

RT PolarFire® Lockstep Processor Application Note

Introduction

Embedded systems for aerospace applications are more susceptible to transient faults due to radiation, EMI, voltage fluctuations, and power supply failure. Transient faults are becoming a concern at the ground level due to low core voltage, decrease in transistor geometry, and increase in switching speeds. Transient faults are real at the ground level in nuclear, automotive, and communication equipment.

To mitigate the transient and intermittent faults, lockstep systems—which are fault-tolerant computer systems—run the same set of operations on two identical systems with a known delay between them. This application note demonstrates dual-core lockstep processor architecture using soft Mi-V processor, which uses redundancy to detect the transient faults. An error is introduced by interrupting a processor and a system Reset is asserted as a response to the lockstep error detection.

Dual-Core Lockstep Processor Architecture

The dual-core lockstep processor architecture provides real-time diagnostics using an additional processor and a comparator. Two identical processors run the same application in lockstep with a known time delay between them. Any differences between two processors outputs are flagged as an error by the comparator on cycle-by-cycle basis. A temporal delay of 0.5 to 2 clock cycles between processors is appropriate to detect most of the common errors.

Design Requirements

The following table lists the hardware and software requirements for this demo design.

Table 1. Design Requirements

Requirement	Description
Hardware	
 RT PolarFire Evaluation Kit (RTPF500-CG1502) 12V, 5A AC power supply adapter and cords USB A to Mini-B cable 	To be released
Software	
Libero [®] SoC	Refer to the readme.txt file provided in the design files for the software versions used with this reference design.

Note: Libero SmartDesign and configuration screenshots provided in this guide are for illustration purpose only. Refer to the provided Libero design to see the latest updates.

Prerequisites

Before you begin:

- 1. Download and install Libero SoC from the following location: https://www.microsemi.com/product-directory/ design-resources/1750-libero-soc#downloads
- 2. For demo design files download: soc.microsemi.com/download/rsc/?f=rtpf_lockstep_df

Table of Contents

Intro	oductio	n1					
	1.	Dual-Core Lockstep Processor Architecture					
	2.	Design Requirements1					
	3.	Prerequisites					
1.	Desig	n Description4					
	1.1.	Clocking Structure					
	1.2.	Reset Structure					
	1.3.	Design Implementation					
2.	Dual (Core Lockstep Simulation9					
3.	Runni	ng the Demo10					
	3.1.	Programming the Device using Libero SoC					
	3.2.	Running the Design					
4.	Appendix: Programming the Device using FlashPro Express						
5.	5. Appendix: Running the TCL Script16						
6.	Revis	on History					
The	Micro	chip Website18					
Pro	duct C	nange Notification Service					
Cus	tomer	Support					
Mic	Microchip Devices Code Protection Feature						
Leg	Legal Notice						
Trac	demark	s					
Qua	ality Ma	nagement System					
Wor	ldwide	Sales and Service					

1. Design Description

The following figure shows high-level block diagram of the dual core lockstep design. The design has two Mi-V processor cores, LSRAM, comparator logic, GPIO IPs, and Debounce logic. Mi-V Processor-1 (Manager) is responsible for all the read or write transactions from or to the system memory (LSRAM) while Mi-V Processor-2 (Subordinate) executes the instructions fetched by Mi-V Processor-1.

A temporal delay of two clock cycles is introduced by releasing the Mi-V Processor-2 Reset two cycles after the Mi-V Processor-1 Reset. The processors execution is temporally separated to detect the faults which causes the same error on both the processors. Comparator logic is used to compare the read and write transactions from the two processors, and asserts a fault if there is a transaction mismatch. This fault bit is connected to the on-board LED. A sample GPIO application which blinks user LEDs on the board is used for testing the lockstep execution.

An external input with a Debounce logic is used to generate an interrupt to Mi-V Processor-1 for introducing a fault condition. As a fault recovery mechanism, a system Reset is applied after 30 seconds of detecting the fault. This 30 seconds timer is only used to visually see the fault LED glow before the system Reset is applied.

