Sensacell Module Communication Protocol- Software VER 1.0.1 **06.23.09**

For Model HSI64-36-RGB Modules

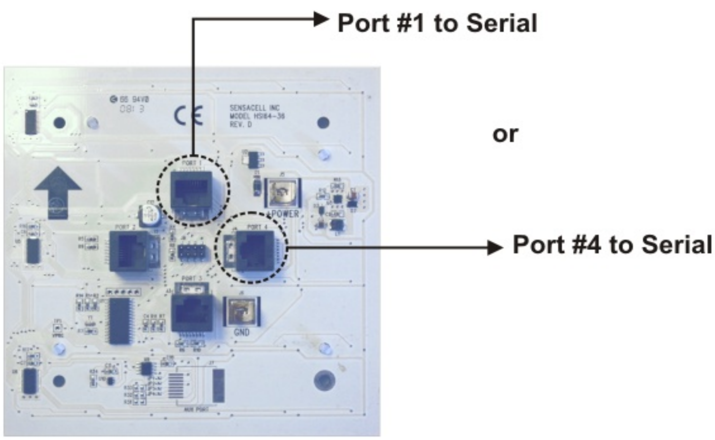
Leo Max Fernekes, Sensacell Inc. Revision A

Physical protocol:

RS-485 Half-Duplex, 2-Wire interface, each Module is an independent node in a multi-drop network.

- **8 Data Bits**
- **1 Start Bit**
- **1 Stop Bit**
- **No Parity**
- **No Handshaking**
- **Baud Rate = 230,400 Baud**

Serial Interface Electrical Connections- RJ45 connector, Port #1 or Port #4



PIN 1: DATA‘A’ RS-485 +
PIN 4: GND
PIN 5: GND
PIN 8 DATA ‘B’ RS-485 -

- Best practice is to connect the serial connection to the MASTER MODULE, as data lines are terminated with 120 ohm resistor and idle state bias at MASTER MODULE, but any module in the array can be a connection point.

- Note that Port #2 and Port #3 can also be used, but DATA connections A and B must be reversed.

- Use care when making connections to avoid short circuits or connections to any other pins.

Array Startup Sequence

On power-up, an array of Sensacell modules begins a series of operations that serve to verify the state of the array and establish a unique physical address for each module.

The startup sequence consists of 5 distinct phases:

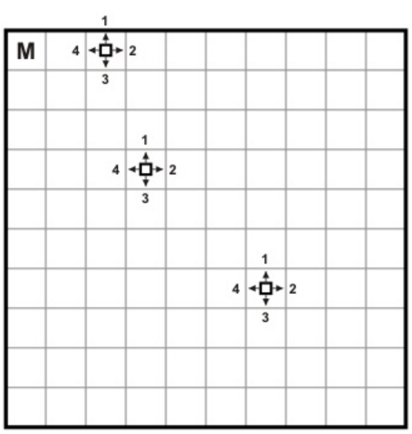
1. **Port-Ping Mode:**
   Establishes the state of all inter-module connections created by placement of jumpers.

2. **Display Connections Mode:**
   Displays the state of all inter-module connections using the LED display on each module.
   This mode allows for quick diagnosis of missing or defective jumpers, or improperly oriented modules.

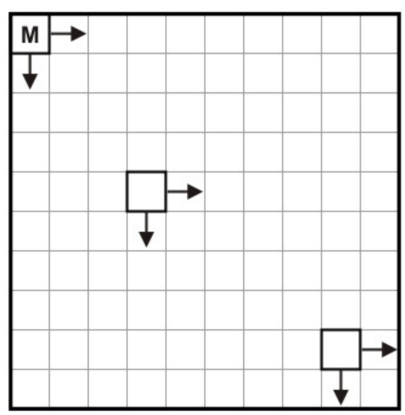
3. **Auto Addressing Mode:**
   Establishes a unique physical address for each module in the system.
   This process is done in two steps, the first step attempts to number the modules in a top-to-bottom, left-to-right order, assuming the Master Module is located in the upper left corner.
   Step two re-scans the array for orphan modules not addressed during the first pass.

4. **Display Address Mode:**
   The modules display their assigned addresses in binary format to visually validate the addressing.
   The display shows the row, column, and physical address sequentially at one second intervals.

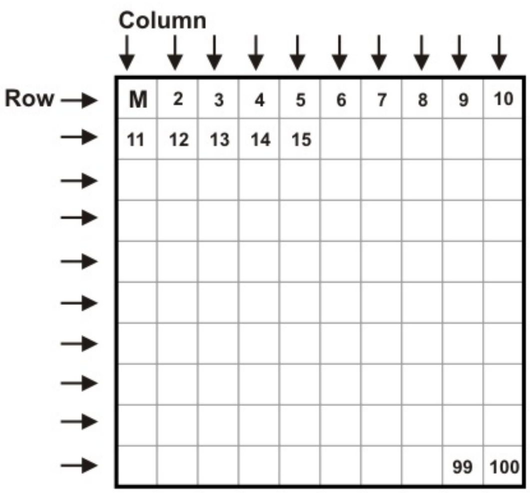
5. **Run Mode:**
   Normal operation. Sensor synchronization signals are propagated throughout the array.



