

Hardware Manual

Version 1.0.0.1

June 15, 2007

Atmark Techno, Inc. http://www.atmark-techno.com/

Table of Contents

1.		action ····· tions ·····	
2.		Safety Precautions	
	2.1. 2.2.	Operational Precautions	•• 2
		Operational Precautions	·· 2
	2.3. 2.4.	Precautions When Using FPGA Software Precautions	3
0			
3.		Getting Started	
	3.1.	Required Items	
4.		ew	
	4.1.	SUZAKU-V Features	
	4.2.	Specifications	··6
	4.3.	Block Diagram	7
	4.4.	Functions	
	4.4.1.	Processor ·····	
	4.4.2.	Bus ·····	
	4.4.3.	Memory ·····	
	4.4.4.	Interrupts	
	4.4.5.	Timer	
	4.4.6.	Serial Console	
	4.4.7.	LAN	•
	4.4.8.	External I/O ·····	
	4.4.9.	FPGA Configuration	
	4.4.10.	Software Reset	
	4.4.11.	JTAG	
	4.4.12.	Setting Jumpers	13
	4.4.13.	LED	
	4.4.14.	Power Input +3.3V ·····	13
	4.4.15.	+3.3V Internal Logic Power Output	14
	4.4.16.	Internal Power Sequence	14
5.	Memor	y Map ·····	15
	5.1.	SUZAKU-V Memory Map	15
6.	FPGA	Pin Assignment	16
7.	Interfa	ce Specifications	21
	7.1.	Interface Layout	21
	7.2.	CON2 External I/O, FPGA Programming JTAG Connector	
	7.3.	CON3 External I/O Connector	23
	7.4.	CON4 External I/O Connector ······	
	7.5.	CON5 External I/O Connector ·····	24
	7.6.	CON7 FPGA JTAG Connector ·····	25
	7.7.	CON1 RS232C Connector ·····	25
	7.8.	JP1 Boot Mode Jumper	26
	7.9.	JP2 FPGA Programming Jumper	26
	7.10.	D3 Power-on LED	27
	7.11.	D1 User-Controllable LED	
	7.12.	CON6 Power Input +3.3V Connector ······	
	7.13.	Ethernet 10/100 Base-T	28
8.		View	
9.		ng a FPGA Project (Example: Adding a UART) ······	
	9.1.	Extracting the fpga_proj Project from the CD-ROM	30
	9.2.	Creating a New Project	31
	9.3.	Setting up Platform Studio	35
	9.4.	Platform Studio Compiling	

9.5.	Editing the	Project Top Fi	le·····	•••••••••••••••••••••••••••••••••••••••		43
9.6.	Project Nav	igator Compili	ing	••••••••••••••••••••••••••••••		
10. FI	PGA Configurat	tion ·····	-			49
10.1.	Converting	a BIT File to a	MCS File	•••••••••••••••••••••••••••••••••••••••		49
10.2.	Programmi	ng Configurati	on Data to SUZAKU-V	/		54
10.3.	Dealing	With	Lbplay2.EXE	"ERROR :	Please	check
WINN	T\system32\d	rivers\windry	r.sys."	••••••		

List of Tables

Table 4-1 SUZAKU-V Specifications 6
Table 5-1 SUZAKU-V Memory Map 15
Table 5-2 Memory Map of 8MByte Flash Memory (0xF000 0000-0xF07F FFFF) ······15
Table 6-1 FPGA Pin Assignment – External I/O (1/2) 16
Table 6-2 FPGA Pin Assignment – External I/O (1/2) 17
Table 6-3 FPGA Pin Assignment – Internal Devices (1/2) 18
Table 6-4 FPGA Pin Assignment – Internal Devices (2/2)19
Table 6-5 FPGA Pin Assignment – JTAG and Configuration 20
Table 7-1 Interface Details 21
Table 7-2 CON2 Connector for External I/O or FPGA Programming22
Table 7-3 External I/O Connector 23
Table 7-4 CON4 External I/O Connector 24
Table 7-5 CON5 External I/O Connector 24
Table 7-6 CON7 Virtex-II Pro JTAG Connector 25
Table 7-7 CON1 RS232C Connector 25
Table 7-8 JP1 Boot Mode Jumper 26
Table 7-9 JP2 FPGA Programming Jumper 26
Table 7-10 D1 User-Controllable LED ······27
Table 7-11 CON6 Power Input +3.3V Connector 27
Table 7-12 Ethernet 10/100 Base-T ······28

List of Figures

Figure 4-1 Block Diagram of SUZAKU-V	7
Figure 4-2 SUZAKU-V Bus Configuration	
Figure 4-3 FPGA Configuration	
Figure 7-1 Layout of Interfaces	
Figure 8-1 SUZAKU-V Board View	29

1. Introduction

Thank you for your purchase of the SUZAKU-V.

This manual introduces the hardware specifications of the SUZAKU-V and also provides information on how to make use of the SUZAKU-V.

We hope the information contained in this document will help you get the best functionality out of the SUZAKU-V.

1

2. Precautions

2.1. Safety Precautions

Please read the following safety precautions carefully to assure correct use of the SUZAKU-V.

This product uses semiconductor components designed for generic electronics equipment such as office automation equipment, communications equipment, measurement equipment and machine tools. Do not incorporate the product into devices such as medical equipment, traffic control systems, combustion control systems, safety equipment, etc. which can directly threaten human life or pose a hazard to the body or property due to malfunction or failure. Moreover, products incorporating semiconductor components can be caused to malfunction or fail due to foreign noise or surge. To ensure there will be no risk to life, the body or property even in the event of malfunction or failure, be sure to take all possible measures in the safety system design, such as using protection circuits like limit switches or fuse breakers, or system multiplexing.

2.2. Operational Precautions

To avoid degradation, damage, malfunction, or fire, the following safety precautions must be observed when handing the product.

Input Power

Do not attempt to apply a voltage higher than 3.3V+5%. Use caution in polarity.

• Interface

Do not attempt to connect a signal other than that specified to each interface (external I/O, RS232C, Ethernet and JTAG). Use caution in the polarity of signals. Take care of the input and output direction of signals.

Modification

Do not make any modifications other than installing additional connectors on the external I/O and JTAG connectors (CON2, CON3, CON4, CON5, CON7).

• FPGA Programming

Be careful not to program the FPGA in a way that can cause a collision between peripheral circuitry (including on-board components) and a signal (i.e. output of the same signal from two devices). Use caution when programming the FPGA.

Power-on

Do not attempt to connect or disconnect FPGA I/O or JTAG connectors while power is supplied to the board or peripheral circuits.

• Static Electricity

This board uses CMOS devices. Store it safely in the antistatic package provided at shipment while it is not being used.



• Latch-up

Due to excessive noise or a surge from the power supply or input/output, or sharp voltage fluctuations, the CMOS devices incorporated in the board can cause a latch-up. Once a latch-up occurs, this situation continues until the power supply is disconnected and thus can damage the device. It is recommended to take safety measures such as adding a protection circuit to the noise-susceptible input/output line or not sharing a power supply with devices that can be the cause of noise.

• Shock and Vibration

Guard against strong impact such as a drop or collision. Do not put this product on anything vibrating or rotating. Guard against strong vibration or centrifugal force.

• **High/Low Temperatures and High Humidity** Do not use the board in areas exposed to very high/low temperatures or high humidity.

• Dust

Do not use the board in dusty areas.

2.3. Precautions When Using FPGA

• FPGA Project Contained in This Product

The FPGA project and documentation contained in this product are provided "AS IS" without warranty of any kind including any warranty of merchantability or fitness for a particular purpose, reliability, correctness or accuracy. Furthermore, Atmark Techno, Inc. does not guarantee any outcomes resulting from the use of this product.

This product uses vendor provided tools (Xilinx's EDK, ISE or other vendors' tools) and IP cores to build and compile FPGA projects and to create configuration data. Atmark Techno, Inc., however, does not distribute, support or guarantee these tools.

2.4. Software Precautions

• Software Contained in This Product

The software and documentation contained in this product are provided "AS IS" without warranty of any kind including any warranty of merchantability or fitness for a particular purpose, reliability, correctness or accuracy. Furthermore, Atmark Techno, Inc. does not guarantee any outcomes resulting from the use of this product.

3. Before Getting Started

3.1. Required Items

The following are necessary to carry out development on the SUZAKU-V.

• Development PC

For hardware development, one PC that can run Windows 2000 or Windows XP and has one serial port and one parallel port is required.

For software development, one PC that can run Linux and has one serial port is required. For more information on software development, refer to the Software Manual.

• D-Sub 9 Pin Cross Cable

A D-Sub 9 pin (female-female) cable for cross-connection

- D-Sub9 Pin-10 Pin Conversion Cable
 A D-Sub 9 pin -10 pin conversion cable for connecting the D-Sub 9 pin to the pin header (10 pin) on the board
- **Development Kit CD-ROM (hereafter referred to as the "Supplied CD")** This kit contains various manuals and source code relating to the SUZAKU-V
- Serial Communication Software Serial communication software such as minicom or Tera Term is required. (Linux software can be found in the directory "tools" contained in the supplied CD-ROM).
- DC3.3V Power Supply A DC3.3V power supply
- Xilinx ISE
 A copy of Xilinx ISE
 For details, contact a Xilinx distributor.
- Xilinx EDK A copy Xilinx EDK For details, contact a Xilinx distributor.
- Xilinx Parallel Cable or Equivalent A Parallel Cable For details, contact a Xilinx distributor.

4. Overview

4.1. SUZAKU-V Features

SUZAKU-V is a board computer based on the Xilinx Virtex-II Pro FPGA.

It includes a hard-core processor "PowerPC405" along with peripheral cores within the FPGA and uses Linux as its default operating system.

Building Soft Processors and Peripheral Cores

Building the PowerPC405 and peripheral cores can be accomplished using the Xilinx EDK (Embedded Development Kit).

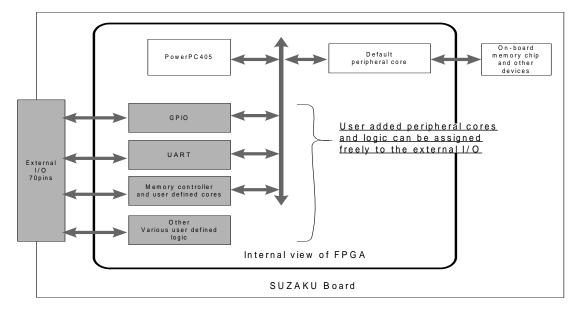
The EDK is a tool that allows for the configuration of the PowerPC405 and peripheral cores under a GUI environment and automatically creates a net list based on that configuration information.

Customization

The FPGA can be customized by the user.

In addition, the board has 70 external I/O pins that can be utilized by the user.

As an example, the number of PIOs and UARTs could be increased and assigned to the external I/O pins.



* Customization of the FPGA requires Xilinx's EDK and ISE. They can be obtained directly from Xilinx or through Xilinx's local distributors.

• LAN

The board has a LAN (10Base-T/100Base-TX) port to connect off-the-shelf LAN cables (UTP).

• Operating System

As Linux is used as the standard operating system, application software can be developed using the GNU assembler, C-compiler and so on.

Additionally, as the LAN controller device driver and the various protocols are provided, network connections can be easily made.

For more information on the operating system, refer to the Software Manual.

4.2. Specifications

The key specifications of the SUZAKU-V are shown in Table 4-1.