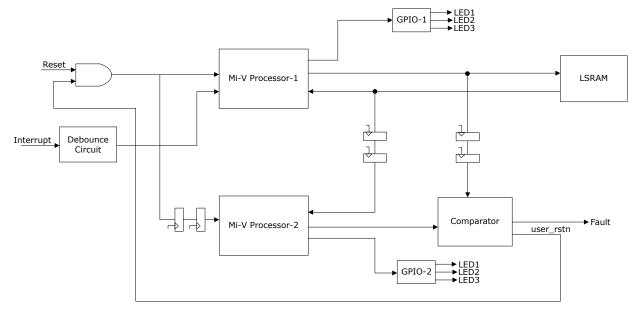
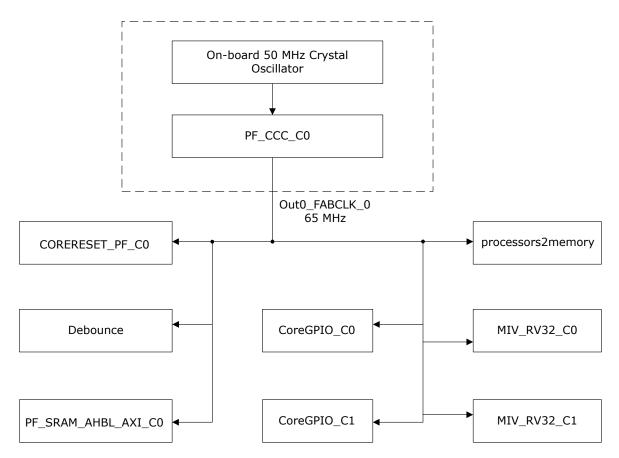


Figure 1-1. Lockstep Design High-level Block Diagram

1.1 Clocking Structure

In the design, there is a single clock domain. The on-board 50 MHz crystal oscillator is connected to the PF_CCC block, which generates 65 MHz system clock that is given to all the design blocks, as shown in following figure.

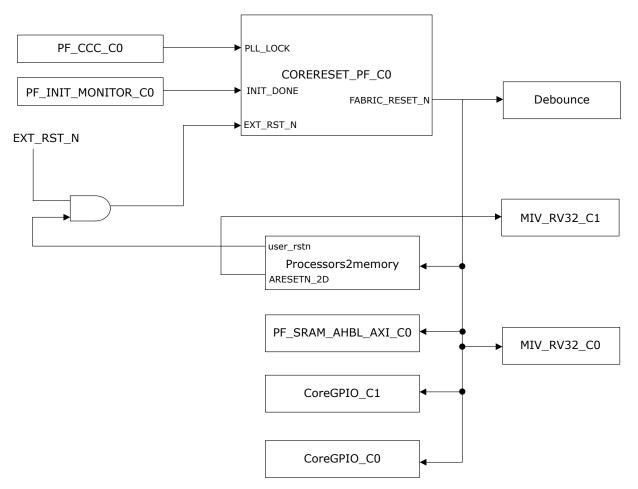
Figure 1-2. Clocking Structure



1.2 Reset Structure

In this design, the CORERESET_PF module generates the Reset signal as shown in the following figure. Reset signal to the MIV_RV32_C1 module is connected through ARESETN_2D port of the processors2memory module. This ensures MIV_RV32_C1 comes out of Reset state two clock cycles after MIV_RV32_C0. The user_rstn signal from the processors2memory module is ANDed with external Reset signal (EXT_RST_N) and the output is connected as an input to the CORERRESET_PF module. This is done to trigger user_rstn 30 seconds after an error is detected, which in-turn triggers a system Reset via the CORERESET_PF_C0. The following figure shows the Reset structure used in the design.

Figure 1-3. Reset Structure



1.3 Design Implementation

The following table lists the IP cores used in the reference design.

