the default settings on Arduino and many other MCUs except for the Baud rate. The baud rate is set high to improve the responsiveness of the system when recording. If you select a different 'Source' alternative defaults will be applied automatically.

IMPORT

You may import IR signals into ANALYSIR in a selection of formats including: IRremote, IRLib, Global Cache (GC-100, iTach, GC_IRL, GC-IRE), Pronto, Command Fusion and Saleae Logic CSV. We plan to add more formats in future.

Menu ⇒ File ⇒ Import IR Signal or CTL+I

Once the new window opens, the normal sequence is to paste the signal data into the large text box on the left. You can then select the Source format from the drop down box (or just leave it at the default of 'Auto Detect') and click the 'Import into Channel' button for channel 1 or 2. If you are unsure if the format is correct, you can click the '>>>>' button first to get a preview of the signal data to be imported. If you select the 'Invert' check box the signal will be inverted during import and before decoding takes place. Clicking the '<< Clear' button erases the content of the text box and assigns focus to the text box (making it easier to clear and paste in a quick sequence). Finally, you may adjust the tolerance level for importing the signal. This operates similarly to the tolerance control on the channel panels of the main window.

The following formats are supported for import:

- Auto Detect (can automatically detect any of the following)
- IRremote
- IRLib
- Pronto CCF
- Global Caché GC-100
- Global Caché iTach
- Global Caché GC-IRL
- Global Caché GC-IRE
- Command Fusion
- Saleae Logic CSV

If you are unsure of the above formats, just export any signal in the format required. (See Export feature below) For Saleae Logic, use the export feature with the following settings:

- *Select only 1 channel*

- *Samples to export = 'All Time'*
- *Export Format = 'CSV'*
- *Include column headings*
- *Comma delimited*
- *Use timestamps [s]*
- *Output one column for every bit*
- *Output one row per change*

EXPORT

You may export IR signals from ANALYSIR in a selection of formats including: IRremote, IRLib, Global Cache (GC-100, iTach, GC_IRL, GC-IRE), Pronto and Command Fusion. We plan to add more formats in future.

Menu ⇒ File ⇒ Export IR Signal or CTL+E

Once the new window opens, the normal sequence is to click the 'Generate from Channel' button for either channel 1 or 2. ANALYSIR then exports the signal from the selected channel into the text box. You can take this export data and copy/paste it into other applications. You must select the required Export format from the drop down box.

Always select the correct modulation frequency for Global Cache and Pronto formats. For Global Cache formats you may also select compression, which will produce a much smaller set of data. 'Auto Clean' analyses the signal and cleans degraded signals which are noisy to their correct values using an ANALYSIR internal algorithm. This means that you can record and import noisy or degraded signals and then export them in perfect format. Remember it is also possible to re-import these perfect signals back into ANALYSIR for comparison against the original signal. This can be useful in investigating the causes of degraded IR signals.

Auto Clean works with successfully decoded signals only.

Check the 'Auto Copy to Clipboard' box to have ANALYSIR automatically place the exported data into your clipboard in addition to the text box.

A character count is also provided below the text box, which can be useful for some data formats with size restrictions.

BATCH EXPORT

This feature allows you to export all of the signals in the displayed Session History to the clipboard for import directly into IRremote or IRLib Arduino sketches. The export data contains prepared 'c' code instructions for the selected library, including useful comments. Also included is a list of keys, decoded IR codes, #bits which can be pasted into a spreadsheet or other system elsewhere for further processing. (Alt+B)

Menu ⇒ File ⇒ Batch Export ⇒ IRremote

Menu ⇒ File ⇒ Batch Export ⇒ IRLib

The following formats are supported for export:

- IRLib
- IRremote
- Global Caché GC-100
- Global Caché GC-IRE
- Global Caché GC-IRL
- Global Caché iTach
- Command Fusion
- Pronto CCF

Note: not all protocols available with AnalysIR will be supported by the standard IRremote/IRLib libraries. If not, search for existing customised implementations for the protocol you require, which may be available.

PROTOCOL MENU

You may enable one or more protocols for decoding by selecting the related IR Protocol name from the Protocol Menu.

