Introduction

The iCE40™ family of devices is a high logic, smallest footprint, low I/O count FPGA for smartphone and mobile applications to support multi-functionalities in a single chip solution. It enables BOM integration, providing higher value by board space, power and cost savings. It enables quick implementation of new functionalities without having to wait for the next generation ASIC or application processor to support the new functions. iCE40 is a key hardware differentiating feature for smartphone and mobile devices manufacturers to differentiate their product from other vendors.

The iCE40 family includes the iCE40 HX, iCE40 LP, iCE40 LM, iCE40 Ultra™, iCE40 UltraLite™ and iCE40 UltraPlus™ series of FPGAs. This document describes the features on the LP, LM, Ultra, UltraLite and UltraPlus series for driving LEDs. The iCE40 family includes features such as Embedded RGB PWM IP, high current drive IOs, open drain driver with constant current sinks that enable LED driving applications. The table below compares the features available on each series.

Table 1. iCE40 Devices LED Driver Features Comparison

<table>
<thead>
<tr>
<th>Feature</th>
<th>iCE40LP (16-WLCSP only)</th>
<th>iCE40 LM</th>
<th>iCE40 Ultra</th>
<th>iCE40 UltraLite</th>
<th>iCE40 UltraPlus</th>
</tr>
</thead>
<tbody>
<tr>
<td>24 mA High Drive/High Current Driver for RGB LED</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Open Drain Driver with up to 24 mA Constant Current Sink for RGB LED</td>
<td></td>
<td>x</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Open Drain Driver with up to 500 mA Constant Current Sink for IR LED</td>
<td></td>
<td></td>
<td>x</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Embedded PWM IP to drive RGB LED</td>
<td>x</td>
<td></td>
<td>x</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Open Drain Driver with up to 100 mA Constant Current Sink for BARCODE LED</td>
<td></td>
<td></td>
<td></td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>Embedded Transceiver IP to drive IR LED</td>
<td></td>
<td></td>
<td></td>
<td>x</td>
<td></td>
</tr>
</tbody>
</table>

Focusing on the iCE40 Ultra, iCE40 UltraPlus and iCE40 UltraLite, Embedded PWM IP combined with the three RGB drivers of up to 24-mA current provide all the necessary logic to directly drive the service LED, reducing the need for external components. The up to 500-mA IR driver output provides a direct interface to external LED for applications such as IrDA. For iCE40 Ultra, you can implement the IR transceiver and modulation logic that meet your requirements and connect the IR driver directly to the LED. For iCE40 UltraLite, the embedded IR transceiver IP is built in. These features on the iCE40 Ultra and iCE40 UltraLite allow you to target mobile applications to perform functions such as IrDA, Service LED, Barcode Emulation, and others.

The featured hardened blocks that together form the IR and Service LED driver solution are:

- Embedded PWM IP
- Embedded IR Transceiver IP
- RGB Driver
- IR Driver
- LED Driver Current
- High Current/High Drive IO’s

This document provides detailed information of each block. Details on the Embedded PWM IP are available in other related Lattice documents. See Appendix C. LED Connection Diagrams for detailed connection diagrams showing how the LED driver is connected to the LEDs.

On the iCE40 Ultra, the hardened blocks have to be interconnected for the IR/Service LED driver solution as shown in Figure 1.

**Figure 1. Connectivity Block Diagram**

![Connectivity Block Diagram](image)

Figure 2 shows the location of the hardened blocks on the device.

**Figure 2. iCE40 Ultra Primitive Location Diagram**

![Primitive Location Diagram](image)
Figure 3. iCE40 UltraLite Primitive Location Diagram

Figure 4 shows a system diagram for a typical application using the RGB and IR driver.

Figure 4. System Diagram for Typical Application Using RGB and IR Driver in iCE40 Ultra
As seen from the example above, the SPI/I2C hardened blocks in the iCE40 Ultra and iCE40 UltraLite can be used to efficiently interface the LED driver blocks with an Application Processor.
Embedded RGB PWM IP - iCE40 Ultra

The embedded PWM IP is available as primitive on the iCE40 Ultra, refer to UG75, iCE40 Ultra RGB LED Controller User's Guide for details on ports and functionality.

Embedded RGB PWM IP - iCE40 UltraLite and iCE40 UltraPlus

The LED Driver hard IP provides logic function to drive multi-color LED (R.G.B), with individual brightness control through Pulse Width Modulation (PWM), automatic blinking control and optional breathe on/off control.

Key features of the iCE40 UltraLite and iCE40 UltraPlus RGB PWM IP:

- Configurable from FPGA fabric through 8 bit wide write only data bus.
- Provide 256 level digital PWM brightness control individually for three colors LED (R, G, B).
- User select flick rate between 125 Hz or 250 Hz.
- Single level sensitive pin for easy ON/OFF control.
- Automatic blink control with configurable ON and OFF period.
- Optional breathe ON, breathe OFF capability with adjustable ramp rate with 16 user options.
- PWM output polarity selection.
- User option to skew the PWM output for R.G.B LED in to reduce simultaneous switching noise.
- User option to select PWM mode between square pulses using linear counter approach or PSUDO random pulses using LFSR with spread spectrum.
- Functional system clock frequency range from 4 MHz to 64 MHz.
- Built in brightness monitor for Red, Green and Blue PWM output to help verification, which could be excluded from final synthesis.
Figure 6. iCE40 UltraLite and iCE40 UltraPlus RGB PWM IP Block Diagram

SCI BLOCK
REGISTERS

LEDDBR
LEDDCR0
LEDDONR
LEDDOFR
LEDDBCRR
LEDDBCFR
LEDDPWRR
LEDDPWRG
LEDDPWRB

Base Clock Period Generator

ON/OFF Control

PWM B

PWM G

PWM R

PWMOUT0
PWMOUT1
PWMOUT2

LEDCLK
LEDDCS
LEDDDEN
LEDDADR [3:0]
LEDDDAT [7:0]
LEDDEXE

LEDDON

FPGA Fabric Interface

32 kHz/64 kHz
125 Hz/250 Hz
FLICK Rate Generator
Figure 7. iCE40 UltraLite and iCE40 UltraPlus RGB PWM IP Port Level Diagram

- LEDDCS
- LEDDCLK
- LEDDAT7
- LEDDAT6
- LEDDAT5
- LEDDAT4
- LEDDAT3
- LEDDAT2
- LEDDAT1
- LEDDAT0
- LEDDADDR0
- LEDDADDR1
- LEDDADDR2
- LEDDADDR3
- LEDDDEN
- LEDDEXE
- LEDRST
- SB_LEDDA_IP
- PWMOUT0
- PWMOUT1
- PWMOUT2
- LEDDON
Table 2. iCE40 UltraLite and iCE40 UltraPlus RGB PWM Port List

<table>
<thead>
<tr>
<th>Name</th>
<th>I/O</th>
<th>Level</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>LEDDSC</td>
<td>I</td>
<td>Digital</td>
<td>Chip Select to write SB_LEDDA_IP registers</td>
<td>Active High</td>
</tr>
<tr>
<td>LEDDCLK</td>
<td>I</td>
<td>Digital</td>
<td>Clock to write SB_LEDDA_IP registers</td>
<td></td>
</tr>
<tr>
<td>LEDDAT7</td>
<td>I</td>
<td>Digital</td>
<td>Bit 7 data to write SB_LEDDA_IP registers</td>
<td></td>
</tr>
<tr>
<td>LEDDAT6</td>
<td>I</td>
<td>Digital</td>
<td>Bit 6 data to write SB_LEDDA_IP registers</td>
<td></td>
</tr>
<tr>
<td>LEDDAT5</td>
<td>I</td>
<td>Digital</td>
<td>Bit 5 data to write SB_LEDDA_IP registers</td>
<td></td>
</tr>
<tr>
<td>LEDDAT4</td>
<td>I</td>
<td>Digital</td>
<td>Bit 4 data to write SB_LEDDA_IP registers</td>
<td></td>
</tr>
<tr>
<td>LEDDAT3</td>
<td>I</td>
<td>Digital</td>
<td>Bit 3 data to write SB_LEDDA_IP registers</td>
<td></td>
</tr>
<tr>
<td>LEDDAT2</td>
<td>I</td>
<td>Digital</td>
<td>Bit 2 data to write SB_LEDDA_IP registers</td>
<td></td>
</tr>
<tr>
<td>LEDDAT1</td>
<td>I</td>
<td>Digital</td>
<td>Bit 1 data to write SB_LEDDA_IP registers</td>
<td></td>
</tr>
<tr>
<td>LEDDAT0</td>
<td>I</td>
<td>Digital</td>
<td>Bit 0 data to write SB_LEDDA_IP registers</td>
<td></td>
</tr>
<tr>
<td>LEDDADDR3</td>
<td>I</td>
<td>Digital</td>
<td>Bit 3 address to write SB_LEDDA_IP registers</td>
<td></td>
</tr>
<tr>
<td>LEDDADDR2</td>
<td>I</td>
<td>Digital</td>
<td>Bit 2 address to write SB_LEDDA_IP registers</td>
<td></td>
</tr>
<tr>
<td>LEDDADDR1</td>
<td>I</td>
<td>Digital</td>
<td>Bit 1 address to write SB_LEDDA_IP registers</td>
<td></td>
</tr>
<tr>
<td>LEDDADDR0</td>
<td>I</td>
<td>Digital</td>
<td>Bit 0 address to write SB_LEDDA_IP registers</td>
<td></td>
</tr>
<tr>
<td>LEDDEN</td>
<td>I</td>
<td>Digital</td>
<td>Data enable to indicate data and address are stable</td>
<td>Active High</td>
</tr>
<tr>
<td>LEDDEXE</td>
<td>I</td>
<td>Digital</td>
<td>Enable the IP to run the blinking sequence. When is low, the sequence stop at the nearest OFF state</td>
<td>Active High</td>
</tr>
<tr>
<td>LEDDRST</td>
<td>I</td>
<td>Digital</td>
<td>Reset all registers in the IP</td>
<td>Active High</td>
</tr>
<tr>
<td>PWMOUT0</td>
<td>O</td>
<td>Digital</td>
<td>Goes to SB_RGBA_DRV, IO Driver for RED LED or FPGA Fabric</td>
<td></td>
</tr>
<tr>
<td>PWMOUT1</td>
<td>O</td>
<td>Digital</td>
<td>Goes to SB_RGBA_DRV, IO Driver for GREEN LED or FPGA Fabric</td>
<td></td>
</tr>
<tr>
<td>PWMOUT2</td>
<td>O</td>
<td>Digital</td>
<td>Goes to SB_RGBA_DRV, IO Driver for BLUE LED or FPGA Fabric</td>
<td></td>
</tr>
<tr>
<td>LEDDON</td>
<td>O</td>
<td>Digital</td>
<td>Goes to FPGA routing, indicating LED is on</td>
<td></td>
</tr>
</tbody>
</table>

LEDDCLK
The clock [LEDDCLK] input coordinates all activities for the internal logic within the LED Control Bus interconnect. And it also served as base clock source for all LED drive IP timing and PWM functionality. All LED Control Bus output signals are registered at the rising edge of [LEDDCLK]. All LED Control Bus input signals are stable before the rising edge of [LEDDCLK].