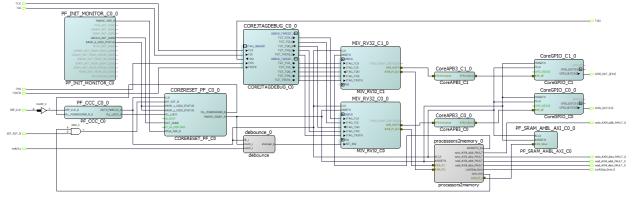
Table 1-1. IP Cores and Description

IP Core	Description
MIV_RV32	Mi-V soft processor
CoreJTAGDEBUG	Facilitates the connection of JTAG compatible soft core processors to the JTAG header for debugging. It provides fabric access to the JTAG interface using the UJTAG macro.
PF_INIT_Monitor	System controller uses this macro to check the status of device initialization. The device initialization includes SRAM initialization from µPROM/sNVM/SPI Flash. The DEVICE_INIT_DONE signal is used as a Reset.
PF_SRAM_AHBL_AXI	PolarFire LSRAM. Used as a system memory for the Mi-V processor.
PF_CCC	Macro to access PolarFire CCC block. It is used to synthesize 65 MHz clock frequency from the CCC with an on-board 50 MHz reference clock.
COREReset_PF	Generates a Reset, which is asserted asynchronously by one of the multiple potential sources and which negates synchronously to a specified clock.
CoreGPIO	Core GPIO provides an APB register-based interface with 32 GPIOs. It is used to check whether the application runs as expected.
CoreAPB3	Provides APB fabric for interconnecting an APB master and up to 16 APB slaves.

Configure two MIV_RV32 processors with Reset vector address (RSA) set to 0x8000_0000. In the Mi-V processors memory map, the 0x8000_0000 to 0x8001_FFFF range is defined for the AXI master interface, and the 0x6000_0000 to 0x6FFF_FFFF range is defined for the APB master interface. MIV-RV32_C0 processor accesses PF_SRAM_AHBL_AXI_C0 (processor main memory, where the application code (hex file) is stored) by sending transactions between addresses 0x8000_0000 to 0x8001_FFFF.

The following figure shows the SmartDesign view of the dual core lockstep design.

Figure 1-4. Smart Design Dual Core Lockstep Design



The processors2memory RTL module directs the MIV-RV32_C0 processor's read and write transactions to the PF_SRAM_AHBL_AXI and delays the read data transactions from the PF_SRAM_AHBL_AXI memory to MIV-RV32_C1 processor by two clock cycles. It also delays the read address, write address, and write data transactions from the MIV-RV32_C0 processor by two clock cycles to compare with MIV-RV32-C1 processor's read address, write address, and write data transactions. This RTL module also has comparison logic to compare the transactions from two processors on cycle-by-cycle basis and flags an error on Lockstep_Error output if there is a transaction mismatch. Lockstep_Error output gets asserted if there is a any mismatch on AXI address, AXI data, or AXI sideband/control signals. This Lockstep_Error output is mapped to an on-board LED.

MIV-RV32 processor's AXI master interface connects to processors2memory module's AXI slave interface while processors2memory module's AXI master interface connects to PF_SRAM_AHBL_AXI module's slave interface.

The APB interface of CoreGPIO IPs is connected to the master APB interface of MIV_RV32 processors through CoreAPB3. CoreGPIO is configured to have 32-bits APB width and four output ports. These output ports are mapped to the debug LEDs on the board to verify whether the application code works as expected.

An I/O pad with Debounce logic is used to introduce an error in the dual core lockstep design by asserting an interrupt (EXT_IRQ) to the MIV-RV32_C0 core. MIV-RV32_C0 core switches the execution to interrupt service routine (ISR) while MIV-RV32_C1 core continues executing its normal transactions. This leads to transactions mismatch and an error is detected.

As a fault recovery mechanism, system Reset is applied to the design after 30 seconds of detecting the fault. Here, the 30 seconds delay is used to prolong the LED glowing when the fault is detected. This is done through user RTL logic where user_rstn output from processors2memory module is ANDed with external Reset (EXT_RST_N).

2. Dual Core Lockstep Simulation

There is a HDL testbench for running the simulation for the dual core lockstep design. Clock generator block generates 50 Mhz clock. Reset generator generates active-low Reset for the design and is asserted for initial 1000 ns of running simulation.

To run the simulation, go to **Stimulus Hierarchy** and verify if the Lock_step_processor_tb file is set as active stimulus. If not set, right-click on the file and select **Set as active stimulus**.

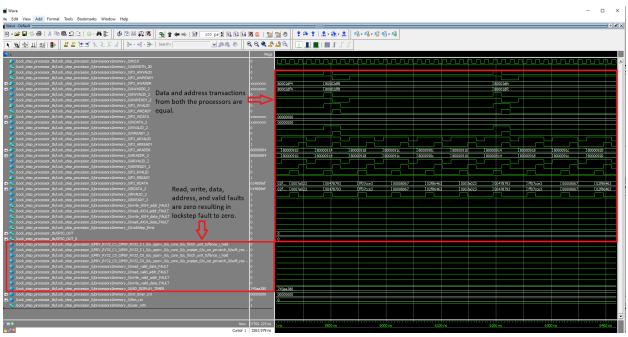
In the **Design Flow** tab, double-click **Simulate** under **Verify Pre-Synthesized Design** to simulate the design as shown in the following figure.

