**Steamworks StormNet I2C Bus Commands**

This paper documents the command protocols for the various sensors on the I2C bus.

New commands should be added below so that the software and electronic teams can both see them.

A “no activity” condition on the I2C bus (the timeout is currently 15 seconds) causes the arduino status LED to flash quickly (100ms on / 100 ms off). Any activity causes this to slow down (1000 ms on / off). This behavior supersedes the flashing effects of S and F below.

|  |  |  |
| --- | --- | --- |
| Device | I2C Channel | Notes |
| IR | 11 | NUM\_LINE\_PINS = 5  NUMSENSORS = 6 (not used) |
| Ultrasonic | 8 | NUMSENSORS = 5 |
| Lights | 5 |  |

In the command string column () indicates no other arguments, (int) indicates an integer, etc.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Command String** | **Device** | **Expected bytes** | **Response** | **Other effect** |
| P(byte) | all | 1 | I2C Address | none |
| S() | all | 4 | “SLOW” | Flash LED slowly (1500ms) |
| F() | all | 4 | “FAST” | Flash LED quickly (250ms) |
| B(long) | all | 4 | Current blink rate | Sets blink rate |
| \0 (or default) | all | 2 (1 short) | Basic counter value | Increment counter |
| U | Ultrasonic | NUMSENSORS 5 (5 bytes) | 5 sensor readings | none |
| “1”, “2”, “3”, “4” | Ultrasonic | 2 (1 short) | Basic counter | Turn on / off shooter, gear light |
| I (not live) | IR | 12 (6 shorts) | 6 sensor readings | none |
| G | IR | 2 (1 short) | Gear State | none |
| D | IR | 1 + NUM\_LINE\_PINS  (all single byte booleans) | [Beam broken Proximity detected for each line sensor] | none |
| L(id, mode, arg1, arg2) [all bytes] | Lights | default | Basic counter value | Set light modes |

* There seems to be a bug returning 0 bytes, so all commands should return something until this is fixed.

Gear state details:

enum gear\_states {

EMPTY\_BIN = 0, // Bin is empty

GEAR\_LIFTING, // Gear is leaving, no longer present but has not broken through the beam sensor

FULL\_BIN, // Gear is in the bin and is sensed by the line sensor

GEAR\_EXITING, // Gear is breaking beam on the way out

UNKNOWN\_STATE // Undefined states (Need more sensor data)

};

**Robot offset** based on gear position as detected by line sensor:

1-?-0-?-1 → 0” (i.e. center robot on lift peg)

0-0-1-1-1 → 1.75” left (offset robot to left of lift peg)

0-1-1-1-0 → 1.25” left or right (offset robot to left or right of lift peg)

1-1-1-0-0 → 1.75” right (offset robot to right of lift peg)

**Light Arduino Commands:**

Format of command: Send 4 integers (light ID, mode #, param1, param2)

|  |  |  |  |
| --- | --- | --- | --- |
| Mode # | Mode Name | Parameter 1 | Parameter 2 |
| 1 | Ring Light (Solid color) | COLOR  0 = off  1 = white  2 = green  3 = red | BRIGHTNESS  Send int from 0-255  0 = no brightness  255 = max brightness |

Light IDs:

|  |  |  |
| --- | --- | --- |
| Light ID | Strip Description | Light Range |
| 0 | 5 on-board Neopixels | 0-4 |
| 1 | Gear ring light | 5-44 |

Lights (part of the same strip?)

Shooter

Gear

Undercarriage

Fenders

Circuitboard

(more)

Different modes:

Chase

Seizure

Other samples like Rainbow, etc.

Other arguments:

speed,

Color

other?