LEDDCS
The Chip Select [LEDDCS] input activate the LED Driver IP block to allow LED Control Bus to communicate to it. This usually connects to the output of the decoding logic from MSB of the address bus, in order to share the same digital bus with other IPs or instances.

LEDDAT[7:0]
The data input array LEDDAT [7:0] is used to pass binary data. The array boundaries are determined by the port size = 8.

LEDDADDR[3:0]
The address input array LEDDADDR [3:0] is used to pass a binary address.

LEDDDEN
The Data Enable input LEDDDEN, when asserted, indicates that the data and address on the LED Control Bus are stabilized and ready to be captured. All register on the LED Control Bus only response to the data and address bus when this LEDDDEN is asserted.
The LED Driver Execute input LEDDEXE, when asserted, starts the LED Driver IP to run the blinking sequence according to the setup defined in the control registers. The LED Driver will keep on repeating the sequence while LEDDEXE remains HIGH. When LEDDEXE goes LOW, the sequence will stop at the nearest OFF state.

An active high Power-On Reset for the whole device, include this IP.

A typical LED Control Bus write operation is demonstrated in Figure 8.

Figure 8. LED Control Bus Write Operation

For details about RGB PWM IP registers, please refer to Appendix D. RGB PWM IP - LED Control Bus Addressable Registers.

Embedded IR Transceiver IP

The IR Transceiver hard IP provides logic function to transmit and receive data through Infrared LED data link. It takes the data from soft IP residing in the FPGA fabric to transmit with user specified frequency. In user enabled learning mode, it receives data from Infrared receiver and sends the received data back to the FPGA fabric along with the measured receiving frequency. The IR Transceiver IP communicates with the host via an 8 bit wide digital bus.

Key features of the iCE40 UltraLite IR Transceiver IP:

- Functional system clock frequency range from 12 MHz to 64 MHz.
- User select IR transmits clock frequency from 25 kHz to 120 kHz.
- Learning mode to discover transmit frequency and receiving data (ON/OFF counts).
- User configurable input digital filter to filter out input noise less than N system clock cycles.
- User selectable output polarity.
- User selectable duty cycle for the ON pulses, 1/2 or 1/3 of the transceiver clock period.
- Optional user specified maximum pulse count in learning mode to save time at the end of IR command sequence.
- User option for the transceiver clock frequency evaluation occurrence, once at the beginning of the IR command sequence or at beginning of every ON pulse group.
- Configurable from FPGA fabric through 8 bit wide write data bus and 8 bit read data bus.
Figure 9. iCE40 UltraLite IR Transceiver IP Block Diagram
Figure 10. iCE40 UltraLite IR Transceiver IP Port Level Diagram

- CLKI
- CSI
- DENI
- WEI
- ADRI[3:0]
- WDATA[7:0]
- RDATA[7:0]
- BUSY
- DRDY
- ERR
- EXE
- LEARN
- RST
- IRIN
- IROUT
Table 3. iCE40 UltraLite IR Transceiver Port List

<table>
<thead>
<tr>
<th>Name</th>
<th>I/O</th>
<th>Level</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLKI</td>
<td>I</td>
<td>Digital</td>
<td>Clock input for IR IP</td>
<td></td>
</tr>
<tr>
<td>CSI</td>
<td>I</td>
<td>Digital</td>
<td>Select Signal to activate the IP. This usually connects to the output of the decoding logic from MSB of the address bus.</td>
<td>Active High</td>
</tr>
<tr>
<td>DENI</td>
<td>I</td>
<td>Digital</td>
<td>Data Enable. When asserted, indicates that the data and address on the IR Transceiver Control Bus are stabilized and ready to be captured.</td>
<td>Active High</td>
</tr>
<tr>
<td>WEI</td>
<td>I</td>
<td>Digital</td>
<td>Data Write Enable. Asserted during WRITE and de-asserted during READ cycle.</td>
<td>Active High</td>
</tr>
<tr>
<td>ADRI3</td>
<td>I</td>
<td>Digital</td>
<td>Control Register Address Bit 3</td>
<td></td>
</tr>
<tr>
<td>ADRI2</td>
<td>I</td>
<td>Digital</td>
<td>Control Register Address Bit 2</td>
<td></td>
</tr>
<tr>
<td>ADRI1</td>
<td>I</td>
<td>Digital</td>
<td>Control Register Address Bit 1</td>
<td></td>
</tr>
<tr>
<td>ADRI0</td>
<td>I</td>
<td>Digital</td>
<td>Control Register Address Bit 0</td>
<td></td>
</tr>
<tr>
<td>WDATA7</td>
<td>I</td>
<td>Digital</td>
<td>Write Data Input Bit 7</td>
<td></td>
</tr>
<tr>
<td>WDATA6</td>
<td>I</td>
<td>Digital</td>
<td>Write Data Input Bit 6</td>
<td></td>
</tr>
<tr>
<td>WDATA5</td>
<td>I</td>
<td>Digital</td>
<td>Write Data Input Bit 5</td>
<td></td>
</tr>
<tr>
<td>WDATA4</td>
<td>I</td>
<td>Digital</td>
<td>Write Data Input Bit 4</td>
<td></td>
</tr>
<tr>
<td>WDATA3</td>
<td>I</td>
<td>Digital</td>
<td>Write Data Input Bit 3</td>
<td></td>
</tr>
<tr>
<td>WDATA2</td>
<td>I</td>
<td>Digital</td>
<td>Write Data Input Bit 2</td>
<td></td>
</tr>
<tr>
<td>WDATA1</td>
<td>I</td>
<td>Digital</td>
<td>Write Data Input Bit 1</td>
<td></td>
</tr>
<tr>
<td>WDATA0</td>
<td>I</td>
<td>Digital</td>
<td>Write Data Input Bit 0</td>
<td></td>
</tr>
<tr>
<td>RDAT7</td>
<td>O</td>
<td>Digital</td>
<td>Read Data Output Bit 7</td>
<td></td>
</tr>
<tr>
<td>RDAT6</td>
<td>O</td>
<td>Digital</td>
<td>Read Data Output Bit 6</td>
<td></td>
</tr>
<tr>
<td>RDAT5</td>
<td>O</td>
<td>Digital</td>
<td>Read Data Output Bit 5</td>
<td></td>
</tr>
<tr>
<td>RDAT4</td>
<td>O</td>
<td>Digital</td>
<td>Read Data Output Bit 4</td>
<td></td>
</tr>
<tr>
<td>RDAT3</td>
<td>O</td>
<td>Digital</td>
<td>Read Data Output Bit 3</td>
<td></td>
</tr>
<tr>
<td>RDAT2</td>
<td>O</td>
<td>Digital</td>
<td>Read Data Output Bit 2</td>
<td></td>
</tr>
<tr>
<td>RDAT1</td>
<td>O</td>
<td>Digital</td>
<td>Read Data Output Bit 1</td>
<td></td>
</tr>
<tr>
<td>RDAT0</td>
<td>O</td>
<td>Digital</td>
<td>Read Data Output Bit 0</td>
<td></td>
</tr>
<tr>
<td>EXE</td>
<td>I</td>
<td>Digital</td>
<td>Execute. when asserted, starts the IR Transceiver Hard IP to transmit or receive IR data.</td>
<td>Active High</td>
</tr>
<tr>
<td>LEARN</td>
<td>I</td>
<td>Digital</td>
<td>Learning Mode control. when asserted, the IR Transceiver is in learning mode. The IR Transceiver will receive data instead of transmit data.</td>
<td>Active High</td>
</tr>
<tr>
<td>BUSY</td>
<td>O</td>
<td>Digital</td>
<td>Busy status output</td>
<td></td>
</tr>
<tr>
<td>DRDY</td>
<td>O</td>
<td>Digital</td>
<td>Data Buffer Ready status output</td>
<td></td>
</tr>
<tr>
<td>ERR</td>
<td>O</td>
<td>Digital</td>
<td>Data Error status</td>
<td></td>
</tr>
<tr>
<td>RST</td>
<td>I</td>
<td>Digital</td>
<td>System Reset. When asserted, all SCI registers will be reset to Zero and IROUT will be reset to OFF state.</td>
<td></td>
</tr>
<tr>
<td>IRIN</td>
<td>I</td>
<td>Digital</td>
<td>Modulated ON/OFF pulse from IR sensor</td>
<td></td>
</tr>
<tr>
<td>IROUT</td>
<td>O</td>
<td>Digital</td>
<td>Modulated ON/OFF pulse for IR Transmit</td>
<td></td>
</tr>
</tbody>
</table>

**CLKI**
The clock input coordinates all activities for the internal logic within the IR Transceiver Control Bus interconnect. And it also served as base clock source for all IR Transceiver Hard IP timing and Modulation functionality. All IR Transceiver Control Bus output signals are registered at the rising edge of CLKI. All IR Transceiver Control Bus input signals are stable before the rising edge of CLKI.
**CSI**
The Chip Select input activates the IR Transceiver Hard IP block to allow IR Transceiver Control Bus to communicate to it. This usually connects to the output of the decoding logic from MSB of the address bus, in order to share the same digital bus with other IPs or instances.

**DENI**
The Data Enable input, when asserted, indicates that the data and address on the IR Transceiver Control Bus are stabilized and ready to be captured. All register on the IR Transceiver Control Bus only respond to the data and address bus when this DENI is asserted.

**WEI**
The Data Write Enable input indicates whether the current IR Transceiver Control Bus is a READ or WRITE cycle. The signal is negated during READ cycles, and is asserted during WRITE cycles.

**ADRI[3:0]**
The address input array ADRI [3:0] is used to pass a binary address.

**WDATA[7:0]**
The data input array is used to pass binary data for write.

**RDATA[7:0]**
The data input array is used to pass binary data for read.

A typical LED Control Bus read operation is demonstrated in Figure 11.

*Figure 11. Typical IR Transceiver Control Bus Read Operation*
iCE40 Ultra RGB Driver

iCE40 Ultra supports an RGB DRV hardened IP block that provides high current drive outputs. This allows the iCE40 Ultra device to drive Service LED signals directly, reducing the need for an external component. There is one such block per device located at the top IO bank. The RGB LED driver block provides an open-drain driver for the LED DIODE with constant current from 4 mA to 24 mA in 4 mA step with +/-10% accuracy. Each of the 4 mA steps is controlled by an hdl attribute. The LED driver reference can be enabled within 100 µs time.

Key features of the iCE40 Ultra RGB Driver:

- Supports three Service LEDs (RGB) with sink current between 4 mA and 24 mA in steps of 4 mA per device ball.
- Supports pins being independently configured as either a high-current sink or an OD GPIO.
- Accuracy of within ±10% of the amount of current being sunk at all steps when the voltage at the device pin is at least 0.5 V. Current matching within ±5% across all three Service LEDs for the same current sink setting (for example, if all three LED pins are programmed to sink 12 mA, their actual sink current is within 5% of each other in the worst case.)
- Consumes ≤ 0.5 µA typical static current (typical, 1.2 V, 25 °C) and ≤ 1 µA max static leakage current per device ball when operating in standby mode (LED off). Consumes ≤ 1.0 mA of current (typical, 1.2 V, 25 °C) per device ball associated with the support circuitry (excluding the actual current being sunk) when operating in LED on mode.
- Wakeup time (from off to on -- fully functional) ≤ 100 µsec.

