Orbit IPv3

Programmer Guide

Gemini 2000 Ltd
www.gemini2k.com
+44 1202 666 700

Note on Applicability

This document accompanies the Orbit IPv3 model smart card reader and is compatible with firmware version 4.1.7. Other firmware versions may not support all functionality listed here.

Product Description

Orbit IP is a TCP/IP Ethernet-based NFC reader for contactless smart cards. It is compliant with ISO 14443 Type A/B and ISO 15693 standards. It is powered via Power over Ethernet interface as standard; mains power supply is optional.

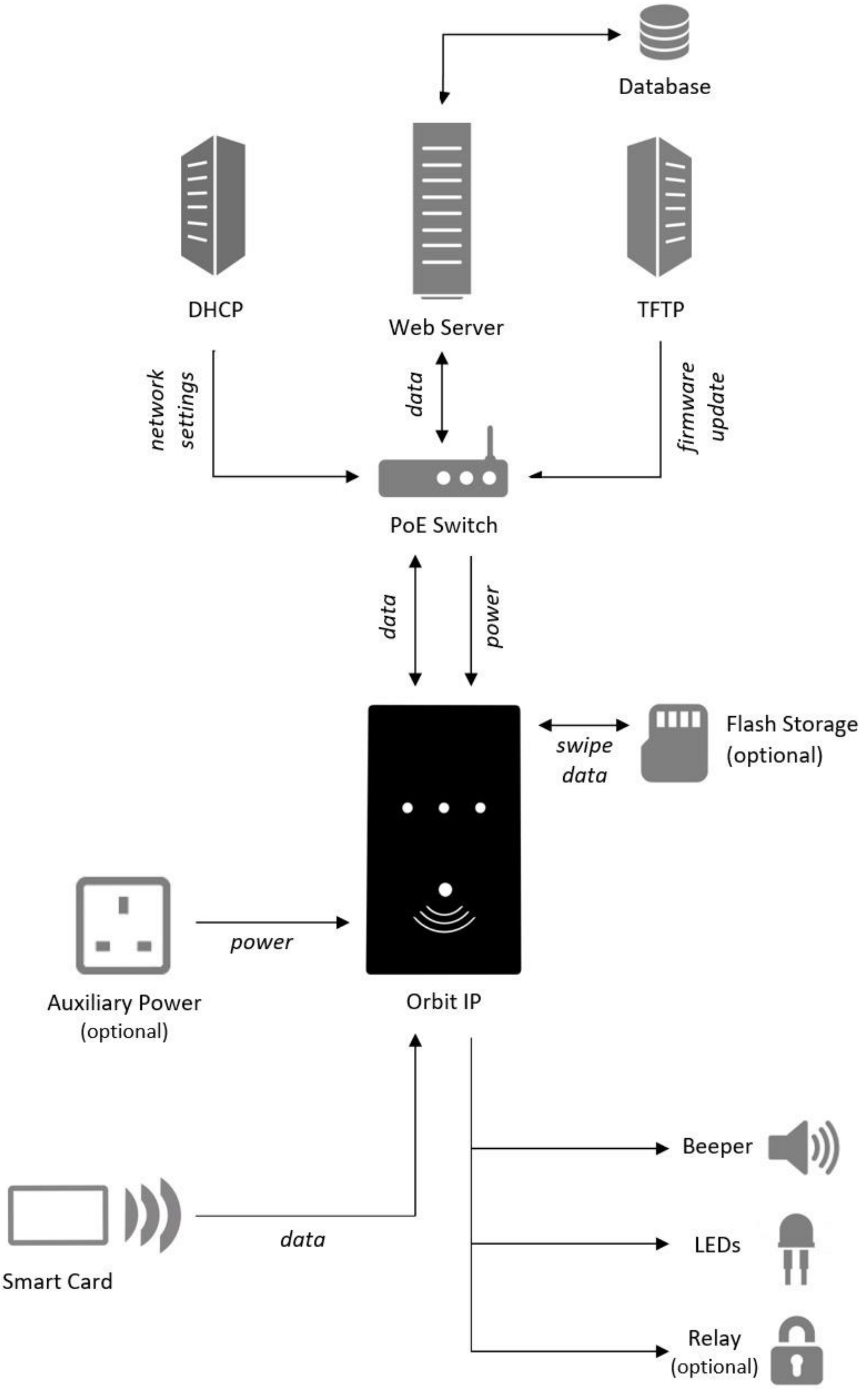
Orbit IP operates as a standalone HTTP web client. It does not require a computer to perform smart card communications. It can connect to a 10/100 Base hub or switch and interact with a web server using common web languages. A complete, powerful access control or time / attendance system can be built on a simple web page.

Various web page extensions are compatible such as .php, .aspx, .cfm, .pl, .html. This capability allows easy integration with various HTTP web server systems like Microsoft IIS with ASP, Apache with PHP, MySQL database server, and more.

When an ISO14443 Type A contactless card is detected by the reader, it generates a GET request to a defined web server. The server can respond with a standard HTTP reply to the reader with embedded controls between <ORBIT> and </ORBIT> tags.

