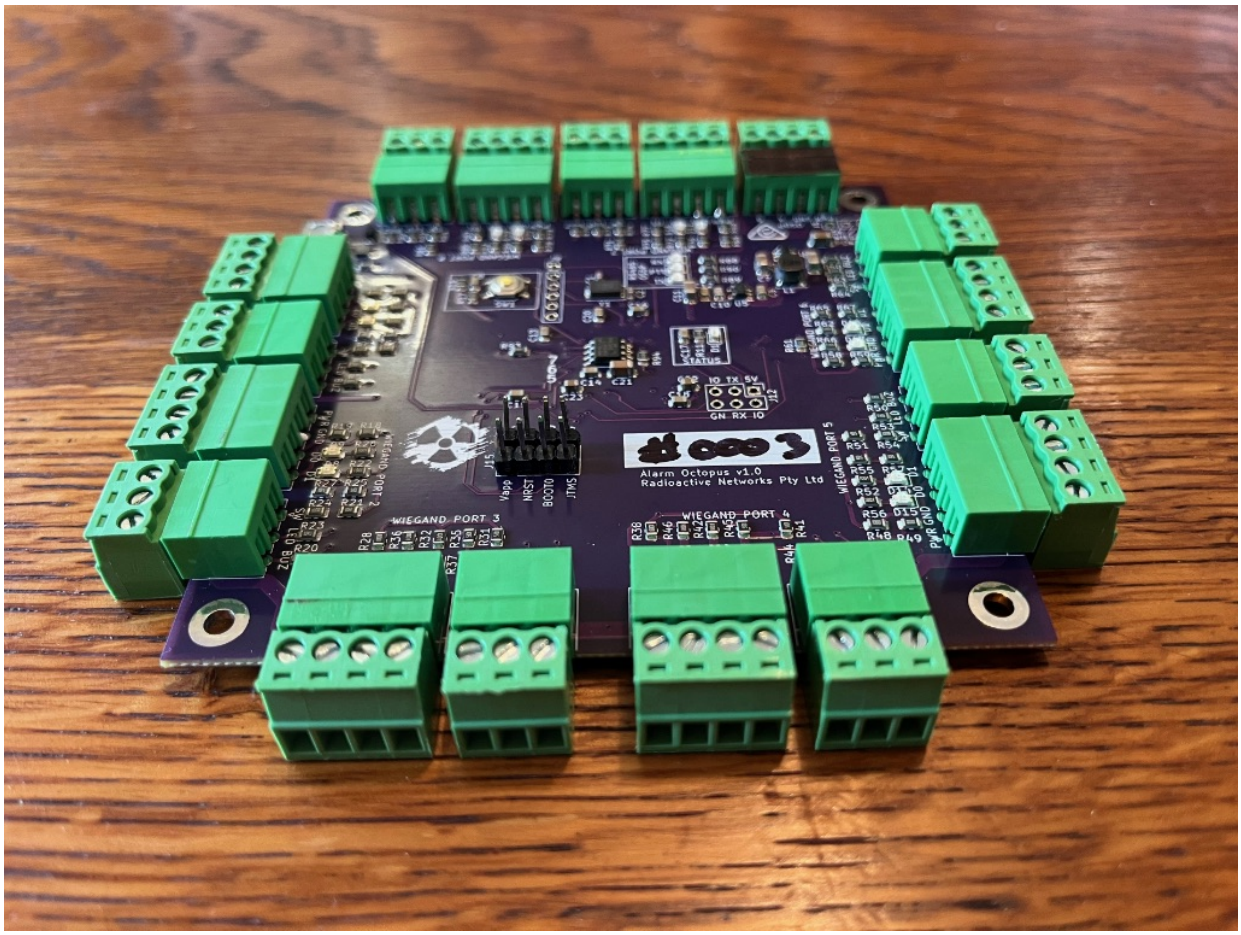
Multi Wiegand to OSDP Converter

Introduction

This device converts signals from Wiegand RFID readers into OSDP protocol for improved communication security. This documentation describes not only the software running on this device, but the device itself. The product is fairly simple. It takes inputs from one or more Wiegand RFID Readers, and converts this into OSDP on a RS-485 serial bus. OSDP has several improvements over Wiegand is that it is encrypted against evesdropping.

We also have a separate getting started document which summarizes much of the content of this document.



Variants

This product line has two variants:

- PD - eight Wiegand RFID inputs with and OSDP output. This is a PD, or Peripheral Device
- CP - OSDP input to eight Wiegand outputs. This is technically a CP, or Control Panel

The PD variant is used to connect multiple Wiegand readers to a control panel, be it an alarm system or access control system. The CP variant is used to connect PD readers of an AlarmOctopus in PD mode into an existing Control Panel with Wiegand inputs.

The products are also be able to be used together to extend a number of Wiegand devices over a much larger distance. This document mostly describes the Alarm Octopus variant. All firmware includes both options.

Hardware

Version one of the product has the following specifications:

- ODSP port operating via RS-485
- USB Serial Port for configuration
- Eight Wiegand inputs
- Wiegand reader voltage is configurable to either 5V or the incoming 12V supply.
- Eight LED and buzzer outputs, normally attached to Wiegand readers
- Eight button inputs, normally assigned to Wiegand readers
- 12V input. The hardware will support up to 24V, but this will also place the 24V input voltage on the 12V line to the Wiegand readers
- The device can also be powered by USB 5V, but any Wiegand readers must be configured to operate on 5V or use an alternate power source. The current of the five volt output is limited to about 200 mA.

The hardware also contains a programming connector, should the firmware need to be updated. Future variants of the product may be updatable via USB.

Pin	Function
1	12V Input
2	Ground
3	RS-485 A
4	RS-485 B

Security Through Local Communications

A conscious design decision has beem made not to enable configuration of this device remotely. This was done to make the unit more secure. Configuration will need to be done via USB.

On Board

LED and Switch

There are two LED's on the OSDP port, one for received data, one to indicate when there is a transmission (labeled PTT).

