List of GoPro Hero 4 I2C Commands

The following tree (Fig. 1) represents all the I2C commands of GoPro Hero 4 camera. Actual I2C packets that the camera understands are in a binary form and further wrapped by packet size, useless constant header bytes, session numbers, etc. So we don't want to go into gory detail of the binary array but we explain human-readable strings that both MewPro Iliad and MewPro4 softwares also understand.

Command Tree

Click '+' to expand a branch and '-' to shrink.

```
-  YY command
      -  * (don't care): extended YY command
            -  0x01: mode change
                  0x00: GET_CAMERA main mode; argc = 0x0000
                -  0x01: SET_CAMERA main mode; argc = 0x0001; argv[] = { mode }
                        mode
                        0x00: video
                        0x01: photo
                        0x02: multi-shot
                        0x04: playback
                        0x05: setup
                        0x06: audio
                  0x04: GET_CAMERA sub mode; argc = 0x0000
                -  0x05: SET_CAMERA sub mode; argc = 0x0002; argv[] = { mode, submode }
                        mode, submode
                        0x00, 0x00: video
                        0x00, 0x01: timelapse video
                        0x00, 0x02: photo in video
                        0x00, 0x03: looping video
                        0x01, 0x00: single photo
                        0x01, 0x01: continuous photo
                        0x01, 0x02: night photo
                        0x02, 0x00: burst photo
                        0x02, 0x01: timelapse photo
                        0x02, 0x02: nightlapse photo
            -  0x02: video
                  0x00: GET_CAMERA default sub mode; argc = 0x0000
                -  0x01: SET_CAMERA default sub mode; argc = 0x0001; argv[] = { submode }
                        submode
                        0x00: video
                        0x01: timelapse video
                        0x02: photo in video
                        0x03: looping video
                  0x02: GET_CAMERA video mode; argc = 0x0000
                -  0x03: SET_CAMERA video mode; argc = 0x0003; argv[] = { resolution, fps, fov }
                        resolution
                        0x01: 4k
                        0x02: 4k SuperView
                        0x04: 2.7k
                        0x05: 2.7k SuperView
                        0x06: 2.7k 4:3
                        0x07: 1440
```

> > 0x08: 1080 SuperView
> > 0x09: 1080
> > 0x0A: 960
> > 0x0B: 720 SuperView
> > 0x0C: 720
> > 0x0D: WVGA
> > fps
> > 0x00: 240
> > 0x01: 120
> > 0x02: 100
> > 0x03: 90
> > 0x04: 80
> > 0x05: 60
> > 0x06: 50
> > 0x07: 48
> > 0x08: 30
> > 0x09: 25
> > 0x0A: 24
> > 0x0B: 15
> > 0x0C: 12.5
> > fov
> > 0x00: wide
> > 0x01: medium
> > 0x02: narrow
> > 0x04: linear (v5 or later firmware only)
> 0x04: GET_CAMERA video + photo; argc = 0x0000
> - 0x05: SET_CAMERA video + photo; argc = 0x0001; argv[] = { interval }
> > interval
> > 0x01: 1 photo per 5 seconds
> > 0x02: 1 photo per 10 seconds
> > 0x03: 1 photo per 30 seconds
> > 0x04: 1 photo per 60 seconds
> 0x06: GET_CAMERA looping; argc = 0x0000
> - 0x07: SET_CAMERA looping; argc = 0x0001; argv[] = { interval }
> > interval
> > 0x00: maximum
> > 0x01: 5 minutes
> > 0x02: 20 minutes
> > 0x03: 60 minutes
> > 0x04: 120 minutes
> 0x08: GET_CAMERA low light; argc = 0x0000
> - 0x09: SET_CAMERA low light; argc = 0x0001; argv[] = { state }
> > state
> > 0x00: off
> > 0x01: on
> 0x0A: GET_CAMERA spot meter; argc = 0x0000
> - 0x0B: SET_CAMERA spot meter; argc = 0x0001; argv[] = { state }
> > state
> > 0x00: off
> > 0x01: on
> 0x0C: GET_CAMERA timelapse rate; argc = 0x0000
> - 0x0D: SET_CAMERA timelapse rate; argc = 0x0001; argv[] = { interval }
> > interval
> > 0x00: 0.5 seconds
> > 0x01: 1 second
> > 0x02: 2 seconds
> > 0x03: 5 seconds
> > 0x04: 10 seconds
> > 0x05: 30 seconds