FPGA		Xilinx Virtex-II Pro XC2VP4 FG256				
Hard Core Processor		PowerPC 405				
Crystal Oscillator		3.6864MHz (frequency multiplied by FPGA's internal DCM)				
Memory	BRAM	16Kbyte				
	SDRAM	32Mbyte				
	FLASH Memory	8Mbyte				
Configurat	tion	Stored on FLASH memory, Controller TE7720				
JTAG		2 ports (FPGA and TE7720)				
Ethernet		10Base-T/100Base-TX				
Serial		UART 115.2kbps				
Timer		2-ch (1-ch is used for OS)				
Free I/O F	Pins	70-pin				
Reset Fur	nction	Software reset				
Power Supply		Voltage: 3.3V±3%				
		Consumption current: 460mA typ. (while processor is operating)				
Board Dim	nensions	72×47mm				

Table 4-1 SUZAKU-V Specifications

4.3. Block Diagram

The block diagram of the SUZAKU-V is shown in Figure 4-1. This is the minimum configuration necessary to run Linux.

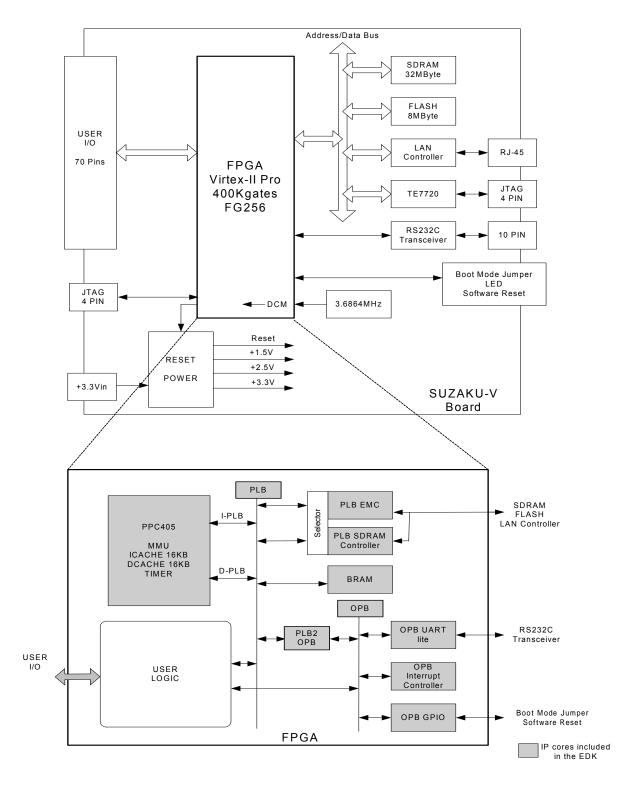


Figure 4-1 Block Diagram of SUZAKU-V

4.4. Functions

4.4.1. Processor

A PowerPC405 is utilized in the FPGA. The following is an overview of the PowerPC405.

- 32-bit RISC processor
- 32-bit fixed length instructions
- 32 generic 32-bit registers
- MMU
- Instruction cache (16 KB, 2-way) and data cache (16 KB, 2-way)

4.4.2. Bus

The bus consists of the following three types.

FPGA Internal PLB

A bus used to connect the PowerPC405, BRAM, PLB-SDRAM Controller and PLB-EMC to the peripheral IP cores (high speed access)

FPGA Internal OPB

A bus used to connect peripheral IP cores such as OPB-UARTlite and OPB-INTC.

FPGA External Bus

A bus used to connect an external memory device via PLB EMC and PLB SDRAM.

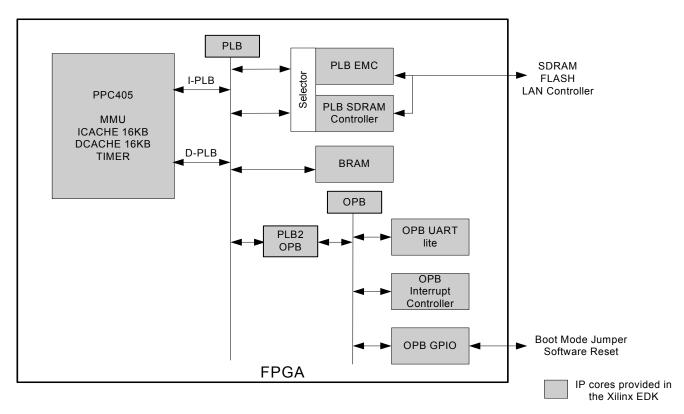


Figure 4-2 SUZAKU-V Bus Configuration

4.4.3. Memory

SUZAKU-V has the following three types of memory.

- BRAM (Default 16Kbyte, Internal to FPGA) Memory size is configurable Used for the boot program.
 - Used for the boot program.

Can be used for user programs after booting.

 FLASH Memory (8Mbyte, External to FPGA) Used to store data such as the full-featured boot loader, the Linux system and FPGA configuration data.

Is connected to PLB EMC.

 SDRAM (32Mbyte, External to FPGA) Used as the main system memory for Linux. Is connected to PLB SDRAM.

4.4.4. Interrupts

The OPB INTC in the FPGA is used as the operating system interrupt controller.

4.4.5. Timer

The PowerPC405's internal timers are used.

4.4.6. Serial Console

The board uses OPB UART Lite inside the FPGA as the operating system serial console. OPB UART Lite is connected to the CON1 connector through a 4-channel RS232C transceiver.

Two channels are used for the operating system serial console and the remaining two channels are unused. With these unused channels, it is possible to connect GPIO or user logic to enable flow control or to connect a seperate OPB UART Lite as a secondary UART port.

Serial console settings:

Data rate:115.2kbpsData:8bitStop bit:1bitFlow Control:Not supported

4.4.7. LAN

SMSC's LAN91C111 is equipped external to the FPGA as the LAN controller. The LAN91C111 is connected to PLB via PLB EMC and standard LAN cables (UTP) can be connected to the mounted RJ-45 connector.

4.4.8. External I/O

The board has 70 external I/O pins (CON2, CON3, CON4 and CON5) that can be used freely by the user. (Connectors are not mounted)

The external I/Os are directly connected to the free I/O pins of the FPGA. For more information, refer to **Section 7**, "**Interface Specifications.**"

The power for the FPGA I/O (VCCO) is supplied from the +3.3V internal logic power supply. For the rated value of I/O voltage and driving current, refer to the Virtex-II Pro data sheet.

Due to the sequence circuit, the +3.3V internal logic power supply takes up to 20msec to start up. Therefore, to avoid latch-up, all devices connecting to the external I/O must use the +3.3V internal power supply of the SUZAKU-V (refer to **Section 4.4.15, "+3.3V Internal Logic Power Output**") or will require a buffer device.

4.4.9. FPGA Configuration

SUZAKU-V is equipped with a TE7720 FPGA configuration IC developed by Tokyo Electron Device Limited.

The TE7720 programs the data from the JTAG (CON2) into the flash memory and also reads from the flash memory to configure the FPGA when the board is restarted (refer to Figure 4.3).

The whole flash memory area can be read and written from the processor.

Linux system and FPGA configuration data received from LAN or RS232C is written to flash memory, allowing the board to perform completely new functions after it has been rebooted.

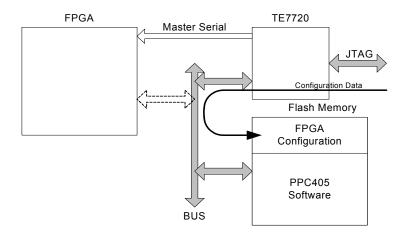
Moreover, SUZAKU-V is equipped with a software-controllable reset circuit, thus enabling remote reconfiguration.

You can obtain free software (LBPLAY2.EXE) for transferring data from JTAG (CON2) to the TE7720 by downloading it from the Tokyo Electron Device home page (the software is also contained in the supplied CD).

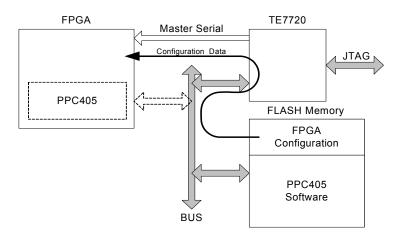
For information on configuration, refer to Section 10, "FPGA Configuration".

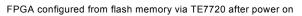
Be careful not to run the SUZAKU-V if incorrect data has been programmed into the FPGA or an error occurred during programming. Collision between signals and an FPGA external circuit component (including on-board components) or abnormal operation could cause heat generation, degradation or damage. To avoid this, once switch off the power, short JP2 and perform a reprogramming.

When the power is turned on again with JP2 shorted, SUZAKU-V will stop FPGA configuration to allow the reprogramming.



Configuration data from JTAG written to flash memory via TE7720





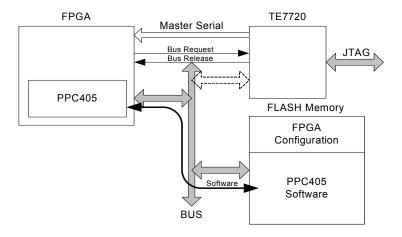


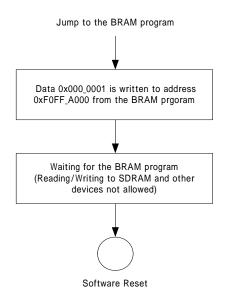


Figure 4-3 FPGA Configuration

4.4.10. Software Reset

When a software reset is performed, configuration data is read from the flash memory and used to reconfigure the FPGA, and reset signals sent to each device IC.

The software reset can be performed either by using the Linux reboot command or writing 0x000_0001 to the address 0xF0FF_A000 directly from the BRAM program. If the software reset is performed directly from the BRAM program, be careful not to read or write to SDRAM or other devices (including executing programs).



Software reset directly from the BRAM program

4.4.11. JTAG

The board provides the following two types of JTAG.

• FPGA Programming Connector (CON2)

This JTAG connector is used to program the FPGA configuration data to the flash memory (connector is not mounted on the board). Connect a JTAG cable such as Xilinx's Parallel Cable to the connector (CON2) to perform the programming using appropriate software (LBPLAY2.EXE). As the I/O voltage of this JTAG is +3.3V, please use a JTAG cable to accommodate the +3.3V output. TMS, TDI, TCK are all pulled up to +3.3V in the board via $4.7k\Omega$. For information on configuration, refer to **Section 10, "FPGA Configuration"**.

FPGA Connector (CON7)

This is a JTAG connector for the FPGA (connector not mounted). The connector is directly connected to the JTAG pins of the FPGA. As the I/O voltage of this JTAG is +2.5V, please use a JTAG cable to accommodate the +2.5V output. TMS, TDI, TCK are pulled up to +2.5V in the board via $4.7k\Omega$ and TDO via 200 Ω .

4.4.12. Setting Jumpers

SUZAKU-V has two jumpers.

• Boot Mode Jumper (JP1)

Under the default configuration, this jumper is used to select the system boot mode. At defaults, Linux will be booted automatically if the jumper is open, while the SUZAKU-V will go into bootloader mode if the jumper is shorted. (For more information on boot modes, refer to the associated software manual).

• FPGA Programming Jumper (JP2) – connected to P14 (INIT-B) of FPGA

This jumper is used when programming configuration data from the FPGA programming JTAG to the flash memory. This jumper is connected to P14 (INIT-B) of FPGA and pin 14 of CON3. The board will boot normally if the jumper is open, while FPGA configuration data will be programmed to the flash memory if it is shorted.

