SDB-750/1000

ProASIC^{PLUS} Development System

User Guide

Version 1.3

Document No. 41397.50.13

INICORE INC.
5600 Mowry School Road
Suite 180
Newark, CA 94560
t: 510 445 1529 *f:* 510 656 0995 *e:* info@inicore.com
www.inicore.com

INICORE, INC. has made every attempt to ensure that the information in this document is accurate and complete. However, INICORE, INC. assumes no responsibility for any errors, omissions, or for any consequences resulting from information included in this document or the equipment it accompanies. INICORE, INC. reserves the right to make changes in its products and specifications at any time without notice.

Any software that is described in this document is furnished under a license or non-disclosure agreement. It is against the law to copy this software on any media for any other purpose than the licensee's personal use.

Acknowledgments:

TABLE OF CONTENTS

TABLE OF FIGURES

1. OVERVIEW

This User Guide was designed for the engineer to provide technical information of the SDB-750/1000 development platform. This guide should allow the user to get a quick start and solid introduction into the world of ProASIC^{PLUS} based systems.

This guide is divided into three sections:

1. The getting started guide that helps with the unpacking and setting up of the system.
2. Detailed explanations of all system components and features.
3. Explanations of the system software that comes installed with the SDB-750/1000 development platform.

The application notes section at the end provides additional information how the SDB-750/1000 development platform can be used to program the ProASIC^{PLUS} FPGA.

1.1. Document History

The following table gives an overview of the document history and can help in the determination if the latest version of this document has been used.

Version	*Date*	*Author*	*Comments*
1.0	2/21/02	DL	Initial version
1.1	5/2/02	JPM	Completely updated version with new chapters
1.2	5/7/02	JPM	Minor updates after verification
1.3	5/9/03	HJK	Corrected memory sizes, added schematic

1.2. References

The following reference documents have been used. Most of them are available on the CD-ROM that were delivered with the SDB development platform. If they are not available with the appropriate institutions they can be requested via Inicore for an additional fee.

- CompactPCI, The PCI Telecom Mezzanine/Carrier Card (PTMC) Specification, PICMG2.15 R1.0, 4/11/2001, © Copyright 2001, PCI Industrial Computer Manufacturing Group
- Standard for a Common Mezzanine Card Family: CMC, P1386, D2.4a, March 2001, © Copyright 2001, Institute of Electrical and Electronic Engineers, Inc.
- Standard Physical and Environmental Layers for PCI Mezzanine Cards: PMC, P1386.1, D2.4, January 2001, © Copyright 2001, Institute of Electrical and Electronic Engineers, Inc.
- AT91X40 Series, ARM® Thumb® Microcontrollers, Rev. 1354B-07/00, © Copyright 2000, ATMEL
- ProASIC^{PLUS}™ APA Family Product Profile, Advanced v0.3, December 2001, © Copyright 2001, Actel Corporation
- SX-A Family FPGAs, Preliminary v1.1, September 1999, © Copyright 1999, Actel Corporation
- 28F320J3A, 3-Volt® StrataFlash® Memory, Datasheet Order # 290667-010, © Copyright 2001, Intel Corporation
- 3.3V CMOS Static RAM 4 Meg (512K x 8-Bit), DSC-3622/03, © Copyright 1999, Integrated Device Technology, Inc.
- CompactFlash™ Memory Card Product Manual, Lit. No. 20-10-00038 Rev. 7 4/2000, © Copyright 2000, SANDISK CORPORATION
- PCA82C250 CAN Controller Interface, 1997 Oct 21, © Copyright 1997, Philips Semiconductors
- AND Application Notes, Intelligent Alphanumeric Displays, 12/17/99, © Copyright 1999, Purdy Electronics Corporation
- I^2C bus SERIAL INTERFACE REAL-TIME CLOCK ICs, RS5C372A/B, Application Manual, NO.EA-04409908, © Copyright 1995, Ricoh Electronic Devices Division
- MAX322x datasheet, 19-0306; Rev 6; 3/99, © Copyright 1999, MAXIM Integrated Products
- Micropower Inverting DC/DC Converters, © Copyright 1999, Linear Technology

1.3. Naming Convention

The following two sections define the acronyms and definitions that have been used within this user guide. For additional information please refer to the corresponding documentation either through the respective agencies or via the Internet.

1.3.1. Acronyms

Within this section we defined the acronyms that have been used throughout this document.

Acronym	Explanations
ARM	Advanced Risk Machine
CF	Standard CompactFlash device that is supported by the SDB development board.
GC	GlueChip: Preprogrammed FPGA that contains the system logic
GND	Ground signal
GPIO	General purpose input and/or output port
Mb	One Mega-bit is 1024 bits or 128 Bytes
MB	One Mega-Byte is 1024 Bytes or 8096 bits
PA	ProASIC^{PLUS} FPGA from Actel Corporation
PCI	Peripheral Component Interconnect
PMC	PCI Mezzanine Card
PTMC	PCI Telecom Mezzanine Card
RCK	Running clock signal that is used to program the ProASIC^{PLUS} FPGA.
SMLink	Surface mounted link that are represented by 0Ω resistors
STAPL	Standard Test and Programming Language
UPI	Microprocessor interface
VDD12	12V power supply signal, only used in connection with the PMC Mezzanine expansion card
VDD25	2.5V power supply signal
VDD33	3.3V power supply signal
VDD50	5.0V power supply signal
VDDCF	Power supply signal for CompactFlash card
VDDDCDC	Power supply signal for DC-DC converter
VDDL	Core voltage signal of the PA FPGA. Usually connected to VDD25
VDDN12	Negative 12V power supply signal, only used in connection with the PMC Mezzanine expansion card

Acronym	Explanations
VDDNP	Negative programming voltage signal that is used for the ProASIC^{PLUS} devices
VDDP	SDB-750/1000 voltage signal of the PA FPGA. Usually connected to VDD33
VDDPP	Positive programming voltage for ProASIC^{PLUS} devices

1.3.2. Definitions

This section should simplify the usage of this document by providing a single place where common definitions are defined for the reader.

Definitions	Explanations
CAN	Controller Area Network, which is a serial data bus system developed by Bosch for the automotive industry. Subsequently it was standardized internationally (ISO11898). The bus itself is a symmetric or asymmetric two wire circuit, which can be either screened or not screened.
eCos	Embedded micro controller operating system that is based on the open standard featured by RedHat Corporation
GCC	GNU open source C compiler that comes with standard Linux installations
GDB	GNU open source debugger for C as part of standard Linux installations
I2C	Inter-IC, a type of bus designed by Philips Semiconductors in the early 1980s, which is used to connect integrated circuits (ICs). I2C is a multi-master bus, which means that multiple chips can be connected to the same bus and each one can act as a master by initiating a data transfer
LVPECL	Low Voltage Positive Emitter Coupled Logic or Low Voltage Pseudo Emitter Coupled Logic which is a differential mode transmission standard that allows for fast signal transmissions
SDB	ProASIC^{PLUS} system design and demonstration board with either 750k (SDB-750) or 1,000k system gates (SDB-1000)
SDRAM	Synchronous DRAM, a type of DRAM (dynamic RAM) that can run at much higher clock speeds than conventional memory
SRAM	Static random access memory that doesn't need any refresh cycles
UART	Universal asynchronous receiver-transmitter, the UART is a chip that handles asynchronous serial communications

1.3.3. Conventions

Throughout this document the following naming conventions are used:

Convention	Explanations
'_n'	Signal names with this convention indicate an active LOW signal.
r/W	Read back operations are executed to access static registers that don't change their information through external events. Write operations are performed to change the register status.
R/W	Read operations are executed to gain new information on the appearance of external events that might have a changed a register status. Write operations are performed to change the register status.
⊗	Entries market with this symbol indicate the default settings

1.4. Features

The SDB development board provides a feature rich prototyping platform for the ProASIC^{PLUS} FPGA family. It allows developers to gain expertise in the specifics of embedded design using the ProASIC^{PLUS} FPGA combined with an ARM based microcontroller.

In particular the SDB supports the following features:

- Easy-to-Use development board that functions without additional hardware or software.
- Contains up to 1,000,000 system gates ProASIC^{PLUS} FPGA from Actel (SDB-1000) that can be reprogrammed using the ARM CPU without external hardware.
- 2MB SRAM and 128Mb Flash memory that is accessible either by the ARM CPU or via ProASIC^{PLUS}
- Simple user interface with LCD (2x16 characters), six programmable LED's that provide programmable indications of events and four input keys to control the user interface
- ARM7TDMI subsystem that uses the AT91M40800 CPU from AMTEL
- Standard PC100 DIMM interface with 64-bit wide SDRAM
- Standard Compact Flash card slot for memory cards
- Standard PMC expansion bus that supports:
  - IEEE P1386.1
  - PCIMG 2.15 (PTMC)
  - 64-bit PCI interface
- Four different serial I/O interfaces such as UART, I2C, CAN and the analog LVPECL standard
- 16 x 16 pins prototyping area with prewired signals from the ProASIC^{PLUS}
- Support for system expansion board with standard PMC Mezzanine card for telecommunications applications
- Independent real time clock with backup battery
- eCos operating system from RedHat

1.5. Block Diagram

The following block diagram gives a high level overview of the SDB development board.

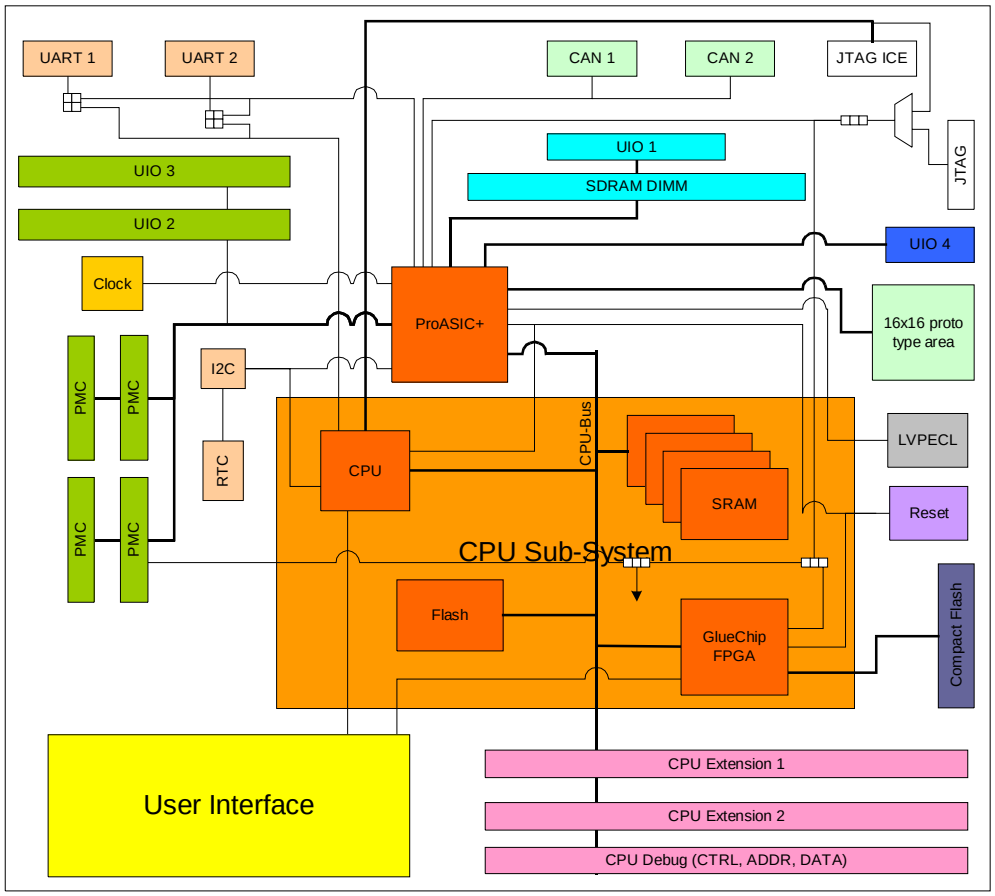


Figure 1 - SDB Block Diagram

The center of the SDB is the ProASIC^{PLUS} FPGA that is connected with the CPU subsystem via the CPU-Bus. This connectivity allows the ProASIC^{PLUS} to be programmed as the main CPU, therefore controlling all major functions of the board.

Furthermore, the ProASIC^{PLUS} is connected to all external interfaces in order to embed the programmable logic into any external system architecture without difficulties or special adaptations.

1.6. Layout

The high level board layout is shown in the following figure.

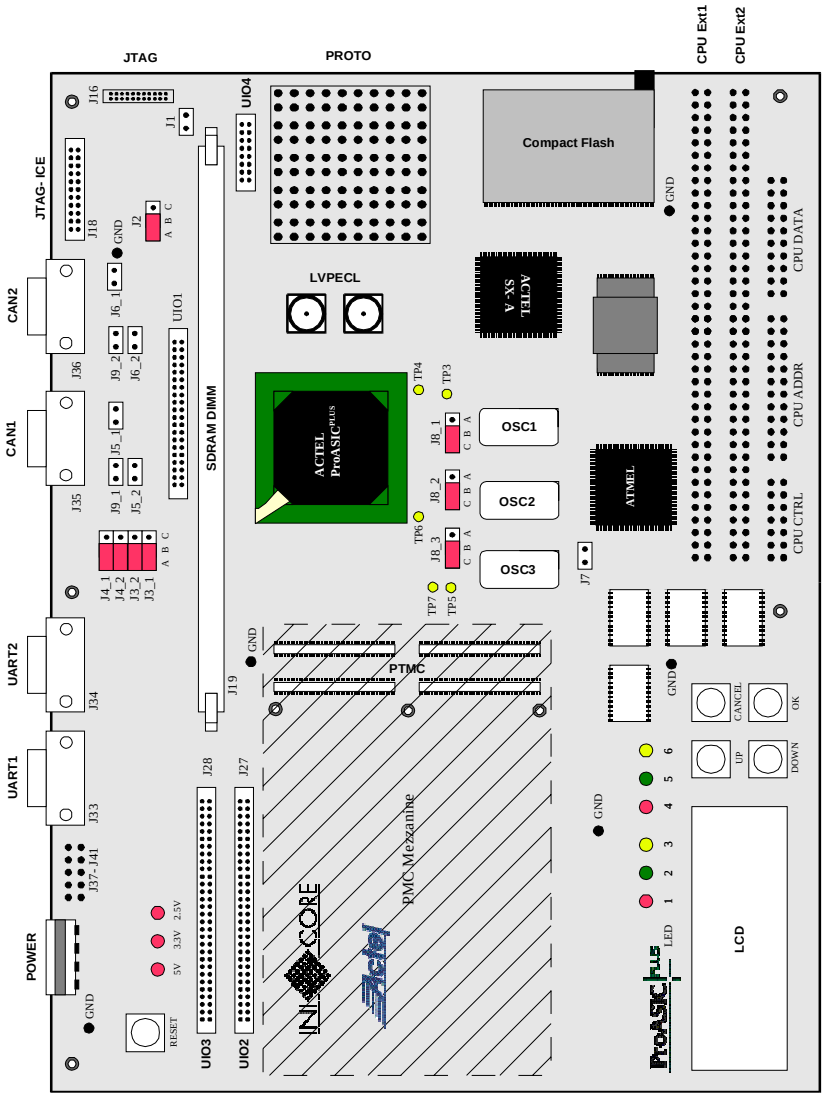


Figure 2 - SDB System Layout

With the above layout we can see that there is additional space on the SDB development board to include an external PMC Mezzanine card.

In addition the SDB development board contains a 16x16 through-hole matrix for add-on electronic or additional connectors.

2. GETTING STARTED

This section provides the new SDB user with all essential information so that the development board can be connected and initialized successfully.

In addition we describe what should have been delivered with the new SDB development board and provide system installation information.

2.1. Package Content

The following figure gives an overview what should have been delivered with your new SDB development board.

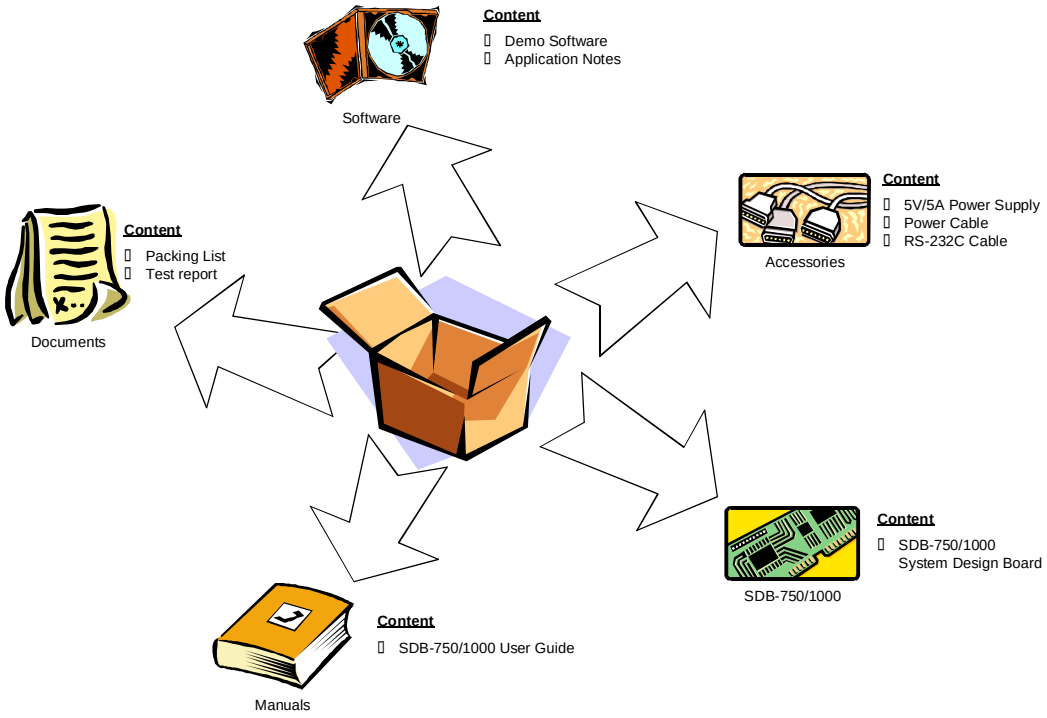


Figure 3 - SDB Packaging Information

2.1.1. Documents

The following documents should have delivered with your new SDB development board:

- A detailed packing list that list the specific content of your delivery
- Test report of SDB-750/1000 development board

2.1.2. Software

The software for the SDB development board is already installed. It contains a Boot Monitor and software that supports the standard operations of the board.

In addition to the already installed software, there is a CD-ROM that contains the following information:

- Electronic version of the documentation
- Necessary files that allow re-creating the manufactured configuration

Please visit http://inicore.com regularly for latest files, additional support information and application notes.

2.1.3. Accessories

The accessories that were delivered with the SDB development board are the following:

- 5V / 4A 110V/230V Power supply (20W)
- US power cord
- 6' long RS-232C serial interface cable

2.1.4. SDB development board

The SDB development board that contains either the ProASIC^{PLUS} FPGA with 750,000 system gates or the ProASIC^{PLUS} FPGA with 1,000,000 system gates.

2.1.5. Manuals

The SDB development board user guide is the main documentation that is delivered with the package. It contains all necessary information to get a new user started quickly.

2.2. Additional Information

Refer to the following web-sites for additional information:

- INICORE, INC. home page at http://www.inicore.com
- Actel Corporation home page for FPGA information at http://www.actel.com
- ATMEL Corporation home page for CPU information at http://www.atmel.com
- RedHat Corporation for additional information on the operating system eCos at http://www.sources.redhat.com/ecos

2.3. System installation

This section contains important information for the setup and installation.

2.3.1. Board setup

The following figure gives a high level overview of the board setup.