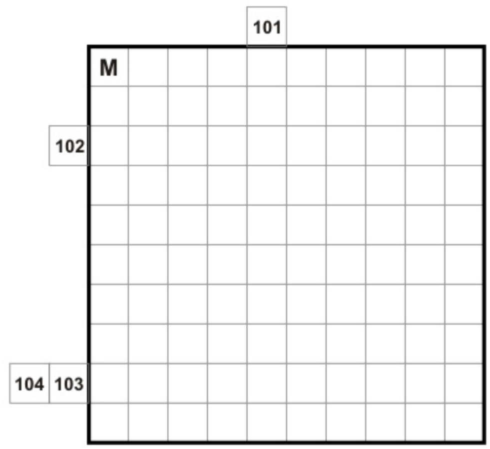
Port - Ping
Identifies valid connections between modules.



Sensor Sync
Establishes operating sensor synchronization.

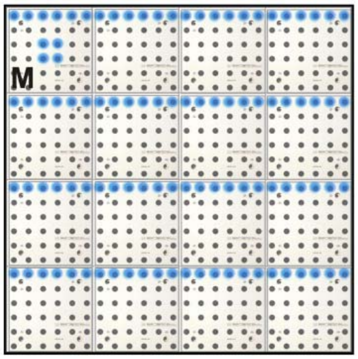


Auto-Address 1st Pass
Addresses modules in a rectangular grid - Left to Right and Top to Bottom.

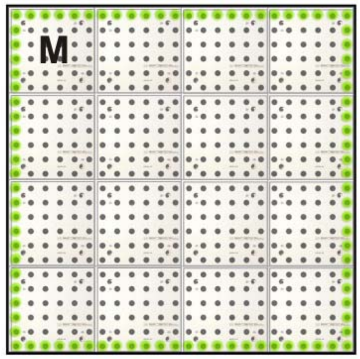


Auto-Address 2nd Pass
Addresses modules outside of primary grid.

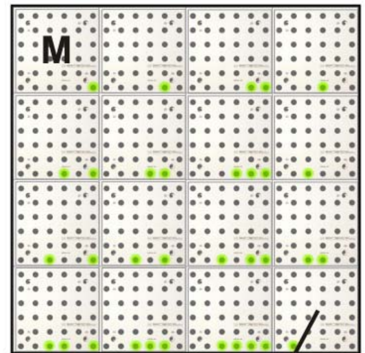
LED display during normal startup:



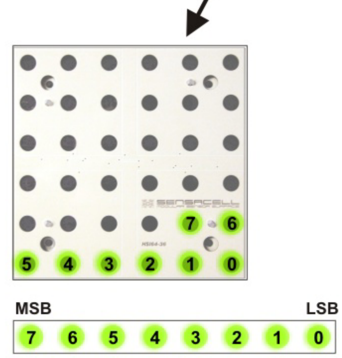
1. Power up



2. Display connections



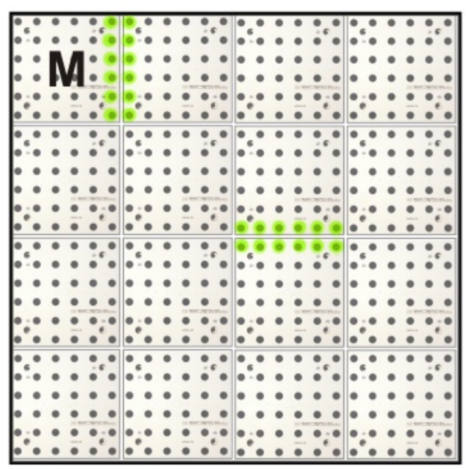
3. Display module address



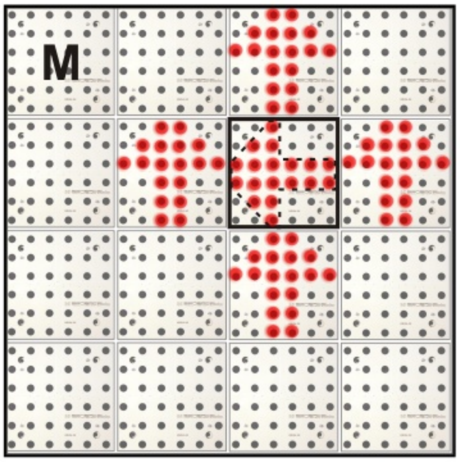
8 Bit binary address

Finding faults by observing the LED display during startup:

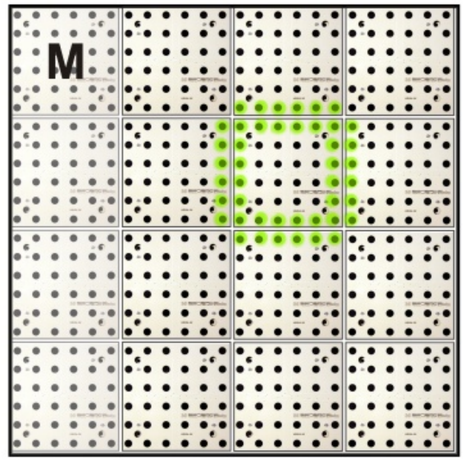
A. Master module not connected



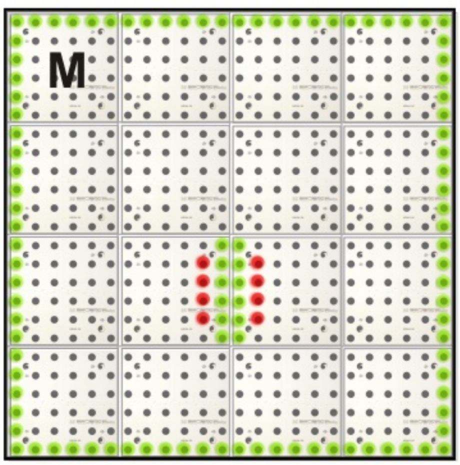
B. Missing or failing jumper(s)
Double lines indicate between which modules



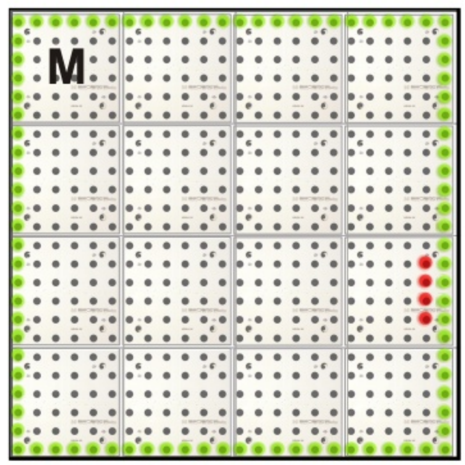
C. Module is turned 90 degrees
Arrow in the middle indicates in which direction.



D. Module is turned upside down



E. Port stuck low
Shorted jumper



F. Port stuck low
Bad module

Non-connected ports are represented by a green bar displayed along the edge of the module; this feature creates a ‘frame’ displayed around the entire array. Missing or failed jumpers can be quickly found and replaced by observing the startup process. Modules that have been installed sideways or upside-down are also easily found.

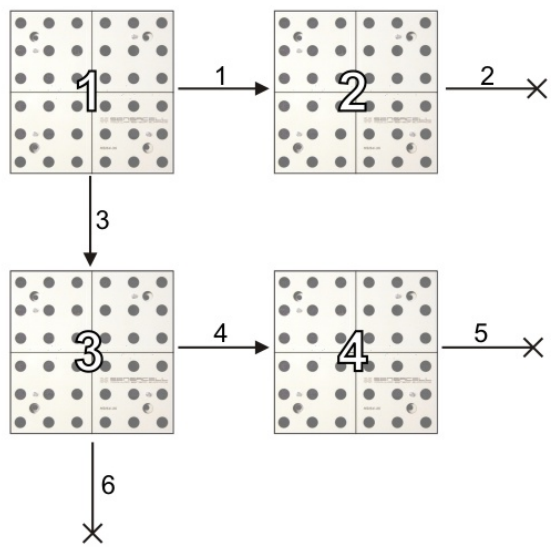
Auto-Addressing

Auto-Addressing automatically begins after Port-Ping with the MASTER module, and continues until all modules in the array have been assigned a unique physical address. This process will repeat every time the array is powered up, and will always be the same unless the array is physically reconfigured. During auto-addressing, the array will emit a series of codes that can be used to validate the integrity of the array; these codes also contain enough information to re-create a symbolic representation of any arbitrarily shaped array.

Two addressing modes are employed; Mode One (1) attempts to number the modules in a rectilinear array, Top-to-Bottom, Left-to-Right. (Assuming that the Master Module is in the upper left corner of the array, with arrows up) Mode Two (2) then takes over using the branching tree logic to detect any “orphan” modules that were not numbered by Mode One. This system creates an orderly, “programmer friendly” address pattern from rectangular arrays, while also allowing arbitrarily shaped arrays to successfully auto number. Mode Two (2) searches for open ports in the order 1,2,3,4. Mode 2 follows all branches until there are no modules left to number.

Auto-Addressing Example

81010101 Port-Ping
82010101 Port-Ping
83010101 Port-Ping
84010101 Port-Ping
81010101 Port-Ping
82010101 Port-Ping
83010101 Port-Ping
84010101 Port-Ping
81010101 Port-Ping
82010101 Port-Ping
83010101 Port-Ping
84010101 Port-Ping
81010101 Port-Ping
82010101 Port-Ping
83010101 Port-Ping
84010101 Port-Ping, 16 times.
02010101 Master begins addressing, passing to the right (1)
10010202 Module #2 announces end of row (2)
23010101 Master passes to next row down (3)
02020103 Module #3 passes to the right (4)
10020204 Module #4 announces end of row (5)
30020103 Module #3 no available ports- end of MODE 1 (6)
42010101 Master starts mode 2, searching for open ports
43010202 Module #2 passes down to Module #4
44020204 Module #4 passes to the left, to Module #3
40020103 Module #3 announces no available ports
70010104 Module #4 sends end-of-addressing command



The sequence above shows the steps taken to address a four module array, as shown on the right.

