

TB 04 - Page 1 of 2

Irisys People Counters

Technical Bulletin



26th September 2013.

Issue: **04**

Introduction

Welcome to the forth technical bulletin. If you would like to see anything in particular in these bulletins then please let me know. Mike.



Bluetooth Setup Module

Beta units of the new Bluetooth setup module have now gone out to a small number of customers for testing and subsequent feedback. Once we receive feedback (and it is positive) we will most likely embark on a production run of units. If you are interested in purchasing a Bluetooth module in the future please let us know, as the more positive responses we get, the more likely it is that we will do a production run. Just so you know, the Bluetooth module is a local setup option to be used in place of the IWC3052 setup module or IWC3062 setup cable - yes it can be used on all 3000 series indoor counter variants (not outdoor)! As with the setup cables that it can substitute for, the Bluetooth module is for configuration only - not for data extraction.



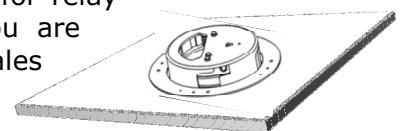
Tailgate Detector Installation Documentation Update

The "Tailgate Detector Installation & Setup Guide" document has been recently updated and is now available from the Irisys partner portal. This will be of interest to those partners regularly purchasing and/or installing the IRS2100-60 tailgate detector, and for those others who are curious about the opportunities it could present. The update brings together everything you need to know about the IRS2100-60 detector in one place, from; whether it is suitable for a particular application - even down to specific door types - through to where to install, how to install, how to configure, and how to integrate with access control systems. Go to the partner portal and search for document number IPU40097.



Recess mount kits for 3000 series Counters now Available

We now have two recess mount kits available for those customers who want to hide counters up in the ceiling. Two kits are available; the IWC3096 is for relay units, and the IWC3097 is for the (deeper base) IP units. If you are interested in these kits then please contact your Irisys sales representative for pricing details.



Irisys Harvester Software Issue

One issue to report for those of you using the 'Harvester' software to collect count data from the 3000 series IP units (e.g. IRC3010, IRC3030 etc.)...

Microsoft has changed the Greek time zone (GTB) in a recent 'Hotfix' update. This only affects customers with counters installed in, and configured with, the Greek time zone locale who are also using an older version of Harvester. In older versions, Harvester used the locale setting retrieved from the counter configuration in order to identify the time zone that the counter is in and therefore correctly log data in the local time. With this recent Hotfix update, data now retrieved from a counter configured with the Greek time zone locale will be recorded with a 2 hour offset.

Only counters configured with the Greek time zone locale will be affected - those installed elsewhere are not affected. It will also not affect any Harvester installations that use the time zone drop down in the counter's settings. This will be fixed in a future release of Harvester, but if this issue affects you now, the workaround is to upgrade to the latest version of Harvester which supports the time zone dropdown in the counter rather than the locale edit field, and to specify the correct time zone in there. On older counter installations this may also require a firmware upgrade to the counters in order to add the time zone drop down functionality if not present.

© 2013 InfraRed Integrated Systems Limited

No part of this publication may be reproduced without prior permission in writing from InfraRed Integrated Systems Limited. This document gives only a general description of the products and except where expressly provided otherwise shall form no part of any contract. From time to time changes may be made in the products.

Telephone: +44(0)1604 594200 Fax: +44(0)1604 594210 e-mail: support@irisys.co.uk web: www.irisys.co.uk

TB 04 - Page 2 of 2

IP Counter Integration Quick Guide

For those who have not yet moved across from relay units and into the world of remote setup and logged data that is provided by the IP range of units, I thought it would be a good idea to explain what you need to do, as the thought of writing new software to interface with the units can be daunting to some. If you really don't want to spend the time and money writing your own software, then the Irisys 'Harvester' data collection package (and optional 'Reporting' module) does provide a very basic system that you can run with. But for purpose built software with your own branding you need to write your own software. This is actually a lot easier than it sounds and anyone with a basic knowledge of programming should be able to manage it.

We provide four different APIs which allow access to the counter functions. You should choose the API version based on the operating system that you want your software to run on and the development platform that you want to write it in. Available APIs are .Net, Win32, Linux and Java, and the same functions are available on one.

Essentially the IP units provide two different methods of accessing data. You can retrieve the current instantaneous values at any time for (almost) real-time data, or you can pull the historical logged count data which is periodically stored inside the counters own flash memory. Data can be logged at any time interval, between every minute and every hour, but is typically logged every 15 minutes. To access the current counts the API function 'GetCurrentCount' must be used, and the currently held register (line) values will be returned one after the other. To access the logged count data 'GetCounts(StartTime, EndTime)' must be used, and again the logged values will be returned along with the date and time of each log, one after the other between the start and end times specified. An alternative 'GetLastNCounts' function is also available in order to pull the last 'n' log entries, but usually the 'GetCounts' function is used unless the data is required real-time, in which case the 'GetCurrentCount' function is used. But if the network goes down for any reason, the 'GetCounts' function must also then be used to pull missing data logged during the offline period.

The available logging memory depends on the number of count lines being logged and how often, obviously the more lines being logged and the more often they are written to: the quicker they will get fill up the available memory and start over writing the older data. To give you an idea, at 15 minute logging of simple IN and OUT counts (2 lines) the logging will last over 45 days before data starts being lost. This is more than enough time to fix any network issues and retrieve the data.

Additionally, logged count data is stored in flash memory so are not lost if power fails.

You also have two methods of connecting to a counter. These are usually referred to as 'Push' or 'Pull' connection. The 'Pull' connection method originates from your software, and once connected you can simply retrieve the count data required. With the 'Push' connection method, the counter initiates the connection into your software, and your software must operate like a 'listening' service - once a connection is received it must be accepted and then (depending on how many counters you have connecting into the software) the counters identity must be established before the required data is retrieved. This is normally done by either looking up the IP address from the counter, or by retrieving additional (configured) data from the counter, such as the serial number ('GetSerialNumber'), the unit's description ('GetDeviceName'), or device ID ('GetDeviceID'). By looking up the counter in a database, and seeing when the last data was retrieved, the software will then know what data is to be extracted, and can then pull the required data.

Regardless of the connection type ('Push' or 'Pull') the software can be written to either, drop the connection once the required data has been retrieved, or to keep the connection open constantly. In all cases the software must be written correctly in order to handle any potential network issues. This consists of implementing a retry mechanism in the event that a counter connection cannot be initially made, and gracefully closing connection sockets when a counter unexpectedly disconnects. The APIs provide many different functions to 'Get' and 'Set' various counter parameters, which can be implemented in your software to make it more user friendly and feature rich, but the only other thing that you must implement is a time synchronisation step at some point every day in order to ensure that the counter always logs data against the correct time interval. This consists of simply using the 'GetUnitTime' function to see if the time has drifted, and then using the 'SetUnitTime(time)' function to correct it, if it has.

Full details of how to use the Irisys API functions are shown in the relevant API documentation available from the partner portal.

Unsubscribe Information:

You have been sent this technical bulletin because you have previously contacted Irisys for technical support, or have been identified as a technical contact within your company. If you do not wish to receive this technical bulletin in future then please send an email to counting.suppopt@irisys.co.uk and we will remove you from the email list.