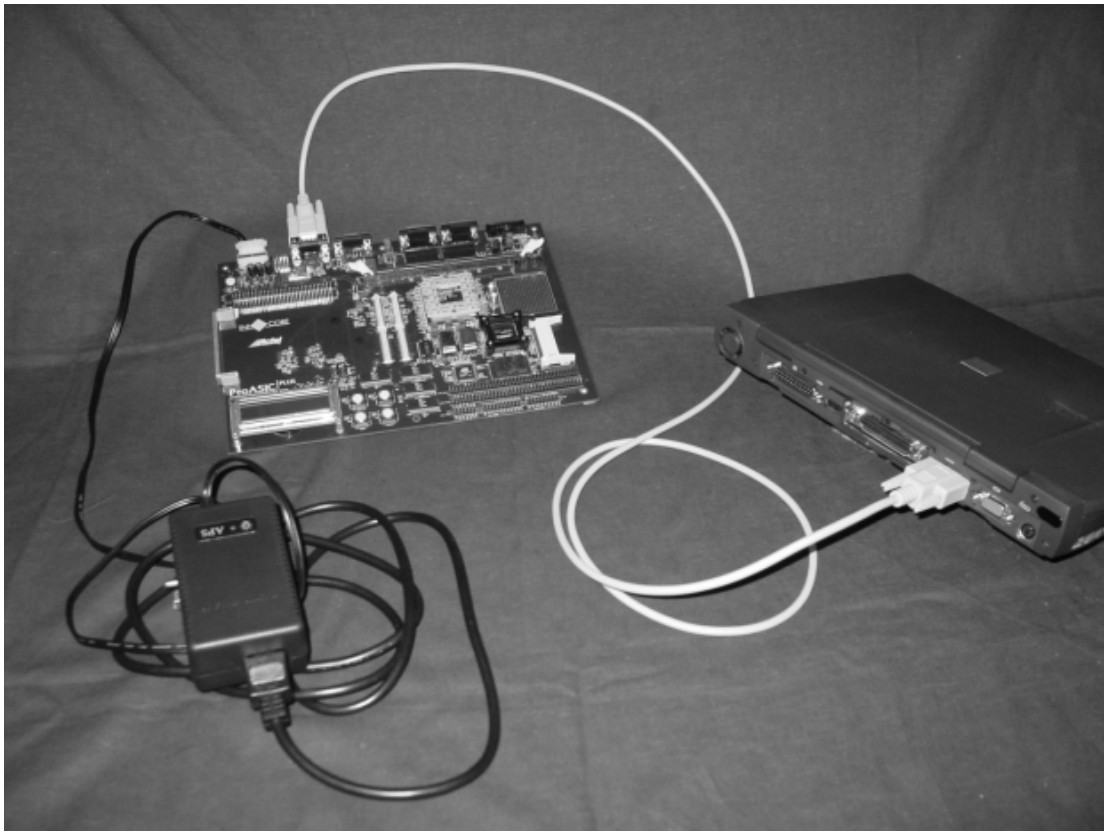


Figure 4 - Board setup

The above illustrated test set up shows the SDB development board with the accompanying power adapter and RS 232 cable connected to a serial port of the PC.

When connecting the RS 232 cable, please use the UART1 connector for easy set up and applying the pre-configured board settings.

2.3.2. Power supply

The SDB development board is powered through a regular disk drive power connector located in the upper left hand corner of the board.

For general purpose operations only a +5V DC power source is required that was included in the package. The following figure shows how the power is connected to the SDB development card with the regular disk drive connector [Jameco P/N: 42067]

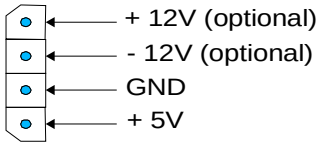


Figure 5 - Power Connector

In case the user would like to work with a PCM Mezzanine card (separate item) that needs to be supplied with ±12V DC, the included power supply will have to be replaced.

The on-board LDOs are powered through the VDD50 signal and have the following maximum rating:

- VDD33: 5A
- VDD25: 1.5A

Please evaluate carefully your power budget and verify that you are within the given limits and that the current rating of your power supply is sufficient.

2.3.3. Host connection

The connection to the host computer is achieved by using an RS-232C serial cable that is connected to a PC, UNIX or Linux machine.

The SDB development board contains two serial RS-232C ports. They can be either connected to the CPU or the ProASIC^{PLUS} FPGA device. The selection is done through the configuration of a jumper block as indicated in the following figure.

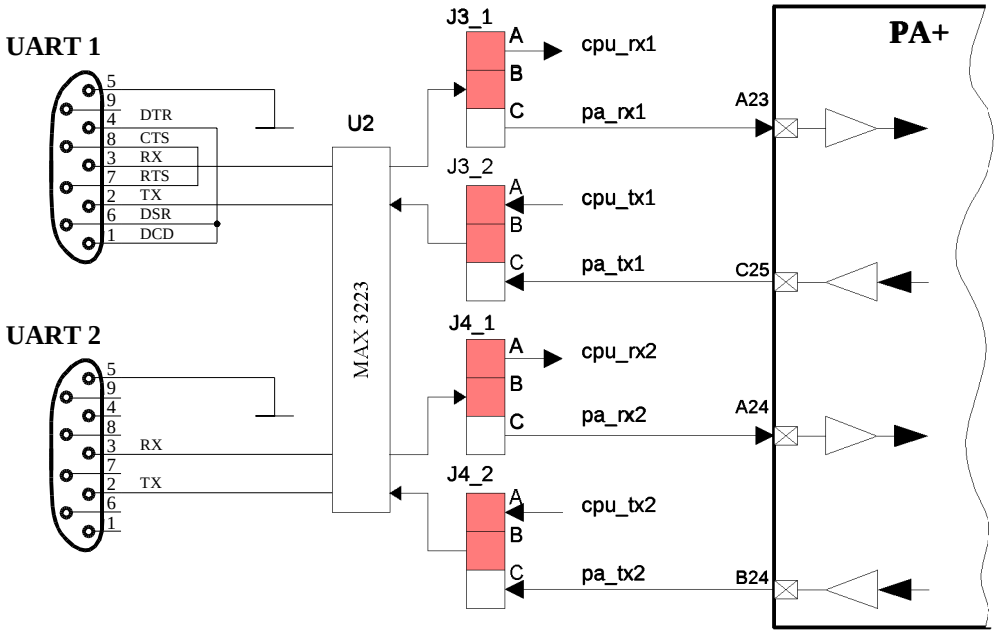


Figure 6 - RS-232C connection

As seen in the figure above, the UART 1 connector is designed for 'no hardware handshaking', e.g., RTS and CTS are connected together as well as DCD, DSR and DTR.

The UART 2 connector doesn't have the RTS, CTS, DCD, DSR and DTR lines connected. Therefore a simple protocol with only RX and TX is used while using a software handshake.

2.3.4. Basic Jumper setting

The following basic jumper settings for J3, J4, J7 and J8 should be verified as specified in the following tables.

2.3.4.1. J3: UART 1 Source Select

The following two jumpers can be used to select the UART device that is connected to the RS-232 transceiver. This can either the be the UART from the CPU or the ProASIC^{PLUS}.

J3_1	*Function*	*Setting*
A-B	The CPU UART port is used for receiving signals	⊗
B-C	The ProASIC^{PLUS} UART port is used for receiving signals	

J3_2	*Function*	*Setting*
A-B	The CPU UART port is used for transmitting signals	⊗
B-C	The ProASIC^{PLUS} UART port is used for transmitting signals	

Note: Please configure both jumpers for the same signal source otherwise your interface will not function properly.

2.3.4.2. J4: UART 2 Source Select

These jumpers can be used to select the UART macro that is connected to the RS232 transceiver. This macro can be either a software routine that runs on the CPU or as an embedded function within the ProASIC^{PLUS} .

J4_1	*Function*	*Setting*
A-B	The CPU UART port is used for receiving signals	⊗
B-C	The ProASIC^{PLUS} pre-configured UART port is used for receiving signals	

J4_2	*Function*	*Setting*
A-B	The CPU UART port is used for transmitting signals	⊗
B-C	The ProASIC^{PLUS} pre-configured UART port is used for transmitting signals	

Note: Please configure both jumpers for the same signal source otherwise your interface will not function properly.

2.3.4.3. J7: CPU Disable

This jumper controls if the CPU will be enabled or disabled during the startup process. If the CPU is disabled, the ProASIC^{PLUS} FPGA or an external source can control the system resources. However, the initial setting should allow for the CPU to be operational.

J7	*Function*	*Setting*
None	The CPU is operational	⊗
A-B	The CPU is disabled	

2.3.4.4. J8: Clock Jumpers

Four different clock sources are available to control the operation of the SDB development platform.

Clock sources:

1. OSC1: Socket for DIP-14 type clock oscillator with typical frequency of 30MHz.
2. OSC2: Socket for DIP-14 type clock oscillator with typical frequency of 60MHz.
3. OSC3: DIP-14 type LVPECL oscillator with typical frequency of 152.52MHz. This is a dedicated network, where the LVPECL oscillator is directly connected to the LVPECL inputs of the ProASIC^{PLUS} FPGA. The impedance of the transmission line is matched with a 50 Ù termination.
4. CLK32k: This is the 32kHz clock from the RTC block.

The clock source for the different clock networks can be selected using jumpers. This is shown in following diagram:

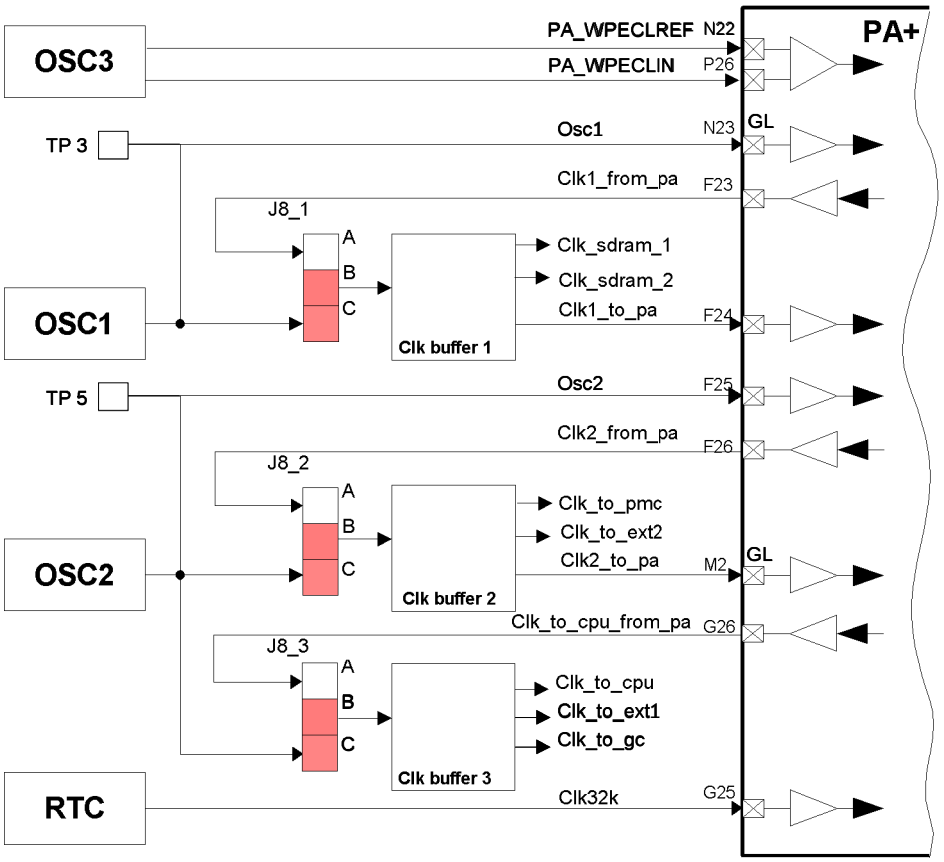


Figure 7 - Clock distribution network

Using the three jumpers that are explained in the following tables, different clock configuration can be chosen. The above figure indicates the default setup for the jumpers.

J8_1	***Function***	***Setting***
A-B	The Clock buffer 1 is driven by the clk1_from_pa signal.	
B-C	The Clock buffer 1 is driven by the osc1 signal	⊗

J8_2	***Function***	***Setting***
A-B	The Clock buffer 2 is driven by the clk2_from_pa signal	
B-C	The Clock buffer 2 is driven by the osc2 signal	⊗

J8_3	***Function***	***Setting***
A-B	The Clock buffer 3 is driven by the clk_to_cpu_from_pa signal	
B-C	The Clock buffer 3 is driven by the osc2 signal	⊗

Note: Please configure all three jumpers for the desired signal source otherwise the clock interface will not function properly.

3. MAIN BUILDING BLOCKS

This section describes the main building blocks as seen in the block diagram in the first section of this document. This section will provide all necessary information for the designer to start creating his/her own system designs.

3.1. ProASIC^{PLUS} and CPU Sub-System

In this section we will highlight the functionality of the ProASIC^{PLUS} and the CPU Sub-System as indicated in the figure below.

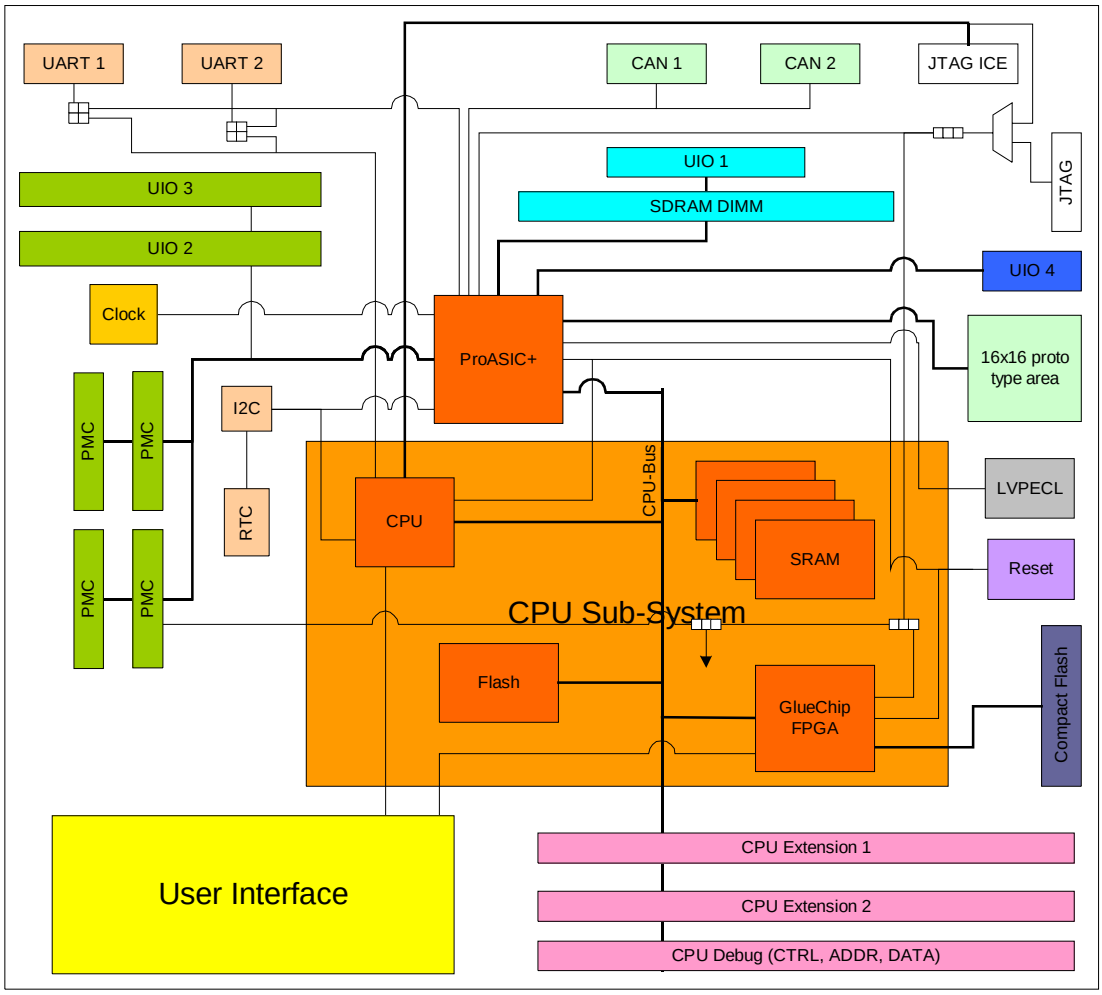


Figure 8 – PA+ & CPU Sub-System Block

As seen in the figure above, the CPU Sub-System contains the following devices that are connected through the CPU-Bus.

- The ProASIC^{PLUS} FPGA that contains the custom logic.
- The ATMEL CPU which controls the SDB functions such as operating system, timer functions, program downloads, etc. when using the standard configuration.
- The GlueChip that controls functions such as SRAM decoding, CompactFlash interface functions, programming reference clock, etc.
- The Flash memory chip that contains the operating system software.
- The SRAM that is used for the system software or as external memory for system functions.
- Two CPU Extension connectors that provide access to the entire CPU-Bus and allow to control the SDB functionality if configured appropriately.
- Three connectors that provide the necessary CPU signals so a logic analyzer can be connected for system analysis and debugging.

The following sections are explaining these components more in detail.

3.1.1. ProASIC^{PLUS} FPGA (750k/1000k System Gates)

The ProASIC^{PLUS} FPGA for the SDB is delivered in two different sizes. The smaller chip contains 750,000 system gates, whereas the bigger size chip comes with 1,000,000 system gates.

This chip combines the advantages of using ASIC's for system designs while having the benefit of using a device that is programmable through nonvolatile Flash technology. With such a device the engineer can create high-density systems using existing ASIC or FPGA designs and/or tools.

3.1.1.1. Block Diagram

Whereas the ProASIC^{PLUS} FPGA allows great flexibility for system designs, using the SDB development board offers the following pre-configured ports. These ports are wired to the appropriate devices, as indicated in the block diagram above, allowing a simplified design process.

The following figure gives an overview of the ProASIC^{PLUS} FPGA with its predetermined functionality on the SDB.

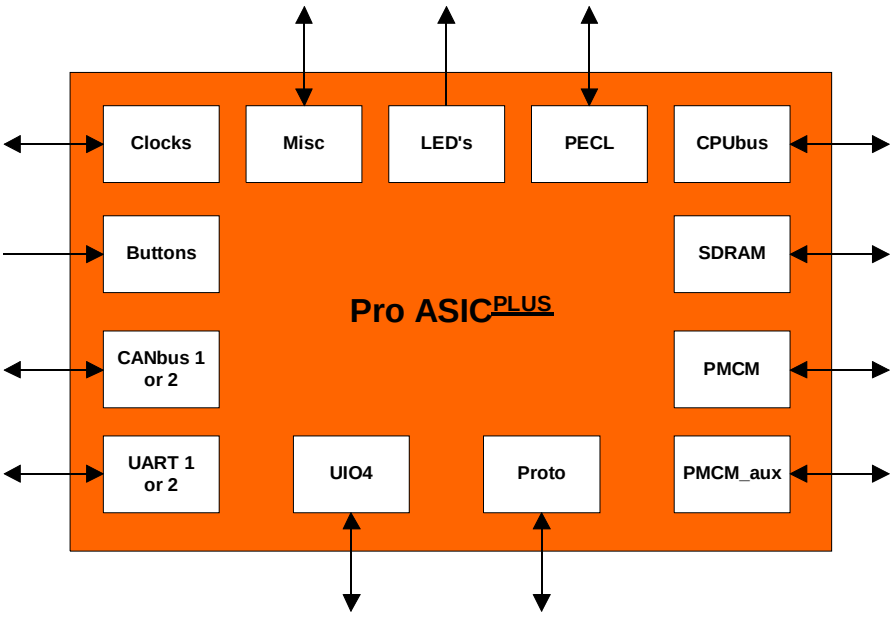


Figure 9 - Block Diagram ProASIC^{PLUS} FPGA

Note: The above mentioned functional blocks are only one possibility on how to use the ProASIC^{PLUS} FPGA. The VHDL/Verilog code for these interface blocks are not included with the board.
By using the connectors on the SDB development board much of this functionality can be replaced and/or customized to the specific needs of the system design.

3.1.1.2. Signal Description

The ProASIC^{PLUS} signal description for the specific layout of the SDB development board can be seen in the table below.

Signal	*Width [# of pins]*	*Description*
CPUbus	52	CPU Bus
SDRAM	96	SDRAM DIMM port
PMCM	61	Standard PMC Mezzanine port
PMCM_aux	92	Standard PMC Mezzanine auxiliary port
Proto	19	Signals for the 16x16 external prototyping area
UIO4	9	User definable I/O port 4 that is connected with the UIO4 connector.
UART 1,2	4	UART port (RX, TX only) for either the UART 1 or 2 connector.
CANbus 1,2	4	CANbus port for either the CAN 1 or CAN 2 connector.
Buttons	4	2x2 Keyboard matrix
Clocks	8	Clock signals
Misc	2	Reset and Flash memory status signals
LEDs	3	Control signals for LEDs, (red/green)
PECL	4	Standard two pin analog PECL interface

Of the above mentioned signals, some are shared with the I/O ports of the ProASIC^{PLUS} FPGA. The following table gives the overview of the shared signal ports.

Port	*Signal names*	*Description*
UIO1	sdram_data<63:32>	User definable I/O port 1 that is connected with the UIO1 connector.
UIO2	PMC_P3<45:0>	User definable I/O port 2 that is connected with the UIO2 connector.
UIO3	PMC_P4<45:0>	User definable I/O port 3 that is connected with the UIO3 connector.