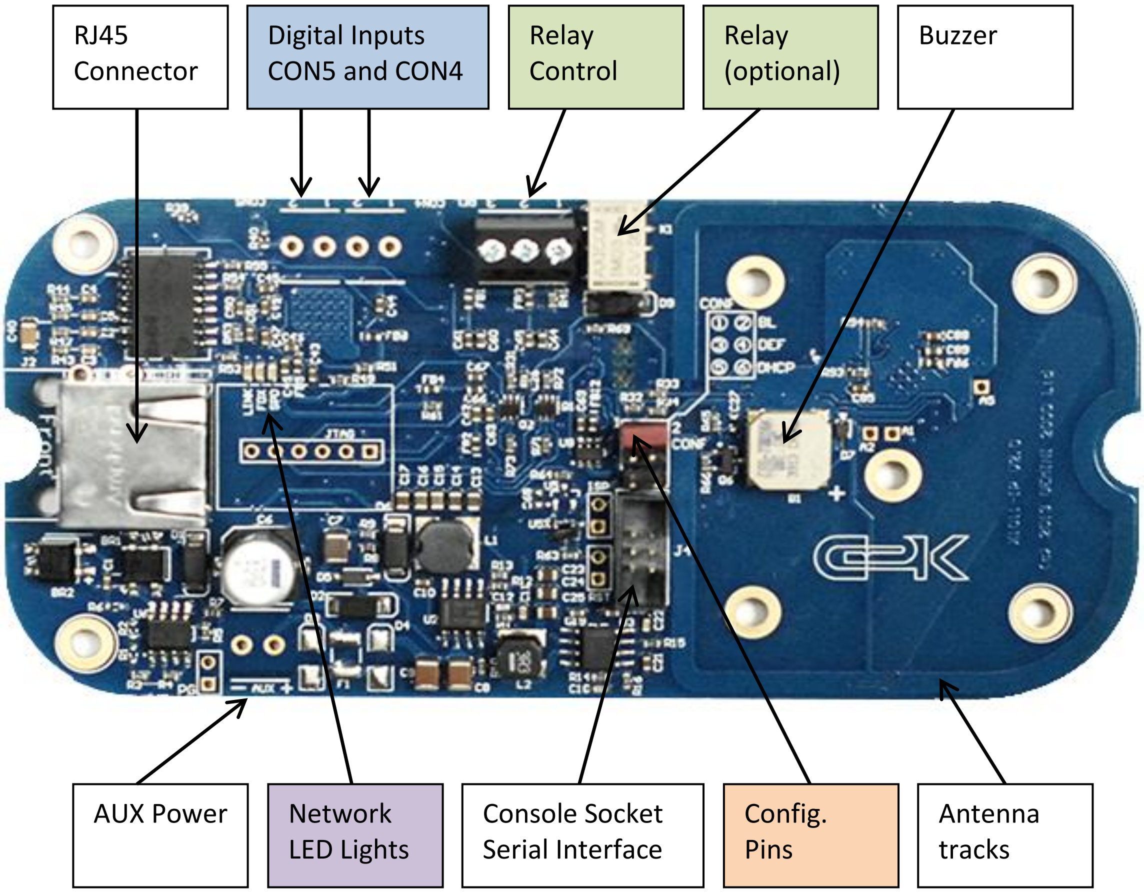
Orbit IP is equipped with flexible server-controlled user interface with 3 LED indicators and a beeper. Optionally, the reader can incorporate a relay control of external devices.

Implementation Diagram



Hardware Description

PCB Top Layer



RJ45 Connector

Digital Inputs CON5 and CON4

Relay Control

Relay (optional)

Buzzer

AUX Power

Network LED Lights

Console Socket Serial Interface

Config. Pins

Antenna tracks

Relay Control black 3-pin connector. Pin status when relay coil not powered:

1 - NO - Normal Open
2 - CT - Central Tap
3 - NC - Normal Close

CONF pins. Prior to power up place jumper to force configuration modes:
1/2 - BL Bootloader firmware update
3/4 - DEF Restore default settings
5/6 - DHCP Force DHCP mode

Network LED Lights. Indicate the reader's network status:
LINK - Link established
FDX - Full duplex link
SPD - Transmission speed

Digital Inputs

CON4 Voltage Controlled Input 1:
1 - +5V Max -0.5V Min
2 - GND

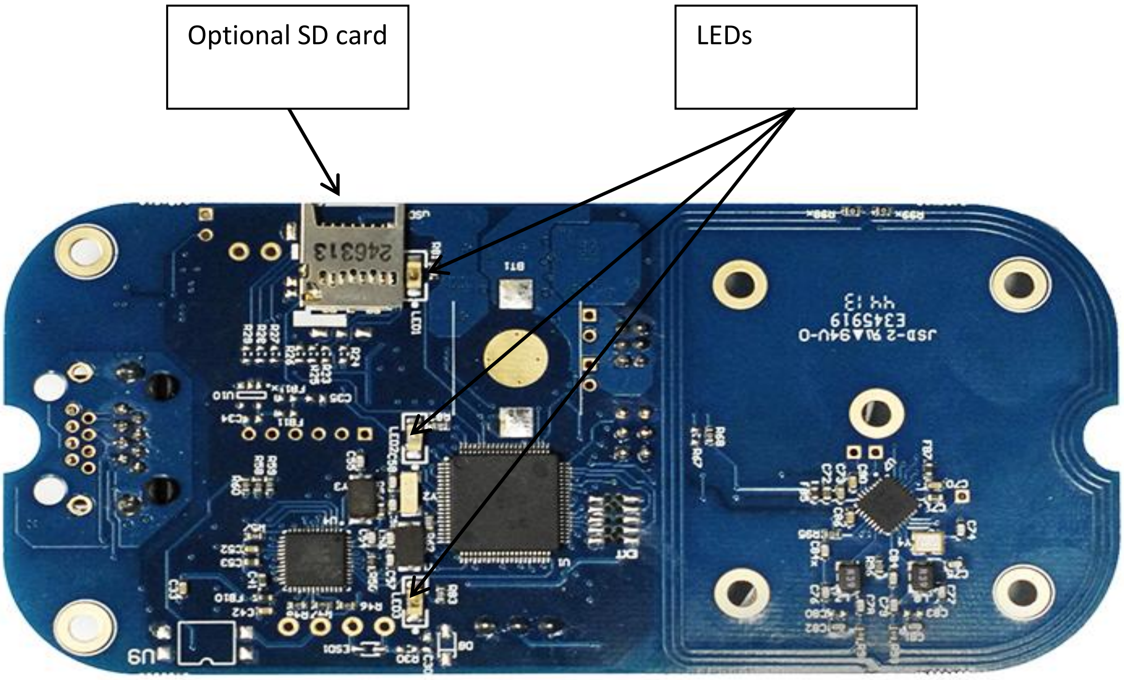
CON5 Current Controlled Input 2:

The terminals are connected through a series resistor (330 Ohm) to the A and K pins of an optocoupler.

1-K – usually connected to external device ground.
2-A – provide 5mA-20mA current.

Exceeding the current limits will damage the reader input. When the current is flowing through the pins the logic level reported is LOW (zero).