For details about IR Transceiver IP registers, please refer to Appendix E. IR Transceiver IP.
Figure 13. iCE40 Ultra RGB Driver Block Diagram

Figure 14. iCE40 Ultra RGB Port Level Diagram
Table 4. iCE40 Ultra RGB Port List

<table>
<thead>
<tr>
<th>Name</th>
<th>I/O</th>
<th>Level</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>RGB0</td>
<td>O</td>
<td>IOPAD</td>
<td>24 mA RGB PAD</td>
<td>64 kHz</td>
</tr>
<tr>
<td>RGB1</td>
<td>O</td>
<td>IOPAD</td>
<td>24 mA RGB PAD</td>
<td>64 kHz</td>
</tr>
<tr>
<td>RGB2</td>
<td>O</td>
<td>IOPAD</td>
<td>24 mA RGB PAD</td>
<td>64 kHz</td>
</tr>
<tr>
<td>RGBLEDEN</td>
<td>I</td>
<td>Digital</td>
<td>Enable Control for RGB LED</td>
<td>Active HIGH</td>
</tr>
<tr>
<td>RGB0PWM</td>
<td>I</td>
<td>Digital</td>
<td>Pulse width modulated control signal for RGB_PAD0</td>
<td>64 kHz, Active HIGH</td>
</tr>
<tr>
<td>RGB1PWM</td>
<td>I</td>
<td>Digital</td>
<td>Pulse width modulated control signal for RGB_PAD1</td>
<td>64 kHz, Active HIGH</td>
</tr>
<tr>
<td>RGB2PWM</td>
<td>I</td>
<td>Digital</td>
<td>Pulse width modulated control signal for RGB_PAD2</td>
<td>64 kHz, Active HIGH</td>
</tr>
<tr>
<td>RGBPU</td>
<td>I</td>
<td>Analog</td>
<td>Power up</td>
<td>Connects to LED_DRV_CUR primitive.</td>
</tr>
</tbody>
</table>

RGB0
Open-drain output of the RGB Driver connected to the device pin for RED LED

RGB1
Open-drain output of the RGB Driver connected to the device pin for GREEN LED

RGB2
Open-drain output of the RGB Driver connected to the device pin for BLUE LED

RGBLEDEN
Input to the RGB Driver, Enable Control for RGB LED, Active HIGH

RGB0PWM
Input to the RGB Driver, pulse width modulated control signal for controlling RGB0 output. Connects to Embedded PWM IP or FPGA logic, Active HIGH

RGB1PWM
Input to the RGB Driver, pulse width modulated control signal for controlling RGB1 output. Connects to Embedded PWM IP or FPGA logic, Active HIGH

RGB2PWM
Input to the RGB Driver, pulse width modulated control signal for controlling RGB2 output. Connects to Embedded PWM IP or FPGA logic, Active HIGH

RGBPU
Input to the RGB Driver, reference current signal must be connected to output of RGB CUR Driver primitive

**SB_RGB_DRV Attribute Description**

The SB_RGB_DRV primitive contains the following parameter and their default values:

Parameter RGB0_CURRENT = "0b00000000";
Parameter RGB1_CURRENT = "0b00000000";
Parameter RGB2_CURRENT = "0b00000000";

Parameter values:

"0b00000000" = 0mA. // Set this value to use the associated SB_IO_OD instance at RGB LED location.
"0b00000001" = 4 mA
"0b00000011" = 8 mA
"0b00000111" = 12 mA
"0b001111" = 16 mA
"0b011111" = 20 mA
"0b111111" = 24 mA

RGB PAD can also be used as an open-drain GPIO with LVCMOS. These are the differences in characteristic compare to regular iCE40 GPIO.

- No P-channel pull up driver.
- No weak pull up.
- LVCMOS input buffer will be power down when using as RGB Driver.

**iCE40 UltraLite and iCE40 UltraPlus RGB Driver**

iCE40 UltraLite and iCE40 UltraPlus support an RGB DRV hardened IP block that provides high current drive outputs. This allows the iCE40 UltraLite and iCE40 UltraPlus devices to drive Service LED signals directly, reducing the need for an external component. There is one such block per device located at the top IO bank. The RGB LED driver block provides an open-drain driver for the LED DIODE with constant current from 4 mA to 24 mA in 4 mA step in full current mode or from 2 mA to 12 mA in 2 mA step in half current mode with up to +/-10% accuracy. Each of the steps is controlled by an hdl attribute. The LED driver reference can be enabled within 100 µs time.

Key features of the iCE40 UltraLite and iCE40 UltraPlus RGB Driver:

- Supports three Service LEDs (RGB) with sink current between 4 mA and 24 mA in steps of 4 mA or 2 mA and 12 mA in steps of 2 mA per device ball.
- Supports pins being independently configured as either a high-current sink or an OD GPIO.
- Accuracy of up to ± 10% of the amount of current being sunk at all steps when the voltage at the device pin is at least 0.5 V. Current matching within ± 5% across all three Service LEDs for the same current sink setting (for example, if all three LED pins are programmed to sink 12 mA, their actual sink current is within 5% of each other in the worst case.)
- Consumes ≤ 0.5 µA typical static current (typical, 1.2 V, 25 °C) and ≤ 1 µA max static leakage current per device ball when operating in standby mode (LED off). Consumes ≤ 1.0 mA of current (typical, 1.2 V, 25 °C) per device ball associated with the support circuitry (excluding the actual current being sunk) when operating in LED on mode.
- Wakeup time (from off to on -- fully functional) ≤ 100 µsec.
Figure 15. iCE40 UltraLite and iCE40 UltraPlus RGB Driver Block Diagram

Figure 16. iCE40 UltraLite and iCE40 UltraPlus RGB Port Level Diagram
Table 5. iCE40 UltraLite and iCE40 UltraPlus RGB Port List

<table>
<thead>
<tr>
<th>Name</th>
<th>I/O</th>
<th>Level</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>RGB0</td>
<td>O</td>
<td>IOPAD</td>
<td>Open-drain output of the RGB Driver connected to</td>
<td>Up to 24 mA RGB PAD</td>
</tr>
<tr>
<td>RGB1</td>
<td>O</td>
<td>IOPAD</td>
<td>the device pin for RED LED</td>
<td></td>
</tr>
<tr>
<td>RGB2</td>
<td>O</td>
<td>IOPAD</td>
<td>Open-drain output of the RGB Driver connected to</td>
<td>Up to 24 mA RGB PAD</td>
</tr>
<tr>
<td>RGBLEDEN</td>
<td>I</td>
<td>Digital</td>
<td>Enable Control for RGB LED</td>
<td>Active HIGH</td>
</tr>
<tr>
<td>RGB0PWM</td>
<td>I</td>
<td>Digital</td>
<td>Pulse width modulated control signal for RGB0</td>
<td>64 kHz, Active HIGH</td>
</tr>
<tr>
<td>RGB1PWM</td>
<td>I</td>
<td>Digital</td>
<td>Pulse width modulated control signal for RGB1</td>
<td>64 kHz, Active HIGH</td>
</tr>
<tr>
<td>RGB2PWM</td>
<td>I</td>
<td>Digital</td>
<td>Pulse width modulated control signal for RGB2</td>
<td>64 kHz, Active HIGH</td>
</tr>
<tr>
<td>CURREN</td>
<td>I</td>
<td>Digital</td>
<td>Power up</td>
<td>Power up signal, Active HIGH</td>
</tr>
</tbody>
</table>

**RGB0**
Open-drain output of the RGB Driver connected to the device pin for RED LED

**RGB1**
Open-drain output of the RGB Driver connected to the device pin for GREEN LED

**RGB2**
Open-drain output of the RGB Driver connected to the device pin for BLUE LED

**RGBLEDEN**
Input to the RGB Driver, Enable Control for RGB LED, Active HIGH

**RGB0PWM**
Input to the RGB Driver, pulse width modulated control signal for controlling RGB0 output. Connects to Embedded PWM IP or FPGA logic, Active HIGH

**RGB1PWM**
Input to the RGB Driver, pulse width modulated control signal for controlling RGB1 output. Connects to Embedded PWM IP or FPGA logic, Active HIGH

**RGB2PWM**
Input to the RGB Driver, pulse width modulated control signal for controlling RGB2 output. Connects to Embedded PWM IP or FPGA logic, Active HIGH

**CURREN**
Input enabling mixed signal control block to supply reference current to RGB driver. Enabling the mixed signal control block takes 100 µs to reach a stable reference current value.
SB_RGBA_DRV Attribute Description
The SB_RGBA_DRV primitive contains the following parameter and their default values:

Parameter CURRENT_MODE = “0b0”;
Parameter RGB0_CURRENT = “0b0000000”;
Parameter RGB1_CURRENT = “0b0000000”;
Parameter RGB2_CURRENT = “0b0000000”;

Parameter values:

“0b0” = Full Current Mode
“0b1” = Half Current Mode
“0b000000” = 0mA. // Set this value to use the associated SB_IO_OD instance at RGB LED location.
“0b000001” = 4 mA for Full Mode; 2 mA for Half Mode
“0b000011” = 8 mA for Full Mode; 4 mA for Half Mode
“0b000111” = 12 mA for Full Mode; 6mA for Half Mode
“0b001111” = 16 mA for Full Mode; 8 mA for Half Mode
“0b011111” = 20 mA for Full Mode; 10 mA for Half Mode
“0b111111” = 24 mA for Full Mode; 12 mA for Half Mode

RGB PAD can also be used as an open-drain GPIO with LVCMOS. These are the differences in characteristic compare to regular iCE40 GPIO.

- No P-channel pull up driver.
- No weak pull up.
- LVCMOS input buffer will be power down when using as RGB Driver.
ICE40 Ultra IR Driver

ICE40 Ultra supports a single IR DRV IP block located at the top IO bank. The IR driver output provides a direct interface to external LED for applications such as IrDA functions.

The user simply implements the modulation logic that meets his needs, and connects the IR driver directly to the LED, reducing the need for external component. The IR LED driver block provides open-drain driver for IR LED DIODE with constant current from 50 mA to 500 mA in steps of 50 mA with +/-10% accuracy. Each of the 50 mA steps is controlled by an hdl attribute. The IR LED driver reference can be enabled within 100 µs time.

Key features of the ICE40 UltraLite IR Driver:

- Supports one IR LED with sink current between 50 mA and 500 mA in 50 mA steps.
- Supports pins being independently configured as either a high-current sink or an OD GPIO.
- Accuracy of within ±10% of the amount of current being sunk at all steps when the voltage at the device pin is at least 0.8 V.
- Consumes ≤ 5 µA static current (typical, 1.2 V, 25 °C) and ≤ 10 µA max static leakage current per device ball when operating in standby mode (LED off) and consume ≤ 1.0 mA of current (typical, 1.2 V, 25 °C) per device ball associated with the support circuitry (excluding the actual current being sunk) when operating in LED on mode.
- Wakeup time (from off to on -- fully functional) ≤ 100 µsec.

