Ramp Profile
Hardware Implementation

User Guide

Table of Contents

Ramp Profile Theory

A motor consists of a stationary stator and a moving rotor. Care should be taken to avoid subjecting the rotor to sudden and large magnitudes of speed changes. Sudden change in the motor speed applies a lot of mechanical stress on the motor system and leads to its malfunctioning in course of time. Hence, the speed of the rotor must be gradually increased with respect to time from the stationary position to the desired or reference speed. This change in speed characteristic of a motor is indicated by ramp-up profile, as shown in Figure 1. Similarly, the speed of the rotor must not come down abruptly from its maximum speed to a significantly lower reference speed or even its stationary position (0 rpm). The speed of the rotor must be gradually reduced. This is indicated by ramp-down profile as shown in Figure 2.

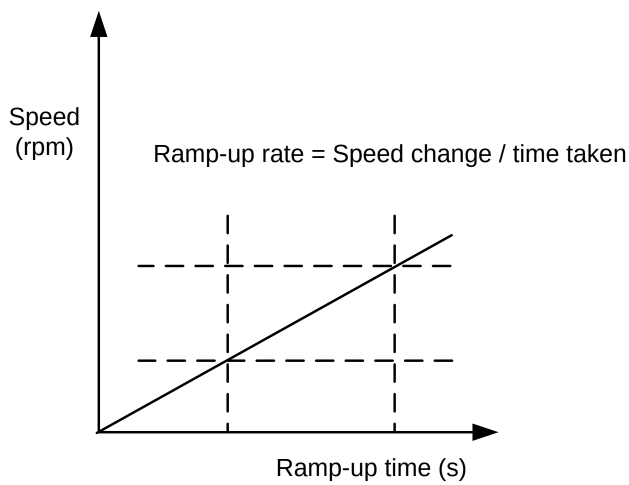

Figure 1 · Acceleration/Ramp-up Profile

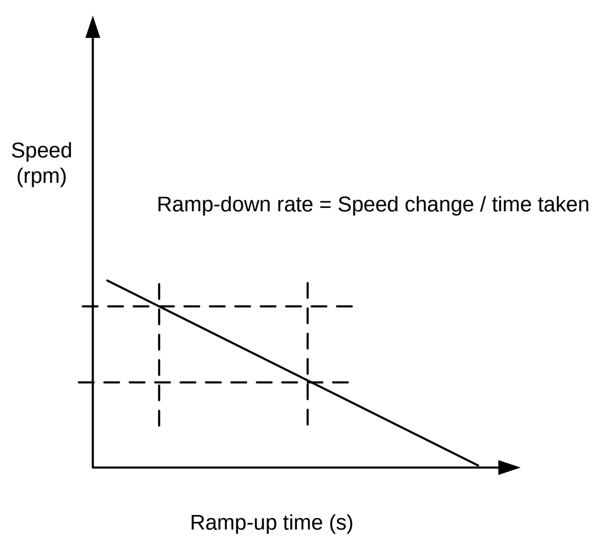

Figure 2 · Deceleration/Ramp-down Profile

The variation of speed with respect to time or slew rate of a motor for a particular application can be expressed in two ways:

1. Slew rate in reference variable count/sec
2. Slew rate in reference variable delay count

Slew Rate in Reference Variable Count/Sec (T_{sr})

For example, the required slew rate is 500 rpm/sec. Assume, the timer or timer interrupt service routine (ISR) is running at time interval T_c; where T_c is the time taken for the timer overflow.

Consider T_c to be equal to 50 µs.

The time taken for each variable count = $1/T_{sr}$ = 1/500 = 2 ms.

Hence, the ramp count = 2ms/50µs = 40.

The ramp count should be calculated at run time. The incremental variable can be of integer type. Care must be taken for achieving the desired ramp for a wide range of dynamic configuration of slew rate. The timer must be initially configured to the desired variable count time. For example, for lower slew rates such as 50 rpm/sec, the time taken for each variable count will be 20 ms and the timer available must be capable of measuring this time.

Slew Rate in Reference Variable Delay Count

The ramp count is given directly in terms of counts. For example, consider a variable delay count of 100 counts and T_c equal to 50 µs. The time taken for each ramp count is 100 * 50 µs = 5 ms.

Ramp Profile Hardware Implementation

The block level view of the Ramp Profile is shown in Figure 3.