PCB Bottom Layer



Optional SD card

LEDs

Housing

Orbit IP comes with a choice of three casings – a plastic case, a metal case, and a lid design that fits a standard double gang pattress box. The plastic and metal cases have desktop and wall mount variants, while the metal case can also be supplied as an IP65 weatherproof modification. Black is the default colour option, but custom colours can be made upon request. Orbit IP is also available as an OEM module.

Aluminium wall mount case
enclosed housing, CAT5 lead provided at rear
available as **Waterproof wall mount option**
available as **Desktop option**
140 x 75 x 40 mm (HWD)

Plastic pattress mount case
open at the back, RJ45 socket provided
fits standard double pattress box

145 x 86 x 11 mm (HWD)

Plastic wall mount case
open at the back, RJ45 socket provided
available as an enclosed **Desktop option**
130 x 67 x 25 mm (HWD)

OEM

123 x 58 x 18 mm (HWD)

Installation

Power Supply

In order to function correctly, Orbit IP must have stable power supply, a 100-baseT Ethernet connection. The reader is compliant with PoE standard IEEE 802.3af-2003 and is configured as Power Class 0 device. It can draw power via a standard Cat 5 network cable from a PoE switch or an active 48V injector. Auxiliary power can be supplied to 'AUX' connector in case the PoE option is not applicable. In this case the power source must provide DC voltage 9V-24V at 2W. The auxiliary power option is available upon customer request.

Warning! Do not use both AUX and PoE supplies at the same time. Doing so will damage the reader and void its warranty.

Power Usage

Shown below is measured power usage when the reader is connected to auxiliary 12v power supply:

Typicalcurrentusage (mA)	Voltage (Volt)	Power (Watt)	Description
100	12	1.26	Typical idle current ranging from 80 to 100 mA
135	12	1.62	Typical card tap peak with a relay populated and powered
140	12	1.68	Typical idle current with 1.8 Inch display
170	12	2.04	Typical card tap peak with a relay and 1.8 Inch display

Connecting to a Network

Use an RJ45 connector to connect the Ethernet port of Orbit IP to a PoE network hub, switch or an active 48V injector. The cable for the connection should be Category 3 or 5 UTP/STP cable, which is compliant with IA/TIA 586 specifications.

The maximum length between the hub and reader is 200ft.

Network link status is indicated by the LEDs next to the RJ45 connector.

Default Settings

When you receive your new Orbit IP reader, it will have default network configuration parameters:

Parameter	Description	Default Setting
IP	Static IP address of the reader	192.168.7.200
GW	Gateway IP address	192.168.7.1
NM	Netmask	255.255.255.0
WS	HTTP web server IP address	192.168.7.191
PT	HTTP server port number	80
EXT	Page extension	PHP

To define new parameters for your reader, refer to **Responses from HTTP server**.

Resetting to Default Settings

Disconnect the reader. Fit a jumper over pins 3 and 4 (marked as 'DEF') in the J4 configuration socket near the centre of the board. When powered on, the settings of Orbit IP will be cleared and set to the default parameters shown above. Wait for the distinct sound indicating successful reset. The jumper must be removed for normal operation after reset.

Serial Port Configuration

The factory default configuration can be overwritten either through the web interface, or using a USB configuration cable and Windows software that can be obtained from Gemini 2000. Refer to the Configuration Tool user manual for details.

Force DHCP Mode

To force DHCP mode, disconnect the reader and fit a jumper over pins 5 and 6marked “DHCP” in the configuration socket J4 near the centre of the board. Power up the reader and DHCP mode will be forced.

The DHCP server must be configured to respond with the www-server, i.e. option 72. If the reader receives a valid DHCP response without option 72, it will use the static webserver IP address stored on the reader.

The reader can enter DHCP mode using the web command DHCP =1.

Sample DHCP Code

Below is a snippet of the configuration file[dhcp.conf] used with ISC DHCP Server available for most Linux distributions.

```
subnet 192.168.7.0 netmask 255.255.255.0 {

   option www-server 192.168.7.191;

   option domain-name-servers 213.120.62.103;

   option routers 192.168.7.1;

   option tftp-server-name "192.168.7.12";

   option dhcp-lease-time 60;

   pool {

          range 192.168.7.210 192.168.7.250;

          max-lease-time 60;

          allow unknown-clients;

        }

}
```

Software Configuration

Orbit IP implements an HTTP client. Actions such as power up, card read or heartbeat generate URL requests to the server with one or more parameters. By default, requests are made to orbit.php file located on the root of the server IP address.

Responses from the web server constitute commands to perform a variety of actions on the reader, such as beep, action LED sequence, change settings, etc.

Requests to HTTP Server

The following requests description assumes PHP extension is configured on the reader. The filename extension can be configured according to the end-user environment. See section EXT= below.

date and time

Date and time of the current request.

The date format is YYYY-MM-DD, for example 2015-03-12 means 12th of March 2015.The format of time is HH:MM:SS where HH is hours, MM is minutes and SS is seconds. The date and time of the reader must be correctly set to ensure proper operation.

Note: The reader's date and time are lost upon power down. It is advised to set them by the server at each power up request. If it is required to keep the date and time during power downs, Gemini 2000 can fit an optional battery on the reader.

id

The current IP address of the reader, for example 192.168.7.200. It must be set to a unique value for each reader in a network.

ulen

Length of the contactless smart card UID. The values can be 4, 7 or 10.

uid

Contactless smart card UID as an ASCII string in hexadecimal format. The value is of variable size; $ulen indicates the UID length. For example:

for $ulen=4, $uid=8A4962A3
for $ulen=7, $uid=887766550102AB
for $ulen=10, $uid=123456789ABCDEF01234

cmd

Indicates the action that was taken by the reader:

- $cmd=PU is sent once after the reader is powered up.
- $cmd=CO is sent when a card is detected by the reader.
- $cmd=HB is sent at regular intervals – this is a programmed heartbeat rate. The default heartbeat rate is 30 seconds.
- $cmd=SW is sent when a level change is detected either on digital Input 1 or Input 2.
- $cmd=PG is sent when the reader received a "ping" request.
- $cmd=CB is sent when an offline stored message is transmitted.
- $cmd=COUT is sent when a card removal is detected by the reader.

ver

Current firmware version, e.g. $ver=4.1.6

blver

Current Boot loader version, e.g. $blver= 00000400

contact1, $contact2

It is used with $cmd=SW to indicate the on/off logic level at the digital inputs.

sid

The value of this parameter is set by the SID response.

data

This is the data of the block readout in hexadecimal format.

psrc

After pinging a reader, the psrc parameter indicates the IP address of the machine which initiated the ping.

pmeth

After pinging a reader, the pmeth parameter indicates the method of the ping. Normal ping utility uses ICMP echo, and those the reader would report ICMP. However, if the reader was TCP pinged, it would report TCP.