Figure 17. Functional Equivalent Block Diagram
Figure 18. IR Port Level

Table 6. IR Port List

<table>
<thead>
<tr>
<th>Name</th>
<th>I/O</th>
<th>Level</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>IRLED</td>
<td>O</td>
<td>OPAD</td>
<td>Up to 400 mA IR PAD</td>
<td>100 kHz</td>
</tr>
<tr>
<td>IRLEDEN</td>
<td>I</td>
<td>Digital</td>
<td>Enable Control for IR LED</td>
<td>Active HIGH</td>
</tr>
<tr>
<td>IRPWM</td>
<td>I</td>
<td>Digital</td>
<td>Pulse width modulated control signal for IR_PAD</td>
<td>100 kHz, Active HIGH</td>
</tr>
<tr>
<td>IRPU</td>
<td>I</td>
<td>Digital</td>
<td>Connects to LED_DRV_CUR primitive</td>
<td>Power up signal</td>
</tr>
</tbody>
</table>

Signal Description

**IRLED**
Output of the IR Driver connected to the device pin for IR LED

**IRLEDEN**
Input to the IR Driver, Enable Control for IR LED, Active HIGH

**IRPWM**
Input to the IR Driver, pulse width modulated control signal for controlling IRLED output. Connects to FPGA logic, Active HIGH

**IRPU**
Input to the RGB Driver, reference current signal must be connected to output of RGB CUR Driver primitive

**SB_IR_DRV Attribute Description**
The SB_IR_DRV primitive contains the following parameter and their default values:

Parameter IR_CURRENT = “0b0000000000”;

Parameter Values:

“0b0000000000” = 0 mA. // Set this value to use the associated SB_IO_OD instance at IR LED location.
“0b0000000001” = 50 mA
“0b0000000011” = 100 mA
“0b0000000111” = 150mA
“0b0000001111” = 200 mA
“0b0000011111” = 250 mA
“0b0000111111” = 300 mA
“0b0011111111” = 350 mA
“0b1111111111” = 400 mA
“0b0111111111” = 450 mA
“0b1111111111” = 500 mA
iCE40 UltraLite IR 400 mA Driver

iCE40 UltraLite supports a single IR DRV IP block located at the top IO bank. The IR driver output provides a direct interface to external LED for applications such as IrDA functions.

The user simply implements the modulation logic that meets his needs, and connects the IR driver directly to the LED, reducing the need for external component. The IR LED driver block provides open-drain driver for IR LED DIODE with constant current from 50 mA to 400 mA in steps of 50 mA in full current mode or from 25 mA to 200 mA in steps of 25 mA in half current mode with up to +/-10% accuracy. Each of the steps is controlled by an hdl attribute. The IR LED driver reference can be enabled within 100 µs time.

Key features of the iCE40 UltraLite 400 mA IR Driver:

- Supports one IR LED with sink current between 50 mA and 400 mA in 50 mA steps in full current mode or between 25 mA and 200 mA in 25 mA steps.
- Supports pins being independently configured as either a high-current sink or an OD GPIO.
- Accuracy of up to ±10% of the amount of current being sunk at all steps when the voltage at the device pin is at least 0.8 V.
- Consumes ≤5 µA static current (typical, 1.2 V, 25 °C) and ≤10 µA max static leakage current per device ball when operating in standby mode (LED off) and consume ≤1.0 mA of current (typical, 1.2 V, 25 °C) per device ball associated with the support circuitry (excluding the actual current being sunk) when operating in LED on mode.
- Wakeup time (from off to on -- fully functional) ≤100 µsec.

Figure 19. Functional Equivalent Block Diagram
Signal Description

**IRLED**
Output of the IR Driver connected to the device pin for IR LED

**IRLEDEN**
Input to the IR Driver, Enable Control for IR LED, Active HIGH

**IRPWM**
Input to the IR Driver, pulse width modulated control signal for controlling IRLED output. Connects to FPGA logic, Active HIGH

**CURREN**
Input enabling mixed signal control block to supply reference current to RGB driver. Enabling the mixed signal control block takes 100 µs to reach a stable reference current value.

**SB_IR400_DRV Attribute Description**
The SB_IR400_DRV primitive contains the following parameter and their default values:

Parameter CURRENT_MODE = "0b0";
Parameter IR400_CURRENT = "0b0000000000";

Parameter Values:

"0b0" = Full Current Mode. // SB_BARDODE_DRV and SB_IR400_DRV are sharing same bit for Current Mode. So they have to be either all Full Current Mode or all Half Current Mode.

"0b1" = Half Current Mode

"0b00000000" = 0 mA. // Set this value to use the associated SB_IO_OD instance at IR LED location.

"0b00000001" = 50 mA for Full Mode; 25 mA for Half Mode

"0b00000011" = 100 mA for Full Mode; 50 mA for Half Mode

"0b00000111" = 150 mA for Full Mode; 75 mA for Half Mode

"0b00001111" = 200 mA for Full Mode; 100 mA for Half Mode

"0b00011111" = 250 mA for Full Mode; 125 mA for Half Mode

"0b00111111" = 300 mA for Full Mode; 150 mA for Half Mode

"0b01111111" = 350 mA for Full Mode; 175 mA for Half Mode

"0b11111111" = 400 mA for Full Mode; 200 mA for Half Mode
iCE40 UltraLite Barcode Driver

The BARCODE driver output provides a direct interface to external LED for an application such as BARCODE scan.

The user simply implements the logic that meets his needs, and connects the BARCODE driver directly to the LED, reducing the need for external component. The BARCODE LED driver block provides open-drain driver for BARCODE LED DIODE with constant current from 0 mA to 100 mA in steps of 16.66 mA in full current mode or from 0 mA to 50 mA in steps of 8.3 mA in half current mode with up to +/-10% accuracy. Each of the steps is controlled by an hdl attribute. The BARCODE LED driver reference can be enabled within 100 µs time.

Key features of the iCE40 UltraLite BARCODE Driver:

- Supports one BARCODE LED with sink current between 0 mA and 100 mA in 16.6 mA steps in full current mode or between 0 mA and 50 mA in 8.3 mA steps.
- Supports pins being independently configured as either a high-current sink or an OD GPIO.
- Accuracy of up to ± 10% of the amount of current being sunk at all steps when the voltage at the device pin is at least 0.8 V.
- Consumes ≤ 5 µA static current (typical, 1.2 V, 25 °C) and ≤ 10 µA max static leakage current per device ball when operating in standby mode (LED off) and consume ≤ 1.0 mA of current (typical, 1.2 V, 25 °C) per device ball associated with the support circuitry (excluding the actual current being sunk) when operating in LED on mode.
- Wakeup time (from off to on -- fully functional) ≤ 100 µsec.

*Figure 21. Functional Equivalent Block Diagram*
Figure 22. IR Port Level

Table 8. IR Port List

<table>
<thead>
<tr>
<th>Name</th>
<th>I/O</th>
<th>Level</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>BARCODE</td>
<td>O</td>
<td>OPAD</td>
<td>Up to 100 mA BARCODE PAD</td>
<td>100 kHz</td>
</tr>
<tr>
<td>BARCODEEN</td>
<td>I</td>
<td>Digital</td>
<td>Enable Control for BARCODE LED</td>
<td>Active HIGH</td>
</tr>
<tr>
<td>BARCODEPWM</td>
<td>I</td>
<td>Digital</td>
<td>Pulse width modulated control signal for BARCODE_PWM</td>
<td>100 kHz, Active HIGH</td>
</tr>
<tr>
<td>CURREN</td>
<td>I</td>
<td>Digital</td>
<td>Power up</td>
<td>Power up signal, Active HIGH</td>
</tr>
</tbody>
</table>

Signal Description

BARCODE
Output of the BARCODE Driver connected to the device pin for BARCODE LED.

BARCODEEN
Input to the BARCODE Driver, Enable Control for BARCODE LED, Active HIGH.

BARCODEPWM
Input to the BARCODE Driver, pulse width modulated control signal for controlling BARCODE output. Connects to FPGA logic, Active HIGH.

CURREN
Input enabling mixed signal control block to supply reference current to RGB driver. Enabling the mixed signal control block takes 100 µs to reach a stable reference current value.

SB_BARCODE_DRV Attribute Description

The SB_BARCODE_DRV primitive contains the following parameter and their default values:

Parameter CURRENT_MODE = "0b0";
Parameter BARCODE_CURRENT = "0b0000000000";

Parameter Values:

"0b0" = Full Current Mode. // SB_BARCODE_DRV and SB_IR400_DRV are sharing same bit for Current Mode. So they have to be either all Full Current Mode or all Half Current Mode.
"0b1" = Half Current Mode
"0b0000" = 0 mA. // Set this value to use the associated SB_IO_OD instance at BARCODE LED location.
"0b0001" = 16.6 mA for Full Mode; 8.3 mA for Half Mode
"0b0011" = 33.3 mA for Full Mode; 16.6 mA for Half Mode
"0b0111" = 50 mA for Full Mode; 25 mA for Half Mode
"0b1001" = 66.6 mA for Full Mode; 33.3 mA for Half Mode
"0b1010" = 83.8 mA for Full Mode; 41.6 mA for Half Mode
"0b1111" = 100 mA for Full Mode; 50 mA for Half Mode
iCE40 UltraLite IR 500 mA Driver

iCE40 UltraLite provides a way to combine IR driver and BARCODE driver to provide up to 500 mA sink current IR driver. The IR driver output provides a direct interface to external LED for applications such as IrDA functions.

The user simply implements the modulation logic that meets his needs, and connects the IR driver directly to the LED, reducing the need for external component. The IR LED driver block provides open-drain driver for IR LED DIODE with constant current from 50 mA to 400 mA in steps of 50 mA in full current mode or from 25 mA to 200 mA in steps of 25 mA in half current mode with up to +/-10% accuracy. Each of the steps is controlled by an hdl attribute. The IR LED driver reference can be enabled within 100 µs time.

Key features of the iCE40 UltraLite 500 mA IR Driver:

- BARCODE pad and IR pad need to be shorted together on the board level.
- Supports one IR LED with sink current between 50 mA and 500 mA in 50 mA steps in full current mode or between 25 mA and 250 mA in 25 mA steps.
- Supports pins being independently configured as either a high-current sink or an OD GPIO.
- Accuracy of up to ± 10% of the amount of current being sunk at all steps when the voltage at the device pin is at least 0.8 V.
- Consumes ≤ 5 µA static current (typical, 1.2 V, 25 °C) and ≤ 10 µA max static leakage current per device ball when operating in standby mode (LED off) and consume ≤ 1.0 mA of current (typical, 1.2 V, 25 °C) per device ball associated with the support circuitry (excluding the actual current being sunk) when operating in LED on mode.
- Wakeup time (from off to on -- fully functional) ≤ 100 µsec.

