

# ANALYSIR Getting Started

## User Guide

Version 1.16.100.8109

AMICO34AC • ASHQ24\_312TAG • CANON33CAM • CANVAC96AC •  
 CHIGO96AC • CHUNLAN96AC • DAEWOO • DAIKIN160AC • DAIKIN280AC •  
 DAIKIN64AC • DENON • DIRECTV • DISH • DISNEY • ELECTRA34AC •  
 ELECTROLUX112AC • ELECTROLUX134AC • ELECTROLUX96\_192AC •  
 FRAGTAG • FUJITSU128\_56AC • FUJITSU88DUALAC • GI16STB •  
 GREE35\_67AC • GWTS • HAIER112AC • HAIER144AC • HELI36IGNITE •  
 HELISH3CH • HITACHI104AC • HITACHI184\_264\_296AC • ICOMBAT •  
 IROBOTROOMBA • JAPANESE • JVC • KOMECO112\_128AC •  
 KONICAMINOLTA32CAM • KONKA • LASKO12AC • LG28AC • LG28AC2 •  
 LG28BAC • LGARTCOOLGALLERY140AC • MAGNA10 • MATSUSHITA •  
 MC144105 • MCE • MICROSOFTRC6 • MILESTAG • MILESTAG1 •  
 MITSUBISHI • MITSUBISHI288AC • MITSUBISHI88AC • MITSUBISHI8TV •  
 MOTOROLA • NAGAKAWA96AC • NEC • NEC2 • NIKON4CAM •  
 NORTHSTAR112AC • OGAC104 • OLIMPIA136AC • PANASONIC •  
 PANASONIC128\_216\_280AC • PANASONIC128AC • PENTAX6CAM • PHILLIPS  
 • PIONEER • PIONEER32 • PROTECNO48AC • RC5 • RC6 • RCA • RCMM •  
 RECS80 • ROGERS • SAMSUNG • SAMSUNG112\_168AC • SAMSUNG2 •  
 SANYO • SANYO152AC • SCIENTIFICATLANTA • SERIAL2400BAUD •  
 SERVODAN16DET • SEVILLEFAN64AC • SHARP • SHARP104AC • SONY •  
 SUMIKURA120AC • TCL112AC • TOSHIBA • TOSHIBA\_80\_96\_144\_160AC •  
 TOSHIBA96AC • TRANE64AC • VESTEL168\_112\_56AC • XMP • XSAT •  
 YORK140AC • YORK64AC

**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 for use by third parties, without prior written consent of ANALYSIR and shall only be used for the purpose of this publication.

**Copyright ANALYSIR © 2012-2016. All rights reserved.**

**TRADEMARKS**

All trademarks, service marks or product names included in this document are recognised as the property of their respective owners. Their inclusion is not meant to infer any commercial relationship with or endorsement of ANALYSIR.

## Table of Contents

<b>CHAPTER 1 .....</b>	<b>6</b>
<b>1. INTRODUCTION .....</b>	<b>6</b>
OVERVIEW .....	6
SYSTEM REQUIREMENTS.....	6
ABOUT THIS GUIDE.....	7
ABOUT ANALYSIR .....	8
<b>CHAPTER 2.....</b>	<b>10</b>
<b>2. GETTING STARTED.....</b>	<b>10</b>
INSTALLING ANALYSIR.....	10
OBTAINING YOUR LICENCE KEY.....	10
DEMO MODE.....	11
CONFIGURATION .....	11
INITIAL SETUP.....	12
NETWORK SETTINGS.....	12
MENUS .....	14
VIDEO TUTORIALS.....	14
<b>CHAPTER 3.....</b>	<b>15</b>
<b>3. USING ANALYSIR.....</b>	<b>15</b>
ANALYSING IR SIGNALS WITH ANALYSIR .....	16
DUAL CHANNEL CONCEPT .....	16
CHANNEL 1 & 2 TRACES .....	16
CHANNEL 1 & 2 SETTINGS .....	17
CHANNEL 1 & 2 TAB.....	20
CHANNEL SYNC .....	20
SWITCHING CHANNELS.....	21
RULES TAB.....	21
LOG TAB.....	22
SESSION HISTORY .....	22
SAVE SCREENSHOT .....	24
EXPLORE .....	24
PROPERTIES VIEW.....	25
CLEAR HISTORY .....	25
CLEAR RAW SIGNALS .....	25
CLEAR BOTH .....	25
CLEAR ALL .....	25
SERIAL PORT .....	25
IMPORT.....	26
EXPORT .....	28
BATCH EXPORT.....	29
TRUNCATE SIGNAL.....	30
PROTOCOL MENU .....	30
SERIAL PORT RESET .....	31
SERIAL PORT DISABLE .....	32
RESTART .....	32
EXIT.....	32
POWER TOOLS MENU.....	33
SEND IR .....	33
SEND SELECTED IR .....	34

MSB / LSB / LSB8 .....	34
PULSE ANALYTICS .....	35
CHECKSUM CALCULATOR.....	35
REVERSE ENGINEERING TOOL.....	36
HISTORY DECODE .....	37
CLEAN HISTORY .....	37
PRIORITY.....	37
SOURCE MENU .....	37
SHOW CARRIER FREQUENCY .....	40
DETAILS TAB .....	40
HELP MENU.....	40
SERIAL PORT SELECTION .....	41
SERIAL PORT INDICATORS .....	41
CUSTOM IR PROTOCOLS .....	42
WINDOW RE-SIZING.....	42
<b>CHAPTER 4.....</b>	<b>44</b>
<b>4. LEARNIR SETUP.....</b>	<b>44</b>
<b>CHAPTER 5.....</b>	<b>45</b>
<b>5. A.IR SHIELD SETUP .....</b>	<b>45</b>
A.IR SHIELD NANO .....	45
A.IR SHIELD PHOTON .....	45
A.IR SHIELD RX .....	45
<b>CHAPTER 6.....</b>	<b>46</b>
<b>6. ITACH FLEX WIFI SETUP.....</b>	<b>46</b>
<b>CHAPTER 7.....</b>	<b>48</b>
<b>7. ARDUINO SETUP.....</b>	<b>48</b>
CIRCUIT DIAGRAM .....	48
<b>CHAPTER 8.....</b>	<b>51</b>
<b>8. RASPBERRY PI SETUP.....</b>	<b>51</b>
CIRCUIT DIAGRAM .....	51
<b>CHAPTER 9.....</b>	<b>55</b>
<b>9. USB IR TOY SETUP.....</b>	<b>55</b>
<b>CHAPTER 10.....</b>	<b>56</b>
<b>10. TEENSY SETUP .....</b>	<b>56</b>
<b>CHAPTER 11.....</b>	<b>57</b>
<b>11. CHIPKIT – FUBARINO MINI .....</b>	<b>57</b>
<b>CHAPTER 12.....</b>	<b>58</b>
<b>12. PARTICLE – PHOTON .....</b>	<b>58</b>

<b>CHAPTER 13.....</b>	<b>59</b>
<b>13. ESP8266 – NODEMCU .....</b>	<b>59</b>
<b>CHAPTER 14.....</b>	<b>60</b>
<b>14. TI LAUNCHPAD SETUP .....</b>	<b>60</b>
<b>CHAPTER 15.....</b>	<b>62</b>
<b>15. PSOC4 SETUP.....</b>	<b>62</b>
<b>CHAPTER 16.....</b>	<b>65</b>
<b>16. CONTACTING SUPPORT.....</b>	<b>65</b>
<b>CHAPTER 17.....</b>	<b>66</b>
<b>17. APPENDIX A – SAMPLE FIRMWARE &amp; RASPBERRY PI CODE.....</b>	<b>66</b>
<b>CHAPTER 18.....</b>	<b>67</b>
<b>18. APPENDIX B – CONFIGURATION FILE, ANALYSIR.INI .....</b>	<b>67</b>

# 1. Introduction

---

This chapter introduces ANALYSIR and includes the following sections:

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

## OVERVIEW

ANALYSIR provides a Windows desktop application to analyse and decode Infrared (IR) signals recorded via devices such as LEARNIR, A.IR Shield Nano, A.IR Shield Photon, A.IR Shield rx, Arduino, Raspberry Pi, iTach Flex, ESP8266/NodeMCU, USB IR Toy, TI LaunchPads, Teensy, Photon, PSOC4 and ChipKit/Fubarino etc.

ANALYSIR can also send IR signals using LEARNIR, A.IR Shield and USB IR Toy to assist in testing.

It is also possible to operate ANALYSIR in 'Demo' mode whereby many of the features can be evaluated before purchasing. In demo mode, the connection to the recording/sending device cannot be used. Instead it is possible to load included sample session histories, for evaluation purposes only. In Demo mode you will be automatically re-directed to the ANALYSIR.com website, with the opportunity to upgrade to a full version.

## SYSTEM REQUIREMENTS

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

- Microsoft Windows PC, with latest updates (including .NET)
- A supported device for recording IR signals:
  - a. Arduino or compatible running at 16MHz (although slower CPU speeds may work) with USB serial interface (LearnIR, A.IR Shield Nano, A.IR Shield Photon, A.IR Shield rx, ESP8266/NodeMCU, Chip

Kit/Fubarino, Teensy, LaunchPad, Photon or other MCUs are also possible). Sample firmware code is available for these devices which can also be easily modified for other platforms.

- b. A copy of the relevant IDE (Arduino, MPIDE, Energia, Creator, Particle and Teensy etc.) is also required for the chosen platforms above, to compile and upload the firmware.
  - c. **LEARNIR** should be connected via USB/ FTDI cable or equivalent and works out-of-the-box.
  - d. **A.IR Shield Nano** should be connected via the Nano's USB cable and works out-of-the-box.
  - e. **A.IR Shield Photon** should be connected via the s USB cable or WiFi and works out-of-the-box once the firmware is uploaded.
  - f. **A.IR Shield rx** can be connected directly to Arduino or any other platform.
  - g. Raspberry Pi, is connected via LAN network or Wi-Fi.
  - h. iTach Flex, which is a commercial product available from Global Cache works after some network configuration..
  - i. USB IR Toy requires IRToy.exe for sending IR signals. The latest USB IR Toy firmware is available from the vendor's web site.
- A copy of the ANALYSIR Licence text and a Valid Licence Key are required at all times. Obviously, a Licence Key is not required for 'Demo Mode'. However, you must agree to the Licence terms to run ANALYSIR in any mode.
  - An IR emitter such as a TV or AC remote control to generate the signals to be analysed.

## ABOUT THIS GUIDE

This guide provides information on the following:

- ❖ *"Getting Started"*
- ❖ *"Using ANALYSIR"*
- ❖ *"LEARNIR setup"*
- ❖ *"A.IR Shield setup"*
- ❖ *"Arduino setup"*
- ❖ *"Photon setup"*
- ❖ *"ESP8266/NodeMCU setup"*
- ❖ *"Raspberry PI setup"*
- ❖ *"USB IR Toy setup"*
- ❖ *"iTach Flex setup"*
- ❖ *"Teensy setup"*

- ❖ "ChipKit/Fubarino setup"
- ❖ "TI LaunchPad setup"
- ❖ "PSOC4 setup"
- ❖ "Contacting Support"
- ❖ "Sample Firmware & RPi code"
- ❖ "Configuration File – ANALYSIR.ini"

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.

**Tip:** If you need to find something quickly, use the search facility of your document viewer or review the table of contents.