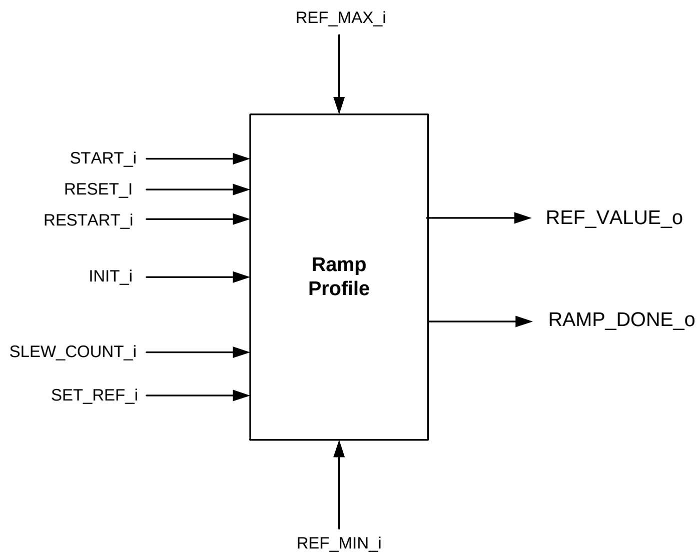

Figure 3 · Ramp Profile Block Diagram

The Ramp Profile block implements the following pseudo code:

```
if(ref_value <set_ref)
  ref_value = ref_value +1;  (For acceleration)
if (ref_value > set_ref)
  ref_value = ref_value – 1;  (For deceleration)
```

Checking for saturation Limit:

```
if(ref_value > ref_max)
  ref_value = ref_max;
if(ref_value < ref_min)
  ref_value = ref_min;
```

where,

ref_value = Current reference value

ref_max, ref_min = Maximum and minimum reference values, respectively

set_ref = Desired reference value

The above pseudo code is implemented as explained in the flow chart shown in Figure 4.