        0x06: 60 seconds
     0x0E: GET_CAMERA protune; argc = 0x0000
- 0x0F: SET_CAMERA protune; argc = 0x0001; argv[] = { state }
        state
        0x00: off
        0x01: on
     0x10: GET_CAMERA protune white balance; argc = 0x0000
- 0x11: SET_CAMERA protune white balance; argc = 0x0001; argv[] = { wb }
        wb
        0x00: auto
        0x01: 3000K
        0x02: 5500K
        0x03: 6500K
        0x04: native
        0x05: 4000K (v4 or later firmware only)
        0x06: 4800K (v4 or later firmware only)
        0x07: 6000K (v4 or later firmware only)
     0x12: GET_CAMERA protune color; argc = 0x0000
- 0x13: SET_CAMERA protune color; argc = 0x0001; argv[] = { color }
        color
        0x00: GoPro color
        0x01: flat
     0x14: GET_CAMERA protune sharpness; argc = 0x0000
- 0x15: SET_CAMERA protune sharpness; argc = 0x0001; argv[] = {
sharpness }
        sharpness
        0x00: high
        0x01: medium
        0x02: low
     0x16: GET_CAMERA protune ISO; argc = 0x0000
- 0x17: SET_CAMERA protune ISO; argc = 0x0001; argv[] = { limit }
        limit
        0x00: 6400
        0x01: 1600
        0x02: 400
        0x03: 3200
        0x04: 800
        0x07: 200 (v4 or later firmware only)
        0x08: 100 (v4 or later firmware only)
     0x18: GET_CAMERA protune exposure value; argc = 0x0000
- 0x19: SET_CAMERA protune exposure value; argc = 0x0001; argv[] = { ev }
        ev
        0x00: +2.0
        0x01: +1.5
        0x02: +1.0
        0x03: +0.5
        0x04: 0
        0x05: -0.5
        0x06: -1.0
        0x07: -1.5
        0x08: -2.0
     0x1A: SET_CAMERA reset protune; argc = 0x0000
     0x1B: SET_CAMERA shutter button depressed. start recording; argc =
0x0000
     0x1C: SET_CAMERA stop recording; argc = 0x0000
- 0x26: SET_CAMERA bulk set video settings; argc = 0x0060; argv[]
        video
        argv[0]: current submode (defunct)
        argv[1]: power on submode

argv[2]: resolution
argv[3]: fps
argv[4]: fov
argv[5]: video + photo interval
argv[6]: looping
argv[7]: spot meter
argv[8]: low light
argv[9]: timelapse interval
argv[10]: protune
argv[11]: color
argv[12]: sharpness
argv[13]: ISO limit
argv[14]: exposure compensation
argv[15]: white balance
argv[16]: shutter
argv[17]: ISO mode
argv[18:23]: (no use)
timelapse video
argv[24]: current submode (defunct)
argv[25]: power on submode
argv[26]: resolution
argv[27]: fps
argv[28]: fov
argv[29]: video + photo interval
argv[30]: looping
argv[31]: spot meter
argv[32]: low light
argv[33]: timelapse interval
argv[34]: protune
argv[35]: color
argv[36]: sharpness
argv[37]: ISO limit
argv[38]: exposure compensation
argv[39]: white balance
argv[40]: shutter
argv[41]: ISO mode
argv[42:47]: (no use)
video + photo
argv[48]: current submode (defunct)
argv[49]: power on submode
argv[50]: resolution
argv[51]: fps
argv[52]: fov
argv[53]: video + photo interval
argv[54]: looping
argv[55]: spot meter
argv[56]: low light
argv[57]: timelapse interval
argv[58]: protune
argv[59]: color
argv[60]: sharpness
argv[61]: ISO limit
argv[62]: exposure compensation
argv[63]: white balance
argv[64]: shutter
argv[65]: ISO mode
argv[66:71]: (no use)
looping
argv[72]: current submode (defunct)