Auto-Addressing Protocol

- The addressing protocol consists of five data fields represented by 8 hexadecimal characters, followed by a carriage return character (0x0D)
  The fields are: **COMMAND, DATA, ROW, COL and ADDRESS - CDRRCCAA** (cr)
- All values are in HEXADECIMAL. Note that all HEX letters are UPPER CASE.
- **COMMAND** indicates the type of action taken.
- **DATA** indicates the port number (direction) of the action taken.
- **ROW** is the modules 8 bit **row address**
- **COL** is the modules 8 bit **column address**
- **ADDRESS** is the modules 8 bit **logical address**
- The start address is always = 01, a single Master Module will have address = 01
- Row, column and address values always represent the module that transmitted the command.
- The end of the auto addressing process is indicated by transmission of **70RRCCAA**, where **AA** is the address of the last and final module in the array.

Auto-Addressing Format: 'CDRRCCAA<cr>'

Where C =	**"Command"**	**(0-F)**
Where D =	**"Data"**	**(0-F)**
Where RR =	**"Row Address"**	**(00-FF)**
Where CC =	**"Column Address"**	**(00-FF)**
Where AA =	**"Module Address"**	**(00-FF)**

Command:

0 =	**Mode 1, New module address**
1 =	**Mode 1, End of row signal**
2 =	**Mode 1, Restart numbering, row master handoff**
3 =	**Mode 1, Row Master has no ports- end of Mode 1**
4 =	**Mode 2, Searching for open ports**
5 =	**Mode 2, New module address**
6 =	**Mode 2, Restart numbering**
7 =	**Mode 2, Master announces end of Mode 2, auto-addressing completed.**
8 =	**Port-Ping Mode**

Data, Vector taken:

0 =	**No available Ports**
1 =	**Address Port 1, UP**
2 =	**Address Port 2, RIGHT**
3 =	**Address Port 3, DOWN**
4 =	**Address Port 4, LEFT**

Large Scale System Architecture

Large scale systems capable of displaying full motion video graphics can be created using the Sensacell system.
Certain design criteria must be considered when designing large systems.

Large-scale design considerations:

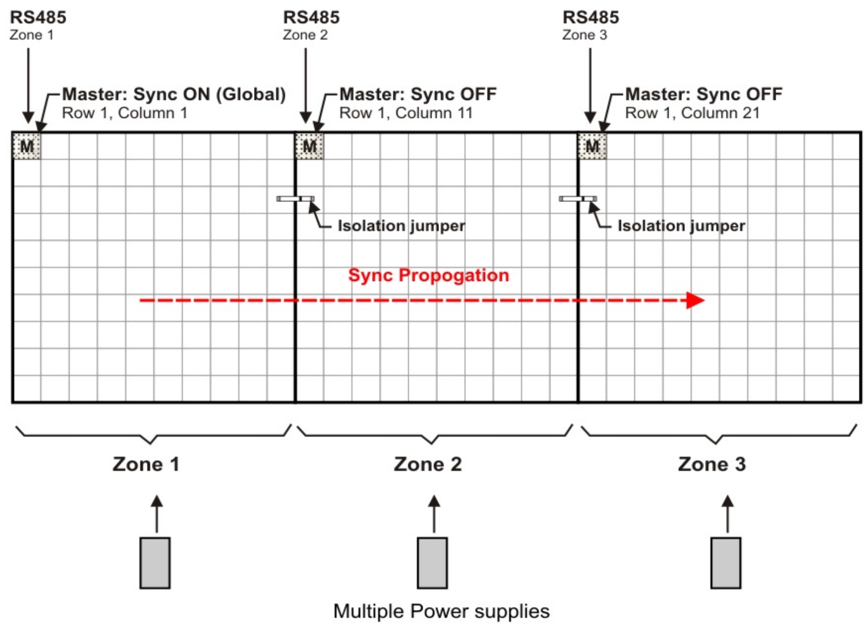
1. System communication bandwidth.
   Calculations must be made to determine the maximum time required to update all module LED's and read the sensors. This calculation determines the maximum real-time frame rate achievable. Use the appropriate display and read mode to make most efficient use of the available communications bandwidth.

2. Physical communications limits.
   The RS-485 network is physically limited to a maximum of 320 modules by electrical loading.

3. Addressing limitations.
   The module addressing protocol is limited to 255 modules maximum.

4. Electrical power distribution factors.
   Large arrays of modules require large amounts of current to operate; powering the entire array using a single circuit becomes impractical.

All the above problems become manageable by breaking the main array into a series of smaller sub-networks, each with a separate, isolated RS-485 communication network and power distribution wiring, and power source. This distributed architecture effectively eliminates any size constraints on the final matrix. The example below shows a large matrix of modules broken into 3 separate zones. Each zone is powered by its own power supply and communicates with the host computer via a separate RS-485 network. The three zones are connected with special isolation jumpers which isolate the power and RS-485, while allowing synchronization signals to propagate across the entire array. The only special consideration in this configuration is to ensure that the row and column addresses are maintained correctly across the entire array. Each of the three master modules must have preset starting row and column addresses preserved in non-volatile memory so that they can initiate auto addressing from the correct row and column position.