RE

A report on the number of failed HTTP requests the reader has sent before the reader has managed to establish connection. The value of this parameters is only valid when offline mode is off.

md5

The MD5 checksum of the user's MD5 secret key, date and time combined. It is sent with the HTTP request for authentication purposes.

For example:

If we have the MD5 key 'S0lut!0n' in ASCII stored as a hexadecimal value in the server-side configuration file as '$md5key= 53306C757421306E' and if the time of the request was '10:54:38' and the date of the request was '2017/02/16' then we have the following parameters:

- MD5 secret key is 53306C757421306E (hex),
- Date of the request is '2017/02/16' (ASCII),
- Time of the request is '10:54:38' (ASCII)

The combined resultant ASCII formatted string would be:
'S0lut!0n2017/02/16-10:54:38' (ASCII)

If we generate MD5 checksum of the previous combined string then we will have the following MD5 hash which the reader would return when it sends request to the server:

- 'C9B9D09B742DAD189E5869BE60E6D727'

The MD5 secret key is known to the HTTP server and the reader.

The server code can calculate the MD5 sum against the data received from the reader in the current request thus validating the authenticity of the data. Only a reader configured with correct MD5 key can produce a correct checksum. The MD5 keys for each reader can have unique values and be stored in a database.

mac

Contains the MAC address of the reader.
Example:
mac=00:08:DC:FF:D0:03

relay

Reports the current configuration of the relay control feature. The value "1" indicates that the relay control is active.
Example: relay=1

sd

Reports the presence and status of the optional internal MicroSD card. sd=1 indicates card present and operational, sd=0 indicates card not present or not operational.
During start-up, a failed card read initiates a beeper sequence alert.
Example: sd=1

Responses from HTTP server

After sending a request, Orbit IP waits for a response from the server. It looks for the starting tag <ORBIT> and the ending tag </ORBIT>, then interprets the ASCII data between them.

The command/value pairs must be delimited with at least one of the following symbols:

\t <TAB>
\n <NL>
\r <CR>

The maximum size of the response packet must be less than 1200 bytes.

A Warning About Persistent Parameters

Persistent parameters are stored permanently on the reader and do not get lost upon loss of power. Users shall be particularly careful with persistent parameters since they're stored in the reader's EEPROM. These components have a large, but limited number of write cycles, usually a minimum of 1,000,000. We therefore advise users to minimise the use of persistent parameters in their implementations, for example do not use at heartbeats.

General Commands

General Peripherals

LED1= (LED2=; LED3=) ***Standard Command***

These commands control the individual LED status. Values are in decimal format. The parameter indicates in milliseconds for how long the LED will be ON. When “traffic light colours LEDs” are fitted (by default) the LED names and colours are mapped as follows:

- LED1 - Red
- LED2 - Yellow/Orange
- LED3 - Green

UI= ***Standard Command***

User interface pattern. Parameters – 3 bytes in HEX format.

First byte: UI bitmap to describe the UI pattern.

LED/buzzer pattern
Bit 0 - 0 for green light off, 1 for on
Bit 1 - 0 for solid, 1 for flashing green
Bit 2 - 0 for amber light off, 1 for on
Bit 3 - 0 for solid, 1 for flashing amber
Bit 4 - 0 for red light off, 1 for on
Bit 5 - 0 for solid, 1 for flashing red
Bit 6 - 0 for buzzer off, 1 for on
Bit 7 - 0 for solid, 1 for intermittent buzzer

Second byte - Number of cycles

Third byte - Interval in milliseconds

$$\underset{A0}{\underline{\overset{\underline{1}\ \underline{0}\ \underline{1}\ \underline{0}\ \underline{0}\ \underline{0}\ \underline{0}\ \underline{0}}{B7\ B6\ B5\ B4\ B3\ B2\ B1\ B0}}}\ \underset{03}{\underline{\overset{\underline{00000011}}{Number\ Of\ Cycles}}}\ \underset{32}{\underline{\overset{\underline{0010010}}{Interval\ In\ Mili\ Seconds}}}$$

Example:
UI=A00332
will instruct the reader to produce intermittent beep and flash the red LED at an interval of 50ms.

LED= *Standard Command*

This is a fixed length command and values are in decimal format. If LED=1, the middle LED is switched ON. If LED=0, the LED is switched OFF.

LDEF= *Persistent Parameter*

Default LED status. This command controls the state the LEDs will be reverted to after UI or LED[1-3] commands. It is a persistent parameter if used as LDEF= XX. To set as a volatile parameter, use LDEF=XX,1.

Bit 0 (LSB) – LED 1
Bit 1 (LSB) – LED 2
Bit 2 (LSB) – LED 3
Example: LDEF=07 – All LEDs are on.

* If the LED= command has been actioned, then the LEDs won't have the prescribed LDEF state until next UI or reboot.

BEEP= *Standard Command*

Activate the beeper on the reader.

Example:
BEEP=1 produces a short beep.
BEEP=0 produces a long beep.

SIL= *Persistent Parameter*

Switches on or off the “Silent mode” which disables or enables the use of the factory-default LED and beeper sequences during operation.

Example: SIL=1 will tell the reader not to perform the default beep on power up or when server connection is lost.

RLY= *Persistent Parameter*

This is a fixed length command and values are in decimal format. If RLY=1, the relay is enabled and can action GRNT commands. RLY=0 disables the use of the relay.

GRNT= *Standard Command*

Set the reader to a “grant access” state. The orange LED will be set to ON for *xx* seconds. If the relay is set to “active” then the coil will be powered to engage the relay from NO state to NC state for *xx* seconds and return to NO state. This is a fixed length data field in decimal format.