argv[73]: power on submode
argv[74]: resolution
argv[75]: fps
argv[76]: fov
argv[77]: video + photo interval
argv[78]: looping
argv[79]: spot meter
argv[80]: low light
argv[81]: timelapse interval
argv[82]: protune
argv[83]: color
argv[84]: sharpness
argv[85]: ISO limit
argv[86]: exposure compensation
argv[87]: white balance
argv[88]: shutter
argv[89]: ISO mode
argv[90:95]: (no use)
0x27: GET_CAMERA exposure time (v4 or later firmware only); argc = 0x0000
- 0x28: SET_CAMERA exposure time (v4 or later firmware only); argc = 0x0001; argv[] = { shutter }
     shutter
     0x00: auto
     0x01: 1/12.5
     0x02: 1/15
     0x03: 1/24
     0x04: 1/25
     0x05: 1/30
     0x06: 1/48
     0x07: 1/50
     0x08: 1/60
     0x09: 1/80
     0x0A: 1/90
     0x0B: 1/96
     0x0C: 1/100
     0x0D: 1/120
     0x0E: 1/160
     0x0F: 1/180
     0x10: 1/192
     0x11: 1/200
     0x12: 1/240
     0x13: 1/320
     0x14: 1/360
     0x15: 1/400
     0x16: 1/480
     0x17: 1/960
0x29: GET_CAMERA protune ISO mode (v4 or later firmware only); argc = 0x0000
- 0x2A: SET_CAMERA protune ISO mode (v4 or later firmware only); argc = 0x0001; argv[] = { mode }
     mode
     0x00: maximum
     0x01: lock
- 0x03: photo
     0x00: GET_CAMERA default sub mode; argc = 0x0000
     - 0x01: SET_CAMERA default sub mode; argc = 0x0001; argv[] = { submode }
          submode
          0x00: single

             0x01: continuous
             0x02: night
        0x02: GET_CAMERA resolution; argc = 0x0000
- 0x03: SET_CAMERA resolution; argc = 0x0001; argv[] = { resolution }
             resolution
             0x00: 12MP wide
             0x01: 7MP wide
             0x02: 7MP medium
             0x03: 5MP medium
        0x04: GET_CAMERA continuous rate; argc = 0x0000
- 0x05: SET_CAMERA continuous rate; argc = 0x0001; argv[] = { interval }
             interval
             0x00: 3 frames per second
             0x01: 5 frames per second
             0x02: 10 frames per second
        0x06: GET_CAMERA spot meter; argc = 0x0000
- 0x07: SET_CAMERA spot meter; argc = 0x0001; argv[] = { state }
             state
             0x00: off
             0x01: on
        0x08: GET_CAMERA exposure time; argc = 0x0000
- 0x09: SET_CAMERA exposure time; argc = 0x0001; argv[] = { shutter }
             shutter
             0x00: auto
             0x01: 2 seconds
             0x02: 5 seconds
             0x03: 10 seconds
             0x04: 15 seconds
             0x05: 20 seconds
             0x06: 30 seconds
        0x0A: GET_CAMERA protune; argc = 0x0000
- 0x0B: SET_CAMERA protune; argc = 0x0001; argv[] = { state }
             state
             0x00: off
             0x01: on
        0x0C: GET_CAMERA protune white balance; argc = 0x0000
- 0x0D: SET_CAMERA protune white balance; argc = 0x0001; argv[] = { wb }
             wb
             0x00: auto
             0x01: 3000K
             0x02: 5500K
             0x03: 6500K
             0x04: native
             0x05: 4000K (v4 or later firmware only)
             0x06: 4800K (v4 or later firmware only)
             0x07: 6000K (v4 or later firmware only)
        0x0E: GET_CAMERA protune color; argc = 0x0000
- 0x0F: SET_CAMERA protune color; argc = 0x0001; argv[] = { color }
             color
             0x00: GoPro color
             0x01: flat
        0x10: GET_CAMERA protune sharpness; argc = 0x0000
- 0x11: SET_CAMERA protune sharpness; argc = 0x0001; argv[] = {
sharpness }
             sharpness
             0x00: high
             0x01: medium
             0x02: low
        0x12: GET_CAMERA protune ISO maximum; argc = 0x0000

- 0x13: SET_CAMERA protune ISO maximum; argc = 0x0001; argv[] = { ISO }
    ISO
    0x00: 800
    0x01: 400
    0x02: 200
    0x03: 100

0x14: GET_CAMERA protune exposure value; argc = 0x0000
- 0x15: SET_CAMERA protune exposure value; argc = 0x0001; argv[] = { ev }
    ev
    0x00: +2.0
    0x01: +1.5
    0x02: +1.0
    0x03: +0.5
    0x04: 0
    0x05: -0.5
    0x06: -1.0
    0x07: -1.5
    0x08: -2.0