Module Running Command Protocol

Once the Auto Addressing sequence has finished, the array enters RUN mode, using a different command protocol than Addressing Mode.

Definitions:

- All commands are case sensitive, command characters are always lower case, numeric data and address values are upper case ASCII hexadecimal format.
- "**AA**" Represents an 8 bit module address, represented as two upper case ASCII hexadecimal. characters, allowing for 256 modules per network.
  Valid range 00 to FF.
- "**D**" represents an ASCII hex data character representing 4 bits of data.
  "**DD**" represents a two hex character 8 bit parameter.
- <cr> represents an ASCII carriage return character (0x0D).
- Commas are shown for clarity; they are not part of the command.
- Command delimiters are always LOWER CASE.
- An address of zero (00) addresses all modules. (Not valid for all commands)
- Some commands are valid only for MASTER MODULE, address 01.
- Addresses shown as 'AA' are valid both as GLOBAL commands (Address ='00')
  and specific individual addresses.

Reading the Sensors- Single Module

Each sensor in the array can be polled by sending a command and reading the response. New sensor data is available every 100 milliseconds.

The sensors can be read in one of two formats:

1. Digital ON / OFF- A single bit of data is returned from each sensor indicating triggered or idle state.
2. Proportional Read- 4 bits of data return a value from 0 to 15 corresponding to the degree of activity detected

Digital Read- 1 hex character representing the binary state of all sensors

Command: “Read”	**r,AA,<cr>**
Returns:	**D<cr>**

Where ‘D’ is a single ASCII hex character representing the binary state of all 4 sensors.

Note that single module Read commands are not valid as Global Commands
(address =00)

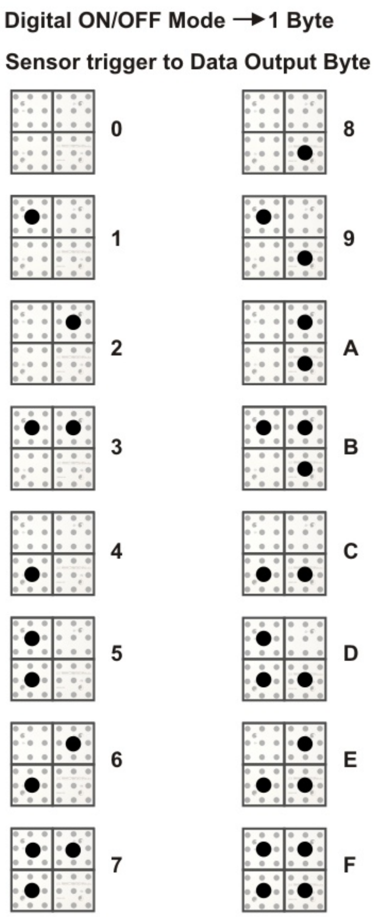


Table of HEX character values versus patterns of sensor activation from a single module

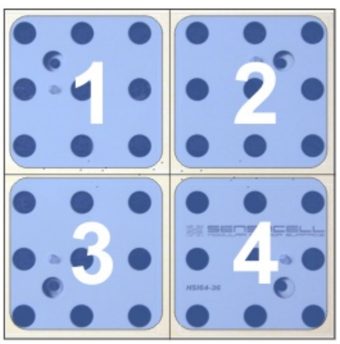
Proportional Read – returns 4 hex characters representing the linear proportional state of all four sensors.
Command: “Proportional”   **p,AA,<cr>**
Returns: **DDDD<cr>**

The data is returned from the sensors in the order **1,2,3,4**

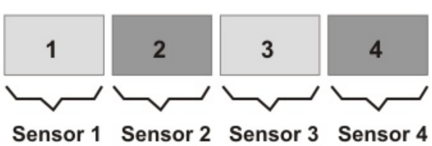
Where **‘D’** is a single ASCII hex character representing one of 16 proportional values, ranging from ‘0’ to ‘F’- ‘0’ represents no sensor activity and ‘F’ represents a full sensor response from an object very close to the module.

Note that single module Read commands are not valid as Global Commands (address =00)

Proportional Mode → 4 Bytes



4 Sensors per Module
1 Byte per sensor
32 Bits total



Proportional read data format, 0-F

Reading the Sensors- Global Read

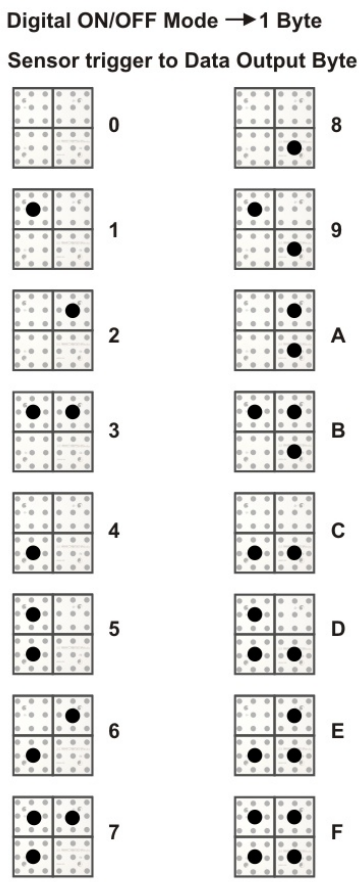
The state of all sensors in an array can be read using a single **Global Read** command for rapid polling of the entire array. Just like the single module read, a group of modules can be read in two different formats:

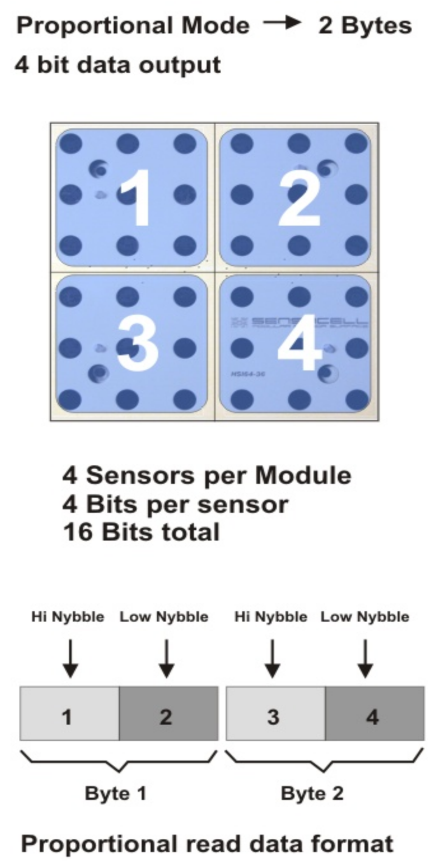
1. Binary- (default mode) returns the digital (on-off) state of each sensor pixel, based on the threshold set by the sensor sensitivity as one hex character per module.
2. Proportional- returns a 4 bit value proportional to the amount of sensor activity detected by each of the 4 sensors. Returns two bytes containing four packed nibbles, in the order 12,34.

Global Read Command- returns a string of data from a group of modules, in order of their physical address.

Command: "Global Read"	**00,NN,a,AA,<cr>**
Returns:	**DDDDDDDDDDDD…….. <cr>**
(<cr> only appended in Binary mode)	

Where **'NN'** is the number of modules to read, **'AA'** is the address of the first module to read.
Where **'D'** is a single ASCII hex character representing the binary state of all 4 sensors (binary mode) or two packed bytes containing four 4- bit proportional values. (Note that Proportional Reads return twice the number of bytes)





Sensor Control Commands

- GLOBAL READ
  00DDaAA<cr>
  Where '**DD**'= Number of modules to read, '**AA**' specifies the starting module address to read.
  This command causes each module to return its sensor state in the order of their physical address.
  You can read sub-sections of the array by specifying an address >1 and the number of modules to read.

- SENSOR RESET.
  0300aAA<cr>
  This command causes the sensor system to re-calibrate to the current conditions.
  Global address **'00'** is valid- all modules.

- SENSOR READ MODE
  0BDDa00<cr>
  Where '**DD**'= 0x01 = Proportional Mode, 0x00 = Digital Mode (default = 0x00)
  This command selects the sensor read mode
  Must be global address **'00'** for all modules.

- Load SENSITIVITY
  09DDaAA<cr>
  Where **'DD'** is the sensor threshold, valid range = 0x00-0x0F (default 0x09)
  This command sets the threshold where the sensors trigger. This trigger threshold relates to reading the sensors in Binary mode, and the trigger point for some Autonomous behaviors. This command does not affect the Proportional Read values.
  Global address **'00'** is valid- all modules.

- Load LOCKOUT TIMER HIGH
  05DDaAA<cr>
  Where **'DD'** is the high byte of a 16 bit value, each count represents 100 ms. (default value 0x0258- one minute)
  This command sets the value of the LOCKOUT TIMER, this timer controls the time that the sensors will stay triggered before they time-out and reset when an object remains in the sensing zone.
  Global address **'00'** is valid- all modules.

- Load LOCKOUT TIMER LOW
  04DDaAA<cr>
  Where **'DD'** is the low byte of a 16 bit value- see above.
  Global address **'00'** is valid- all modules.

- Load NEGATIVE TIMEOUT TIMER
  07DDaAA<cr>
  Where **'DD'** is a single byte of data, each count represents 100 ms. (default value = 0xB4- 18 seconds)
  This command sets the value of the NEGATIVE TIMEOUT TIMER, this timer controls the time it takes for a negative sensor input to trigger a sensor reset. Negative sensor signals occur when an object that was present for a long period of time is suddenly removed.
  Global address **'00'** is valid- all modules.

- Load INTEGRATION COUNTER
  08DDaAA<cr>
  Where **'DD'** is a single byte of data, each count represents 100 ms. Valid range- 0x01-0x7F
  (default value = 0x14- 2 seconds)
  This command sets the value of the INTEGRATION RATE COUNTER; this counter sets the number of cycles that occur between adjustments of the tracking baseline value.
  Global address **'00'** is valid- all modules.

- Load TRACKING TRIGGER MASK
  06DDaAA<cr>
  Where **'DD'** is a single byte of data, used as a bit mask. (default 0xF8- '11111000')
  This command sets the value of the TRACKING TRIGGER MASK, used to mask sensor output signal to determine if signal is within the tracking band. Bits set to '0' are ignored, i.e. '11111000' = anything under the value of 8 is within the tracking band.
  Global address **'00'** is valid- all modules.