Additional information on the ProASIC^{PLUS} FPGA can be found in the corresponding specification that is mentioned within the reference section.

For further information regarding programming tools, please refer to Actel's website at http://www.actel.com.

3.1.2. ATMEL CPU

The following block diagram gives an overview of the ATMEL CPU.

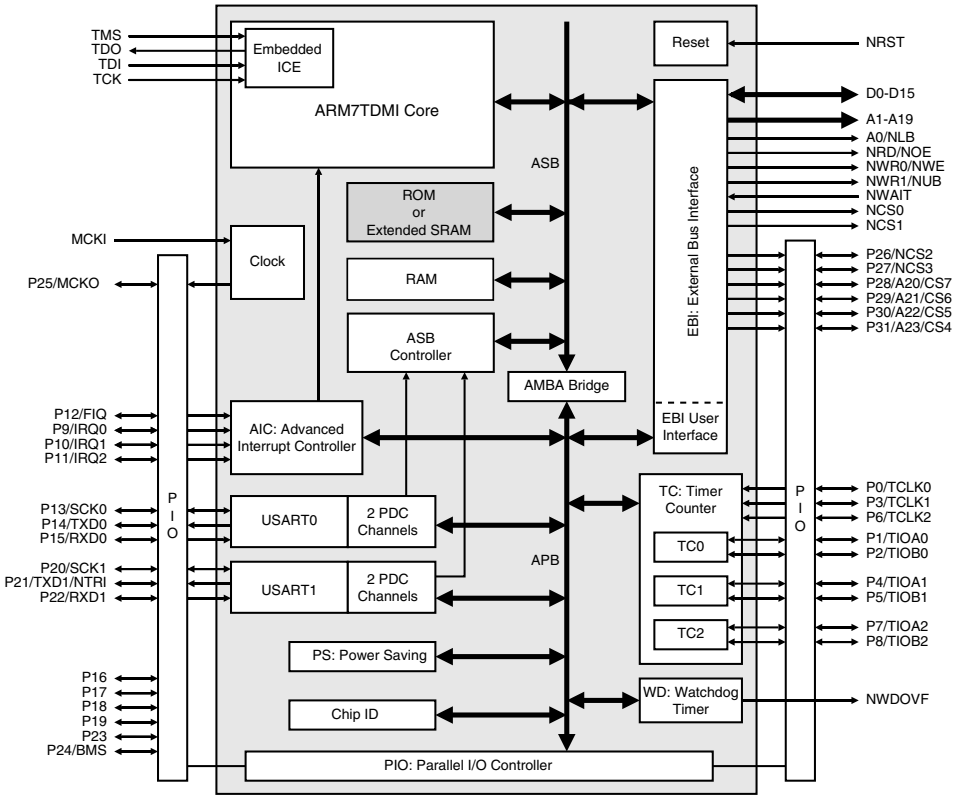


Figure 10 - Block Diagram AT91X40 Series CPU (© 2000 ATMEL Corporation)

The ATMEL AT91x40 family of controllers are ARM based. This controller uses the ARM7DTMI macrocell and is complemented with a standard set of peripheral functions such as Timers, UARTs, EBI and Interrupt Controller.

One special feature of this CPU that is used with the SDB development board are the flexibility to put the entire CPU into an inactive state (tri-state mode) by using one external jumper (J7). When this jumper is activated, it enables the use of either one of the following:

- Use a separate external CPU board that is connected to the CPU extension connectors 1 or 2.
- Use a ProASIC^{PLUS} FPGA with an integrated CPU macrocell as a system-on-chip design.

Additional information on the ATMEL AT91x40 CPU can be found in the corresponding specification that is mentioned within the reference section.

3.1.2.1. Bus & I/O Signals

The CPU Bus signals that are used on the SDB development card are explained in the table below.

Signal Name	*Width [# of pins]*	*Description*	*Activation*
cpu_d<15:0>	16	Data bus	<High>
cpu_a<23:1>	23	Address bus	<High>
cpu_cs<3:0>_n	4	Chip selects	<High>
cpu_wr_n	1	Write enable	<Low>
cpu_oe_n	1	Output enable	<Low>
cpu_ub_n	1	Upper byte enable	<Low>
cpu_lb_n	1	Lower byte enable	<Low>
cpu_wait_n	1	CPU wait	<Low>
cpu_firq_n	1	Fast interrupt request	<Low>
cpu_irq_n<2:0>	3	Interrupt request	<Low>
Total	**52**	The number of pins that are used for the CPU Bus	

The following pins are used to control the operation of the SDB development board. These signals are also available on the CPU Extension 1 and 2 connectors for external CPU boards.

Signal Name	*Width [# of pins]*	*Port*	*Description*
cpu_gpio_tms	1	P16	JTAG TMS
cpu_gpio_tck	1	P2	JTAG TCK
cpu_gpio_tdi	1	P17	JTAG TDI
cpu_gpio_tdo	1	P18	JTAG TDO
cpu_gpio_trst	1	P19	JTAG TRST
cpu_gpio_jtag_sel	1	P6	JTAG select
cpu_gpio_tx1	1	n/a	UART 1, TX
cpu_gpio_rx1	1	n/a	UART 1, RX
cpu_gpio_tx2	1	n/a	UART 2, TX
cpu_gpio_rx2	1	n/a	UART 2, RX
cpu_gpio_led1	1	P7	LED 4 (red)
cpu_gpio_led2	1	P13	LED 5 (green)
cpu_gpio_led3	1	P8	LED 6 (yellow)

Signal Name	***Width [# of pins]***	***Port***	***Description***
cpu_gpio_key<3:0>	4	P1,P23, P4,P5	Status of keyswitch buttons
cpu_gpio_flash_busy	1	P20	Flash status indicator (STS)
cpu_gpio_sda	1	P0	I2C sda
cpu_gpio_scl	1	P3	I2C scl

3.1.2.2. Memory Mapping

The following table gives an overview of the SDB development board memory mapping after a hardware reset.

Address	*Peripheral*	*Peripheral Select*	*Size*
0x00000000	On-Chip RAM	N/A	8 kB
0x00001FFF			
0x00002000	Reserved		
0x00FFFFFF			
0x01000000	Flash	cpu_cs0_n	16 MB
0x01FFFFFF			
0x02000000	External SRAM	cpu_cs1_n	16 MB
0x02FFFFFF			
0x03000000	ProASIC^{PLUS}	cpu_cs2_n	8 MB
0x037FFFFF			
0x03800000	GlueChip	cpu_cs2_n	8 MB
0x03FFFFFF			
0x04000000	User device	cpu_cs3_n	16 MB
0x04FFFFFF			
0x05000000	Reserved		
0xFFDFFFFF			
0xFFE00000	On-chip peripherals	N/A	2 MB
0xFFFFFFFF			

The memory mapping above provides the default values used by the SDB platform. Depending on your application requirements, the address space configuration can be modified. For additional details please check either the software section of this manual or the specification from ATMEL.

3.1.3. GlueChip

In order to provide additional functionality that doesn't put restrictions on the CPU or the ProASIC[PLUS] FPGA the SDB development board contains a specific FPGA that implements these independent functions.

3.1.3.1. Block Diagram

The figure below highlights the main building blocks of the GlueChip FGPA:

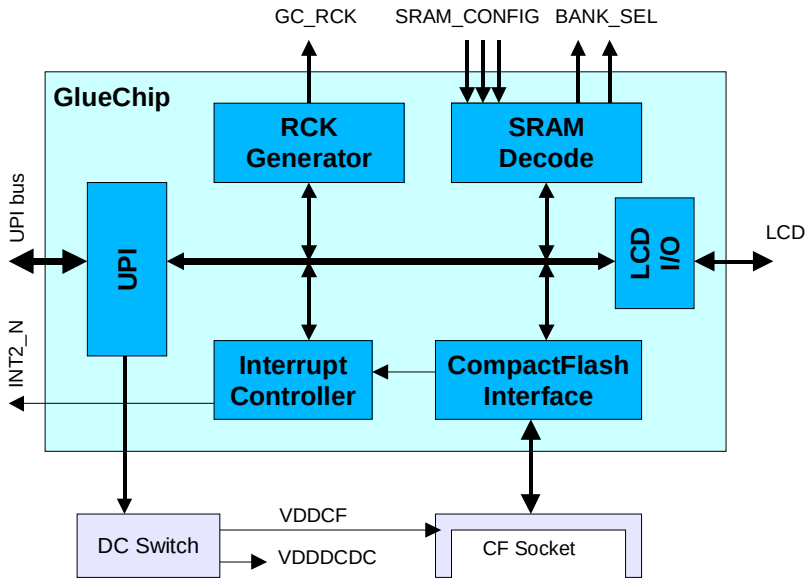


Figure 11 - GlueChip Block Diagram

All internal blocks of the GlueChip are connected to the Microcontroller interface via internal bus.

- RCK Generator:
  This block creates the running clock signal that is used for the in-system programming of the ProASIC[PLUS] FPGA on the SDB development board.
  This signal is configured through the CPU software.
- SRAM Decode:
  Here the SRAM chip-select signals for different memory banks are generated.
- LCD I/O:
  This block maps the LCD controller into the memory space of the CPU via the CPU bus. Additional control logic is included to simplify software interaction.
- CompactFlash Interface:
  This block mainly maps the external CompactFlash device into the memory space of the CPU. It also includes logic that allows the detection of the presence for a new CF device.

- Interrupt Controller:
  This function provides the masking of all GlueChip interrupts on to one interrupt line.
- UPI (Microprocessor Interface):
  Provides the main interface to the CPU where all GlueChip signals are centralized. In addition it provides one register to control the power source for the DC-DC converters.

Note: This FPGA can not be re-programmed and remains fixed within the system.

3.1.3.2. Register Mapping

The internal register mapping of the GlueChip is described in the following table. Please note that the CPU must be configured to access the GlueCip with one-wait-cycle (cpu_cs2_n).

GlueChip base address: 0x03800000

Chip select: cpu_cs2_n = <Low> and cpu_addr(23) = <High>

Interrupt Controller

Offset	*Register*	*Description*	*Access*	*Reset State*
0x0000	Interrupt Enable Register	<High> enables a particular interrupt source. (Connected to the CPU interrupt IRQ2)		
		[0]: CCD: Card Change Detection An insertion or removal of a CompactFlash card is indicated.	R	0x00
		[1]: LCD Ready Indicates an LCD status change from <Busy> to <Ready>	R	0x00
0x0002	Interrupt Status Register	[1:0]: '0': no interrupt pending '1': interrupt pending	R	0x00
		[1:0]: '0': no action '1': acknowledge pending interrupt	W	

Note: The CPU interrupt IRQ2 signal is connected to the GlueChip. The software acknowledges the pending interrupt by writing '1' into the corresponding status register.

LCD I/O

Offset	*Register*	*Description*	*Access*	*Reset State*
0x0010	LCD Command Register	[7:0]: Data/Command Byte to LCD [8]: A0 is the Register Selection port (RS) of the LCD. '0' ⇒ Commands '1' ⇒ Characters	r/W W	-
0x0012	LCD Status Register	[6:0]: LCD cursor address [7]: '0' ⇒ LCD is ready for new command '1' ⇒ LCD is busy	R	- 0x00
0x0014	Auto Poll LCD Status Enable	[0]: '0' ⇒ GC does not check LCD status '1' ⇒ GC checks LCD status after each write command	r/W	0x00

Note: If Auto Poll Status Enable = '0':
The GlueChip does not check the LCD status.

If Auto Poll Status Enable ='1':
Bit [6:0] in the LCD status register represent the current cursor address and Bit [7] shows the current LCD status. Additionally after each data transfer to the LCD an interrupt is requested from the CPU when the status changes from <Busy> to <Ready>. Please note that Bit [1] in Interrupt Enable Register must be set.

CompactFlash Status and Configuration register

Offset	*Register*	*Description*	*Access*	*Reset State*
0x0020	CF Status Register	[0]: CF_WP [1]: CF_CD1 [2]: CF_CD2 [3]: CF_VS1 [4]: CF_VS2 [5]: CF_RDYBSY_N [6]: CF_BVD1 [7]: CF_BVD2 [8]: CF_WAIT_N	R R R R R R R R R	N/A
0x0022	CF Hardware Reset Register	[0]: CF_RESET	r/W	0x00

Offset	Register	Description	Access	Reset State
0x0024	Wait Cycle Configuration Register	[4:0]: CF_COM_WAIT_CYCLE Min. number of wait cycles for memory access of CF common area. After 4 clock cycles the CF interface state machine waits for the cf_wait_n = '1' or cf_bsyrdy_n = '1'.	r/W	0x04
		[9:5]: CF_ATT_WAIT_CYCLE Min. number of wait cycles for memory access of CF attributes	r/W	0x09
		[10]: CF_WAIT_SIGNAL_SEL Select the source of wait signals from the CF '0' ⇒ cf_wait_n '1' ⇒ cf_rdybsy_n	r/W	0x00
0x0026	CF Configuration Register	[0]: CF_VDD_SELECT can be changed only if CF_VDD_EBL is '0' '0' ⇒ VDDCF = 3.3V '1' ⇒ VDDCF = 5V	r/W	0x00
		[1]: CF_VDD_EBL '0' ⇒ CF power disabled '1' ⇒ CF power enabled	r/W	0x00
		[2]: CF_INTERFACE_EBL Enable CF Interface '0' ⇒ CF is disabled and ports are in tri-state mode '1' ⇒ CF is enabled	r/W	0x00

Note: The GlueChip-CompactFlash interface is designed for memory mode operation. In this mode the CF reset is active <High> and the software can read the CF status signals by accessing the registers described above.

GlueChip internal registers

Offset	*Register*	*Description*	*Access*	*Reset State*
0x0040	GC_RCK Pre-Scaler.	A prescaler is used to have a system clock independent reference clock of 1MHz: $f_{rck} = f_{clksys}/[2*(pre_{scale}+1)]$ [5:0]: Prescale register	r/W	0x00
0x0042	RCK generator	[0]: '1' ⇒ Enable register [1]: '0' ⇒ Disable register	r/W	0x00
0x0044	DCDC Converter command register	[0]: '1' ⇒ Disable DC-DC power supply '0' ⇒ Enable DC-DC power supply	r/W	0x00
0x0046	Configuration of assembled SRAMs	[1:0]: "11" ⇒ 128kb x 8 "10" ⇒ 256kb x 8 "01" ⇒ 512kb x 8	R	-
0x0048	GlueChip Version Register.	[7:0]: Revision of GlueChip [15:8]: Version of GlueChip	R	-
0x0050 0x0FFF		Reserved		
0x1000 0x17FF	CF Common Memory	This is the common memory section of the CompactFlash.	R/W	-
0x1800 0x18FF	CF Attribute Memory	This is the attribute memory section of the CompactFlash.	R/W	-

Note: Please see the CompactFlash reference guide for additional information on the programming and usage of the common and attribute memory areas.

3.1.3.3. Pin Description

The following table shows the pin description of the GlueChip FPGA as it applies to the SDB development board. The pins are grouped by their appropriate functionality.

CPU bus

Name	*Pin count*	*GlueChip Pin*	*Description*
CPU_D<15:0>	16	24, 23, 21, 18, 17, 16, 15, 14, 13, 12, 8, 7, 6, 5, 4, 3	CPU data bus
CPU_A<12:1>	12	130, 132, 133, 134, 135, 136, 137, 138, 139, 141, 142, 143	CPU address bus
CPU_A<20:18>	3	122, 123, 124	CPU address bus
CPU_A<23>	1	121	CPU address bus
CPU_CS1_N	1	113	Chip select SRAM
CPU_CS2_N	1	114	Chip select GlueChip and ProASIC
CPU_WR_N	1	118	Write enable
CPU_OE_N	1	117	Output enable
CPU_UB_N	1	119	Upper byte enable
CPU_LB_N	1	120	Lower byte enable
CPU_WAIT_N	1	116	CPU wait
CPU_IRQ2	1	111	Interrupt request

LCD Interface

Name	*Pin count*	*GlueChip Pin*	*Description*
DISP_D<7:0>	8	92, 93, 94, 95, 96, 97,100, 103	LCD data
DISP_A0	1	106	LCD address
DISP_RWN	1	105	Read/ write not
DISP_CS	1	104	Chip select

SRAM

Name	*Pin count*	*GlueChip Pin*	*Description*
BANK_SEL<1:0>	2	26, 25	SRAM bank select
SRAM_CONFIG<1:0>	2	107, 108	SRAM configuration

RCK

Name	*Pin count*	*GlueChip Pin*	*Description*
GC_RCK	1	110	RCK clock for ProASIC

Power Switch

Name	*Pin count*	*GlueChip Pin*	*Description*
EBL_DCDC_N	1	27	DC-DC controller enable
CF_EBL_33_N	1	32	CompactFlash 3.3v enable
CF_EBL_50_N	1	31	CompactFlash 5v enable

Global Signals

Name	*Pin count*	*GlueChip Pin*	*Description*
RESET_N	1	126	Global reset
CLKA_TO_GC	1	125	Global Clock
CLK_TO_GC	1	60	Global Clock (Dedicated input :Hard wire Clock HCLK)

TEST PORTS

Name	*Pin count*	*GlueChip Pin*	*Description*
PRA / CF_BVD2	1	131	Probe A / Can be connected to CF_BVD2
PRB	1	54	Probe B
GC_TMS	1	9	JTAG Test mode select (dedicated input)
GC_TCK	1	144	JTAG Test clock
GC_TRST	1	22	JTAG Reset

Name	*Pin count*	*GlueChip Pin*	*Description*
GC_TDI	1	2	JTAG Test data input
GC_TDO	1	71	JTAG Test data output

CompactFlash

Name	*Pin count*	*GlueChip Pin*	*Description*
CF_D<15:0>	16	77, 82, 84, 86, 88, 34, 37, 39, 78, 83, 85, 87, 91, 38, 40, 42	Data
CF_A<10:0>	11	74, 67, 65, 63, 61, 55, 52, 50, 48, 46, 43	Address
CF_CE1_N	1	76	Chip enable
CF_CE2_N	1	75	Chip enable
CF_VS1_N	1	72	Voltage select
CF_VS2_N	1	53	Voltage select
CF_RESET	1	51	Reset
CPU_GPIO_SDA / SDA_IORDN	1	69	I2C SDA / can be connected to CF IO read
CPU_GPIO_SCL / SCL_IOWRN	1	66	I2C SCL / can be connected to CF IO write
CF_WE_N	1	64	Write enable
CF_OE_N	1	70	Output enable
CF_RDYBSY_N	1	62	Ready/busy not
CF_WAIT_N	1	49	Wait indicator
CF_INPACK_N	1	47	INPACK
CF_REG_N	1	45	Register write enable
CV_BVD1	1	41	BVD1
CF_CSEL_N	1	59	Card select
CF_WP	1	35	Write protect
CF_CD1_N	1	112	Card detection 1
CF_CD2_N	1	33	Card detection 2

3.1.4. Flash

The SDB development board is designed for Intel's StrataFlash device family. The SDB board logic can support memory sizes of 32Mb, 64Mb and 128Mb. The SDB development board is by default delivered with a 32Mb Flash.

The Flash is 16-bit wide and supports byte access. It is connected to cpu_cs0_n which maps the 16MB memory block between `0x01000000` and `0x01FFFFFF` as explained in the Memory Mapping section of this document.

The following special enhancements have been made on the SDB development board.

- The Flash memory is placed in a socket. This allows the user to have different images of the system without the need for reprogramming.
- The Flash programming status indicator (STS) signal is available on the cpu_gpio and the ProASIC^{PLUS} I/O port. Thereby allowing the system to control the reprogramming cycle of the Flash memory.

Please verify with the Intel memory datasheet mentioned in the reference section for detailed specifications about this chip.

3.1.5. SRAM

The SDB development board supports a total of 16-Mbit SRAM. The SRAM block are divided into two banks each being 16-bit wide.

The chip select signal is connected to cpu_cs1_n which maps the 16MB memory block between `0x02000000` and `0x02FFFFFF` as explained in the Memory Mapping section earlier in this document.

The chip is manufactured in four different memory sizes while using the same package information (SOJ-36, 400 mil). This allows the SDB development board to be assembled with either one of the following memory types 128kx8, 256kx8 or 512kx8. The SDB development board is by default delivered with 4 x 512kx8 SRAMs.

In case the user wishes to upgrade the memory to a bigger size, the SRAM configuration links LK7 have to be changed as well. For further information please verify the IDT datasheet mentioned in the reference section or contact Inicore for additional detail on setting the LK7 configuration links.