Figure 2-1. Run Simulation

🖻 🕨 Create Design	
SD Create SmartDesign	
Create HDL	
🚟 🄐 Create SmartDesign Testbench	
Create HDL Testbench	
Verify Pre-Synthesized Design	
🛄 🚟 Simulate	
Constraints	-
🔤 👔 Manage Constraints	
🔰 🖨 🕨 Implement Design	
🖓 Open Netlist Viewer	
Synthesize	

The following figure shows the simulation waveform for the dual core lockstep design highlighting the data and address transactions from both the processors. The address, data, and control signals coming out from the two processors are matching and their corresponding faults are zero until the error is introduced as also highlighted. The design is kept in the Reset state for initial 1000 ns.





3. Running the Demo

This chapter describe the steps to program the RT PolarFire device with the dual core lockstep design and detailed procedure to inject faults in the design through the SmartDebug tool.

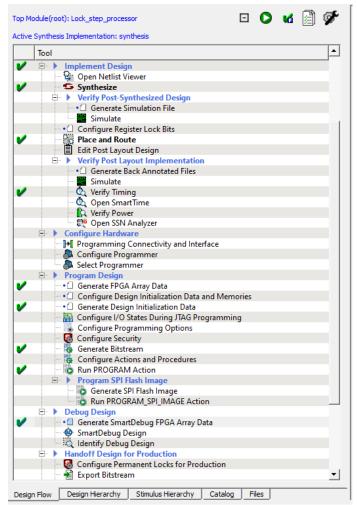
Before programming the RT PolarFire device, check the following:

- Ensure the jumper settings on the board is same as the default jumper settings specified in *RT PolarFire Evaluation Kit User Guide*.
- Connect the host PC to the J24 connector using the USB cable.
- Connect the power supply to the J19 connector and switch ON the power supply switch, SW7.

3.1 Programming the Device using Libero SoC

The RT PolarFire device can be programmed using the provided Libero SoC project. The following figure shows the Libero design flow.

Figure 3-1. Libero Design Flow Program Action



To program the device, follow these steps:

- 1. Open the design in Libero SoC. The Libero **Design Flow** window appears.
- 2. Double-click **Configure Design Initialization Data and Memories** on the **Design Flow** tab.

- 3. In the Fabric RAMs tab, double-click on PF_SRAM_AHBL_AXI_C0_0 from the list of logical instances and click Edit.
- 4. In the Edit Fabric RAM Initialization Client window, ensure that the storage type is selected as sNVM and in the Import Memory file location dialog box, sample_gpio.hex is properly located as shown in the following figure. The system controller initializes the RAM instance with the sNVM client's content at power-up.

Figure 3-2. Edit Fabric RAM Initialization Client

Apply Discard Help Usage statistics	Clients									
LSRAM Memory	Load design con	figuration Edit Initialize all clients from: Initialize all Clients from sNVM	•							
Available Memory(Bytes): 3891200	Filter out Inferred		_							
Used Memory(Bytes): 163840 Free Memory(Bytes) : 3727360		Edit Fabric RAM Initialization Client	?	×		PORTA Depth * Width	PORTB Depth * Width	Memory Content	Storage Type	Memory So
	1 MIV_RV32_C0_0	Client name: PF_SRAM_AHBL_AXI_C0_0		-	em[31:0]	32x32	32x32	No content	sNVM	Synthesis
	2 MIV_RV32_C0_0	Physical Name:	RAM1K20	IP	em_1[31:0]	32x32	32x32	No content	sNVM	Synthesis
	3 MIV_RV32_C1_0	RAM Initialization Options			:m[31:0]	32x32	32x32	No content	sNVM	Synthesis
	4 MIV_RV32_C1_0	Initialized Content from Synthesis Content Initialized from configurator(using content editor option)			em_1[31:0]	32x32	32x32	No content	sNVM	Synthesis
Used space	5 PF SRAM AHB	Content from file: sample_gpio.hex				32768x40	32768×40	sample_gpio.hex	sNVM	Configurator
Prec space		Imported Memory file location : sample_gpio.hex C Content filed with 0s No content (ident is a placeholder and will not be programmed) Optimize firs: C High Speed Storage Type Help	Close		Ź.					

