# LETOH HW/SW INTERFACE DESCRIPTION

Kimmo Lindholm 2015-01-20, Version 1.0

## 1 HARDWARE

LeTOH contains two PCA9685 led driver IC's controlled directly by Jolla I2C bus, eeprom and a step-up converter for generating higher voltage for blue leds.

Following table shows PCA9685 connections. Position - phone display facing front.

| Driver address | Output pin | Led color | Position |
|---|---|---|---|
| 0x40 | LED0 | Green | Top right |
| 0x40 | LED1 | Red | Top right |
| 0x40 | LED2 | Blue | Top right |
| 0x40 | LED3 | Green | Right upper |
| 0x40 | LED4 | Red | Right upper |
| 0x40 | LED5 | Blue | Right upper |
| 0x40 | LED6 | Green | Right middle |
| 0x40 | LED7 | Red | Right middle |
| 0x40 | LED8 | Blue | Right middle |
| 0x40 | LED9 | Green | Right lower |
| 0x40 | LED10 | Red | Right lower |
| 0x40 | LED11 | Blue | Right lower |
| 0x40 | LED12 | Green | Bottom right |
| 0x40 | LED13 | Red | Bottom right |
| 0x40 | LED14 | Blue | Bottom right |
| 0x40 | LED15 | Step-up control *) | |
| 0X41 | LED0 | Green | Bottom left |
| 0X41 | LED1 | Red | Bottom left |
| 0X41 | LED2 | Blue | Bottom left |
| 0X41 | LED3 | Green | Left lower |
| 0X41 | LED4 | Red | Left lower |
| 0X41 | LED5 | Blue | Left lower |
| 0X41 | LED6 | Green | Left middle |
| 0X41 | LED7 | Red | Left middle |

| | | | |
|---|---|---|---|
| 0X41 | LED8 | Blue | Left middle |
| 0X41 | LED9 | Green | Left upper |
| 0X41 | LED10 | Red | Left upper |
| 0X41 | LED11 | Blue | Left upper |
| 0X41 | LED12 | Green | Top left |
| 0X41 | LED13 | Red | Top left |
| 0X41 | LED14 | Blue | Top left |
| 0X41 | LED15 | | |

*) Driver 0x40 output LED15 is used to control step-up dcdc converter on/off. This is due that the PCA9685 outputs are driven low when power is applied, which would cause all leds to be full on. This situation consumes too much power, and the phone 3.3V output will hit current limiter.

# 2 INITIALISATION

The basic initialisation to chips needs to be done in correct order, first 0x41 then 0x40 due the above explained step-up converter enable.

1. Write 0x20 to the MODE1 register at address 0x00 to set register auto-increment and normal mode, internal clock. Do not respond to sub-addresses or led-all-call.
2. Wait 1 ms
3. Write 0xA0 to the MODE1 register at address 0x00 to trigger restart.
4. Write 0x10 to the MODE2 register at address 0x01 to invert output states, change outputs on I2C STOP, set outputs as open-drain.

Above step 4 will leave all LEDn outputs in tristate. Pullup resistor will enable step-up converter, and all leds will be OFF.

# 3 DRIVING LEDS

Individual led control registers starts at register address 0x06. Each led has four 8-bit registers with 12 bits PWM value for ON and OFF. These registers also contains bits for turning leds fully ON or OFF.

Following table hopefully explains these a bit. They show registers for LED0

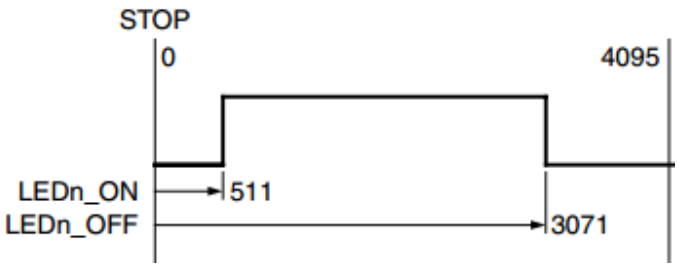
| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 0x06 | 8 LSB Bits of ON PWM count | | | | | | | |
| 0x07 | 0 | 0 | 0 | Full ON | 4 MSB Bits of ON PWM count | | | |
| 0x08 | 8 LSB Bits of OFF PWM count | | | | | | | |
| 0x09 | 0 | 0 | 0 | Full OFF | 4 MSB Bits of ON PWM count | | | |

By default, all led is Fully OFF, so that bit is 1. It needs to be cleared. Preferable keep Full ON and Full OFF bits always 0.

The PWM controller runs from 0 to 4095 constantly, and when counter matches ON PWM count, it will turn led ON, and when it matches OFF PWM count, it will turn led OFF.

e.g. if PWM ON count = 0, led will be turned on at start of cycle, and if OFF PWM count is 2048, led will be turned off at middle of cycle, thus giving pulse width ratio of 50%

In following picture, ON PWM count is set to 511, and OFF PWM count to 3071



In our setup, we use fixed ON PWM counts as follows:

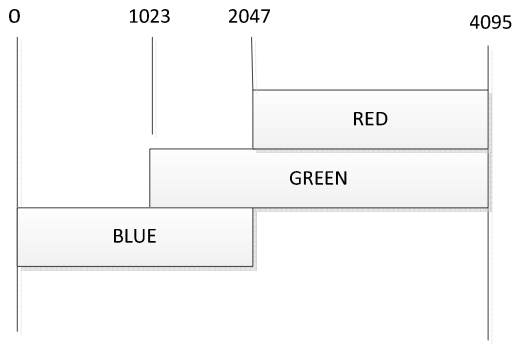
RED = 2047
GREEN = 1023
BLUE = 0

For the RED LED, OFF PWM count must be set to a value between 2047 … 4095.

For the GREEN LED, OFF PWM count must be set to a value between 1023 … 4095.

For the BLUE LED, OFF PWM count must be set to a value between 0 … 4095, but here we keep it between 0 … 2047.

The current implementation takes 8 –bit color value, multiplies it by 8 for RED, by 12 for GREEN and by 8 for BLUE. Following diagram shows how LEDs are lit within single cycle:




RED and BLUE has maximum duty cycle of 50%, GREEN 75%.

All 15 leds of single driver IC are updated with one I2C write starting from address 0x06, which is 60 bytes. The LED15 registers are not written, default state is good for step-up converter.

Single led can be updated by writing corresponding four registers with one I2C write to address 0x06 + 4*LEDn.

As the auto-increment mode is enabled, the register address is automatically advanced per byte. When starting new I2C write, first byte is always the register address.

Example of writing only LED0:

| Start | Device address + W/R bit | Register address e.g 0x06 | Data to register 0x06 | Data to register 0x07 | Data to register 0x08 | Data to register 0x09 | Stop (leds updated) |
|---|---|---|---|---|---|---|---|