

# **Evaluating SiFive RISC-**V Core IP

 $\mathbf{O}$ 

Drew Barbier – January 2018 drew@sifive.com



© 2018 SiFive. All Rights Reserved.

# 3 Part Webinar Series

Webinar Recordings and Slides: <a href="https://info.sifive.com/risc-v-webinar">https://info.sifive.com/risc-v-webinar</a>

- RISC-V 101
  - The Fundamentals of RISC-V architecture

- Introduction to SiFive RISC-V Core IP
  - Introduction to the SiFive E31 and E51 Core Complexes
- Getting Started with SiFive RISC-V Core IP



#### SiFive RISC-V Core IP Products

- This presentation is targeted at hardware and software embedded designers who want to get started evaluating SiFive RISC-V Core IP
- This presentation will walk through the RTL Evaluation deliverable of the E31 Core Complex as well programing, writing software, and Debugging the E31 Arty FPGA image







# **Obtaining the Evaluations**



# SiFive RISC-V Core IP Evaluations

Free, Instant Access to Evaluation RTL and FPGA bitstreams

#### • Verilog RTL Evaluation

- Fully functional, synthesizable, Verilog RTL
  - Synthesize into your process/eda flow to obtain accurate PPA for your process node
  - Simulate in your environment
- Restrictions on Evaluation RTL
  - Obfuscated Verilog
  - Reduced DTIM and System/Peripheral Port address space
- FPGA Evaluation
  - Pre-built FPGA bitstreams for low cost FPGA Platforms
  - Useful for software development





#### SiFive Product Evaluations are 2 Clicks Away





ŝ

5

#### **Evaluation License Agreement**





- Fill in your information on the web form
- An Evaluation agreement is then sent to the email address provided via DocuSign
- Sign the document via DocuSign being sure to click Finish at the bottom of the document
- Help:
  - sales@sifive.com
  - order-support@sifive.com



ŝ

ψ

(5)

1

# Download via SiFive Developer Dashboard

#### https://dev.sifive.com



## Evaluation Product Refresh – January 2018

- RTL Evaluation Changes
  - ITIM
  - 4 Hardware Breakpoints
  - Performance Counters
  - Port Interfaces
    - E31 AHB-Lite
    - E51 AXI
  - Number of Global Interrupts
    - E31 127
    - E51 255
- FPGA Evaluation Changes
  - ITIM
  - 64kB of DTIM
  - 8 Hardware Breakpoints



(5)



# E31 RTL Evaluation



• •

© 2018 SiFive. All Rights Reserved.

# The Verilog RTL Bundle



- *Sifive\_coreip\_E31\_AHB\_rtl\_eval\_v2p0/* 
  - info/
    - metadata about the design (DTS and memory dimensions)
  - tests/
    - simple tests to ensure a functional delivery
    - Also includes device header files
  - verilog/
    - design/
      - DUT (device under test)
    - memories/
      - behavioral models of SRAMs
    - testbench/
      - simulation testbench
  - Makefile
    - Runs VCS simulations

(?)

ψ

(5)

# **E-Series Core Complex Clocking**

- core\_clock\_0
  - Main CPU and L1 memory clock
- clock
  - Secondary peripheral clock
  - Frequency must be an integer multiple of core\_clock\_0
  - 1:1 ratio is acceptable
- rtc\_toggle
  - Real Time Clock input as defined by RISC-V Architecture (mtime)
  - Must run at strictly less than half the rate of *clock*
- core\_clock\_0 ≥ clock > (2x rtc\_toggle)



#### Synthesis Constraints and Clocking Recommendations

- For Maximum frequency, it is recommended for:
  - $clock = \frac{1}{2}(core_clock_0)$
- If targeting lower frequencies, *clock* = *core\_clock\_0* is OK
- Typical *rtc\_toggle* frequencies
  - 32.768kHz
  - 1MHz

########	############	+########	#########	<i>!</i> ##########	#####	####		
# Local	Synopsys De	esign Cons	straints	(SDC)				
# ########	###########	*#######	#########	*##########	#####	####		
######## # Timing #########	######################################	########## 5s ###########	########## ##########	:############ :#############	+#### +####	#### ####		
# # Clocks #								
# set pe	riod to cre	eate desi	red core	frequency				
set CLK_ set SYS_	PERIOD 5 CLK_PERIOD	[expr \$CL	K_PERIOD	* 2]				
# create create_c	e core clock lock -name	''core_cl	k'' -per	iod \$CLK_PE	RIOD	[get_ports	core_clock	₹_0]
# create	system clo	ock						



5



# Core Complex Memory Instances

- Behavioral models of the RAMS are provided as part of the deliverable in the file:
  - sifive\_coreip\_E31\_AHB\_rtl\_eval\_v2p0/verilog/memories/ECoreIP SubsystemAllPortRAMTestHarness.\*.rams.v
- For implementation it is necessary to generate memory instances which are specific to your foundry/process
  - Typically done using Memory Compilers from the foundry, or 3rd party IP providers

Name	Depth	Address Width (N <sub>addr</sub> )	Data Width (N <sub>data</sub> )	Write Mask Granularity (N <sub>part</sub> )	Description
data_arrays_0_ext	4096	12	32	8	DTIM data array
data_arrays_0_0_ext	2048	11	64	32	I-cache data array
tag_array_ext	256	8	38	19	I-cache tag array

Table 3.1: SRAM Modules and Configuration

Name	Direction	Width	Description
RW0_clk	Input	1	Memory clock.
RW0_en	Input	1	Active-high signal indicating that the memory is be-
	-		ing accessed. This may be used for clock gating.
RW0_addr	Input	$N_{addr}$	Address of access.
RW0_rdata	Output	$N_{data}$	Read data.
RW0_wmode	Input	1	Active-high signal indicating that the access is a write
			operation.
RW0_wdata	Input	$N_{data}$	Write data.
RW0_wmask	Input	$N_{data}/N_{part}$	Active-high write mask. Each bit controls whether or
	-		not the corresponding $N_{part}$ -bit subword is written.
			This is present only in memories that require masked
			write functionality.