For information on configuration, refer to Section 10, "FPGA Configuration". (When the jumper is shorted at power-on, configuration of the FPGA will stop to allow the programming to flash memory).

4.4.13. LED

The following two types of LEDs are present on SUZAKU-V.

• Power-on LED (Green, D3)

This LED indicates 3.3V is being supplied to the board.

User-Controllable LED (Red, D1)

This LED is user-controllable.

It will light at "LO" level.

It is connected to the A9 pin of the FPGA.

4.4.14. Power Input +3.3V

- Power can be fed to the board from the +3.3V power inputs at CON2, CON3 and CON6.
- The +3.3V must be ±3% in accuracy and simple increment.
- · Do not repeatedly turn the board on and off at very short time intervals.
- A 22µF laminated ceramic condenser is used for the input.

4.4.15. +3.3V Internal Logic Power Output

The +3.3V internal logic power output supplies the FPGA I/O (VCCO) and other ICs. Up to 400mA^{*1} can be supplied in total to the external devices from CON1, CON2, CON3 and CON5. However, due to large load variation from the external devices, voltage variation can occur dependent on the response of the power input +3.3V.

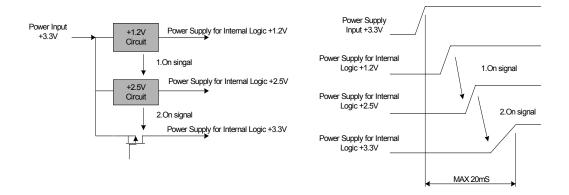
Power Input +3.3V	On resister 100m		Power output for Internal Logic → +3.3V MAX400mA *1
(CON2,3,6)		FPGA I/O (VCCO) and other ICs	(CON1,2,3,5)

*1 If signals are sent from the external I/O, total maximum current is given by:

Total Maximum Current = 400mA – Output Current of External I/O Signal

4.4.16. Internal Power Sequence

The internal power supply boots up in the following sequence.



5. Memory Map

5.1. SUZAKU-V Memory Map

The memory map of the board is shown in Table 5-1. It is the minimum configuration required to run Linux.

Start Address	End Address	Peripheral	Device
0x0000 0000	0x01FF FFFF	PLB-SDRAM Controller	SDRAM 32MByte
0x0200 0000	0xEFFF FFFF	Free	
0xF000 0000	0xF07F FFFF	PLB-EMC	FLASH Memory 8MByte
0xF080 0000	0xF0DF FFFF	Free	
0xF0E0 0000	0xF0EF FFFF	PLB-EMC	LAN Controller
0xF0F0 0000	0xF0FF 1FFF	Free	
0xF0FF 2000	0xF0FF 20FF	OPB-UART Lite	RS232C
0xF0FF 2100	0xF0FF 2FFF	Free	
0xF0FF 3000	0xF0FF 30FF	OPB-Interrupt Controller	
0xF0FF 3100	0xF0FF 9FFF	Free	
0xF0FF A000	0xF0FF A1FF	OPB-GPIO	Boot Mode Jumper
			Software Reset
0xF0FF A200	0xF0FF A3FF	OPB-GPIO	User Control LED
0xF0FF A400	0xFFFF BFFF	Free	
0xFFFF C000	0xFFFF FFFF	BRAM	BRAM 16KByte

Table 5-1 SUZAKU-V Memory Map

Table 5-2 Memory Map of 8MByte Flash Memory (0xF000 0000-0xF07F FFFF)

Start Address	End Address	Peripheral
0xF000 0000	0xF007 FFFF	Free
0xF008 0000	0xF00F FFFF	FPGA Configuration Data
0xF010 0000	0xF011 FFFF	Hermit Boot Loader Binary Image
0xF012 0000	0xF07E FFFF	Linux Binary Image
0xF07F 0000	0xF07F FFFF	Linux Configuration Data

6. FPGA Pin Assignment

Table 6-1 shows all pin assignments of the FPGA (Xilinx Virtex-II Pro C2VP4 FG256).

Table 6-1 FPGA Pin Assignment – External I/O (1/2)

No.	Bank	Signal Name	I/O	Function	Connected to
E14	2	L01N_2/VRP_2	I/O	External I/O	CON2 (Refer to Section 7)
E15	2	L01P_2/VRN_2	I/O	23	33
E13	2	L02N_2	I/O	23	33
F12	2	L02P_2	I/O	23	33
F13	2	L03N_2	I/O	33	33
F14	2	L03P_2	I/O	33	33
F15	2	L04N_2/VREF_2	I/O	23	33
F16	2	L04P_2	I/O	33	33
G13	2	L06N_2	I/O	33	33
G14	2	L06P_2	I/O	23	33
G15	2	L85N_2	I/O	33	33
G16	2	L85P_2	I/O	33	33
G12	2	L86N_2	I/O	23	33
H13	2	L86P_2	I/O	33	33
H14	2	L88N_2/VREF_2	I/O	33	33
H15	2	L88P_2	I/O	33	33
H16	2	L90N_2	I/O	33	33
J16	2	L90P 2	I/O	33	33
J15	3	L90N_3	I/O	33	33
J14	3	L90P_3	I/O	23	33
J13	3	L89N_3	I/O	23	33
K12	3	L89P_3	I/O	23	33
K16	3	L87N_3/VREF_3	I/O	23	33
K15	3	L87P_3	I/O	23	33
K14	3	L85N_3	I/O	23	33
K13	3	L85P_3	I/O	23	33
L16	3	L06N_3	I/O	23	33
L15	3	L06P_3	I/O	23	33
L14	3	L05N_3	I/O	33	33
L13	3	L05P_3	I/O	33	33
L12	3	L03N_3/VREF_3	I/O	"	CON3 (Refer to Section 7)
M13	3	L03P_3	I/O	33	33
M16	3	L02N_3	I/O	33	33
N16	3	L02P_3	I/O	33	33
M15	3	L01N_3/VRP_3	I/O	"	33
M14	3	L01P_3/VRN_3	I/O	"	33

No.	Bank	Signal Name	I/O	Function	Connected to
P15	4	L01N_4/BUSY/DOUT1	I/O	External I/O	CON3 (Refer to Section 7)
P14	4	L01P_4/INIT_B	I/O	"	CON3 (Refer to Section 7)
					JP2
					Also used as configuration pin
R14	4	L02N_4/D0/DIN1	I/O	33	CON3 (Refer to Section 7)
					Also used as configuration pin
P13	4	L02P_4/D1	I/O	"	CON3 (Refer to Section 7)
T15	4	L03N_4/D2	I/O	33	"
T14	4	L03P_4/D3	I/O	33	"
N12	4	L06N_4/VRP_4	I/O	"	"
P12	4	L06P_4/VRN_4	I/O	33	"
N11	4	L07P_4/VREF_4	I/O	33	"
M11	4	L09N_4	I/O	33	"
M10	4		I/O	"	CON3 (Refer to Section 7)
		L09P_4/VREF_4			CON5 (Refer to Section 7)
N10	4	L69N_4	I/O	33	"
P10	4	L69P_4/VREF_4	I/O	33	"
N9	4	L74N_4/GCLK3S	I/O	33	CON3 (Refer to Section 7)
P9	4	L74P_4/GCLK2P	I/O	"	"
R9	4	L75N_4/GCLK1S	I/O	"	33
Т9	4	L75P_4/GCLK0P	I/O	33	53
T8	5		I/O	33	CON3 (Refer to Section 7)
		L75N_5/GCLK7S			CON5 (Refer to Section 7)
R8	5	L75P_5/GCLK6P	I/O	"	"
P8	5	L74N_5/GCLK5S	I/O	33	"
N8	5		I/O	33	CON3 (Refer to Section 7)
		L74P_5/GCLK4P			CON4 (Refer to Section 7)
P7	5	L69N_5/VREF_5	I/O	"	"
N7	5	L69P_5	I/O	33	"
M7	5	L09N_5/VREF_5	I/O	33	"
M6	5	L09P_5	I/O	33	"
N6	5	L07N_5/VREF_5	I/O	"	33
P5	5	L06N_5/VRP_5	I/O	"	33
N5	5	L06P_5/VRN_5	I/O	"	33
T3	5	L03N_5/D4	I/O	"	>>
T2	5	L03P_5/D5	I/O	"	>>
P4	5	L02N_5/D6	I/O	"	CON4 (Refer to Section 7)
R3	5	L02P_5/D7	I/O	"	37
P3	5	L01N_5/RDWR_B	I/O	"	"
P2	5	L01P_5/CS_B	I/O	"	33

Table 6-2 FPGA Pin Assignment – External I/O (1/2)

No.	Bank	Signal Name	I/O	Function	Connected to
J2	6	LA(22)	0	FPGA external bus	SDRAM, FLASH memory, LAN
					controller
J3	6	LA(21)	0	"	37
J4	6	LA(20)	0	"	37
K5	6	LA(19)	0	"	"
K1	6	LA(18)	0	"	"
K2	6	LA(17)	0	"	"
K3	6	LA(16)	0	"	"
K4	6	LA(15)	0	"	"
L1	6	LA(14)	0	"	"
L2	6	LA(13)	0	"	"
L3	6	LA(12)	0	"	"
L4	6	LA(11)	0	33	33
L5	6	LA(10)		33	33
M4	6	LA(10)_RAM	0	33	" (for SDRAM)
M1	6	LA(9)	0	33	33
N1	6	LA(8)	0	33	33
M2	6	LA(7)	0	33	"
M3	6	LA(6)	0	33	"
E3	7	LA(5)	0	33	"
E2	7	LA(4)	0	33	"
E4	7	LA(3)	0	33	33
F5	7	LA(2)	0	33	"
F4	7	LA(1)	0	33	"
F3	7	LA(0)	0	22	"
F2	7	LD(15)	I/O	FPGA external data bus	SDRAM, FLASH memory, LAN controller
F1	7	LD(14)	I/O	"	17
G4	7	LD(13)	I/O	"	17
G3	7	LD(12)	I/O	"	"
G2	7	LD(11)	I/O	"	"
G1	7	LD(10)	1/O	"	57
G5	7	LD(9)	I/O	"	17
H4	7	LD(8)	I/O	"	17
H3	7	LD(7)	I/O	"	17
H2	7	LD(6)	I/O	"	17
H1	7	LD(5)	I/O	"	"
J1	7	LD(4)	I/O	"	17
A8	0	SYS_RST_IN		System reset input	Reset circuit

Table 6-3 FPGA Pin Assignment – Internal Devices (1/2)