In the sNVM tab, ensure that the sNVM client is added as shown in the following figure.
 Figure 3-3. sNVM Tab

Design Initialization UPROM	sNVM	SPI I	Flash Fabric RA	AMs			
Apply Discard	He	<u> </u>					
Usage statistics		Clier	nts				
Available memory (in pages):	221						
Used memory (in pages):	117		Add 🔻	Edit	Delete	Load design config	uration
Free memory (in pages):	104	=					
			Clien	t Name	Start Page	Number of bytes	End Page
		1	INIT_STAGE_1_	SNVM_CLIENT	202	4376	219
		2	INIT_STAGE_2_	3_SNVM_CLIENT	0	24704	98

6. Double-click **Run PROGRAM Action.** Once the device is successfully programmed, a green tick appears on **Run PROGRAM Action**.

Note: To program the device using FlashPro Express, see 4. Appendix: Programming the Device using FlashPro Express.

3.2 Running the Design

Once the device is programmed, LED2, LED3, LED6, and LED7 continuously toggle which ensures that the MIV-RV32_C0 and MIV-RV32_C1 are in lockstep. To inject fault, follow these steps:

- 1. Press external push button, SW1.
- 2. LED8 on the board glows to show the lockstep error is introduced.
- 3. System Reset must be asserted after 30 seconds of detecting the fault. When the system Reset is released, application code restarts and lockstep error is zero. LED1, LED5, and LED8 stops glowing.

Note: LED1 glows as the interrupt service routine (ISR) is called after the external IRQ is triggered for MIV-RV32_C0. The same set of instructions to execute the ISR for MIV-RV32_C0 reaches MIV-RV32_C1 (via the read bus) which glows LED5.

When interrupt is asserted, there is a difference in MIV-RV32_C0 and MIV-RV32_C1 output ports for few clock cycles, which asserts the lockstep error, but the MIV-RV32_C1 might sync-up with MIV-RV32_C0 as the valid instructions are delivered to MIV-RV32_C1 as well. Hence, LED2, LED3, LED6, and LED7 continue to toggle. Ideally, there should be a system Reset or corrective action immediately after the lockstep error.

This concludes the demo.

4. Appendix: Programming the Device using FlashPro Express

This section describes how to program the RT PolarFire device with the .job programming file using FlashPro Express. The .job file is available at the following design files folder location: <download_folder>\Programming_Job

To program the device, complete the following steps:

- 1. On the host PC, launch the **FlashPro Express** software.
- 2. Click **New** or select **New Job Project** from **Project** menu to create a new job project, as shown in the following figure.

Figure 4-1. New Job Project - FlashPro Express

Project View Tools	Help	
New Job Project	Ctrl+N	
🚰 Open Job Project	Ctrl+O	
× Close Job Project		
Save Job Project	Ctrl+Shift+A	A
Set Log File)÷.
Export Log File		
Export Devices In	fo	
Preferences		
Execute Script	Ctrl+U	
Export Script File		
Recent Projects		•
Exit	Ctrl+Q	

- 3. Enter the following in the **Create New Job Project** dialog box:
 - Programming job file: Click Browse and navigate to the location where the .job file is located and select the file. The default location is: <download_folder>\Programming_Job
 - FlashPro Express job project location: Click Browse and navigate to the location where you want to save the project.

Create New Job Project			\times
Import FlashPro Express job file			
hit S\ockstep_w_interrupt_2021.	2\designer\Lock_step_processor\export\Lock_step_processor.job	Browse	
C Construct automatically (developer	mode) 🚺		
Connected programmers:	▼ Refresh		
Programming interface; JTAG			
FlashPro Express job project name:	Lock_step_processor		
FlashPro Express job project location:	ep_w_interrupt_2021.2\designer\Lock_step_processor\export	Browse	
Help	0	K Cancel	

- 4. Click OK. The required programming file is selected and ready to be programmed in the device.
- 5. The **FlashPro Express** window appears as shown in the following figure. Confirm that a programmer number appears in the Programmer field. If it does not, confirm the board connections and click **Refresh/Rescan Programmers**.