Table 3.2: SRAM Signals



#### E31 Synthesized Area Hierarchy









• •

# Digilent Arty FPGA Platform

- Popular, Low cost, FPGA development board
  - Xilinx Artix-35T FPGA 33,280 Xilinx logic cells
  - 16MB QSPI serial flash interface
  - USB-UART, buttons, switches, LEDs, etc...
- Our FPGA evaluations execute at 65MHz
  - Allows for fast Software execution
- Purchase directly from Digilent:
  - <u>http://store.digilentinc.com/arty-a7-artix-7-fpga-development-board-for-makers-and-hobbyists/</u>



**Si**Five

ŝ

s

# E31 FPGA Evaluation Configuration

- SiFive Peripherals and Integration to Digilent Arty Platform
  - **QSPI** serial flash
  - GPIO to LEDs, Buttons, PMOD
  - **PWM** to LEDs
  - JTAG to PMOD-D
  - Buttons and Switches to Local and PLIC interrupt inputs
- Core Complex features
  - User Mode
  - 8 Hardware Breakpoints
  - 8 Region PMP
  - ITIM
  - Vectored Interrupts
  - Performance Counters



**Si**Five

## FPGA Memory Map, Interrupts, and Pins

- Manual is available from developer dashboard
  - Memory Map
  - Pinout of peripherals to PMOD connector, LEDs, Buttons, switches
  - Interrupt numbers for connected devices

Base	Тор	Description	Notes
$0000_{-}0000x0$	0x0000_00FF	Reserved	
$0x0000_0100$		Halt Notification	
$0x0000_0104$		Start Notification	
$0x0000_0108$		Resume Notification	
$0x0000_010C$		Exception Notification	Debug (4 KiB)
0x0000_0110	0x0000_02FF	Reserved	
$0x0000_0300$	0x0000_03FF	Debug RAM ( $\leq$ .25 KiB)	
$0x0000_0400$	0x0000_07FF	Debug Flags (≤ 1 KiB)	
$0x000_{0}080_{0}$	0x0000_0FFF	Debug ROM ( $\leq 2 \text{ KiB}$ )	
$0x0000_{-1000}$	0x01FF_FFFF	Reserved	
0x0200_0000	0x0200_FFFF	Core Complex-Local Inter-	on core complex Devices
		rupts (CLINT) (≤64 KiB)	(224 Mip)
$0x0201_{0000}$	0x0BFF_FFFF	Reserved	(224 1010)
$0x0C00_{-}0000$	0x0FFF_FFFF	Platform-Level Interrupt Con-	
		trol (PLIC) (64 MiB)	
0x1001_0000	0x1FFF_FFFF	Reserved	
0x2000_0000	0x2000_0FFF	UART Peripheral	
0x2000_1000	0x2000_1FFF	Reserved	
0x2000_2000	0x2000_2FFF	GPIO Peripheral (16 pins)	Off Care Complex address
0x2000_3000	0x2000_3FFF	Reserved	concerce complex address
0x2000_4000	0x2000_4FFF	Quad SPI Flash Control	and Paripharal busses
0x2000_5000	0x2000_5FFF	8-bit, 4-comparator PWM	and Peripheral busses
0x2000_3000	0x3FFF_FFFF	Reserved	
$0x4000_{-}0000$	0x5FFF_FFF	Memory Mapped Quad SPI	
		Interface	
0x6000_0000	0x7FFF_FFFF	Reserved	
0x8000_0000	0x8000_FFFF	Data Tightly Integrated Mem-	
		ory (DTIM) (64kB)	
0x8000_4000	0xFFFF_FFF	Reserved	

 Table 7.1: E31/E51 Core Complex FPGA Eval Kit Physical Memory Map



ψ

(5)

# Program FPGA with Vivado

- Download Vivado HLx Edition from Xilinx (free):
  - <u>https://www.xilinx.com/products/design-tools/vivado.html</u>
- From Vivado with the Arty board connected over USB
  - Open Hardware Manager
  - Tools Auto Connect
  - Tools Add Configuration Memory Device
  - Select Micron n25q128-3.3V
  - Select the Configuration File (mcs file downloaded from SiFive)
- Hit the PROG button on the Arty to reboot and load the new image



# Connecting the FPGA to the Debugger

- Debug signals are connected to the Arty Board's JD PMOD header (closest to the reset button)
- Tested Probes:
  - Olimex ARM-USB-Tiny (OpenOCD)
  - SEGGER JLINK
  - Lauterbach TRACE32
- JTAG Signal Mapping
  - Purple TDO Yellow TDI
  - Orange- nTRST Green TMS
  - Blue TCK Grey nRST
  - Black GND White GND
  - Brown VREF RED VREF



JD PMOD



ARM 20 Pin Header



ŝ

(5)

## Software Development with Freedom Studio

- Freedom Studio
  - Eclipse + CDT + GNU MCU Eclipse
  - Bundled Toolchain and OpenOCD Binaries
  - Examples for SiFive platforms