*Figure 23. Functional Equivalent Block Diagram*
Figure 24. IR Port Level

Table 9. IR Port List

<table>
<thead>
<tr>
<th>Name</th>
<th>I/O</th>
<th>Level</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>IRLED1</td>
<td>O</td>
<td>OPAD</td>
<td>Up to 400 mA IR PAD</td>
<td>100 kHz</td>
</tr>
<tr>
<td>IRLED2</td>
<td>O</td>
<td>OPAD</td>
<td>Up to 100 mA BARCODE PAD</td>
<td>100 kHz</td>
</tr>
<tr>
<td>IRLEDEN</td>
<td>I</td>
<td>Digital</td>
<td>Enable Control for IR LED</td>
<td>Active HIGH</td>
</tr>
<tr>
<td>IRPWM</td>
<td>I</td>
<td>Digital</td>
<td>Pulse width modulated control signal for IR_PAD</td>
<td>100 kHz, Active HIGH</td>
</tr>
<tr>
<td>CURRENT</td>
<td>I</td>
<td>Digital</td>
<td>Power up</td>
<td>Power up signal, Active HIGH</td>
</tr>
</tbody>
</table>

Signal Description

**IRLED1**
Output of the IR Driver connected to the device pin for IR LED

**IRLED2**
Output of the Barcode Driver connected to the device pin for Barcode LED

**IRLEDEN**
Input to the IR Driver, Enable Control for IR LED, Active HIGH

**IRPWM**
Input to the IR Driver, pulse width modulated control signal for controlling IRLED output. Connects to FPGA logic, Active HIGH

**CURRENT**
Input enabling mixed signal control block to supply reference current to RGB driver. Enabling the mixed signal control block takes 100 µs to reach a stable reference current value.
SB_IR500_DRV Attribute Description

The SB_IR500_DRV primitive contains the following parameter and their default values:

Parameter CURRENT_MODE = "0b0";
Parameter IR400_CURRENT = "0b00000000000000";

Parameter Values:

- "0b0" = Full Current Mode
- "0b1" = Half Current Mode
- "0b00000000000000" = 0 mA. // Set this value to use the associated SB_IO_OD instance at IR LED location.
- "0b00000000000001" = 50 mA for Full Mode; 25 mA for Half Mode
- "0b00000000111111" = 100 mA for Full Mode; 50 mA for Half Mode
- "0b00000001111111" = 150 mA for Full Mode; 75 mA for Half Mode
- "0b00000011111111" = 200 mA for Full Mode; 100 mA for Half Mode
- "0b00000111111111" = 250 mA for Full Mode; 125 mA for Half Mode
- "0b00001111111111" = 300 mA for Full Mode; 150 mA for Half Mode
- "0b00011111111111" = 350 mA for Full Mode; 175 mA for Half Mode
- "0b01111111111111" = 400 mA for Full Mode; 200 mA for Half Mode
- "0b11111111111111" = 450 mA for Full Mode; 225 mA for Half Mode
- "0b11111111111111" = 500 mA for Full Mode; 250 mA for Half Mode

LED Driver Current

For iCE40 Ultra, the LED Driver Current block is used to activate the mixed signal control block which supplies reference current to the RGB and IR Drivers. This block connects a stable 40 µA reference current for the LED drivers. In iCE40 UltraLite, the user no longer needs to connect this block since it has been connected by default.

Figure 25. LED Port Level

Table 10. LED Port List

<table>
<thead>
<tr>
<th>Name</th>
<th>I/O</th>
<th>Level</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>EN</td>
<td>I</td>
<td>Digital</td>
<td>Enable mixed signal block</td>
<td>Active HIGH</td>
</tr>
<tr>
<td>LEDPU</td>
<td>O</td>
<td>Analog</td>
<td>Output LED Driver power up signal</td>
<td>Analog</td>
</tr>
</tbody>
</table>

LED Signal Description

EN
Input, enables mixed signal control block to supply reference current to the LED drivers. When it is not enabled (EN=0), no current is supplied, and the LED drivers are powered down. Enabling the mixed signal control block takes 100 µs to reach a stable reference current value. SW models the output to be LOW during the 100 µs.

LEDPU
Output, LED Power Up signal. Connects to *PU signals of SB_RGB_DRV and SB_IR_DRV primitives
High Current/High Drive Output

iCE40LP and iCE40LM FPGAs feature three high current/high drive outputs that can source/sink up to 24 mA. These outputs provide significantly higher drive capability compared to normal IOs on the device and are ideal to drive three white LEDs or one RGB LED. These pins are labelled as HCIO in iCE40LP devices and HD on iCE40LM devices. These are not constant current drivers and require an external current limiting resistor when connecting to LEDs.

The HCIO on the iCE40LP are available on the LP640 and LP1K devices in the 16-WLCSP package only. Refer to the pinout file for the High Current and High Drive IO location on the iCE40LP and iCE40LM respectively.

To configure an IO with specific drive value, specify the DRIVE_STRENGTH synthesis attribute on the IO instance.

The Synthesis Attribute Syntax is:

/* synthesis DRIVE_STRENGTH = <Drive value> */

Table 11. Drive Value

<table>
<thead>
<tr>
<th>Drive Strength Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>x1</td>
<td>Default drive strength. No replication of SB_IO.</td>
</tr>
<tr>
<td>x2</td>
<td>Increase default drive strength by 2. SB_IO replicated once.</td>
</tr>
<tr>
<td>x3</td>
<td>Increase default drive strength by 3. SB_IO replicated twice.</td>
</tr>
</tbody>
</table>
Technical Support Assistance
Submit a technical support case through www.latticesemi.com/techsupport.

Revision History

<table>
<thead>
<tr>
<th>Date</th>
<th>Version</th>
<th>Change Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>June 2016</td>
<td>1.2</td>
<td>Added support for iCE40 UltraPlus.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Updated Introduction section.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>— Added iCE40 UltraPlus to the series of devices.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>— Added iCE40 UltraPlus data to Table 1, iCE40 Devices LED Driver Features Comparison.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>— Revised figure caption to Figure 5. System Diagram for Typical Application Using RGB, IR and Barcode Driver in iCE40 UltraLite and iCE40 UltraPlus.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Updated Embedded RGB PWM IP - iCE40 UltraLite and iCE40 UltraPlus section.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>— Revised section heading to include iCE40 UltraPlus.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>— Added iCE40 UltraPlus to key features lead in sentence.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>— Revised figure caption to Figure 6. iCE40 UltraLite and iCE40 UltraPlus RGB PWM IP Block Diagram.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>— Revised figure caption to Figure 7. iCE40 UltraLite and iCE40 UltraPlus RGB PWM IP Port Level Diagram.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>— Revised table caption to Table 2. iCE40 UltraLite and iCE40 UltraPlus RGB PWM Port List.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Updated iCE40 UltraLite and iCE40 UltraPlus RGB Driver section.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>— Revised section heading to include iCE40 UltraPlus.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>— Added iCE40 UltraPlus to introductory paragraph.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>— Added iCE40 UltraPlus to key features lead in sentence.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>— Revised figure caption to Figure 15. iCE40 UltraLite and iCE40 UltraPlus RGB Driver Block Diagram.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>— Revised figure caption to Figure 16. iCE40 UltraLite and iCE40 UltraPlus RGB Port Level Diagram.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>— Revised table caption to Table 5. iCE40 UltraLite and iCE40 UltraPlus RGB Port List.</td>
</tr>
<tr>
<td>October 2014</td>
<td>1.1</td>
<td>General revision.</td>
</tr>
<tr>
<td>June 2014</td>
<td>1.0</td>
<td>Initial release.</td>
</tr>
</tbody>
</table>
Appendix A. Instantiation Templates for Primitives

IR Driver in iCE40 Ultra

SB_IR_DRVIR_DRIVER(
  .IRLEDEN(ENABLE_IRLED),
  .IRPWM(IR_INPUT),
  .IRPU(led_power_up),
  .IRLED(IR_LED)
  ),
Defparam IR_DRIVER.IR_CURRENT = “1111111111”,

RGB Driver in iCE40 Ultra

SB_RGB_DRV RGB_DRIVER(
  .RGBLEDEN(ENABLE_LED),
  .RGB0PWM(RGB0),
  .RGB1PWM(RGB1),
  .RGB2PWM(RGB2),
  .RGBPU(led_power_up),
  .RGB0(LED0),
  .RGB1(LED1),
  .RGB2(LED2)
  ),
Defparam RGB_DRIVER.RGB0_CURRENT = “111111”,
Defparam RGB_DRIVER.RGB1_CURRENT = “111111”
Defparam RGB_DRIVER.RGB2_CURRENT = “111111”

LED Driver Current

LED_DRV_CUR LED_CUR_inst
  (  
    .EN(enable_led_current),
    .LEDPU(led_power_up)
  );

High Current/High Drive Output

module highdriveio (a, b, output_clk, c);
input a, b, output_clk;
output c;
assign x = a & b;
SB_IO #(.PIN_TYPE("010101")
  x_inst
  (.PACKAGE_PIN(c),
   .OUTPUT_CLK(output_clk),
   .D_OUT_0(x)
  ) /* synthesis DRIVE_STRENGTH= x2 */;
endmodule

IR400 Driver in iCE40 UltraLite

SB_IR400_DRV IR_DRIVER(
  .IRLEDEN(ENABLE_IRLED),
  .IRPWM(IR_INPUT),
  .CURREN(led_power_up),
  .IRLED(IR_LED)
  ),
Defparam IR_DRIVER. CURRENT_MODE = “0”,
Defparam IR_DRIVER.IR400_CURRENT = “11111111”
IR500 Driver in iCE40 UltraLite

SB_IR500_DRV IR_DRIVER ( 
    .IRLEDEN(ENABLE_IRLED),
    .IRPWM(IR_INPUT),
    .CURREN(led_power_up),
    .IRLED1(IR_LED1),
    .IRLED2(IR_LED2)
),
Defparam IR_DRIVER. CURRENT_MODE = "0",
Defparam IR_DRIVER.IR500_CURRENT = "1111111111"

BARCODE Driver

SB_BARCODE_DRV BARCODE_DRIVER ( 
    .BARCODEEN(ENABLE_BARCODE),
    .BARCODEPWM(BARCODE_INPUT),
    .CURREN(led_power_up),
    .BARCODE(BARCODE)
),
Defparam BARCODE_DRIVER. CURRENT_MODE = "0",
Defparam BARCODE_DRIVER.BARCODE_CURRENT = "111"

RGB Driver in iCE40 UltraLite and iCE40 UltraPlus

SB_RGBA_DRV RGB_DRIVER ( 
    .RGBLEDEN(ENABLE_LED),
    .RGB0PWM(RGB0),
    .RGB1PWM(RGB1),
    .RGB2PWM(RGB2),
    .CURREN(led_power_up),
    .RGB0(LED0),
    .RGB1(LED1),
    .RGB2(LED2)
),
Defparam RGB_DRIVER.CURRENT_MODE = "0",
Defparam RGB_DRIVER.RGB0_CURRENT = "111111"
Defparam RGB_DRIVER.RGB1_CURRENT = "111111"
Defparam RGB_DRIVER.RGB2_CURRENT = "111111"
Appendix B. Using RGB and IR Pins as User IO

To use the RGB and IRLED pins as general io user must instantiate the SB_IO_OD primitive, see example below:

```verilog
module top(a, o1);
  input a;
  output o1;

  wire o1i;

  assign o1i = a;

  SB_IO_OD OpenDrainInst0
  (#
    .PACKAGEPIN (o1), // User's Pin signal name
    .LATCHINPUTVALUE (), // Latches/holds the Input value
    .CLOCKENABLE (), // Clock Enable common to input and output clock
    .INPUTCLK (), // Clock for the input registers
    .OUTPUTCLK (), // Clock for the output registers
    .OUTPUTENABLE (), // Output Pin Tristate/Enable control
    .DOUT0 (o1i), // Data 0 - out to Pin/Rising clk edge
                  // Data 1 - out to Pin/Falling clk edge
    .DIN0 (), // Data 0 - Pin input/Rising clk edge
    .DIN1 () // Data 1 - Pin input/Falling clk edge
  );

  defparam OpenDrainInst0.PIN_TYPE = 6'b011001;
  defparam OpenDrainInst0.NEG_TRIGGER = 1'b0;

endmodule
```
Appendix C. LED Connection Diagrams