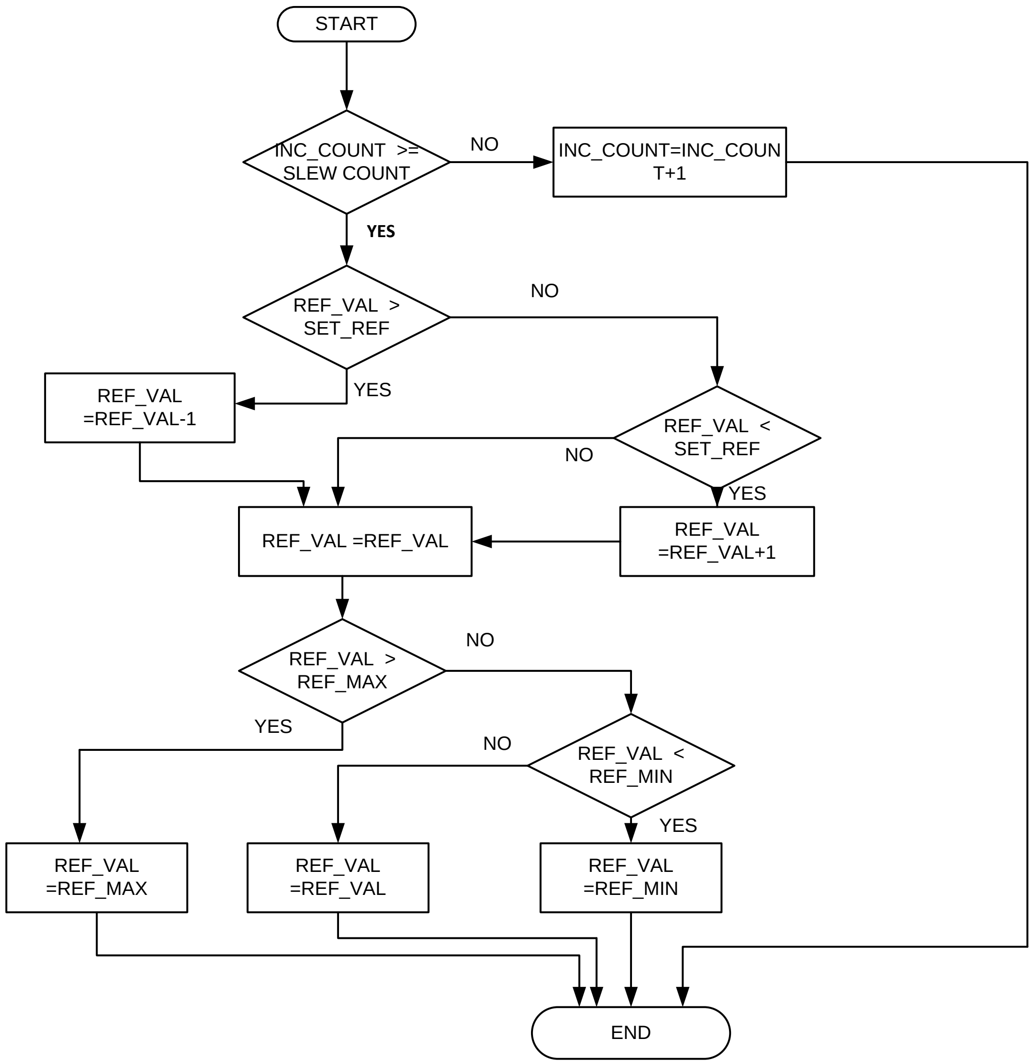

Figure 4 · Ramp Profile Implementation Flow

where,

INC_COUNT = Counter variable

SLEW COUNT = Maximum count value of the counter

When the Ramp Profile block is triggered, counter variable increments for every iteration of loop until it reaches the slew count value. A rising edge on the start signal must be detected for every iteration to begin. Once the increment count reaches the slew count, the reference value is incremented or decremented based on the set reference value. Then, the reference value obtained after incrementing (acceleration) or decrementing (deceleration) is compared with the set maximum and minimum output limits (ref_max and ref_min).

If the current reference value is greater than the ref_max (maximum output value), then the ref_max value is assigned as the current reference value. If the current reference value is less than the ref_min value, then the ref_min value is assigned as the current reference value. In case, the current reference value lies within the limits of ref_max and ref_min then it remains the same.

Inputs and Outputs

Table 1 describes the input and output ports of Ramp Profile block.

Table 1 · Input and Output Ports of Ramp Profile

Signal Name	Direction	Description
RESET_i	Input	Asynchronous reset signal to design. Active state is defined by RESET_STATE.
INIT_i	Input	Active high signal to initialize the whole Ramp Profile system.
RESTART_i	Input	Active high signal to restart the ramp-up/ramp-down process.
SYS_CLK_I	Input	System clock
REF_MAX_i	Input	Saturation maximum limit
REF_MIN_i	Input	Saturation minimum limit
SET_REF_i	Input	Desired reference value
START_i	Input	Start signal to trigger finite state machine (FSM).
SLEW_COUNT_i	Input	The rate at which the desired reference value is to be achieved.
REF_VALUE_o	Output	Current reference value
RAMP_DONE_o	Output	Active high signal to indicate ramp-up or ramp-down is done.

Configuration Parameters

Table 2 describes the configuration parameters used in the hardware implementation of the Ramp Profile. These are generic parameters and can be varied as per the application requirements.

Table 2 · Configuration Parameters

Name	Description
g_RESET_STATE	0: Supports active low reset 1: Supports active high reset
g_MAX_WIDTH	The maximum width of the reference value
g_MIN_WIDTH	The minimum width of the reference value
g_COUNT_WIDTH	The counter width
g_REF_WIDTH	The width of the reference value
g_INIT_SYNC	The boolean logic of SYNC pulse for Init signal
g_RESTART_SYNC	The boolean logic of SYNC pulse for restart
g_START_SYNC	The boolean logic of SYNC pulse for start

FSM Implementation

The FSM of the Ramp Profile in the current implementation has the following states:

- IDLE
- Ramp_count
- Ramp_acc
- Ramp_great or Ramp_less
- Ramp_sat
- Ramp_max or Ramp_min
- Ramp_s_check

The FSM of the Ramp Profile implemented is shown in Figure 5.