Image: Second control of the second control	<pre>Problems Console 23 Problems Console 24 P</pre>	Image: Second	Debug 33 Debugger Consol Debug 33 Debugger Consol Debug 34 Debugger Consol Debugger 24 Debugger Consol Debugger 24 Debugger 24 Debugger Debugger 24 Debugger 24 Debugger Debugger 24 Debugger 24 Debugger 24 Debugger Debugger 24 Debugger 24 Debug	bie         Pag Breakpoints           GDB Open-OCD Debugging]         :           : Breakpoint)         Iterrupts.c:110 0x40400714           Iterrupts.c:12 0x40400714         Iterrupts.c:12 0x40400714           : Breakpoint)         Iterrupts.c:12 0x40400714           : Bit 0x4040070         Iterrupts.c:12 0x40400714           : Bit 0x4040070         Iterrupts.c:12 0x40400714           : Bit 0x4040071         Iterrupts.c:12 0x40400714           : Bit 0x4040071         Iterrupts.c:12 0x40400714           : Bit 0x40400714         Iterrupts.c:12 0x40400714           : Bit 0x4040775         : Bit 0x40400714           : Bit 0x4040776         : Bit 0x40400714           : Bit 0x4040776         : Bit 0x4040776           : Bit 0x4040776         : Bit 0x40400776           : Bit 0x40400776         : Bit 0x40400	Couch Access
Peripherals         P           Description         General Purpose and           General Purpose and         P           Market State         State           Participation         Control           Control         State           Control         Control           Control         Control           Control         Control           Control         Control           Control         Control	Console 22 Problems GNU MCU Eclipse Packs console 288-01-07-2218-17 Persing SVD file "Visers/drew/wsFreedomStudio/global_interrupts/e3 Completed in 25ms. global_interrupts.c 23 @ start.5 @ init.c @ entry.5 [ gdd 22 33 write (STDDUT_FILENO, instructions_msg, strlen(instruction 35 } 36 37 wold involid_global_isr() { 38 printf("Unexpected global interrupt!\n"); 39 } 30 318 wold involid_local_isr() { 32 printf("Unexpected global interrupt!\n"); 33 } 318 wold involid_isr() { 32 printf("Switch 1 is on! Even if Switch 2 is on, Switch 1 r 35 } 319 // Ste Green LED 310 GPID_REC(GPID_OUTPUT_VAL) b= -(@xi< REEN_LED_OFFSET)); 311 GPID_REC(GPID_OUTPUT_VAL) b= -(@xi< REEN_LED_OFFSET)); 312 printf("Switch 2.Londler() { 313 } 314 Jone Content (Switch 2 is on [Switch 1 is on. Switch 2 is on, Switch 2 is on [Switch 1 is on, Switch 2 is on, S	Image:	Debug 23 Debugger Consol Colosal.Interrupts OpenOCD (6 Colosal.Interrupts of Debug 2007) Debug 2007 Debug 2	ble <sup>©</sup> <sub>0</sub> Breakpoints DBD DenOCD Debugging] :Breakpoint) terrupts.c:110 0x40400714 terrupts.c:12 0x40400716 :E2 0x4040070 :Si50 0x40400710 :E10 0x4040074 :Enter location here @ 10 8x500 0x10400510	i→ ▼ □ [
oory 23 □ □ oory 23 □ □ Adress Value 0×2000. 0×0000. 0×2000. 0×0000. 0×2000. 0×0000. 0×2000. 0×0000.	<pre>global_interrupts.c 32 @ start.S @ bitc @ entry.S [ [gdt 33 write (STDOUT_FILENO, instructions_msg, strlen(instruction 34 yr=void involid_global_isr() { 35 yr=void involid_global_isr() { 36 printf("Unexpected global interrupt!\n"); 37 yr=void involid_local_isr() { 38 printf("Unexpected local interrupt!\n"); 39 } 39 } 30 woid switch_local_isr() { 36 } 37 evoid switch_localer() { 36 } 37 evoid switch_localer() { 36 } 37 evoid switch_localer() { 37 evoid switch_localer() { 38 } 39 } 39 } 31 evoid switch_localer() { 39 } 31 evoid switch_localer() { 31 evoid switch_localer() { 32 } 33 } 33 evoid switch_localer() { 33 } 34 } 35 } 35 } 35 } 35 } 35 } 35 } 35 } 35</pre>	b[2]proc[42000]:threadGroup[1].gdb[2].proc[4 ns_msg)); must have higher priority right now.\n*	<pre></pre>	SISEO 0x40400510         B: Outline       ST Disassembly S2       ST Terminal         Image: State of the state	(0x1 << GREEN_LED_( (0x1 << GREEN_LED_( (0x1 << GREEN_LED_0) -((0x1 << RED_LED_0)
nory 23	<pre>global_interruptsc S2 @ start.S @ bitc @ entry.S @ [gdb 32 92 93 94 95 95 95 97 97 97 97 97 97 97 97 97 97 97 97 97</pre>	b[2]proc(42000].threadGroup[1],gdb[2]proc(4 ns_meg)); must have higher priority right now.\n*	<pre></pre>	Best Coutine         Set Disassembly Signet Terminal           Entre location here         ●           40400772:         jal 0x4400944 optintfs           9110         GPID.REGC(FDID.OUTPUT_VAL)  =           40400778:         lw of,12(sp)           115         }           40400776:         lw ra,12(sp)           116         GPID.REGC(FDID.OUTPUT_VAL)  =           40400776:         lw ra,12(sp)           118         GPID.REGC(FDID.OUTPUT_VAL)  =           40400876:         sw of,12(sp)           1110         GPID.REGC(FDID.OUTPUT_VAL)  =           40400886:         sw of,12(sp)           1113         debounce():           113         debounce():           1140         sw of,12(sp)           115         j           40400886:         add1           113         debounce():           114         sw of,12(sc)           115         j           116         printf("Switch 2 is on Even i           117         void switch_2.handler:           40400886:         add1           118         printf("Switch 2 is on Even i           118         printf("Switch 2 is on Even i           118         printf("S	(0x1 << GREEN_LED_( (0x1 << GREEN_LED_( (0x1 << GREEN_LED_( (0x1 << RED_LED_0)
oory 23 □ □ □	<pre>gobaLinterupts.c % start.s white @ entry.s [ (gdt gobaLinterupts.c % g start.s white @ entry.s [ (gdt gobaLinterupts.c % gobaListr() {     set of the start of the start</pre>	p[2]proc[42000]threadGroup[1],gdb[2]proc[4 ns_meg]); must have higher priority right now.\n*	42000).OSthread(1))	3         Coutine         Stassembly S2         Starminal           Inter location nere           40400772:         joil         0x4440934 oprintFs           9110         GP10.REG(CP10.0UTPUT_VAL)  -           40400774:         lw         a5.8620802           110         GP10.REG(CP10.0UTPUT_VAL)  -           40400775:         lw         c7.12(cs)           111         GP10.REG(CP10.0UTPUT_VAL)  -           40400806:         sm         c4.12(cs)           111         GP10.REG(CP10.0UTPUT_VAL) &=           40400806:         sm         c4.12(cs)           40400806:         sm         c4.12(cs)           111         GP10.REG(CP10.0UTPUT_VAL) &=           40400806:         sm         c4.12(cs)           113         debounce():         sm           404008080:         sm         debounce():           113         debounce():         sdebounce():           404008080:         idd1         sp.9.16           113         grintf("Smitch" 2 is onl Even 1           404008080:         idd1         sp.9.16           118         grintf("Smitch" 2 is onl Even 1           40400800:         idd1         sp.9.16           <	(0x1 << GREEN_LED_( (0x1 << GREEN_LED_( (0x1 << GREEN_LED_0) -((0x1 << RED_LED_0)
nory 22 □ □ Address Value 0×2000. 0×00000. 0×2000. 0×00000. 0×2000. 0×00000.	<pre>@ global.interrupts.c 23 @ start.s @ lot.c @ entry.s [ gpd global.interrupts.c 23 @ start.s @ lot.c @ entry.s [ gpd gr write (STDOUT_FILENO, instructions_msg, strlen(instruction s printf"Unexpected global interrupt!\n"); gr printf"Unexpected global interrupt!\n"); gr printf"Unexpected local interrupt!\n"); gr printf"Unexpected local interrupt!\n"); gr printf"Switch 1: son! Even if Switch 2 is on, Switch 1: gr printf"Switch 1: son! Even if Switch 2 is on, Switch 1: gr printf"Switch 1: son! Even if Switch 2 is on, Switch 1: gr printf"Switch 1: son! Even if Switch 2 is on, Switch 1: gr printf"Switch 1: son! Even if Switch 2 is on, Switch 1: gr printf"Switch 2: son! Even if Switch 1: gr printf"Switch 2: son! Even if Switch 1: gr printf"Switch 2: son! Even if Switch 1: son. Switch 2: gr printf"Switch 2: son! Even if Switch 1: son. Switch 2: gr printf"Switch 2: son! Even if Switch 1: son. Switch 2: gr printf"Switch 2: son! Even if Switch 1: son. Switch 2: gr printf"Switch 2: son! Even if Switch 1: son. Switch 2: gr printf"Switch 2: son! Even if Switch 1: son. Switch 2: gr printf"Switch 2: son! Even if Switch 1: son. Switch 2: gr printf"Switch 2: son! Even if Switch 1: son. Switch 2: gr printf"Switch 2: son! Even if Switch 1: son. Switch 2: gr printf"Switch 2: son! Even if Switch 1: son! Switch 2: gr printf"Switch 2: son! Even if Switch 1: son! Switch 2: gr printf"Switch 2: son! Even if Switch 1: son! Switch 2: gr printf"Switch 2: son! Even if Switch 1: son! Switch 2: gr printf"Switch 2: son! Even if Switch 2: gr printf"Switch 2: son! Even if Switch 2: gr printf"Switch 2: son! Even if Switch 2: gr printf"Switch 2</pre>	b(2)proc(42000)threadGroup(1)_gdb(2)proc(4 ns_msg)); must have higher priority right now.\n"	42000).05thread(1))  (1)	3         ⊕: Outline ﷺ Disasembly 33         ⊕ Terminal           44040076:           Terminal           40400076:           Terminal           40400076:           Terminal           40400076:           Terminal           40400080:           Terminal           40400080:           Terminal           Terminal           Terminal           40400080:           Terminal           Terminal           Terminal           Terminal           Terminal <td>(0x1 &lt;&lt; GREEN_LED_( (0x1 &lt;&lt; GREEN_LED_( (0x1 &lt;&lt; GREEN_LED_( (0x1 &lt;&lt; RED_LED_0)</td>	(0x1 << GREEN_LED_( (0x1 << GREEN_LED_( (0x1 << GREEN_LED_( (0x1 << RED_LED_0)