Figure 4-3. Refresh/Rescan Programmers

Project View lools Help			
Refresh/Rescan Programmers			
Programmer	 		
1 💽 🗹 🍂 87302 IDLE	IDLE		
		IDLE	
RUN			
og			8 ×
ENCONTRACTOR A Warnings I Info			
Created FlashPro Express Job Project. Rescanning for Programmers Programmer '87302' : JTAG TCK frequency = 1 MHz	:		
Programmer: Embedded FF6 '5BC8E8F8' : Version =	EF9B		
Programmer 'E2003VPMPG' : JTAG TCK / SPI SCK fr programmer '87302' : FlashPro4	equency = 1 MHz		_
programmer '5BC8E8F8' : Embedded FlashPro6 programmer 'E2003VPMPG' : FlashPro5 Rescanning for Programmers DONE.			-
			<u>·</u>

6. Click **RUN** to program the device. When the device is programmed successfully, a **PROGRAMMER(S) PASSED** status is displayed as shown in the following figure.

Figure 4-4.	Programming	Successful
-------------	-------------	------------

Project View Tools Help			
Refresh/Rescan Programmers			
Programmer	 ктребоотз тво тво тво 		
1 🚺 🗹 🎢 87302 RUN PASSED	PASSED		
PROGRAM	1	PROGRAMMER(S) PASSED	
RUN			
og			8×
programmer '87302' : device 'RTPF500TS' : EXPO	075e0deb654c06c6ef RT DSN[128] = df7823217fe	19228814ccdb2053e7a91	<u> </u>
programmer '87302': device 'RTF500T5': Exec programmer '87302': Chain programming PASSED. Chain Programming Finished: Wed Sep 1 10:22:0	uting action PROGRAM PASS	ED.	
0 - 0 - 0 - 0 - 0 - 0			•

7. Close FlashPro Express, Project > Exit.

See 3. Running the Demo section to run the demo.

5. Appendix: Running the TCL Script

TCL scripts are provided in the design files folder under the directory **TCL_Scripts**. If required, the design flow can be reproduced from Design Implementation before the job file is generated.

To run TCL, follow these steps:

- 1. Launch the Libero software
- 2. Select **Project > Execute Script....**
- 3. Click Browse and select script.tcl from the downloaded TCL_Scripts directory.
- 4. Click Run.

After successful execution of the TCL script, the Libero project is created within the TCL_Scripts directory.

For more information about TCL scripts, see <design_name>/TCL_Scripts/readme.txt.

See *Libero[®] SoC TCL Command Reference Guide* for more information about TCL commands. Contact Technical Support for any queries about running the TCL script.

6. Revision History

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

Table 6-1. Revision History

Revision	Date	Description
A	10/2021	The first publication of the document.

The Microchip Website

Microchip provides online support via our website at www.microchip.com/. This website is used to make files and information easily available to customers. Some of the content available includes:

- **Product Support** Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** Frequently Asked Questions (FAQs), technical support requests, online discussion groups, Microchip design partner program member listing
- **Business of Microchip** Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

Product Change Notification Service

Microchip's product change notification service helps keep customers current on Microchip products. Subscribers will receive email notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, go to www.microchip.com/pcn and follow the registration instructions.

Customer Support

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Embedded Solutions Engineer (ESE)
- Technical Support

Customers should contact their distributor, representative or ESE for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in this document.

Technical support is available through the website at: www.microchip.com/support

Microchip Devices Code Protection Feature

Note the following details of the code protection feature on Microchip products:

- · Microchip products meet the specifications contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is secure when used in the intended manner, within operating specifications, and under normal conditions.
- Microchip values and aggressively protects its intellectual property rights. Attempts to breach the code protection features of Microchip product is strictly prohibited and may violate the Digital Millennium Copyright Act.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of its code. Code protection does not mean that we are guaranteeing the product is "unbreakable". Code protection is constantly evolving. Microchip is committed to continuously improving the code protection features of our products.

Legal Notice

This publication and the information herein may be used only with Microchip products, including to design, test, and integrate Microchip products with your application. Use of this information in any other manner violates these terms. Information regarding device applications is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. Contact your local Microchip sales office for additional support or, obtain additional support at www.microchip.com/en-us/support/design-help/client-support-services.

THIS INFORMATION IS PROVIDED BY MICROCHIP "AS IS". MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE, OR WARRANTIES RELATED TO ITS CONDITION, QUALITY, OR PERFORMANCE.