3.2. Power Regulators

The SDB development board comes equipped with one external 5V power supply. However, the ProASIC^{PLUS} FPGA requires two additional voltage levels to be programmed within the system.

To provide this functionality, the SDB contains two on-board DC-DC converters. They are specified as follow:

- VDDPP: +16.2V ±300mV @ 30mA
- VDDPN: -13.6V ±200mV @ 20mA

The following configuration is used to provide the necessary signals:

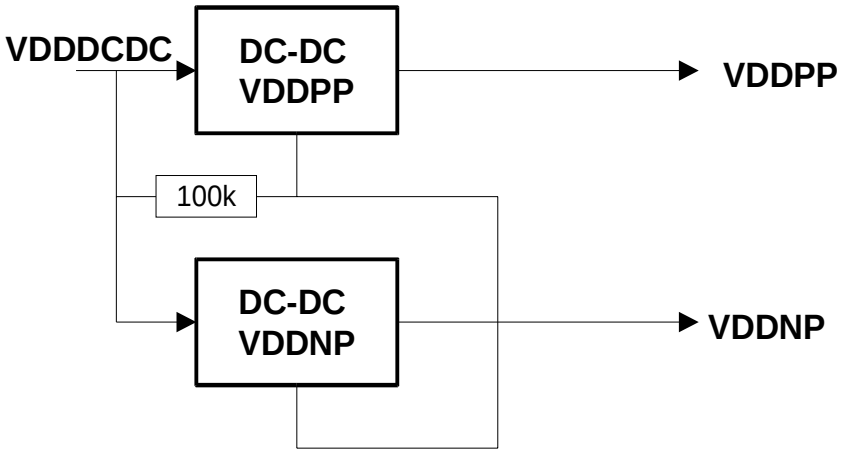


Figure 12 - DC-DC converter configuration

These DC-DC converters can be controlled using the register settings of the GlueChip FPGA. For additional information please refer to the GlueChip section above.

3.3. Reset Logic

The next figure gives an indication of the reset logic and its connecting components.

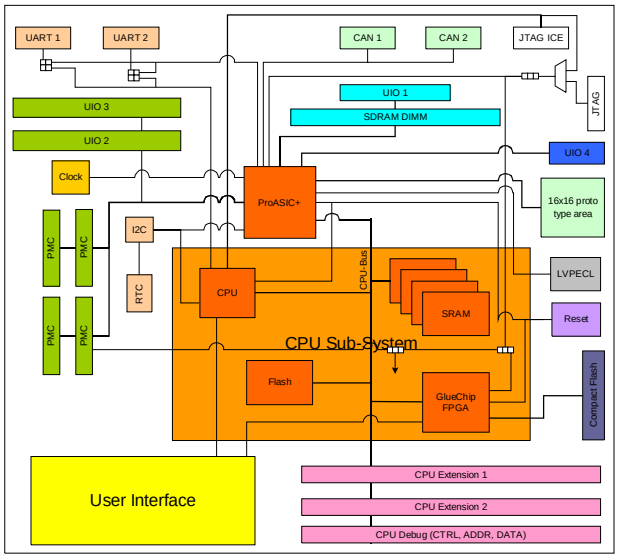


Figure 13 - Reset Logic block diagram

A power-on-reset circuit is used to initialize the system upon power-up. As it can be seen in the figure above, the reset logic controls the ProASIC^{PLUS}, the CPU and the GlueChip.

The following figure gives additional details on the implementation of the reset logic.

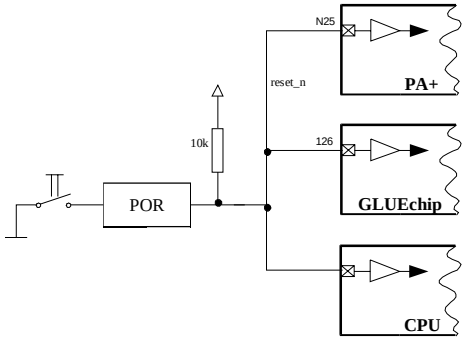


Figure 14 - Reset Logic

If desired, the ProASIC^{PLUS} FPGA could be implemented as a reset controller using a time-out watchdog. In that type of application the watchdog (ProASIC^{PLUS}) can pull the reset_n line to low in order to reset the CPU and GlueChip.

For the timing of the reset signal, please refer to the CPU and FPGA specifications.

Note: The reset line should be implemented using an open collector output so that the FPGA doesn't drive the reset_n line to high state.

3.4. Prototyping Area

The following block diagram indicates the prototyping area that is integrated on the SDB development board.

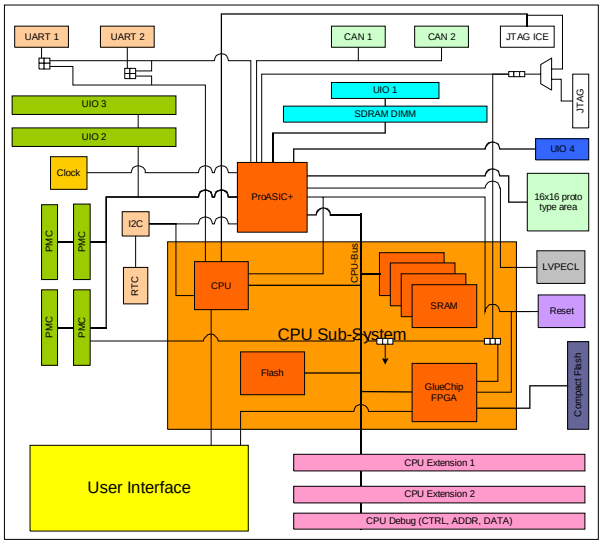


Figure 15 - Prototyping area block diagram

As indicated above, the prototyping area is connected directly to the ProASIC^{PLUS} FPGA.

A 16x16 through-hole solder point area is provided for prototyping as indicated below.

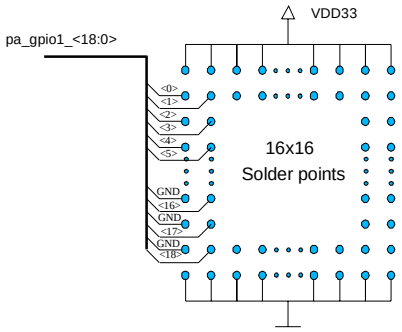


Figure 16 - Prototyping Area

The top row is connected to VDD33 and the bottom row of the prototyping area is connected to GND.

In addition, there are two different types of signal path preconditioned on the SDB development board.

1. The upper half two rows on the left of the area are the signals named with pa_gpio1_<9:0>. They are directly connected between the ProASIC^{PLUS} FPGA and the prototyping area.
2. The lower half two rows on the left of the area with the signal names pa_gpio1_<18:10> are provisioned for high-speed I/O. This means that they have no vias on the signal path between the ProASIC^{PLUS} and the prototyping area. In addition there are GND signals positioned in between two high speed signals that terminate on the first left hand row.

All solder points are through-whole and allow to assemble parts on either side of the SDB development board.

3.5. CANbus

The figure below shows the CANbus connectivity of the SDB development board.

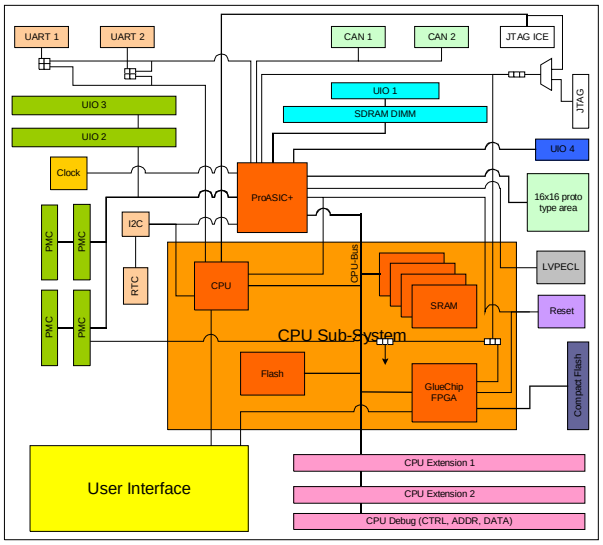


Figure 17 - CANbus block diagram

As indicated above, two standard CANbus ports are available on two external connectors. These ports are connected to the ProASIC[PLUS] FPGA through an on-board 5V transceiver as indicated in the following figure.

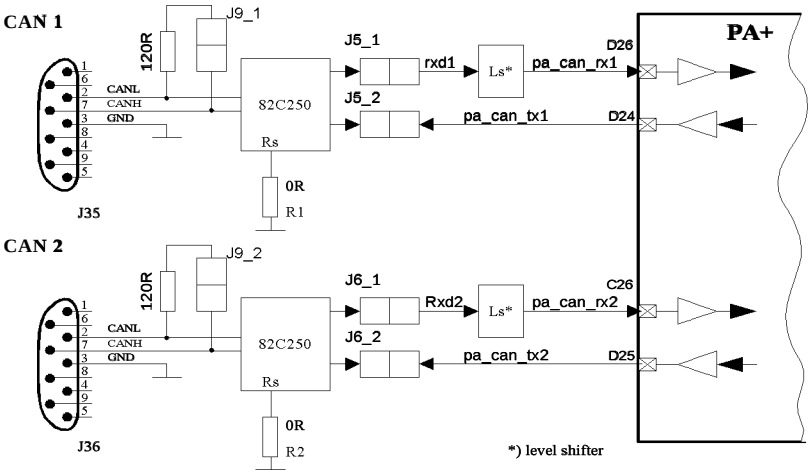


Figure 18 - CANbus network

Three jumper blocks are available to configure the CANbus interface (J5, J6 and J9). They are used as explained in the following tables.

3.5.0.1. J5: CAN Port 1 Configuration

J5_1	*Function*	*Setting*
None	The ProASIC^{PLUS} FPGA pin is a GPIO pin	⊗
A-B	The CAN1 RX signal is connected to the ProASIC^{PLUS} FPGA	

J5_2	*Function*	*Setting*
None	The ProASIC^{PLUS} FPGA pin is a GPIO pin	
A-B	The CAN1 TX signal is connected to the ProASIC^{PLUS} FPGA	⊗

3.5.0.2. J6: CAN Port 2 Configuration

J6_1	*Function*	*Setting*
None	The ProASIC^{PLUS} FPGA pin is a GPIO pin	⊗
A-B	The CAN2 RX signal is connected to the ProASIC^{PLUS} FPGA	

J6_2	*Function*	*Setting*
None	The ProASIC^{PLUS} FPGA pin is a GPIO pin	⊗
A-B	The CAN2 TX signal is connected to the ProASIC^{PLUS} FPGA	

3.5.0.3. J9: CAN Bus Termination

The CAN bus network termination can be set using the J9_1/J9_2 jumpers as follow.

J9_1	*Function*	*Setting*
None	The CAN bus 1 is not terminated	⊗
A-B	The CAN bus 1 is terminated with 120Ω	

J9_2	*Function*	*Setting*
None	The CAN bus 2 is not terminated	⊗
A-B	The CAN2 bus 2 is terminated with 120Ω	

Note: The input cells of the ProASIC^{PLUS} FPGA are not 5V compliant, therefore two level shifters were used in connection with the CANbus RX signals. If your CAN system runs with 500kbaud or less, you might consider enabling the CAN transceiver's slew rate control by exchanging R1 and R2 with a different value based on your slew rate requirements and the specification of the 82C250.

3.6. LVPECL Differential Data

The block diagram of the LVPECL differential data interface can be seen below

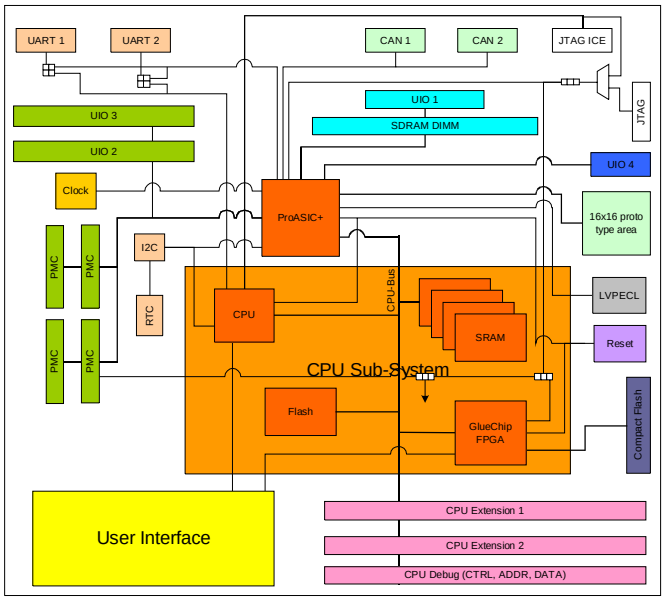


Figure 19 - LVPECL block diagram

One pair of standard SMA connectors are available to feed a data stream or an external clock to the LVPECL inputs on the eastern side of the ProASIC^{PLUS} FPGA. The signal lines are 50Ω impedance matched with their respective termination as indicated in the following figure.

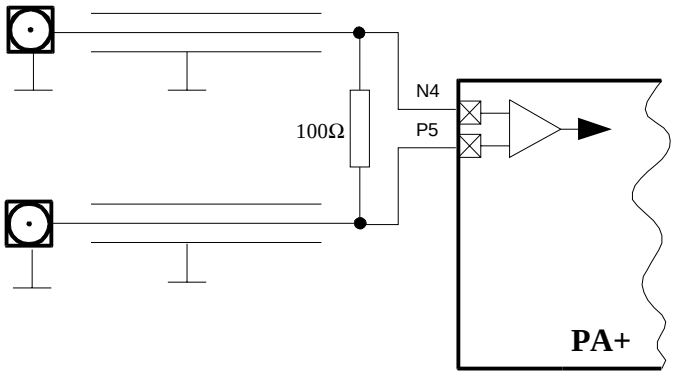


Figure 20 - LVPECL impedance match

3.7. User Interface

The next figure indicates the components of the user interface.

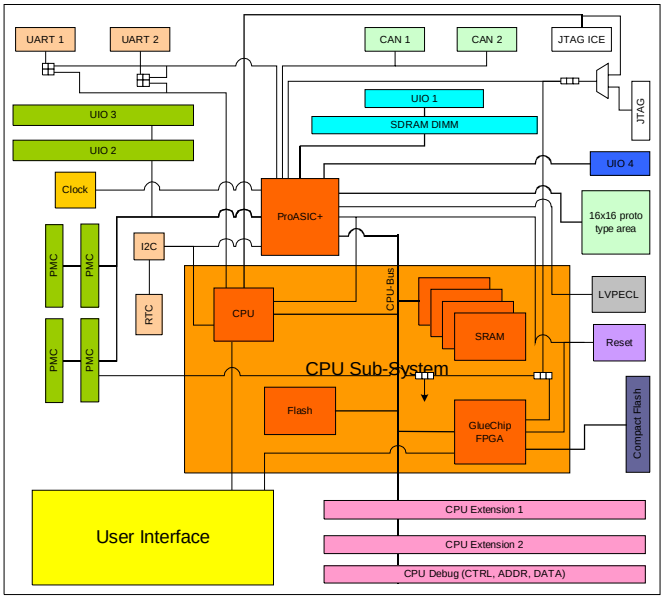


Figure 21 - User Interface block diagram

The user interface is comprised of two elements:

1. One 2 lines x 16 characters intelligent LCD. It has an on-board LCD controller and driver circuit. The device can display up to 160 characters (numericals, letters, symbols and Kana letters), as well as up to eight custom designed characters. Please refer to the appropriate specification for further information.
2. Five mechanical keyswitches that allow entering and selecting information according to the user programming. The five switches have the following functions using the pre-installed demo application:

KeySwitch	***Function***
UP	By pressing this switch once, the previous menu is displayed on the LCD
DOWN	By pressing this switch once, the next menu is displayed on the LCD
CANCEL	By pressing this switch once, the current menu choice is terminated
SELECT	By pressing this switch once, the current menu is selected
RESET	By pressing this switch once, the SDB development board is forced into a hardware reset

As the above figure indicates, the LCD is controlled by the GlueChip. In addition keyswitch events such as pressing a button can be processed either by the CPU or the ProASIC[PLUS] FPGA.

3.8. Real Time Clock

The real time clock block is highlighted in the following figure.

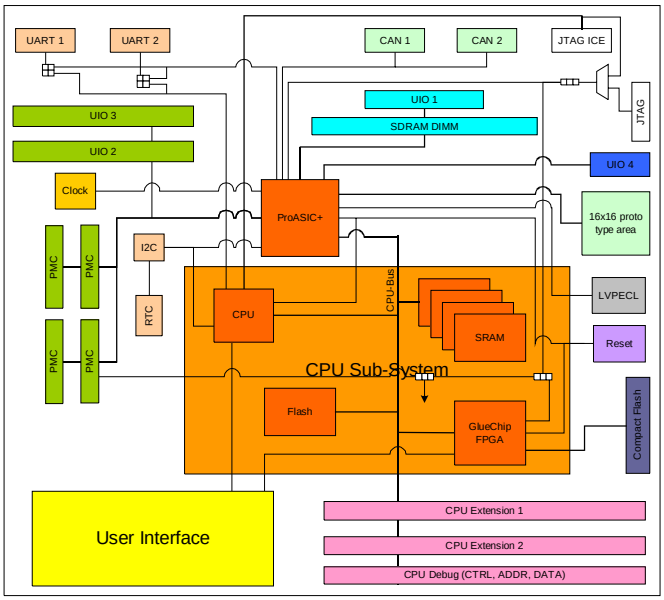


Figure 22 - RTC block diagram

The RTC (real time clock) is used to provide a time reference for the SDB development board. The RTC IC is connected to the CPU and the ProASIC^{PLUS} through the I2C bus.

The RTC is operational even when the board is disconnected from an external power source since it uses its own battery that is located underneath the LCD. The power source is a coin cell Lithium battery with 3V,180mAh and Ref #: CR2032.

In order to replace the battery, the LCD has to be disconnected by removing the two screws that mount the unit onto the SDB development board.

4. EXTERNAL HARDWARE

The SDB development board supports three hardware extensions via standardized connectors as indicated in the following figure.

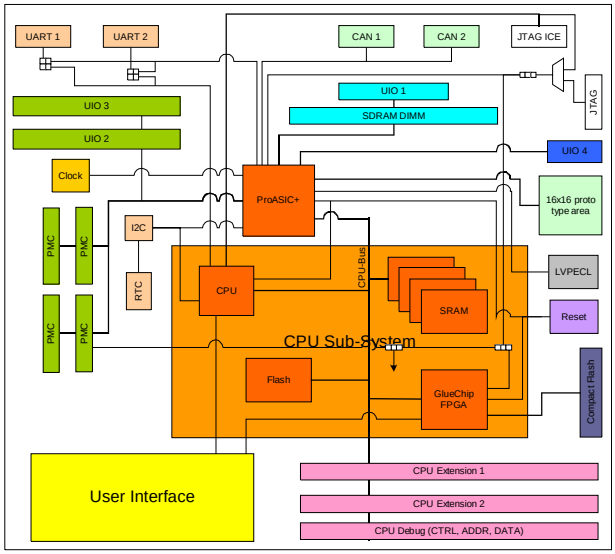


Figure 23 - External Hardware block diagram

The ProASIC[PLUS] FPGA can be used to control the PMC and the SDRAM interface with appropriate programming, whereas the GlueChip is preprogrammed to interface the CompactFlash Card in memory mode.

4.1. Overview

These three standard interfaces that are supported with the SDB development board are as follow:

1. One standard 3.3V PCI Mezzanine card interface where PTMC specifications are met for telecommunication busses up to 50MHz. This includes such standards as UTOPIA level 1 & 2, POS-PHY, RMII, IEEE 802.3 and others.
   The user can either connect the external hardware through the PMC connector bus or via UIO2 and UIO3 directly to the ProASIC[PLUS] FPGA.
2. Standardized 168-pin SDRAM DIMM connector where up to 256MB additional memory can be provided for the ProASIC[PLUS] FPGA.
3. A CompactFlash plug in interface that allows to connect one memory card with up to 16MB. This memory card is used to hold STAPL files for ProASIC[PLUS] FPGA in-circuit programming downloads.

4.2. PCI Mezzanine card (PMC)

Any standardized 3.3V PCI Mezzanine card can directly be plugged into the SDB development board. The SDB platform supports either 32-bit or 64-bit PMC hardware. If the 32-bit PCI is used, bits <63:32> are usable through the UIO2 extension connector for other purposes. When using the 64-bit PCI, the UIO2 connector is not available for other functions.

The host interface is also PTMC compatible so that PCI Telecom Mezzanine cards can be used.

