Microsemi makes no warranty, representation, or guarantee regarding the information contained herein or the suitability of its products and services for any particular purpose, nor does Microsemi assume any liability whatsoever arising out of the application or use of any product or circuit. The products sold hereunder and any other products sold by Microsemi have been subject to limited testing and should not be used in conjunction with mission-critical equipment or applications. Any performance specifications are believed to be reliable but are not verified, and Buyer must conduct and complete all performance and other testing of the products, alone and together with, or installed in, any end-products. Buyer shall not rely on any data and performance specifications or parameters provided by Microsemi. It is the Buyer’s responsibility to independently determine suitability of any products and to test and verify the same. The information provided by Microsemi hereunder is provided “as is, where is” and with all faults, and the entire risk associated with such information is entirely with the Buyer. Microsemi does not grant, explicitly or implicitly, to any party any patent rights, licenses, or any other IP rights, whether with regard to such information itself or anything described by such information. Information provided in this document is proprietary to Microsemi, and Microsemi reserves the right to make any changes to the information in this document or to any products and services at any time without notice.

About Microsemi

Microsemi Corporation (Nasdaq: MSCC) offers a comprehensive portfolio of semiconductor and system solutions for aerospace & defense, communications, data center and industrial markets. Products include high-performance and radiation-hardened analog mixed-signal integrated circuits, FPGAs, SoCs and ASICs; power management products; timing and synchronization devices and precise time solutions, setting the world’s standard for time; voice processing devices; RF solutions; discrete components; enterprise storage and communication solutions, security technologies and scalable anti-tamper products; Ethernet solutions; Power-over-Ethernet ICs and midspans; as well as custom design capabilities and services. Microsemi is headquartered in Aliso Viejo, Calif., and has approximately 4,800 employees globally. Learn more at www.microsemi.com.
Contents

1 Revision History ......................................................................................................................... 1
  1.1 Revision 2.1 ........................................................................................................................... 1
  1.2 Revision 2.0 ........................................................................................................................... 1
  1.3 Revision 1.1 ........................................................................................................................... 1
  1.4 Revision 1.0 ........................................................................................................................... 1

2 Overview .................................................................................................................................... 2
  2.1 Key Features .......................................................................................................................... 2
  2.2 Supported Families ................................................................................................................. 2
  2.3 Theory of Operation .............................................................................................................. 2

3 Hardware Implementation ......................................................................................................... 3
  3.1 Inputs and Outputs .................................................................................................................. 4
  3.2 Configuration Parameters ...................................................................................................... 4
  3.3 Timing Diagram ...................................................................................................................... 5
  3.4 Resource Utilization ............................................................................................................... 5
Figures

Figure 1  Signal Generation in Resolver ................................................................. 2
Figure 2  System-Level Block Diagram of Hardware Implementation ......................... 3
Figure 3  Timing Diagram ..................................................................................... 5
### Tables

<table>
<thead>
<tr>
<th>Table 1</th>
<th>Inputs and Outputs of Resolver Interface Block</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table 2</td>
<td>Configuration Parameters</td>
<td>4</td>
</tr>
<tr>
<td>Table 3</td>
<td>Resource Utilization of Resolver Interface Block</td>
<td>5</td>
</tr>
</tbody>
</table>
1 Revision History

The revision history describes the changes that were implemented in the document. The changes are listed by revision, starting with the current publication.

1.1 Revision 2.1

The following is a summary of the changes made in revision 2.0 of this document

- Figure 1, page 3 is updated to add the input port: direction_config_i.
- Table 1, page 4 is updated to add new signal name: direction_config_i and its description. Also, updated description for the signal name: pp_ratio_i.

1.2 Revision 2.0

The following is a summary of the changes made in revision 2.0 of this document

- Figure 1, page 3 is updated to add the input ports, clear_buffer_i and pp_ratio_i.
- Table 1, page 4 is updated to add new signal names, clear_buffer_i, pp_ratio_i and their descriptions.
- The configuration parameter g_PP_RATIO is deleted in Table 2, page 4.
- The count value of the resource, “Sequential elements” is changed from 960 to 980 in Table 3, page 5.
- The equation to calculate theta_factor constant is edited in Hardware Implementation, page 3.