oory 33 □ □	<pre>write (STOUT_FILENO, instructions_msg, strlen(instruction     write (STOUT_FILENO, instructions_msg, strlen(instruction     printf("Unexpected global interrupt!\n");     p; }  100 void invalid_local_isrO {     isoprintf("Unexpected local interrupt!\n"); 101 102 103 104 105 105 105 105 105 105 105 105 105 105</pre>	ns_msg)); must have higher priority right now.\n*	.;	Image: Control of the second	<pre>E 10 35 00 13 14 (0x1 &lt;&lt; GREEN_LED_(</pre>
nory 23 □ □ → 1 Address Value 0×2000. 0×0000. 0×2000. 0×0000. 0×2000. 0×0000. 0×2000. 0×0000.	<pre>write (STDOUT_FILENO, instructions_msg, strlen(instruction ) ) ) ) ) ) ) ) ) ) ) ) ) ) ) ) ) ) )</pre>	ns_msg)); must have higher priority right now.\n" ≩	D;	Entre location here         Image: Control of the location here           4404077:         joil         0x4040924         0x1115.           44040776:         lw         0x50x20002         0x1115.           44040776:         lw         0x50x20002         0x1215.           115         }         40400776:         lw         rot	(0x1 << GREEN_LED_( (0x1 << GREEN_LED_( (0x1 << GREEN_LED_( (0x1 << RED_LED_0))
ory ⊠ □ □	<pre>mile (SIGOU_FILEN, Intruction_ms, strigh(Instruction ); }; }; mile (SIGOU_FILEN) [ seventh invoid invoid_global_isr() { seventh invoid invoid_local_isr() { seventh invoid invoid_local_isr() { seventh invoid invoid_isr() { seventh invoid invoid_seventh is on seventh is on, seventh is printf("Switch 1 is on! Even if Switch 2 is on, Switch 1 is printf("Switch 1 is on! Even if Switch 2 is on, Switch 1 is printf("Switch 1 is on! Even if Switch 2 is on, Switch 1 is printf("Switch 2 is on! Switch 2 is on, Switch 1 is printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is printformed(); printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is printformed(); printforme</pre>	must have higher priority right now.\n"	: ۲	40400772:         jol 0x4400934 oprintF>           9110         670.875(C70.0079UT.VA)  -           40400774:         lui a5,0x20002           115         }           40400774:         lwi a5,0x20002           115         }           40400776:         ori a4,012,025)           115         }           40400776:         ori a4,04,2           40400776:         ori a4,04,2           40400076:         ori a4,04,2           40400076:         ori a4,04,2           40400080:         ord a4,04,2           40400080:         ord a4,04,2           40400080:         ord a4,04,2           4040080:         ord a4,04,2           113         ord a5,0,5,16           113         ord a4,04,2.2.Andler:           114         ord a4040078: cdebounces           115         >           116         printf("Switch 2 is on Even i           117         ord a4040078: cdebounces           40400800:         ord a4,04,2.2.nonler:           118         printf("Switch 2 is on Even i           118         ord a5,04,04,04,04,04,04,04,04,04,04,04,04,04,	<pre>(0x1 &lt;&lt; GREEN_LED_( (0x1 &lt;&lt; GREEN_LED_( ~((0x1&lt;&lt; RED_LED_0) ~((0x1&lt;&lt; RED_LED_0) ); f Switch 1 is on, Switch</pre>
nory 23 □ □	<pre>3 } 35 378 void involid_global_isrc) { 378 void involid_global_isrc) { 378 void involid_local_isrc) { 378 void involid_local_isrc) { 379 void involid_local_isrc) { 379 void involid_local_isrc) { 381 void smitch_lhandler() { 382 void smitch_lhandler() { 383 void smitch_lhandler() { 384 void smitch_lhandler() { 385 void smitch_lhandler() { 385 void smitch_lhandler() { 386 void smitch_lhandler() { 386 void smitch_lhandler() { 387 void smitch_lhandler() { 388 void smitch_lhandler() { 388 void smitch_lhandler() { 389 void smitch_lha</pre>	must have higher priority right now.∖n" ≩	°;	p110         GP10_REGC#P10_OUTPUT_VAL)  -           40400774:         lui         a5,0x20002           40400775:         lw         a6,12(c5)           115         }         a40400776:           116         GP10_REGC#ID_OUTPUT_VAL)  -           40400776:         lw         ra,12(c5)           118         GP10_REGC#ID_OUTPUT_VAL)  -           40400802:         lw         ra,12(c5)           40400802:         lw         c4,12(c5)           40400808:         andi         c4,c4,2           40400808:         andi         s,c4,12(c5)           115         J         debounce();           40400808:         addi         s,p,sp,16           113         debounce();         idebounce();           void switch_2_handler:         addi sp,sp,16           118         cmit(C_Stitch 2 is and Even if           40400808:         addi sp,sp,16           113         addi sp,sp,16           114         adjr c0,db,2(a); a is and Even if           40400800:         adjr c0,db,2(a); a is and Even if	<pre>(0x1 &lt;&lt; GREEN_LED_( (0x1 &lt;&lt; GREEN_LED_( ~((0x1&lt;&lt; RED_LED_O))))))))))))))))))))))))))))))))))))</pre>
nory 33	<pre>&gt; '  '  '  '  '  '  '  '  '  '  '  '  '</pre>	must have higher priority right now.∿n" ≌	";	<pre>b 404007f4; lut a5,0x2002 404007f4; lw af,12(c5) 115 } 404007fc lw rc,12(c5) 116 GFL0,REG(CFL0,DUPUT,VAL)  = 404007ex; w af,12(c5) 111 GFL0,REG(CFL0,DUPUT,VAL) = 40400808; and af,c4,c2 40400808; lw af,12(c5) 115 } 40400808; and af,c4,c2 40400808; sw af,12(c5) 115 } 40400808; dbsunce(); 40400808; and af,c4,c2 40400808; sw af,12(c5) 115 } 40400808; and af,c4,c2 40400808; sw af,12(c5) 115 } 40400808; and af,c4,c2 40400808; and af,c4,c2 40400808; and af,c4,c2 40400808; and af,c4,c2 4040080; and af,c4,c2 40400808; and af,c4,c2 4040080; and af,c4,c2 40400; and af,c4,c2 4040; and</pre>	(0x1 << GREEN_LED_( ~((0x1<< RED_LED_0) (5 Switch 1 is on, Sv
nory X □ □	<pre>&gt;</pre>	must have higher priority right nom.∖n" §	");	40400778:         1w         o4,12(c5)           115         }           4040076:         1w         rc,12(sp)           115         >            116         GP10.REG(CF10.0UTPUT_VAL) !=            40400806:         sw         c4,12(c5)           111         GP10.REG(CF10.0UTPUT_VAL) &=           40400806:         sw         c4,12(c5)           40400808:         sw         c4,12(c5)           115         >            126         >         debounceC):           40400808:         adv1,12(c5)           135         >           143         debounceC):           vid avid.L2.handler:            40400808:         add1           113         debounceC):           40400808:         add1           113         adbounceC):           40408080:         add1           114         add1           115         >           116         add1           117         void avid.L2.handler:           40408080:         adir.p.g.g.hand1           118         adir.p.g.g.hand2           118         adir.p.g.g.	(0x1 << GREEN_LED_( ~((0x1<< RED_LED_OF