Menu ⇒ Protocol ⇒

The menu also provides the ability to select 'All' protocols, 'None' or to 'Toggle' the list of protocols selected. If a protocol is selected, ANALYSIR will include that protocol in the list of protocols it attempts to decode. Otherwise ANALYSIR will not attempt to decode that protocol. Disabling a particular protocol is appropriate when you know that it is very rare or specialised or if you always know when the signal type is being decoded.

Menu ⇒ Protocol ⇒ All

Menu ⇒ Protocol ⇒ None

Menu ⇒ Protocol ⇒ Toggle

By Default, less common IR Protocols are not enabled.

You may also disable automatic decoding for new IR signals by un-checking the 'Decoding' menu option. When unchecked, ANALYSIR will not attempt to decode IR signals received. This option also impacts on the 'History Decode' feature in the 'Power Tools' Menu. To re-enable just check the option again. Use 'ALT+D' to toggle this option from the keyboard. This option can be useful when signals are being decoded incorrectly or when you want to force the signal into RAW mode for export using the 'Batch export' facility of the File menu.

Menu ⇒ Protocol ⇒ Decoding

PULSE ANALYTICS

You may visualise analytics on a signal via:

Menu ⇒ File ⇒ Pulse Analytics or CTL+9

Once the new window opens, click either the Channel 1 or 2 buttons to view an analysis of the signal in either channel. You may adjust the granularity in 10 microsecond steps, which groups the count of marks and spaces in ranges set by the granularity value (default is every 50 microseconds).

You may save a copy of the chart image to your data directory by clicking the 'Save chart Image' button

You may also view analytics by clicking the chart icon on the channel panel of the main window.

SERIAL PORT RESET

You may manually reset the Serial Port from the menu:

Menu ⇒ File ⇒ Serial Port Reset or ALT+S

ANALYSIR will then attempt to close and reopen the serial port selected in the dropdown at the bottom right of the screen. On success the indicator box will turn green and it will turn red on failure. The best way to reset the Serial Port is to re-select the required COM port from the dropdown at the bottom right of the main window. If that fails then try the following:

- a) Press the reset button on the device (e.g. Arduino).
- b) Use this menu option.
- c) If the problem is not resolved then disconnect your USB Serial cable and wait 10 seconds before re-connecting it.

This feature is provided in an attempt to overcome some issues with some Arduino models not being recognised properly on some Windows platforms.

ANALYSIR checks the Serial port every few seconds and attempts to automatically reset any problems that are detected. It is important that you have the correct COM port selected at all times.

Note: in order to avoid problems with Windows COM ports it is recommended to close applications before physically disconnecting serial USB cables, to avoid causing occasional issues with device recognition.

EXIT

You may close and exit the application from the main window:

Menu ⇒ File ⇒ Exit or ALT+F4

Alternatively you may click the 'X' box in the top right of the main window.

On exit, you will be prompted to confirm your request (OK) or cancel the request (Cancel). Pressing the 'ESC' key cancels the request and returns to the Main Window.

On all Windows, pressing the 'ESC' key will close that window or in the case of the main window, prompt you as described above.

On resume from Standby or Hibernation, ANALYSIR will automatically try to reconnect to the serial port being used before

the PC was suspended, after a short delay of 15 to 20 seconds. In some rare cases, Windows may assign a different COM port number to the recording device. If this happens, just manually select the new COM port from the drop down box at the lower right of the Main Window.

POWER TOOLS MENU

You may access the 'Power Tools' via the main menu as follows:

Menu ⇒ Power Tools

You will then be presented with a menu list of power tools available on your system. If one or more items are disabled or greyed-out, then these options are not included in your package. You may be able to upgrade at additional cost, by visiting www.AnalysIR.com.

Select the power tool option you require from the menu as described below.

HISTORY DECODE

This feature allows you to 're-analyze' every signal in the on-screen History and any changes in decoding results will be recorded. This can be of great benefit when lots of new signals are imported, you want to adjust the parameters for decoding or if support for new protocols is added to AnalysIR in future. You may access the 'History Decode' feature via the Power Tools menu as follows: (Alt+H)

Menu ⇒ Power Tools => History Decode

You will then see what appears to be an animated playback of every signal in the history, where the results of decoding are overwritten into the history. You may then save this history in a new history file.

SEND IR

This feature allows you to send one or more IR signals from the history using supported devices. Note: not every connected device will have the ability to send IR or may not be supported for sending.