Please refer to the PCI Mezzanine Card User Guide for additional information about how to connect and operate PCM cards.

4.2.1. SDRAM-DIMM

A standard 168-pin DIMM connector is available to plug in an unbuffered SDRAM DIMM if desired. The SDRAM signals are directly connected to the ProASIC FPGA.

The I2C interface for the configuration PROM is connected to the local I2C bus of the SDB development board.

The following information is specific for the SDB development board usage:

- The serial PD base address is configured at: 0b10100000
- The serial PD write protect is not connected
- Registered DIMM can be configured using the provided SMLink
- The higher 4-bytes of the data bus are available on the UIO1 connector for systems where not the entire 64-bit wide memory is required. However, the user should verify that the respective DQMB<7:4> bits are used accordingly within the system.

4.3. Compact Flash

The CompactFlash socket allows downloading different ProASIC^{PLUS} configurations. Using this technology provides a faster way to download ProASIC^{PLUS} configurations compared with using the conventional JTAG interface.

The CompactFlash controller resides inside the GlueChip FPGA on the SDB development board. All required signals are directly connected to this FPGA. For additional register information please refer to the GlueChip section of this document.

The following information are specific to the SDB development board:

- The socket is configured for 3.3V operation.
- The Life insertion function for CompactFlash Memory cards is supported
- Per default, the CompactFlash device is memory mapped into the CPU address space as indicated in the memory mapping section described above.

5. SOFTWARE

This section gives an overview of the memory mapping, the RedBoot monitor, how application programs are downloaded and the functionality of the demo application that is included with the delivery of the SDB development board.

The following figure gives an overview of the software structure that is used for the SDB development board.

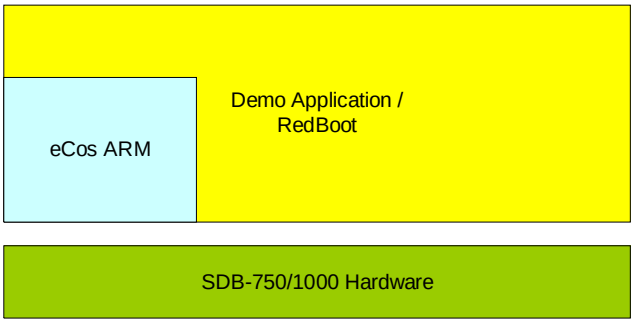


Figure 24 - Layer structure

Two different applications are delivered with the SDB development board.

1. RedBoot monitor from RedHat with specific configuration files
2. Demo Application from Inicore, Inc. to exhibit the different features

Each application uses the eCos library for ARM processors during compilation in order to create downloadable software versions for the SDB.

For additional information on eCos or how to install the library for Windows environments, please refer to the RedHat website at http://sources.redhat.com/ecos/.

5.1. Program Memory Mapping

The following memory mapping indicates where the RedBoot monitor and the Demo Application have been pre-installed on the SDB. Furthermore, it indicates where user applications are located either for RAM or ROM memory execution.

The following table is only a section of the entire memory space. For additional details refer to the memory mapping section under the ATMEL CPU above.

Address	*Peripheral*	*Application*
0x01000000 0x0101FFFF	Flash	RedBoot Monitor
0x01020000 0x01FFFFFF	Flash	User Application
0x02000000 0x02FFFFFF	RAM	User Application

Note: After reset the program execution begins at the address 0x0.
If the Cancel key is pressed during reset, the RedBoot Monitor will be executed at 0x01000000. Otherwise the program at address 0x1020000 is executed.

Please do not modify the section between 0x1000000 and 0x1020000. This would corrupt the RedBoot Monitor what could make it impossible to load software upgrades without externally reprogramming the Flash chip with the original RedBoot Monitor software.

5.2. RedBoot Monitor

The RedBoot Monitor is located in the Flash memory and is pre-loaded with the SDB. It can be activated when pressing the Cancel key during a board Reset cycle.

The program provides the following features:

- Serial interface through UART1 connection with the PC
- Allows downloading new software on to the SDB and re-programming the software Flash
- Provides system functions such as read and write operations into specific memory addresses
- Allows to process ARM CPU in-circuit information for debugging purposes
- Provides GDB (GNU debugger) port using UART1

The RedBoot Monitor can be operated via PC and terminal software through the UART1 interface.

The following commands are available in the RedBoot monitor:

- Manage aliases kept in FLASH memory
- Manage machine caches
- Display/switch console channel
- Compute a 32bit checksum [POSIX algorithm] for a range of memory
- Display (hex dump) a range of memory
- Manage FLASH images
- Manage configuration kept in FLASH memory
- Execute code at a location
- Help about help?
- Load a file
- Reset the system
- Display RedBoot version information
- Display (hex dump) a range of memory

In addition there is a help library included in the RedBoot monitor that provides detailed information for each function. It can be retrieved when typing help followed by the command.

Please refer to the RedBoot on-line help for additional details on the operation of these commands.

5.3. Application Installation Process

For the installation process of the application there is a straight forward process as indicated in the following figure.

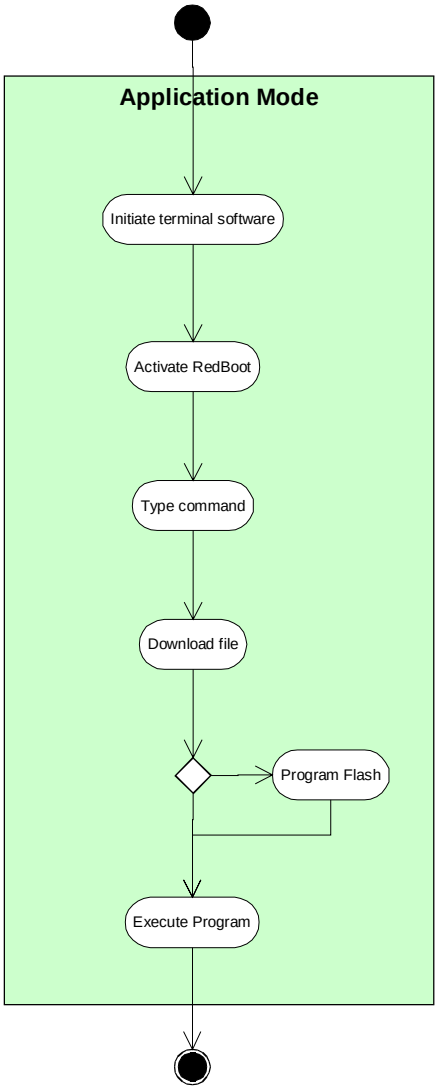


Figure 25 - Installation process

The application group has access to the *.bin files that contain the compiled program.

The following section describes the above process for the application group.

5.3.1. Application Mode Installation

This section gives step-by-step instructions on how to download an application when the finalized *.bin file is available and the program location has been determined.

#	*User action*	*System reaction*
1.	Link the SDB development board to the host PC by using the serial cable on the UART1 connector.	
2.	Start up terminal software (Minicom, HyperTerminal, etc.)	Screen for port configuration appears
3.	Configure the following protocol values: • Baudrate: 38,400 • Data bits: 8 • Parity: None • Stop bits: 1 • Flow Control: None	The terminal software waits for activation
4.	Press the Reset button on the SDB development board while pushing the Cancel button to activate the RedBoot Monitor	The RedBoot Monitor prompt "RedBoot>" will appear in the terminal window
5.	Type the following command in the terminal window: >load -r -m ymodem -b <ram address> example: >load -r -m ymodem -b 0x2020000	After pressing return, the RedBoot Monitor waits for the file to download
6.	Send the file using the menu command and the ymodem protocol for the transfer • Transfer -> send File (make a note of the file length used for step 8 below)	The file is downloaded into the SDB development board
7.	If the program is compiled for RAM execution please continue with step 9, otherwise the file has to be written into the Flash memory before execution.	
8.	Use the "fis" command to program the flash as follow: >fis write -f <Flash addr> -b <ram addr> -l <file length> example: >fis write -f 0x1020000 -b 0x2020000 -l 0x20000	The file is downloaded from the RAM into the Flash memory
9.	Execute the program by typing the following command: >go <address>	The program is executed

Note: For ROM execution the file has to be located on address 0x1020000 or higher.
If the address area is locked, the Flash memory can't be programed. Type "help fis" for more information.

5.4. Demo Application Overview

The following figure gives an overview of the pre-loaded demo application and its functionality.

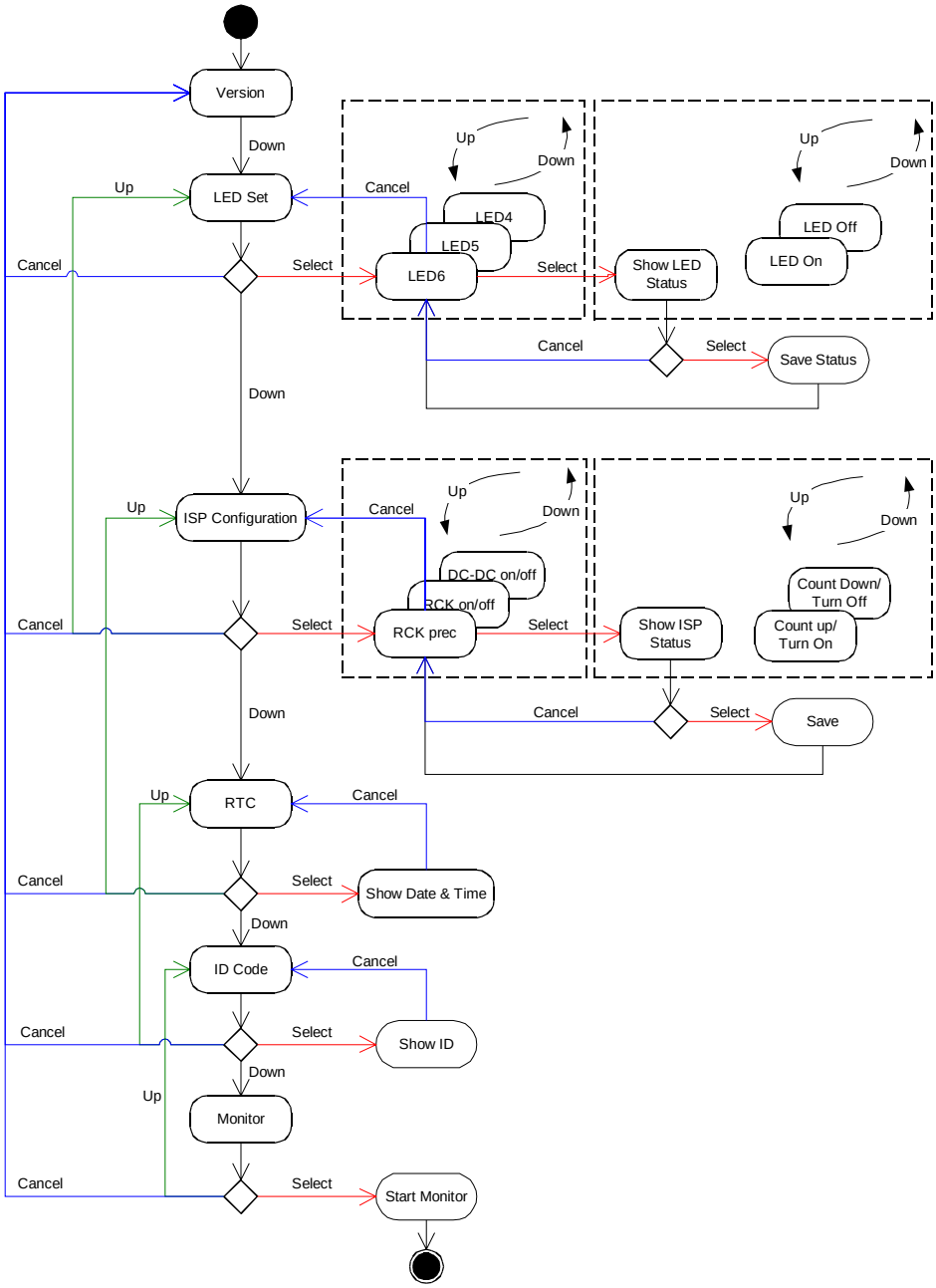


Figure 26 - Demo Application screen flow

Please refer to the separate application note of the demo design for additional information on the programming of the demo application. This document can be downloaded from the Inicore, Inc. website at http://www.inicore.com.

5.5. Additional Tools

Although the SDB development boards comes with many features and tools, the following two additional investments would help the developer to be more efficient to speed up the development process.

Even though the ARM processor supports the use of the UART1 for debugging of CPU software, it is highly recommended to purchase additional hardware such as the JEENI JTAG embedded ICE Ethernet Interface for ARM (http://www.epitools.com/products/arm/hardware/jeeni.shtml) in order to speed up the process.

With the current release of the SDB development board software V1.1, autonomous in-system programming of the ProASIC^{PLUS} FPGA is not yet supported. In order to program the ProASIC^{PLUS} FPGA an external FlashPro programmer from Actel Corporation can be used. For additional information please refer to the FlashPro programmer product specification at Actel's website. (http://www.actel.com/products/tools/flashpro/index.html)

6. CONFIGURATION

This section contains the technical details of the SDB development board. The jumper settings are explained within the document as well at their appropriate locations, whereas the assembly options are for your information only.

For additional questions, please refer to the Inicore, Inc. website at www.inicore.com

6.1. Jumper Settings

This section summarizes the jumper settings for the SDB development board.

6.1.1. J1: JTAG Source Select

J1	*Function*	*Setting*
None	The external JTAG header is the JTAG source for the SDB	⊗
A-B	The CPU is the JTAG source	

6.1.2. J2: JTAG PCI Mezzanine card

J2	*Function*	*Setting*
A-B	The PMC TAP controller is NOT part of the JTAG chain. (TDI/TDO shortcut for the JTAG is activated)	⊗
B-C	The PMC TAP controller is part of the JTAG chain. (TDI/TDO shortcut for the JTAG is deactivated)	

6.1.3. J3: UART 1 Source Select

J3_1	*Function*	*Setting*
A-B	The CPU UART port is used for receiving signals	⊗
B-C	The ProASIC^{PLUS} UART port is used for receiving signals	

J3_2	*Function*	*Setting*
A-B	The CPU UART port is used for transmitting signals	⊗
B-C	The ProASIC^{PLUS} UART port is used for transmitting signals	

6.1.4. J4: UART 2 Source Select

J4_1	*Function*	*Setting*
A-B	The CPU UART port is used for receiving signals	⊗
B-C	The ProASIC^{PLUS} pre-configured UART port is used for receiving signals	

J4_2	*Function*	*Setting*
A-B	The CPU UART port is used for transmitting signals	⊗
B-C	The ProASIC^{PLUS} pre-configured UART port is used for transmitting signals	

6.1.5. J5: CAN Port 1 Configuration

J5_1	*Function*	*Setting*
None	The ProASIC^{PLUS} FPGA pin is a GPIO pin	⊗
A-B	The CAN1 RX signal is connected to the ProASIC^{PLUS} FPGA	

J5_2	*Function*	*Setting*
None	The ProASIC^{PLUS} FPGA pin is a GPIO pin	⊗
A-B	The CAN1 TX signal is connected to the ProASIC^{PLUS} FPGA	

6.1.6. J6: CAN Port 2 Configuration

J6_1	*Function*	*Setting*
None	The ProASIC^{PLUS} FPGA pin is a GPIO pin	⊗
A-B	The CAN2 RX signal is connected to the ProASIC^{PLUS} FPGA	

J6_2	*Function*	*Setting*
None	The ProASIC^{PLUS} FPGA pin is a GPIO pin	⊗
A-B	The CAN2 TX signal is connected to the ProASIC^{PLUS} FPGA	

6.1.7. J7: CPU Disable

J7	*Function*	*Setting*
None	The CPU is operational	⊗
A-B	The CPU is disabled	

6.1.8. J8: Clock Jumpers

J8_1	*Function*	*Setting*
A-B	The Clock buffer 1 is driven by the clk1_from_pa signal.	
B-C	The Clock buffer 1 is driven by the osc1 signal	⊗

J8_2	*Function*	*Setting*
A-B	The Clock buffer 2 is driven by the clk2_from_pa signal	
B-C	The Clock buffer 2 is driven by the osc2 signal	⊗

J8_3	*Function*	*Setting*
A-B	The Clock buffer 3 is driven by the clk_to_cpu_from_pa signal	
B-C	The Clock buffer 3 is driven by the osc2 signal	⊗

6.1.9. J9: CAN Bus Termination

J9_1	*Function*	*Setting*
None	The CAN bus 1 is not terminated	⊗
A-B	The CAN bus 1 is terminated with 120Ω	

J9_2	*Function*	*Setting*
None	The CAN bus 2 is not terminated	⊗
A-B	The CAN2 bus 2 is terminated with 120Ω	

6.2. Options

The following tables describe different options for the SDB development board. These options are achieved by using 0Ω resistors to specify the functionality that is described within the tables.

Note: Modifying the board with a soldering iron voids the warranty of the SDB.

6.2.1. LK1: JTAG PMC Enable

The JTAG control signal are available on the mezzanine card but can be disconnected by changing the following settings if desired.

LK1_1	***Function***	***Setting***
A-B	The JTAG-TMS signal of the mezzanine card is tied <High>	⊗
B-C	The JTAG-TMS signal is connected to the mezzanine card's TMS signal.	

LK1_2	***Function***	***Setting***
A-B	The JTAG-TCK signal of the mezzanine card is tied <High>	
B-C	The JTAG-TCK signal is connected to the mezzanine card's TCK signal.	⊗

LK1_3	***Function***	***Setting***
A-B	The JTAG-TRST signal of the mezzanine card is tied <High>	
B-C	The JTAG-TRST signal is connected to the mezzanine card's TRST signal.	⊗

6.2.2. LK2: JTAG GlueChip FPGA Enable

The GlueChip's TAP controller can be included in the JTAG chain if desired.

LK2_1	***Function***	***Setting***
A-B	The JTAG-TMS signal of the GlueChip FPGA is tied <High>	⊗
B-C	The JTAG-TMS signal is connected to the GlueChip FPGA's TMS signal	

LK2_2	*Function*	*Setting*
A-B	The GlueChip's TAP controller is NOT included in the JTAG chain (JTAG-TDI/TDO shortcut is activated)	⊗
B-C	The GlueChip's TAP controller is included in the JTAG chain (JTAG-TDI/TDO shortcut is disconnected)	

6.2.3. LK3: Flash Byte Mode Select

To select the byte mode option of the Intel Flash memory. This option can be used when a 8-bit CPU core is integrated into the ProASIC^{PLUS} FPGA design.

LK3	*Function*	*Setting*
A-B	The 16-bit mode is enabled for the Flash memory	⊗
B-C	The 8-bit mode is enabled for the Flash memory	

6.2.4. LK4: DC-DC Converter

Instead of using the on-board DC-DC converters for the two programming voltages of the ProASIC^{PLUS} FPGA, an external power supply can be used.

LK4_1	*Function*	*Setting*
None	The external programming voltage VDDPP is used.	
A-B	The VDDPP signal is generated by the on-board DC-DC converters	⊗

LK4_2	*Function*	*Setting*
None	The external programming voltage VDDNP is used.	
A-B	The VDDNP signal is generated by the on-board DC-DC converters	⊗

Note: The power source for the DC-DC converters can be turned off using the GlueChip registers.

6.2.5. LK5: Registered SDRAM DIMM Configuration

This configuration option is used to define the type of SDRAM modules. The choices are either select buffered or registered mode. This SMLink option is left open for unbuffered DIMMs.

LK5	*Function*	*Setting*
None	Select unbuffered DIMM mode	⊗
A-B	Select buffered DIMM mode	
B-C	Select registered DIMM mode	

6.2.6. LK6: BVD2 Enable

The PRA port from the GlueChip can be connected to the CompactFlash BVD2 port.

LK5	*Function*	*Setting*
None	The PRA port is defined as a test port	⊗
A-B	The PRA port is defined as an input port and connected to the CF_BVD2 signal	

6.2.7. LK7: SRAM Configuration

The SDB development board can be assembled with different SRAM devices. Using the following table, the respective decoding logic can be selected.

LK7_1	*LK7_2*	*Function*	*Setting*
A-B	A-B	Configured for 128kx8 SRAM	
A-B	B-C	Configured for 256kx8 SRAM	
B-C	A-B	Configured for 512kx8 SRAM	⊗

Note: All banks have to be assembled with the same type of SRAM devices. However, it is possible to only assemble bank 0 and leave bank 1 unused.

6.2.8. LK8, LK9: ProASIC^{PLUS} Power

These two SMLinks are used to provide the ProASIC^{PLUS} FPGA with optional power levels and they can not be reconfigured.