## 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, Installation & Repair Engineers.

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

ANALYSIR is a software only product and requires additional 3<sup>rd</sup> Party equipment such as a Windows PC and Arduino (or similar IR Source) 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 carrier frequency range from 20 – 60 kHz.

The list of 100+ base IR protocols decoded by ANALYSIR, representing many tens of thousands of devices, includes:

AMICO34AC, ASHQ24\_312TAG, CANON33CAM, CANVAC96AC, CHIGO96AC, CHUNLAN96AC, DAEWOO, DAIKIN160AC, DAIKIN280AC, DAIKIN64AC, DENON, DIRECTV, DISH, DISNEY, ELECTRA34AC, ELECTROLUX112AC, ELECTROLUX134AC, ELECTROLUX96\_192AC, FRAGTAG, FUJITSU128\_56AC, FUJITSU88DUALAC, GI16STB, GREE35\_67AC, GWTS, HAIER112AC, HAIER144AC, HELI36IGNITE, HELISH3CH, HITACHI104AC, HITACHI184\_264\_296AC, ICOMBAT, IROBOTROOMBA, JAPANESE, JVC, KOMECO112\_128AC, KONICAMINOLTA32CAM, KONKA, LASKO12AC, LG28AC, LG28AC2, LG28BAC, LGARTCOOLGALLERY140AC, MAGNA10, MATSUSHITA, MC144105, MICROSOFTRC6, MILESTAG, MILESTAG1, MITSUBISHI, MITSUBISHI288AC, MITSUBISHI88AC, MITSUBISHI8TV, MOTOROLA,



NAGAKAWA96AC, NEC, NEC2 , NIKON4CAM,  
NORTHSTAR112AC, OGAC104, OLIMPIA136AC, PANASONIC,  
PANASONIC128\_216\_280AC, PANASONIC128AC,  
PENTAX6CAM, PHILLIPS, PIONEER, PIONEER32,  
PROTECNO48AC, RC5, RC6, RCA, RCMM , RECS80, ROGERS,  
SAMSUNG, SAMSUNG112\_168AC, SAMSUNG2, SANYO,  
SANYO152AC, SCIENTIFICATLANTA, SERIAL2400BAUD ,  
SERVODAN16DET, SEVILLEFAN64AC, SHARP, SHARP104AC,  
SONY, SUMIKURA120AC, TCL112AC, TOSHIBA,  
TOSHIBA\_80\_96\_144\_160AC, TOSHIBA96AC, TRANE64AC,  
VESTEL168\_112\_56AC, XMP , XSAT, YORK140AC, YORK64AC.

– With many of these decoding multiple variants of the base protocol.

All Infrared remote control protocols can be analysed with ANALYSIR.

## 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"*
- ❖ *"Menus"*
- ❖ *"Video Tutorials"*

### INSTALLING ANALYSIR

To install ANALYSIR you must first download the installation package from our 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.

**TIP:** Click the email icon to the right of the Signature field and all of the required information will be copied to your clipboard for pasting into your registration email request.

- 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. There are multiple sample signal sets provided with ANALYSIR, via  
Menu ⇒ File ⇒ Load Saved Session ⇒ Load Sample Signals
- Be sure to make of the tooltips which provide more information about the various fields and options. They are displayed as you hover the mouse over on-screen objects.
- On a small number of systems it may be necessary to obtain special registration keys because the PC can have its hardware reconfigured between restarts/resumes (e.g. docking stations). In these cases you may be asked to enter a special KEY as a workaround for this situation.

## 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 (e.g. 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 most features except sending & receiving live IR signals from the recording device. Importing of 3<sup>rd</sup> 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 COM port for your recording device (e.g. Arduino) in the combo box at the bottom of the screen. The default is set to

'COMxx' and 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. *(See Appendix B for more details on the Configuration file)*

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

Detailed instructions for installing the Arduino, RPi, USB IR Toy, Teensy, ChipKit/Fubarino, PSOC4 and LaunchPad firmware are contained later 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. Also refer to the 'Infographic' provided for your particular device.

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

It is assumed that you are familiar with the Arduino IDE and loading new software onto your Arduino or equivalent for 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 those currently supported, it should be relatively simple to alter the provided source 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 115,200 bps. In the case of the Raspberry Pi, Photon WiFi & iTach FLEX the communication is over the LAN or Wi-Fi network and is in a different format.

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 serial COM port on the PC. In order for this to work the 'Source' must be configured to work with ANALYSIR (See specific instructions provided 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.ini

configuration 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
#WAIT_NETWORK_RPi=750        #Leave this line commented out, unless advised
otherwise

#

IPAddress_iTachFlex=000.000.000.000 #Change this to IP address of your iTach Flex
Port_iTachFlex =25             #Change this to Port number used with your iTach Flex
#WAIT_NETWORK_iTachFlex=750    #Leave this line commented out, unless advised
otherwise

#

IPAddress_Photon=000.000.000.000 #Change this to IP address of your RPi
Port_Photon=25                 #Change this to Port number used with LIRC & socat on your RPi
#WAIT_NETWORK_Photon=750      #Leave this line commented out, unless advised
otherwise

#

IPAddress_ESP8266NodeMCU=192.168.0.94 #Change this to IP address of your
ESP8266NodeMCU
PORT_ESP8266NodeMCU=25 #Change this to Port number used on your ESP8266NodeMCU
#WAIT_NETWORK_ESP8266NodeMCU=750 #Leave this line commented out, unless advised
otherwise