This feature provides the ability to select one or more signals from the onscreen History and send them in sequence with optional delays between each signal.

Let's say you want to send 2 signals in a sequence with a 45 millisecond gap in between. First select the signals by selecting the 'History Row' number of the signals in the slots for Signal 1 & 2. Then enter 45000 into the 'Delay' text box after Signal 1. Make sure that the history row for all other Signals 3-6 display 'none'. Next click the 'Send All' button to transmit your IR sequence.

If you just want to send a single IR signal, then just select the History row in any of the six available slots and just click the associated button (Send 1 for Signal 1, Send 2 for Signal 2 etc.) to send that IR signal from the on-screen History.

You may access the 'Send IR' feature via the Power Tools menu as follows: (Alt+I)

Menu ⇨ Power Tools => Send IR

The 'Reset' button sets all Signals and delays to their default values. Use this after adding more signals to the on-screen history of the main window.

A powerful feature is the 're-Send Last Signal' button. When you click this button the last signal is re-sent. However, this feature uses the contents of the signal timings text box (initially generated from the History signals), instead of the current history contents. This means that you can now make manual adjustments to the timings of the signal, which can be useful when trying to decode or test out new or undocumented signals. Please note that the signal must begin with a Mark (+) and be followed by a space (-) and so on. If you want to manually paste a signal in, make sure to include the +'s and -'s in the correct order. Including the order mentioned above you only need to ensure that the values are comma separated and that there are no non-numeric characters included (other than commas, +'s, -'s and spaces. The following is an example of a valid format

+1000, -500, +600, -1200,.....,-600,+600

Values less than 10 (decimal) are ignored, although typical IR marks/spaces are rarely less than 100 microseconds in duration.

Note: The full recorded signal in each selected history row will be transmitted, not just the first part of the signal.

SEND SELECTED IR

This feature allows you to send one IR signal from the currently selected history row using supported devices. Note: not every connected device will have the ability to send IR or may not be supported for sending.

You may access the 'Send IR' feature via the Power Tools menu as follows: (CTL+Alt+I)

Menu ⇨ Power Tools => Send Selected IR

This is a versatile feature which allows you to check or confirm the validity of any previously recorded or imported signal.

Note: The full recorded signal in the selected history row will be transmitted, not just the first part of the signal.

SOURCE

The 'Source' menu allows you to select the IR input source for AnalysIR

You may access the 'Source' menu via the Power Tools menu as follows:

Menu ⇨ Source => Arduino (Default)

Menu ⇒ Source => Raspberry Pi

Menu ⇒ Source => MSP430 LaunchPad

Menu ⇒ Source => USB IR Toy

Before you can use AnalysIR, you must select the correct 'Source' device. The default is **Arduino** and the source will always be reset to Arduino, whenever a fault is detected with any source device. Always remember to re-select your source device after any fault. The set baud rate for the Arduino is 2,000,000 bps.

The currently selected IR source is always displayed in the taskbar and also checked in the 'Source' menu.

The Raspberry Pi is always connected via the LAN/Wi-Fi connection – as per the installation instructions. (Baud Rate is ignored for RPi as it is not applicable)

The **MSP430 LaunchPad** option supports only the MSP430F5529 LaunchPad at this time and the set BAUD rate is 115,200 bps. Note: The MSP430 option is classed as Beta and supports only the 16MHz clock for now. It also does not support modulation frequency measurement at this time, due to issues with clock/DCO accuracy.

If you have a '**hybrid**' **Arduino**, which does not support the 2,000,000 serial bps rate, then select the MSP430 LaunchPad as your source as it is an identical interface with the only difference being the set BAUD rate. When using an Arduino with the MSP430 as the selected source, modulation frequency measurement is supported.

The **USB IR** Toy is also supported for both receiving and sending IR and requires that firmware version V22 is loaded on to the device for operation with AnalysIR. Once connected, the firmware version is displayed in the taskbar at the bottom of the window and if it displays a firmware value other than V222 or V122 then you need to update the firmware version.