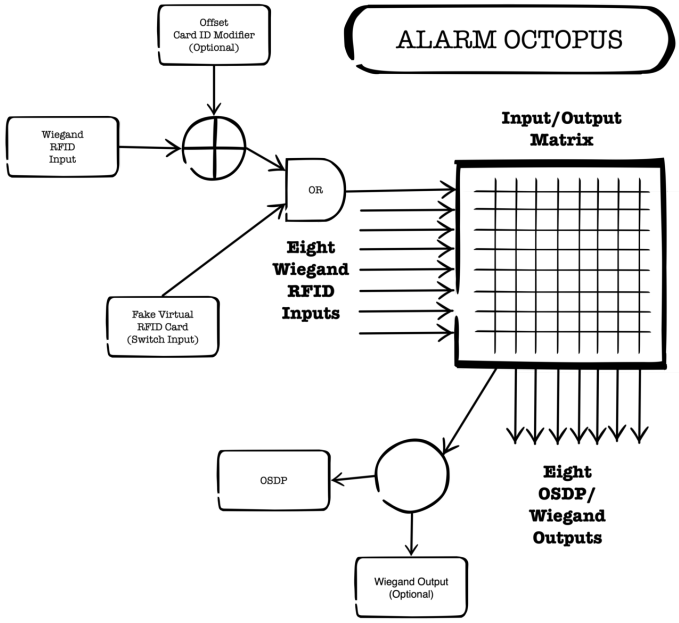
The device also has an on-board LED and a pushbutton switch.

The LED will flash periodically. Normally it will toggle whenever there is a transmission made via the OSDP port. With eight OSDP channels operating, this LED should be almost constantly lit.

You can use the `led` command to change the functionality of this LED.

1. Press and hold button SW1.
2. Apply power to the board.
3. Release the button:

- After 5 seconds to reset to factory settings, and reset the password to the serial number.
- After 30 seconds to reset encryption keys, reset the password, remove master card, fake cards and card offsets.
- After 60 seconds for normal startup.



Pinouts

NOTE: When supplying power via the OSDP port, it is important that the length and gauge of the OSDP cable is sufficient to power any Wiegand attached readers.. Alternatively, it is possible to connect a local power supply INSTEAD of being powered from the control panel.

OSDP Port

1. 12V Input
2. Ground
3. RS-485 A
4. RS-485 B

The OSDP port has two LED's. One for transmit data, and one indicating when the RS-485 device is transmitting.

The default OSDP serial parameters are 115200:8:N:1. They can be changed with the `speed` command.

Main Wiegand Reader Port

1. Power - 12V or 5V
2. Ground
3. Wiegand D0
4. Wiegand D1

Whilst the Wiegand ports normally operate in reader mode, they can be configured to send Wiegand data to other devices.

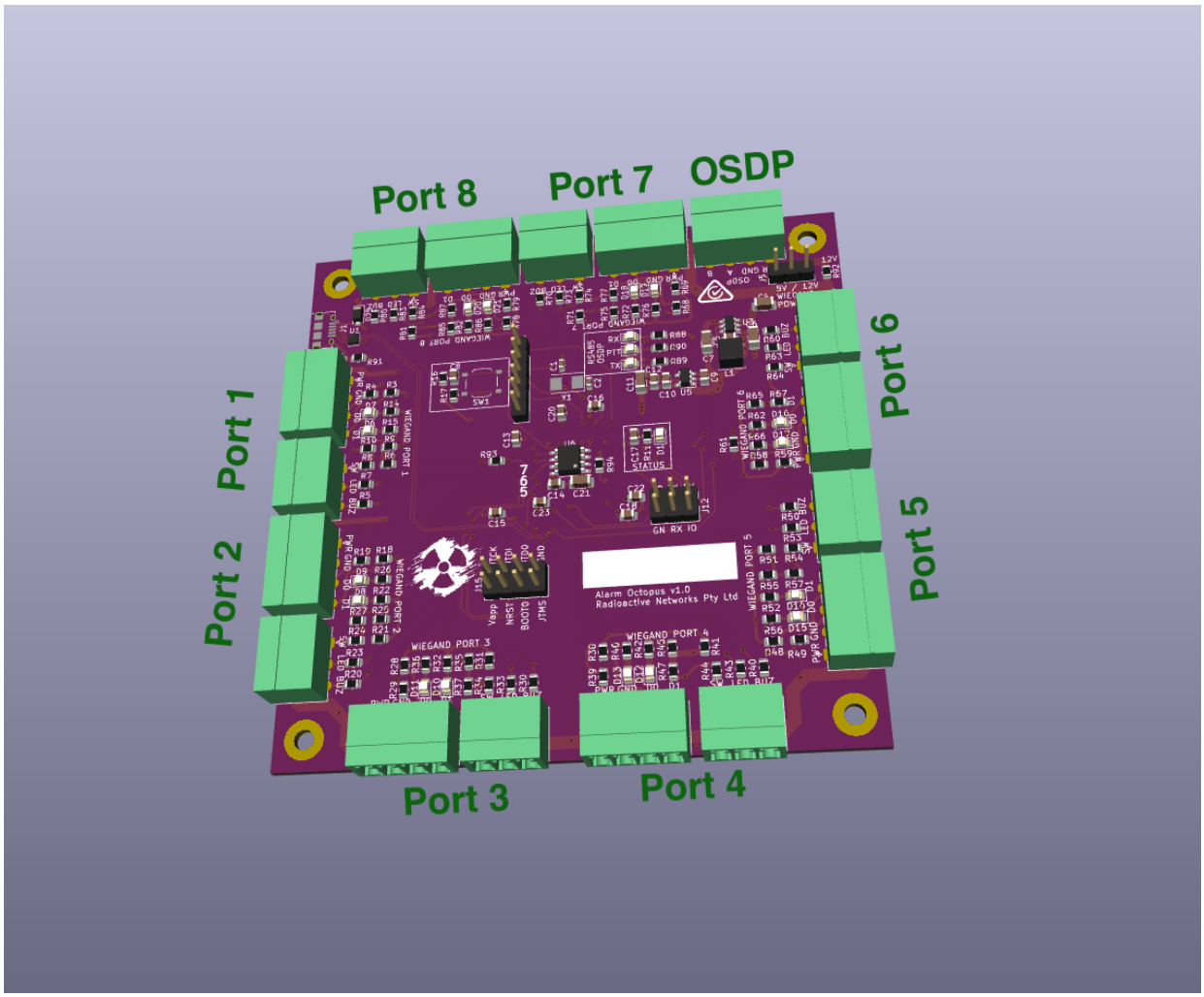
Secondary Wiegand Port

1. Switch Input - Ground input to select
2. LED Output for Wiegand Reader
3. Buzzer Output for Wiegand Reader

The LED and Buzzer outputs can also be configured as inputs in CP mode.