#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 Arduino Yún
Port_Yun=25                   #Change this to Port number used with LIRC & socat on your RPi
#WAIT_NETWORK_Yun=750        #Leave this line commented out, unless advised
otherwise
```

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⇒Raspberry Pi) or

(Menu⇒Source⇒iTach Flex) or

(Menu⇒Source⇒Photon WiFi)

(Menu⇒Source⇒ESP8266 NodeMCU WiFi)

The default setting for network timeout is commented out above. However, if you are experiencing network problems or are on a slow/busy/noisy Wi-Fi network then uncommenting the 'Wait\_Network' parameter for your device and setting it to 2000 or 5000 may resolve some issues and/or improve performance. Normally, it is not necessary to adjust this parameter.

Currently, only the A.IR Shield Photon, Raspberry Pi, iTach & Photon are supported for network connection. The Arduino Yún is currently supported via Serial and should be configured as an Arduino in the Source menu & Leonardo in the firmware source code.

**Note:** Connection of RPi via serial COM port is not supported.

## MENUS

AnalysIR features a main menu along the top of the main window and several context menus which popup on right-clicking other parts of the main window with the mouse. Drop down menus will automatically expand on mouse hover or when shortcut keys are pressed.

There are 3 context menus (History Context Menu, Serial Port Menu and channel Sync Menu) which popup with a right mouse click. The History context menu is active over the History panel; the channel sync menu is active over the channel tabs and panels. The serial context menu is active over the com port drop-down box, the baud rate field and serial colour status box located in the status bar.

The context menus are also available via the main menu system under the Power tools menu, File menu and Channel 1|2 menus respectively.

**Tip:** Clicking the Logo or the website address will open a browser window at [www.AnalysIR.com](http://www.AnalysIR.com), where you can find all the latest information about AnalysIR activities.

## VIDEO TUTORIALS

We have published a number of brief video tutorials on our YouTube channel (AnalysIRchannel) covering the following topics:

- [Tutorial: AnalysIR Reverse Engineering Tool for Infrared AC Signals](#)
- [Tutorial: AnalysIR Checksum Calculator for AC Infrared Signals](#)
- [AnalysIR Demo Batch Infrared Decoding](#)
- [GUI update video AnalysIR](#)
- [AnalysIR Demo video1 1](#)

The channel is available via <https://www.youtube.com/user/AnalysIRchannel> and we hope to upload additional video content over time covering both the AnalysIR application and other products as they are released.

## 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"*
- ❖ *"Dual Channel Concept"*
- ❖ *"Channel 1 & 2 Traces"*
- ❖ *"Channel 1 & 2 Settings"*
- ❖ *"Channel 1 & 2 Tab"*
- ❖ *"Channel Sync"*
- ❖ *"Switching Channels"*
- ❖ *"Rules Tab"*
- ❖ *"Log Tab"*
- ❖ *"Session History"*
- ❖ *"Save Screenshot"*
- ❖ *"Explore"*
- ❖ *"Properties View"*
- ❖ *"Clear History"*
- ❖ *"Clear RAW Signals"*
- ❖ *"Clear Both"*
- ❖ *"Clear All"*
- ❖ *"Serial Port"*
- ❖ *"Import"*
- ❖ *"Export"*
- ❖ *"Batch Export"*
- ❖ *"Truncate Signal"*
- ❖ *"Protocol Menu"*
- ❖ *"Serial Port Reset"*
- ❖ *"Serial Port Disable"*
- ❖ *"Restart"*
- ❖ *"Exit"*
- ❖ *"Power Tools Menu"*
- ❖ *"Send IR"*
- ❖ *"Send Selected IR"*
- ❖ *"MSB / LSB / LSB8"*
- ❖ *"Pulse Analytics"*
- ❖ *Checksum calculator*
- ❖ *Reverse Engineering Tool*
- ❖ *"History Decode"*
- ❖ *"Clean History"*
- ❖ *"Priority"*
- ❖ *"Source Menu"*
- ❖ *"Show Carrier Frequency"*
- ❖ *"Details Tab"*
- ❖ *"Help Menu"*
- ❖ *"Serial Port Selection"*
- ❖ *"Serial Port Indicators"*
- ❖ *"Custom IR Protocols"*
- ❖ *Window resizing*

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 device (e.g. 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.

When using a source device that can report carrier frequency, AnalysIR will record the carrier frequency for each signal. Both the LearnIR & A.IR shields from AnalysIR were designed with this feature as standard. If you configure your device with the TSMP58000 (available in our kits), you will also get this feature by using our firmware. Alternatively, if you use the IR Toy you will get a slightly more limited feature set for recording carrier frequency.

The recorded carrier frequency, is then used when saving, sending or exporting signals from AnalysIR. The recorded carrier frequency can be easily overridden via the 'History Context Menu'. Where no carrier frequency is available for a signal, it is recorded as '0', '-' or ' '.

## DUAL CHANNEL CONCEPT

ANALYSIR supports 2 'virtual' channels. This means that the recording device (e.g. 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. Users can also copy signals between channel traces and compare different 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 from the signal start, 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, provided the 'measure' function is enabled for that channel. 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 either or both signal traces may be saved to your filesystem via:

Menu ⇒ File ⇒ Save Trace Image (or CTRL+8)

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). A file save dialog is displayed to help with choosing the target file location.

File paths are remembered when saving an image and when AnalysIR is restarted, provided the application has been closed normally.

## 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 may 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 be reset to muted 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)
- **Plot Colour** – You can click on this box to change the colour of the signal plot for new signals.  
(Menu ⇒ Channelx ⇒ Change Plot Colour)
- **Overlay Colour** – You can click on this box to change the colour of the overlay trace for new signals.  
(Menu ⇒ Channelx ⇒ Change Overlay 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.

However, the signal will remain in the History for later use.

(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. The new signal is also created in the History.

(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, but not saved. 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) table 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 recording device (e.g. 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 mark or space, in microseconds.
- **Duration** – This represents the duration of the current mark or space of the signal, in microseconds.
- **State** – The state can be '0' or '1'. A space is indicated by '0' and a mark is indicated by '1'

Clicking any row in either table will display a graphical indicator in the associated channel trace at a point matching that mark or space.

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

## CHANNEL SYNC

When changing parameters for channels it is now possible to 'sync' both channels such that when a value is changed for one channel the same value is set for the other channel. The parameters supported for syncing are:

- Overlay
- Invert
- Lock
- Beep
- Discrete
- Scale
- Start
- Tolerance
- Clear
- Channel Tables

For example, if scale is synced, zooming in and out of the signal in channel 1, will do the same for the signal in channel 2. When Channel tables is synced, selecting a row in the channel 1 table will select the same row in the channel 2 table, which can prove useful when comparing signals and mark or space durations.

When a parameter is synced its label will be underlined and in the case of 'Channel tables' the tab labels will be underlined – as visual indicators of this setting.

## SWITCHING CHANNELS

When recording signals, the signal will be placed into the currently selected channel, which is indicated in the status bar and by the currently selected Channel tab.

You may switch channels in a number of ways, as follows:

- Clicking the channel tab
- Clicking a channel panel
- Clicking the Selected channel indicator text in the status bar
- Double-clicking the timeline for a channel, located below each trace
- Clicking the 'Select Channel 1|2 option in the channel menus
- From the keyboard by pressing ALT+1 | ALT+2

## 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 recording device (e.g. Arduino or other MCU) is displayed. In addition some debug information from ANALYSIR is also displayed.

If you want to replace the Arduino with another MCU which is not already supported, 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 automatically 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 recorded on.

**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.

**Button** - 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 buttons on your remotes controls. You can then save all the signals 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.

If a new signal is recorded and successfully decoded and matches a signal already in the History – it will be automatically populated with the button ID of the existing signal.

**Value** – a Hexadecimal value of the decoded signal.

**Value2** – for some protocols a second value is also recorded. For RC5 an alternative representation of the value is recorded here for convenience.

**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.

**Carrier** – displays the carrier frequency for each signal, if available. Can also be edited using the context or History menu.

Session histories can be saved to your filesystem (Menu ⇒ File ⇒ Save Session or CTL+S). Additionally, Saved sessions can be reloaded for later review (Menu ⇒ File ⇒ Load Saved Session). You will then be presented with a number of options via the menu:

- Replace History – the current history will be cleared and the new file loaded in its place
- Append to History – the current history will be retained and the new file will be appended to History.
- Append to Channel 1 – the current history will be retained and the new file will be appended to History, with all new signals allocated to channel 2, regardless of channel setting in file.
- Append to Channel 2 – the current history will be retained and the new file will be appended to History, with all new signals allocated to channel 2, regardless of channel setting in file.

**Tip:** *Clearing the history first, before issuing 'Append to Channel 1|2' allows you to easily overwrite the channel number in a session.*

A file open/save dialog is displayed to help with choosing the file location. 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 should uniquely describe 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.

When saving session files, AnalysIR remembers the file paths used and will retain these values when closed and restarted. Any invalid characters in the file name will be replaced. The description will remain unchanged.

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.

You may delete a History row by selecting the row and pressing the 'Del' key). You may copy the contents of a single row by clicking the row and pressing CTRL+C on your keyboard. The contents are now available in your clipboard for pasting into other applications. (Alternatively use the context menu for



deleting/copying)

The total number of signals in the session history, is displayed at the top right of the Session History panel.

The '**History Context Menu**' is available by right clicking your mouse over the history grid or via Menu⇒Power Tools⇒History Menu. The context menu allows you to delete the selected row, copy the selected row to your clipboard, move the selected row up/down, reset the protocol of the selected/all rows to RAW, or update the carrier frequency of the selected row. You can also set all rows:

- to the common 38 kHz carrier frequency
- to the frequency of the currently selected row
- to automatically round off the frequency to the nearest standard frequency for the selected row or all rows (Auto or Auto All Rows).

You can also set the carrier frequency for the current row to one of the common (30, 33, 36, 38, 40, 56 kHz) carrier frequencies

**Tip:** *If you want to set all rows to a carrier frequency other than 38 kHz, first set one row to the required frequency and then all rows to the frequency of the currently selected row.*

When selecting a new row, the stored carrier frequency is displayed in the status bar, if available.

By clicking the 'Auto Insert Button IDs' in the History context menu, the Button column will be populated with automatically generated button IDs (e.g. B1, B2, B3, ..., Bn). The sequence will start from the next available number and will be written to all empty positions. This feature was added to assist in situations where large numbers of signals are being recorded for later processing, typically in commercial environments.

## SAVE SCREENSHOT

This menu option allows users to save an image of the main window to your file system. This can be of great assistance when seeking support or for publishing examples of your work. A file save dialog is displayed to help with choosing the file location. File paths are remembered when AnalysIR is restarted, provided it has been closed normally.

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 (AnalysIR.ini file) 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 or CTRL+P). 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. This can be useful for sending debugging information to support.

## CLEAR HISTORY

Selecting this option from the menu (Menu ⇒ File ⇒ Clear History or CTRL+H) clears or deletes all signals from the displayed history. There is no undo option.

## CLEAR RAW SIGNALS

Selecting this option from the menu (Menu ⇒ File ⇒ Clear RAW Signals or CTRL+R) clears or deletes all RAW signals from the displayed history. Other signals remain.

## 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. The signals are not deleted from history.

## 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 set 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 COMxx. 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. To find the correct directory, select Menu ⇒ File ⇒ Explore.

The default settings for the serial port are as follows:

**Baud:** 115,200

**Data Bits:** 8  
**Start Bits:** 1  
**Stop Bits:** 1  
**Parity:** none

These settings match the default settings on LEARNIR, A.IR Shield, Arduino and many other MCUs except for the Baud rate.

**Tip:** *If you are porting AnalysIR firmware to unsupported firmware, it is now possible to override the default Baud rate in the 'ini' settings. Most users should never need to do this.*

## IMPORT

You may import IR signals into ANALYSIR in a selection of formats including: LEARNIR, IRremote, IRLib, Global Caché (GC-100, iTach, GC\_IRL, GC-IRE), Pronto, Command Fusion and Saleae Logic CSV. The full list is shown below. We expect 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 one 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)
- LEARNIR
- IRremote
- IRLib
- Pronto CCF
- Global Caché GC-100
- Global Caché iTach
- Global Caché GC-IRL
- UEI
- Global Caché GC-IRE
- Global Caché iLearn
- Command Fusion
- Saleae Logic CSV
- USB IR Toy .bin
- LIRC RAW
- IRScope ICT

**Tip:** If you are unsure of the above formats, just export any signal in the format required. (See Export feature below)

From within 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*