IN NO EVENT WILL MICROCHIP BE LIABLE FOR ANY INDIRECT, SPECIAL, PUNITIVE, INCIDENTAL, OR CONSEQUENTIAL LOSS, DAMAGE, COST, OR EXPENSE OF ANY KIND WHATSOEVER RELATED TO THE INFORMATION OR ITS USE, HOWEVER CAUSED, EVEN IF MICROCHIP HAS BEEN ADVISED OF THE POSSIBILITY OR THE DAMAGES ARE FORESEEABLE. TO THE FULLEST EXTENT ALLOWED BY LAW, MICROCHIP'S TOTAL LIABILITY ON ALL CLAIMS IN ANY WAY RELATED TO THE INFORMATION OR ITS USE WILL NOT EXCEED THE AMOUNT OF FEES, IF ANY, THAT YOU HAVE PAID DIRECTLY TO MICROCHIP FOR THE INFORMATION.

Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

Trademarks

The Microchip name and logo, the Microchip logo, Adaptec, AnyRate, AVR, AVR logo, AVR Freaks, BesTime, BitCloud, CryptoMemory, CryptoRF, dsPIC, flexPWR, HELDO, IGLOO, JukeBlox, KeeLoq, Kleer, LANCheck, LinkMD, maXStylus, maXTouch, MediaLB, megaAVR, Microsemi, Microsemi logo, MOST, MOST logo, MPLAB, OptoLyzer, PIC, picoPower, PICSTART, PIC32 logo, PolarFire, Prochip Designer, QTouch, SAM-BA, SenGenuity, SpyNIC, SST, SST Logo, SuperFlash, Symmetricom, SyncServer, Tachyon, TimeSource, tinyAVR, UNI/O, Vectron, and XMEGA are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

AgileSwitch, APT, ClockWorks, The Embedded Control Solutions Company, EtherSynch, Flashtec, Hyper Speed Control, HyperLight Load, IntelliMOS, Libero, motorBench, mTouch, Powermite 3, Precision Edge, ProASIC, ProASIC Plus, ProASIC Plus logo, Quiet- Wire, SmartFusion, SyncWorld, Temux, TimeCesium, TimeHub, TimePictra, TimeProvider, TrueTime, WinPath, and ZL are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Adjacent Key Suppression, AKS, Analog-for-the-Digital Age, Any Capacitor, AnyIn, AnyOut, Augmented Switching, BlueSky, BodyCom, CodeGuard, CryptoAuthentication, CryptoAutomotive, CryptoCompanion, CryptoController, dsPICDEM, dsPICDEM.net, Dynamic Average Matching, DAM, ECAN, Espresso T1S, EtherGREEN, GridTime, IdealBridge, In-Circuit Serial Programming, ICSP, INICnet, Intelligent Paralleling, Inter-Chip Connectivity, JitterBlocker, Knob-on-Display, maxCrypto, maxView, memBrain, Mindi, MiWi, MPASM, MPF, MPLAB Certified logo, MPLIB, MPLINK, MultiTRAK, NetDetach, NVM Express, NVMe, Omniscient Code Generation, PICDEM, PICDEM.net, PICkit, PICtail, PowerSmart, PureSilicon, QMatrix, REAL ICE, Ripple Blocker, RTAX, RTG4, SAM-ICE, Serial Quad I/O, simpleMAP, SimpliPHY, SmartBuffer, SmartHLS, SMART-I.S., storClad, SQI, SuperSwitcher, SuperSwitcher II, Switchtec, SynchroPHY, Total Endurance, TSHARC, USBCheck, VariSense, VectorBlox, VeriPHY, ViewSpan, WiperLock, XpressConnect, and ZENA are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

The Adaptec logo, Frequency on Demand, Silicon Storage Technology, Symmcom, and Trusted Time are registered trademarks of Microchip Technology Inc. in other countries.

GestIC is a registered trademark of Microchip Technology Germany II GmbH & Co. KG, a subsidiary of Microchip Technology Inc., in other countries.

All other trademarks mentioned herein are property of their respective companies.

[©] 2021, Microchip Technology Incorporated and its subsidiaries. All Rights Reserved.

ISBN: 978-1-5224-9007-4

Quality Management System

For information regarding Microchip's Quality Management Systems, please visit www.microchip.com/quality.