hory 23 □ □ Address Value 0x2000. 0x0000. 0x20000. 0x0000. 0x20000. 0x0000. 0x20000. 0x0000. 0x2000. 0x0000.	<pre>printf("Switch 2 is on [Switch 1 is on. Switch 2 is printf("Switch 2 is on [Switch 1 is on. Switch 2 is printf("Switch 2 is on [Switch 1 is on. Switch 2 is printf("Switch 1 is on [Switch 2 is on, Switch 1 if printf("Switch 1 is on [Switch 2 is on, Switch 1 if printf("Switch 2 is on [Switch 2 is on, Switch 1 if printf("Switch 2 is on [Switch 2 is on, Switch 1 if printf("Switch 2 is on [Switch 2 is on, Switch 2 is printf("Switch 2 is on [Switch 2 is on, Switch 2 is printf("Switch 2 is on [Switch 1 is on, Switch 2 is printf("Switch 2 is on [Switch 1 is on, Switch 2 is printf("Switch 2 is on [Switch 1 is on, Switch 2 is printf("Switch 2 is on [Switch 1 is on, Switch 2 is printf("Switch 2 is on [Switch 1 is on, Switch 2 is on]</pre>	must have higher priority right nom.∖n*	°;	115 } 404007c; lw ra,12(sp) 110 GPL0.REG(GPL0.OUTPUT.VAL)  = 4040087c; ori d, d, d, d, 40400808; sw ad, 12(c5) 111 GPL0.REG(GPL0.OUTPUT.VAL) &= 40400808c; lw ad, 12(c5) 115 } 40400808c; addi sp,sp,16 113 debounce() 115 debounce() 116 py,16 118 debounce() 117 void smitch_landler: 40400808c; addi sp,sp,16 118 py,16 118 py,16 119 py,16 118 py,16	(0x1 << GREEN_LED_( ~((0x1<< RED_LED_O) );f Switch 1 is on, Se
nory 23 □ □	<pre>9 primer Understand ground interfact of y 9 primer Understand ground interfact of y 10 void invalid_local_isrC { 10 printf("Unsequected local interrupt!\n"); 10 } 10 void switch_l_handler() { 10 printf("Switch 1 is on! Even if Switch 2 is on, Switch 1 n 10 printf("Switch 1 is on! Even if Switch 2 is on, Switch 1 n 9/10 [GTO_REG(GPI0_OUTPUT_VAL) = (0x1 &lt;&lt; GREA_LED_OFFSET); 11 GFI0_REG(GPI0_OUTPUT_VAL) = ((0x1 &lt;&lt; GREA_LED_OFFSET); 12 debounce(); 13 debounce(); 13 printf("Switch_l_handler() { 13 printf("Switch_lis on! Even if Switch 1 is on. Switch 2 is on!</pre>	must have higher priority right nom.∖n" §	<b>D</b> i	404007/a:         lw ra.12(sp)           110         GPIO.REGCFU.OUTPUT.VAL) I=           404007/c:         ori a4,04,2           40400806;         sw a4.12(a5)           111         GPIO.REGCFU.OUTPUT.VAL) &=           40400806;         sw a4.12(a5)           40400808;         lw a4.12(a5)           40400808;         sw a4.12(a5)           113         debounce(c);           40400808;         addi sp.sp.16           113         sw a4.147(Swith 2 is and Ewen (if a4.147)           40400808;         addi sp.sp.16           118         printf(Swith 2 is and Ewen (if a4.147)           4040080;         adjs cm (AA.2164)           404080;         adjs cm (AA.2164)	(0x1 << GREEN_LED_( ~((0x1<< RED_LED_OF
nory 23 □ □ → ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓	<pre>inf ' isis wold involid.locel_isr() { isis wold involid.locel_isr() { isis wold involid.locel_isr() { isis } isis vold switch_l_handler() { isis printf("Switch 1 is on! Even if Switch 2 is on, Switch 1 i isis // Set Green LED isis GPIO.REG(GPIO.UPPUT_VAL) = (Bet &lt;&lt; GREEN_LED_OFFSET); isis GPIO.REG(GPIO.UPPUT_VAL) = (Get &lt;&lt; GREEN_LED_OFFSET); isis debounce(); is printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is </pre>	must have higher priority right now.∖n*		110         GPD.REGCGP10_OUTPUT_VAL)  -           404007C: ori d, d, d, d,         -           40400800: sw d, 12(c5)         -           111         GPD.REGCGP10_OUTPUT_VAL) &-           40400802: lw d, 12(c5)         -           40400808: addi sp, sp, 16         -           113         debounces           40400808: addi sp, sp, 16         -           113         debounces           114         debounces           115         -           116         switch_2.handler:           40400808: addi sp, sp, 16           113         debounces           114         debounces           115         -           116         printf("Switch 2 is on Even i           40400808: addi sp, sp, 16         -           118         calif addi she, dage addi se addi s	<pre>(0x1 &lt;&lt; GREEN_LED_( ~((0x1&lt;&lt; RED_LED_O))))))))))))))))))))))))))))))))))))</pre>
nory 23 □ □ → de 18 29 18 • ▼ Adress Value 0x2000. 0x0000. 0x2000. 0x0000. 0x2000. 0x0000. 0x2000. 0x0000.	<pre>init world in JoseL.isrC {     int world intervent \n");     int ("Unexpected local intervent \n");     int {         int ("Unexpected local intervent \n");     int {         int intervent \n", in</pre>	must have higher priority right now.∖n" ≩	; ;	404007fc:         ori         ad, c4, 2           44040807         sm         ad, 12(c5)           111         GP10_REG(GP10_OUTPU_VAL) &-           440408082         lw         ad, 12(c5)           440408083         addi sp, sp, 16           113         deboneCQ):         deboneCQ):           440408083         addi sp, sp, 16           113         deboneCQ):         deboneCQ):           40400808:         jm         0x/4400788: debounceS):           113         debounceQ):         debounceQ):           40400808:         addi sp, sp, 16           113         ge1, sp, -16           114         addi sp, sp, -16           118         grintf("Switch Z is and Even i           40400805:         addi sp, sp, -16           118         grintf("Switch Z is and Even i           4040805:         addi sp, sp, -16           118         grintf("Switch Z is and Even i           4040805:         addi sp, sp, -16           118         grintf("Switch Z is and Even i           4040805:         addi sp, sp, -16           118         grintf("Switch Z is and Even i	~((0x1<< RED_LED_OF
nory 22 □ □ → 1 □ 00 22 → 10 Address Value 0×2000. 0×00000. 0×2000. 0×00000. 0×2000. 0×00000. 0×2000. 0×00000.	<pre>is:</pre>	must have higher priority right now.\n"		40400800:         sw         od,12(c5)           111         GPIO.REG(CPIO.UNTPUT.VAL) &-           40400802:         lw         od,12(c5)           40400808:         andi         od,od,-2           40400808:         addi         sp,sp,16           113         debounce):         0x40408768           114         debounce):         0x40408768           117         void switch.2.handler:           40400808:         addi         sp,sp,16           118         printf("Switch 2 is of leven i           40400808:         addi         sp,sp,-16           118         cadi;         sp,do,do,2)         a40027	~((0x1<< RED_LED_Of
nory 33 □ → 00 33 → Address Value 0x2000. 0x0000. 0x2000. 0x0000. 0x2000. 0x0000. 0x2000. 0x0000.	<pre>183 }</pre>	must have higher priority right now.\n"	Di .	111         GPID_REG(GPID_OUTPUT_VAL) &-           440400802:         lw of J2(2G5)           440400802:         lw of J2(2G5)           1000081:         oddi sp. sp. 16           113         debounceC):           40400808:         jw oddi sp. sp. 16           113         debounceC):           40400808:         jw oddi sp. sp. 16           113         debounceC):           40400808:         jw odd switch_2_hondler:           40400808:         jw odd switch_2_hondler:           4040808:         iddi sp. sp16           118         printf("Switch 2 is onl Even i           4040808:         outps: dbd2_gbd2_gbd2_gbd2_gbd2_gbd2_gbd2_gbd2_	-((@x1<< RED_LED_OF
ory 33 ↓ ↓ ory 33 ↓ ↓ ory 34 ↓ ↓ ory 30 ↓ ↓ ory 3	<pre>104 ** 105@void switch_Lhondler() { 106@void switch_Lhondler() { 107 108 109 109 109 109 109 109 109 100 100 100</pre>	must have higher priority right now.∿n" ≣	י;	<ul> <li>94000002: lw a4,12(c5) 40400006: andi a4,c4,-2 40400006: sw a4,12(c5) 115 } 116 } 113 debounce() 40400008: j 0x40400768 </li> <li>40400080: j 0x40400768 </li> <li>40400080: addi sp,sp,16 awitch_2handler: 40400800: addi sp,sp,16 on Even ( 118 printf('Smith 2 is on Even ( 40400800: adjg a 0x40007)</li> </ul>	if Switch 1 is on, Sw
nory 23 □ □ → ota 15 00 10 00000 0x2000 0x0000 0x2000 0x0000 0x2000 0x0000 0x2000 0x0000 0x2000 0x0000	<pre>IDES wold switch_lhandler() { IDES wold switch_lhandler() { IDES printf("Switch 1 is on! Even if Switch 2 is on, Switch 1 n IDES // Set Green LED IDES IDES (GPL0_DUTPUT_VAL) = (%x1 &lt;&lt; GREN_LED_OFFSET) IDES IDES (GPL0_DUTPUT_VAL) = (%x1 &lt;&lt; GREN_LED_OFFSET); IDES debunce(); IDES } IDES IDES IDES IDES IDES IDES IDES IDES</pre>	must have higher priority right now.\n"	"); 	494080894:         andi         a, c4, -2           49408086:         sw         a, 12(c5)           115         }         ddi         sp.sp. f6           49108080:         j         Box (4400786 sector);           49408080:         addi         sp.sp16           118         printf("Switch 2 is onl Even i           49408080:         aujer do Age (Age 2 and	if Switch 1 is on, Sv