For the USB IR Toy, select this option from the drop-down box and you will be presented with a file open dialogue. Then open the required file to proceed. Alternatively, you can just paste the full file path into the large text box and use the auto-detect feature.

It is also possible to import bulk signals at once by separating the signals by '\$\$\$' on a line between each signal to be imported. In this case the signals may be of any supported type or format. Mixed signal types are supported during bulk import. To import paste a properly formatted set of signals into the text box and click the 'Bulk Import' button. All signals will be imported into Channel 1. The size of text to be imported should not exceed 2Mbytes. If you require more, then split your data up into <2MB chunks and do several imports.

e.g. (bulk import of 2 Global Caché formatted signals)

GC-  
IRE,38000,21,63AAAAAAA21,23BB21,19BBBBBAAAAAAAABBBBBBBAAAA  
AAAABBBBBBBBA19,277

**\$\$\$**

GC-  
IRE,38000,21,63AAAAAAA21,23BB21,19BBBBBAAAAAAAABBBBBBBAAAA  
AAAABBBBBBBBA19,277

**\$\$\$**

GC-  
IRE,38000,21,63AAAAAAA21,23BB21,19BBBBBAAAAAAAABBBBBBBAAAA  
AAAABBBBBBBBA19,277

*Note: LIRC RAW files can contain multiple signals and does not require this 'bulk import (\$\$\$)' feature. Instead you can simply import a LIRC RAW file with the multiple signals by just pressing the Bulk Import button, after loading the contents into the text area.*

An alternative method for importing signals is directly from the file system. This is identical to the traditional way of importing, except, instead of manually pasting the signal to be imported into the text field, it is now read from a file. To import from file, click on the file open icon (button) and you will be presented with a file open dialog. Navigate to the required file and click the open button. The contents of the selected file will be copied to the text field. Thereafter, importing proceeds as normal and as described above. This feature will work for single and batch import of

signals.

It is also possible to add the Key name (button, function or ID) when importing a signal. This will ensure that the signal is identified with that name in the History. Simply enter the Key name into the 'Key ID' field before importing. *(This feature is not supported for bulk imports, as all signals would end up having an identical name.)*

**TIP:** If loading a saved session file, do not use this Import feature. Instead use the Load Saved Session feature in the File menu.

## EXPORT

You may export IR signals from ANALYSIR in a selection of formats including: LEARNIR, A.IR Shield, IRremote, IRLib, Global Caché (GC-100, iTach, GC\_IRL, GC-IRE), Pronto, USB IR Toy .bin, LIRC RAW 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 always select the required Export format from the drop down box.

Always ensure that the correct carrier frequency is set or stored for LEARNIR, A.IR, Global Caché, USB IR Toy and Pronto formats, before exporting. For Global Caché formats you may also select compression, which will produce a much smaller set of data. LEARNIR is always in compressed format. 'Auto Clean' analyses the signal and cleans degraded signals which are noisy or inaccurate 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 or saving to file. This can be useful in investigating the causes of degraded IR signals. *(See also the 'Clean History' feature under the Power Tools menu)*

Auto Clean works with successfully decoded signals only, but not RAW signals. During export the 'Auto clean' feature is now defaulted to 'off', to avoid inadvertent adjustments to the exported signal. To enable just click the check box. It is also possible to clean all signals in history via the Power Tools menu.

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

As AnalysIR now stores carrier frequency with signals (when available), we have provided the option to override these stored frequencies with the 'Override carrier' checkbox. When checked, the Carrier frequency in the respective channel text box will be used instead of the stored value. If there is no valid stored value then this invalid value will be overridden, regardless of the state

of the checkbox. Remember carrier frequency is not relevant for all export formats.

Checking the 'Brief Description' checkbox the Button ID stored in History, is included before each signal.

Checking the 'Verbose Description' checkbox inserts comments along with the exported signals, including any useful information stored with the signal, such as button name or carrier frequency etc.

Checking the 'Auto Save' checkbox, will automatically prompt you to save the exported signal(s) in the file system. This can be useful when doing large scale captures from multiple remote controls. AnalysIR will remember the file path selected, for the next time.

Clicking the 'Export from History' button, will 'bulk' export every signal contained in History, in one simple step.

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

The following formats are supported for export:

- LEARNIR
- IRLib
- IRremote
- Command Fusion
- Pronto CCF
- USB IR Toy .bin
- A.IR Shield (send)
- LIRC RAW
- Global Caché GC-100
- Global Caché GC-IRE
- Global Caché GC-IRL
- Global Caché iTach

## 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.

Menu ⇒ File ⇒ Batch Export ⇒ IRremote

Menu ⇒ File ⇒ Batch Export ⇒ IRLib

Menu ⇒ File ⇒ Batch Export ⇒ LIRC RAW

Menu ⇒ File ⇒ Batch Export ⇒ PSOC4

If the signal's carrier frequency is recorded in AnalysIR, it will also be available in the output from batch export, for RAW signals.

**Tip:** The A.IR shield can be loaded with IRremote & IRLib firmware.

The following formats are supported for Batch export:

- IRLib

- IRremote
- LIRC RAW
- PSOC4

*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. Alternatively, just use the RAW format with IRLib.*

## TRUNCATE SIGNAL

You may make a partial copy of an existing signal by using the 'truncate' feature available in the 'Channel 1|2' menu.

Menu ⇒ Channelx ⇒ Truncate ⇒ Before

Menu ⇒ Channelx ⇒ Truncate ⇒ After

For example, if the displayed signal is repeated multiple times and you just want to use the first part of the signal:

- Click on the Channel 1 or 2 tab as appropriate
- Move the cursor to the (long) space after the first signal, such that it is also highlighted in the channel tab display.
- Select Truncate ⇒ Before from that channel menu (or CTRL+ALT+B if channel 1)
- The first part of the signal will then be copied to the other channel.
- Remember if you get it wrong, you can delete the new signal from history by selecting the row and pressing the 'del' key. (or use the context menu)
- You may extract an internal part of a signal by repeating the process above multiple times.

Refer to the menu system for the 4 shortcut key codes available with this feature

*Note:*

- *If a space is highlighted, it is not included in the new signal.*
- *If a mark is highlighted, it is included in the new signal.*

## 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, AC & 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 menu 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

You can limit decoding to only AC signals by selecting the AC only option from the Protocols menu. This will disable decoding for all protocols except Air Conditioner signals. You can also combine this with the toggle menu option to decode all protocols except AC signals.

Menu ⇒ Protocol ⇒ AC Only

The DISNEY protocol is actually a serial protocol at 2,400 BAUD. Because of this it is quite easy to get 'false positives' for other protocols when DISNEY is enabled. Accordingly, this protocol must always be enabled manually and is defaulted to disabled.

## SERIAL PORT RESET

You may manually reset the Serial Port from the File or context menus:

Menu ⇒ File ⇒ Serial Port ⇒ Serial Port Reset or ALT+S

Serial Port Context Menu ⇒ Serial Port Reset

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 USB connected devices not being recognised properly on 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.***

This reset feature is also available via a context menu at the bottom right of the main window (Right click when the mouse is over any of the serial port related controls).

## SERIAL PORT DISABLE

You may manually disable the Serial Port from the menu:

Menu ⇒ File ⇒ Serial Port Disable or CTRL+ALT+S

Serial Port Context Menu ⇒ Serial Port Disable

ANALYSIR will then close the serial port selected in the dropdown at the bottom right of the screen (if open). This allows users to access their device from other applications without having to close ANALYSIR. The colour status box will turn grey until the disable option is unchecked.

This feature is also available via a context menu at the bottom right of the main window. (Right click when the mouse is over any of the serial port related controls).

## RESTART

You may manually restart ANALYSIR from the menu:

Menu ⇒ File ⇒ Restart or ALT+R

ANALYSIR will then close and re-open immediately. This feature is mainly useful when you have changed some of the settings or properties in the 'ini' file and want to reload them.

## 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](http://www.ANALYSIR.com).

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

## SEND IR

This feature allows you to send one or more IR signals from the displayed history using supported devices. Note: not every connected device will have the ability to send IR or may not be supported for sending. Sending IR is supported for LEARNIR, A.IR Shield Nano, A.IR Shield Photon and USB IR Toy only.

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

Menu ⇒ Power Tools => Send IR

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. This can prove to be useful to ensure that the signal you have captured can be read by the target device.

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.

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 sent 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.

The total number of marks or spaces and the total duration (µSecs) of each sequence sent is displayed below the 'Signal timings' text box.

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

LEARNIR and A.IR Shield both work seamlessly for sending with ANALYSIR, because they are tightly integrated. When using the USB IR Toy the first time for sending, you will be prompted to place a copy of IRTToy.exe into the ANALYSIR directory. (Loosely integrated and available only from the vendor's website).

When sending, always be sure to select the correct carrier frequency from the drop-down box and refer to the signal history in the main window for details, if available.

## 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.

LEARNIR and A.IR Shield both work seamlessly for sending with ANALYSIR, because they are tightly integrated. When using the USB IR Toy the first time for sending, you will be prompted to place a copy of IRTToy.exe into the ANALYSIR directory. (Loosely integrated and available only from the vendor's website).

When sending, always be sure to select the correct carrier frequency from the drop-down box and refer to the signal history in the main window for details, if available.

## MSB / LSB / LSB8

When ANALYSIR decodes a signal into HEX (binary) format it treats the first data bit received in the signal as the most significant bit or MSB by default. If you select LSB, from the menu, ANALYSIR will reverse this and treat the first bit receives as the least significant bit. Finally, LSB8 takes the MSB format hex value and turns each 8 bit byte of the overall HEX value into LSB.

For example, the MSB HEX (Binary) value below is converted into

LSB or LSB8 as follows:

MSB: 01 02 03 (0000 0001 0000 0010 0000 0011)

LSB: C0 40 80 (1100 0000 0100 0000 1000 0000)

LSB8: 80 40 C0 (1000 0000 0100 0000 1100 0000)

This can be particularly useful when reverse engineering Air Conditioner protocols which are almost always LSB8, unlike TV signals.

To convert to LSB8 format, the signal should ideally have a multiple of 8 bits (If not, AnalysIR makes a best effort attempt at rearranging the bit-stream to LSB8). We suggest to leave the setting at MSB, when recording signals and to change the setting only when attempting to reverse engineer the signal.

## PULSE ANALYTICS

You may visualise analytics on a signal via:

Menu ⇒ Power Tools ⇒ 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.

## CHECKSUM CALCULATOR

To assist in reverse engineering Air Conditioner protocols (or any checksum), we have provided a feature to automatically generate or verify the checksum of a decoded signal. The algorithms used are:

SUM (Byte & Nibble)

XOR (Byte & Nibble)

CRC8 (Byte only)

CRC16 (Byte only)

In the pop-up window you may select any of the signals in history and ANALYSIR will calculate the checksum for that signal and display the result for each of the algorithms above. If any of the checksums match and you verify this for multiple signals then you will know how the checksum is calculated for that signal. You may need to change the bit order of decoded signal to LSB8, MSB or LSB to find the solution. You can also enter your own Hex values into the 'Hex Value' field for signals not in the History or if you want to check parts of the signal. Alternatively, you can ignore sections of the signal by populating the 'Ignore Fields'.  
*(Tip: Often the last byte is the checksum and should be ignored)*

*for many AC signals.)*

The binary text field displays a binary representation of the Hex value. Signals can be selected from the 'Signal' drop-down' and can be refreshed or updated by clicking the 'Refresh' button.

The checksum calculator can be accessed via the Power Tools menu.

Menu ⇒ Power Tools => Checksum Calculator

## REVERSE ENGINEERING TOOL

To assist in reverse engineering Air Conditioner protocols (and checksums), we have provided a tool to simplify this process. The Reverse Engineering Tool can be accessed via the Power Tools menu.

Menu ⇒ Power Tools => Reverse Engineering Tool

First make sure that you have captured a good set of signals and that the protocol has been recognised by AnalysIR. The tool will not work for signals recorded as Raw. As most uses of this tool will be related to AC signals, it is usually a good idea to select LSB8 from the Power Tools menu. (If you find that LSB8 format does not work then try MSB and LSB in that order). When you are recording signals, record them in a logical fashion – such as temperature from 16 degC to 32 degC (or in Fahrenheit), fan speed in order etc. Only change one parameter for each button press and you will be easily able to see which bits/bytes change for that button press – thus simplifying the reverse engineering process. It is also advisable to enter a description into the Button column of History, for every signal, as an aid to deciphering the fields later on.

Once the option is selected a new window will appear. Click the LSB8 button. You should then see all of your signals with the Hex value split into 2 character columns (Hex values).