No.	Bank	Signal Name	I/O	Function	Connected to
B8	0	BOOTMODE	Ι	Boot mode detection	JP1 (Refer to Section 7)
C8	0	SYS_CLK_IN	I	System clock input	Crystal Oscillator 3.6864MHz
D8	0	RAM_CLK	I	SDRAM clock DCM	SDRAM
				feedback input	
C7	0	SYS_CLK_OUT	0	Clock output to SDRAM	SDRAM
D7	0	LD(3)	I/O	FPGA external data bus	SDRAM, FLASH memory,
				33	LAN controller
E7	0	LD(2)	I/O	33	21
E6	0	LD(1)	I/O	33	21
D6	0	LD(0)	I/O		
C5	0	FLASH_CE*	0	FLASH memory CE	FLASH memory
D5	0	FLASH_OE*	0	FLASH memory OE	33
A3	0	FLASH_WE*	0	FLASH memory WE	"
A2	0	MAC_BE1*	0	LAN controller BE1	LAN controller
C4	0	MAC_BE0*	0	LAN controller BE0	"
B3	0	MAC_RD*	0	LAN controller RD	"
C3	0	MAC_WR*	0	LAN controller WR	"
C2	0	MAC_INTR		LAN controller INTR	"
C15	1	BUS_REQ	0	Bus request	TE7720
C14	1	BUS_REL	I	Bus acquisition	TE7720
B14	1	RAM_CS*	0	SDRAM CS	SDRAM
C13	1	RAM_RAS*	0	SDRAM RAS	27
A15	1	RAM_CAS*	0	SDRAM CAS	33
A14	1	RAM_WE*	0	SDRAM WE	33
D12	1	RAM_CKE	0	SDRAM CKE	33
C12	1	RAM_UQDM	0	SDRAM UQDM	33
D11	1	RAM_LQDM	0	SDRAM LQDM	33
E11	1	RAM_BS(1)	0	SDRAM BS	33
E10	1	RAM_BS(0)	0	SDRAM BS	33
D10	1	CNSL_CTS*	I	Console CTS	RS232C transceiver =>CON1
		_			(Refer to Section 7)
C10	1	CNSL_RXD		Console RXD	"
D9	1	CNSL_RTS	0	Console RTS	27
C9	1	CNSL_TXD	0	Console TXD	33
B9	1	FPGA_RESET_EN	0	Self reset output	Reset circuit
A9	1	LED*	0	User-controllable LED	D1 (Refer to Section 7)

Table 6-4 FPGA Pin Assignment – Internal Devices (2/2)

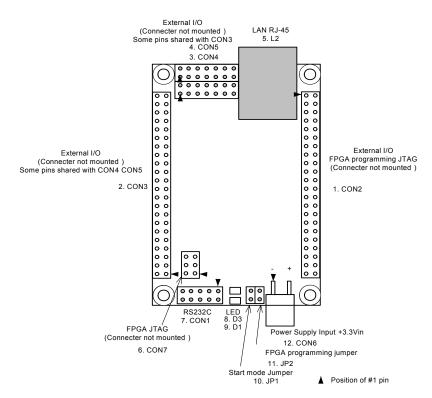
The MSB of LA(0 to 22), LD(0 to 15) and RAM_BS(0 to 1) in top.vhd are set to bit (0). This is opposite that of bit labels of normal external devices (Normally the LSB side is set to bit (0)). The above tables show the bit labels when connecting to standard external devices (LSB side is set to bit (0)).

No.	Bank	Signal Name	I/O	Function	Connected to
D16		TCK	I	JTAG	CON7 (Refer to Section 7)
E1		TDI		JTAG	33
E16		TDO	0	JTAG	33
C16		TMS		JTAG	33
N15		CFG_CLK	0	Configuration CLK	TE7720
D1		PROG_B		Configuration PROG_B	Reset circuit
P16		CFG_DONE	0	Configuration DONE	TE7720
C1		HSWAP_EN		Open	
N3		MO	I	Configuration mode	Ground
N2		M1	I	Configuration mode	Ground
P1		M2		Configuration mode	Ground

Table 6-5 FPGA Pin Assignment – JTAG and Configuration

7. Interface Specifications

7.1. Interface Layout



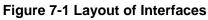


Table 7-1 Interface Details

	Part Number	Description
1	CON2	External I/O, FPGA programming JTAG connector
		Total I/Os 32 pins
2	CON3	External I/O connector
		Total I/Os 34 pins (some pins are the same signal as CON4 or 5)
3	CON4	External I/O connector
		Total I/Os 10 pins (some pins are the same signal as CON3)
4	CON5	External I/O connector
		Total I/Os 10PIN (some pins are the same signal as CON3)
5	L2	Ethernet 10/100 Base-T connector
6	CON7	FPGA JTAG connector
7	CON1	RS232C connector
8	D3	Power-on LED Green
9	D1	User-controllable LED Red
10	JP1	Start mode jumper
11	JP2	FPGA programming jumper
12	CON6	Power input +3.3V connector

7.2. CON2 External I/O, FPGA Programming JTAG Connector

This is a JTAG connector used for external I/O or FPGA programming (connector not mounted).

Table 7-2 CON2 Connector for External I/O or FPGA Programming

No.	Signal Name	I/O	Function	
1	GND		Ground	
2	+3.3VOUT		Internal logic power output +3.3V	
3	CFG_TCK		JTAG for FPGA programming	TCK
4	CFG_TDI		13	TDI
5	CFG_TDO		33	TDO
6	CFG_TMS		33	TMS
7	01N_2/VRP_2		External I/O Virtex-2 Pro connection pin number	E14
8	01P_2/VRN_2		13	E15
9	02N_2		33	E13
10	02P_2		33	F12
11	03N_2		33	F13
12	03P_2		33	F14
13	04N_2/VREF_2		33	F15
14	04P_2		33	F16
15	06N_2		19	G13
16	06P_2		33	G14
17	85N_2		33	G15
18	85P_2		33	G16
19	GND		Ground	
20	74N_4/GCLK3S		External I/O Virtex-2Pro connection pin number	N9
21	GND		Ground	
22	74P_4/GCLK2P		External I/O Virtex-2Pro connection pin number	P9
23	86N_2		19	G12
24	86P_2		33	H13
25	88N_2/VREF_2		33	H14
26	88P_2		33	H15
27	90N_2		33	H16
28	90P_2		33	J16
29	90N_3		55	J15
30	90P_3		19	J14
31	89N_3		19	J13
32	89P_3		13	K12
33	87N_3/VREF_3		13	K16
34	87P_3		13	K15
35	85N_3		13	K14
36	85P_3		13	K13
37	06N_3		13	L16
38	06P_3		19	L15
39	05N_3		19	L14
40	05P_3		33	L13
41	GND		Ground	
42	GND		Ground	
43	+3.3VIN		Power input +3.3V	
44	+3.3VIN		Power input +3.3V	

7.3. CON3 External I/O Connector

This is a JTAG connector for external I/O or TE7720 (connector not mounted).

No.	Signal Name	I/O	Function		
1	+3.3VIN		Power input +3.3V		
2	+3.3VIN		Power input +3.3V		
3	GND		Ground		
4	GND		Ground		
5	03N 3/VREF 3		External I/O Virtex-2 Pro connection pin number	L12	
6	03P 3		33	M13	
7	02N 3		33	M16	
8	02P_3		33	N16	
9	01N 3/VRP 3		33	M15	
10	01P 3/VRN 3		33	M14	
11	01N_4/BUSY/D1		33	P15	
12	02P_4/D1		33	P13	
13	02N_4/D0/DIN		33	R14	Note 1
14	01NP_4/INIT_B		33	P14	Note 1
15	03N_4/D2		33	T15	
16	03P_4/D3		33	T14	
17	06N_4/VRP_4		13	N12	
18	06P_4/VRN_4		33	P12	
19	07P_4/VREF_4		11	N11	
20	09N_4		11	M11	
21	09P_4/VREF_4		"	M10	Note 2
22	69N_4		27	N10	Note 2
23	75N_4/GCLK1S		"	R9	
24	GND		Ground		
25	75P_4/GCLK0P		External I/O Virtex-2Pro connection pin number	Т9	
26	GND		Ground		
27	69P_4/VREF_4		External I/O Virtex-2Pro connection pin number	P10	Note 2
28	75N_5/GCLK7S		33	T8	Note 2
29	75P_5/GCLK6P		11	R8	Note 2
30	74N_5/GCLK5S		"	P8	Note 2
31	74P_5/GCLK4P		"	N8	Note 3
32	69N_5/VREF_5		33	P7	Note 3
33	69P_5		33	N7	Note 3
34	09N_5/VREF_5		33	M7	Note 3
35	09P_5		"	M6	Note 3
36	07N_5/VREF_5		3) 11	N6	Note 3
37	06N_5/VRP_5		33	P5	Note 3
38	06P_5/VRN_5		33	N5	Note 3
39	03N_5/D4		23	T3	Note 3
40	03P_5/D5			T2	Note 3
41			Unused		
42	EXRESET*		No connection (Caution: Do not input signals)		
43	+3.3VOUT Internal logic power output +3.3V				
44	GND Ground				

7.4. CON4 External I/O Connector

This is an external I/O connector (not mounted on the board).

Table 7-4 CON4 External I/O Connector	
---------------------------------------	--

No.	Signal Name	I/O	Function		
1			Unused		
2			Unused		
3	74P_5/GCLK4P		External I/O Virtex-2Pro connection pin number	N8	Note 3
4	69N_5/VREF_5			P7	Note 3
5	69P_5			N7	Note 3
6	09N_5/VREF_5			M7	Note 3
7	09P_5			M6	Note 3
8	07N_5/VREF_5			N6	Note 3
9	06N_5/VRP_5			P5	Note 3
10	06P_5/VRN_5			N5	Note 3
11	03N_5/D4			Т3	Note 3
12	03P_5/D5			T2	Note 3

7.5. CON5 External I/O Connector

This is an external I/O connector (not mounted on the board).

Table 7-5 CON5 External I/O Connector

No.	Signal Name	I/O	Function		
1	GND		Ground		
2	+3.3VOUT		Internal logic power output +3.3V		
3	02N_5/D6		External I/O Virtex-2Pro connection	P4	
			pin number		
4	02P_5/D7			R3	
5	01N_5/RDWR_B			P3	
6	01P_5/CS_B			P2	
7	09P_4/VREF_4			M10	Note 2
8	69N_4			N10	Note 2
9	69P_4/VREF_4			P10	Note 2
10	75N_5/GCLK7S			T8	Note 2
11	75P_5/GCLK6P			R8	Note 2
12	74N_5/GCLK5S			P8	Note 2

Note:

- 1. Pin 13 of CON3 (signal name: 02N_4/D0/DIN) and pin 14 of CON3 (signal name: 01NP_4/INIT_B) are also used as FPGA configuration pins.
- 2. Pins 21 and 22 and 27 to 30 of CON3 and pins 7 to 12 of CON5 wire the same signal.
- 3. Pins 31 to 40 of CON3 and pins 3 to 12 of CON4 wire the same signal.

7.6. CON7 FPGA JTAG Connector

This is a FPGA JTAG connector (not mounted on the board). As the I/O voltage of this JTAG is +2.5V, please use an appropriate JTAG cable compatible with the +2.5V.

No.	Signal Name	I/O	Function
1	GND		Ground
2	+2.5VOUT		Internal logic power output +2.5V
3	TCK		JTAG
4	TDI	I	JTAG
5	TDO	0	JTAG
6	TMS		JTAG

Table 7-6 CON7 Virtex-II Pro JTAG Connector

7.7. CON1 RS232C Connector

This is a RS232C connector. It is connected to the FPGA via a level buffer. Type and manufacturer of the connector used on the board is A1-10PA-2.54DSA/Hirose (or equivalent).

Serial console settings:

- Data rate 115.2kbps
- Data 8bit
- Stop bit 1bit
- Flow control
 Not supported

Table 7-7 CON1 RS232C Connector

No.	Signal Name	I/O		Function
1			Unused	
2			Unused	
3	RXD	I	Virtex-2 Pro connection pin	C10 (for serial console)
			number	
4	RTS	0	33	D9
5	TXD	0	33	C9 (for serial console)
6	CTS	I	33	D10
7			Unused	
8			Unused	
9	GND		Ground	
10	+3.3VOUT		Internal logic power output +	3.3V

7.8. JP1 Boot Mode Jumper

This jumper is used to switch the boot mode. If it is set to open, the board auto boots. If it is set to short, the board goes into boot loader mode. The jumper is connected to the FPGA. For more information on the boot modes, refer to the Software Manual.