Example:GRNT=04 sets the relay to NC state and orange LED to ON for four seconds.

DENY= *Standard Command*

Set the reader to a deny access state –the relay is switched in NO state.

DEFRLY= *Persistent Parameter*

This is a fixed length command and values are in decimal format. If DEFRLY=1, the relay’s default mode is Normally Open, if DEFRLY=0 the default is Normally Closed. Set the defaults during reader power up.

System Commands

CK= *Standard Command*

Set the date and clock of the reader, the command takes data formatted as [YYYY-MM-DD HH:MM:SS], The year must be between 2000 and 2099. This is a fixed length data field in decimal format.

The clock data is not updated when the reader loses power unless the optional clock battery is populated.

CCAL= *Standard Command*

Clock calibration. Decimal with plus or minus sign.

Examples:
CCAL=-1500
This command tells the reader to skip one second at every 1500 seconds - backwards calibration.

CCAL=3400
The reader will add one second every 3400 seconds - forwards calibration.

PBKT= *Standard Parameter*

This is a fixed length command and values are in decimal format. It specifies the pass back time in milliseconds, or the period between the reader responding to the same card. E.g. PBKT=1000 will send requests every 1000 milliseconds (1 second) when a card is kept constantly in the field.

SID= *Standard Parameter*

This command sets the current session ID - fixed length command and values are in hexadecimal format.

MD5= *Persistent Parameter*

This is a fixed length command that sets the MD5 secret key in hexadecimal format. The key can be used to validate reader responses' authenticity.

RBT= *Standard Command*

Orbit IP can be rebooted through the web interface.

The RBT command is to be included in the webserver response to the 'PG' ping request from the reader. A random number generated by the reader is included in the parameters of the 'PG' request. The server has to send the RBT command with a value set to the MD5 digest of the concatenated random number and the MD5 secret previously configured for the reader.

This is an example of how to calculate and format the RBT parameter in PHP:

```
$rn=pack("H*",$_GET["rn"]);
$rbt=md5($rn .pack("H*", $md5key));
echo "RBT=".$rbt."\n";
```

When the reader receives the response, it checks the RBT parameter against its internally calculated digest and if a match is found then the reader performs the reboot.

COUT=

Standard Parameter

Enables or disables the Card Out command. Command is sent when a card removal is detected by the reader. Disabled by default.

Example:
COUT=1

CO_INPUTS=

Standard Parameter

Enables or disables reader inputs CON4 and CON5 status to be sent with "CO" and "COUT" commands as contact1, $contact2. Disabled by default.

Example:
CO_INPUTS=1

PG Command

In almost all cases, the reader has to start the communication with the server, and the server then can request it executing some commands. However, there is a single exception for this, the Server/ User can request a "PG" request in which the server can execute any commands it desire.
This request can be requested using ping utility, or by sending a special TCP packet at port 6320 of the reader. That packet shall contain the following hex stream: $a5\ 05\ fb\ 00\ 00\ dd\ 37$. There are multiple tools available for sending TCP packets, a free open source example is *Packet Sender*.

See pmeth.

Offline Mode Commands

This section is applicable only to Orbit IP models with installed MicroSD card and firmware version 1.7.1 and up. Older versions do not offer this functionality.

Offline behaviour

Upon every card swipe, Orbit IP tries to establish a connection with the host. If offline mode is enabled (the OFLE parameter is set to 1), the reader will try to establish a connection for a period of time (the duration is “HRTR times HRTM”, see the parameters’ section below for details). If the connection fails, or it is refused by the host, the reader switches to offline mode.

In offline mode, the reader begins to log all card swipes on its internal MicroSD card*. The reader no longer attempts to connect to the host. The behaviour of the beeper and LEDs is determined by the OFUI parameter.

Orbit IP can return to online mode only when prompted by the host, in the form of a ping message. Once pinged, the reader looks for an RBM command – if it is set to 1, it proceeds to send all logged swipes in a quick succession as “CB” requests to the host. CB requests are identical to CO requests, but indicate stored, rather than live, card swipes. When all records from the log are sent, the reader switches back to online mode.

Note that Orbit IP has the ability to read cards while transmitting CB requests. While the two processes run simultaneously a slight decrease in card read speed can be experienced temporarily.

OFLE=

Persistent Parameter

Offline mode enable/disable. If this flag is cleared then the reader will never enter offline mode even if the server is down. By default, offline mode is NOT enabled.

Example:

To enable offline mode:
OFLE=1
To disable offline mode:
OFLE=0

* The SD card content is synched every 4 card taps or every 60 seconds.

RBM= *Standard Parameter*

Enable (x=1) or disable (x=0) the sending of CB requests following a ping. Insert this command in any code processing ping requests (cmd=PG).

OFUI= *Persistent Parameter*

User interface pattern for offline operation only. Parameters – 3 bytes in HEX format.

First byte: UI bitmap to describe the UI pattern.

LED/buzzer pattern
Bit 0 - 0 for green light off, 1 for on
Bit 1 - 0 for solid, 1 for flashing green
Bit 2 - 0 for amber light off, 1 for on
Bit 3 - 0 for solid, 1 for flashing amber
Bit 4 - 0 for red light off, 1 for on
Bit 5 - 0 for solid, 1 for flashing red
Bit 6 - 0 for buzzer off, 1 for on
Bit 7 - 0 for solid, 1 for intermittent buzzer

Second byte - Number of cycles

Third byte - Interval in milliseconds.

Example:
OFUI=A00332
will tell the reader to produce intermittent beep and flash the red LED at an interval of 50ms.

LOFL= *Persistent Parameter*

This is a persistent parameter in HEX format. It sets the default standby status of the LEDs during offline operation. 1 byte parameter in HEX – Bitmap of the LED status – 1 is ON.

Bit 0 (LSB) – LED 1
Bit 1 (LSB) – LED 2
Bit 2 (LSB) – LED 3

Example
LOFFL=07 – All LEDs are on.

Network Configuration Commands

ROOT= *Persistent Parameter*

Change the root page name to be accessed by the reader. This is a field of up to 8 characters in an alphanumeric format, stored in persistent memory. Use ROOT=000000000 to reset configuration to default page.