If you experience problems connecting to your source device, this can be due to a number of reasons, including:

- Wrong COM port selected.
Select the correct com port in the taskbar at the bottom of the main window. You can find the COM port in the Device Manager under Ports. Then select the correct source device from the Menu.
- Wrong IP address or Port configured for Raspberry Pi.
Please correct the incorrect configuration. It is possible to test your connection to the RPi by using telnet from the command prompt of your Windows PC. A command similar to "telnet 102.168.1.123 25", if successful should display a burst of characters when pressing your remote control. *(Note the IP address and port number above are just examples for illustration only. You will need to replace them with your own)*
- Issues with windows handling of COM devices.
First click on the Serial Port Reset option (Alt+S) in the File Menu and select the correct COM port in the taskbar and then select the correct Source device from the Source Menu. If you continue to experience problems, it is best to

disconnect your device from the USB port, wait for 20 seconds and reconnect. Finally, repeat the above and if this doesn't solve the problem, you may have to re-start Windows and start again.

- Wrong firmware on USB IR Toy.
Visit the supplier's website and follow the instructions for updating the Firmware. Note: Version 1 hardware (V122) does not have a sensor for measuring modulation frequency.

Remember, always re-select the source device after experiencing problems connecting to the IR toy.

SHOW MODULATION FREQUENCY

The Show Modulation Frequency Menu option is primarily for the USB IR Toy, due to the way it operates. This device must be polled for the modulation frequency after a signal is received. This feature can be accessed via the Menu, as follows:

Menu ⇒ Source ⇒ Show Modulation Frequency

We have provided a standalone script to measure the modulation frequency on the Raspberry Pi (see installation package).

The Modulation Frequency is reported automatically for the Arduino and displayed in the taskbar, provided you have an IR Learner connected to your circuit.

HELP MENU

You may view the About Window via:

Menu ⇒ Help ⇒ About

Or

Menu ⇒ Help ⇒ Licence Agreement

You will then be presented with a window showing the licence agreement, registered email, signature, licence key and ANALYSIR version number.

You may open a copy of this User Guide via:

Menu ⇒ Help ⇒ User Guide or CTRL+U

SERIAL PORT

You may select a new or reset the current serial port by selecting the 'Com#' from the drop down box at the bottom right of the screen. ANALYSIR regularly checks the status of the serial port and changes the colour to red if there is a problem and green if the com port is open. Please ensure you select the correct COM port number in your system. Once you change the COM port ANALYSIR will remember this the next time you start the application.

If you experience problems with your serial device, try disconnecting it from the PC and then reconnect it before waiting for 5 to 10 seconds to see if it has recovered. Also, refer to guidance under Serial Port Reset Menu and Source Menu options above.

Note: to make a connection via the LAN (Ethernet or Wi-Fi) see the section on network settings in the previous chapter.

CUSTOM IR PROTOCOLS

In this release we have added a feature to allow the addition of custom IR Protocols. This was motivated by the wide range of protocols for Air Conditioners (AC). As this initial release is 'beta', please email support for assistance with this feature.

Unfortunately, this facility will not work for RC5 or RC6 style protocols, at present, but will also work on non AC signals. This feature will allow us to add more protocols without having to make a new software release.

4. Arduino Setup

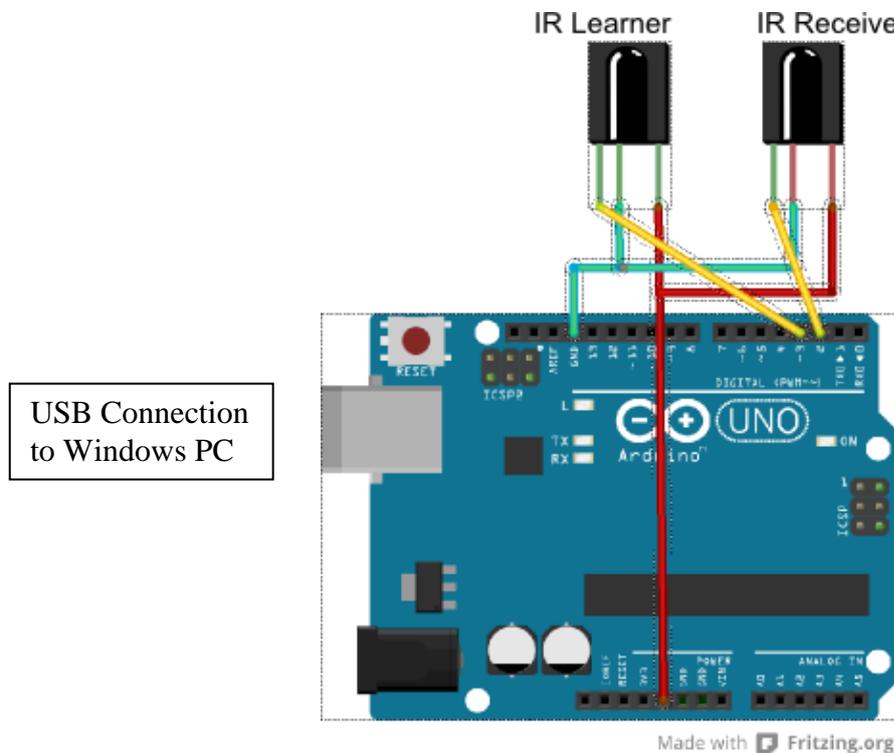
This provides an example circuit layout for the Arduino based IR receiver:

❖ *"Circuit Diagram"*

In addition, we make some suggestions for the IR receiver(s) to use.

CIRCUIT DIAGRAM

The following is a 'Fritzing' circuit layout for connecting the IR receiver and optional IR Learner to the Arduino, which is a very simple circuit.



The circuit above may be used with any Arduino compatible with an ATmega328 class MCU and Leonardo or equivalent with a USB serial connection to a Windows PC running ANALYSIR. If using other models of Arduino it may be necessary to make minor adjustments to the code. If you are using platforms such as Raspberry Pi or other MCUs you will

need to implement code which is similar to that provided. Further instructions for Arduino & Raspberry Pi including code, are available in the installation documents in your data folder. Make sure you read the READMEs **and** the comments within the source code files.

In the circuit above, Green wires are connected to 'GND', Red are connected to '+5V' and the Yellow wire is the Signal output from the IR receiver and connected to Digital Input Pin 2 of the Arduino (INT0). The second Yellow wire is the connection from the optional IR learner to Digital Input Pin 3 of the Arduino (INT1). (Note: The sample Arduino code provided, works across all supported Arduino platforms, including compatibles, such that the IR Receiver is connected to Pin 2 and the IR Learner is connected to Pin 3.

Advanced users may have enough knowledge to change pin assignments, if necessary. For most users the default assignments above will suffice.

Important: The pin-out for different IR receivers varies between models, even from the same manufacturer, so confirm the correct pin-out for your device before applying power.

The IR learners we recommend are the TSMP58000 from Vishay. These pass through the modulated IR signal to the Arduino. The IR receiver, in contrast, passes through the de-modulated signal.

It is not mandatory to have an IR learner component installed to use ANALYSIR, but you will not be able to measure the IR modulation frequency without it.

If you run ANALYSIR just with an IR receiver, you will be able to use every function except those that require the modulation frequency and in most cases it is possible to guess the modulation frequency for the majority of common protocols.

The IR receiver we use and recommend is the TSSP4038 or TSSP58038 from VISHAY which are actually proximity sensors or light barrier sensors (or IR Learner) without a specific AGC scheme. We find these useful for decoding a wide range of protocols, but would recommend a dedicated TSOP device and AGC scheme for applications with a single protocol, particularly in noisy environments. Details of IR receivers can be found in the kit section of our website.

However, you will also get good results with any of the TSOP38xxx TSOP58xxx devices from VISHAY or equivalent receivers from other manufacturers. Note that the TSOP devices typically have AGCs optimised for particular protocols and modulation frequencies.

Please check the datasheet before use and note that some IR receivers can operate 'out-of-the-box' in both 5V and 3.3V circuits.

See <http://www.vishay.com/ir-receiver-modules/> for more details of selecting an IR receiver to use.