0x16: SET_CAMERA reset protune; argc = 0x0000
0x17: SET_CAMERA shutter button depressed. start recording; argc = 0x0000
0x18: SET_CAMERA stop recording (continuous submode only); argc = 0x0000
- 0x1B: SET_CAMERA bulk set photo settings; argc = 0x003C; argv[]
    single
    argv[0]: current submode (defunct)
    argv[1]: power on submode
    argv[2]: resolution
    argv[3]: interval
    argv[4]: spot meter
    argv[5]: shutter
    argv[6]: protune
    argv[7]: color
    argv[8]: sharpness
    argv[9]: ISO maximum
    argv[10]: exposure compensation
    argv[11]: white balance
    argv[12]: ISO minimum
    argv[13:19]: (no use)
    continuous
    argv[20]: current submode (defunct)
    argv[21]: power on submode
    argv[22]: resolution
    argv[23]: interval
    argv[24]: spot meter
    argv[25]: shutter
    argv[26]: protune
    argv[27]: color
    argv[28]: sharpness
    argv[29]: ISO maximum
    argv[30]: exposure compensation
    argv[31]: white balance
    argv[32]: ISO minimum
    argv[33:39]: (no use)
    night
    argv[40]: current submode (defunct)
    argv[41]: power on submode
    argv[42]: resolution
    argv[43]: interval

               argv[44]: spot meter
               argv[45]: shutter
               argv[46]: protune
               argv[47]: color
               argv[48]: sharpness
               argv[49]: ISO maximum
               argv[50]: exposure compensation
               argv[51]: white balance
               argv[52]: ISO minimum
               argv[53:59]: (no use)
          0x1C: GET_CAMERA protune ISO minimum (v4 or later firmware only); argc = 0x0000
        - 0x1D: SET_CAMERA protune ISO minimum (v4 or later firmware only); argc = 0x0001; argv[] = { ISO }
              ISO
              0x00: 800
              0x01: 400
              0x02: 200
              0x03: 100
    - 0x04: multi-shot
          0x00: GET_CAMERA default sub mode; argc = 0x0000
        - 0x01: SET_CAMERA default sub mode; argv[] = { submode }
              submode
              0x00: burst
              0x01: timelapse
              0x02: nightlapse
          0x02: GET_CAMERA resolution; argc = 0x0000
        - 0x03: SET_CAMERA resolution; argc = 0x0001; argv[] = { resolution }
              resolution
              0x00: 12MP wide
              0x01: 7MP wide
               0x02: 7MP medium
               0x03: 5MP medium
          0x04: GET_CAMERA burst rate; argc = 0x0000
        - 0x05: SET_CAMERA burst rate; argc = 0x0001; argv[] = { interval }
              interval
              0x00: 3 photos / 1 second
              0x01: 5 photos / 1 second
              0x02: 10 photos / 1 second
              0x03: 10 photos / 2 seconds
              0x04: 10 photos / 3 seconds
              0x05: 30 photos / 1 second
              0x06: 30 photos / 2 seconds
              0x07: 30 photos / 3 seconds
              0x08: 30 photos / 6 seconds
          0x06: GET_CAMERA timelapse rate; argc = 0x0000
        - 0x07: SET_CAMERA timelapse rate; argc = 0x0001; argv[] = { interval }
              interval
              0x00: 1 photo / 0.5 seconds
              0x01: 1 photo / 1 second
              0x02: 1 photo / 2 seconds
              0x03: 1 photo / 5 seconds
              0x04: 1 photo / 10 seconds
              0x05: 1 photo / 30 seconds
              0x06: 1 photo / 60 seconds
          0x08: GET_CAMERA nightlapse rate; argc = 0x0000
        - 0x09: SET_CAMERA nightlapse rate; argc = 0x0001; argv[] = { interval }
              interval
              0x00: 10 seconds