EXT= *Persistent Parameter*

File extension parameter. Set x to select:

x	File extension
0	.php
1	.asp
2	.cfm
3	.pl
4	.htm
5	.html
6	.aspx
7	.jsp

DHCP= *Persistent Parameter*

Boolean parameter that controls the DHCP feature of the reader.

Use DHCP=1 to enable and DHCP=0 to disable the DHCP option.

If DHCP is enabled then the reader sends requests to DHCP server to acquire the following:

- Host IP
- Netmask
- Default gateway
- www-server IP

IP= *Persistent Parameter*

This command set the IP address of the reader. It is a fixed length command and values are in decimal format.

Example:IP=192.168.007.200

GW= *Persistent Parameter*

This command is used to configure the default gateway (router) of the reader. This is a fixed length command and values are in decimal.

Example:GW=192.168.007.001

NM= *Persistent Parameter*

This command used to configure the subnet mask of the reader. It is a fixed length command with values in decimal.

Example:NM=255.255.255.000

WS=

Persistent Parameter

This command is used to configure the server IP address to be accessed by the reader. It is a fixed length command with values in decimal.

Example:WS=192.168.007.012

PT=

Persistent Parameter

This command is used to configure the port number to be used to access the web server. It is a fixed length command with values in decimal.

Example: P=00080

WN=

Persistent Parameter

This command is used to set a server name in the Host field of the HTTP request. Maximum domain name length is 16 chars, including dots. Total name length including subdomains and dots is 24 chars. Ensure you set up the DNS parameter and provide DNS network functionality so the reader can communicate with the host.

Example:WN=testserver.org

DNS=

Persistent Parameter

Boolean parameter that controls the DNS feature of the reader.

Use DNS=1 to enable and DNS=0 to disable the DNS option.

If DNS is enabled, the reader connects to 8.8.8.8, enquires about the IP address of the stored web server name, and use the value as its web server IP address parameter.

HB=

Persistent Parameter

Set the heartbeat rate of the reader, in seconds. It is a fixed length field in decimal format.

Example:HB=0200 configures the reader to send HB request to the HTTP server every 200 seconds.

RTR=

Persistent Parameter

Sets ARP and TCP timeout, in milliseconds. This is handled at IP level. Takes values from 0 to 0xFFFF. By default (set via jumper) it is set to 2000ms. The minimum value guaranteed by the firmware is 500ms. After firmware update the value of this parameter is unknown – the user should set this parameter through the 'PU' request/response.

Example: echo "RTR=2000\t";

RCR=

Persistent Parameter

Sets the number of retransmissions (at ARP and TCP level) if the time as set with RTR expired before a response was received. Takes values from 0 to 99. The minimum value is 2. The default (set via jumper) is 3.

Example: echo "RCR=3\t";

HRTM=

Persistent Parameter

Sets HTTP request/response(this involves the exchange of several TCP packets) overall timeout. Default - 6 seconds. Min is 5 seconds.

Example: echo "HRTM=6\t";

HRTR=

Persistent Parameter

Sets the number of failed HTTP requests in a row after which the reader will assume web-server is down. Default is 3, minimum is 2.

Example: echo "HRTR=3\t";

Card Commands

MAUTH= *Persistent Parameter*

Used to configure card reading modes. This is a fixed length command.

- Use MAUTH=0 to disable Mifare Classic data block reading, i.e. authentication is disabled – LOOSE mode.

- Use MAUTH=1 to read Mifare Classic block of data using the key specified by MREAD. If authentication was successful, the block data and the card UID is sent to the server. If authentication failed, only the UID is sent to the server.

- Use MAUTH=2 to read Mifare Classic data block using the key specified by MREAD. If authentication was successful, the block data and the card UID are sent to the server. If authentication failed, then no request is sent to the server.

- Use MAUTH=3 to read NTAG213/215/216 that has no password protection of its pages.

- Use MAUTH=4 to read NTAG213/215/216 page of data using the password specified by NPWD. If authentication was successful, the page data and the card UID are sent to the server. If authentication failed, only the UID is sent to the server.

Along with the above options the user can create new ones by using the following pattern:

$$MAUTH = \frac{0}{B7}\frac{0}{B6}\frac{0}{B5}\frac{0}{B4}\frac{RSV}{B3}\frac{RSV}{B2}\frac{RSV}{B1}\frac{RSV}{B0}$$

B7: Authenticate with NTAG cards (1 or 0)
B6: Authenticate with Mifare cards (1 or 0)
B5: Authenticate with NTAG password (1 or 0)
B4:Return UID on authentication failure (1 or 0)
B(3-0): Reserved.

For instance, if we want to read NTAG along with Mifare cards, and the reader shall use the defined password for NTAG cards, and it shall return UIDs even when authentication is failed then the user can use MAUTH=240 (decimal of MAUTH=0b11110000.)

Please note that by default the reader would store NTAG password in Key A, so if using both cards, Mifare keys has to be stored in Key B.

Mifare Classic Specific Commands

MKEY=Nxxxxxxxxxxxx *Persistent Parameter*

Use to configure Mifare Classic authentication key to read data blocks. This is a fixed length command and values are in hexadecimal format.

N is either 'A' or 'B' to specify the key type, xxxxxxxxxxxx is the value of the key as 12 hexadecimal digits.

For example: MKEY=ACB1234567890 will set KEYA to value CB1234567890 (hex).

This command is not encrypted and it is recommended to transmit the key in secure and controlled environment. Do not set the key over a public network.

MREAD=Nxx *Persistent Parameter*

Use to read a data block from a MIFARE Classic card. This is a fixed length command and values are in hexadecimal format.

N is either 'A' or 'B' to specify the key type.XX is the block number to be read in hexadecimal format.

For example: MREAD=A0D will set the reader to read block number 13 (decimal) using KEY A.

NTAG 213/215/216 Specific Commands

NSTART=xx *Persistent Parameter*

Use to define the starting page to be read from a tag. This is a fixed length command and values are in decimal format.

NEND=xx *Persistent Parameter*

The end page number (inclusive) to be read from a tag. The reader will concatenate all page data into a single string and report it back to the webserver in the request parameter. This is a fixed length command and values are in decimal format.

