H01R00 RGB LED

Technical Specifications
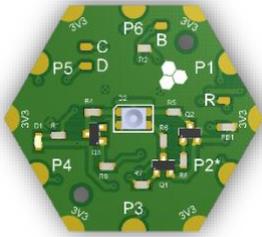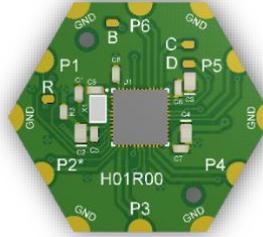
Top (1:1) Bottom (1:1)

- Six array ports and six power ports (+3.3V and GND).
- Access to 6xUART, 2xI^{2}C, SWD, BOOT0, RESET.
- Cree CLVBA-FKA-CC1F1L1BB7R3R3 RGB LED:
  - Dominant Wavelength: Red (619 - 624nm) Green (520 - 540nm) Blue (460 - 480nm).
  - Luminous Intensity (mcd) @IF=20mA: Red (224 - 560) Green (280 - 900) Blue (90 - 355).
- STM32F091CBU6 MCU.
- 8MHz external oscillator.

Available colors:

Commands *use with any serial terminal software*

```
command parameter1 parameter2 …
```

on *intensity*

Turn RGB LED on (white color) at specified *intensity* (`0-100`).

off

Turn RGB LED off.

toggle *intensity*

Toggle RGB LED on (white color) at specified *intensity* (`0-100`).

color *colorName intensity*

Set RGB LED to a *colorName* from a predefined color list (`black, white, red, blue, yellow, cyan, magenta, green`) at specified *intensity* (`0-100`).

RGB *red green blue intensity*

Set RGB LED to a *red* (`0-255`), *green* (`0-255`), *blue* (`0-255`) color at specified *intensity* (`0-100`).

pulseColor *colorName period duty*

Pulse the RGB LED with a *colorName* from a predefined color list at specified *period* (`1-2`^{32})ms, *duty* cycle (`1-2`^{32})ms, and *repeat* times (`1-2`^{31}) (`inf` for a periodic signal).

Commands *continued*

pulseRGB *red green blue period duty repeat*

Pulse the RGB LED with a *red* (`0-255`), *green* (`0-255`), *blue* (`0-255`) color at specified *period* (`1-2`^{32}) ms, *duty* cycle (`1-2`^{32}) ms, and *repeat* times (`1-2`^{31}) (`inf` for a periodic signal).

sweep *mode period repeat*

Color sweep on the RGB LED with the specified sweep *mode* (`basic`: sweep colors in the color list, `fine`: sweep all colors), sweep *period* (`1-2`^{32}) ms and *repeat* times (`1-2`^{31}) (`inf` for a periodic signal).

dim *colorName mode period wait repeat*

Dim the specified *colorName* on the RGB LED using a dim *mode* (`up, upwait, down, downwait, updown, downup, updownwait, downupwait`), dim *period* (`1-2`^{32}) ms, *wait* time (`1-2`^{32}) ms and *repeat* times (`1-2`^{31}) (`inf` for a periodic signal).

Examples

```
RGB 220 165 33 75

Sweep fine 3000 3

dim redupdownwait1000500inf
```