USB

1. Mini-USB Port



OSDP - The Open Supervised Device Protocol

OSDP is a serial protocol that communicates predominantly between RFID readers and alarm control panels. Under OSDP, this is between multiple Periheperal Devices (PD's) and a Control Panel (CP). A Control Panel may have multiple Periheperal Devices (PD's) connected to a single Control Panel (CP). There can only be one Control Panel (CP) on a network. The network runs over a RS-485 serial link.

According to the standard, there may be up to 126 PD's on the RS-485 network. In practice many CP's limit the number of PD's.

When operating in CP mode, there is a limit of 16 OSDP endpoints.

OSDP Limits On Certain Devices

This table will be updated with devices that have been reported to work with the AlarmOctopus. We believe that most, if not all OSDP devices are compatible.

System	Limit
InnerRange Inception	8
InnerRange Simple Lan Access Module (SLAM)	4
InnerRange Intelligent Lan Access Module (ILAM)	16

Please note that we have only tested this hardware on a limited selection of hardware. If you have hardware that is not listed, please contact us, and we will determine the best way of ensuring that the device works with that hardware.

You may need to disable one or more of the devices OSDP interfaces. If the OSDP comms activity link is not mostly on, this could be an indication that the Control Panel really does not like having so many OSDP devices on the LAN. Of course, it could be an indicator of other issues as well.

Scope of the Device

As mentioned above, this device takes Wiegand inputs and sends them out over OSDP

- Eight OSDP devices sharing a single RS-485 link within the device
- Eight Wiegand RFID Reader inputs, along with optional beep and LED outputs
- An extra pushbutton input for each RFID reader that will send an emulated card read with a user defined RFID card
- Each Wiegand RFID Reader can be assigned to any of the OSDP devices

Custom Functionality and Future Devices

Should you have the need for additional functionality from the device, please conact us and we will see what we can do. Additional software functionality is obviously much easier to implement than new hardware functionality.

We are considering additional devices for the future. They might include devices with some type of relay output or PIR style input. If you have a need for any of those type of devices with an OSDP interface, please contact us.

Physical Security of the Device

This device converts the inherently insecure Wiegand protocol into the significantly more secure OSDP protocol. That is not to say that OSDP is perfectly secure. It is not. As with any technology, there is a tradeoff between ease of use and security. OSDP is fairly secure, but the choice to use it is your choice, not ours.

OSDP can be secured with the use of custom encryption keys on the LAN. Using non-defaulty keys will significantly increase the security of the entire system, and should be considered. Having said that, the weakest part of any system is physical security. This device due to its very natue stores the encryption keys locally. In order to make the entire system as secure as possible, you should use different encryption keys on each OSDP device, and also keep each OSDP device as physically secure as possible. The mounting of this device should be secure, protected by locks and tamper switches.

For optimal security, ensure each OSDP device has unique encryption keys, is installed in a secure location, and all physical tamper measures (e.g., locks, switches) are in place.

In addition, the on board USB port should not be connected in production, and should not be left connected.

Updating the Firmware

At the time of writing, there are two methods for updating the firmware. We hope to have further methods in the future.

- ST Link V2 or JTAG using the STM32CubeProgrammer software
- DFU via the USB Port after setting the Boot0 mode

Using the ST Link V2 cable is the easiest way to update the firmware. If a firmware update is needed, we will work with you to determine the best way to update the firmware, and forward a copy of the Firmware Update instructions.

Use Cases

Access Control for Eight Doors

The device by default attaches each Wiegand input to a OSDP device. In this way, the device can control up to eight doors.

Exit Button over OSDP

Each Wiegand port has an additional switch input. This input can be programmed to send a virtual card read. This might be of use if you wanted a Push To Exit button on the inside of a door and wanted to do that via OSDP. The ID of the virtual card is configurable and applies only to an individual reader/door.

In this case, an RFID Card serial number needs to be entered into the Control Panel and into the AlarmOctopus. That serial number will be sent every time the button is pressed. This serial number can be from a card, or generated at random.

Multiple Wiegand Readers in Parallel

The device can be configured so that multiple RFID readers can be sent to the same OSDP port. This would allow, for instance, to have RFID readers on both sides of a door. This would appear to the Control Panel as a single reader, and it could not determine which side of the door the read was on, but it would save an OSDP input in the Control Panel.

Combining Multiple Wiegand Readers Without OSDP

Another way the device can be configured is to combine several RFID readers together to a Wiegand output. In this mode you could have seven Wiegand inputs and a one Wiegand output. Or have two gropus of three inputs being combined into two outputs. Or four inputs and four outputs. The only constraint is that each Wiegand port can only be an input or an output.

Wireless Remote with Multiple Wiegand Outputs

We know of one wireless remote reader that has four Wiegand outputs, one for each button on the wireless remote control. Each button on the remote outputs the same Wiegand ID for the same remote, but each remote sends a different ID. This device can be configured so that all four Wiegand inputs are sent to the Control Panel over a single RS-485 cable without needing to purchase individual converters.

Extending multiple Wiegand Readers

Two devices can be connected together via RS-485 to create a Wiegand extender. One board could have up to eight Wiegand readers attached to it. It could be then connected to another board over RS-485 some distance away. That other board would be configured with up to eight Wiegand outputs feeding into other systems, saving a significant amount of caling.

Configuration Interface

Only a small number of the internal settings of the devices may be configured remotely, via OSDP, such as encrption keys. Most settings must be configured when connected to the device locally via a serial connection. Settings are saved automatically onto the device. They can also be downloaded as a text file as a backup.

The serial connection works best if the terminal emulator handles `bs` and `del`. This is often a setting within the emulation package.

The serial parameters are 115200:8:N:1.

Hardware

The device works with most Wiegand devices that use either 26 or 34 bit formats. Most Wiegand hardware will use one of these two bit lengths. Contact us if you have devices using other formats and we will see what we can do to add them. Other capabilities such as a PIN Keypad may be possible, but this would likely require more work as these devices are less standardised.