nory ⊠ □ □ → ct □ 00 3 → 1 Address Value 0x2000 0x0000 0x2000 0x0000 0x2000 0x0000 0x2000 0x0000	<pre>106 107 printf("Switch 1 is on! Even if Switch 2 is on, Switch 1 i 108 109 // Set Green LED 100 GPI0_REG(GPI0_OUTPUT_VAL) = (0x1 &lt;&lt; GREEN_LED_OFFSET) 111 GPI0_REG(GPI0_OUTPUT_VAL) &amp;= ~((0x1 &lt;&lt; GREEN_LED_OFFSET)); 112 113 debounce(); 113 115 } 115 J 116 printf("Switch_2_hondler() { 117 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is on! Even if Switch 1 is on. Switch 2 is on! Even if Switch 1 is on. Switch 2 is on! Even if Switch 1 is on. Switch 2 is on! Even if Switch 1 is on. Switch 2 is on! Even if Switch 1 is on. Switch 2 is on! Even if Switch 1 is on. Switch 2 is on! Even if Switch 1 is on. Switch 2 is on! Even if Switch 1 is on. Switch 2 is on! Even if Switch 1 is on. Switch 2 is on! Even if Switch 1 is on. Switch 2 is on! Even if Switch 1 is on. Switch 2 is on! Even if Switch 1 is on. Switch 2 is on! Even if Switch 1 is on. Switch 2 is on! Even if Switch 1 is on. Switch 2 is on! Even if Switch 1 is on. Switch 2 is on! Even if Switch 1 is on. Switch 2 is on! Even if Switch 1 is on! Even if Switch 1 is on! Even if Switch 2 is</pre>	must have higher priority right now.∖n" ≣	°);	40400006: sw ad,12(c5)           115         ,           115         ,           116         ,           117         ,           04040000:         ,           04040000:         ,           04040000:         ,           04040000:         ,           118         ,           04040000:         ,           117         ,           04040000:         ,           118         ,           118         ,           118         ,           118         ,           118         ,           04000000:         ,           118         ,           118         ,           118         ,           04000000:         ,           0400000:         ,           0400000:         ,           04000000:         ,           04000000:         ,           04000000:         ,           04000000:         ,           04000000:         ,           04000000:         ,           04000000:         ,           04000000:         ,	if Switch 1 is on, Sw
nory ☆	<pre>INF printf("Switch 1 is on! Even if Switch 2 is on, Switch 1 n 108 1// Set Green LED 109 1// Set Green LED 101 GPIO_REG(GPIO_OUTPUT_VAL) = (&amp;x1 &lt;&lt; GREEN_LED_OFFSET) 111 GPIO_REG(GPIO_OUTPUT_VAL) &amp;= -((@x1&lt;&lt;&lt; RED_LED_OFFSET)); 112 113 debounce(); 114 115 } 115 116 117 = Void switch_2_handler() { 115 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 12 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 13 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 14 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 15 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 16 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 17 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 18 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 18 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 19 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 10 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 10 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 10 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 10 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 10 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 11 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 12 printf("Switch 2 is on! Even if Switch 1 is on! Even if Switch 2 is on! Even if Switc</pre>	must have higher priority right now.∖n" ≣	;"	115         }           113         debuoncC);           40408080:         0.404400768 cdebounces;           113         woid switch_2_hondler;           40408080:         0.404400768 cdebounces;           117         switch_2_hondler;           40408080:         addit sp.sp.156           118         printf("switch_2 is on Even i           40408080:         addit sp.sp.156           118         printf("switch_2 is on Even i           40408080:         addit sp.sp.166	if Switch 1 is on, Sw
Image: state	<pre>108 109 1// Set Green LED 2010 [GPI0.REG(GPI0.OUTPUT_VAL)  + (0x1 &lt;&lt; GREEN_LED_OFFSET) 111 GPI0.REG(GPI0.OUTPUT_VAL) &amp;= -((0x1 &lt;&lt; RED_LED_OFFSET)); 112 113 debounce(); 114 115 115 115 115 115 115 115 115 115</pre>			00000008:         addi         sp.sp.16           113         debounce):         b&d400768         debounce):           104040080a:         j         b&d400768         debounce):           117         void switch_2.handler:         debounce):         debounce):           118         existent_2.handler:         debounce):         debounce):	if Switch 1 is on, Sw
With III         With IIII         With IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII	<pre>109  // Set Green LED 109  101 GFD1REC(GPD_OUTPUT_VAL) =- (0x1 &lt;&lt; GREN_LED_OFFSET) 111 GFD1REG(GPD_OUTPUT_VAL) &amp;= ~((0x1 &lt;&lt; RED_LED_OFFSET)); 112 113 debounce(); 114 115 } 115 116 117 @void switch_2_handler() { 118 printf("Switch_2 is on [Switch 1 is on. Switch 2 is on [Switch_2 i</pre>	1		113	if Switch 1 is on, Sv
>>1           Address         Value           0x2000         0x0000           0x2000         0x0000           0x2000         0x0000           0x2000         0x0000           0x2000         0x0000           0x2000         0x0000	<pre>pile GPID_REGCGPID_OUTPUT_VAL) = (Bx1 &lt;&lt; GREEN_LED_OFFSET) ili GPID_REGCGPID_OUTPUT_VAL) &amp; ~((Bx1&lt;&lt; RED_LED_OFFSET)); ili debounce(); ili debounce(); ili } ili debounce() { ili printf("Switch 2_hondler() { ili printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is on! Even if Switch 1 is on. Switch 2 is on!</pre>			Improvement         j         excense of columnes           117         void switch_2_handler() {         switch_2_handler() {           4040080c:         addi sp.p.16         ll           118         printf("Switch 2 is on! Even i           4040080c:         aujpc 00,0x2           4040080c:         aujpc 00,0x2	if Switch 1 is on, Sv
**************************************	<pre>111 GPI2.REG(GPI0_OUTPUT_VAL) &amp; ~((%x1&lt;&lt; RED_LED_OFFSET)); 112 113 debounce(); 114 115 } 116 117 wold switch_2_handler() { 118 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 128 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 139 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 140 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 140 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 140 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 140 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 140 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 140 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 140 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 140 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 140 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 140 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 140 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 140 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 140 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 140 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 140 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 140 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 140 printf("Switch 2 is on! Even if Switch 2 is on! Even if Switch 2 is 140 printf("Switch 2 is on! Even if Switch 2 is on! Even if Switch 2 is 140 printf("Switch 2 is on! Even if Switch 2 is on! Ev</pre>				if Switch 1 is on, Sw
Address         Value           0x2000         0x2000           0x2000         0x0000           0x2000         0x0000           0x2000         0x0000           0x2000         0x0000	112 113 debounce(); 114 115 } 115 printf(Switch_2.handler() { 116 printf(Switch 2 is on! Even if Switch 1 is on. Switch 2 ;			4040080c: addi sp.sp16 118 printf("Switch 2 is on! Even i 4040080e: auipc a0,0x2 4040080e: auipc a0,0x2	if Switch 1 is on, Sw