Table 7-8 JP1 Boot Mode Jumper

No.	Signal Name	I/O	Function
1	DLOAD		Open: Auto boot mode Short: Boot loader mode Virtex-2 Pro connection pin number B8
2	GND		Ground

7.9. JP2 FPGA Programming Jumper

This jumper is used to program configuration data to the flash memory from the FPGA programming JTAG.

For information on configuration, refer to Section 10, "FPGA Configuration".

No.	Signal Name	I/O	Function
1	TE77PRG		Open: Normal boot
			Short: Configuration data program
2	GND		Ground

7.10. D3 Power-on LED

The LED lights (green) when fed 3.3V.

7.11.D1 User-Controllable LED

This LED is user-controllable. It lights at "LO" level (red). It is connected to the FPGA.

Table 7-10 D1 User-Controllable LED

No.	Signal Name	I/O	Function	1
	LED0		LO. Level: ON HI. Level: OFF Virtex-2Pro connection pin number	A9

7.12. CON6 Power Input +3.3V Connector

This is a power input connector. The +3.3V power input must be +3.3V±3% and simple increment. It is internally connected to the +3.3V power inputs of CON2 and CON3. Type and name of manufacturer of the connector for the board side is B2PS-VH/J.S.T. (or equivalent). Type and name of manufacturer of the connector for the cable side is Housing VHR-2N/J.S.T., Contact BVH-21T-P1.1/ J.S.T. or BVH-41T-P1.1/ J.S.T. (or equivalent).

No.	Signal Name	I/O	Function
1	GND		Ground
2	+3.3VIN		+3.3V Power input

Table 7-11 CON6 Power Input +3.3V Connector

7.13. Ethernet 10/100 Base-T

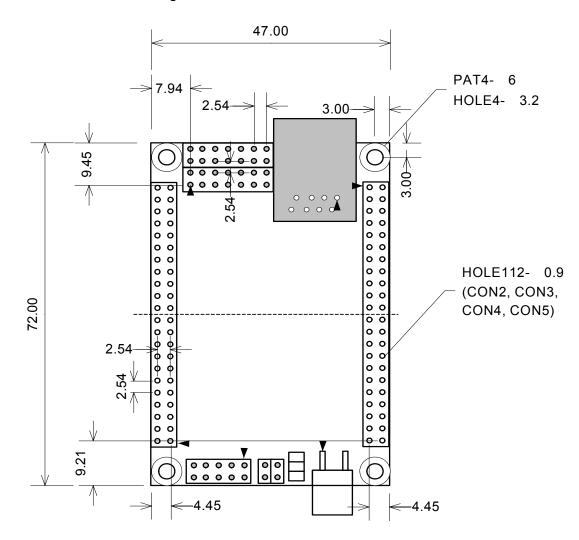
Type and name of manufacturer of the connector for the board side is J0026D21B/PULSE.

No.	Signal Name	I/O	Function	
1	TX+		Differential twist pair output +	
2	TX-		Differential twist pair output -	
3	RX+		Differential twist pair input +	
4			75Ω termination (pin 4 and 5 are shorted)	
5			75Ω termination (pin 4 and 5 are shorted)	
6	RX-		Differential twist pair input -	
7			75Ω termination (pin 7 and pin 8 are shorted)	
8			75Ω termination (pin 7 and pin 8 are shorted)	

Table 7-12 Ethernet 10/100 Base-T

8. Board View

The board view is shown in Figure 8-1.



Unit: mm

Figure 8-1 SUZAKU-V Board View

9. Creating a FPGA Project (Example: Adding a UART)

As an example, the following describes adding UART to the default FPGA project in the attached CD-ROM.

9.1. Extracting the fpga_proj Project from the CD-ROM

Extract the compressed suzaku-v-******.zip file form \fpga_proj\suzaku-v on the supplied CD-ROM to the hard-disk (* indicates revision date).

The following shows the structure of the folder.

- xps_proj : EDK project folder
- top.vhd : Project top file (calls xsp_proj.xmp)
- top.ucf : FPGA pin assignment data file
- top.mcs : FPGA configuration data file
- Ibplay2.exe : FPGA programming application (for configuration device TE7720)
- device.def : FPGA programming application settings file

😂 suzaku-v-20050212			
<u> </u>	p		
🚱 Back 🝷 🕥 🚽 🏂 🔎 Search	Folders		
Address 🗁 C:\suzaku\suzaku-v-20050212			🖌 🄁 Co
Folders ×	Name 🔺	2	iize Type
 Desktop My Documents My Computer J.5 Floppy (A:) Local Disk (C:) Documents and Settings Program Files suzaku suzaku 	xps_proj device.def lbplay2.exe top.mcs top.ucf top.vhd	64 1,033 3 16	File Folder KB DEF File KB Application KB MCS File KB UCF File KB VHD File
6 objects (Disk free space: 36.3 GB)		1.08 MB 🔍 My	Computer ;
		3 ,	

9.2. Creating a New Project

Activate Xilinx Project Navigator and then choose New Project from the File menu. Enter "top" in the Project Name field and specify the name of the folder that the project was copied to in the Project Location field. Lastly, click the Next button.

New Project		
Enter a Name and Location for the F	roject	
Project <u>N</u> ame:	Project <u>L</u> ocation:	
top	C:\suzaku\suzaku-v-20050212	
K	>	·
Select the type of Top-Level module	for the Project	
Top-Level Module Type:		
HDL	-	Specify the folder that the
	<u>\</u>	project was copied to.
Ente	er "top".	
	< Back Next > Cancel	

Set the device and design flow for the project as follows.

Device Family	 Virtex2P
Device	 xc2vp4
Package	 fg256
Speed Grade	 -5
Synthesis Tool	 XST(VHDL/Verilog)

Property Name Device Family	Value Virtex2P
Device	xc2vp2
Package	fg256
Speed Grade	-5
Top-Level Module Type	HDL
Synthesis Tool	XST (VHDL/Verilog)
Simulator	Other
Generated Simulation Language	VHDL

Click Next until you reach the following screen and then finally click the Finish button.

N	ew Project Information	
	Project Navigator will create a new Project with the following specifications:	
	Project Name: top Project Location: C:\suzaku\suzaku-v-20050212 Project Type: HDL Device:	
	Device Family: Virtex2P Device: xc2vp2 Package: fg256 Speed Grade: -5	
	: Top-Level Module Type: HDL Synthesis Tool: XST (VHDL/Verilog) Simulator: Other Generated Simulation Language: VHDL	
	< <u>B</u> ack Finish Cancel Help	

Click Add Source from the Project menu.

Select top.ucf and top.vhd and then click the Open button.

Add Existing	s Sources	? 🛛
Look jn: ଢ	suzaku-v-20050212 💌 🗲 🗈	r 📰 📩
Cprojnav xps_proj top.ucf top.vhd		
File <u>n</u> ame:	"top.vhd" "top.ucf"	<u>O</u> pen
Files of <u>type</u> :	Sources (*.txt;*.vhd;*.vhdl;*.v;*.abl;*.xco;*.sc	Cancel

Select VHDL Design File and then click the OK button.

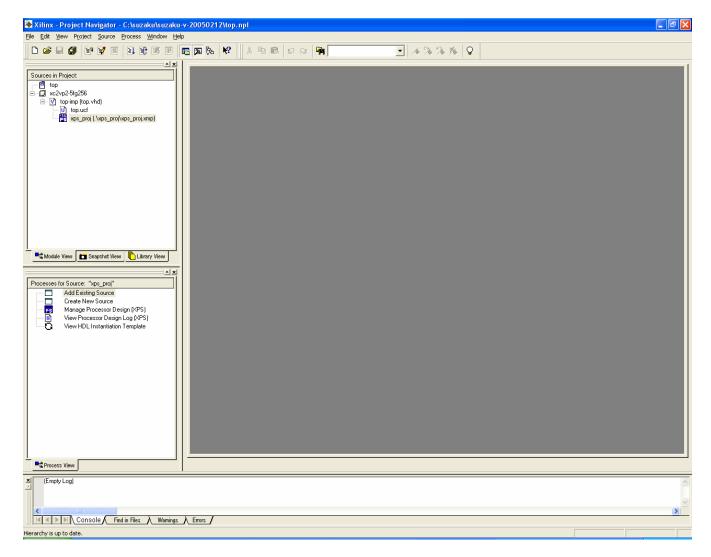
Choose Source Type	×
top.vhd is which source type? The suffix is ambiguous as to type	э.
VHDL Design File VHDL Test Bench File	OK]
	Cancel
	Help

Again, from the Add Source menu, select xps_prj\xps_proj.xmp and then click the Open button.

Add Existing	g Sources 🔹 🥐 🔀
Look jn: ଢ	xps_proj 💽 🗲 🗈 📸 📰 -
xps boot code data etc pcores	ppc405_i ₽s _proj.xmp
File <u>n</u> ame:	xps_proj.xmp
Files of <u>t</u> ype:	Sources (*.txt;*.vhd;*.vhd;*.v;*.abl;*.xco;*.sc Cancel

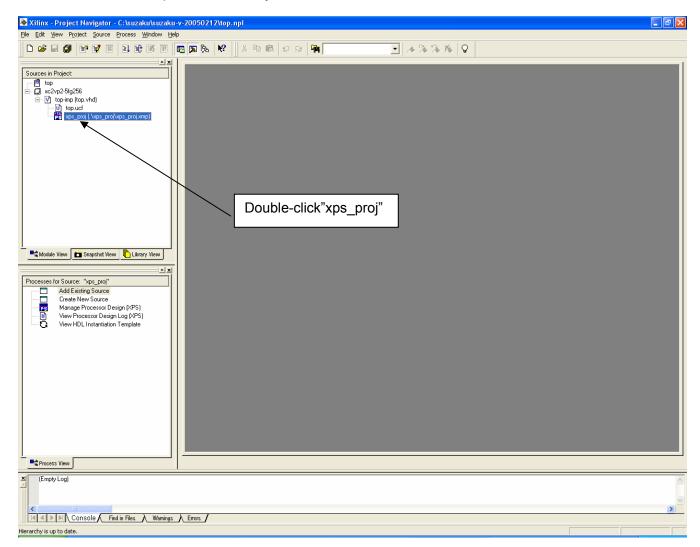


You have now created a new Project. The screen will look as follows.



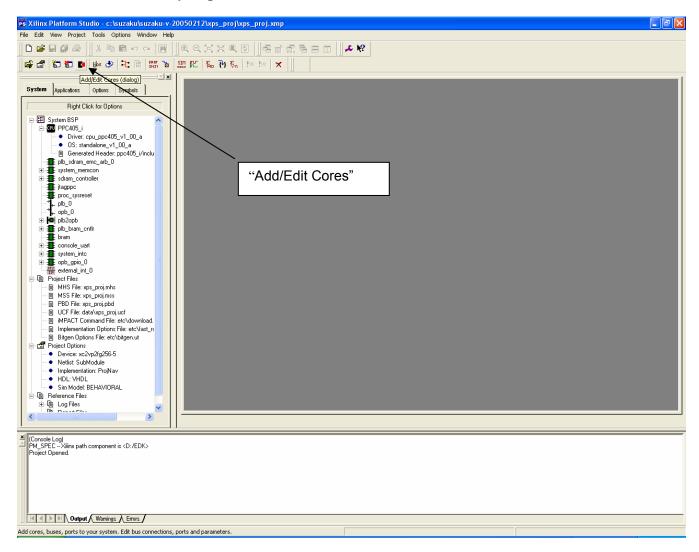
9.3. Setting up Platform Studio

From the Project Navigator menu, double-click xps_proj. Platform Studio will open automatically.