Figure 26. iCE40 UltraLite and iCE40 UltraPlus Circuit Diagram

Note:
The LED driver for iCE40 devices are designed for the supply of 3.8 V to 4.3 V.
The recommended voltage range for iCE40 device outputs is shown in the diagram.
The LED leakage current, forward and reverse voltage drop is different depending upon the
LED manufacturer. Designer can use external components such as diodes, resistors or isolation
FETs along with the LED to meet the recommended voltage range on the outputs for the iCE40 device.
Figure 27. iCE40 Ultra Circuit Diagram

Note:
The LED driver for iCE40 devices are designed for the supply of 3.8 V to 4.3 V.
The recommended voltage range for iCE40 device outputs is shown in the diagram.
The LED leakage current, forward and reverse voltage drop is different depending upon the
LED manufacturer. Designer can use external components such as diodes, resistors or isolation
FETs along with the LED to meet the recommended voltage range on the outputs for the iCE40 device.
Figure 28. iCE40LP and iCE40LM Circuit Diagram (HCIO/HD Output Sinking)

Red/Green/Blue: $I_F = \frac{(V_{DD} - V_F)}{(R + Ron/N)}$

$R_{on} = 38\Omega - 50\Omega$

$N = \text{Drive Value}$

$V_{LEDOUT} = V_{DD} - V_F$

$0.5\,V \leq V_{LEDOUT} \leq 3.6\,V_{(abs.\,max)}$

Note:
The LED driver for iCE40 devices are designed for the supply of 3.8 V to 4.3 V.
The recommended voltage range for iCE40 device outputs is shown in the diagram.
The LED leakage current, forward and reverse voltage drop is different depending upon the
LED manufacturer. Designer can use external components such as diodes, resistors or isolation
FETs along with the LED to meet the recommended voltage range on the outputs for the iCE40 device.
Appendix D. RGB PWM IP - LED Control Bus Addressable Registers

<table>
<thead>
<tr>
<th>LEDD_ADR[3:0]</th>
<th>Name</th>
<th>Usage</th>
<th>Access</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000</td>
<td>LEDDCR0</td>
<td>LED Driver Control Register 0</td>
<td>W</td>
</tr>
<tr>
<td>1001</td>
<td>LEDDBR</td>
<td>LED Driver Pre-scale Register</td>
<td>W</td>
</tr>
<tr>
<td>1010</td>
<td>LEDDONR</td>
<td>LED Driver ON Time Register</td>
<td>W</td>
</tr>
<tr>
<td>1011</td>
<td>LEDDOFR</td>
<td>LED Driver OFF Time Register</td>
<td>W</td>
</tr>
<tr>
<td>0101</td>
<td>LEDDBCRR</td>
<td>LED Driver Breathe On Control Register</td>
<td>W</td>
</tr>
<tr>
<td>0110</td>
<td>LEDDBCFR</td>
<td>LED Driver Breathe Off Control Register</td>
<td>W</td>
</tr>
<tr>
<td>0001</td>
<td>LEDDPWRR</td>
<td>LED Driver Pulse Width Register for RED</td>
<td>W</td>
</tr>
<tr>
<td>0010</td>
<td>LEDDPWRR</td>
<td>LED Driver Pulse Width Register for GREEN</td>
<td>W</td>
</tr>
<tr>
<td>0011</td>
<td>LEDDPWRR</td>
<td>LED Driver Pulse Width Register for BLUE</td>
<td>W</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit7</th>
<th>Bit6</th>
<th>Bit5</th>
<th>Bit4</th>
<th>Bit3</th>
<th>Bit2</th>
<th>Bit1</th>
<th>Bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>LEDDEN</td>
<td>FR250</td>
<td>OUTPOL</td>
<td>OUTSKEW</td>
<td>QUICK STOP</td>
<td>PWM MODE</td>
<td>BRMSBEXT</td>
<td></td>
</tr>
</tbody>
</table>

Table 12. LEDDCR0 Field Description

<table>
<thead>
<tr>
<th>Bit</th>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>LEDDEN</td>
<td><strong>LED Driver Enable Bit</strong> — This bit enables the LED Driver. If LEDDEN is cleared, the LED Driver is disabled and the system clock into the LED Driver block will be gated off. 0 = LED Driver disabled 1 = LED Driver enabled</td>
</tr>
<tr>
<td>6</td>
<td>FR250</td>
<td><strong>Flick Rate Select Bit</strong> — This bit selects the flick rate for the PWM logic between 125 Hz and 250 Hz. 0 = 125 Hz 1 = 250 Hz</td>
</tr>
<tr>
<td>5</td>
<td>OUTPOL</td>
<td><strong>PWM Outputs Polarity Select Bit</strong> — This bit selects the PWM outputs polarity. 0 = Active High 1 = Active Low</td>
</tr>
<tr>
<td>4</td>
<td>OUTSKEW</td>
<td><strong>PWM Output Skew Enable Bit</strong> — This bit enables the PWM slew to reduce simultaneous switching noise, based on BRMSBEXT [1:0] 0 = Disable Output Skew 1 = Enable Output Skew</td>
</tr>
<tr>
<td>3</td>
<td>QUICK STOP</td>
<td><strong>Blinking Sequence Quick Stop Enable Bit</strong> — This bit enables the quick stop when LEDD_EXE going low, instead of waiting current ON period finished when breathe on is enabled. 0 = Stop the blinking sequence when current ON period finished when LEDD_EXE goes low. 1 = Immediately terminate the blinking sequence after LEDD_EXE goes low. (within 5 ledd_clk cycles)</td>
</tr>
</tbody>
</table>

For LED Control registers access timing, please refer to (section reference here).

**LED Driver Control Register 0 (LEDDCR0)**

LEDDCR0 can be written through LED Control Bus.

For LED Control registers access timing, please refer to (section reference here).
The blink ON time could be set from 0 to 8.16 seconds, with 0.032 seconds incremental step. The actual blink ON time could be calculated by using the formula below. Also all available blink ON time options are shown in the table following the formula.

\[
\text{Blink ON Time} = 0.032 \times \text{NON} \quad \text{(Sec)}
\]

### LED Driver ON Time Register (LEDDONR)

LEDDONR can be written through LED Control Bus.

<table>
<thead>
<tr>
<th>Bit7</th>
<th>Bit6</th>
<th>Bit5</th>
<th>Bit4</th>
<th>Bit3</th>
<th>Bit2</th>
<th>Bit1</th>
<th>Bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>LED Blink ON Time Setup (NON)</td>
</tr>
</tbody>
</table>

The Blink ON time could be calculated by using the formula below. Also all available blink ON time options are shown in the table following the formula.

\[
\text{Blink ON Time} = 0.032 \times \text{NON} \quad \text{(Sec)}
\]
LED Driver OFF Time Register (LEDDOFR)

LEDDOFR can be written through LED Control Bus.

<table>
<thead>
<tr>
<th>Bit7</th>
<th>Bit6</th>
<th>Bit5</th>
<th>Bit4</th>
<th>Bit3</th>
<th>Bit2</th>
<th>Bit1</th>
<th>Bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>LED Blink OFF Time Setup (NOFF)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The blink OFF time could be set from 0 to 8.16 seconds, with 0.032 seconds incremental step. The actual blink OFF time could be calculated by using the formula below. Also all available blink OFF time options are shown in the table following the formula.

\[
\text{Blink OFF Time} = 0.032 \times \text{NOFF (Sec)}
\]

LED Driver Breathe ON Control Register (LEDDBCRR)

LEDDBCRR can only be written through the LED Control Bus.

<table>
<thead>
<tr>
<th>Bit7</th>
<th>Bit6</th>
<th>Bit5</th>
<th>Bit4</th>
<th>Bit3</th>
<th>Bit2</th>
<th>Bit1</th>
<th>Bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Breathe ON</td>
<td>Breathe Edge</td>
<td>Breathe Mode</td>
<td>RSVP</td>
<td></td>
<td></td>
<td></td>
<td>Breathe ON Rate</td>
</tr>
</tbody>
</table>
Table 13. LEDDBCRR Field Description

<table>
<thead>
<tr>
<th>Bit</th>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>Breathe ON Enable</td>
<td>Breathe ON Enable Bit — This bit enables the breathe ON feature setup in bit[5] and bit [3:0]. 0 = The Breathe control if disabled; NO Breathe ON 1 = The breathe control is enabled</td>
</tr>
<tr>
<td>6</td>
<td>Breathe Edge</td>
<td>Breathe Edge Selection Bit — This bit enables the breathe ON control present in this byte be applied for both breathe ON and OFF 0 = The breathe control in this byte only be applied for ON ramp. 1 = The Breathe control in this byte will be applied for both ON and OFF ramp.</td>
</tr>
<tr>
<td>5</td>
<td>Breathe Mode</td>
<td>Breathe Mode Select Bit — This bit selects the breathe ON/OFF mode. If this bit is cleared, the LED Driver with breathe ON/OFF] with fix rate set in bit [3:0] for all colors; If this bit is set, the LED Driver will breathe ON/OFF] using modulated rate based on the its destination brightness level. 0 = Unique rate for breathe ON/OFF] 1 = Modulate rate for breathe ON/OFF], based on the destination brightness level.</td>
</tr>
<tr>
<td>4</td>
<td>RSVD</td>
<td>-</td>
</tr>
<tr>
<td>3:0</td>
<td>Breathe ON Rate</td>
<td>User setup of the breathe ON/OFF rate. 4'b0000 = No Breathe ON/OFF]</td>
</tr>
</tbody>
</table>

The optional breathe ON/OFF range is show in Table 14.

Table 14. Optional Breath ON/OFF Range

<table>
<thead>
<tr>
<th>UI</th>
<th>0000</th>
<th>0001</th>
<th>0010</th>
<th>0011</th>
<th>0100</th>
<th>0101</th>
<th>0110</th>
<th>0111</th>
<th>1000</th>
<th>1001</th>
<th>1010</th>
<th>1011</th>
<th>1100</th>
<th>1101</th>
<th>1110</th>
<th>1111</th>
</tr>
</thead>
<tbody>
<tr>
<td>T_{ramp} (sec)</td>
<td>0.128</td>
<td>0.256</td>
<td>0.384</td>
<td>0.512</td>
<td>0.640</td>
<td>0.768</td>
<td>0.896</td>
<td>1.024</td>
<td>1.152</td>
<td>1.280</td>
<td>1.408</td>
<td>1.536</td>
<td>1.664</td>
<td>1.792</td>
<td>1.920</td>
<td>2.048</td>
</tr>
</tbody>
</table>

1. UI is the user input value in binary.

The modulated ramp rate could be achieved by 16 bit counter which will increase on every flick rate cycle (125Hz) with the step size internally calculated based on the formula below:

\[
N_{step} = \frac{256 \times \frac{\text{Brightness}}{16}}{UI + 1}
\]

During the ramp up, the PWM engine will take the MSB 8 bits of the 16 bit counter as input. This will result the ramp size of \(\frac{256 \times \frac{\text{Brightness}}{16}}{256}\) (of 255) increment per flick rate cycle (125 Hz). The effective breathe-on ramp rates are shown in the figure below.
LED Driver Breathe OFF Control Register (LEDDBCFR)

LEDDBCFR can only be written through the LED Control Bus.