6.2.9. LK10: I2C / CF_IO signals

With these SMLinks, it is possible to select which signals are connected to GlueChip.

LK10_1	*Function*	*Setting*
A-B	The CPU_GPIO_SCL signal is connected to pin 64 of the GlueChip	
B-C	The CF_IOWR_N signal is connected to pin 64 of the GlueChip	⊗

LK10_2	*Function*	*Setting*
A-B	The CPU_GPIO_SDA signal is connected to pin 65 of the GlueChip	
B-C	The CF_IORD_N signal is connected to pin 65 of the GlueChip	⊗

6.2.10. LK11: Boot-Mode Select

LK11	*Function*	*Setting*
A-B	The CPU boots from a 8-bit external flash memory	
B-C	The CPU boots from a 16-bit external flash memory	⊗

6.2.11. LK12: Shutdown - INTC Select

LK12	*Function*	*Setting*
A-B	The DC-DC converters on the mezzanine board are driven by the GlueChip	⊗
B-C	The PMC_INTC_N signal is linked to the PMC connector	

6.2.12. LK13: RCK - INTD Select

LK12	*Function*	*Setting*
None	The PMC_INTD_N signal is linked to the PMC connector	
A-B	The RCK signal that is used during the ProASIC^{PLUS} FPGA programming cycle is generated by the PMC.	⊗

6.3. Connector Definitions

The following tables provide a summary of the connector signal definitions as they pertain to the SDB development board.

6.3.1. Power

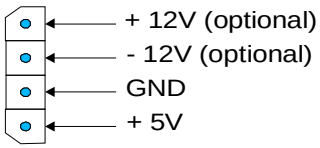


Figure 27 - Power connector

6.3.2. PMC connectors

The following four connectors describe the pinout for the PMC connectors.

6.3.2.1. J11: PMC-Jn1

Signal	*Pin*	*Pin*	*Signal*	*Signal*	*Pin*	*Pin*	*Signal*
PMC_TCK	1	2	VDDN12	PMC_FRAME_N	33	34	GND
GND	3	4	PMC_INTA_N	GND	35	36	PMC_IRDY_N
PMC_INTB_N	5	6	INTCN_SHDN	PMC_DEVSEL_N	37	38	VDD50
PMC_BM1_N	7	8	VDD50	GND	39	40	PMC_LOCK_N
PMC_INTD_N	9	10	N/C	CPU_GPIO_SCL	41	42	CPU_GPIO_SDA
GND	11	12	VDD33	PMC_PAR	43	44	GND
CLK_TO_PMC	13	14	GND	VDD33	45	46	PMC_AD15
GND	15	16	PMC_GNT_N	PMC_AD12	47	48	PMC_AD11
PMC_REQ_N	17	18	VDD50	PMC_AD9	49	50	VDD50
VDD33	19	20	PMC_AD31	GND	51	52	PMC_CBE0_N
PMC_AD28	21	22	PMC_AD27	PMC_AD6	53	54	PMC_AD5
PMC_AD25	23	24	GND	PMC_AD4	55	56	GND
GND	25	26	PMC_CBE3_N	VDD33	57	58	PMC_AD3
PMC_AD22	27	28	PMC_AD21	PMC_AD2	59	60	PMC_AD1
PMC_AD19	29	30	VDD50	PMC_AD0	61	62	VDD50
VDD33	31	32	PMC_AD17	GND	63	64	PMC_REQ64_N

6.3.2.2. J12: PMC-Jn2

Signal	Pin	Pin	Signal	Signal	Pin	Pin	Signal
VDD12	1	2	PMC_TRST	GND	33	34	N/C
PMC_TMS	3	4	PMC_TDO	PMC_TRDY_N	35	36	VDD33
PMC_TDI	5	6	GND	GND	37	38	PMC_STOP_N
GND	7	8	N/C	PMC_PERR_N	39	40	GND
N/C	9	10	N/C	VDD33	41	42	PMC_SERR_N
PMC_BM2_N	11	12	VDD33	PMC_CBE1_N	43	44	GND
PMC_RST_N	13	14	PMC_BM3_N	PMC_AD14	45	46	PMC_AD13
VDD33	15	16	PMC_BM4_N	PMC_M66E	47	48	PMC_AD10
PCM_PME_N	17	18	GND	PMC_AD8	49	50	VDD33
PMC_AD30	19	20	PMC_AD29	PMC_AD7	51	52	N/C
GND	21	22	PMC_AD26	VDD33	53	54	N/C
PMC_AD24	23	24	VDD33	N/C	55	56	GND
PMC_IDSEL	25	26	PMC_AD23	N/C	57	58	N/C
VDD33	27	28	PMC_AD20	GND	59	60	N/C
PMC_AD18	29	30	GND	PMC_ACK64_N	61	62	VDD33
PMC_AD16	31	32	PMC_CBE2_N	GND	63	64	N/C

6.3.2.3. J13: PMC-Jn3

Signal	Pin	Pin	Signal	Signal	Pin	Pin	Signal
PMC_J3_1	1	2	GND	GND	33	34	PMC_AD48
GND	3	4	PMC_CBE7_N	PMC_AD47	35	36	PMC_AD46
PMC_CBE6_N	5	6	PMC_CBE5_N	PMC_AD45	37	38	GND
PMC_CBE4_N	7	8	GND	PMC_VIO3	39	40	PMC_AD44
PMC_VIO1	9	10	PMC_PAR64	PMC_AD43	41	42	PMC_AD42
PMC_AD63	11	12	PMC_AD62	PMC_AD41	43	44	GND
PMC_AD61	13	14	GND	GND	45	46	PMC_AD40
GND	15	16	PMC_AD60	PMC_AD39	47	48	PMC_AD38
PMC_AD59	17	18	PMC_AD58	PMC_AD37	49	50	GND
PMC_AD57	19	20	GND	GND	51	52	PMC_AD36
PMC_VIO2	21	22	PMC_AD56	PMC_AD35	53	54	PMC_AD34
PMC_AD55	23	24	PMC_AD54	PMC_AD33	55	56	GND
PMC_AD53	25	26	GND	PMC_VIO4	57	58	PMC_AD32
GND	27	28	PMC_AD52	PMC_J3_2	59	60	PMC_J3_3
PMC_AD51	29	30	PMC_AD50	PMC_J3_4	61	62	GND
PMC_AD49	31	32	GND	GND	63	64	PMC_J3_5

6.3.2.4. J14: PMC-Jn4

Signal	Pin	Pin	Signal	Signal	Pin	Pin	Signal
PMC_UIO1	1	2	GND	GND	33	34	PMC_UIO24
GND	3	4	PMC_UIO2	PMC_UIO26	35	36	PMC_UIO25
PMC_UIO4	5	6	PMC_UIO3	PMC_UIO27	37	38	GND
PMC_UIO5	7	8	GND	PMC_UIO29	39	40	PMC_UIO28
PMC_UIO6	9	10	PMC_UIO7	PMC_UIO31	41	42	PMC_UIO30
PMC_UIO8	11	12	PMC_UIO9	PMC_UIO32	43	44	GND
PMC_UIO10	13	14	GND	GND	45	46	PMC_UIO33
GND	15	16	PMC_UIO11	PMC_UIO35	47	48	PMC_UIO34
PMC_UIO13	17	18	PMC_UIO12	PMC_UIO36	49	50	GND
PMC_UIO14	19	20	GND	GND	51	52	PMC_UIO37
PMC_UIO16	21	22	PMC_UIO15	PMC_UIO39	53	54	PMC_UIO38
PMC_UIO18	23	24	PMC_UIO17	PMC_UIO40	55	56	GND
PMC_UIO19	25	26	GND	PMC_UIO42	57	58	PMC_UIO41
GND	27	28	PMC_UIO20	PMC_UIO44	59	60	PMC_UIO43
PMC_UIO22	29	30	PMC_UIO21	PMC_UIO45	61	62	GND
PMC_UIO23	31	32	GND	GND	63	64	PMC_UIO46

6.3.3. J15 : I2C

The I2C system bus is available on a standard 4-pin header.

Signal	Pin	Pin	Signal
VDD33	1	2	CPU_SDA
CPU_SCL	3	4	GND

6.3.4. J16: JTAG ProASIC^{PLUS}

This connector is configured for the ProASIC^{PLUS} FPGA.

Signal	*Pin*	*Pin*	*Signal*
N/C	1	2	N/C
N/C	3	4	N/C
N/C	5	6	N/C
GND	7	8	N/C
GND	9	10	GND
N/C	11	12	TCK
N/C	13	14	TDI
N/C	15	16	TDO
N/C	17	18	TMS
N/C	19	20	RCK
N/C	21	22	TRST
N/C	23	24	N/C
N/C	25	26	N/C

6.3.5. J17: JTAG ProASIC

This connector is only available when the board is assembled with a ProASIC device.

6.3.6. J18: JTAG-ICE

An ARM-standard 20-pin box header is provided for the CPU internal JTAG-ICE port.

Signal	*Pin*	*Pin*	*Signal*
VDD33	1	2	VDD33
N/C	3	4	GND
JI_TDI	5	6	GND
JI_TMS	7	8	GND
JI_TCK	9	10	GND
JI_TCK	11	12	GND
JI_TDO	13	14	GND
JI_TRST	15	16	GND
N/C	17	18	GND
N/C	19	20	GND

6.3.7. J19: SDRAM DIMM (PC100)

Pin	Signal	Pin	Signal	Pin	Signal	Pin	Signal
1	GND	43	GND	85	GND	127	GND
2	SD_DQ0	44	N/C	86	SD_DQ32	128	SD_CKE
3	SD_DQ1	45	SD_S2_N	87	SD_DQ33	129	SD_S3_N
4	SD_DQ2	46	SD_DQMB2	88	SD_DQ34	130	SD_DQMB6
5	SD_DQ3	47	SD_DQMB3	89	SD_DQ35	131	SD_DQMB7
6	VDD33	48	N/C	90	VDD33	132	SD_A13
7	SD_DQ4	49	VDD33	91	SD_DQ36	133	VDD33
8	SD_DQ5	50	N/C	92	SD_DQ37	134	N/C
9	SD_DQ6	51	N/C	93	SD_DQ38	135	N/C
10	SD_DQ7	52	N/C	94	SD_DQ39	136	N/C
11	SD_DQ8	53	N/C	95	SD_DQ40	137	N/C
12	GND	54	GND	96	GND	138	GND
13	SD_DQ9	55	SD_DQ16	97	SD_DQ41	139	SD_DQ48
14	SD_DQ10	56	SD_DQ17	98	SD_DQ42	140	SD_DQ49
15	SD_DQ11	57	SD_DQ18	99	SD_DQ43	141	SD_DQ50
16	SD_DQ12	58	SD_DQ19	100	SD_DQ44	142	SD_DQ51
17	SD_DQ13	59	VDD33	101	SD_DQ45	143	VDD33
18	VDD33	60	SD_DQ20	102	VDD33	144	SD_DQ52
19	SD_DQ14	61	N/C	103	SD_DQ46	145	N/C
20	SD_DQ15	62	N/C	104	SD_DQ47	146	N/C
21	N/C	63	SD_CKE	105	N/C	147	J_REGE
22	N/C	64	GND	106	N/C	148	GND
23	GND	65	SD_DQ21	107	GND	149	SD_DQ53
24	N/C	66	SD_DQ22	108	N/C	150	SD_DQ54
25	N/C	67	SD_DQ23	109	N/C	151	SD_DQ55
26	VDD33	68	GND	110	VDD33	152	GND
27	SD_WE0_N	69	SD_DQ24	111	SD_CAS_N	153	SD_DQ56
28	SD_DQMB0	70	SD_DQ25	112	SD_DQMB4	154	SD_DQ57
29	SD_DQMB1	71	SD_DQ26	113	SD_DQMB5	155	SD_DQ58
30	SD_S0_N	72	SD_DQ27	114	SD_S1_N	156	SD_DQ59
31	N/C	73	VDD33	115	SD_RAS_N	157	VDD33
32	GND	74	SD_DQ28	116	GND	158	SD_DQ60
33	SD_A0	75	SD_DQ29	117	SD_A1	159	SD_DQ61
34	SD_A2	76	SD_DQ30	118	SD_A3	160	SD_DQ62
35	SD_A4	77	SD_DQ31	119	SD_A5	161	SD_DQ63
36	SD_A6	78	GND	120	SD_A7	162	GND
37	SD_A8	79	CLK_SD2	121	SD_A9	163	CLK_SD2
38	SD_A10	80	N/C (WP)	122	SD_BA0	164	N/C
39	SD_BA1	81	N/C	123	SD_A11	165	GND (SA0)
40	VDD33	82	SD_SDA	124	VDD33	166	GND (SA1)
41	VDD33	83	SD_SCL	125	CLK_SD1	167	GND (SA2)
42	CLK_SD1	84	VDD33	126	SD_A12	168	VDD33

6.3.8. J20, J21: CPUext 1, CPUext 2 respectively

Two 80-pin headers that can be used to connect an external memory extension daughter card or to connect a different CPU to the SDB development board.

Signal	*Pin*	*Pin*	*Signal*
GND	1	2	GND
CPU_WR_N	3	4	CPU_UB_N
CPU_OE_N	5	6	CPU_WAIT_N
CLK_FROM_CPU	7	8	CLK_TO_EXT1
CPU_CS0_N	9	10	CPU_CS1_N
CPU_CS2_N	11	12	CPU_CS3_N
VDD33	13	14	RESET_N
CPU_LB_N	15	16	CPU_A1
CPU_A2	17	18	CPU_A3
CPU_A4	19	20	CPU_A5
CPU_A6	21	22	CPU_A7
GND	23	24	GND
CPU_A8	25	26	CPU_A9
CPU_A10	27	28	CPU_A11
CPU_A12	29	30	CPU_A13
CPU_A14	31	32	CPU_A15
VDD33	33	34	VDD33
CPU_A16	35	36	CPU_A17
CPU_A18	37	38	CPU_A19
CPU_A20	39	40	CPU_A21
CPU_A22	41	42	CPU_A23
GND	43	44	GND
CPU_D0	45	46	CPU_D1
CPU_D2	47	48	CPU_D3
CPU_D4	49	50	CPU_D5
CPU_D6	51	52	CPU_D7
VDD33	53	54	VDD33
CPU_D8	55	56	CPU_D9
CPU_D10	57	58	CPU_D11
CPU_D12	59	60	CPU_D13
CPU_D14	61	62	CPU_D15
GND	63	64	GND
CPU_GPIO_IRQ_0	65	66	CPU_GPIO_IRQ_1
CPU_GPIO_IRQ_2	67	68	CPU_GPIO_FIRQ
CPU_GPIO_SDA	69	70	CPU_GPIO_SCL
CPU_GPIO_TMS	71	72	CPU_GPIO_TDI
CPU_GPIO_TCK	73	74	CPU_GPIO_TDO
CPU_GPIO_TRST	75	76	CPU_GPIO_KEY_0
CPU_GPIO_KEY_1	77	78	CPU_GPIO_KEY_2
CPU_GPIO_KEY_3	79	80	CPU_GPIO_FLASH_BUSY

6.3.9. J22: Display Connector

A 1x14 header connects the GlueChip directly to the LCD module.

Signal	*Pin*	*Pin*	*Signal*
GND	1	2	VDD50
VDD50	3	4	DISP_A0
DISP_RWN	5	6	DISP_CS
DISP_D0	7	8	DISP_D1
DISP_D2	9	10	DISP_D3
DISP_D4	11	12	DISP_D5
DISP_D6	13	14	DISP_D7

6.3.10. J23, J24: LVPECL Data

Two SMA footprints are provided to connect an external differential LVPECL high speed data or clock source.

6.3.10.1. J23: EPECLIN, Positive PECL

Pin	*Signal*
1	GND
2	EPECLIN

6.3.10.2. J24: EPECLREF, Negative PECL

Pin	*Signal*
1	GND
2	EPECLREF

6.3.11. J25: CompactFlash

The CompactFlash connector is configured for 3.3V operation. The default operating mode is 'memory mapped' which defines the port names accordingly.

Signal	*Pin*	*Pin*	*Signal*
GND	1	26	CF_CD1_N
CF_D3	2	27	CF_D11
CF_D4	3	28	CF_D12
CF_D5	4	29	CF_D13
CF_D6	5	30	CF_D14
CF_D7	6	31	CF_D15
CF_CE1_N	7	32	CF_CE2_N
CF_A10	8	33	CF_VS1
CF_OE_N	9	34	CF_IORD_N
CF_A9	10	35	CF_IOWR_N
CF_A8	11	36	CF_WE_N
CF_A7	12	37	CF_RDYBSY_N
VDDCF	13	38	VDDCF
CF_A6	14	39	CF_CSEL_N
CF_A5	15	40	CF_VS2
CF_A4	16	41	CF_RESET
CF_A3	17	42	CF_WAIT_N
CF_A2	18	43	CF_INPACK_N
CF_A1	19	44	CF_REG_N
CF_A0	20	45	CF_BVD2
CF_D0	21	46	CF_BVD1
CF_D1	22	47	CF_D8
CF_D2	23	48	CF_D9
CF_WP	24	49	CF_D10
CF_CD1_N	25	50	GND

6.3.12. J26: User IO 1

The upper 32-bit of the 64-bit SDRAM signals are available on connector UIO1

Signal	*Pin*	*Pin*	*Signal*
GND	1	2	SD_DQ32
SD_DQ33	3	4	SD_DQ34
SD_DQ35	5	6	SD_DQ36
GND	7	8	SD_DQ37
SD_DQ38	9	10	SD_DQ39
SD_DQ40	11	12	SD_DQ41
GND	13	14	SD_DQ42
SD_DQ43	15	16	SD_DQ44

Signal	Pin	Pin	Signal
SD_DQ45	17	18	SD_DQ46
GND	19	20	SD_DQ47
SD_DQ48	21	22	SD_DQ49
SD_DQ50	23	24	SD_DQ51
GND	25	26	SD_DQ52
SD_DQ53	27	28	SD_DQ54
SD_DQ55	29	30	SD_DQ56
GND	31	32	SD_DQ57
SD_DQ58	33	34	SD_DQ59
SD_DQ60	35	36	SD_DQ61
GND	37	38	SD_DQ62
SD_DQ63	39	40	GND

6.3.13. J27: User IO 2

The signal mapping on this connector is compatible to the PTMC J3 pin mapping.

Signal	Pin	Pin	Signal	Signal	Pin	Pin	Signal
PMC_J3_1	1	2	GND	GND	33	34	PMC_AD48
GND	3	4	PMC_CBE7_N	PMC_AD47	35	36	PMC_AD46
PMC_CBE6_N	5	6	PMC_CBE5_N	PMC_AD45	37	38	GND
PMC_CBE4_N	7	8	GND	PMC_VIO3	39	40	PMC_AD44
PMC_VIO1	9	10	PMC_PAR64	PMC_AD43	41	42	PMC_AD42
PMC_AD63	11	12	PMC_AD62	PMC_AD41	43	44	GND
PMC_AD61	13	14	GND	GND	45	46	PMC_AD40
GND	15	16	PMC_AD60	PMC_AD39	47	48	PMC_AD38
PMC_AD59	17	18	PMC_AD58	PMC_AD37	49	50	GND
PMC_AD57	19	20	GND	GND	51	52	PMC_AD36
PMC_VIO2	21	22	PMC_AD56	PMC_AD35	53	54	PMC_AD34
PMC_AD55	23	24	PMC_AD54	PMC_AD33	55	56	GND
PMC_AD53	25	26	GND	PMC_VIO4	57	58	PMC_AD32
GND	27	28	PMC_AD52	PMC_J3_2	59	60	PMC_J3_3
PMC_AD51	29	30	PMC_AD50	PMC_J3_4	61	62	GND
PMC_AD49	31	32	GND	GND	63	64	PMC_J3_5

6.3.14. J28: User IO 3

The signal mapping on this connector is compatible to the PTMC J4 pin mapping.