Commands

Command	Description
led	Toggles the LED functionality
speed	Changes OSDP baud rate
reset	Resets the device to factory settings

?????

Configuration Cookbook

We have developed a Getting Started Guide with a number of default configurations.

Operations

The device may take a few moments to start up fully. This is because multiple OSDP devices need to synchronise with the Control Panel. Depending on the situation it may take five seconds. Other times we have seen it take a minute. If the Control Panel and the device are both confused because the unit has been restarted, it may take five minutes before both devices synchronise. This is a function of the Control Panel and not this device.

Web Programming

https://www.radio-active.net.au/web4/files/alarm/settings.html

There is a web page to assist with configuring the device. The link is `[HERE](https://www.radio-active.net.au/web4/files/alarm/settings.html)`. The page is self contained with only HTML and JAVASCRIPT and no external libraries so can be downloaded and saved on your computer for use without internet access. You can do anything with the commands below - it is just that the web page makes it a bit easier to navigate the settings.

The web page works best with an external terminal program. Once you have connected and logged into the hardware, as described elsewhere in this document, you may perform the following actions:

1. Type the `info` command using a serial terminal program
2. Copy the output of the command and paste it into the text box at the end of the page.
3. Click the `parse commands` button on the page. Your settings will now be shown in the GUI on the page

Make whatever changes you need to. Once complete:

1. Press the `generate commands` button.
2. Copy all the commands from the large text box and then paste them into the serial terminal.
3. Depending on the settings changed, you may need to power cycle the board.

The only settings that need to be done manually are restarting the device and changing the password.

Web Page With Built In Termain Program

The settings page will now connect via USB to the hardware. This works with Google Chrome, Edge and Opera, amongst some other browsers. It DOES NOT work with Apple's Safari browser, nor with Firefox. This is still experimental and should be used with care.

Command Interface

The command interface is used by using a serial terminal.

System Commands

- reset
- FACTORY
- serial
- speed
- password

OSDP Commands

- osdp
- route
- encrypt
- enable
- debugosdp
- mask
- address

Wiegand Commands

- fake
- info
- card

Control Commands

- basic
- pause
- save
- info
- stats
- hardware
- help
- led

CP Commands

In addition, CP mode has the following command available

- CP

Values

Function	Minimum	Maximum	Example
PD OSDP Port	1	8	
CP OSDP Port	1	16	
OSDP Address	0	126	
OSDP Physical Address (Read Only)	4 Hex Digits	4 Hex Digits	3FDC
Wiegand Port	1	8	
Encryption Key	32 Hex Digits	32 Hex Digits	303132333435363738393A3B3C3D3E3F
RFID Card Number	8 Hex Digits	10 Hex Digits	22015993D5C1
On Time and Off Time	0	255	5 = 0.5 Seconds; 12 = 1.2 Seconds

Hex Digits - A hex digit is between 0-9, and A-F.

RESET - restarts the device - PD and CP

This command simply causes the device to restart as if it was power cycled.

`WARNING`: This command will cause the device to restart

FACTORY - reset to factory settings - PD and CP

This command is in UPPER CASE. Sending this command will cause the device to be set back to factory settings and then restarted.

`WARNING`: This command will reset all settings.

OSDP - set the OSDP Address for each port - PD Only

This device contains a number of OSDP interfaces. This command without any parameters will print the OSDP address for each port. OSDP Port numbers start at 0. OSDP Addresses are between 0 and 126.

```
osdp [PD OSDP Port] [OSDP Address]
```

```
osdp 2 26 // Set the address of OSDP port #2 to 26.
```

```
cmd >osdp
osdp port address
osdp 1 8 default
osdp 2 9 default
osdp 3 10 default
osdp 4 11 default
osdp 5 12 default
osdp 6 5 default
osdp 7 6 default
osdp 8 15 default
cmd >
```

ROUTE - set the OSDP Port for each Wiegand reader - PD Only

Unlike most Wiegand to OSDP interfaces, this device lets you assign multiple Wiegand readers to a single OSDP port. There is no limit to the number of Wiegand devices that can be assigned to an OSDP port. Wiegand interfaces start at 1. OSDP port numbers also start at 1.

```
route [Wiegand Port] [PD OSDP Port]
```

```
route 4 3 // Assign Wiegand port 4 to OSDP port 3
```

```
cmd >route
route port osdp_port
route 1 1 // OSDP Address 8
route 2 2 // OSDP Address 9
route 3 3 // OSDP Address 10
route 4 4 // OSDP Address 11
route 5 5 // OSDP Address 12
route 6 6 // OSDP Address 5
route 7 7 // OSDP Address 6
route 8 8 // OSDP Address 15
cmd >
```

ENCRYPT - set the encryption key for each OSDP port - PD Only

OSDP can be reasonably secure, ensuring the security of both ends. OSPD permits equipment to operate, depending on configuration, unencrypted, encrypted using a well know encryption key, or using a private encryption key.

OSDP encryption keys are 16 characters long, making up 128 bits. These are entered by converting the characters to a pair of hex digits be between '0' and '9', as well as between 'A' and 'F'. Certain encryption keys are better than others. Having a key where the characters in the key are repeated a lot is seen as bad, as are encryption keys where each character or character pair is just one higher or lower than the one before it.

Absolutely the worst is using the default encryption key of '303132333435363738393A3B3C3D3E3F'. This key is often used to initially lock the device before the Control Panel sends a new encryption key to the device. Some Control Panels will provide the encryption key to the user, and it will be their responsibility to transfer it onto a Periheperal Device. This command assists in that regard.

Typing the 'encrypt' command by itself will display the encryption keys for each OSDP port. The OSDP port number starts at 0. The encryption key, as noted above, is 16 characters long consisting of the numbers '0' to '9' and letters 'A' to 'F'.

There is a specific encryption command for CP mode.

```
encrypt [PD OSDP Port] [Encryption Key]
```

```
encrypt 2 303132333435363738393A3B3C3D3E3F
```

ENABLE - enable and disable Wiegand, LED, Buzzer, Relay and OSDP ports - PD and CP

Depending on the product, this product might have sixteen physical Wiegand ports and eight logical OSDP ports. The product only has one physical OSDP port, which is needed for the product to work and cannot be turned off. Any of the physical Wiegand and logical OSDP ports can be enabled or disabled.