To add a new Peripheral Core, click Add/Edit Cores.

If compiling the default configuration without adding any peripheral cores, please move on to **Section 9.4**, **"Platform Studio Compiling"**.



On the Peripherals tab, select opb_uartlite from the list of peripheral cores at the right of the window. Click the Add button to add it and then set the Instance field. In this example it has been set as follows.

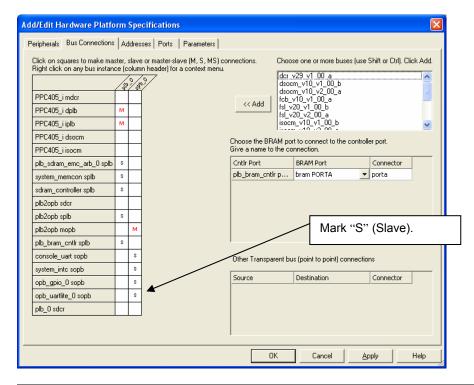
in this example it has been set as follows

Instance opb_uartlite_0

The configuration of other default peripherals (PowerPC405~system_external_int) is set to meet the minimum configuration requirements for running Linux.

Add/Edit Hardware		i <mark>pecifications</mark> dresses Ports Parameters	
Cells with where the second se	nite backgroun shoose one or	ds can be edited. To delete more rows and click Delete.	Show All Component Filter Processor G Specific C Debug
Peripheral ppc405 pib_sdram_emc pib_sdram jtagppc_cnth proc_sys_reset pib_obridge pib_bram_if_cnth bram_block opb_uartlite opb_gpio externat_int opb_uartlite	1.00.a 1.00.b 1.00.a 1.00.b	itagppc proc_sysreset plb2opb plb_bram_cnthr bram console_uart system_intc	C MicroBlaze Only C PowerPC Only Either Processor Bus C Clocking C Clocking C Communications C
			opb_sysace and then click the opb_time Add button. opb_uartile pb_atrice plb_atrice Image: Click the opt

On the Bus Connections tab click OPB Bus (d_opb_v20) under opb_uartlite, marking it with "S" (Slave).



On the Addresses tab set the Base Address and Size of opb_uartlite as follows:

Base Address 0xF0FFB000 Size 256

Instance	Prefix	Base Address	High Address	Size		Min	Info/Error	Lock	I C	D C
PPC405_i	ISOCM_DCR			UNSPEC	_					
PPC405_i	DSOCM_D			UNSPEC	_					
system_memcon	MEMO	0×F0E0_0000	0xf0efffff	1 MB	•	0×08			Г	
system_memcon	MEM1	0×F000_0000	0×f07fffff	8 MB	•	0×08			Г	
sdram_controller		0×0000_0000	0×01ffffff	32 MB	-	0x08				
plb_0				UNSPEC	•	0				
opb_0				UNSPEC	•	0×100				
plb2opb	RNG0	0×F0F00000	0×f0ffffff	1 MB	-	0				
plb2opb	RNG1			UNSPEC	•	0				
plb2opb	RNG2			UNSPEC	-	0				
plb2opb	RNG3			UNSPEC	•	0				
plb2opb	DCR			UNSPEC	•	0				
plb_bram_cntlr		0×FFFFC000	0×FFFFFFFF	16 KB	•	0x4				
console_uart		0×F0FF2000	0×f0ff20ff	256	•	0×100				
system_intc		0×F0FF3000	0×f0ff30ff	256	•	0x20		Ē		
opb_gpio_0		0×F0FFA000	0×f0ffa0ff	256	-	0×100				
opb_uartlite_0		0×F0FFB000	0×f0ffb0ff	256	-	0×100				
Generate addresses for peripherals that do not have lock checkbox checked										

On the Ports tab select OPB_CLK, RX and TX for opb_uartlite_0 from the list of signals at the right side of the window and click the Add button.

Give the following for each Net Name (Net Name is case sensitive).

OPB_CLK	SYS_CLK (capitals)
RX	opb_uartlite_0_RX
ТХ	opb_uartlite_0_TX

	Ad	d/Edit Hardware	Platform Specific	ations									×	
	Peripherals Bus Connections Addresses Ports Parameters													
		XE External Ports	Connections:								Show ports with def	fault conne	ections	
		Port Name	Net Name	Pola	Range	Class	Sensit		^		Ports Filter:			
		MEMCON_OEn	MEMCON_OEn	0	[0:1]				-				-	
		MEMCON_WEn	MEMCON_WEn	0							List of Ports. Click Add t	to add por		
		MEMCON_CEn	MEMCON_CEn	0	[0:1]						system_intc OPB Clk		^	
		MEMCON_BEn	MEMCON_BEn	0	[0:1]						Intr			
		SYS_Rst	SYS_Rst	I						Delete	lrq	~~~~~		
		CONSOLE_TX	CONSOLE_TX	0						Delete	opb_gpio_0			
		CONSOLE_RX	CONSOLE_RX	I						Add Port	OPB_CIk GPIO IO			
		GPIO IO 0	GPIO IO N	10	[0:7]							~~~~~		
	F	nter the Ne	et Names						~		external_int_0			
τı											0	~~~~~		
			added to th							Make	opb uartlite 0			
E	cte	ernal Port (Connection		ne	Pol.	Range	Class	^	External	OPB_CIk			
_	_				errupt	• •		INTER			Interrupt RX			
		plb_0	SYS_Rst	sys_bus		▼ I				<< Add	TX	~~~~~		
		plb_0	PNB_Clk	SYS_CL		▼ I	Four C	CLK						
		plb_0	M_basLock	net_gno		▼ I ▼ I	[0:C	•		Delete	PLB_SK SYS_Rst			
		opb_0 opb_0	SYS_R	sys_bus	-	—		CLK			ArbAddrVIdReg			
		opb_o	OPB CIK	SYS_CL						Connect	Bus_Error_Det	<u> </u>		
		opb_uartite_0	RX	_	rtlite_0			CLK		$\overline{}$	opb_0		Solo	ect "OPBCLK", "RX" and
Q		opb_uartlite_0	TX		rtlite_0	_			-	\mathcal{I}	OPB_Clk SYS_Rst			-
	\sim		1	opo_aa	caco_on	•		5	سخر	/	<	"T	X" to	or opb_uartlite_0 and then
		,			_							clic	ck th	e Add button.
								0	К	Ca	ncel Apply			

Select opb_uartlite_0_RX and opb_uartlite_0_TX and then click Make External.

They will be added to External Ports Connections. The names registered here are used as port names when called as components in top.vhd (project top file).

	ections Addresses	11 diamo	010					1	
External Ports	Connections:						Show ports Ports Filter:	s with default connections	
Port Name	Net Name	Pola Range	Class	Sensit		^		a addad to the	
MEMCON_CEn	MEMCON_CEn	0 [0:1]						e added to the	
MEMCON_BEn	MEMCON_BEn	O [0:1]			/	/	External Po	ort Connections	
SYS_Rst	SYS_Rst	I							
CONSOLE_TX	CONSOLE_TX	0					Intr		
CONSOLE_RX	CONSOLE_RX	I					Delete Irq		
GPIO_IO_0	GPIO_IO_0	IO [0:7]					opb_gpio_0		
MAC_INTERRUPT	MAC_INTERRUPT	1		×			Add Port OPB_Clk GPI0_I0		
opb_uartlite_0_RX	opb_uartlite_0_RX	I		>			~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~	~~~~~	
opb_uartlite 0_TX	opb_uartlite_0_TX	0					external_int_0)	
Internal Ports Connec Instance	tions: Port Name	Net Name	Pol.	. Range	Class	^	Make Apb_uartite_C		
external_int_0	0	mac_interrupt	• 0		INTER		Interrupt	Make External.	
plb_0	SYS_Rst	sys_bus_reset	▼ I				<< Add RX TX	Marce External.	
plb_0	PLB_Clk	SYS_CLK	▼ I		CLK			~~~~~	
plb_0	M_busLock	net_gnd	- I	[0:C			Delete PLB Clk		
opb_0	SYS_Rst	sys_bus_reset	▼ I				SYS_Rst		
opb_0	OPB_Clk	SYS_CLK	- I			/	Connect ArbAddrVldF Bus Error D		
opb uartlite 0	OPB_Clk	SYS_CLK	▼ I		C.				
opb_daraice_o	RX	opb_uartlite_0.	🔻 I				opb_0 OPB_Clk		
opb_uartlite_0	ТХ	opb uartlite 0.	🔻 0			~	SYS_Rst		
	18								

On the Parameters tab select all parameters from the list at the right side and then click the Add button. Then specify the peripheral parameters.

In this example they have been set as follows:

8

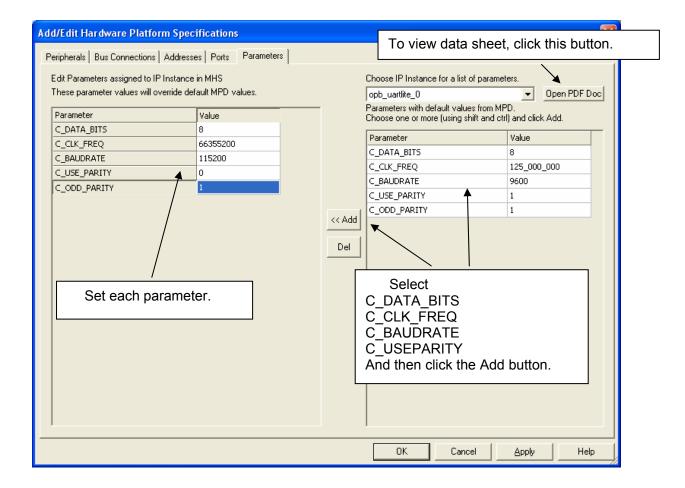
C_DATA_BITS

C_CLK_FREQ 66355200 <= SUZAKU-V oscillator (18 times the 3.6864MHz by DCM) C BAUDRATE 115200 0

C USEPARITY

To view the data sheet of each peripheral, click Open PDF Doc.

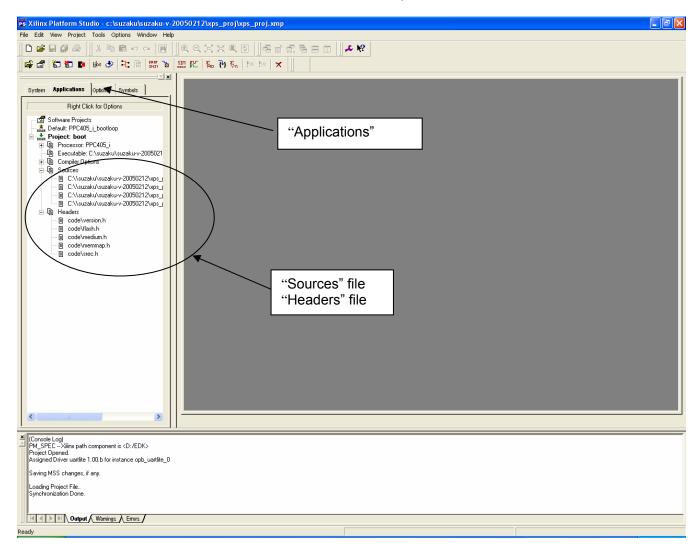
Lastly, click the OK button.