            0x01: 15 seconds
            0x02: 20 seconds
            0x03: 30 seconds
            0x04: 1 minute
            0x05: 2 minutes
            0x06: 5 minutes
            0x07: 30 minutes
            0x08: 60 minutes
        0x0A: GET_CAMERA spot meter; argc = 0x0000
- 0x0B: SET_CAMERA spot meter; argc = 0x0001; argv[] = { status }
            state
            0x00: off
            0x01: on
        0x0C: GET_CAMERA exposure time; argc = 0x0000
- 0x0D: SET_CAMERA exposure time; argc = 0x0001; argv[] = { shutter }
            shutter
            0x00: auto
            0x01: 2 seconds
            0x02: 5 seconds
            0x03: 10 seconds
            0x04: 15 seconds
            0x05: 20 seconds
            0x06: 30 seconds
        0x0E: GET_CAMERA protune; argc = 0x0000
- 0x0F: SET_CAMERA protune; argc = 0x0001; argv[] = { status }
            state
            0x00: off
            0x01: on
        0x10: GET_CAMERA protune white balance; argc = 0x0000
- 0x11: SET_CAMERA protune white balance; argc = 0x0001; argv[] = { wb }
            wb
            0x00: auto
            0x01: 3000K
            0x02: 5500K
            0x03: 6500K
            0x04: native
            0x05: 4000K (v4 or later firmware only)
            0x06: 4800K (v4 or later firmware only)
            0x07: 6000K (v4 or later firmware only)
        0x12: GET_CAMERA protune color; argc = 0x0000
- 0x13: SET_CAMERA protune color; argc = 0x0001; argv[] = { color }
            color
            0x00: GoPro color
            0x01: flat
        0x14: GET_CAMERA protune sharpness; argc = 0x0000
- 0x15: SET_CAMERA protune sharpness; argc = 0x0001; argv[] = { sharpness }
            sharpness
            0x00: high
            0x01: medium
            0x02: low
        0x16: GET_CAMERA protune ISO; argc = 0x0000
- 0x17: SET_CAMERA protune ISO; argc = 0x0001; argv[] = { ISO }
            ISO
            0x00: 800
            0x01: 400
            0x02: 200
            0x03: 100
        0x18: GET_CAMERA protune exposure value; argc = 0x0000

- 0x19: SET_CAMERA protune exposure value; argc = 0x0001; argv[] = { ev }
  - ev
  - 0x00: +2.0
  - 0x01: +1.5
  - 0x02: +1.0
  - 0x03: +0.5
  - 0x04: 0
  - 0x05: -0.5
  - 0x06: -1.0
  - 0x07: -1.5
  - 0x08: -2.0
- 0x1A: SET_CAMERA reset protune; argc = 0x0000
- 0x1B: SET_CAMERA shutter button depressed. start recording; argc = 0x0000
- 0x1C: SET_CAMERA stop recording; argc = 0x0000
- 0x20: SET_CAMERA bulk set photo settings; argc = 0x0042; argv[]
  - burst
  - argv[0]: current submode (defunct)
  - argv[1]: power on submode
  - argv[2]: resolution
  - argv[3]: rate
  - argv[4]: timelapse interval
  - argv[5]: nightlapse interval
  - argv[6]: spot meter
  - argv[7]: shutter
  - argv[8]: protune
  - argv[9]: color
  - argv[10]: sharpness
  - argv[11]: ISO maximum
  - argv[12]: exposure compensation
  - argv[13]: white balance
  - argv[14]: ISO minimum
  - argv[15:21]: (no use)
  - timelapse
  - argv[22]: current submode (defunct)
  - argv[23]: power on submode
  - argv[24]: resolution
  - argv[25]: rate
  - argv[26]: timelapse interval
  - argv[27]: nightlapse interval
  - argv[28]: spot meter
  - argv[29]: shutter
  - argv[30]: protune
  - argv[31]: color
  - argv[32]: sharpness
  - argv[33]: ISO maximum
  - argv[34]: exposure compensation
  - argv[35]: white balance
  - argv[36]: ISO minimum
  - argv[37:43]: (no use)
  - nightlapse
  - argv[44]: current submode (defunct)
  - argv[45]: power on submode
  - argv[46]: resolution
  - argv[47]: rate
  - argv[48]: timelapse interval
  - argv[49]: nightlapse interval
  - argv[50]: spot meter
  - argv[51]: shutter