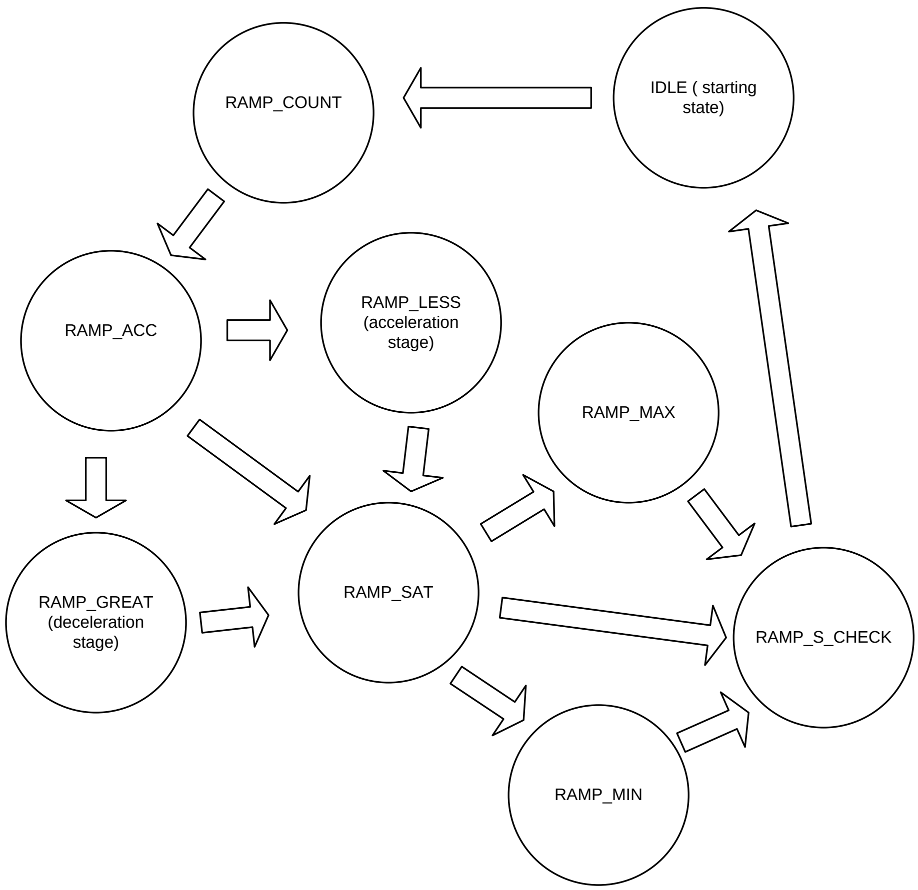

Figure 5 · Ramp Profile FSM

Note: For the FSM to be triggered, the system must be initialized (the INIT_i input signal must be made high) and also a rising edge on the start signal must be detected.

IDLE state: The FSM begins from this state. It comes to this state when the system is reset or after the counter variable is incremented (till it reaches the slew count value). In this state, ramp done signal remains low.

Ramp_count state: The counter variable is incremented in this stage in every iteration (triggered by start pulse) until it reaches the slew count value. If the counter (increment_count) value is less than the slew count value, then FSM goes to the idle state in the next clock cycle or else it moves to the Ramp_acc state.

Ramp_acc state: In this state, the current reference value is compared with the set reference value (desired reference value). If it is greater than the set reference value, then FSM moves to the Ramp_great state. If it is less than the set reference value, then the FSM moves to the Ramp_less state. In case it is equal to set reference value, then FSM moves to the Ramp_sat state.

Ramp_sat state: In this state, the current reference value is compared with the set maximum (ref_max) and minimum (ref_min) reference values. The current reference value is saturated to the maximum value if it is greater than ref_max and saturated to the minimum value if it is lesser than ref_min. If the reference value is within the range of ref_max and ref_min values, then the value remains same and the FSM moves to Ramp_s_check state in the next clock cycle.

Ramp_less state: In this state, the reference value is incremented by one. This state corresponds to ramp-up or acceleration. The FSM moves to the Ramp_sat state in the next clock cycle.

Ramp_great state: In this state, the reference value is decremented by one. This state corresponds to ramp-down or deceleration. The FSM moves to the Ramp_sat state in the next clock cycle.

Ramp_s_check state: In this state, the ramp done signal is made high (Reflected in the next clock cycle) indicating that the ramping action is performed. The FSM moves to the Idle state in the next clock cycle.

Ramp_max state: The reference value is assigned the ref_max value in this state. Since the current reference value is greater than the ref_max value, it is saturated to this value. In the next clock cycle, the FSM moves to the Ramp_s_check state.