This command also gives the ability to enable and disable LED, Buzzer and Relay connections connected to each Wiegand port.

The state may be `1`, `0`, `true` or `false`, `yes` or `no`.

This `enable install` command sets or removes install mode on an OSDP port. Install mode will allow the control panel to set an encryption key for that OSDP port. To do this, the encryption key for the port is temporarily set to a default value. The Control Panel can then send a command to set the encryption key. The device will restart when the encryption key has been saved, and the port will exit install mode. It should be noted that this functionality has not been extensively tested.

When an encryption key has been set by the Control Panel, this will be saved, and the install mode will be removed for that port. It should be noted that the key will not be enacted until the

device is restarted.

The secure mode in OSDP enforces a number of constraints for security. It requires that the channel be encrypted and prohibits the standard install encryption key from being used. Operating with it enabled on each port is good practice. Enabling `install` mode on a port will disable secure mode as the two cannot be active at the same time.

Wiegand, LED and buzzer also function in CP mode.

```
enable [Wiegand|osdp|led|buzzer|relay|install|secure] [port] [state]

enable WIEGAND 3 1 # enable Wiegand port 3

enable WIEGAND 1 1 # enable Wiegand port 1

enable WIEGAND 1 0 # disable Wiegand port 1

enable led 1 1

enable buzzer 2 1

enable relay 4 0

enable osdp 6 1 # enable OSDP port 6

enable osdp 2 false # disable OSDP port 2

enable install 1 true # Enable install mode for OSDP port 1

enable secure 3 false # Disable secure mode for OSDP port 3
```

FAKE - set a 'fake' card for each Wiegand port to be sent on button press - PD and CP

This is an easy feature that is hard to describe. Each Wiegand interface has an additional pushbutton input that can be used if required. When this button is pressed, the device sends the details of a pre-set RFID card to the Control Pane;, as if the person who had pressed the button actually had that card. This feature is there to be used if needed. The card ID is set by this command. Any card ID can be set, but it is probably best if the card ID is used only in accordance with pressing the pushbutton and not issued to a real user.

26 and 34 bit RFID Card ID's are currently supported. More can be added as needed. 26 bit cards are stored as 8 HEX characters. 34 bit cards are stored as 10 HEX characters. In both cases, the numbers are sent LSB first, and are 0 padded at the end. In practice, read a real card and use the information displayed to determine what to enter. On the InnerRange Inception, a 34 bit card might read as '22015993D5C1'. In this case, drop the leading '22' and enter the card ID as '015993D5C1'

The Wiegand port number starts at 1'

```
fake [Wiegand Port] [RFID Card Number]
```

`fake 2 015993D5C1 # send the card ID of 015993D5C1 whenever Wiegand port 2's button is pressed.`

```
cmd >fake
fake 1 0001020304 34
fake 2 0203040506 34
fake 3 0405060708 34
fake 4 060708090a 34
fake 5 08090a0b0c 34
fake 6 0a0b0c0d0e 34
fake 7 0c0d0e0f10 34
fake 8 0e0f101112 34
cmd >
```

MASTER - use a card to log in to the console - PD and CP

Rather than using a password to access the console, it is possible to use a specific RFID card. This will unlock the console as if the password had been entered. It is possible to deactivate this featre by entering a card number with eight or ten 0's. This will unlock the console when used with any active RFID reader connected to this board. No programming is needed within the control panel. In CP mode, the card is read over OSDP.

`master [RFID Card Number]`

`master 0123456789 # unlock the console whenever the card ID of 0123456789 is used`

`WARNING`: Ensure that the master RFID card is kept in a secure location.

SERIAL - Display the device serial number - PD and CP

This command displays the hardware serial number of the device. It is based on the serial number of the CPU and therefore cannot be changed. The serial number may be between 0 and FFFFFFFF. This serial number is also independant of any serial number that may be written on the board. The OSDP port serial numbers are based on this number.

`serial`

```
cmd >serial
serial 0370f399
cmd >
```

LED - change the functionality of the on-board LED - PD and CP

The on board LED has a number of modes. Mode 0 has the led changing state as the main loop executes. In mode 2 the LED toggles in response to OSDP transmissions. Mode 3 momentarily flashes the LED in line with Wiegand output.

```
led [mode]
```

MASK - display the status of the OSDP connections - PD and CP

Most of the comands deal with settings. This command deals with actual status. It reports the status of each OSDP logical port, if it is active and if encryption is active. It only displays OSDP status for ports that are not disabled. The first status returned indicats if there has been a sesssion successfully made on each port. The second indicates if the connection has been secured with encryption. It should be noted that the session may report as being offline between when the connection is initially made and when it is secured.

If a duplicate address is detected, that OSDP port will be locked out until the address is changed or the issue goes away. It takes up to half a minute for the issue to resolve itself.

```
mask
```

```
cmd >mask
mask 1 address  8 connected    secure - Address OK
mask 2 address  9 connected    secure - Address OK
mask 3 address 10 connected    secure - Address OK
mask 4 address 11 connected    secure - Address OK
mask 5 address 12 connected    secure - Address OK
mask 6 address  5 disconnected        - Duplicate Address
mask 7 address  6 connected    secure - Address OK
mask 8 address 15 connected    secure - Address OK
cmd>
```

CARD - send false card code - PD and CP

As noted by the 'fake' command, it is possible to send a preset card read to the Control Panel when a button is pressed. For testing, the 'card' command has been created to emulate that button being pressed. It can be used on any of the Wiegand ports.

```
card [Wiegand Port]
```

```
card 3 // Emulate Wiegand button 3 being pressed
```

OFFSET - Modify the Wiegand code sent from a reader - PD Only

This is another unusual command that has a rather interesting use case. Many Control Panels are limited in the number of OSDP ports that they support. They may only support four OSDP readers for expansion board. There are valid technical reasons why a manufacturer may wish to do this.

Some manufacturers for equally as valid reasons make devices with multiple Wiegand outputs. One device that we have come across is a reasonably priced RF FOB reader which has four Wiegand outputs. FOBs that work with this reader have four buttons and will send a code out a different Wiegand port depending on what button was pressed. The situation though is that the

only difference between any button press for a particular FOB is the Wiegand output the code comes out. Different FOBs have different codes of course.