                    argv[52]: protune
                    argv[53]: color
                    argv[54]: sharpness
                    argv[55]: ISO maximum
                    argv[56]: exposure compensation
                    argv[57]: white balance
                    argv[58]: ISO minimum
                    argv[59:65]: (no use)
              0x21: GET_CAMERA protune ISO minimum (v4 or later firmware only); argc = 0x0000
            - 0x22: SET_CAMERA protune ISO minimum (v4 or later firmware only); argc = 0x0001; argv[] = { ISO }
                    ISO
                    0x00: 800
                    0x01: 400
                    0x02: 200
                    0x03: 100
       - 0x07: global settings
              0x00: GET_CAMERA LCD brightness; argc = 0x0000
            - 0x01: SET_CAMERA LCD brightness; argc = 0x0001; argv[] = { brightness }
                    brightness
                    0x00: high
                    0x01: medium
                    0x02: low
               0x02: GET_CAMERA LCD sleep; argc = 0x0000
            - 0x03: SET_CAMERA LCD sleep; argc = 0x0001; argv[] = { sleep }
                    sleep
                    0x00: never
                    0x01: 1 minute
                    0x02: 2 minutes
                    0x03: 3 minutes
              0x04: GET_CAMERA LCD lock; argc = 0x0000
            - 0x05: SET_CAMERA LCD lock; argc = 0x0001; argv[] = { state}
                    state
                    0x00: off
                    0x01: on
              0x06: GET_CAMERA LCD power; argc = 0x0000
            - 0x07: SET_CAMERA LCD power; argc = 0x0001; argv[] = { state }
                    state
                    0x00: off
                    0x01: on
              0x08: GET_CAMERA orientation; argc = 0x0000
            - 0x09: SET_CAMERA orientation; argc = 0x0001; argv[] = { orientation }
                    orientation
                    0x00: auto
                    0x01: up
                    0x02: down
              0x0A: GET_CAMERA default mode; argc = 0x0000
            - 0x0B: SET_CAMERA default mode; argc = 0x0001; argv[] = { mode }
                    mode
                    0x00: video
                    0x01: photo
                    0x02: multi-shot
              0x0C: GET_CAMERA quick capture; argc = 0x0000
            - 0x0D: SET_CAMERA quick capture; argc = 0x0001; argv[] = { state }
                    state
                    0x00: off
                    0x01: on
              0x0E: GET_CAMERA LEDs; argc = 0x0000

- 0x0F: SET_CAMERA LEDs; argc = 0x0001; argv[] = { LEDs }
    LEDs
    0x00: off
    0x01: 2
    0x02: 4
- 0x10: GET_CAMERA beeps; argc = 0x0000
- 0x11: SET_CAMERA beeps; argc = 0x0001; argv[] = { beeps }
    beeps
    0x00: 100%
    0x01: 70%
    0x02: off
- 0x12: GET_CAMERA video format; argc = 0x0000
- 0x13: SET_CAMERA video format; argc = 0x0001; argv[] = { format }
    format
    0x00: NTSC
    0x01: PAL
- 0x14: GET_CAMERA OSD; argc = 0x0000
- 0x15: SET_CAMERA OSD; argc = 0x0001; argv[] = { state }
    state
    0x00: off
    0x01: on
- 0x16: GET_CAMERA auto power down; argc = 0x0000
- 0x17: SET_CAMERA auto power down; argc = 0x0001; argv[] = { minute }
    minute
    0x00: never
    0x01: 1 minute
    0x02: 2 minutes
    0x03: 3 minutes
    0x04: 5 minutes
- 0x1A: GET_CAMERA datetime; argc = 0x0000
- 0x1B: SET_CAMERA datetime; argc = 0x0001; argv[] = { year_h, year_l, month, day, hour, minute, second }
    year_h: high byte of year
    year_l: low byte of year
    month: month (1 - 12)
    day: day (1 - 31)
    hour: hour (0 - 23)
    minute: minute (0 - 60)
    second: second (0 - 60)
- 0x1F: GET_CAMERA language; argc = 0x0000
- 0x20: SET_CAMERA language; argc = 0x0001; argv[] = { language }
    language
    0x00: English
    0x01: Simplified Chinese
- 0x21: SET_CAMERA bulk set global settings; argc = 0x0023; argv[]
    argv[0]: 0x20 (unknown constant)
    argv[1]: 0x16 (unknown constant)
    argv[2]: 0x01 (unknown constant)
    argv[3]: 0x11 (unknown constant)
    argv[4]: 0x00 (unknown constant)
    argv[5]: 0x00 (unknown constant)
    argv[6]: OSD
    argv[7]: beeps
    argv[8]: auto power off
    argv[9]: LEDs
    argv[10]: quick capture
    argv[11]: orientation
    argv[12]: LCD brightness
    argv[13]: LCD sleep