NPWD = xxxxxxxx *Persistent Parameter*

Set password for authentication to the tag. 4 byte integer stored in memory, LSByte first.

NPACK=Nxx *Persistent Parameter*

Set password acknowledgement value. When password authentication is activated the reader checks the PACK against the configured value in order to read and return data to the webserver.
2 byte integer stored in memory, LSByte first.

*Page numbers checks are not performed at this stage.

DESFIRE specific commands

Orbit IP implements Desfire card data read using pre-configured communication parameters.

DF_AID=

Persistent Parameter

Application ID on the card.

DF_KEY=

Persistent Parameter

Concatenated Desfire key type (1 byte) and key value (N bytes):
Key types are defined as:

0	AES 128 Key [16 Bytes]
1	AES 192 Key [24 Bytes]
2	AES 256 Key [32 Bytes]
3	DES Single Key [8 Bytes]
4	2 Key Triple Des [16 Bytes]
5	3 Key Triple Des [24 Bytes]

DF_KEYN=

Persistent Parameter

Key number on the card. For plain communication (DF_COMM=0) set DF_KEYN=14.

DF_COMM=

Persistent Parameter

Communication method defined as:

0	PLAIN
1	MAC
2	ENCRYPTED

DF_FID=

Persistent Parameter

File ID on the card.

DF_FT=

Persistent Parameter

File type defined as:

0	STANDARD DATA
1	BACKUP DATA
2	VALUE WITH BACKUP
3	LINEAR RECORD
4	CYCLIC RECORD WITH BACKUP

DF_DLEN=

Persistent Parameter

Length of data to be read from the file or the record size in the cases when the file is of record type.

DF_OFF=

Persistent Parameter

Offset in the file.

DF_NREC=

Persistent Parameter

Number of records to be read from the card. Applies in the cases when the file is of record type.

ISO14443 Part 4 Compliant Commands

JC_AID=

Standard Parameter

Sets the Applet ID which the reader would automatically * select when a P4 smart card is detected.

JC_APDU=

Standard Command

Execute a single APDU and return its response in R_APDU request.

* When a smart card is detected, the discovery loop is paused after successful selection of the given applet, and the reader would wait for APDU commands from the host or for the card removal before continuing the discovery loop.

Example

Sample PHP code for Desfire configuration:

`echo"DF_AID=A4C360\n";`	Set DF Application ID
`echo"DF_KEY=003638EDF8BA4B72825021C2347C86554F\n";`	Set Key type to AES128, and the key
`echo"DF_KEYN=2\n";`	Set the Key number to 2
`echo"DF_COMM=2\n";`	Select encrypted communication method
`echo"DF_FID=0\n";`	Select File ID to 0
`echo"DF_FT=0\n";`	Select File Type to Standard Data
`echo"DF_DLEN=4\n";`	Set Length of data to be 4
`echo"DF_OFF=0\n";`	Set Offset in Desfire file to be 0

PHP Code Example:

```
<html><body>
<?php
$mycard=EA8740BF;
$st=date('Y-m-d H:i:s',time());
$cmd=$_GET["cmd"];

echo"<ORBIT>";

switch($cmd){
case"PU":
echo"CK=$st\n";
echo"MKEY=BA637F128CCDC\r\n";
echo"MREAD=B04\r\n";
echo"MAUTH=1\r\n";

echo"MD5=53306C757421306E\n";

echo"SID=ABBA0123\n";
echo"WS=192.168.007.191\n";
echo"EXT=0\t";
echo"HB=10\t";
echo"RLY=1\t";

break;

case"CO":
$uid=$_GET["uid"];
if($uid==$mycard)
{
echo"BEEP=1\t";
echo"GRNT=05\t";
}
else{
echo"BEEP=0\t";
echo"LED1=2000\t";
echo"DENY\t";
}
break;

case"HB":
echo"CK=$st";
break;
case"PG":
break;

case"SW":
break;

}
echo"</ORBIT>";

?>

</body></html>
```

This is an example of PHP script hosted at the HTTP server. The filename is *orbit.php*.

In the example, we use a switch case to detect different types of request from the reader and respond with the requested behaviour accordingly.

For instance when a reader sends a ‘PU’ request to the server (upon power-up), The server can respond with Initialization sequence.

Or When we detect a card, we check if it matches some defined card UID, then we switch on the relay for 5 second while if it's not then we warn the user

And we can recalibrate the clock on every hear beat

Please note the MD5 key, Card keys, and other sensitive data commands must be performed only in secure environment, for example use a WWW server accessible via LAN only and no other devices connected to the LAN switch but the server and the reader being configured.

This configuration task may be automated by implementing requests to a database to extract predefined MD5 keys and configure the relevant readers based on their MAC address.

Example of a request sent to the server upon 'ping'

```
"GET /orbit.php?cmd=PG&sid=&dhcp=0&dns=0
&ws=192.168.7.216&pt=80&gw=192.168.7.1&nm=255.255.254.0&mac=00:08:DC:E4:98
:E9&psrc=192.168.7.243&pmeth=TCP&relay=1&sd=1&date=2019/01/11&time=11:30:1
6&md5=CBBF9AAED344581004AFFF4715A54326&contact1=1&contact2=1&ver=2.2.9&id=
192.168.7.200&rn=F033B5A3E3970FB0& HTTP/1.0" 200 253 "-" "ORBIT-HTTP-
CLIENT "
```

Example of a request sent to the server upon card detection

```
"GET
/orbit.php?cmd=CO&id=192.168.7.219&sid=&uid=0E980F53&ulen=4&date=2018/12/0
6&time=09:12:36&md5=35EBCB79E169D889E638AEFE8EB715C6&mac=54:10:EC:9C:42:12
& HTTP/1.0" 200 354 "-" "ORBIT-HTTP-CLIENT"
```

Example of a request sent to the server upon reader power up

```
"GET
/orbit.php?cmd=PU&id=192.168.7.244&mac=54:10:EC:9C:5F:B9&ver=2.2.0&blver=0
0000201&sd=0& HTTP/1.0" 200 433 "-" "ORBIT-HTTP-CLIENT"
```

Example of a regular heartbeat request sent to the server

```
"GET /orbit.php?cmd=HB&id=192.168.7.227&RE=0&mac=54:10:EC:9C:73:9E&
HTTP/1.0" 200 337 "-" "ORBIT-HTTP-CLIENT"
```

UDP Messages Mode

Alongside the normal HTTP client operation mode, the reader can operate in UDP client mode sending UDP messages to a UDP server similar to the operation mode of X1010 readers .