<table>
<thead>
<tr>
<th>Bit7</th>
<th>Bit6</th>
<th>Bit5</th>
<th>Bit4</th>
<th>Bit3</th>
<th>Bit2</th>
<th>Bit1</th>
<th>Bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Breathe OFF Enable</td>
<td>PWM Range Extend</td>
<td>Breathe Mode</td>
<td>RSVD</td>
<td>Breathe OFF Rate</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 15. LEDDBCFR Field Description

<table>
<thead>
<tr>
<th>Bit</th>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>Breathe OFF Enable</td>
<td><strong>Breathe OFF Enable Bit</strong> — This bit enables the breathe OFF feature setup in bit [5] and bit [3:0]. This bit will be overridden by LEDDBCRR [7] if the LEDDBCRR [6] is set. 0 = The Breathe OFF control if disabled; NO Breathe OFF 1 = The breathe OFF control is enabled</td>
</tr>
<tr>
<td>6</td>
<td>PWM Range Extend</td>
<td><strong>PWM Range Extend</strong> — This bit extend the original 255/256 PWM pulse width for the linear counter mode to 256/256 to provide constant on PWM output for testing. 0 = PWM extension OFF 1 = Extend the PWM pulse with from 255/256 to 256/256 for Linear Counter Mode.</td>
</tr>
<tr>
<td>5</td>
<td>Breathe Mode</td>
<td><strong>Breathe Mode Select Bit</strong> — This bit selects the breathe ON/OFF mode. If this bit is cleared, the LED Driver with breathe ON/OFF with fix rate set in bit [3:0] for all colors; If this bit is set, the LED Driver will breathe ON/OFF using modulated rate based on the its destination brightness level. This bit will be overridden by LEDDBCRR [5] if the LEDDBCRR [6] is set. 0 = Unique rate for breathe OFF 1 = Modulate rate for breathe OFF, based on the destination brightness level.</td>
</tr>
<tr>
<td>4</td>
<td>RSVD</td>
<td>-</td>
</tr>
<tr>
<td>3:0</td>
<td>Breathe OFF Rate</td>
<td>User setup of the breathe OFF rate. 4'b0000 = No Breathe OFF</td>
</tr>
</tbody>
</table>

The optional breathe OFF range is show in Table 16.

Table 16. Optional Breath OFF Range

<table>
<thead>
<tr>
<th>UI</th>
<th>0000</th>
<th>0001</th>
<th>0010</th>
<th>0011</th>
<th>0100</th>
<th>0101</th>
<th>0110</th>
<th>0111</th>
<th>1000</th>
<th>1001</th>
<th>1010</th>
<th>1011</th>
<th>1100</th>
<th>1101</th>
<th>1110</th>
<th>1111</th>
</tr>
</thead>
<tbody>
<tr>
<td>U1</td>
<td>0.128</td>
<td>0.256</td>
<td>0.384</td>
<td>0.512</td>
<td>0.640</td>
<td>0.768</td>
<td>0.896</td>
<td>1.024</td>
<td>1.152</td>
<td>1.280</td>
<td>1.408</td>
<td>1.536</td>
<td>1.664</td>
<td>1.792</td>
<td>1.920</td>
<td>2.048</td>
</tr>
</tbody>
</table>

1. UI is the user input value in binary.

Opposite to the ramp on period, the modulated ramp rate could be achieved by 16 bit counter which will decrease on every flick rate cycle (125 Hz) with the step size internally calculated based on the formula below:

\[ N_{step} = \frac{256 \times \text{Brightness}}{UI + 1} \times \frac{16}{256} \]

During the ramp up, the PWM engine will take the MSB 8 bits of the 16 bit counter as input. This will result the ramp size of \( \frac{256 \times \text{Brightness}}{UI + 1} \times \frac{16}{256} \) (of 255) increment per flick rate cycle (125 Hz).

LED Driver RED Pulse Width Register (LEDDPWRR)

LEDDPWRR can only be written through System Bus

<table>
<thead>
<tr>
<th>Bit7</th>
<th>Bit6</th>
<th>Bit5</th>
<th>Bit4</th>
<th>Bit3</th>
<th>Bit2</th>
<th>Bit1</th>
<th>Bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>RED Pulse Width (PW_R)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
The LEDDPWRR allow user to setup the brightness of the RED LED through Pulse Width Modulation (PWM) with total 256 brightness level. Based on the PWR value, the modulated pulse with could be generated from 0 to 100% of the flick rate cycle in \( \frac{1}{256} \) % per step. The Active Duty Cycle could be calculated as:

\[
\text{ADC} (%) = \frac{PWR}{256}
\]

In Linear Counter Mode (Non-LSFSR Mode), if the LEDDBCFR[6] bit is set, with PWR = 8HFF setting, the Active Duty Cycle of the PWM output will be 100% instead of 255/256%. This way we could provide constant on PWM output for characterization and validation testing.

**LED Driver GREEN Pulse Width Register (LEDDPWRG)**

LEDDPWRG can only be written through System Bus

<table>
<thead>
<tr>
<th>Bit7</th>
<th>Bit6</th>
<th>Bit5</th>
<th>Bit4</th>
<th>Bit3</th>
<th>Bit2</th>
<th>Bit1</th>
<th>Bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>GREEN Pulse Width (PWG)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The LEDDPWRG allow user to setup the brightness of the GREEN LED through Pulse Width Modulation (PWM) with total 256 brightness level. Based on the PWG value, the modulated pulse with could be generated from 0 to 100% of the flick rate cycle in \( \frac{1}{256} \) % per step. The Active Duty Cycle could be calculated as:

\[
\text{ADC} (%) = \frac{PWG}{256}
\]

In Linear Counter Mode (Non-LSFSR Mode), if the LEDDBCFR[6] bit is set, with PWR = 8HFF setting, the Active Duty Cycle of the PWM output will be 100% instead of 255/256%. This way we could provide constant on PWM output for characterization and validation testing.

**LED Driver BLUE Pulse Width Register (LEDDPWRB)**

LEDDPWRB can only be written through System Bus

<table>
<thead>
<tr>
<th>Bit7</th>
<th>Bit6</th>
<th>Bit5</th>
<th>Bit4</th>
<th>Bit3</th>
<th>Bit2</th>
<th>Bit1</th>
<th>Bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLUE Pulse Width (PWB)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The LEDDPWRB allow user to setup the brightness of the BLUE LED through Pulse Width Modulation (PWM) with total 256 brightness level. Based on the PWB value, the modulated pulse with could be generated from 0 to 100% of the flick rate cycle in \( \frac{1}{256} \) % per step. The Active Duty Cycle could be calculated as:

\[
\text{ADC} (%) = \frac{PWB}{256}
\]

In Linear Counter Mode (Non-LSFSR Mode), if the LEDDBCFR[6] bit is set, with PWR = 8HFF setting, the Active Duty Cycle of the PWM output will be 100% instead of 255/256%. This way we could provide constant on PWM output for characterization and validation testing.
LEDD Control Register Waveform Shaping

Waveform When BREATHE_MODE = 0

The LEDD Control Register waveform shaping, when BREATHE_MODE (LEDDBCRR/LEDDBCFR Bit [5]) is “0”, is demonstrated in the Figure below.

Waveform When BREATHE_MODE = 1

The LEDD Control Register waveform shaping, when BREATHE_MODE (LEDDBCRR/LEDDBCFR Bit [5]) is “1”, is demonstrated in the Figure below.
Appendix E. IR Transceiver IP

IRTCV Control Bus Addressable Registers

The IRTCV Control Bus addressable registers are shown in Table 17.

### Table 17. IRTCV Control Bus Addressable Register

<table>
<thead>
<tr>
<th>IRTCV_ADR[3:0]</th>
<th>Name</th>
<th>Usage</th>
<th>Access</th>
</tr>
</thead>
<tbody>
<tr>
<td>0001</td>
<td>IRTCVCRR</td>
<td>IR Transceiver Control Register</td>
<td>W</td>
</tr>
<tr>
<td>0010</td>
<td>IRSYSFR3</td>
<td>IR Transceiver System Clock Frequency Register 3</td>
<td>W</td>
</tr>
<tr>
<td>0011</td>
<td>IRSYSFR2</td>
<td>IR Transceiver System Clock Frequency Register 2</td>
<td>W</td>
</tr>
<tr>
<td>0100</td>
<td>IRSYSFR1</td>
<td>IR Transceiver System Clock Frequency Register 1</td>
<td>W</td>
</tr>
<tr>
<td>0101</td>
<td>IRSYSFR0</td>
<td>IR Transceiver System Clock Frequency Register 0</td>
<td>W</td>
</tr>
<tr>
<td>0110</td>
<td>IRTCVFR2</td>
<td>IR Transceiver Clock Frequency Register 2</td>
<td>R/W</td>
</tr>
<tr>
<td>0111</td>
<td>IRTCVFR1</td>
<td>IR Transceiver Clock Frequency Register 1</td>
<td>R/W</td>
</tr>
<tr>
<td>1000</td>
<td>IRTCVFR0</td>
<td>IR Transceiver Clock Frequency Register 0</td>
<td>R/W</td>
</tr>
<tr>
<td>1001</td>
<td>IRTCVDRI</td>
<td>IR Transceiver Data 1</td>
<td>R/W</td>
</tr>
<tr>
<td>1010</td>
<td>IRTCVDRO</td>
<td>IR Transceiver Data 0</td>
<td>R/W</td>
</tr>
<tr>
<td>1011</td>
<td>IRTCVSR</td>
<td>IR Transceiver Status Register</td>
<td>R</td>
</tr>
</tbody>
</table>

IR Transceiver Control Register (IRTCVCRR)

IRTCVCRR can be written through IR Transceiver Control Bus.