Signal	*Pin*	*Pin*	*Signal*	*Signal*	*Pin*	*Pin*	*Signal*
PMC_UIO1	1	2	GND	GND	33	34	PMC_UIO24
GND	3	4	PMC_UIO2	PMC_UIO26	35	36	PMC_UIO25
PMC_UIO4	5	6	PMC_UIO3	PMC_UIO27	37	38	GND
PMC_UIO5	7	8	GND	PMC_UIO29	39	40	PMC_UIO28
PMC_UIO6	9	10	PMC_UIO7	PMC_UIO31	41	42	PMC_UIO30
PMC_UIO8	11	12	PMC_UIO9	PMC_UIO32	43	44	GND
PMC_UIO10	13	14	GND	GND	45	46	PMC_UIO33
GND	15	16	PMC_UIO11	PMC_UIO35	47	48	PMC_UIO34
PMC_UIO13	17	18	PMC_UIO12	PMC_UIO36	49	50	GND
PMC_UIO14	19	20	GND	GND	51	52	PMC_UIO37
PMC_UIO16	21	22	PMC_UIO15	PMC_UIO39	53	54	PMC_UIO38
PMC_UIO18	23	24	PMC_UIO17	PMC_UIO40	55	56	GND
PMC_UIO19	25	26	GND	PMC_UIO42	57	58	PMC_UIO41
GND	27	28	PMC_UIO20	PMC_UIO44	59	60	PMC_UIO43
PMC_UIO22	29	30	PMC_UIO21	PMC_UIO45	61	62	GND
PMC_UIO23	31	32	GND	GND	63	64	PMC_UIO46

6.3.15. J29: User IO 4

The UIO4 represents the GPIO port 2 of the ProASIC^{PLUS} FPGA.

Signal	*Pin*	*Pin*	*Signal*
VDD33	1	2	GND
PA_GPIO2_0	3	4	PA_GPIO2_1
PA_GPIO2_2	5	6	PA_GPIO2_3
PA_GPIO2_4	7	8	PA_GPIO2_5
PA_GPIO2_6	9	10	PA_GPIO2_7
PA_GPIO2_8	11	12	GND
VDD33	13	14	GND

6.3.16. J30, J31, J32: Debug Header

The following three connectors can be used for an ICE to connect directly to the CPU.

6.3.16.1. J30: CPU_CTRL

Signal	*Pin*	*Pin*	*Signal*
GND	1	2	CPU_LB_N
CPU_WR_N	3	4	CPU_UB_N
CPU_OE_N	5	6	CPU_WAIT_N
CPU_CS0_N	7	8	CPU_CS1_N
CPU_CS2_N	9	10	CPU_CS3_N
CPU_IRQ0	11	12	CPU_IRQ1
CPU_IRQ2	13	14	CPU_FIRQ

6.3.16.2. J31: CPU_Addr

Signal	*Pin*	*Pin*	*Signal*
GND	1	2	CPU_A1
CPU_A2	3	4	CPU_A3
CPU_A4	5	6	CPU_A5
CPU_A6	7	8	CPU_A7
CPU_A8	9	10	CPU_A9
CPU_A10	11	12	CPU_A11
CPU_A12	13	14	CPU_A13
CPU_A14	15	16	CPU_A15
CPU_A16	17	18	CPU_A17
CPU_A18	19	20	CPU_A19
CPU_A20	21	22	CPU_A21
CPU_A22	23	24	CPU_A23

6.3.16.3. J32: CPU_Data

Signal	*Pin*	*Pin*	*Signal*
GND	1	2	GND
CPU_D0	3	4	CPU_D1
CPU_D2	5	6	CPU_D3
CPU_D4	7	8	CPU_D5
CPU_D6	9	10	CPU_D7
CPU_D8	11	12	CPU_D9
CPU_D10	13	14	CPU_D11
CPU_D12	15	16	CPU_D13
CPU_D14	17	18	CPU_D15
GND	19	20	GND

6.3.17. J33, J34: UART 1, UART 2 respectively

The next figure indicates the front view pinout of the DB9 female connector:

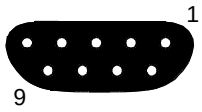


Figure 28 - UART connector pinout

(Front View)

The pins in the following table are provided with signals on the connector:

Pin	Signal	Description
2	uart_rx	Receive data
3	uart_tx	Transmit data
5	GND	Signal ground

6.3.18. J35, J36: CAN 1, CAN 2 respectively

The next figure shows the front view pinout of the DB9 male connector:

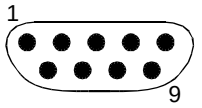


Figure 29 - CANbus connector pinout

(Front View)

The pins in the following table are provided with signals on the connector:

Pin	Signal	Description
2	can_l	CAN_L bus line, dominant <Low>
3	GND	Ground
7	can_h	CAN_H bus line, dominant <High>

6.3.19. J37, J38, J39, J40, J41: Local Power Connectors

These 2-pin connectors are connected to the different power suppliers on the board. Pin one of each is connected to -12V, +12V, 5V, 3.3V and 2.5V respectively, pin two is connected to ground.

6.4. Test Points

There are 14 test points on the board that allow easy access to several signals.

Test Point	*Source*	*Color*
TP1	GlueChip PRA	yellow
TP2	GlueChip PRB	yellow
TP3, TP4, TP5, TP6, TP7	OSC1, CLK1_TO_PA, OSC2, CLK2_TO_PA, CLK_TO_CPU	yellow
TP8, TP9, TP10, TP11, TP12, TP13	GND	black

6.5. Prototyping Area

The following table indicates the pin assigment between the ProASIC^{PLUS} FPGA and the prototyping area of the SDB development board.

PROTO PIN	Signal name
0	PA_GPIO1_0
1	PA_GPIO1_1
2	PA_GPIO1_2
3	PA_GPIO1_3
4	PA_GPIO1_4
5	PA_GPIO1_5
6	PA_GPIO1_6
7	PA_GPIO1_7
8	PA_GPIO1_8
9	PA_GPIO1_9
11	PA_GPIO1_10(GL)
13	PA_GPIO1_11
15	PA_GPIO1_12
17	PA_GPIO1_13
19	PA_GPIO1_14
21	PA_GPIO1_15
23	PA_GPIO1_16
25	PA_GPIO1_17
27	PA_GPIO1_18

6.6. SDB-750/1000 connectivity table

The following tables provides the pin-to-pin association between the ProASIC^{PLUS} FPGA and its connecting devices.

6.6.1. CPU

PRO ASIC^{PLUS} PIN	CPU PIN	Signal name
P23	31	CPU_D0
P22	32	CPU_D1
R24	33	CPU_D2
R23	34	CPU_D3
R22	35	CPU_D4
T24	37	CPU_D5
T23	38	CPU_D6
T22	39	CPU_D7
U26	40	CPU_D8
U25	41	CPU_D9
U24	42	CPU_D10
U23	43	CPU_D11
U22	45	CPU_D12
V26	46	CPU_D13
V25	47	CPU_D14
V24	48	CPU_D15
V23	3	CPU_A1
V22	4	CPU_A2
W26	5	CPU_A3
W25	6	CPU_A4
W24	7	CPU_A5
W23	8	CPU_A6
Y26	9	CPU_A7
Y25	11	CPU_A8
Y24	12	CPU_A9
Y23	13	CPU_A10
AA26	14	CPU_A11
AA25	15	CPU_A12
AA24	16	CPU_A13
AA23	17	CPU_A14
AB26	20	CPU_A15

PRO ASIC^{PLUS} PIN	CPU PIN	Signal name
AB25	21	CPU_A16
AB24	22	CPU_A17
AB23	23	CPU_A18
AC26	24	CPU_A19
AD26	25	CPU_A20
AC25	26	CPU_A21
AD25	29	CPU_A22
AF24	30	CPU_A23
AD23	97	CPU_CS0_N
AE22	98	CPU_CS1_N
AF22	99	CPU_CS2_N
AE21	100	CPU_CS3_N
AF21	60	CPU_IRQ0
AC20	63	CPU_IRQ1
AD20	64	CPU_IRQ2
AE20	66	CPU_FIRQ
AF20	93	CPU_WR_N
AC19	92	CPU_OE_N
AD19	77	CPU_UB_N
AE19	1	CPU_LB_N
AF19	96	CPU_WAIT_N

6.6.2. PMC J1 & J2

PRO ASIC^{PLUS} PIN	PMC PIN	Signal name
E2	J1/4	PMC_INTA_N
E1	J1/6	PMC_INTC_N
F4	J1/5	PMC_INTB_N
F3	J1/7	PMC_BM1_N

PRO ASIC PLUS PIN	PMC PIN	Signal name
F2	J1/9	PMC_INTD_N
F1	J2/11	PMC_BM2_N
G4	J2/14	PMC_BM3_N
G3	J2/13	PMC_RST_N
G2	J2/16	PMC_BM4_N
G1	J1/16	PMC_GNT_N
H4	J2/17	PMC_PME_N
H3	J1/17	PMC_REQ_N
H2	J2/20	PMC_AD29
H1	J2/19	PMC_AD30
J5	J1/20	PMC_AD31
J4	J2/22	PMC_AD26
J3	J1/22	PMC_AD27
J2	J1/21	PMC_AD28
J1	J2/23	PMC_AD24
K5	J1/23	PMC_AD25
K4	J2/26	PMC_AD23
K3	J2/25	PMC_IDSEL
K2	J1/26	PMC_CBE3_N
K1	J2/28	PMC_AD20
L5	J1/28	PMC_AD21
L4	J1/27	PMC_AD22
L3	J2/29	PMC_AD18
L2	J1/29	PMC_AD19
L1	J2/32	PMC_CBE2_N
M5	J2/31	PMC_AD16
M4	J1/32	PMC_AD17
M3	J1/33	PMC_FRAME_N
N2	J2/35	PMC_TRDY_N
M1	J1/36	PMC_IRDY_N
N1	J2/38	PMC_STOP_N
P1	J1/37	PMC_DEVSEL_N
P2	J2/39	PMC_PERR_N
P3	J1/40	PMC_LOCK_N
P4	J2/42	PMC_SERR_N
R1	J2/43	PMC_CBE1_N
R2	J1/43	PMC_PAR
R3	J2/46	PMC_AD13
R4	J2/45	PMC_AD14
R5	J1/46	PMC_AD15
T1	J2/48	PMC_AD10
T2	J2/47	PMC_M66E
T3	J1/48	PMC_AD11
T4	J1/47	PMC_AD12
T5	J2/49	PMC_AD8
U1	J1/49	PMC_AD9
U2	J2/51	PMC_AD7
U3	J1/52	PMC_CBE0_N
U4	J1/54	PMC_AD5
U5	J1/53	PMC_AD6
V1	J1/55	PMC_AD4
V2	J1/58	PMC_AD3
V3	J1/60	PMC_AD1
V4	J1/59	PMC_AD2
V5	J2/61	PMC_ACK64_N
W1	J1/61	PMC_AD0
W2	J1/64	PMC_REQ64_N

6.6.3. PMC J3 & J4

PRO ASIC PLUS PIN	PMC PIN	Signal name
W3	J4/1	PMC_UIO1
W4	J3/1	PMC_J3_1
Y1	J4/4	PMC_UIO2
Y2	J3/4	PMC_CBE7_N
Y3	J4/6	PMC_UIO3
Y4	J4/5	PMC_UIO4
AA1	J3/6	PMC_CBE5_N
AA2	J3/5	PMC_CBE6_N
AA3	J4/7	PMC_UIO5
AA4	J3/7	PMC_CBE4_N
AB1	J4/10	PMC_UIO6
AB2	J4/9	PMC_UIO7
AB3	J3/10	PMC_PAR64

PRO ASIC PLUS PIN	PMC PIN	Signal name
AB4	J3/9	PMC_VIO1
AC1	J4/12	PMC_UIO8
AC2	J4/11	PMC_UIO9
AC3	J3/12	PMC_AD62
AD1	J3/11	PMC_AD63
AD2	J4/13	PMC_UIO10
AD4	J3/13	PMC_AD61
AE3	J4/16	PMC_UIO11
AE4	J3/16	PMC_AD60
AF3	J4/18	PMC_UIO12
AF4	J4/17	PMC_UIO13
AC5	J3/18	PMC_AD58
AD5	J3/17	PMC_AD59
AE5	J4/19	PMC_UIO14
AF5	J3/19	PMC_AD57
AC6	J4/22	PMC_UIO15
AD6	J4/21	PMC_UIO16
AE6	J3/22	PMC_AD56
AF6	J3/21	PMC_VIO2
AC7	J4/24	PMC_UIO17
AD7	J4/23	PMC_UIO18
AE7	J3/24	PMC_AD54
AF7	J3/23	PMC_AD55
AB8	J4/25	PMC_UIO19
AC8	J3/25	PMC_AD53
AD8	J4/28	PMC_UIO20
AE8	J3/28	PMC_AD52
AF8	J4/30	PMC_UIO21
AB9	J4/29	PMC_UIO22
AC9	J3/30	PMC_AD50
AD9	J3/29	PMC_AD51
AE9	J4/31	PMC_UIO23
AF9	J3/31	PMC_AD49
AB10	J4/34	PMC_UIO24
AC10	J3/34	PMC_AD48
AD10	J4/36	PMC_UIO25
AE10	J4/35	PMC_UIO26
AF10	J3/36	PMC_AD46

PRO ASIC PLUS PIN	PMC PIN	Signal name
AB11	J3/35	PMC_AD47
AC11	J4/37	PMC_UIO27
AD11	J3/37	PMC_AD45
AE11	J4/40	PMC_UIO28
AF11	J4/39	PMC_UIO29
AB12	J3/40	PMC_AD44
AC12	J3/39	PMC_VIO3
AD12	J4/42	PMC_UIO30
AE12	J4/41	PMC_UIO31
AF12	J3/42	PMC_AD42
AB13	J3/41	PMC_AD43
AC13	J4/43	PMC_UIO32
AD13	J3/43	PMC_AD41
AE13	J4/46	PMC_UIO33
AF13	J3/46	PMC_AD40
AB14	J4/48	PMC_UIO34
AC14	J4/47	PMC_UIO35
AD14	J3/48	PMC_AD38
AE14	J3/47	PMC_AD39
AF14	J4/49	PMC_UIO36
AB15	J3/49	PMC_AD37
AC15	J4/52	PMC_UIO37
AD15	J3/52	PMC_AD36
AE15	J4/54	PMC_UIO38
AF15	J4/53	PMC_UIO39
AB16	J3/54	PMC_AD34
AC16	J3/53	PMC_AD35
AD16	J4/55	PMC_UIO40
AE16	J3/55	PMC_AD33
AF16	J4/58	PMC_UIO41
AB17	J4/57	PMC_UIO42
AC17	J3/58	PMC_AD32
AD17	J3/57	PMC_VIO4
AE17	J4/60	PMC_UIO43
AF17	J4/59	PMC_UIO44
AB18	J3/60	PMC_J3_3
AC18	J3/59	PMC_J3_2
AD18	J4/61	PMC_UIO45

PRO ASIC PLUS PIN	PMC PIN	Signal name
AE18	J3/61	PMC_J3_4
AF18	J4/64	PMC_UIO46
AB19	J3/64	PMC_J3_5

6.6.4. SDRAM

PRO ASIC PLUS PIN	SDRAM PIN	Signal name
E4	2	SD_DQ0
E3	3	SD_DQ1
D3	86	SD_DQ32
D2	4	SD_DQ2
D1	87	SD_DQ33
C4	88	SD_DQ34
C2	5	SD_DQ3
B4	89	SD_DQ35
B3	8	SD_DQ5
A4	7	SD_DQ4
A3	91	SD_DQ36
D5	92	SD_DQ37
C5	9	SD_DQ6
B5	93	SD_DQ38
A5	10	SD_DQ7
D6	94	SD_DQ39
C6	11	SD_DQ8
B6	95	SD_DQ40
A6	13	SD_DQ9
D7	97	SD_DQ41
C7	14	SD_DQ10
B7	98	SD_DQ42
A7	15	SD_DQ11
D8	99	SD_DQ43
C8	16	SD_DQ12
B8	100	SD_DQ44
A8	17	SD_DQ13
E9	101	SD_DQ45
D9	19	SD_DQ14
C9	103	SD_DQ46
B9	20	SD_DQ15
A9	104	SD_DQ47
E10	27	SD_WEO_N
D10	111	SD_CAS_N
C10	28	SD_DQMB0
B10	112	SD_DQMB4
A10	29	SD_DQMB1
E11	113	SD_DQMB5
D11	30	SD_S0_N
C11	114	SD_S1_N
B11	115	SD_RAS_N
A11	33	SD_A0
E12	117	SD_A1
D12	34	SD_A2
C12	118	SD_A3
B12	35	SD_A4
A12	119	SD_A5
E13	36	SD_A6
D13	120	SD_A7
C13	37	SD_A8
B13	121	SD_A9
A13	38	SD_A10
E14	122	SD_BA0
D14	39	SD_BA1
C14	123	SD_A11
B14	126	SD_A12
A14	45	SD_S2_N
E15	129	SD_S3_N
D15	46	SD_DQMB2
C15	130	SD_DQMB6
B15	47	SD_DQMB3
A15	131	SD_DQMB7
E16	132	SD_A13
D16	55	SD_DQ16
C16	139	SD_DQ48
B16	56	SD_DQ17
A16	140	SD_DQ49
E17	57	SD_DQ18
D17	141	SD_DQ50
C17	58	SD_DQ19
B17	142	SD_DQ51
A17	60	SD_DQ20
E18	144	SD_DQ52
D18	65	SD_DQ21
C18	149	SD_DQ53
B18	66	SD_DQ22
A18	150	SD_DQ54
E19	67	SD_DQ23
D19	151	SD_DQ55
C19	69	SD_DQ24
B19	153	SD_DQ56
A19	70	SD_DQ25

PRO ASIC[PLUS] PIN	SDRAM PIN	Signal name
D20	154	SD_DQ57
C20	71	SD_DQ26
B20	155	SD_DQ58
A20	72	SD_DQ27
D21	156	SD_DQ59
C21	74	SD_DQ28
B21	158	SD_DQ60
A21	75	SD_DQ29
D22	159	SD_DQ61
C22	76	SD_DQ30
B22	160	SD_DQ62
A22	77	SD_DQ31
C23	161	SD_DQ63
H25	63, 128	SD_CKE

6.6.5. Prototype Area

PRO ASIC[PLUS] PIN	PROTO PIN	Signal name
K24	0	PA_GPIO1_0
K23	1	PA_GPIO1_1
K22	2	PA_GPIO1_2
L25	3	PA_GPIO1_3
L24	4	PA_GPIO1_4
L23	5	PA_GPIO1_5
L22	6	PA_GPIO1_6
M24	7	PA_GPIO1_7
M23	8	PA_GPIO1_8
P24	9	PA_GPIO1_9
M22	11	PA_GPIO1_10(GL)
L26	13	PA_GPIO1_11
M26	15	PA_GPIO1_12
M25	17	PA_GPIO1_13
P25	19	PA_GPIO1_14
R26	21	PA_GPIO1_15
R25	23	PA_GPIO1_16
T26	25	PA_GPIO1_17
T25	27	PA_GPIO1_18

6.6.6. Keyboard

PRO ASIC[PLUS] PIN	Signal name
E23	cpu_gpio_key0
E24	cpu_gpio_key1
E25	cpu_gpio_key2
E26	cpu_gpio_key3

6.6.7. PA_GPIO's

PRO ASIC[PLUS] PIN	UIO4 PIN	Signal name
K25	2	PA_GPIO2_0
K26	3	PA_GPIO2_1
J22	4	PA_GPIO2_2
J23	5	PA_GPIO2_3
J24	6	PA_GPIO2_4
J25	7	PA_GPIO2_5
J26	8	PA_GPIO2_6
H23	9	PA_GPIO2_7
H24	10	PA_GPIO2_8

6.6.8. UART 1/2 (Transceiver)

PRO ASIC[PLUS] PIN	UART1 / UART2 PIN	Signal name
C25	UART1 / TX	PA_TX1
B24	UART2 / TX	PA_TX2
A23	UART1 / RX	PA_RX1
A24	UART2 / RX	PA_RX2

6.6.9. CANbus 1/2

PRO ASIC^{PLUS} PIN	CAN1 / CAN2 PIN	Signal name
D24	CAN1 / TX	PA_TX1
D25	CAN2 / TX	PA_TX2
D26	CAN1 / RX	PA_RX1
C26	CAN2 / RX	PA_RX2

6.6.10. Clocks and Reset

PRO ASIC^{PLUS} PIN	Signal name
N23	OSC1
F23	CLK1_FROM_PA
F24	CLK1_TO_PA
F25	OSC2
F26	CLK2_FROM_PA
M2	CLK2_TO_PA
G26	CLK_TO_CPU_FROM_PA
G25	CLK32K
N25	RESET_N