There is also an informative IR receiver selector guide available at: http://www.vishay.com/docs/49845/49845_sg2145.pdf

Arduino models verified working with ANALYSIR:

- Uno
- Leonardo
- Nano
- DuemilenoVA
- Mega1280 (untested, but should work)
- Mega2560 (untested, but should work)
- Due
- Yún Serial over USB(CDC) OK
- Yún via Wi-Fi in testing (Q1/2 2014)

5. Raspberry Pi Setup

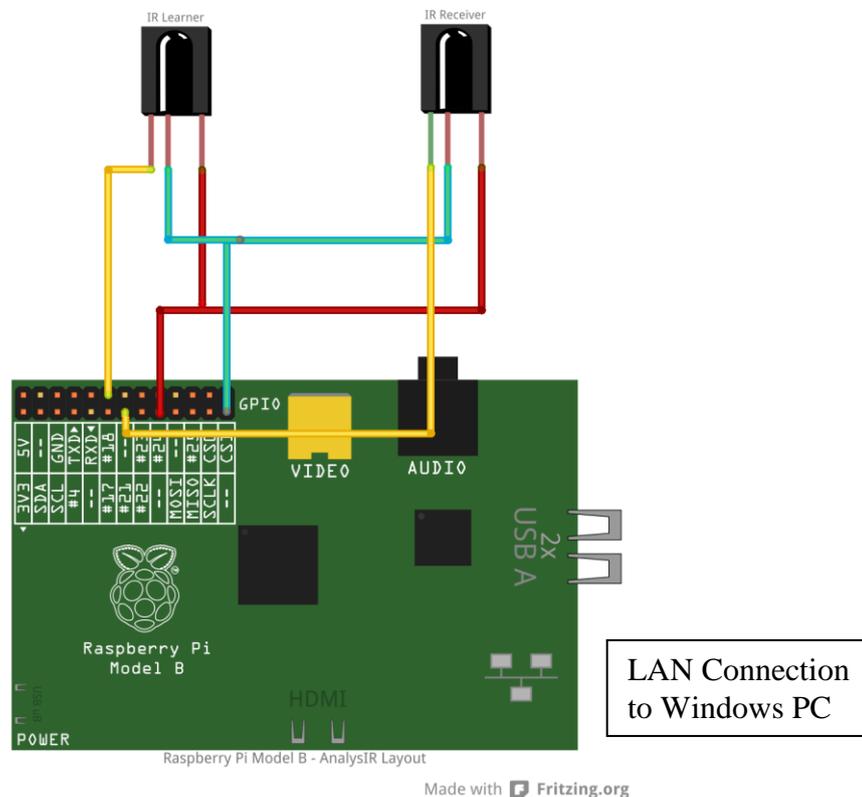
This provides an example circuit layout for the Raspberry Pi based IR receiver:

❖ "Circuit Diagram"

In addition, we make some suggestions for the IR receiver to use.

CIRCUIT DIAGRAM

The following is a 'Fritzing' circuit layout for connecting the IR receiver and optional IR Learner to the Raspberry Pi (RPI), which is a very simple circuit.



The circuit above may be used with an **RPi model B** connected over a local LAN to a Windows PC running ANALYSIR. If you have your RPi configured with Wi-Fi, this will also work. If using other models of RPi it

may be necessary to make minor adjustments to the code & pin-out. Make sure you read the READMEs **and** the comments within the source code files.

In the circuit above, Green(Blue-ish) wires are connected to 'GND', Red are connected to '+3.3V' and the Yellow wires are the Signal outputs from the IR receiver and IR Learner.

With reference to the Raspberry Pi Header connector pins:

- GND is connected to Header Pin 25
- Power to both IR Receivers is connected to Header Pin 17 (3V3)
- The signal output from the IR receiver is connected to Header Pin 13
- The signal output from the IR Learner is connected to Header Pin 12
- We have also designated Header Pin 11 for use with the LIRC example we provided which should be connected to the signal output of an IR receiver and should not be active simultaneously with ANALYSIR (RPi code).

Alternative Pin designations:

- Header Pin 25 is also known as 0V or GND
- Header Pin 17 is also known as 3.3V or Vcc
- Header Pin 13 is also known as GPIO27 or WiringPi/2
- Header Pin 12 is also known as GPIO18 or WiringPi/1
- Header Pin 11 is also known as GPIO17 or WiringPi/0

Important: when using LIRC to record the IR signal, which is the preferred approach, make sure that your IR receiver signal out pin is connected to the correct header pin used by LIRC.

The above pin designations have been taken from several sources, so please ensure the naming convention you use is correct, as we have seen different & conflicting numbering/naming schemes. The circuit diagram on the previous page is the best reference for physical location of the pins (for RPi Model B only).