                                    argv[14]: LCD lock
                                    argv[15]: LCD power
                                    argv[16]: video format
                                    argv[17]: language
                                    argv[18:24]: (no use)
                                    argv[25]: 0x01 (unknown constant)
                                    argv[26]: default mode
                                    argv[27]: default submode
                                    argv[28]: high byte of year
                                    argv[29]: low byte of year
                                    argv[30]: month
                                    argv[31]: day
                                    argv[32]: hour
                                    argv[33]: minute
                                    argv[34]: second

- 0x09: delete
  - 0x09: SET_CAMERA delete last; argc = 0x0000
  - 0x0A: SET_CAMERA delete all/format; argc = 0x0000
- ZZ command
  - 0x01:
    - 0x00: GET_BACPAC (unknown)
  - 0x02:
    - 0x01: SET_BACPAC power on
  - 0x03:
    - 0x00: GET_CAMERA version string
  - 0x00: extended ZZ command
    - 0x00: SET_CAMERA sync source; argc = 0x01; argv[] = { source, 0x00 }
      - source
      - 0x00: external (multi camera)
      - 0x01: internal (standalone)
    - 0x01: SET_BACPAC camera sync status
      - 0x01; argv[] = { status, 0x00 }
        - status
        - 0x00: camera is recording
        - 0x01: camera is finished
      - 0x02; camera is blocking until microSD write complete; argv[] = { 0x00, 0x00 }
      - 0x07: camera is requesting external sync signal; argv[] = { 0x01, fps_h, fps_l, hsync_h, hsync_l, dummy_h, dummy_l }
        - fps_h: high byte of fps * 100
        - fps_l: low byte of fps * 100
        - hsync_h: high byte of number of HSYNC pulses in a frame (buggy)
        - hsync_l: low byte of number of HSYNC pulses in a frame (buggy)
        - dummy_h: (unknown. always equal to hsync_h)
        - dummy_l: (unknown. always equal to hsync_l)
    - 0x02: SET_BACPAC heartbeat; argc = 0x00
    - 0x03: SET_CAMERA power off; argc = 0x01; argv[] = { 0x01 }
    - 0x05: SET_CAMERA bacpac firmware version; argc = 0x03; argv[] = { version[3], 0x00 }
    - 0x06: SET_CAMERA bacpac serial number; argc = 0x08; argv[] = { serial_number[7], check_sum, 0x00 }

Fig. 1

Usage

The commands listed above are translated into I2C binary arrays for Hero 4 camera by MewPro Iliad or MewPro4 softwares. Both Iliad and MewPro can receive these strings through their default UART port (aka `Serial` in Arduino's terminology) at 57600 baud. A command consists of hexadecimal

characters preceded by either YY or ZZ and followed by a newline character \n. These commands can be sent from your PC's terminal or Arduino Monitor easily.

Each node of the tree corresponds a command byte that is denoted by using two hexadecimal digits (0-9 and/or A-F).

Example 1: Changing camera mode to multi-shot = "YY000101000102\n".
This command string is obtained by clicking

- "YY command" →
- "* (don't care): extended YY command" →
- "0x01: mode change" →
- "0x01: SET_CAMERA main mode; argc = 0x0001; argv[] = { mode }" where mode=0x02 (multi-shot).

These branch expansions respectively correspond to the following nodes (cf. Fig. 2).