1.3 Revision 1.1

The following is a summary of the changes made in revision 1.1 (Published in January 2017) of this document.

- Key features were added. For more information, see Key Features, page 2.
- Supported family information was added. For more information, see Supported Families, page 2.
- Information on scaling the calculated speed using the theta factor was added. For more information, see Hardware Implementation, page 3.
- The description for the calib_angle_i input signal was updated. For more information, see Table 1, page 4.
- The PP_RATIO configuration parameter was added. For more information, see Table 2, page 4.

1.4 Revision 1.0

Revision 1.0 (Published in November 2016) was the first publication of this document.
2 Overview

A resolver is a position sensor or transducer which measures the absolute angular position of the rotating shaft to which it is attached.

2.1 Key Features
The resolver interface IP block supports the following features:

• Provides a high frequency signal for excitation
• Demodulates sine and cosine winding inputs
• Computes angle and speed

2.2 Supported Families
The resolver interface IP block supports the following families:

• SmartFusion®2
• IGLOO®2
• RTG4™

2.3 Theory of Operation
The operating principle of a resolver is similar to the operating principle of a synchro. Resolvers are typically built like small motors with a rotor (attached to the shaft whose position is to be measured) and a stator (stationary part) which takes the excitation signals and produces the output signals. A resolver typically consists of a primary winding, also called excitation winding and two secondary windings called cosine and sine windings. The secondary windings are geometrically placed such that winding signals are cosine and sine function of rotor angle.

The following figure shows the signals generated by resolver.

Figure 1 • Signal Generation in Resolver

The device used to process the resolver signals and convert them in to a digital angle format is called a resolver to digital converter (RDC). The resolver IP implements the functionality of a RDC. The IP provides excitation signal in the form of a square wave to the primary winding whose frequency can be configured. The IP processes the secondary signals, demodulates them and calculates the angle and speed of the rotor.
3 Hardware Implementation

The following figure shows the block diagram of the resolver interface.

**Figure 1 • System-Level Block Diagram of Hardware Implementation**

The resolver interface IP generates a square wave that is fed to the primary winding of the resolver. The frequency of the square wave can be configured through `hf_sig_period_i` input. The `cos_i` and `sin_i` signals from the secondary windings are demodulated and filtered to get effective cosine and sine signals. A phase-locked loop (PLL) is used to extract angle and speed from cosine and sine signals.

The PLL uses a PI controller whose gains `pll_pi_kp_i` and `pll_pi_ki_i` can be tuned to get required response time. A higher value for gains results in quick response to angle and speed changes but can also induce noise in angle and speed outputs.

In motor control application, the resolver zero position must be aligned with motor magnetic zero position. To achieve this, a `calib_angle_i` signal is used. During calibration process, the signal goes high and the motor is forced to align its rotor to magnetic zero position. The angle output is reset to zero during this period and is taken as reference for measuring absolute angle. A motor and resolver can have multiple pole pairs in which the motor control algorithm needs multiple theta transitions (3600) for one mechanical rotation of the rotor. This feature can be configured through the `pp_ratio_i` port, listed in Table 1, page 4.

The `theta_factor` constant is calculated by using the following equation. The calculated speed can be scaled to per unit using `theta_factor_i`.

$$\text{theta_factor} = \frac{18.12 \times \text{Rated Motor Speed (RPM)}}{\text{System Clock (MHz)}}$$

*EQ1*

The `hf_sig_period` input determines the frequency of square wave injected into resolver primary, calculated by using the following equation.

$$hf_{-}sig_{-}period_{-}i = \frac{f_{sys_{-}clk}}{hf_{-}freq \times 2}$$

*EQ2*

where,

- `hf_freq` = Frequency of the square wave injected into resolver primary
- `fsys_clk` = Frequency of the system clock provided at `sys_clk_i` input
3.1 Inputs and Outputs

The following table describes the input and output ports of the resolver interface block.