- Load PROPORTIONAL READ SENSITIVITY
  0ADDaAA<cr>
  Where **'DD'** is a single byte of data, valid range = 0x01-0x08 (default value = 0x03)
  This command sets the value of the proportional read sensitivity. The value is inverse to the sensitivity, i.e. 1 is the most sensitive.
  Global address **'00'** is valid- all modules.

System Commands

- SYSTEM RESET
  13EAa00<cr>
  Data field must be equal to '**EA**'
  This command causes the module to perform a soft reset- equivalent to power cycling the array.
  Must be global address **'00'** for all modules.

- SENSOR DIAGNOSTICS
  15DDaAA<cr>
  Where '**DD**'= 0x01 = Diagnostics ON, 0x00 = Diagnostics OFF (default = 0x00)
  This command causes the module to continuously output sensor diagnostic information.
  Not valid as Global Command- one module only.

- WRITE EEPROM (Master Module only)
  14EAa01<cr>
  Data field must be equal to '**EA**'
  This command causes the module to write it's current row and column address along with CLOCK GEN status to EEPROM
  Not valid as Global Command- Master Module only.

- MASTER MODULE CLOCK GEN
  16DDa01<cr>
  Where '**DD**'= 0x01 = CLOCK ON, 0x00 = CLOCK OFF (default = 0x01)
  This command controls the Master Module sync clock output, used when combining multiple arrays with one clock source.
  Not valid as Global Command- Master Module only.

- DIAL-A-MODE DISABLE
  1700a01<cr>
  Disables demonstration 'Dial-A-Mode' for use with PC software. (Demo kit modules only)
  Not valid as Global Command- Master Module only.

Address Control Commands

- Load VIRTUAL ADDRESS
  1ADDaAA<cr>
  Where ‘**DD**’= new VIRTUAL ADDRESS, 0x01 to 0xFF
  This command loads data into the VIRTUAL ADDRESS register. Use this command to re-number an array of modules.
  Not valid as Global Command- one module only.

- Enable VIRTUAL ADDRESS
  1D00a00<cr>
  This command changes the module address to the value previously stored in the VIRTUAL ADDRESS register.
  Must be global address **‘00’** for all modules.

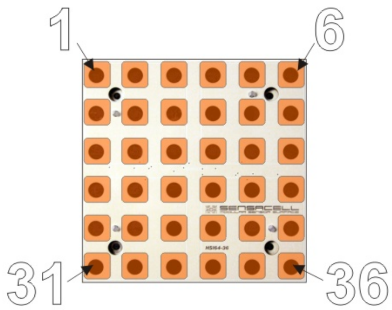
- Enable DEFAULT ADDRESS
  1C00a00<cr>
  This command restores the original default address assigned by Auto-Number.
  Must be global address **‘00’** for all modules.

- Load MODULE ROW ADDRESS
  1DDDaAA<cr>
  Where ‘**DD**’= new MODULE ROW ADDRESS, 0x01 to 0xFF
  This command loads data into the module row address. (See section about large systems)
  Not valid as Global Command- one module only.

- Load MODULE COL ADDRESS
  1EDDaAA<cr>
  Where ‘**DD**’= new MODULE COL ADDRESS, 0x01 to 0xFF
  This command loads data into the module column address. (See section about large systems)
  Not valid as Global Command- one module only.

Lighting Dimmer Control

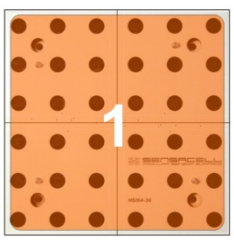
The HSI64-36-RGB module has 36 individually controllable RGB LED pixels arranged in a square grid. Each pixel has 3 independent dimmer channels that can produce 256 different brightness levels. (16 million colors) The pixels are numbered as shown, viewing the module from the front, with the module oriented with the arrows UP:



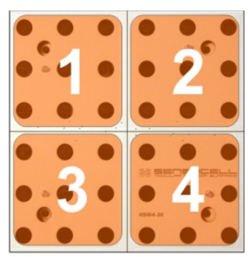
36 LED's as individual pixels
(Sensacell Module)

The modules can be configured to operate in three distinct LED control modes:

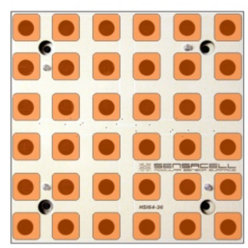
1. As a single RGB pixel- all 36 LED's display the same color and brightness level.
2. As four pixel zones of 9 LED's each- directly corresponding to the four sensor electrodes.
3. As 36 individually controllable RGB pixels.
4.



