LOLED
VIRTUAL

INDIEMARK/GLASSMARK II ENCODER USB GUIDE

The Indiemark lens encoder uses a new USB protocol compared to the Glassmark I. This document is intended as an addendum to the Glassmark Developer Guide.

General Info

The Indiemark Lens Encoder communicates through USB HID using an interrupt interval of 1ms. In practice, it also takes about 1ms to transfer the data, so the practical refresh rate is 500Hz.

Using USB HID rather then Serial allows the host system to deal with the data transfer rather than the application, ensuring the lowest possible latency and fastest possible speeds. HID is a more complicated protocol, so we've provided some sample code to serve as a reference.

Additionally, the encoder can be written to at an interval of 10ms. All data that the encoder provides can be changed, including the current position, LED blackout, LED blinking, and zero.

Sync

Timecode sync is an important part of many workflows. The encoder itself has no knowledge of sync, it simply reports it's position as fast as possible to the host. If sync is required, that operation will need to take place in the users application—the HID interface will ensure you always have the most up-to-date encoder position, so you simply need to tie that position to an external timecode source.

1. USB HID Protocol

An Indiemark USB packet consists of four little endian bytes.

[Encoder Assignment][Encoder Status][Packed encoder position 1][Packed encoder position 2]

When writing to the encoder, any bytes set to 0xFF will be ignored. If not writing the encoder position, both encoder bytes must be 0xFF.

The Indiemark encoder will have a VID of 0x4B4, and a PID of 0x8051. Each encoder will report a unique serial number.

1. Encoder Assignment

This is two 4-bit values packed into a byte, with the most significant byte representing the assigned camera (0-15) and the least significant byte representing the encoder slot (0-15.)

[CCCCSSSS]

A value of 0 means the parameter is unassigned.

For the camera, a value of 1 equates to A camera, 2 to B camera, etc.

We only use cameras 1-4 and slots 1-3 in LONET. If you bypass LONET, you can use up to 14.

EX:

- 0x11 is an assignment of camera A, slot 1
- 0x23 is camera B, slot 3

2. Encoder Status

This is an 8-bit bitmask that displays the current status of the encoder, or allows those elements to be controlled.

[01234567]

- **Bit 0** is 1 if the LED is flashing
- **Bit 1** is 1 if the encoder is zero’d
- **Bit 3** is 1 if the encoder is in blackout mode

If writing this byte, all bits will be accounted for—in order to change only one of these parameters, you must first cache the current encoder status and only flip the bit you’re trying to change.

The exception to this is bit 1, as the encoder can’t be “de-zero’d.” Set this to 0 unless you want to re-zero the encoder.

3. Encoder value

The last two bytes are a packed 16-bit number representing the current encoder position. This value is always positive.