The following is the list of all possible UDP messages in the latest firmware and their formats:

SN message

This message is sent when the reader detect a new card in its field and its format is as follows:

SNUID,MAC $e.g ==>$ SN7ABAFBCF,0008DC70C4B4

HB message

This message is sent at HB intervals and its format is as follows:

HB Counter,MAC $e.g ==>$ HB0022,0008DC70C4B4

PU message

This message is sent when the reader powers up and its format is as follows:

PU FWVersion,MAC $e.g ==>$ PU4.1.0,0008DC70C4B4

SW message

This message is sent when the reader powers up to inform the server about the initial state of the inputs and when any of the input changes and its format is as follows:

SW Contact Status,MAC $e.g ==>$ SW1.1,0008DC70C4B4

PG message

This message is sent when the reader receives an ICMP packet and its format is as follows:

PG Source IP,MAC $e.g ==>$ PG192.168.7.236,0008DC70C4B4

Bootloader Mode

Perquisites:

- Web server
- *DHCP server
- TFTP server
- The firmware binary file

Setting up the environment:

1. Obtain the required .bin file you need to upload in the reader.
2. Change the file name to **X1011_V3_L59_FW.bin** .
3. Place the file in the root directory of the TFTP server.
4. *Configure the DHCP server, and add TFTP server address to option 66:

   In "ISC DHCP server" this can be done by appending the following line to the configuration file

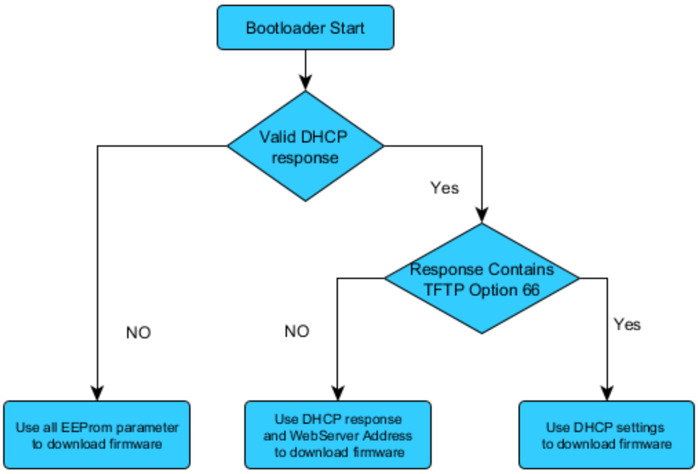
   ```
   option tftp-server-name "192.168.7.12";
   ```

   replace 192.168.7.12 with your TFTP server IP address.

* Starting from boot-loader version 4.1.0, DHCP server and option 66 are not strictly required.

If the DHCP server response didn't contain option 66, then the reader would assume that the web server address configured in firmware mode is running a TFTP server, and it would try to load the firmware from that address.

If there wasn't any DHCP server running on the network, then the reader would use static settings from firmware mode for IP and Gateway, and assume that a TFTP server is running on the web server address.


Bootloader Start
Valid DHCP response
Yes
Response Contains TFTP Option 66
NO
NO
Yes
Use all EEProm parameter to download firmware
Use DHCP response and WebServer Address to download firmware
Use DHCP settings to download firmware

Updating Firmware Using Configuration Jumper

The manual update procedure is as follows:

1. Power-down the reader.

2. Place the jumper across 1-2 pins (position BL) at CONF connector.

3. Power-up the reader.

The reader will produce a 2-tone beep and the amber LED will light up. At this stage the reader is in bootloader mode and will start DHCP discovery. It will obtain its IP address and the TFTP server IP address from the DHCP server if running.

4. The reader starts to download * **X1011_V3_L59_FW.bin** file. This activity is indicated by flashing red LED.

5. When finished the reader will start in normal application mode and will beep.

6. Power down the reader and remove the jumper from BL position. Move it as desired by your application.

7. Power-up the reader. It should be running in normal application mode.

Please note that the file name depends on the versions of the firmware and the reader:

For versions	(x < 3.0.0): X1011_firmware.bin
	(4>x≥3) : X1011_V3_FW.bin
	(x ≥4) : X1011_V3_L59_FW.bin

Updating Firmware Remotely

Orbit IP allows the activation of the bootloader through the HTTP interface.

Upon receipt of the BLREQ command the reader is rebooted in bootloader mode similar to that when the configuration jumper is set to BL position.

The BLREQ command is to be included in the web-server response to the ‘PG’ request from the reader. A random number generated by the reader is included in the parameters of the ‘PG’ request. The web-server has to send the BLREQ command with a value set to the MD5 digest of the concatenated random number and the MD5 secret previously configured for the reader.

This is an example of how to calculate and format the BLREQ parameter in PHP:

```
...
$rn=pack("H*",$_GET["rn"]);
$blreq=md5($rn . pack("H*", $md5key));
echo "BLREQ=".$blreq."\n";
...
```

When the reader receives the response it checks the BLREQ parameter against its internally calculated digest and if a match is found then the reader performs the reboot in bootloader mode.

Once the reader is in bootloader mode it will try to download a firmware binary file from a TFTP server. If no server is available then the reader will timeout and reset in normal application mode with the current firmware.

Legal Disclaimer

These materials contain confidential and proprietary information in the nature of, for example, trade secrets and know-how, and are not to be distributed or divulged to third parties, or duplicated in whole or in part without prior written permission from Gemini 2000 ltd., and are subject to use, copying, and disclosure restrictions contained in an agreement with Gemini 2000 ltd.

These materials are to be used only for the intended purpose agreed upon in the related contract with Gemini 2000 ltd. In no event shall Gemini 2000 ltd. be liable for special, indirect, or consequential damages in connection with or arising from the use of this document or any programs contained herein. Gemini 2000 ltd expressly disclaims any warranty of merchantability or fitness for a particular purpose in relation to this document or any programs contained herein.