Advanced users may have enough knowledge to change pin assignments, if necessary. For most users the default assignments above will suffice, but do take note of LIRC pin assignments on your own system.

Important: The pin-out for different IR receivers varies between models, even from the same manufacturer, so confirm the correct pin-out for your device before applying power, by referring to the correct data-sheet. The voltage supply on the RPi is 3.3V (not 5V). Please verify all pin-outs and connections against official RPi documentation, before making any connections.

The IR learners we recommend are the TSMP58000 from Vishay. These pass through the full modulated IR signal to the RPi. The IR receiver passes through the de-modulated signal.

It is not mandatory to have an IR learner component installed to use ANALYSIR, but you will not be able to measure the IR modulation frequency without it.

If you run ANALYSIR just with an IR receiver, you will be able to use every function except those that require the modulation frequency and for many of those it is possible to guess the frequency for common protocols.

The IR receiver we use and recommend is the TSSP4038 or TSSP58038 from VISHAY which are actually proximity sensors or barrier sensors (or IR Learner) without specific AGC. We find these useful for decoding a wide range of protocols, but would recommend a dedicated TSOP device and AGC scheme for applications with a single protocol, particularly in noisy environments. Details of IR receivers can be found in the kit section of our website.

However, you will also get good results with any of the TSOP38xxx TSOP58xxx devices from VISHAY or equivalent receivers from other manufacturers. Note that the TSOP devices typically have AGCs tuned to particular protocols and modulation frequencies.

Please check the datasheet before use and note that some IR receivers can operate 'out-of-the-box' in both 5V and 3.3V circuits. NB: RPi will be damaged with 5V signals.

See <http://www.vishay.com/ir-receiver-modules/> for more details of selecting an IR receiver to use.

There is also an informative IR receiver selector guide available at: http://www.vishay.com/docs/49845/49845_sg2145.pdf

RPi verified working:

Raspberry Pi – Model B

Raspberry Pi – Model A

(Use of other GPIO pins possible)

(Unconfirmed & unavailable for testing,
pin-out varies from Model B diagram)

6. USB IR Toy Setup

The USB IR Toy requires minimal setup as long as you have the latest firmware installed on the device.

To install the latest firmware onto the USB IR Toy (IRT), visit the vendor's website and follow the instructions provided. If you experience problems updating the firmware, try to get it resolved by seeking support from the vendor or via their community forum. If you cannot get it resolved, then seek support via our own IR forum.

Once you have the latest firmware installed, just plug the IRT into a USB port on your Windows PC and make a note of the COM port number assigned (via Control panel->Device Manager->Ports).

Then start up AnalysIR and select the assigned COM port number from the dropdown in the taskbar. Then select the IRT as your source from within the Source menu (Menu->Source-.USB IR Toy).

You should begin to see signals plotted on the screen as you press keys on your remote control.

If you experience issues, follow the instructions for troubleshooting serial port issues in this document.

7. TI LaunchPad Toy Setup

The TI MSP430 F5529 support is currently in beta. If you would like to evaluate this, please contact AnalysIR Directly. If you have a non-standard Arduino that does not support 2 MBPS serial, then select the LaunchPad as your source and programme the Baud Rate in the Arduino code to 115,200 bps.

8. Contacting Support

If you have any problems using ANALYSIR you should report this via email to your designated support contact.

Details of your support contact will be provided with your Licence Key

support@ANALYSIR.com

www.ANALYSIR.com

Follow the menu links to support or go directly via:
<http://www.analysir.com/IRforum/>

9. Appendix A – Sample Arduino & Raspberry Pi code

The sample code for the IR receiver device is included in your data directory after running ANALYSIR for the first time, with a valid Licence key. It should work on most standard Arduino systems with onboard USB serial connections and the RPi via LAN. The wiring for Arduino & RPi is shown in earlier chapters and may require some modifications for certain models of the Arduino or compatible platforms. The sample code may be used or ported to another platform freely without any restriction for use with ANALYSIR.

We have provided some documentation throughout the sample code and associated README which should be sufficient for anyone familiar with the Arduino system & IDE or RPi. It has been tested using Arduino 1.0.5 IDE and should be easily adaptable to other versions. If you prefer to run on another platform, there should be enough information here to help in porting.

A copy of the code is available in both the application data and installation directories after installation.