Ramp_min state: The reference value is assigned the ref_min value in this state. Since the current reference value is lesser than the ref_min value, it is saturated to this value. In the next clock cycle, the FSM moves to the Ramp_s_check state.

Timing Diagram

The timing waveform of the Ramp profile block is shown in Figure 6:

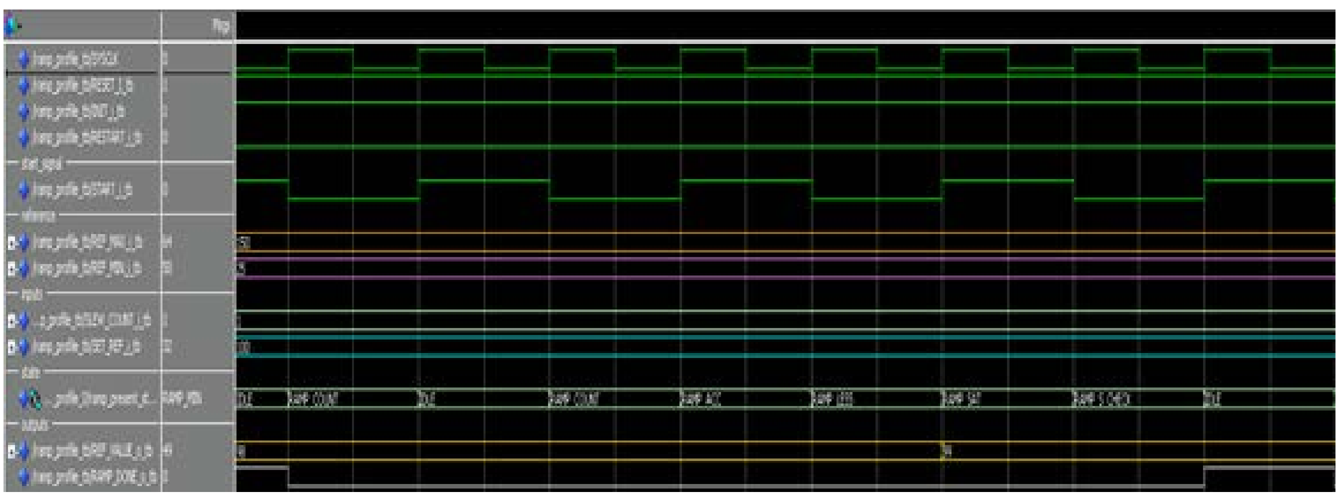

Figure 6 · Ramp Profile Timing Diagram

Color code:

Orange => ref_max

Purple=> ref_min

Cyan => set_ref

Yellow => ref_value (current reference value)

The waveform for ramp-up is shown in Figure 7.