This means that ultimately by using this device that you have just used up four Wiegand inputs or OSDP devices.

The offset command can help. It is not an ideal solution, but it might be just what is needed. This command modifes any Wiegand reads on a port by applying the offset to the code that is sent. This works best when you have multiple Wiegand ports going to a single OSDP device. In the situation above you could send the four Wiegand ports into a single OSDP device. But in this case, you wouldn't know which button was pressed.

So you then apply the offset command to each Wiegand port - for instance applying an offset of 100 to port 1, 200 to port 2 and do on. Then when a button was pressed on a remote, the code would be different for each button. So instead of enrolling a single FOB, you need to enroll each button. This is not elegant but might be just what the customer needs.

The actual offset is a bit more complex than adding a nunber to the value on each port. It actually performs an 'Exclusive Or' to the card value. Because of that, the offsets in hex are generally best to be a binary sequence. That is, good offsets for the four ports might be something like 00000000, 00010000, 00020000 and 00030000.

As the product matures, the description of this command will likely improve. Right now, if you can't get this command to do what you need, please contact us and we would be happy to help out.

The value for the offset is sent in the same was as the 'fake' command with either eight or ten hex digits.

The Wiegand port number starts at 1'

```
offset [Wiegand Port] [RFID Card Number]
```

```
offset 2 0001000000 // Apply this offset to any card reads on Wiegand port 2
```

OUTPUT - PD Only

This is a feature which works on OSDP ports, regardless if that OSDP port is active or not. When `output` is active for an OSDP port, any Wiegand reads on that OSDP port are also output on the selected Wiegand port as an output. This can be useful when you want to combine two or more Wiegand readers to a single Wiegand output, regardless of if OSDP is needed. Enter a Wiegand port of 0 to disable.

It can also be useful if you want to transition to a new access control system, where the Alarm Octopus can be installed between the readers and the old control panel, and any reads can also be sent to a new control panel via OSDP.

A restart is required when disabling a Wiegand output when you want to use that port for Wiegand input. A restart is suggested needed when enabling a Wiegand input

```
output [AlarmOctopus OSDP Port] [Wiegand Port]

output 2 4 # Send all reader data on OSDP port 2 to Wiegand port 4
```

PROTECT - PD and CP

By default, very little information is shown on the serial console unless the user has entered a password. By setting `protect 0`, all status information will be displayed even if the console is locked. A warning, however, is that the console has the potential to leak information that could be used to bypass security.

```
protect [level]

protect 1 // Protect card data from being examined without a password
```

STATS - PD and CP

The stats page collects a number of statistics for the device. The statistics are colelcted since the last reboot.

For Wiegand status, the numbers line up to the eight physical ports. For OSDP, the numbers line up to the eight local OSDP ports.

Hardware uptime is recorded in eigths of a minute.

```
cmd >stats
Wiegand Status
  Reads 3 6 0 0 0 0 3 0
  Buttons 0 0 0 0 0 0 0 0
OSDP
  Reads 3 6 0 0 0 0 3 0
Dropouts 0
  CP Dropouts 7
  PD Dropouts 7
Hardware
  Uptime 4362
  Reboots 1
cmd >
```

ADDRESS - PD Only

The address command displays the OSDP serial number for each OSDP port, to allow each port to be uniquely identified. The OSDP serial number will be of the form xxx0 for the first OSDP port, xxx1 for the next, etc; where xxx are HEX digits. The OSDP serial number is based on the hadware serial number, and cannot be changed.

```
address
```

```
cmd >address
Address OSDP Port 1: f390
Address OSDP Port 2: f391
Address OSDP Port 3: f392
Address OSDP Port 4: f393
Address OSDP Port 5: f394
Address OSDP Port 6: f395
Address OSDP Port 7: f396
Address OSDP Port 8: f397
cmd >
```

DEBUGOSDP - PD and CP

This is a command that hopefully never needs to be used. It provides low level debugging of the communications between the control panel and the board. Debugging is on a per OSDP port basis. The port number is just the index of the OSDP port. Flags are a bit more complex. Flags is a number. To determine the value of the flag, add up all the desired options.

1 = Data Trace. 2 = Packet Trace. 4 = Monitor POLL packets. 128 for debug of LED and BUZZER

If you want to do a Data Trace (1) and Monitor Poll Packets (4), add up the values for both, and in this case you will get a flag value of 5. If you want everything, select a flag value of 7. It can be disabled on a per port basis with the value of zero.

Just a warning. If you start using this feature and want to turn it off, either send the commands to turn it off without being able to see what you are typing, or disconnect the OSDP port.

`debugosdp [OSDP Port] [flags]`

`debugosdp 2 7`

basic - PD Only

The `basic` command very quickly sets up a few basic settings. It removes any OSDP debugging on all ports. It also only enables a single OSDP port. This is enough to be able to type more commands without being bamboozled.

`basic`

PAUSE - PD and CP

The `pause` command pauses saving settings until the command is issued again or the device is restarted. This can be useful when saving a lot of settings as it will reduce the number of writes to memory. This command should not be needed.

`pause`

SAVE - PD and CP

The `save` command saves settings, when the `pause` command is active.

```
save
```

PASSWORD - PD and CP

The password command is used to unlock the console, lock the console and change the password. This functionality is designed to stop casual tampering. It is not designed to stop someone who has the means and resources to bypass the password. It should be reasonably secure. Obviously, the password used should be unique and not used on any other system. We do not use the password directly, but store a transform of the password using a one way function.

To lock the console, type 'password'.

To unlock the console, type 'password' followed by your password. The default password is the serial number of the unit. It is recommended that you change this.

To change the password, type 'password set' followed by your desired password. The password must be eight characters or more. There are two special 'passwords'. The first is 'blank' which will remove the password. The second is 'serial' which sets the passowrd to the serial number of the device. This is the default password.