Worldwide Sales and Service

ASIA/PACIFIC		
Austrolia Sudrau	ASIA/PACIFIC	EUROPE
	-	Austria - Wels Tel: 43-7242-2244-39
		Fax: 43-7242-2244-39
		Denmark - Copenhager
•		Tel: 45-4485-5910
		Fax: 45-4485-2829
•. •	•	Finland - Espoo
		Tel: 358-9-4520-820
•••		France - Paris
		Tel: 33-1-69-53-63-20
-		Fax: 33-1-69-30-90-79
		Germany - Garching
-		Tel: 49-8931-9700
		Germany - Haan
		Tel: 49-2129-3766400
		Germany - Heilbronn
		Tel: 49-7131-72400
		Germany - Karlsruhe
-		Tel: 49-721-625370
		Germany - Munich
China - Shanghai	Singapore	Tel: 49-89-627-144-0
		Fax: 49-89-627-144-44
China - Shenyang	Taiwan - Hsin Chu	Germany - Rosenheim
Tel: 86-24-2334-2829	Tel: 886-3-577-8366	Tel: 49-8031-354-560
China - Shenzhen	Taiwan - Kaohsiung	Israel - Ra'anana
Tel: 86-755-8864-2200	Tel: 886-7-213-7830	Tel: 972-9-744-7705
China - Suzhou	Taiwan - Taipei	Italy - Milan
Tel: 86-186-6233-1526	Tel: 886-2-2508-8600	Tel: 39-0331-742611
China - Wuhan	Thailand - Bangkok	Fax: 39-0331-466781
Tel: 86-27-5980-5300	Tel: 66-2-694-1351	Italy - Padova
China - Xian	Vietnam - Ho Chi Minh	Tel: 39-049-7625286
Tel: 86-29-8833-7252	Tel: 84-28-5448-2100	Netherlands - Drunen
China - Xiamen		Tel: 31-416-690399
Tel: 86-592-2388138		Fax: 31-416-690340
China - Zhuhai		Norway - Trondheim
Tel: 86-756-3210040		Tel: 47-72884388
		Poland - Warsaw
		Tel: 48-22-3325737
		Romania - Bucharest
		Tel: 40-21-407-87-50
		Spain - Madrid
		Tel: 34-91-708-08-90
		Fax: 34-91-708-08-91
		Sweden - Gothenberg
		Tel: 46-31-704-60-40
		Sweden - Stockholm
		Tel: 46-8-5090-4654
		UK - Wokingham
		Tel: 44-118-921-5800
		Fax: 44-118-921-5820
		1 a. ++-110-321-3020
	Tel: 86-21-3326-8000 China - Shenyang Tel: 86-24-2334-2829 China - Shenzhen Tel: 86-755-8864-2200 China - Suzhou Tel: 86-186-6233-1526 China - Wuhan Tel: 86-27-5980-5300 China - Xian Tel: 86-29-8833-7252 China - Xiamen Tel: 86-592-2388138 China - Zhuhai	Tel: 61-2-9868-6733 Tel: 91-80-3090-4444 China - Beijing India - New Delhi Tel: 86-10-8569-7000 Tel: 91-11-4160-8631 China - Chengdu India - Pune Tel: 86-28-8665-5511 Tel: 91-20-4121-0141 China - Chongqing Japan - Osaka Tel: 86-23-8980-9588 Tel: 81-6-6152-7160 China - Dongguan Japan - Tokyo Tel: 86-769-8702-9880 Tel: 81-3-6880-3770 China - Guangzhou Korea - Daegu Tel: 86-71-8792-8115 Tel: 82-2-554-7200 China - Hong Kong SAR Malaysia - Kuala Lumpur Tel: 86-571-8792-8115 Tel: 60-3-7651-7906 China - Nanjing Malaysia - Penang Tel: 86-512-8502-7355 Tel: 60-3-2634-9065 China - Shanghai Singapore Tel: 86-21-3326-8000 Tel: 86-334-8870 Tel: 86-24-2334-2829 Tel: 86-3577-8366 China - Shenyang Taiwan - Hsin Chu Tel: 86-755-8864-2200 Tel: 886-3-771-830 China - Suzhou Taiwan - Taipei Tel: 86-186-6233-1526 Tel: 886-2-2508-6600 China - Suzhou Taiwan - Taipei Tel: 86-27-5980-5300 <

Application Note