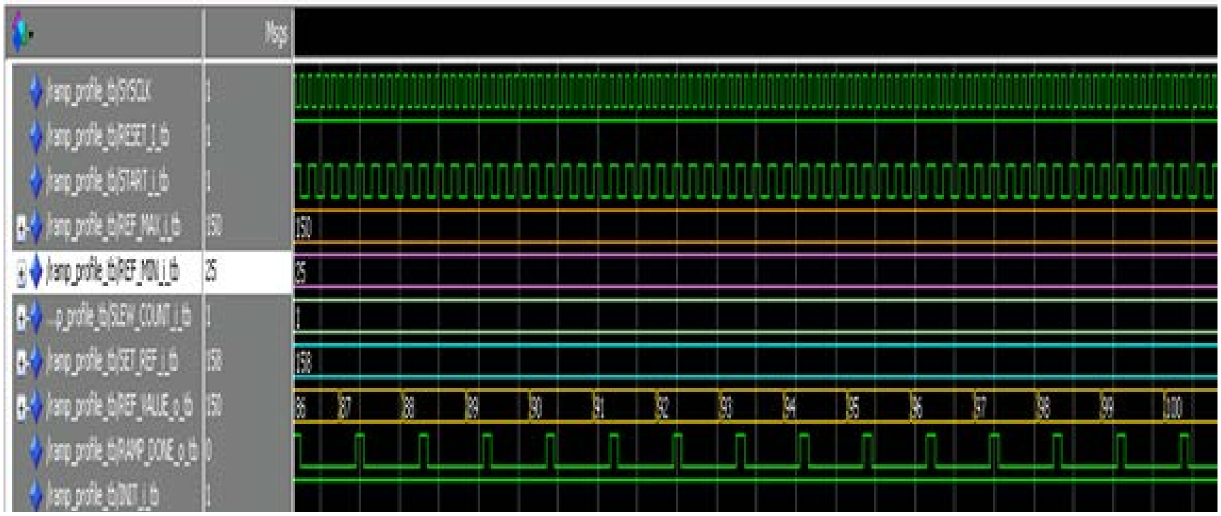

Figure 7 · Ramp-up

The waveform for ramp-down is shown in Figure 8.

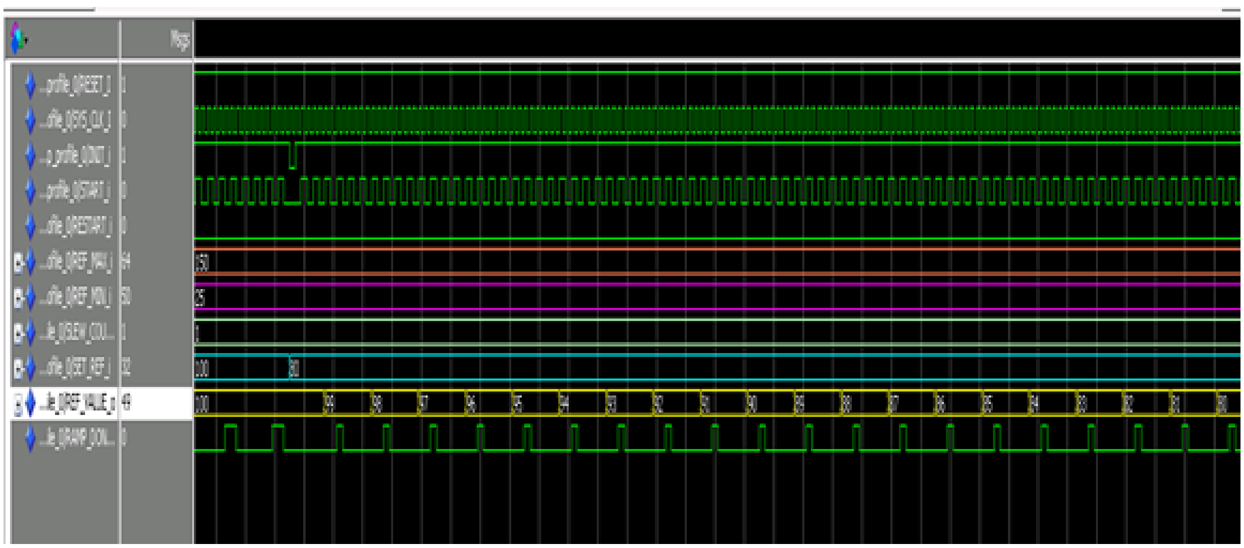

Figure 8 · Ramp-down

Resource Utilization

The Ramp Profile block is implemented on a SmartFusion2 system-on-chip (SoC) field programmable gate array (FPGA) device. The resource utilization report after synthesis is shown in Table 3.

Table 3 · Resource Utilization of Ramp Profile

Resource Usage Report for Ramp_profile	
Cell Usage	**Description**
CLKINT	2 uses
CFG2	44 uses
CFG3	21 uses

Resource Usage Report for Ramp_profile	
Cell Usage	**Description**
CFG4	125 uses
Carry primitives used for arithmetic functions	
ARI1	183 uses
Sequential Cells	
SLE	214 uses
Registers not packed on I/O Pads	214
DSP Blocks	0
I/O ports	166
I/O primitives	166
INBUF	133 uses
OUTBUF	33 uses
Global Clock Buffers	2
Total LUTs	190