If a password is set, the console will lock automatically about five minutes after the last command is entered, or when 'password' is typed by itself.

```
password
```

```
password [password]
```

```
password set [password|blank|serial]
```

SPEED - PD and CP

This command changes the speed of the OSDP port. You must perform a restart of the board to enable these settings. The board cannot have the OSDP serial speed changed with the osdp_COMSET command under the OSDP protocol. Valid speeds are 9600, 19200, 38400, 57600 or 115200.

```
speed
```

```
speed [speed]
```

INFO - PD and CP

Display information on all parameters

```
info
```

HELP - PD and CP

Display bacic help. Type help byitself to list all known commands, and help followed by the name of the function for specific help on that function

```
help
```

SHOW - PD and CP

Display the last few Wiegand card reads, along with the port and the age. The `ago` number is the approximate number of seconds since the card was read.

```
show
```

```
cmd >show
Port:7 Card:035983b7a0 Bits:34 Ago:20
Port:2 Card:035983b7a0 Bits:34 Ago:19
Port:1 Card:015993d5c1 Bits:34 Ago:18
Port:7 Card:035983b7a0 Bits:34 Ago:17
Port:2 Card:015993d5c1 Bits:34 Ago:14
Port:1 Card:015993d5c1 Bits:34 Ago:14
Port:2 Card:035983b7a0 Bits:34 Ago:13
Port:1 Card:015993d5c1 Bits:34 Ago:9
Port:2 Card:035983b7a0 Bits:34 Ago:8
Port:7 Card:015993d5c1 Bits:34 Ago:7
Port:2 Card:035983b7a0 Bits:34 Ago:7
Port:2 Card:035983b7a0 Bits:34 Ago:6
cmd >
```

INJECT - PD and CP

Inject RFID from the command line as if it was sent by an actual card reader. It works on an OSDP port. The Card number needs to be eight or ten hex digits long. This command can be used to integrate other hardware to the alarm system.

```
inject [Wiegand Port] [RFID Card Number]
```

```
inject 2 ABCD1234EF
```

CP or Control Panel Mode

Control Panel, or CP mode is a different philosophy than the main AlarmOctopus mode. It is a destination for one or more OSDP devices. There are two components

- Wiegand outputs
- Binary inputs

In this mode, multiple readers are connected to the board via OSDP. The board then has multiple Wiegand outputs which can then be used as inputs to a separate access control system. In this way an access control system with Wiegand inputs can be converted to OSDP without any chnages to the access control system.

Wiegand Data

When the device receives Wiegand data via OSDP, that data is transmitted out the assigned physical Wiegand port. This port would be in turn normally connected to an access control system.

Binary Inputs

There are two binary inputs attached to each Wiegand port, one for an LED and the other for a BUZZER. When either of these is triggered, a command is sent over OSDP to the relevant reader. How these are connected up depend on the system that the device is connected to. If the connected hardware contains an LED and BUZZER output then all good. Another solution would be connect one or both to the door output on the connected hardware, giving audible or visible indication that the read was successful.

Both the LED and BUZZER have tempoary and permament settings. Temporary settings operate for a nominated period, after which the permament settings are enacted. Temporary settings operate for a set number of cycles.

Buzzers operate with on and off times only. An LED not only has an on and off time, but also has on and off colours. This all sounds complex, it will becomre more understanable once you start using it. Alas, there is really no simple way to explain this all.

Multiple OSDP Devices

Multiple OSDP devices can send data to a single Wiegand port. There is a limitation of 16 OSDP devices attached to the device however. This permits, for instance, multiple OSDP devices to send reports out a single Wiegand port connected to an an access control system. This might be of use where there are two RFID readers at different heights at the gate to a facility. Another use case might be readers on both sides of a door, where only certain users may pass through that door.

LED COLORS

Value	LED Color
0	Off
1	Red
2	Green
3	Amber
4	Blue
5	Magenta
6	Cyan
7	White

CP (Control Panel) Commands

There is an extra set of commands speficically for CP mode. These are grouped together. We envisage that most people using this product will use it in PD mode. Therefore, we have tried to place all the CP specific commands together.

Common Commands

Some of the commands work identically or in very similar ways in CP and PD mode.

The first group of commands are configuration commands that will operate in CP mode.

```
enable [LED|WIEGAND|BUZZER] [PORT] [STATE] // Enables and disables
functions on the Wiegand port master 0123456789 // unlock the console
whenever the card ID of 0123456789 is used via OSDP on any port fake
[Wiegand Port] [RFID Card Number] // Set the CARD to be sent when the
button input on an RFID port is pressed card [Wiegand Port] // Emulate
the button input being pressed on the Wiegand port
```

The following simple commands operate identically in PD and CP modes:

```
RESET, FACTORY, SERIAL, LED, MASK, PROTECT, STATS, DEBUGOSDP, PAUSE,
BASIC, PASSWORD, SPEED, INFO, SHOW, INJECT
```

Enable CP Mode

This command enables or disables CP mode. The device will reboot once this command has been saved.

```
CP MODE [TRUE|FALSE]
```

```
CP MODE TRUE
```

OSDP CP Commands

The next set of commands operate on a OSDP Port. There are 16 OSDP ports, numbered 1-16. Each OSDP Port can be forwarded to one or more physical Wiegand ports.

Enable and Disable OSDP Ports

This command enables or disables OSDP ports

```
CP OSDP [CP OSDP Port] ENABLE [TRUE|FALSE] CP OSDP 3 FALSE - Disable OSDP
port 3
```

Set the OSDP address of the OSDP port

Each of the 16 OSDP ports need a unique OSDP address. This can be the same as the OSDP port number, but can be different. It must be between 0 and 126.

```
CP OSDP [OSDP PORT] ADDRESS [OSDP ADDRESS] CP OSDP 4 ADDRESS 16 # Set OSDP port 4 to OSDP address of 16
```

Set the Wiegand HARDWARE port for an OSDP port

OSDP Ports are LOGICAL constructs. That is, they exist in software only. Each OSDP Port needs to be assigned a PHYSICAL Wiegand port. For instance, OSDP Port 1 might be assigned to Wiegand 5 and OSDP Port 2 might be assigned to Wiegand 2. This can get confusing.

Also, more than one OSDP Port may be assigned to a single Wiegand port. This allows multiple readers to be assigned to a single Wiegand output on the CP.

```
CP OSDP [OSDP PORT] WIEGAND [WIEGAND PORT] CP OSDP 1 WIEGAND 5 # Assign Wiegand physical port 5 to OSDP port 1
```

Set the OSDP port to INSTALL or SECURE mode

There are two main security settings available on a per-port basis. INSTALL mode sets a default encryption key on the port. This makes setting up readers easier. In many cases, it is the only way to set up readers.