Address         Value           0x2000         0x2000           0x2000         0x0000           0x2000         0x0000           0x2000         0x0000           0x2000         0x0000	<pre>113 debounce(); 114 115 } 116 1179 void switch_2_handler() { 118 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 / 118 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 / 118 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 / 118 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 / 118 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 / 119 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 / 119 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 / 119 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 / 119 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 / 119 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 / 119 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 / 119 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 / 119 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 / 119 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 / 119 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 / 119 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 / 119 printf("Switch 2 / 119 pr</pre>			118 printf("Switch 2 is on! Even i 4040080e: auipc a0,0x2 40400812: addi a0,0x2	if Switch 1 is on, Sw
0x2000 0x2000 0x0000 0x2000 0x0000 0x2000 0x0000 0x2000 0x0000	<pre>114 115 } 116 1170 void switch_2_handler() { 118 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 118 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 118 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 118 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 118 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 119 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 119 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 119 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 119 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 119 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 119 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 119 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 119 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 119 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 119 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 119 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 119 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 119 printf("Switch 2 is on! Even if Switch 2 is on! Even if Switch 2 is 119 printf("Switch 2 is on! Even if Switch 2</pre>			4040080e: auipc a0,0x2	-0
0x2000 0x0000 0x2000 0x0000 0x2000 0x0000 0x2000 0x0000	<pre>115 3 116 1170 void switch_2_handler() { 118 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is </pre>			40400012; oddi 00 00 -1062 # 0-404022	
0x2000 0x0000 0x2000 0x0000 0x2000 0x0000	<pre>11b 117@ void switch_2_handler() { 118 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 118 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 118 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 118 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 118 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 118 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 118 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 118 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 118 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 118 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 118 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 118 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 118 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 118 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 118 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 118 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 118 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 118 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 118 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 is 118 printf("Switch 2 is on! Even if Switch 2 i</pre>			+0+00012: 0001 00,00,-1002 # 0X404023	000
0x2000 0x0000 0x2000 0x0000	<pre>11/@ void switch_2_nandler() { 118 printf("Switch 2 is on! Even if Switch 1 is on. Switch 2 n </pre>			117 void switch_2_handler() {	
0x2000 0x0000	110 primit Switch 2 is on! Even if Switch 1 is on. Switch 2 r	much hous bishes estenibu sisht	·	40400816; sw ra,12(sp)	
	110	must have higher priority right now. Vn"	25	118 printf("Switch 2 is on! Even i	if Switch 1 is on, Sw
[0] 0x0	120 // Set RED LED			40400818: jal 0x40400940 <printf></printf>	
[1] 0x1	121 GPTO REG(GPTO OUTPUT VAL) &= ~(0x1 cc (PEEN LED DEESET)			121 GPIO_REG(GPIO_OUTPUT_VAL) &=	~(0x1 << GREEN_LED_
[2] 0x0	122 GPIO REG(GPIO OUTPUT VAL) = (0x1<< RED   FD OFFSET)	,		4040081a: lui a5,0x20002	
[3] 0x0	123			4040081e: lw a4,12(a5)	
[4] 0×0	124 debounce();			125 }	
(5) 040	125 }			40400820: lw ra,12(sp)	40.4
[0] 0X0	126			121 GPIO_REG(GPIO_OUTPUT_VAL) &=	~(0x1 << GREEN_LED_
[6] UXU	127⊕ // We use PWM 0 as a			40400822: andi a4,a4,-3	
[7] 0x0	128 // timer interrupt for debouncing.			= 40400824: SW 04,12(05)	(A.1 DED. LED. 077
[8] 0x0	129			122 GPI0_KEG(GPI0_00TPUT_VAL) I=	(WX1<< KED_LED_OFF
[9] 0x0	130 void pwm_0_handler() {			40400826: LW 04,12(05)	
[10] 0x0	131			40400028: 0F1 04,04,1 4040002c; sw o4 12(05)	
[11] 0x0	132			125 3	
[12] 0x0	133 1f (g_debounce == 0) {			4840882e: addi sp.sp.16	
[13] 0x0	<pre>134 printf(" Done debouncing.\n");</pre>			= 124 debounce():	
[14] 0x0	135			4848838: j 9x494997b8 <debounce></debounce>	
[15] 0x0	136 //Lower the threshold s.t. the switches can hit.			atexit:	
[16] 0v0	<pre>137 PLIC_set_threshold(&amp;g_plic, 1);</pre>			40400832: mv a1,a0	
(10) 000	138 (/ Class the DMM intermed			40400834; li a3,0	
[17] 0x0	139 // Clear the PWM Interrupt			40400836; li a2,0	
[18] 0x0	140 FMM0_KEG(PMM_CFG) = 0;			40400838; li a0,0	
[19] 0x0	141 142 Lelse J			4040083a: j 0x40400990 <register.< td=""><td>_exitproc&gt;</td></register.<>	_exitproc>
[20] 0x0	142 // Keen waiting			exit:	
[21] 0x0	145 // weep watering				
terre te e t				A	
-					
	[4]         0x0           [5]         0x0           [6]         0x0           [8]         0x0           [9]         0x0           [10]         0x0           [11]         0x0           [12]         0x0           [13]         0x0           [14]         0x0           [15]         0x0           [16]         0x0           [17]         0x0           [18]         0x0           [19]         0x0           [20]         0x0           [21]         0x0	[4]       0x0       123       125       126         [5]       0x0       127       // Euse PMM 0 as a       127       127       // Euse PMM 0 as a       127       127       // Euse PMM 0 as a       128       127       // Euse PMM 0 as a       128       129       128       127       // Euse PMM 0 as a       129       129       129       120       128       129       129       129       120       129       120       120       120       120       120       120       120       120       120       120       120       123       129       128       129       129       129       129       129       129       129       120       120       120       120       120       120       120       120       120       120       120       120       12	[4]       0x0       125       debounce();         [5]       0x0       127       // w use PMM 0 as a         [7]       0x0       128       127         [8]       0x0       128       128         [9]       0x0       128       128         [11]       0x0       131       131         [12]       0x0       133       print(*       Dana Ashbancing.\n");         [13]       0x0       134       print(*       Dana Ashbancing.\n");         [14]       0x0       135       // Lower the threshold s.t. the writches can hit.         [17]       0x0       138       // Clase the PMM interrupt         [18]       0x0       143       // Clase the PMM interrupt         [19]       0x0       144       // Keep woiting	[4]       0x0       124       decontect;         [5]       0x0       125       ////////////////////////////////////	[4]       000       124       det0000000000000000000000000000000000