To change the BRAM software source, double-click the Sources or Headers files on the Applications tab to edit them directly or delete with "delete file" and then add a new file.

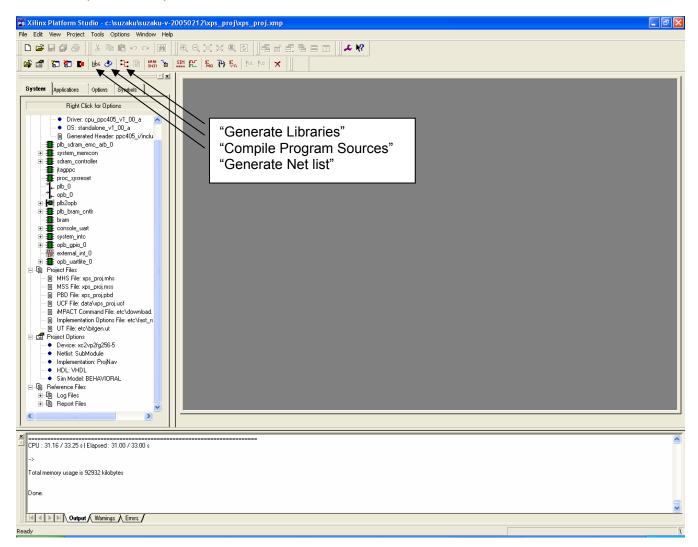
If using the files without making any changes, please proceed to the next step.

The default software source contains the initialization code required to run Hermit and Linux.



9.4. Platform Studio Compiling

Compile by clicking Generate Libraries, Compile Program Sources and Generate Net list in order. This completes all required tasks in Platform Studio.



9.5. Editing the Project Top File

The files automatically created in Platform Studio are then edited so that they are called by the project top file.

The files to be edited are top.vhd and top.ucf.

First return to the Project Navigator window and select Open from the File menu to open the following two files.

xps_proj\hdl\xps_proj_stub.vhd top.vhd

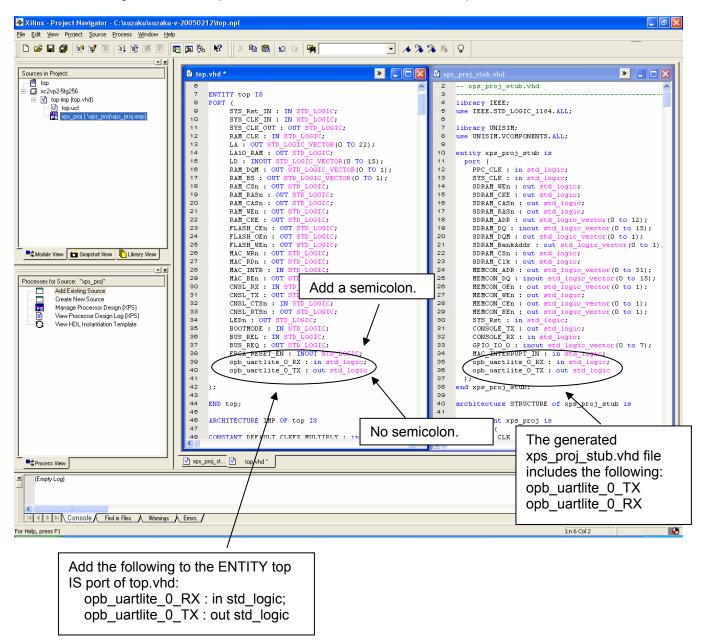
If compiling the default configuration without adding any peripheral cores, please move on to **Section 9.6**, **"Compiling a Project Navigator"**.

Open			? 🗙
Look in: 🔎	hdl	• 🗲 主	•
sdram_con	eset_wrapper.vhd troller_wrapper.vhd cc_wrapper.vhd emcon_wrapper.vhd	xps_proj_stub.vhd	
<			>
File <u>n</u> ame:	xps_proj_stub.vhd		<u>O</u> pen
Files of <u>t</u> ype:	Source Files (*.v,*.vhd,*.v	hdl,*.abl,*.abv,*.uc 💌	Cancel

Open					? 🗙
Look in: 🛅	suzaku-v-20050212	•	(÷ 🔁	Ċ	
Cprojnav c_xps_proj top.ucf top.vhd					
File <u>n</u> ame:	top.vhd				<u>O</u> pen
Files of type:	Source Files (*.v,*.vhd,*.vhd,*.abl,	*.abv,*	.uc 🔻		Cancel

The xps_proj_stub.vhd file is automatically created when the PowerPC405 and peripheral cores have been generated in XPS.

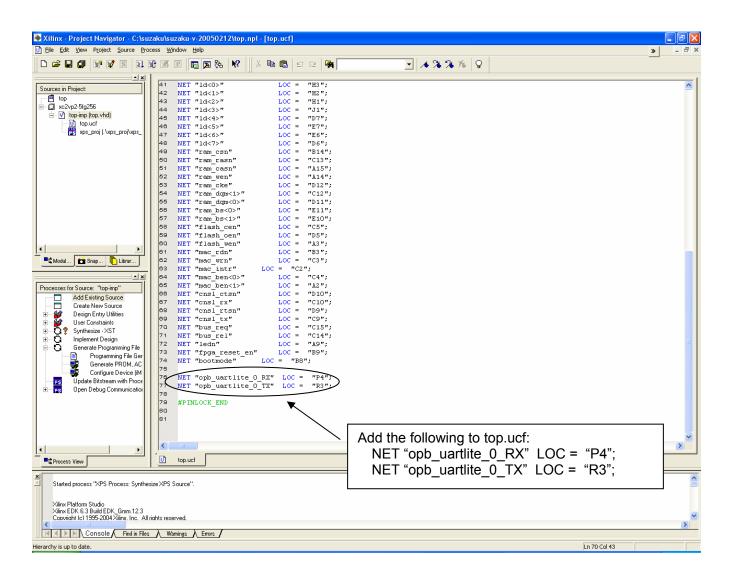
Write the following Port and Component names described in this file to top.vhd.



📚 Xilinx - Project Navigator - C:\suza	:aku\suzaku-v-20050212\top.npl	
File Edit View Project Source Process		
	\$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$	
Sources in Project:	top.yhd	
top	00 SDRAM_BankAddr: out std_logic;vector(0 to 1); 19 SDRAM_DO; inout std 07 SDRAM_CIN: out std_logic; 20 SDRAM_DON: out std 08 SDRAM_CIN: out std_logic; 21 SDRAM_BankAddr: out 09 MAC_INTERNUPT_IN: in std_logic; 21 SDRAM_SN: out 09 MAC_INTERNUPT_IN: in std_logic; 22 SDRAM_CIN: out 70 MEMCON_ADR: out std_logic;vector(0 to 1); 23 SDRAM_CIN: out 71 MEMCON_ECN: out std_logic;vector(0 to 1); 24 MEMCON_ADR: out std_logic; 73 MEMCON_CEN: out std_logic;vector(0 to 1); 26 MEMCON_OR: out std_logic;vector(0 to 1); 74 MEMCON_EEN: out std_logic_vector(0 to 1); 27 MEMCON_WEN: out std_logic_vector(0 to 1); 76 SDRAM_DQ_O: out std_logic_vector(0 to 15); 29 MEMCON_EEN: out std_logic 77 SDRAM_DQ_T: out std_logic_vector(0 to 15); 29 MEMCON_EEN: out std_logic 77 SDRAM_DQ_T: out std_logic_vector(0 to 15); 29 MEMCON_EEN: out std_logic 78 GPIO_TO_O_I: in std_logic_vector(0 to 7); 32 CONSOLE TX: cut std_logic 79 GPIO_TO_O_O_I out std_logic_vector(0 to 7); 32 <	LL; fic; logic; logic; logic; logic; logic vector (0 to 12); logic vector (0 to 15); logic vector (0 to 1); end logic vector (0 to 1); logic vector (0 to 1); logic vector (0 to 1); logic vector (0 to 1); logic vector (0 to 1); logic; logic vector (0 to 1); logic
Process View	to the COMPONET xp	s_proj of top.vhd
Empty Log)	λ Warnings λ Errors /	
For Help, press F1	V Annual V V	Ln 55 Col 2

💸 Xilinx - Project Navigator - C:\suzaku\suzaku-v-20050212\to	p. npl		
Eile Edit View Project Source Process Window Help			
		▼ <u>▲ % % %</u> ♀	
Sources in Project: 🚺 top. vhd *	»	💶 🔀 🛯 xps_proj_stub.vhd	» = = X
top 295 DSSEN =		A 1	
	> PPC_CLK_dl1_2x,	2 xps_proj_stub.vho	1
top.ucf 297 LOCKED	=> DLL_LOCK_o);	3	
xps_proj (.\xps_proj\xps_ 298		4 library IEEE; 5 use IEEE.STD LOGIC :	164 MLL:
300		6	
301 xps_proj_1: x	ps_proj	7 library UNISIM;	
302 PORT MAP (303 MEMCON C	En => MEMCON CEn,	8 use UNISIM.VCOMPONE	ITS.ALL;
	=> SYS CLK,	10 entity xps proj stub) is
	=> PPC_CLK,	11 port (
	E => RAM_CKE,	12 PPC_CLK : in sto	
	En => MEMCON_OEn, k => SDRAM Clk,	13 SYS_CLK : in sto 14 SDRAM WEn : out	
	RX => CNSL RX,	14 SDRAM_WEN : out 15 SDRAM_CKE : out	
310 SYS_Rst	=> SYS_Rst,	16 SDRAM CASh : out	
	En => MEMCON_WEn,	17 SDRAM_RASn : out	
	nkåddr => RåM_BS, n => RåM CSn,		<pre>std_logic_vector(0 to 12); std_logic_vector(0 to 15);</pre>
	n -> RAN_CSN, Sn => RAM RASn,		std_logic_vector(0 to 15);
	Sn => RAM_CASn,		out std logic vector (0 to 1);
	n => RAM_WEn,	22 SDRAM_CSn : out	
	M => RAM_DQM, TX => CNSL TX,	23 SDRAM_Clk : out 24 MEMCON ADR : out	<pre>std_logic; std logic vector(0 to 31);</pre>
OCESSES TOT SOURCE: Xps_proj 319 MEMCON A	DR => MEMCON ADR,		at std logic vector(0 to 31);
Add Existing Source 320 MENCON B	En => MEMCON BEn,		std_logic_vector(0 to 1);
Managa Processor Design MP 321 SURAN_AU	R => SDRAM_ADR,	27 MEMCON_WEn : out	
View Processor Design Log (XF 222 SDP AM DC	_O => SDRAM_DQ_O, I => SDRAM_DQ_I,		<pre>std_logic_vector(0 to 1); std_logic_vector(0 to 1);</pre>
	T => SDRAM DQ T,	30 SYS Rst : in sto	
325 MAC_INTE	RRUPT_IN => MAC_INTR,	31	
	Q_O => MEMCON_DQ_O,	Add a co	mma.
	$Q_I => MEMCON_DQ_I,$ $Q_T => MEMCON_DQ_T,$	33 34 HRC_INTERROPT_IN	- veccor(u co /);
	0 I => GPIO IO 0 I,		X : in std logic;
	0_0 => GPI0_10_0,	36 opb_uartlite_0	X : out std_logic
	lite_O_RX => opb_uartlite_O_RX, lite O_TX => opb_uartlite_O_TX	> 37); 38 a	
333 332 4000 uar 0	Tite_0_1X -> Opb_darcifice_0_1X	No comma	
334);	•		ps proj stub is
335		41	
336 Free R5232 337 CNSL RTSn <=		42 component xps_proj 43 port (is 🗸
		43 port (
	Add the	following:	
Process View View top.vh			
×I (Empty Log)	opb	uartlite_0_RX => opb_ua	artlite_0_RX,
	· _ · _	uartlite_0_TX => opb_ua	
	to the i	stance text of xps_proj_	i : xps proj of top.vhd.
Console Find in Files Warmings Errors		·	
or Help, press F1			Ln 295 Col 2