The bolded & underlined blue values, show changes from the previous signal. Thus it should be easy to work out which bits relate to the function or value that has changed. Hovering the mouse pointer over any value will pop-up a tooltip showing the binary representation of that Hex value.

Often a checksum will be located at the far right of the signal and should be easy to notice as the value should change for every signal.

The various buttons within the window provide the following functionality:

- Reload History – reloads the current History into the tool (in case new signals have been recorded).
- Copy Contents – places the displayed contents into your clipboard for pasting into other application such as spreadsheets. (HTML formatted)
- MSB Format – loads values in MSB format and may take some time as all signals must be fully analysed again.

- LSB Format – loads values in LSB format and may take some time as all signal must be fully analysed again.
- LSB8 Format – loads values in LSB8 format and may take some time as all signal must be fully analysed again.

**Important:** *It is assumed users are familiar with Hex and Binary notation and manipulation before attempting to use this tool. An understanding of complex AC signal formats and how checksums are calculated is also assumed.*

## HISTORY DECODE

This feature allows you to 're-analyse' 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 the same or new history file.

## CLEAN HISTORY

Similar to the option available in the Export window, you can automatically clean all decoded signals in the history grid. It is not possible to clean RAW signals. ANALYSIR corrects any timing errors in each signal contained within the on-screen history. It is usually a good idea to save your 'cleaned' signals in a new session history. As every received signal is always distorted, this feature will ensure greatly improved performance when sending these recorded signals. (CTL+Alt+H)

Menu ⇒ Power Tools => Clean History

## PRIORITY

We have noticed on some systems that issues can arise with older or slower PCs when using the USB IR Toy for sending. To work-around this we included an option within ANALYSIR to increase the runtime priority from Normal to High or Runtime. Under normal use both of these change the priority to High. To achieve runtime priority you must also 'Run ANALYSIR as Administrator'. Increasing the runtime priority of ANALYSIR should only be considered if experiencing issues with the USB IR Toy or if you are using a slow or sluggish PC. For all other users Normal priority is more than adequate.

## SOURCE MENU

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

## ANALYSIR

You may access the 'Source' menu as follows:

- Menu ⇒ Source => LEARNIR
- Menu ⇒ Source => A.IR Shield Nano
- Menu ⇒ Source => A.IR Shield rx
- Menu ⇒ Source => A.IR Shield Photon Serial
- Menu ⇒ Source => A.IR Shield Photon WiFi
- Menu ⇒ Source => Arduino (Default)
- Menu ⇒ Source => ESP8266 NodeMCU Serial
- Menu ⇒ Source => ESP8266 NodeMCU WiFi
- Menu ⇒ Source => Fubarino Mini
- Menu ⇒ Source => iTach Flex
- Menu ⇒ Source => MSP430 LaunchPad
- Menu ⇒ Source => Photon Serial
- Menu ⇒ Source => Photon WiFi
- Menu ⇒ Source => PSOC4
- Menu ⇒ Source => Raspberry Pi
- Menu ⇒ Source => Teensy
- Menu ⇒ Source => USB IR Toy

Before you can record signals using ANALYSIR, you must select the correct 'Source' device. The default is **Arduino** and the source will always be reset to Arduino, whenever a fault or reset is detected with any source device. Always remember to re-select your source device after any fault. The baud rate is now 115,200 bps for all platforms connected over Serial. However, the default baud rate can be overridden in the 'ini' file, for unsupported devices.

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 and all network connected devices as it is not applicable)

The **MSP430 LaunchPad** option supports only the Tiva C and MSP430F5529 LaunchPads at this time and the set BAUD rate is 115,200 bps. Note: The MSP430 option is classed as Beta. Except on the Tiva C, it also does not support carrier frequency measurement at this time, due to issues with clock/DCO accuracy.

Both **LearnIR** and **A.IR Shield** devices (designed & sold by AnalysIR) support both receiving and sending of IR out of the box. They also report carrier frequency measurement as standard and are the premium devices for use with AnalysIR. LearnIR is a sophisticated IR learning device (with its own dedicated application) and A.IR shield is specifically designed for use with AnalysIR and can also be configured for use with IRremote &

IRLib or other Arduino sketches. The A.IR Shield rx does not support IR sending. The A.IR Shield Photon devices are designed for use with the Photon from Particle and compatible devices. (Note: at the time of writing Send IR from AnalysIR using Photon devices is in development)

The **iTach Flex WiFi** available from Global Caché can be used to record and send IR signals via WiFi over the LAN.

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. You will be instructed to install a copy of IRToy.exe (Available from the vendor's website) into the ANALYSIR directory before you can send with the USB IR Toy.

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. Next select the correct source device from the Menu.
- Wrong IP address or Port configured for network connected device. Please correct the configuration. It is possible to test your connection to any network device by using telnet from the command prompt of your Windows PC. A command similar to "*telnet 102.168.1.123 25*", if successful you should see 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 or context 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 10 seconds and reconnect. Finally, repeat the above and if this doesn't solve the problem, you may have to re-start AnalysIR and/or 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 carrier frequency.
- Something messed up during installation. In this case just restart your PC and try again.

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



## SHOW CARRIER FREQUENCY

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

Menu ⇒ Source ⇒ Show Carrier Frequency

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

The Carrier Frequency is reported automatically for the recording device (e.g. LearnIR, A.IR Shield, Arduino) and displayed in the taskbar, provided you have an IR Learner connected to your circuit.

LEARNIR and A.IR Shield report carrier frequency with every signal recorded and it is displayed & recorded automatically.

For a source device to report carrier frequency it must have a sensor capable of measuring it. Normal IR receivers cannot perform this task.

## DETAILS TAB

A new feature has been introduced to display the individual fields of protocols in the Details Tab. Support for this is initially limited to the RC5, NEC, RECS80, Milestag1 and XMP protocols, with additional protocols to be added in some future release. Other protocols will display only basic information and all signals will display carrier frequency, if recorded or available. You can also activate this using the menu as follows:

Menu ⇒ File ⇒ Details or CTL+D

The count of marks and spaces in each signal is also displayed for each signal along with the protocol name as part of the basic information mentioned above.

*Note: It is necessary to click on the Details Tab, use the menu or keystroke above to update the details displayed, as it is not an automatic update. The Details Tab is also updated by clicking on a row in the History view for a supported signal or navigating with UP/DOWN arrow keys on the keyboard. (Listed above)*

## 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



You can check for updates to AnalysIR, by clicking the 'Check for on-line Update now' menu option. AnalysIR will then make a web query to the AnalysIR website and report your current version and latest available versions of AnalysIR. Clicking 'Yes' on the pop-up window will open a browser link to the AnalysIR download page, where you can get the updated version.

The pop-up window may also inform you about available 'development releases' and how to obtain a copy. 'Dev' releases may have additional protocols, fixes or new features not yet within the latest #official' release. The associated icon will change from 'Information' to 'Warning' if there is a newer official or 'dev' release available.

You can also instruct AnalysIR to automatically check for updates, every time you start AnalysIR. Simply select the 'Check for Update at Start-up' menu option. When it is checked, AnalysIR will check for updates every time it starts. You can always un-check this option to stop these automatic checks. Online access is required to check for updates.

The Help menu also provides options to visit several online services, such as AnalysIR's website, Shop, contact page and our IRforum.

Tooltips, with helpful text, will pop-up as you hover the mouse over certain controls of AnalysIR. Tooltips can be enabled or disabled by checking or unchecking the 'Show tooltips' menu option.

## SERIAL PORT SELECTION

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.

Right clicking on this control will display a context menu for the serial port.

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

## SERIAL PORT INDICATORS

There are 2 visual indicators related to the Serial Port. The first is a small square box to the left of the COM port drop down box. It is colour coded as follows:

- Red: the selected serial port is not open and not being

used by ANALYSIR. ANALYSIR attempts to open this port every few seconds and if it remains red for an extended period, it means that the device is not plugged in or there is an issue of some sort.

- Green: the selected serial COM port is opened by ANALYSIR and is operating normally. You must also ensure that the correct source type is selected.
- Light Green: the selected COM port is not opened by ANALYSIR and is operating normally via a network connection to an RPi, iTach or Photon device.
- Grey: The serial port has been disabled by the user, which means that ANALYSIR does not control the serial port.

Right clicking on this control will display a context menu for the serial port.

The second indicator is a small dot in the bottom right of the main window, which indicates if serial data is being received from the device. The default colour is grey and it will briefly flash red, every time data is received over serial from the device.

Right clicking on this control will display a context menu for the serial port.

## CUSTOM IR PROTOCOLS

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). 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.

Currently, there are a large number of protocols assigned under 'Custom'. To see a full list, check the properties, rules or '.ini' file. To enable decoding for any protocol in the custom list, you must check the custom menu item in the protocols menu.

## WINDOW RE-SIZING

ANALYSIR contains a feature to allow dynamic re-sizing of the main window of ANALYSIR. Both vertical and horizontal resizing is supported. The maximum window size supported is 1980 x 1200 pixels. The initial default window size is 1024 x 768 pixels. Any changes you make to the window size will be remembered by ANALYSIR between sessions.

For users with displays larger than the default size, this provides a greatly enhanced view of longer signal traces and the other on-screen information. If not, you can always make use of the existing zoom and scrollbar features.

If the window is resized to less than the default size, scrollbars will appear, within the window, otherwise there will be no (window) scrollbars visible.

If you have a display larger than the max supported size

mentioned above, it may be possible to use a larger size system by editing the setting in the '.ini' file and restarting ANALYSIR, as follows:

[WINDOWSIZE]

#default width is 1024 & default height is 768

Max\_Width = **1980** # maximum width of ANALYSIR window in pixels (Larger than 1980 may work, but unsupported)

Max\_Height = **1200** # maximum height of ANALYSIR window in pixels (Larger than 1200 should work, but unsupported)

Remember, if you change the maximum window size it is not 'officially' supported.

To change the maximum window size, edit the Max\_Width & Max\_Height parameters above as required.

## 4. LEARNIR Setup

---

LearnIR is really plug & play. Ensure you have the latest LearnIR firmware loaded (default). Then simply connect LearnIR to the USB to Serial adapter & connect the USB cable to the PC. Then select the COM port from the dropdown box in AnalysIR and only then select LearnIR from the Source menu.

Menu ⇒ Source ⇒ LearnIR (CTRL+ALT+1)

## 5. A.IR Shield Setup

---

### A.IR SHIELD NANO

The A.IR Shield Nano is really plug & play. Ensure you have the correct AnalysIR firmware loaded (default) and have the jumpers at their default settings. Then simply connect the shield to the Nano & connect the USB cable to the PC. Select the COM port from the dropdown box in AnalysIR and only then select A.IR Shield Nano from the Source menu. Please refer to the detailed instructions provided with your device.

Menu ⇒ Source ⇒ A.IR Shield Nano (CTRL+ALT+2)

### A.IR SHIELD PHOTON

The A.IR Shield can be interfaced by Serial/USB or over WiFi. The provided firmware works with whichever option is selected on the Source Menu. Please refer to the detailed instructions provided with your device.

Menu ⇒ Source ⇒ A.IR Shield Photon Serial (CTRL+ALT+4)

Menu ⇒ Source ⇒ A.IR Shield Photon WiFi (CTRL+ALT+5)

### A.IR SHIELD RX

The A.IR Shield Rx is designed for use with AnalysIR for rx only. It is a very compact solution. Please refer to the detailed instructions provided with your device. A.IR shield Rx can be configured to work with most MCU development platforms.

Menu ⇒ Source ⇒ A.IR Shield rx (CTRL+ALT+3)

## 6. iTach Flex WiFi Setup

---

The iTach Flex is manufactured by Global Caché. The set-up is relatively simple.



Follow the steps below:

- Before starting, make sure you can operate your device using the iLearn utility, available from the manufacturer. Do not proceed until you have completed this step. If you need support on this, please contact the manufacturer or supplier directly.
- Find the IP address of the device on your local network. For details contact the manufacturer or supplier directly.
- Update the AnalysIR.ini file to configure the network settings for this device. Make sure to edit the correct file via Menu ⇒ File ⇒ Explore.
- Once your device is connected to the network, you can access it from AnalysIR via Menu⇒Source⇒iTach Flex. (CTRL+ALT+7)  
However, make sure you have previously closed all Global Cache applications (iLearn, iTest etc) that may have been connected to the iTach device.
- The name "iTach Flex" should appear in the status bar and the status box should turn Light Green.
- Simply point your remote at the device and you should see signals appearing in the trace of AnalysIR.
- Tip: make sure you are not connected to iTach Flex from any other application (e.g. iLearn), as only one connection is allowed for Learning, at the same time.
- If you experience any issues, please ensure you verify the correct operation of your device with the iLearn utility, before contacting AnalysIR support.

### INI File Settings:

```
#
IPAddress_iTachFlex=000.000.000.000 #Change this to IP address of your Photon
PORT_iTachFlex=4998 #Change this to Port number used on your
iTach Flex (It is always 4998, unless you have a special configuration)
#WAIT_NETWORK_iTachFlex=750 #Leave this line commented out, unless
advised otherwise
.....
```

Normally, you should only have to change the IP address to the IP address of your iTach Flex device. The PORT number is fixed at 4998 for the iTach Flex and the Wait value should not be altered unless instructed to do so by support.



## 7. Arduino Setup

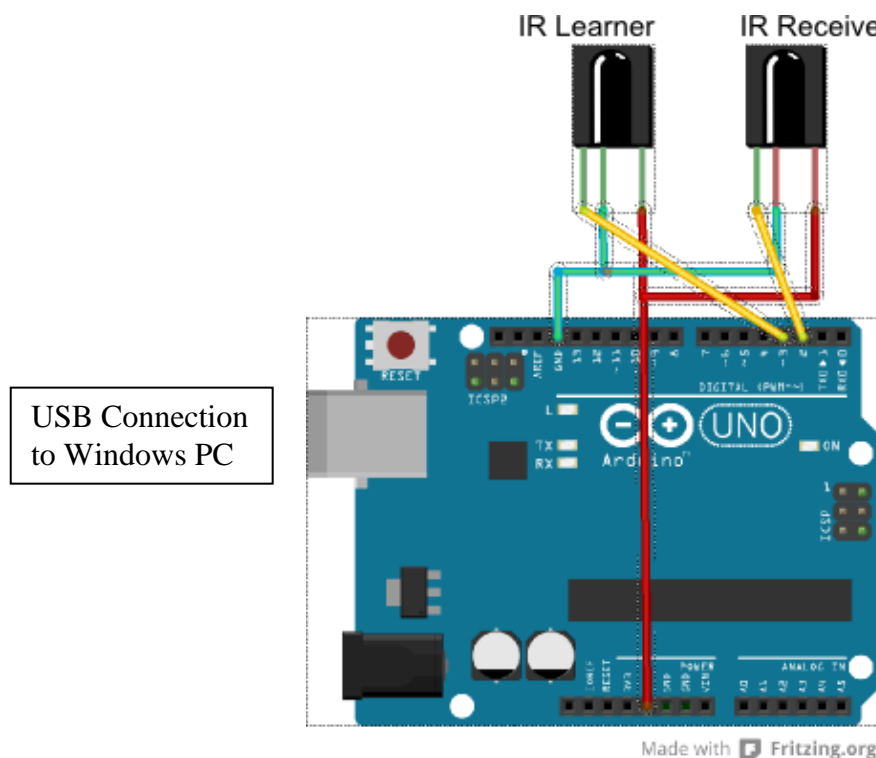
This provides an example circuit layout for the Arduino based IR receiver:

### ❖ "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 and your APPDATA folder. (Menu ⇒ File ⇒ Explore) 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. On 5V systems connect the power of the IR receivers to 5V & on 3V3 systems connect the power of the IR receivers to 3V3. (Do not mix voltage levels)

The IR learners we recommend are the TSMP58000 from Vishay. These pass through the carrier IR signal to the Arduino, complete with carrier. 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 carrier 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 carrier frequency and in most cases it is possible to guess the carrier frequency for the majority of common protocols.

The IR receiver we use and recommend is the TSOP4038 or TSOP58038 from VISHAY which are actually light barrier sensors 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 carrier 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](http://www.vishay.com/docs/49845/49845_sg2145.pdf)

Arduino models verified working with ANALYSIR:

- Uno
- Leonardo
- Nano
- Pro-Micro
- Micro
- Duemilenova
- Mega1280
- Mega2560 (untested, but should work)
- Due
- Yún Serial over USB(CDC) OK

The pins assigned are as follows:

**Arduino (All supported models)**

Function	Pin
IR Receiver:	2
IR Learner:	3

## 8. Raspberry Pi Setup

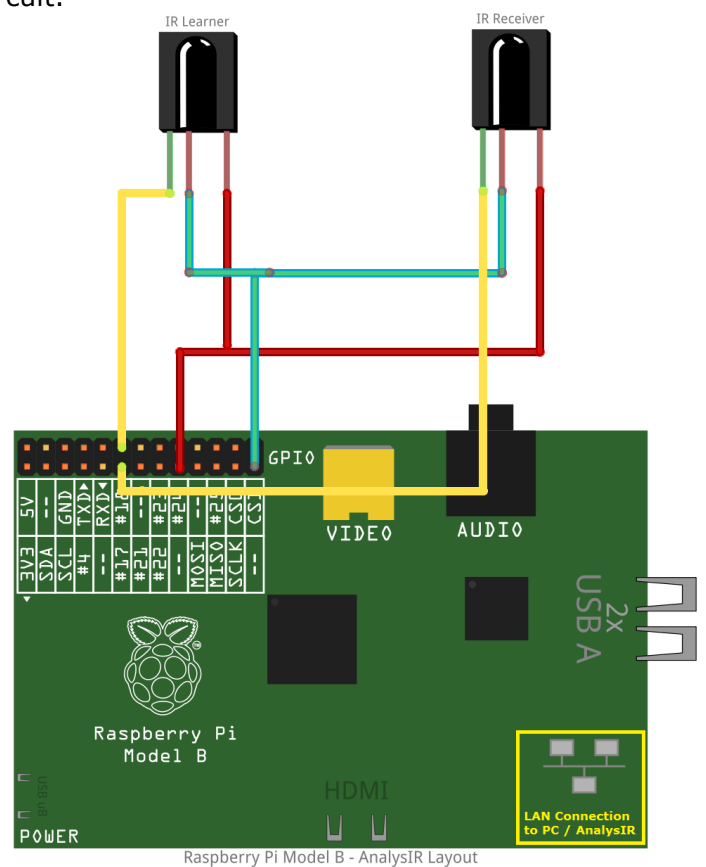
This provides an example circuit layout for the Raspberry Pi based IR receiver:

### ❖ "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.



fritzing

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.

*Note: The Model B+ has not been tested yet, but should work in an identical fashion. We are happy to provide support for the B+.*

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 Learner is connected to Header Pin 12
- The signal output from the IR receiver is connected to Header Pin 11.
- We have designated Header Pin 11 for use with the LIRC example we provided which should be connected to the signal output of an IR receiver (**NB: This is different from the original/early pin-out we used. Pin 13 is no longer used.**)

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 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 and only supported approach, make sure that your IR receiver signal out pin is connected to the correct header pin used by LIRC, which could be different from the one we have designated.

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). However, be aware that the IR Receiver is now connected to pin 11 (not pin 13). 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 level 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, complete with carrier. 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 carrier 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 carrier 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 TSOP4038 or TSOP58038 from VISHAY which are actually light barrier sensors 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 tuned to particular protocols and carrier 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](http://www.vishay.com/docs/49845/49845_sg2145.pdf)

#### **RPi models compatibility:**

Raspberry Pi – **Model B** (Confirmed. Use of other GPIO pins possible)

Raspberry Pi – **All other models** (except compute module)  
*Unconfirmed & unavailable for testing, pin-out may vary from Model B diagram. However, functionality should be similar as long as you pay close attention to pinouts and install LIRC correctly.*

#### **Troubleshooting:**

- Ensure LIRC is stopped before starting eh mode2 command.
- Increase the default network timeout in the AnalysIR.ini file to 2000 or 5000 if you are having network issues or your network is busy/noisy.
- You can quickly test the setup by issuing "Telnet <RPI IP address> <RPI port number>" in a windows command window, to see if data is being received from the RPi. Then press buttons on your remote and you should see data arriving in your telnet window. If you see nothing then you have a problem with your RPi setup or network or firewall.

Make sure you close AnalysIR before issuing the Telnet command. If you do see the data arriving then just close the telnet window and restart AnalysIR and select RPi from the source menu and start pressing buttons on your remote. You should see signals appearing in the AnalysIR window.

## 9. USB IR Toy Setup

---

The USB IR Toy requires minimal setup provided 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 via Menu⇒Source⇒USB IR Toy (or CTRL+ALT+G)).

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.

For sending IR you will be instructed to place a copy of IRTToy.exe into the ANALYSIR directory, on first use only.

## 10. Teensy Setup

Teensy 3.x is now supported. For earlier versions of Teensy, just follow the instructions for the equivalent version of Arduino.

You will require an Arduino IDE which supports Teensy (or equivalent). See the vendor's website for instructions.



The pins assigned are as follows:

### Teensy 3

Function	Pin
IR Receiver:	2
IR Learner:	3

Note: Verified on teensy 3.1 only. But according to the supplier both 3.0 & 3.1 should be identical for our purposes.

*Teensy 2.0 and Teensy+++ 2.0 have not been tested yet, but experienced users should be able to figure out the correct adjustments to make. You can contact us for support, which we are happy to provide.*

*When connecting your IR Receiver & IR Learner, please observe the pin-outs from the datasheet and use the correct voltage level for Vcc (NB: Do not mix 5v & 3.3V levels)*

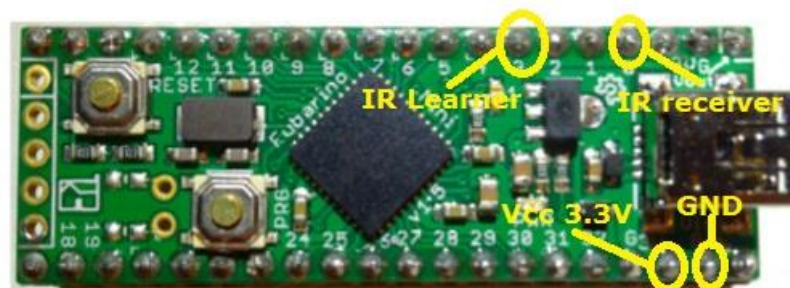
In addition, please read the setup information provided for Arduinos.



## 11. ChipKit – Fubarino Mini

Fubarino Mini is now supported once the provided firmware is uploaded.

You will require an MPIDE IDE which supports Fubarino (or equivalent). See the vendor's website for instructions.