<table>
<thead>
<tr>
<th>Signal Name</th>
<th>Direction</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>reset_i</td>
<td>Input</td>
<td>Active low asynchronous reset signal to design</td>
</tr>
<tr>
<td>sys_clk_i</td>
<td>Input</td>
<td>System clock</td>
</tr>
<tr>
<td>clear_buffer_i</td>
<td>Input</td>
<td>When 1, internal speed filter buffer is cleared</td>
</tr>
<tr>
<td></td>
<td></td>
<td>When 0, buffer is normally operated</td>
</tr>
<tr>
<td>calib_angle_i</td>
<td>Input</td>
<td>The IP enters calibration state when this signal goes high. The angle offset between the resolver zero and the motor magnetic zero is calculated in this state</td>
</tr>
<tr>
<td>direction_config_i</td>
<td>Input</td>
<td>Specifies the direction of motor rotation</td>
</tr>
<tr>
<td>pp_ratio_i</td>
<td>Input</td>
<td>The ratio of number of motor poles to number of resolver poles expressed as exponent of 2. Example for motor poles 16, resolver poles 2, pp_ratio_i = 3; For motor poles 8, resolver poles 2, pp_ratio_i = 2; For motor poles 4, resolver poles 4, pp_ratio_i = 0</td>
</tr>
<tr>
<td>cos_i</td>
<td>Input</td>
<td>Cosine winding input (from ADC)</td>
</tr>
<tr>
<td>sin_i</td>
<td>Input</td>
<td>Sine winding input (from ADC)</td>
</tr>
<tr>
<td>pll_pi_kp_i</td>
<td>Input</td>
<td>Proportional gain of PI controller used for PLL</td>
</tr>
<tr>
<td>pll_pi_ki_i</td>
<td>Input</td>
<td>Integral gain of PI controller used for PLL</td>
</tr>
<tr>
<td>dc_filter_factor</td>
<td>Input</td>
<td>Filter time constant of high-pass filter used to eliminate DC value from Sine and Cosine signals</td>
</tr>
<tr>
<td>ac_filter_factor</td>
<td>Input</td>
<td>Filter time constant of low-pass filter used to eliminate modulation wave frequency component for Sine and Cosine signals</td>
</tr>
<tr>
<td>theta_factor_i</td>
<td>Input</td>
<td>Theta factor constant, as calculated from EQ1</td>
</tr>
<tr>
<td>hf_sig_period_i</td>
<td>Input</td>
<td>Half the value of the high frequency square wave time period, as calculated from EQ2</td>
</tr>
<tr>
<td>hf_signal_o</td>
<td>Output</td>
<td>Square wave signal used to drive primary winding of resolver</td>
</tr>
<tr>
<td>theta_o</td>
<td>Output</td>
<td>Angle output of resolver; equivalent to motor electrical angle</td>
</tr>
<tr>
<td>speed_o</td>
<td>Output</td>
<td>Speed output of resolver IP</td>
</tr>
</tbody>
</table>

3.2 Configuration Parameters

The following table describes the configuration parameters used in the hardware implementation of the resolver Interface. These are generic parameters and can be varied as per the application requirements.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>g_NO_MCYCLE_PATH</td>
<td>The number of clock delays required before the multiplication product ready signal is asserted</td>
</tr>
</tbody>
</table>
3.3 Timing Diagram

The following figure shows the timing diagram of the resolver interface block. The timing diagram represents a typical use case of the resolver interface block.

![Timing Diagram](image)

3.4 Resource Utilization

The following table lists the resource utilization of the resolver interface block after synthesis.

<table>
<thead>
<tr>
<th>Resource</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sequential elements</td>
<td>980</td>
</tr>
<tr>
<td>Combinational logic</td>
<td>1090</td>
</tr>
<tr>
<td>MACC</td>
<td>2</td>
</tr>
<tr>
<td>RAM1kx18</td>
<td>0</td>
</tr>
<tr>
<td>RAM64x18</td>
<td>0</td>
</tr>
</tbody>
</table>