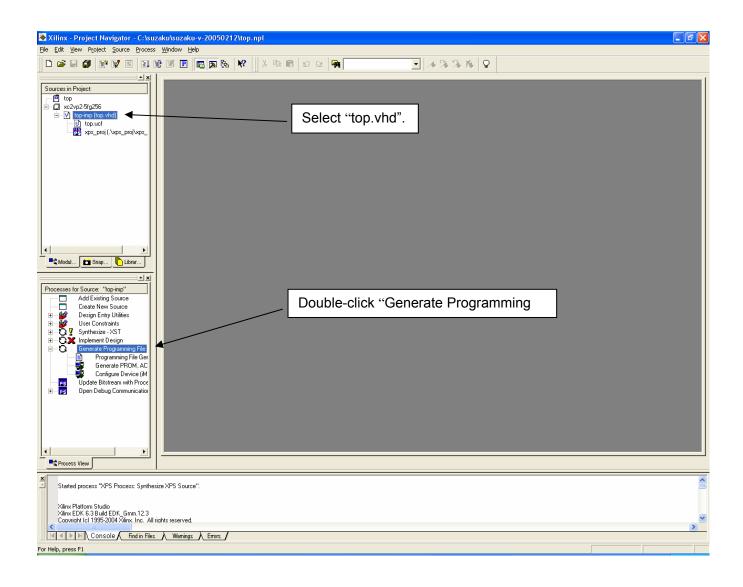
Open the \suzaku_v00\xps_proj\top.ucf file to assign signals to the free I/O pins of the Virtex-II Pro. In this example signals are assigned to #3-pin (P15) and #4-pin (P14) of CN5. This completes the editing of the project top file.



9.6. Project Navigator Compiling

Select top.vhd from the Souces in Project and then double-click Generate Programming File to compile it. This completes all required set up and compiling to generate the top.bit file.

(The generated top.bit file is used as the base file for FPGA configuration. To actually program the configuration, the top.bit file must be converted to a MCS file. Refer to "FPGA Configuration" in the next section).

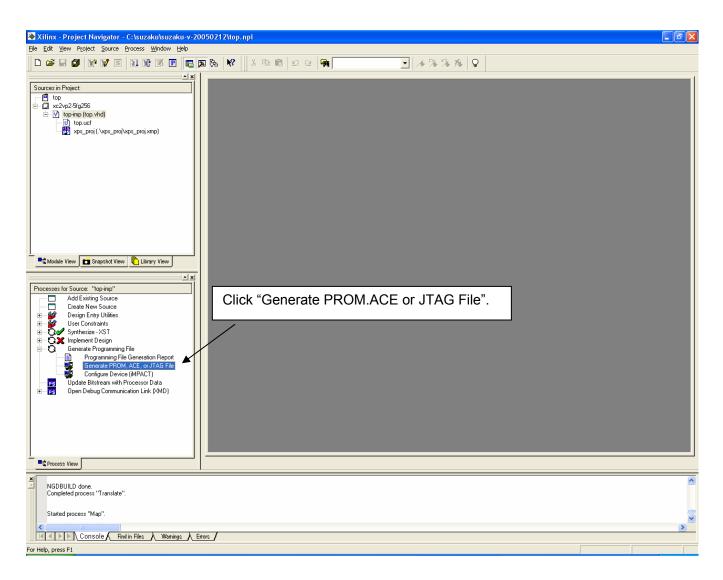


10. FPGA Configuration

This section describes how to configure the SUZAKU-V.

10.1. Converting a BIT File to a MCS File

Click Generate PROM.ACE or JTAG File.



Select PROM File and then click the Next button.

Prepare Configuration Files	
I want to create a : System ACE File PROM File Boundary-Scan File	
< <u>B</u> ack <u>N</u> ext > Can	cel Help

Select Xilinx Serial PROM. Specify the name of the PROM file and its location and then click the Next button.

Prepare PROM Files	
I want to target a : • Xiinx Serial PROM • Parallel PROM • Xiinx PROM with Design Revisioning Enabled • Compress Data • MCS • MCS • MEX • MCS • MEX • MCS • MEX • Memory Fill Value (2 Hex Digit): • Swap Bits Memory Fill Value (2 Hex Digit): • PROM File Name: • Location: • C:\suzaku\suzaku-v-20050212\	Here, the file name is set to: "top" And its location to: "c:\suzaku\suzaku-*****

Specify Xilinx PROM	Device 🔀	
Auto Select PROM		
Select a PROM:	xc18v xc18v04 <u>A</u> dd	
	0 xc18%04	cc18v04".
Number of Revisions:	Delete All	c18v".
	Back Next> Cancel Help	

Select xc18v 04. Click the Add button and then the Next button.

Click the Next button.

File Generation Summary	×
You have entered following information PROM Type: Serial File Format: mcs Fill Value: FF PROM Filename: top Number of PROMs: 1 Position Part Name 0 xc18v04	
Click 'Next' to add device file.	
< <u>B</u> ack <u>N</u> ext > Cancel Help	

Click Add File...

Add Device File	×
Data Stream : 0	
Starting Address (Max 8 Hex Digits) :	0
Mary start adding double (lafe) .	Add File
Now start adding device file(s) :	
(<u>K</u> ext >	Cancel Help

Select the generated BIT file and then click the Open button.

Add Device			?×
Look jn: 🔎	C:\suzaku\suzaku-v-20050212	💌 🕂 🗈 🕂 🎹	
projnav ngo xps_proj xst top.bit			
File <u>n</u> ame:	top.bit	<u></u> p	en
Files of type:	All Design Files	▼ Car	icel

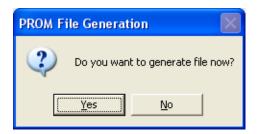
Click "No".

Add Device 🛛 🕅		
2	Would you like to add another design file to Data Stream: 0 ?	
	Yes No	

Click Finish.

Add Device File		
Data Stream : 0		
Starting Address (Max 8 Hex Digits) :	0	
Now start adding device file(s):	Add File	
Click 'Finish' to start generating file.		
Click 'Cancel' to go to user screen.		
< <u>B</u> ack (Finish)	Cancel Help	

Click Yes.



This completes the Bit file to MCS file conversation.

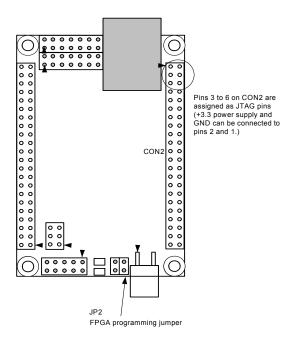
10.2. Programming Configuration Data to SUZAKU-V

This section explains how to actually program the SUZAKU-V with the MCS file generated in the previous section. The SUZAKU-V uses the TE7720 (developed by Tokyo Electron Device Limited) for FPGA configuration. For details on the operation of the TE7720, refer to Section 4.4.9, "FPGA Configuration".

You can also download detailed materials on the TE7720 from the home page of Tokyo Electron Device Limited.

First, copy the new MCS file to the same folder in which device.def and lbplay2.exe are stored. (device.def and lbplay2.exe are stored in the same folder as the project top file, suzaku-v-*****).

- Make sure that the SUZAKU-V is powered off.
- Connect a JTAG cable such as the Xilinx Parallel Cable to the FPGA Programming JTAG at CON2.
- Short the JP2 jumper.



• Power on the SUZAKU-V.

• Open a command prompt and proceed to the folder in which the above MCS file, device.def and lbplay2.exe are stored.

• Enter lbplay2 –deb top.mcs

• If the following error message is generated, install the appropriate driver as described in the \fpga_proj \lbplay2 driver install.txt in the supplied CD-ROM. ERROR: Please check WINNT\system32\drivers\windrvr.sys.

- 🗆 X

🔤 Command Prompt

D:\suzaku\suzaku-20050115>lbplay2 -deb top.mcs LittleBearPlayer2 0.17 mcs file1 = top.mcs *** WinNT mode *** MaxDeviceNumber=1 Please Hit Enter. (ESC:quit)
Device 1 start. It is being erasedDone.
SendByte :212392
VerifyByte:524288 CheckSum (writedata) : c396 CheckSum (verifydata) : c375
Execution Time = 34.016 seconds
D:\suzaku\suzaku-20050115>

• When the programming is complete, verify that the checksum is correct and power off the SUZAKU-V. If an error occurs due to some reason or the checksum is wrong, do not run the SUZAKU-V. If incorrect data is programmed or an error is generated during programming, turn off the power switch and short JP2 to perform reprogramming.

- Open the JP2 jumper.
- Turn on the SUZAKU-V.
- The SUZAKU-V will boot with the newly programmed configuration data.

In this example the UART is pin-assigned to the external I/O, resulting in +3.3V input/output level. Because of this, it cannot be directly connected to a standard PC RS232C. For details on the RS232C connection method, refer to the How to page at the following site:

SUZAKU Official Site: http://suzaku.atmark-techno.com/

10.3. Dealing With Lbplay2.EXE "ERROR: Please check WINNT\system32\drivers\windrvr.sys."

The following explains what to do if "ERROR: Please check WINNT\system32\drivers\windrvr.sys" appears when using Lbplay2.EXE.

1. Decompress the \fpga_proj\ wb_TE7720_software_20031104.zip file from the supplied CD-ROM. Within the decompressed folder, there will be a file named Release204.zip. Please decompress this file as well.

2. After confirming there is no file with the same name, copy windrvr.sys to the following folder under a user with Administrator privileges.

- For WindowxNT/2000: C:\WINNT\system32\drivers
- For WindowsXP C:\WINDOWS\system32\drivers
- 3. Open a command prompt window, move to the decompressed folder and execute "wdreg install"

These steps will allow Lbplay2.EXE to function properly.

For more detailed information on Lbplay2.EXE or the TE7720, please see Tokyo Electron Device's home page, http://www.teldevice.co.jp/

R	evision History	
Ver.	Date	Description
1.0.0	2005/02/12	Initial release
1.0.0.1	2007/06/15	Made multiple corrections to the memory addresses in Table 5-1, "SUZAKU-V Memory Map"

SUZAKU-V Hardware Manual

June 15, 2007 version 1.0.0.1

Atmark Techno, Inc. AFT Building 6F, North 5 East 2, Chuo-ku, Sapporo, Hokkaido 060-0035 TEL: 011-207-6550 FAX: 011-207-6570