Appendix

The entity definition of Ramp Profile block as defined in the source code as follows:

```
ENTITY Ramp_profile IS
GENERIC (
  g_START_SYNC : BOOLEAN := FALSE ;
  g_INIT_SYNC : BOOLEAN := FALSE ;
  g_RESTART_SYNC : BOOLEAN := FALSE ;
  g_RESET_STATE : STD_LOGIC:= '0' ;
  g_MAX_WIDTH : INTEGER := 32 ;
  g_MIN_WIDTH : INTEGER := 32 ;
  g_COUNT_WIDTH : INTEGER := 32 ;
  g_REF_WIDTH : INTEGER := 32
  ) ;
PORT (
  RESET_I : IN STD_LOGIC ;
  SYS_CLK_I : IN STD_LOGIC ;
  INIT_i : IN STD_LOGIC ;
  START_i : IN STD_LOGIC ;
  RESTART_i : IN STD_LOGIC ;
  REF_MAX_i : IN STD_LOGIC_VECTOR((g_MAX_WIDTH-1) downto 0) ;
  REF_MIN_i : IN STD_LOGIC_VECTOR((g_MIN_WIDTH-1) downto 0) ;
  SET_REF_i : IN STD_LOGIC_VECTOR((g_REF_WIDTH-1) downto 0) ;
  SLEW_COUNT_i : IN STD_LOGIC_VECTOR((g_COUNT_WIDTH-1) downto 0) ;
  REF_VALUE_o : OUT STD_LOGIC_VECTOR((g_REF_WIDTH-1) downto 0) ;
  RAMP_DONE_o : OUT STD_LOGIC
) ;
END Ramp_profile;
```

Product Support

Microsemi SoC Products Group backs its products with various support services, including Customer Service, Customer Technical Support Center, a website, electronic mail, and worldwide sales offices. This appendix contains information about contacting Microsemi SoC Products Group and using these support services.

Customer Service

Contact Customer Service for non-technical product support, such as product pricing, product upgrades, update information, order status, and authorization.

From North America, call **800.262.1060**
From the rest of the world, call **650.318.4460**
Fax, from anywhere in the world **408.643.6913**

Customer Technical Support Center

Microsemi SoC Products Group staffs its Customer Technical Support Center with highly skilled engineers who can help answer your hardware, software, and design questions about Microsemi SoC Products. The Customer Technical Support Center spends a great deal of time creating application notes, answers to common design cycle questions, documentation of known issues and various FAQs. So, before you contact us, please visit our online resources. It is very likely we have already answered your questions.

Technical Support

Visit the Microsemi SoC Products Group Customer Support website for more information and support (http://www.microsemi.com/soc/support/search/default.aspx). Many answers available on the searchable web resource include diagrams, illustrations, and links to other resources on website.

Website

You can browse a variety of technical and non-technical information on the Microsemi SoC Products Group home page, at http://www.microsemi.com/soc/.

Contacting the Customer Technical Support Center

Highly skilled engineers staff the Technical Support Center. The Technical Support Center can be contacted by email or through the Microsemi SoC Products Group website.

Email

You can communicate your technical questions to our email address and receive answers back by email, fax, or phone. Also, if you have design problems, you can email your design files to receive assistance. We constantly monitor the email account throughout the day. When sending your request to us, please be sure to include your full name, company name, and your contact information for efficient processing of your request.

The technical support email address is soc_tech@microsemi.com.

My Cases

Microsemi SoC Products Group customers may submit and track technical cases online by going to My Cases.

Outside the U.S.

Customers needing assistance outside the US time zones can either contact technical support via email (soc_tech@microsemi.com) or contact a local sales office. Sales office listings can be found at www.microsemi.com/soc/company/contact/default.aspx.

ITAR Technical Support

For technical support on RH and RT FPGAs that are regulated by International Traffic in Arms Regulations (ITAR), contact us via soc_tech_itar@microsemi.com. Alternatively, within My Cases, select **Yes** in the ITAR drop-down list. For a complete list of ITAR-regulated Microsemi FPGAs, visit the ITAR web page.

Microsemi Corporate Headquarters
One Enterprise, Aliso Viejo CA 92656 USA
Within the USA: +1 (949) 380-6100
Sales: +1 (949) 380-6136
Fax: +1 (949) 215-4996

Microsemi Corporation (NASDAQ: MSCC) offers a comprehensive portfolio of semiconductor solutions for: aerospace, defense and security; enterprise and communications; and industrial and alternative energy markets. Products include high-performance, high-reliability analog and RF devices, mixed signal and RF integrated circuits, customizable SoCs, FPGAs, and complete subsystems. Microsemi is headquartered in Aliso Viejo, Calif. Learn more at **www.microsemi.com**.

50200466-0/11.13