36 LED's as one pixel
(Sensacell Module)

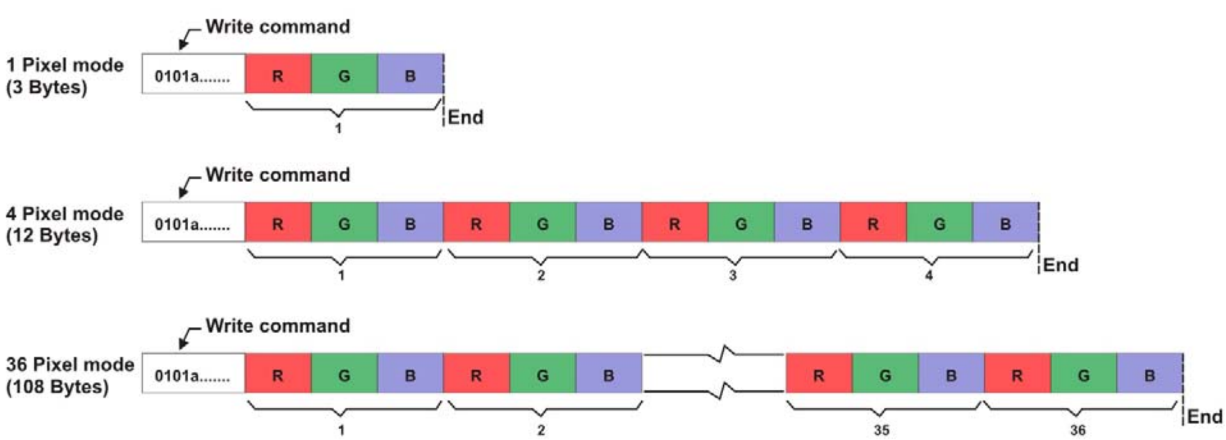


9 LED's as one pixel
(Sensacell Module)



36 LED's as individual pixels
(Sensacell Module)

In all three modes, the pixel data consists of three 8-bit (0-255) dimmer bytes sent in triplets of three bytes. Data is always sent in the order RED, GREEN, BLUE. (RGB)



Lighting Dimmer Control Commands

Using the Global Write command you can write to the entire array of modules in one command, effectively performing a “Frame Update” of the entire display- or you can update a single module by specifying its address while the number of modules = ‘01’.

Global Write command syntax:
Command: “Global Write” **01,NN,a,AA,<cr>**
Returns: **-NA-**

‘NN’ is the number of modules to write, (typically this is the number of modules in the array or ‘01’ if you are writing to one module)

‘AA’ is the address, normally set to ‘00’ unless you are writing to a single module, in which case set this to the address of the specific module you want to control.

Following the <cr> you must send a continuous string of **raw 8 bit binary brightness values** in a continuous sequence. Failure to send the whole sequence will cause the panel to ignore any commands sent- they will be interpreted as brightness values until the entire panel is fully updated.

For example, using three modules in 36 pixel mode, the command to update all of them at once would look like this: **0103a00<cr> <324 bytes of pixel data>** (each module = 36 pixels, 3 bytes per pixel = 108, 108 * 3 modules = 324 total bytes.

The array will update the LED brightness after the final byte of data is received, completing a ‘frame update.’ This prevents any ‘tearing’ of rapidly moving images.

Dimmer Control Commands

- GLOBAL WRITE
  01DDaAA<cr>
  Where ‘**DD**’= Number of modules to write, ’**AA**’ specifies the starting module address to write.
  Writes in the order of their physical address. Must be followed by the correct number of data bytes.
  You can write sub-sections of the array by specifying an address >1 and the number of modules to write.

- Set DISPLAY MODE
  0CDDa00<cr>
  Where ‘**DD**’= 0x00 = 36 Pixel Mode, 0x01 = SINGLE Pixel Mode, 0x02 = QUAD Pixel Mode
  This command sets the visual display resolution mode of a the module.
  Must be global address **‘00’** for all modules.

Autonomous Behavior Modes

Sensacell Modules can be used in stand alone modes where the behavior of the LED lighting is controlled directly by the computer embedded with each module, without any external controller.

These modes are selected using the BEHAVIOR MODE command.

BEHAVIOR MODE

Command: "Behavior"	**DD,b,AA,<cr>**
Returns:	**-NA-**

Where 'DD' represents the number of the autonomous mode to be enabled- '00'- 'FE'
Setting the behavior mode to 0xFF disables the autonomous modes.

<u>Autonomous Modes</u>

00 Simple On/Off mode (default)
LED's follow sensor state, all 9 LED's corresponding to the sensor illuminate when sensor is triggered. (White light output)

01 Glow mode
LED's follow sensor state, all 9 LED's corresponding to the sensor dim up slowly when sensor is trigged. (White light output)

02 Cross-Fade mode
LED's fade slowly from blue to red on sensor trigger.

03 Color Oscillator mode
Creates an oscillating blue color field- emulating water.

04 Color Oscillator 2 mode
Same as above, pink colors, slower, like sunset clouds.

05 Paint mode
Generates a new random color on every sensor rising edge.

06 Sparkle mode
Creates random candy colored sparkles against a soft blue background.

07 Sparkle 2 mode
Same as above, sparkles have persistent afterglow.

08 Color Speed mode
Orange and green patterns oscillate at a speed proportional to sensor activity.

09 RGB Color Space mode
Creates slow changing false color map proportional to sensor activity.

Consult Sensacell for customized autonomous modes.
Sensacell reserves the right to add / modify modes at any time.