<table>
<thead>
<tr>
<th>Bit7</th>
<th>Bit6</th>
<th>Bit5</th>
<th>Bit4</th>
<th>Bit3</th>
<th>Bit2</th>
<th>Bit1</th>
<th>Bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>IRTCVEN</td>
<td>DUTY33</td>
<td>OUTPOL</td>
<td>DISOE</td>
<td>USRMAX</td>
<td>REMEASEN</td>
<td>INFILTSEL</td>
<td></td>
</tr>
</tbody>
</table>

### Table 18. IRTCVCRR Field Description

<table>
<thead>
<tr>
<th>Bit</th>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>IRTCVEN</td>
<td>IR Transceiver Enable Bit — This bit enables the IR Transceiver IP. If IRTCVEN is cleared, the IR Transceiver is disabled. 0 = IR Transceiver disabled. 1 = IR Transceiver enabled</td>
</tr>
<tr>
<td>6</td>
<td>DUTY33</td>
<td>ON Pulse Duty Cycle Select Bit — This bit selects the IR ON pulse duty cycle between 1/2 and 1/3 of the Transmit clock period. 0 = 1/2 TFTCV. 1 = 1/3 TFTCV</td>
</tr>
<tr>
<td>5</td>
<td>OUTPOL</td>
<td>PWM Outputs Polarity Select Bit — This bit selects the PWM outputs polarity. 0 = Active High. 1 = Active Low</td>
</tr>
<tr>
<td>4</td>
<td>DISOE</td>
<td>Disable Output On Error Bit — This bit disable the IR Transceiver transmit output upon Error. 0 = Continue On Error. 1 = Disable IR_OUT On Error</td>
</tr>
<tr>
<td>3</td>
<td>USRMAX</td>
<td>User defined the Maximum pulse count in learning mode — This bit enable capability to allow user specify the Maximum pulse count through IRTCVDRI in learning. 0 = Maximum count is 15H7FFF. 1 = User specify the Maximum count through IRTCVDRI.</td>
</tr>
<tr>
<td>2</td>
<td>REMEASEN</td>
<td>Learning TCV Clock Frequency Re-Measure Enable Bit — This bit enables the TCV clock frequency measurement on the beginning of every Active ON cycle group. 0 = Re-measuring Disabled, TCV frequency is evaluated only on the very first Active ON Cycle group. 1 = Re-measuring Enabled,</td>
</tr>
</tbody>
</table>
IR Transceiver Status Register (IRTCVSR)

IRTCVSR can be read through the IR Transceiver Control Bus. It will report the IR Transceiver status with the bit definition shown below.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>BUSY</td>
<td><strong>IR Transceiver BUSY</strong> — This bit indicates the IR Transceiver IP is busy, transmitting, receiving or calculating. 0 = Idle 1 = Busy</td>
</tr>
<tr>
<td>6</td>
<td>TIP</td>
<td><strong>Transceiver In Progress</strong> — This bit indicates the IR Transceiver is in the middle of transmitting or receiving. 0 = Not Transmitting or Receiving 1 = Transmitting or receiving</td>
</tr>
<tr>
<td>5</td>
<td>RSVD</td>
<td>-</td>
</tr>
<tr>
<td>4</td>
<td>LFCNTOFL</td>
<td><strong>Receiving Frequency Counter Overflow Flag</strong> — This bit indicates the IR Receiver Frequency evaluation Counter overflow when measuring TCV clock frequency. 0 = No Frequency Counter Overflow 1 = Frequency Counter Overflow</td>
</tr>
<tr>
<td>3</td>
<td>LDATOFL</td>
<td><strong>Receiving Counter Overflow Flag</strong> — This bit indicates the IR Receiver Counter overflow when detecting the ON/OFF pulses/cycles in learning mode. 0 = No Data Counter Overflow 1 = Data Counter Overflow</td>
</tr>
<tr>
<td>2</td>
<td>DATERR</td>
<td><strong>Data Error Flag</strong> — This bit indicates the IR Transceiver Data Error caused by data buffer under-run in transmit mode or data buffer over-run in learning mode. 0 = No Data Error 1 = Data Error Occurred</td>
</tr>
<tr>
<td>1</td>
<td>RFRDY</td>
<td><strong>Receiving Clock Frequency Ready</strong> — These bit indicates the receiving frequency have been detected and calculated in learning mode. 0 = Receiving Frequency Value is not valid 1 = Receiving Frequency Value is valid</td>
</tr>
<tr>
<td>0</td>
<td>DBUFRDY</td>
<td><strong>Data Buffer Ready Flag</strong> — These bit indicates the Transmit buffer is empty in transmit mode or the receiving buffer is full in the learning mode. IRTCVDR0 write/read activity triggered... 0 = Data Buffer is NOT Ready 1 = Data Buffer is Ready</td>
</tr>
</tbody>
</table>

The BUSY status flag active (High) causes the BUSY hand shaking signal to go high to inform the host in the FPGA fabric that the IR Transceiver IP is currently busy transmitting, receiving or performing internal parameter calculation. The IRTCV_BUSY pin is logically equivalent to the BUSY status flag.

The DBUFRDY status flag is used in both transmitting mode and learning mode. In transmitting mode, it indicates that the IRTCVDR buffer is ready for writing by the host in FPGA fabric. The DBUFRDY flag is set when the IRTCVDR data is fetched into the transmitter by the IRTCV IP logic. Writing data into the IRTCVDR0 clears the DBU-
FRDY flag. Delay in writing data into the IRTCVDR (0) before current data transmitting is finished causes the DATERR flag to become high, to indicate that the IRTCVDR is under-run and that a transmitting sequence error occurred. Once the DATERR flag is set, it remains high until the IRTCVCR is re-written, or a new IRTC event occurs (EXE rising). In learning mode, the DBUFRDY indicates that the IRTCVDR buffer is ready to read by the host. Reading the data from IRTCVDR0 clears the FBUFFRDY flag. Delay in reading data from the IRTCVDR (0) before next data detected and evaluated causes the DATERR flag to be set, and remains set until the IRTCVCR is re-written, or a new IRTC event occurs. The DRDY pin is logically equivalent to the DBUFRDY status flag. The ERR pin, on the other hand, is logically equivalent to the DATERR status flag.

The RFRDY status flag indicates that the receiving IR signal frequency is detected, calculated and ready for read from IRTCVFR (f2:0) in learning mode. Reading the IRTCVFR (0) clears the DBUFRDY status flag. Writing to the IRTCVCR or a new IRTC event also clears the DBUFRDY status flag.

The LDATOFL status flag indicates that the counter, used to detect the ON pulses or OFF cycles, reached its maximum count in learning mode. It remains set until IRTCVCR is re-written or a new learning event is started.

The LFCNTOFL status flag indicates that the counter, used to detect the IR_IN clock period, exceeds the (16 bits) range. Once it occurs, the previously detected IRIN rising edge is abandoned, and the clock period detection is started over. The LFCNTOFL status flag is cleared when the IRTCVCR is re-written or a new learning event is started.

**IR Transceiver System Clock Frequency Registers (IRSYSFR 0-3)**

IRSYSFRs can be written through LED Control Bus.

<table>
<thead>
<tr>
<th>IRSYSFR3 [3:0]</th>
<th>IRSYSFR2 [7:0]</th>
<th>IRSYSFR1 [7:0]</th>
<th>IRSYSFR0 [7:0]</th>
</tr>
</thead>
<tbody>
<tr>
<td>System Clock Frequency FSYS (Hz)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The IR System Clock Frequency Registers, total 28 bits, holding the binary number representing the system clock frequency in Hertz. For normal application, the user should set the IRSYSFRs prior to setting the IRTCVFRs. When writing IRSYSFR3, the MSB four bits from the data bus are “don’t care”.

**IR Transceiver Clock Frequency Registers (IRTCVFR 0-2)**

IRTCVFRs can be written or read through the IR Transceiver Control Bus.

<table>
<thead>
<tr>
<th>IRTCVR2 [7:0]</th>
<th>IRTCVR1 [7:0]</th>
<th>IRTCVR0 [7:0]</th>
</tr>
</thead>
<tbody>
<tr>
<td>IR Transceiver Clock Frequency FTCV (Hz)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The IR Transceiver Clock Frequency Register, totals 24 bits, and holds the binary number representing the IR Transceiver clock frequency in Hertz. In transmit mode, you should write these four bytes sequentially from IRTCVR2 to IRTCVR0. In learning mode, the detected transceiver frequency is available from this 24-bit register once the DRDY signal is set. The transceiver frequency is evaluated at the beginning of every ON period.

Internal clock count to generate the IR Transceiver Clock is computed using the formula below.

\[ N = \frac{F_{SYS}}{F_{TCV}} \]
IR Transceiver Data Registers (IRTCVDR 0-1)

IRTCVDR0-1 can be written or read through the IR Transceiver Control Bus.

<table>
<thead>
<tr>
<th>BIT 15</th>
<th>BIT [14:0]</th>
</tr>
</thead>
<tbody>
<tr>
<td>IRTCVR1 [7]</td>
<td>IRTCVR1 [6:0]</td>
</tr>
<tr>
<td>IRTCVR0 [7:0]</td>
<td>IR_FLAG</td>
</tr>
</tbody>
</table>

Table 20. IRTCVR 0-1 Field Description

<table>
<thead>
<tr>
<th>Bit</th>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>IR_FLAG</td>
<td>IR FLAG Bit — This bit indicates the data represented in BIT[14:0] should be number of ON cycles or OFF cycles. 0 = Bit [14:0] is number of OFF cycles to transmit. 1 = Bit [14:0] is number of ON cycles to transmit.</td>
</tr>
<tr>
<td>14:0</td>
<td>N_CYCLES</td>
<td>Number of ON/OFF Cycles</td>
</tr>
</tbody>
</table>

During a typical transmit session, the host logic inside the FPGA fabric should monitor the DRDY flag. Once the DRDY flag is high, then the host should write next ON-OFF cycle count, with IR_FLAG at MSB, into the IRTCVDR0-1. The host logic has minimal 30 system clock (CLKI) cycles (worst case when $F_{SYS_{-}CLK} = 4$ MHz, $F_{TCV} = 120$ kHz and IRTCVR = 1) to complete written data into the IRTCVDR0-1 (2 bytes). Failure to do so causes the transmit error to occur and the ERR flag to become high. The transmit session should be then terminated by the host. The N_CYCLE for ON-OFF should be non-zero number. If N_CYCLE = 0 accidentally happens, the decimal 1 is assumed by the hardware.

During a typical learning session, the host logic inside the FPGA fabric should monitor the DRDY flag. Once the DRDY flag is high, both the IRTCVFR (4 bytes) and the IRTCVR (2 bytes) are ready for read. The host logic has minimal 30 system clock (CLKI) cycles (worst case when $F_{SYS_{-}CLK} = 4$ MHz, $F_{TCV} = 120$ kHz and IRTCVR = 1) to fetch the IRTCVR (2 bytes). Failure to do so causes the learning error to occur and the ERR flag becomes high. The learning session should then be terminated by the host. The host should continuously examine the received OFF cycles count to determine the end of each frame (Large than 4096). The maximum count is 15'H7FFF (32,767). The Learning mode could not co-exist with the transmit mode.

When accessing IRTCVRs, IRTCVR1 should be written/read first for two bytes data access, or the host could only write/read the IRTCVR0 if the IRTCVR1 is un-changed (or all Zero).
IR Transceiver Waveform

Transmitting Waveform

The typical IR Transmitting waveform (LEARN = 0) is demonstrated in Figure 29.

Figure 29. Typical IR Transmitting Waveform

TtcvMinimal 30 CLKI (Fabric FSM Clock) cycles at worst case (ON-OFF Count = 1, FIRC\_CLK = 4 MHz, FTCV = 120 KHz).
Learning Waveform
The typical IR Learning waveform (LEARN = 1) is demonstrated in Figure 30.

Figure 30. Typical IR Learning Waveform

Minimal 30 CLK (Fabric FSM Clock) cycles at worst case (ON-OFF Count > 1, F[IRCV_CLK] = 4MHz, F[TCV] = 120 KHz).
Carrier Frequency Register (IRTCVFR) is ready for read after first valid DRDY flag.