Messages *for inter-array communication*

```
code, parameter1[value], parameter2[value], …
```

```
CODE_H01R0_ON, intensity

CODE_H01R0_OFF

CODE_H01R0_TOGGLE, intensity

CODE_H01R0_COLOR, 0, colorName, intensity

CODE_H01R0_COLOR, 1, red, green, blue, intensity

CODE_H01R0_PULSE, 0, colorName, period 4th
byte (MSB), period 3rd byte, period 2nd
byte, period 1st byte, duty 4th byte (MSB),
duty 3rd byte, duty 2nd byte, duty 1st byte,
repeat 4th byte (MSB), repeat 3rd byte,
repeat 2nd byte, repeat 1st byte
```
Use repeat = -1 or 0xFFFFFFFF for periodic signal.

```
CODE_H01R0_PULSE, 1, red, green, blue, …
```
Same parameters as above. Use repeat = -1 or 0xFFFFFFFF for periodic signal.
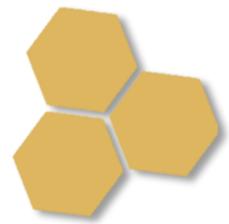
Messages *continued*

```
CODE_H01R0_SWEEP, mode [RGB_SWEEP_BASIC,
RGB_SWEEP_FINE],period 4th byte (MSB), period
3rd byte, period 2nd byte,period 1st byte,
repeat 4th byte (MSB), repeat 3rd byte,
repeat 2nd byte,repeat 1st byte
```
Use repeat = -1 or 0xFFFFFFFF for periodic signal.

```
CODE_H01R0_DIM, colorName, mode [RGB_DIM_UP,
RGB_DIM_UP_WAIT, RGB_DIM_DOWN,
RGB_DIM_DOWN_WAIT, RGB_DIM_UP_DOWN,
RGB_DIM_DOWN_UP, RGB_DIM_UP_DOWN_WAIT,
RGB_DIM_DOWN_UP_WAIT],period 4th byte (MSB),
period 3rd byte, period 2nd byte,period 1st
byte,wait 4th byte (MSB), wait 3rd byte, wait
2nd byte,wait 1st byte, repeat 4th byte
(MSB), repeat 3rd byte, repeat 2nd
byte,repeat 1st byte
```
Use repeat = -1 or 0xFFFFFFFF for periodic signal.

Examples

```
messageParams[0] = 0; messageParams[1] =
yellow; messageParams[2] = 75;

SendMessageToModule(2, CODE_H01R0_COLOR, 3);

SendMessageToModule(2, CODE_H01R0_OFF, 0);

// color
messageParams[0] = CYAN;
// mode
messageParams[1] = RGB_DIM_UP_DOWN_WAIT;
// period
messageParams[2] = (uint8_t)(3000>>24);
messageParams[3] = (uint8_t)(3000>>16);
messageParams[4] = (uint8_t)(3000>>8);
messageParams[5] = (uint8_t)(3000);
// wait
messageParams[6] = (uint8_t)(1000>>24);
messageParams[7] = (uint8_t)(1000>>16);
messageParams[8] = (uint8_t)(1000>>8);
messageParams[9] = (uint8_t)(1000);
// repeat
messageParams[10] = 0;
messageParams[11] = 0;
messageParams[12] = 0;
messageParams[13] = 1;

SendMessageToModule(BOS_BROADCAST,
CODE_H01R0_DIM, 14);
```

APIs *getting your hands dirty!*

*output*API_function(inputs)

```
H01R0_StatusRGB_LED_on(uint8_t intensity)

H01R0_StatusRGB_LED_off()

H01R0_StatusRGB_LED_toggle(uint8_t intensity)

H01R0_StatusRGB_LED_setRGB(uint8_t red,
uint8_t green, uint8_t blue, uint8_t
intensity)

H01R0_StatusRGB_LED_setColor(uint8_t color,
uint8_t intensity)

H01R0_StatusRGB_LED_pulseRGB(uint8_t red,
uint8_t green, uint8_t blue, uint32_t period,
uint32_t dc, int32_t repeat)

H01R0_StatusRGB_LED_pulseColor(uint8_t color,
uint32_t period, uint32_t dc, int32_t repeat)

H01R0_StatusRGB_LED_sweep(uint8_t mode,
uint32_t period, int32_t repeat)

H01R0_StatusRGB_LED_dim(uint8_t color,
uint8_t mode, uint32_t period, uint32_t wait,
int32_t repeat)
```

Examples

```
RGB_LED_pulseRGB(200, 220, 50, 1000, 50, 5);

H01R0_Status stat = H01R0_OK;

stat = RGB_LED_sweep(RGB_SWEEP_BASIC, 10000,
-1);
```