The pins assigned are as follows:

### Fubarino Mini

Function	Pin
IR Receiver:	0
IR Learner:	3

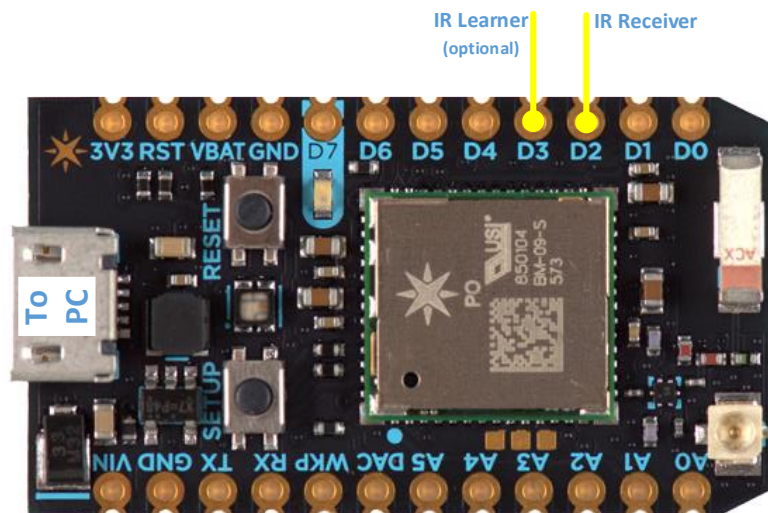
*Other ChipKit platforms have not been tested yet, but experienced users should be able to figure out the correct adjustments to make. You can contact us for support, which we are happy to provide.*

*When connecting your IR Receiver & IR Learner, please observe the pin-outs from the datasheet and use the correct voltage level for Vcc (NB: Do not mix 5v & 3.3V levels)*

In addition, please read the setup information provided for Arduinos.

## 12. Particle – Photon

The Photon is now supported via serial over USB connection to PC and also via WiFi. The RedBear DUO is also supported as it has the same hardware and pinout. (Follow instructions for Photon when using the DUO). Don't forget to checkout our **A.IR Shield photon** which is an advanced plug and play shield for the Photon & AnalysIR. The A.IR Shield Photon is supplied with its own set of instructions and has its own selection options in the source menu. You will require the Photon web IDE. See the vendor's website for instructions.



The pins assigned are as follows:

### Photon

Function	Pin
IR Receiver:	D2
IR Learner:	D3

*Note: Separate instructions for the **A.IR Shield Photon** are provided with purchase of the module, with sending support for AnalysIR*

*Other 'Particle' platforms have not been tested yet, but experienced users should be able to figure out the correct adjustments to make to easily port the code. You can contact us for support, which we are happy to provide.*

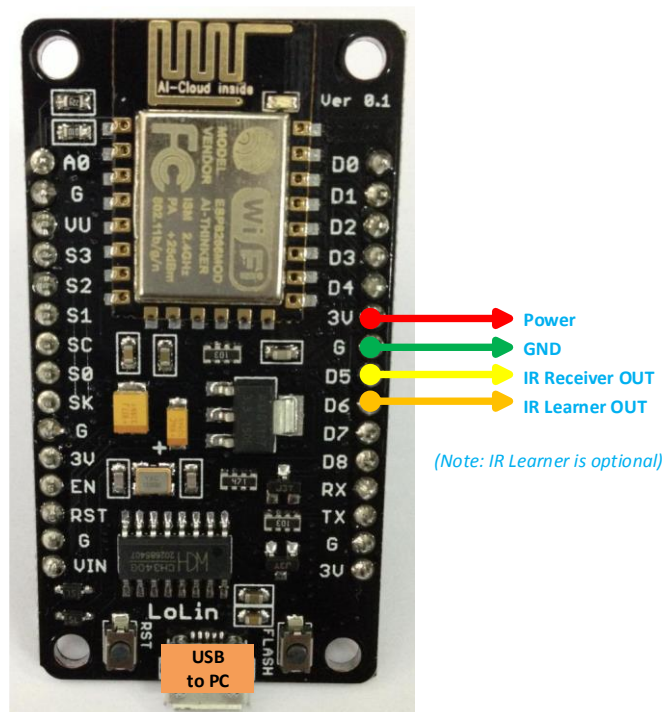
*When connecting your IR Receiver & IR Learner, please observe the pin-outs from the datasheet and use the correct voltage level for Vcc (NB: Do not mix 5v & 3.3V levels). When making connections please ensure all power is removed.*

*In addition, please read the setup information provided for Arduinos.*

Chapter  
13

## 13. ESP8266 – NodeMCU

The ESP8266 is now supported via serial over USB connection to PC and also via WiFi. Use the Arduino IDE with NodeMCU support for ESP8266. For other ESP8266 variants you may need to port the AnalysIR firmware.



The pins assigned are as follows:

### ESP8266 / NodeMCU

Function	Pin
IR Receiver:	D5
IR Learner:	D6

*When connecting your IR Receiver & IR Learner, please observe the pin-outs from the datasheet and use the correct voltage level for Vcc (NB: Do not mix 5v & 3.3V levels). When making connections please ensure all power is removed.*

In addition, please read the setup information provided for Arduinos.

As there are several variants of this platform, you may need to make some adjustments to firmware, if different.

## 14. TI LaunchPad Setup

The TI MSP430 F5529 support is currently in beta, mainly because of inconsistencies between different versions of the Energia IDE. For simplicity we have included support for LaunchPads in the default sketch.

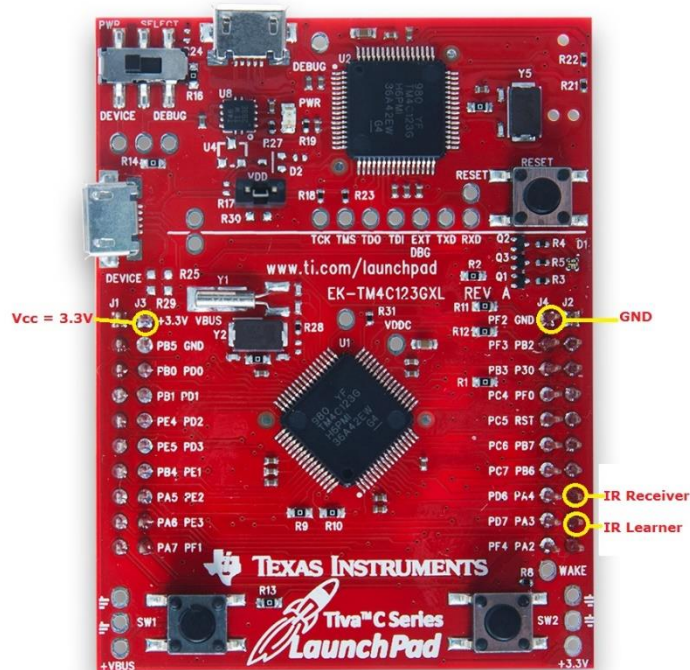
We have tested with the Tiva C & MSP430F5529. The Tiva works very well & because of oscillator related issues, the MSP430F5529 does not support measurement of carrier frequency.

In our latest testing we noted that using Energia release energia-0101E0013 broke the operation, for as yet unknown reasons. So to get these platforms working you will need to use energia-0101E0011 as the IDE.

The pins assigned are as follows:

### Tiva C

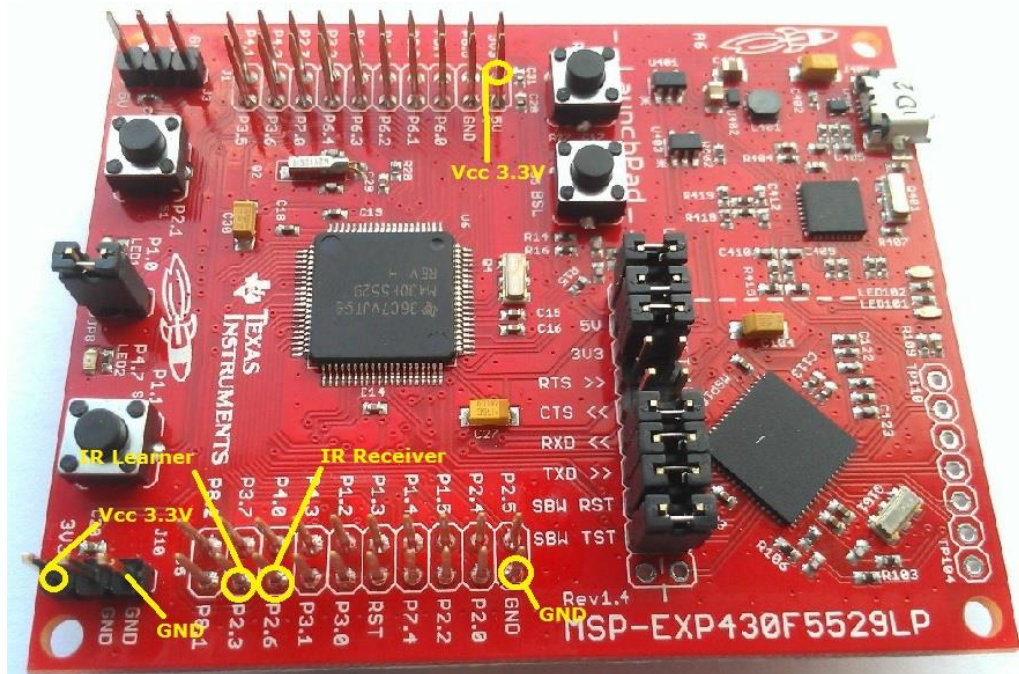
Function	Pin
IR Receiver:	PA_4
IR Learner:	PA_3



Connections for Tiva C LaunchPad

## MSP430F5529

Function	Pin
IR Receiver:	P2_6
IR Learner:	P2_3 (not operational)



Connections for MSP430F5529 LaunchPad

*When connecting your IR Receiver & IR Learner, please observe the pin-outs from the datasheet and use the correct voltage level for Vcc (NB: Do not mix 5v & 3.3V levels)*



Chapter  
15

## 15. PSOC4 Setup

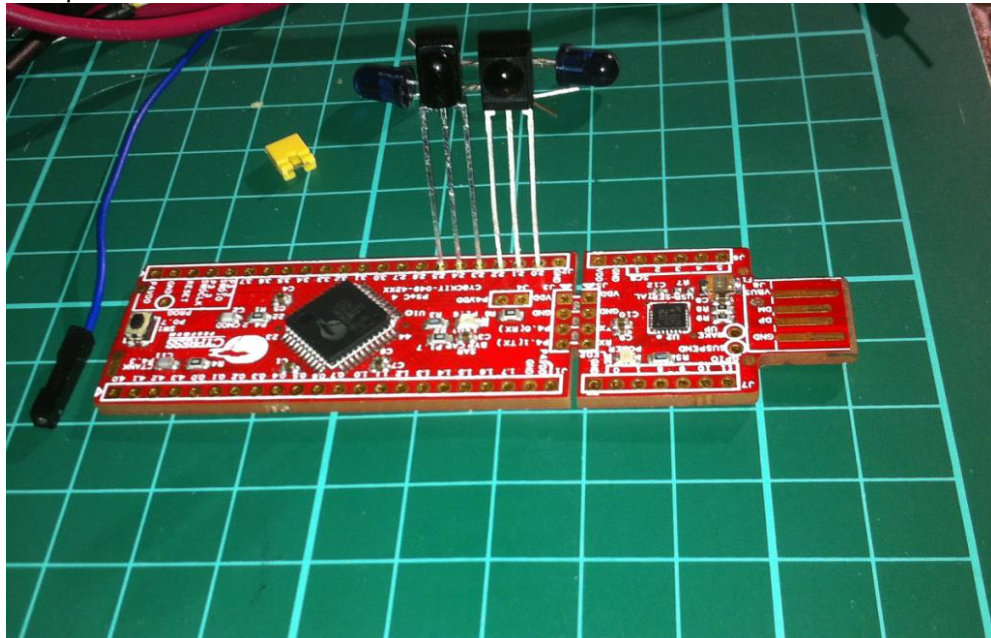
---

PSOC4 is now supported. Specifically, the CY8CKIT-049-42xx PSOC 4 Prototyping Kit is supported. However, experienced PSOC users should have little problem converting to other PSOC platforms, provided they can integrate a serial to USB connection at 115,200 bps and circa 2k SRAM available (minimum 1k).

We have tested with PSoC Creator 3.0 only. A sample, Bootloader based, Creator project is provided for recording Infrared signals with ANALYSIR. Another example project is available which demonstrates sending of signals using the same PSoC with the PSOC4 batch export facility of ANALYSIR.

**IMPORTANT:** Users should be familiar with PSOC & Creator before attempting to use them for ANALYSIR. In particular, you should be familiar with Bootloader projects using this platform. ANALYSIR, should not be your first project using PSOC or Creator.

The pinouts used for the IR receiver & IR Learner are as follows:



## Connections for PSOC4

In the photo above, the 'IR Learner' is on the left and the 'IR Receiver' is on the right.

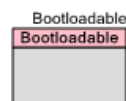
PSOC Pin PSOC4	IR receiver Pin TSOP4038 <sup>5</sup>	IR Learner Pin TSSMP58000 <sup>4</sup>
<b>P2.0</b>	Vcc	
<b>P2.1</b>	GND	
<b>P2.2</b>	Signal Out	
<b>P2.3</b>		Vcc
<b>P2.4</b>		GND
<b>P2.5</b>		Signal Out

### Notes:

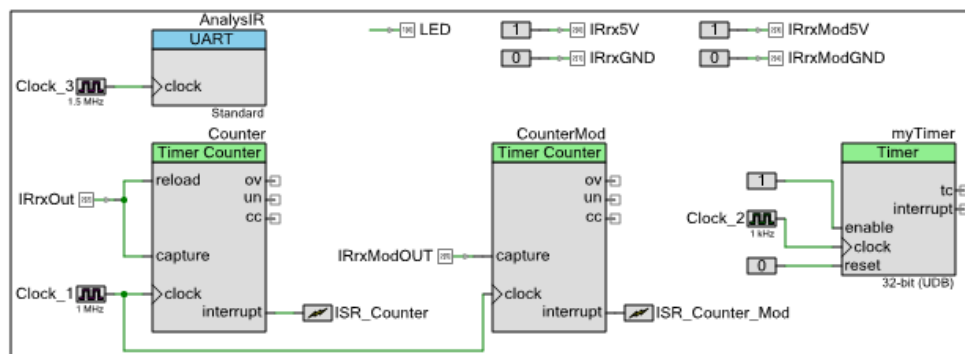
1. Do not mix 3.3V & 5V logic levels.
2. PSOC IO pins are used in this example to provide Vcc & GND to the IR devices, for convenience, as the current required is minimal and well within the capability of the IO pins.
3. Experienced PSOC users will know how to wire up differently & how to change the pin assignments within the firmware.
4. The IR Learner is optional and not required unless you want to measure & report the carrier frequency of received signals.
5. It is possible to replace the TSOP4038 with a standard IR receiver to match the signals you are try to analyse.

When connecting your IR Receiver & IR Learner, please observe the pin-outs from the device datasheet, adjust accordingly and use the correct voltage level for Vcc (NB: Do not mix 5v & 3.3V levels. Not all IR receivers have the same pin-outs)

## Bootloadable AnalysIR PSOC4



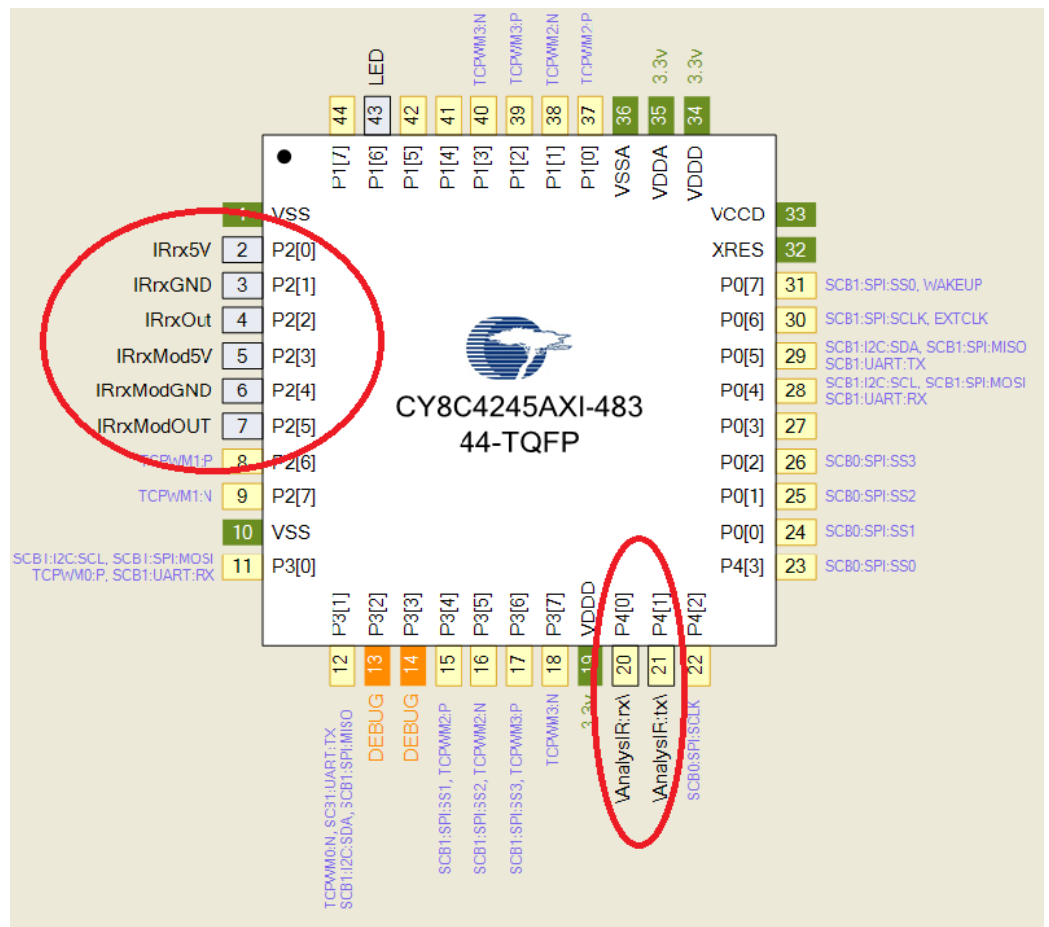
The Bootloadable component links the application to the Bootloader project's .hex file



### Overview of PSOC project for ANALYSIR

Below we show the pin-out assignments from within PSOC Creator.

## Pinout Assignment



You can see our blog post about PSOC support, here:  
<https://www.AnalysIR.com/blog/2014/11/04/analysir-supports-psoc-4-cypress/>