🗲 SiFive

### Freedom Studio – Set Up



- Download and Extract to desired location
  - <u>https://www.sifive.com/products/tools/</u>
- If on Windows, install platform drivers located in:

ିଙ୍

🗲 SiFive

(5)

- FreedomStudio/SiFive/Drivers
- That's It!



#### Freedom Studio - Demo

- Importing Examples
- Freedom Studio SiFive Perspective
- Programming and Debugging the E31 Arty FPGA Platform
  - Dhrystone Demo use the debugger to change the number of iterations

ŝ

🗲 SiFive

໌ ສີ



#### :: •••• •••



# Questions

#### info@sifive.com

© 2018 SiFive. All Rights Reserved.



# 3 Part Webinar Series

Webinar Recordings: <a href="https://info.sifive.com/risc-v-webinar">https://info.sifive.com/risc-v-webinar</a>

- RISC-V 101
  - The Fundamentals of RISC-V architecture

- Introduction to SiFive RISC-V Core IP
  - Introduction to the SiFive E31 and E51 Core Complexes
- Getting Started with SiFive RISC-V Core IP



#### Resources

- <u>https://www.sifive.com/</u>
  - RISC-V Core IP and Development Boards
  - RISC-V Tools
  - SiFive Youtube Channel:
    - <u>https://www.youtube.com/channel/UCqpdhncf4nxTfy0QZh1YWLQ</u>
- <u>https://riscv.org/</u>
  - RISC-V Specifications
  - Links to the RISC-V mailing lists
  - Workshop proceedings
- GitHub
  - https://github.com/sifive/
  - https://github.com/riscv/





RISC-V

🗲 SiFive

ŝ

( 5)

