

Insteon - Task #862

Parse Insteon send messages

10/17/2014 02:35 AM - Luke Murphey

Status:	Closed	Start date:	10/16/2014
Priority:	Normal	Due date:	
Assignee:	Luke Murphey	% Done:	100%
Category:		Estimated time:	0.00 hour
Target version:	0.5		
Description			
See https://github.com/hollie/misterhouse/blob/stable/lib/Insteon/MessageDecoder.pm			

History

#1 - 10/17/2014 02:36 AM - Luke Murphey

I have two questions outstanding at this point:

- 1) How do I determine the length of a insteon send message?
- 2) How does pyinsteon currently ignore messages without confusing the length?

#2 - 10/17/2014 02:44 AM - Luke Murphey

The socket is opened with MSG_DONTWAIT. Wondering if read() with a buffer size greater than the data-size causes it to block or just grabs the packet available.

#3 - 10/17/2014 04:36 AM - Luke Murphey

__process_StandardInsteonMessagePLMEcho() accepts the send messages. Some open questions:

- Are 7 bytes enough for the entire body? Packet captures show more.
- How is 0x0062 packet being considered in the 0x0050 function?

#4 - 10/17/2014 04:43 AM - Luke Murphey

Here is the message:

```
02 62 30.46.F6 05 19 01 06
02 50 30.46.F6 2C B8 4E 25 00 01
```

I see now that a single packet includes both the sent message and the response.

The second call is:

```
02 62 30.46.F6 05 19 02 06
02 50 30.46.F6 2C B8 4E 25 00 00
```

#5 - 10/17/2014 05:02 AM - Luke Murphey

From <https://github.com/hollie/misterhouse/blob/stable/lib/Insteon/MessageDecoder.pm>:

```
} elsif($plm_cmd_id eq '0262'){
    $plm_message .= sprintf("%24s: ", 'To Address').substr($plm_string,4,2).":".substr($plm_string,6,2).":".substr($plm_string,8,2)."\n";
    $plm_message .= sprintf("%24s: ", 'Message Flags').substr($plm_string,10,2)."\n";
    $plm_message .= insteon_message_flags_decode(substr($plm_string,10,2));

    my $flag_ext = hex(substr($plm_string,10,1)&0b0001);

    $plm_message .= sprintf("%24s: ", 'Insteon Message').substr($plm_string,12,($flag_ext ? 32 : 4))."\n";
    $plm_message .= insteon_decode(substr($plm_string,10));
    $plm_ack_pos = $flag_ext ? 44 : 16;
}
```

This code parses the messages as:

- to address: 3 bytes
- message flags: 1 byte
- message: 2 bytes

Note sure what the `insteon_decode()` call is for at position 10.

#6 - 10/17/2014 05:11 AM - Luke Murphey

I think I can ignore the last byte. `insteon_decode()` seems to refer to it as the "ack byte" but ignores it:

```
#Truncate $command_string to remove PLM ACK byte
$command_string = substr($command_string,0, ($extended ? 34 : 8));
```

#7 - 10/17/2014 06:36 AM - Luke Murphey

- Status changed from New to Closed

- % Done changed from 0 to 100