- "YY" →
- "00" (This is a don't-care byte. Let it be 00 for simplicity.) →
- "01" →
- "01" "0001" "02". (The last node may contain arguments to the command: argc followed by argv's.)
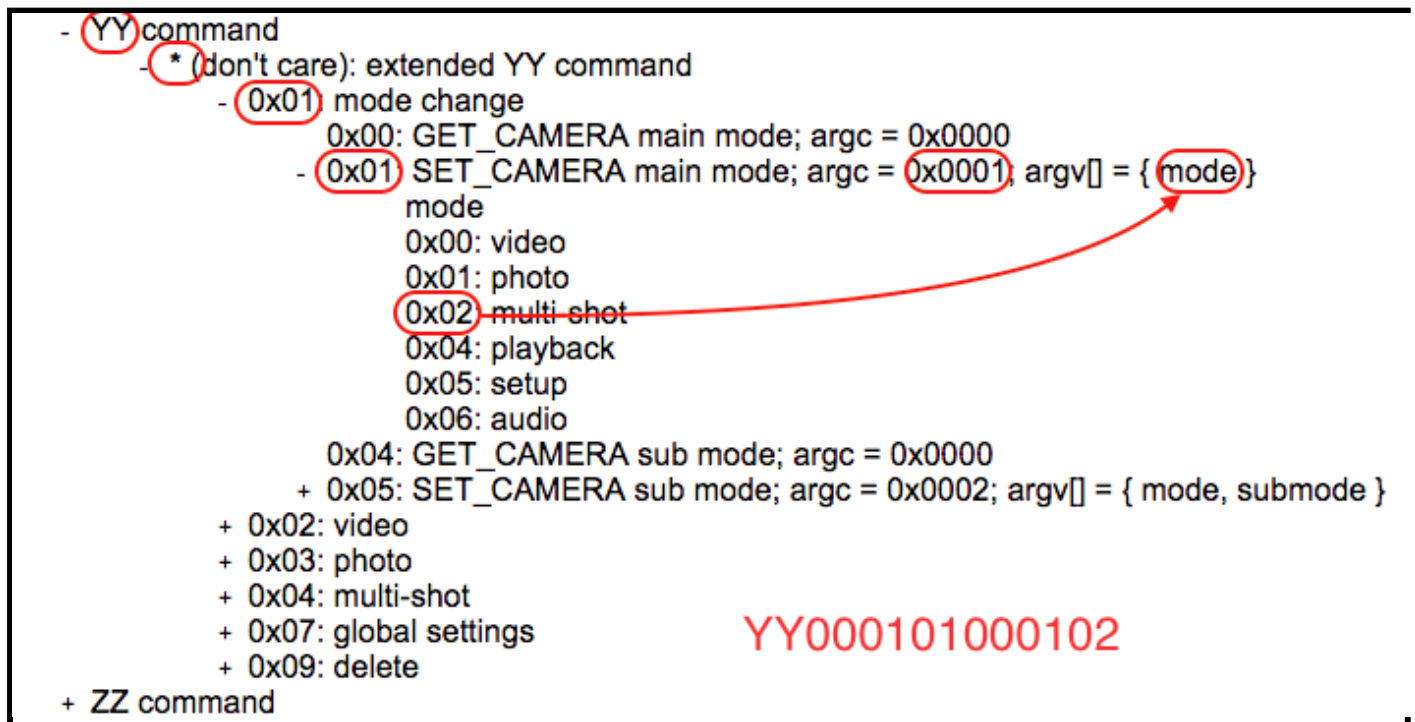

Fig. 2

Example 2: Start recording in video mode = "YY00021B0000\n".
This command string is obtained by clicking

- "YY command" →
- "* (don't care): extended YY command" →
- "0x02: video" →
- "0x1B: SET_CAMERA shutter button depressed. start recording; argc = 0x0000".

These branch expansions respectively correspond to the following nodes

- "YY" →
- "00" (don't care) →
- "02" →
- "1B" "0000". (There are zero arguments to the command but argument length.)

Example 3: Change video mode to 4K 30fps Wide = "YY0002030003010800\n".
This command string is obtained by clicking

- "YY command" →
- "* (don't care): extended YY command" →
- "0x02: video" →
- "0x03: SET_CAMERA video mode; argc = 0x0003; argv[] = { resolution, fps, fov }"

where resolution=0x01, fps=0x08, and fov=0x00.
These branch expansions respectively correspond to the following nodes

- "YY" →
- "00" (don't care) →
- "02" →
- "03" "0003" "01" "08" "00". (There are three arguments to the command.)

Example 4: Power off camera = "ZZ00030101\n".
This command string is obtained by clicking

- "ZZ command" →
- "0x00: extended ZZ command" →
- "0x03: SET_CAMERA power off; argc = 0x01; argv[] = { 0x01 }".

These branch expansions respectively correspond to the following nodes

- "ZZ" →
- "00" →
- "03" "01" "01". (There is one argument to the command.)

Note

In order to improve readability the following strings, for example, share the same meaning for
MewPro Iliad and/or MewPro4 softwares.

- ZZ00030101\n
- ZZ 00 03 01 01\n (inserting arbitrary spaces at 8-bit boundary)
- ZZ0 3 1 1\n (omitting upper nibble 0)

Useful Links

- MewPro Iliad: Shop, Usage, GitHub repo
- MewPro 2: Shop, Usage, GitHub repo