## 16. 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

Alternatively, you can use the help menu:

Menu ⇒ Help ⇒ Visit IRforum

Menu ⇒ Help ⇒ Contact us (via website)

Or directly via:

**[support@ANALYSIR.com](mailto:support@ANALYSIR.com)**

**[www.ANALYSIR.com](http://www.ANALYSIR.com)**

*Then follow the menu links to support or go directly via:  
<https://IRforum.ANALYSIR.com/>*

## 17. Appendix A – Sample Firmware & Raspberry Pi code

---

The sample code for the IR receiver device is included in your installation package or APPDATA (via Menu ⇒ File ⇒ Explore) directory after running AnalysIR for the first time, with a valid Licence key. It should work on most standard Arduino systems with on-board 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 only.

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 1.6.5 IDE, MPIDE & Energia 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.

There is no additional code required for LEARNIR, A.IR Shield or USB IR Toy devices, which should both work out of the box. Please ensure you have V22 firmware installed on your USB IR Toy. All LEARNIR firmware versions are supported.

A.IR Shield comes pre-loaded with the required firmware for use with AnalysIR. However, as it can also be re-programmed as an Arduino, you should ensure that the native A.IR Shield firmware is loaded with the correct jumper settings when using with AnalysIR. *(Source code is provided and is very easy to customise, for other uses, using the Arduino IDE)*

# Chapter 18

## 18. Appendix B – Configuration File, ANALYSIR.ini

This file, which is located in your 'APPDATA' directory and named AnalysIR.ini, contains all of the important configuration details for your installation. You can find a list of parameters via:

Menu ⇒ File ⇒ Properties

Below we highlight some of the important settings in Configuration file.

Item	Description	Format
[.....]	Section Heading for Protocols	n/a
[CUSTOMnn]	Section Heading for a custom protocol definition, where <b>nn</b> is a number starting at 1.	Text
[SERIALPORT]	Section Heading denoting the start of the parameters for the Serial Port	Text
[WINDOWSIZE]	Section Heading denoting the settings for maximum window width and height.	Text
[NETWORK]	Section Heading denoting the start of the parameters for the network connection	Text
<b>MARK</b>	Duration of the Mark	µSecs
<b>MARK0</b>	Duration of the Mark for a zero bit	µSecs
<b>MARK1</b>	Duration of the Mark for a one bit	µSecs
<b>SPACE</b>	Duration of the Space	µSecs
<b>SPACE0</b>	Duration of the Space for a zero bit	µSecs
<b>SPACE1</b>	Duration of the Space for a one bit	µSecs
<b>HEADER</b>	Duration of the Header Mark	µSecs
<b>HEADERSPACE</b>	Duration of the Header Space	µSecs
<b>CARRIER</b>	The carrier frequency for information only.	Hz
<b>BITS</b>	The number of bits in this protocols. For Custom protocols this can be set to zero, to indicate any number of bits is OK.	numeric
<b>DELTA</b>	The default tolerance parameter for this protocol. For example a value of 200 means that ANALYSIR will accept signal timings that are within +/- 200	µSecs

	of the configured time for that protocol.	
<b>TOGGLEMASK</b>		
<b>ALTNAME</b>	An alternative name for this protocol. Normally the protocol name is Taken from the Section Header. For Custom protocols the Altname is used as the actual name.	Text
<b>GENERIC</b>	If this is set to true then this protocol definition uses the ANALYSIR automatic decoding syntax.	True Or False
<b>SYNTAX</b>	This defines the syntax used by the ANALYSIR automatic decoding algorithm.	Text
<b>MARKS</b>	The list of Mark timings ( $\mu$ Secs) used by the ANALYSIR automatic decoding algorithm.	Text
<b>SPACES</b>	The list of Space timings ( $\mu$ Secs) used by the ANALYSIR automatic decoding algorithm.	Text
<b>SIGNALX</b>	Internal use only.	$\mu$ Secs
<b>SIGNALY</b>	Internal use only.	$\mu$ Secs
<b>COMPORT</b>	The COM port to which your recording device is attached (e.g. COM4). <u>Alternatively</u> , a 2-part value with the COM port followed by a ':' and followed by the type of device. e.g. COM50:LEARNIR COM51:ARDUINO COM52:USB_IR_Toy COM53:MSP430 COM54:RPi Using this approach means that ANALYSIR will automatically assume that a particular device is attached to the COM port. Otherwise it will assume the default of Arduino.	Text
<b>COM_PORT_IGNORE</b>	A comma separated list of COM ports to be ignored by ANALYSIR. This feature is required on some systems that do not behave as expected.	Text
<b>COM_PORT_INCLUDE</b>	A list of COM ports to be included always in the ANALYSIR drop down list, even if they are normally ignored by ANALYSIR or not visible/present.	Text
<b>CUSTOM_BAUD_RATE</b>	Override the default baud rate of 115,200 bps. Valid values are 9600,19200,38400,57600 & 115200.	Numeric
<b>IPADDRESS_RPI</b>	The IP address of the Raspberry Pi on the network. e.g. 192.168.1.123	Text
<b>PORT_RPI</b>	The port number that the RPi has been configured to listen on.	Numeric
<b>WAIT_NETWORK_RPI</b>	Sets a custom network timeout.	Numeric
<b>IPADDRESS_iTachFlex</b>	The IP address of the iTach Flex WiFi on the network. e.g. 192.168.1.70	Text

<b>PORT_iTachFlex</b>	The port number for the iTach Flex – always 4998.	Numeric
<b>WAIT_NETWORK_iTachFlex</b>	Sets a custom network timeout.	Numeric
<b>IPADDRESS_YUN</b>	Reserved for future use only	Text
<b>PORT_YUN</b>	Reserved for future use only	Numeric
<b>WAIT_NETWORK_Yun</b>	Reserved for future use only	Numeric
<b>IPAddress_PHOTON</b>	The IP address of the Photon on the network. e.g. 192.168.1.123	Text
<b>PORT_PHOTON</b>	The port number that the Photon has been configured to listen on.	Numeric
<b>WAIT_NETWORK_PHOTON</b>	Sets a custom network timeout.	Numeric
<b>IPAddress_ESP8266NodeMCU</b>	The IP address of the ESP8266 on the network. e.g. 192.168.1.123	Text
<b>PORT_ESP8266NodeMCU</b>	The port number that the ESP8266 has been configured to listen on.	Numeric
<b>WAIT_NETWORK_ESP8266NodeMCU</b>	Sets a custom network timeout.	Numeric
<b>Max_Width</b>	The maximum width of the main ANALYSIR window in Pixels.	Numeric (Pixels)
<b>Max_Height</b>	The maximum height of the main ANALYSIR window in Pixels.	Numeric (Pixels)

#### Notes:

- The file is in a Windows text file format, for easy editing.
- All text on any line after and including a '#' is ignored.
- Spaces & Tabs are ignored
- [xxxx] signifies a Section heading
- AAAA = BBBB signifies a parameter setting.
- Unknown or malformed Section headings/parameters are ignored.
- Parameters belong to the last correctly detected Section headings.