The other setting is SECURE mode, where the OSDP link MUST be encrypted, and must be configured with a non-default key.

```
CP OSDP [OSDP PORT] SECURE [Y/N] CP OSDP [OSDP PORT] INSTALL [Y/N]

CP OSDP 4 INSTALL Y
```

Set the ENCRYPTION for an OSDP Port

Each OSDP port will have an encryption key assigned to it. Best practice is for this key to be different for each port. The encryption key is 32 hex digits long

```
CP OSDP [OSDP PORT] ENCRYPTION xxx

CP OSDP 3 ENCRYPTION 30313233343536373839 3A3B3C3D3E3F
```

Wiegand CP Commands

When operating in command mode, pulling the LED or Buzzer input low will send a command via OSDP to a connected reader, activating the LED or Buzzer. Inputs to the CP will generally be from an access control system, and will become active to generally signal either a valid or invalid card read. The commands are almost identical for LED and Buzzer.

LED settings can either be temporary or permament. Buzzer settings may only be temporary.

Any temporary settings are applied for a set time, and are then replaced by permament settings, if configured. If there are no permament settings configured, the output is just turned off. Obviously, you dont want a buzzer always making a noise, which is why you cannot turn it on permamently.

Temporary settings assume that the LED or Buzzer will be cycling. For the Buzzer, it will cycle between on and off. For LED it can cycle between two colours. The ON and OFF times are in 1/10ths of a second. Setting an ON TIME of 5 and an OFF TIME of 5 will activate the output for half of each second. COUNT is the number of ON and OFF cycles before moving to either a permament setting or turning the output off.

If [ON TIME] or [OFF TIME] are 0, the other time will be applied, effectively leaving the output ON or OFF for the specified time, rather than switching it on and off. If [ON TIME] and [OFF TIME] are both 0, the device will ignore this rule.

Different commands can be sent when the input goes LOW, and when it returns to its normal HIGH state.

When configuring the LED, you need to select the LED colours. The LED Color Codes are shown above.

```
CP WIEGAND [WIEGAND PORT]   LED     [HIGH|LOW] TEMP  [ON TIME] [OFF
TIME] [COUNT] [ON COLOR] [OFF COLOR]
CP WIEGAND [WIEGAND PORT]   BUZ     [HIGH|LOW] TEMP  [ON TIME] [OFF
TIME] [COUNT]
CP WIEGAND [WIEGAND PORT]   LED     [HIGH|LOW] PERM  [ON TIME] [OFF
TIME] [ON COLOR] [OFF COLOR]

# When the LED port on Wiegand 3 goes low, FLASH the LED 0.5 seconds
red and 0.5 seconds green for a total of 5 seconds
CP WIEGAND 3 LED LOW TEMP 5 5 5 2 1
# When the previous rule has finished, set the LED to steady
CP WIEGAND 3 LED LOW PERM 10 0 4 0
# When the Buzzer port on Wiegand 3 goes high, turn the buzzer on for
0.2 seconds, and then silent for 0.2 seconds, for a total of 2
seconds
CP WIEGAND 4 BUZ HIGH TEMP 2 2 5
```

Glossary

Word	Definition
Wiegand	A commonly used interface between RFID Card Readers and Access Control Systems. It uses two signal wires, ground and optionally a power signal
RFID Card	Radio Frequency IDentification - a device, generally the size of a credit card, where a unique ID serial number will be transmitted wirelessly when it is brought close to a compatible reader. RFID cards commonly operate on 125 kHz or 13.8 MHz

Word	Definition
OSDP	Open Supervised Device Protocol - a software protocol, generally operaing in a two wire RS485 bus, that provides an interface between an access control system Control Panel and a up to 126 RFID readers.
RS485	A two wire communications interface for transmitting data long distances, up to 1000m under ideal circumstances. An earth is normally used in additiion to the two data lines.

RoHS/Reach Statement - Hardware

Regulatory Statements

RoHS

All PCBs sold here are manufactured according to the RoHS-3 Directive (2015/863).

REACH

All PCBs sold here do not contain more then 0.1% of the 219 substances on the SVHC candidate list (1907/2006, updated July' 2021)

Within the meaning of REACH Radioactive Networks is exclusively a "downstream user". Our PCBs are exclusively items that do not release substances under standardized conditions of use. Therefore no registration or safety datasheets are required.

PCB Manufacturer

Manufacturer Link: https://jlcpcb.com/ RoHS Test Report: PCB-FR-4-ROHS-Test-Report.pdf REACH Test Report: REACH-SVHC-211-substances-compliance.pdf Certifications: Link

Open Source

As with most modern software, this product uses a number of Open Source libraries. We are more than happy to provide you with copies of the libraries we use. Alternately, you can download the software on the links below. Importantly, we have modified the LibOSDP package, and have made our modifications available for free. Please be aware however that our own software and hardware are both convered by copyright. Likewise, we do not offer support on any of these libraries, including our modifications to the LibOSDP package.

LibOSDP

- https://libosdp.sidcha.dev/license - Apache
- Modifications available on https://github.com/vk2tds/libosdp_arduino

Automaton

- https://github.com/tinkerspy/Automaton/blob/master/LICENSE - MIT
- https://github.com/tinkerspy/Automaton/

FlashStorage-STM32

- https://github.com/khoih-prog/FlashStorage_STM32/blob/main/LICENSE - MIT
- https://github.com/khoih-prog/FlashStorage_STM32

ParseCommands

- https://github.com/Gfy63/ParseCommands/blob/main/LICENSE - GPL
- https://github.com/Gfy63/ParseCommands

Yet Another Arduino Wiegand Library

- https://github.com/paulo-raca/YetAnotherArduinoWiegandLibrary/blob/master/LICENSE - Public Domain
- https://github.com/paulo-raca/YetAnotherArduinoWiegandLibrary

ArduinoUniqueID

- https://github.com/ricaun/ArduinoUniqueID/blob/master/LICENSE - MIT
- https://github.com/ricaun/ArduinoUniqueID