6.6.11. PECL

PRO ASIC^{PLUS} PIN	Signal name
N4	WPECLIN
P5	WPECLREF
N5	AVDDW
N3	AGNW
P26	EPECLIN
N22	EPECLREF
N24	AVDDE
N26	AGNE

6.6.12. JTAG

PRO ASIC^{PLUS} PIN	Signal name
AF23	TDI
AC22	PA_TDO
AD21	TCK
AE24	TRST
AC21	TMS

6.6.13. Programming

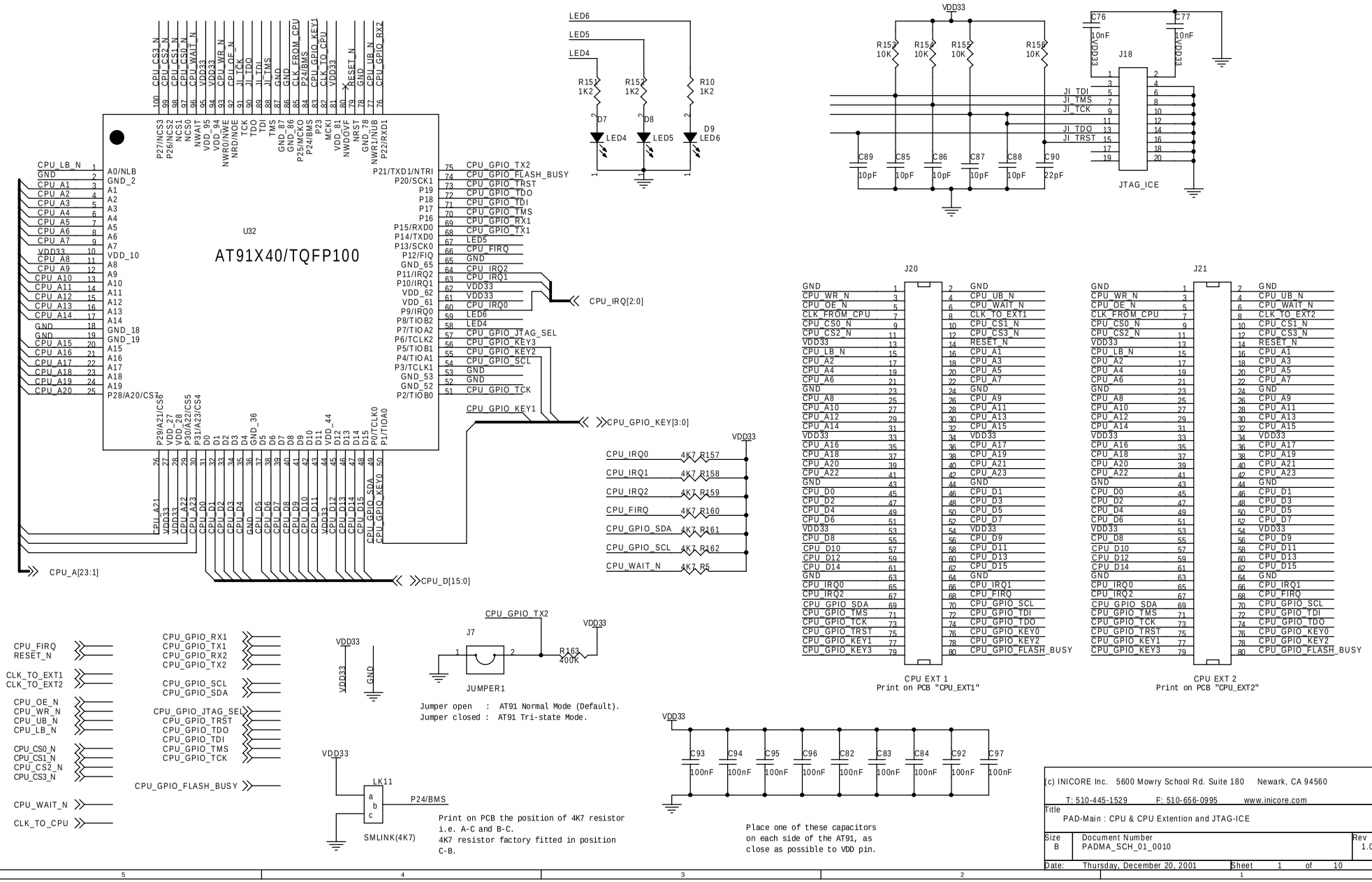
PRO ASIC^{PLUS} PIN	Signal name
AE23	VPN
AD22	VNN
AC24	RCK

6.6.14. Miscellaneous

PRO ASIC^{PLUS} PIN	Signal name
B23	FLASH-STATUS
H26	LED1
G23	LED2
G24	LED3

7. SCHEMATIC

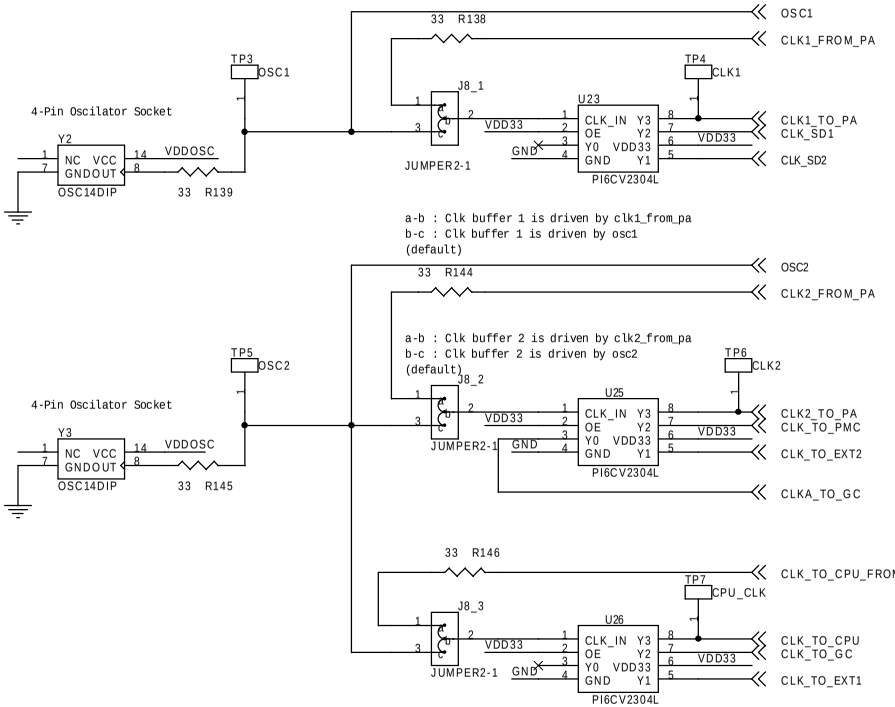
The schematic of the board is shown on the following few pages.



(c) INICORE Inc. 5600 Mowry School Rd. Suite 180 Newark, CA 94560

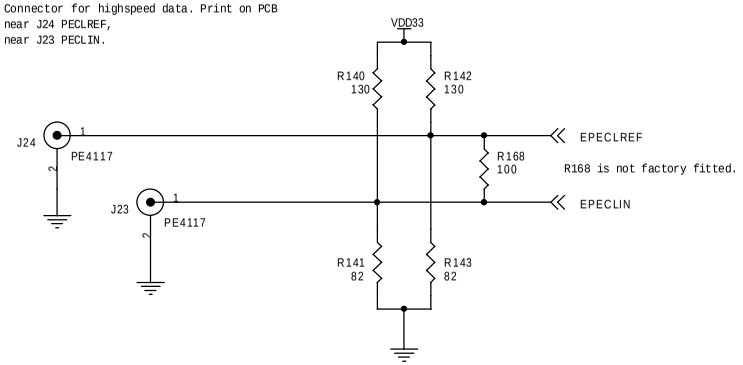
T: 510-445-1529 F: 510-656-0995 www.inicore.com

Title	PAD-Main : CPU & CPU Extention and JTAG-ICE	
Size B	Document Number PADMA_SCH_01_0010	Rev 1.06
Date: Thursday, December 20, 2001	Sheet 1 of 10	

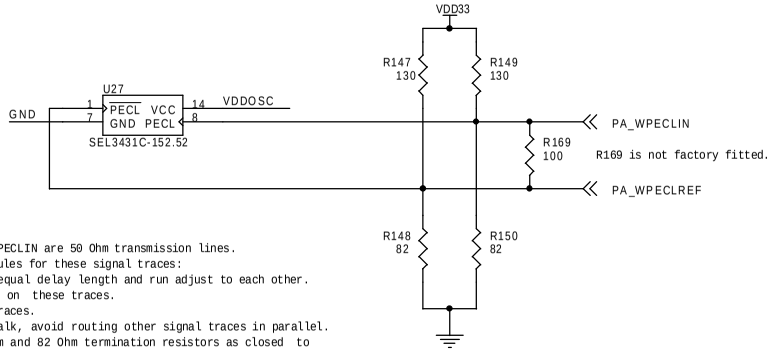


NO VIAS ON CLOCK TRACES AND MAKE THEM AS SHORT AS POSSIBLE

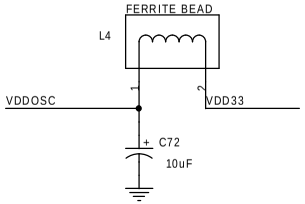
Connector for highspeed data. Print on PCB
near J24 PECLREF,
near J23 PECLIN.



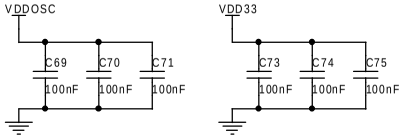
EPECLREF and EPECLIN are 50 Ohm transmission lines.
Note the following rules for these signal traces:
a) They should have equal delay length and run adjust to each other.
b) Avoid sharp angles on these traces.
c) No vias on these traces.
d) To prevent crosstalk, avoid routing other signal traces in parallel.
e) Locate the 130 Ohm and 82 Ohm termination resistors as closed to
the ProASIC's pins as possible.



PA_WPECLREF and PA_WPECLIN are 50 Ohm transmission lines.
Note the following rules for these signal traces:
a) They should have equal delay length and run adjust to each other.
b) Avoid sharp angles on these traces.
c) No vias on these traces.
d) To prevent crosstalk, avoid routing other signal traces in parallel.
e) Locate the 130 Ohm and 82 Ohm termination resistors as closed to
the ProASIC pins as possible.



The powerplane for the clock generators is isolated
from the board powerplane by a ferrite bead



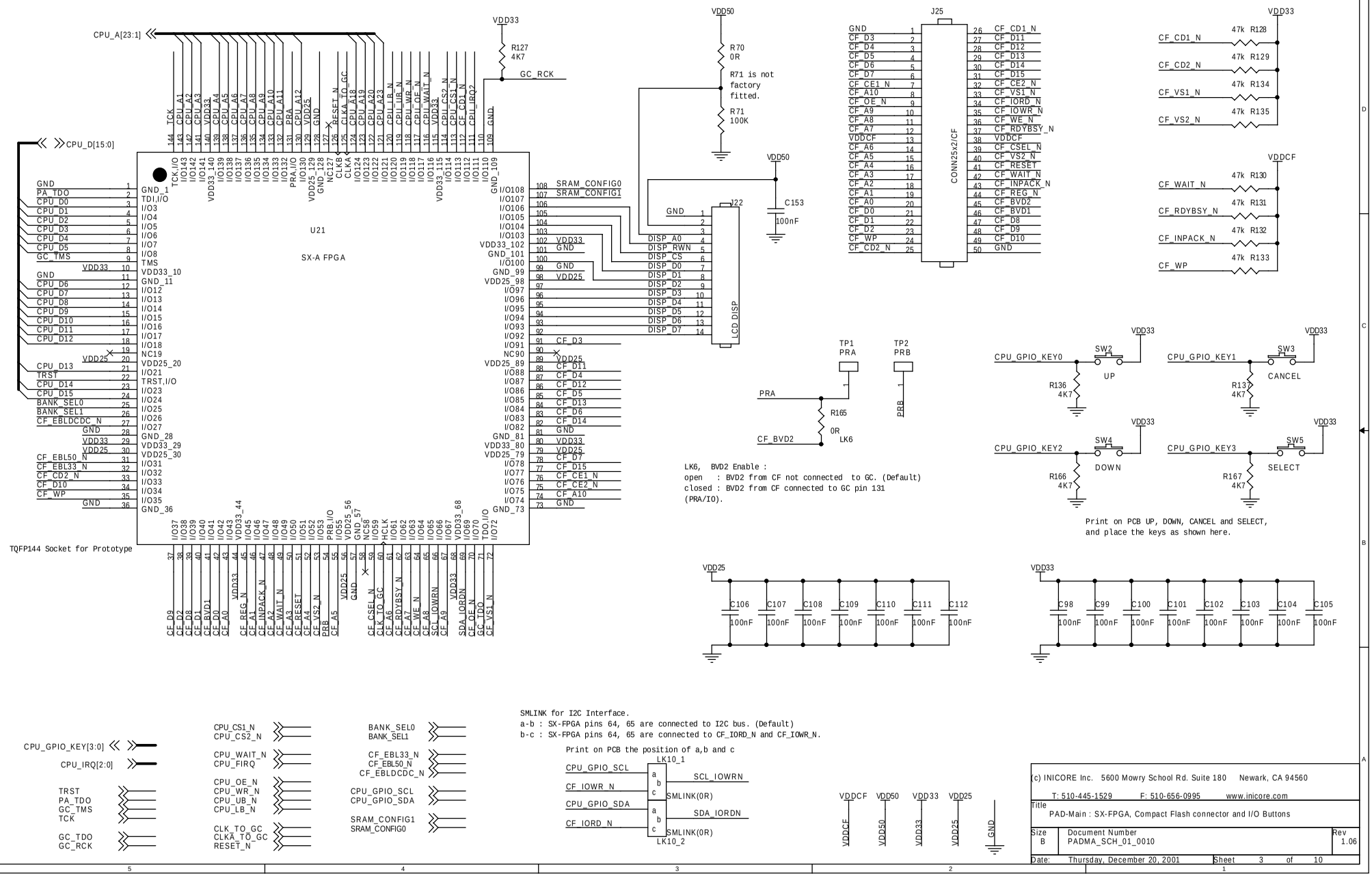
One capacitor for each OSC and one for each clk
Buffer (PI6CV304) .See clk_layout.pdf.

(c) INICORE Inc. 5600 Mowry School Rd. Suite 180 Newark, CA 94560
T: 510-445-1529 F: 510-656-0995 www.inicore.com

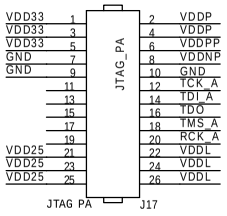
Title
PAD-Main : Clock circuit, LVPECL, 50 Ohm Transmission line and Isolated powerplane

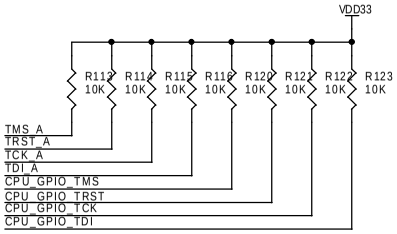
Size	Document Number	Rev
B	PADMA_SCH_01_0010	1.06

Date: Thursday, December 20, 2001 Sheet 2 of 10


U21
SX-A FPGA
TQFP144 Socket for Prototype
CPU_A[23:1]
CPU_D[15:0]
R127 4K7
GC_RCK
R70 0R
R71 is not factory fitted.
R71 100K
VDD50
VDD33
C153 100nF
J22
LCD DISP
J25
CONN25x2/CF
R128 47k
R129 47k
R134 47k
R135 47k
R130 47k
R131 47k
R132 47k
R133 47k
CF_CD1_N
CF_CD2_N
CF_VS1_N
CF_VS2_N
CF_WAIT_N
CF_RDYBSY_N
CF_INPACK_N
CF_WP
TP1 PRA
TP2 PRB
R165 0R
LK6
LK6, BVD2 Enable :
open : BVD2 from CF not connected to GC. (Default)
closed : BVD2 from CF connected to GC pin 131 (PRA/IO).
SW2 UP
SW3 CANCEL
SW4 DOWN
SW5 SELECT
R136 4K7
R13 4K7
R166 4K7
R167 4K7
Print on PCB UP, DOWN, CANCEL and SELECT, and place the keys as shown here.
C106 C107 C108 C109 C110 C111 C112 100nF
C98 C99 C100 C101 C102 C103 C104 C105 100nF
SMLINK for I2C Interface.
a-b : SX-FPGA pins 64, 65 are connected to I2C bus. (Default)
b-c : SX-FPGA pins 64, 65 are connected to CF_IORD_N and CF_IOWR_N.
Print on PCB the position of a,b and c
LK10_1
LK10_2
SMLINK(0R)
CPU_GPIO_SCL
CF_IOWR_N
CPU_GPIO_SDA
CF_IORD_N
SCL_IOWRN
SDA_IORDN
CPU_GPIO_KEY[3:0]
CPU_IRQ[2:0]
TRST
PA_TDO
GC_TMS
TCK
GC_TDO
GC_RCK
CPU_CS1_N
CPU_CS2_N
CPU_WAIT_N
CPU_FIRQ
CPU_OE_N
CPU_WR_N
CPU_UB_N
CPU_LB_N
CLK_TO_GC
CLKA_TO_GC
RESET_N
BANK_SEL0
BANK_SEL1
CF_EBL33_N
CF_EBL50_N
CF_EBLDCDC_N
CPU_GPIO_SCL
CPU_GPIO_SDA
SRAM_CONFIG1
SRAM_CONFIG0
VDDCF VDD50 VDD33 VDD25 GND
(c) INICORE Inc. 5600 Mowry School Rd. Suite 180 Newark, CA 94560
T: 510-445-1529 F: 510-656-0995 www.inicore.com
Title
PAD-Main : SX-FPGA, Compact Flash connector and I/O Buttons
Size B
Document Number PADMA_SCH_01_0010
Rev 1.06
Date: Thursday, December 20, 2001
Sheet 3 of 10

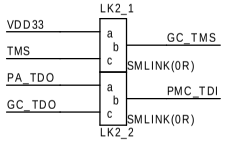
Actel Jtag Header
See jtag_header.pdf for footprint
Not factory fitted



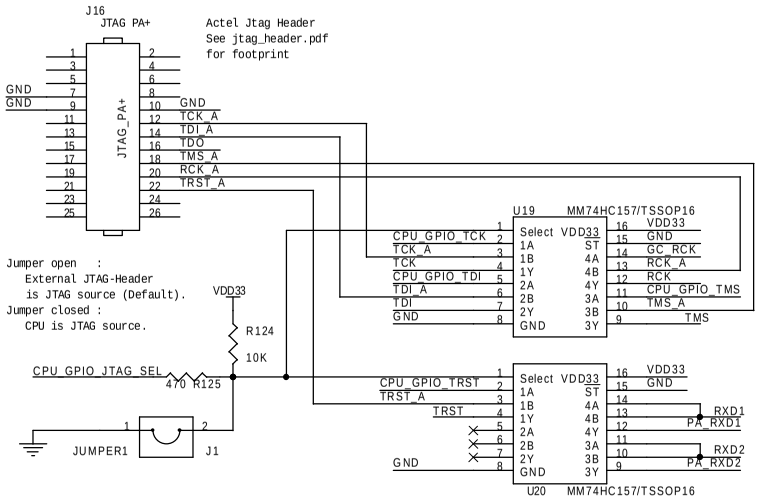


Pullups for TMS, TRST, TCK and TDI
for CPU and JTAG connectors.

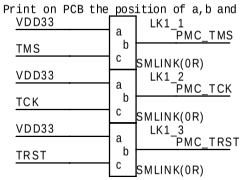
Print on PCB the position of a,b and c



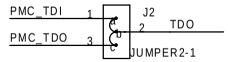
SMLINK for SX-FPGA JTAG
A-B : JTAG signals are short cuted. SX-FPGA
is not in JTAG chain.(Default)
B-C : SX-FPGA is in JTAG chain .



Print on PCB the position of a,b and c

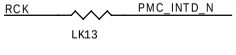


SMLINK block for PMC JTAG Signals
PMC_TMS, PMC_TCK and PMC_TRST :
A-B : JTAG signals are by-passed. PMC is not
connected to JTAG chain.
B-C : PMC is connected to JTAG chain. (Default)

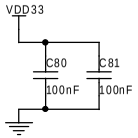


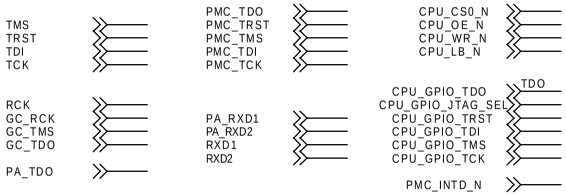
Print on PCB the position of a,b and c

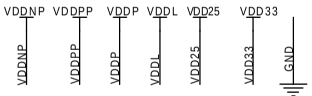
Jumper for JTAG Signal TDO:
A-B : PMC is not in JTAG chain. (Default)
B-C : PMC is in JTAG chain.



Resistor not factory fitted





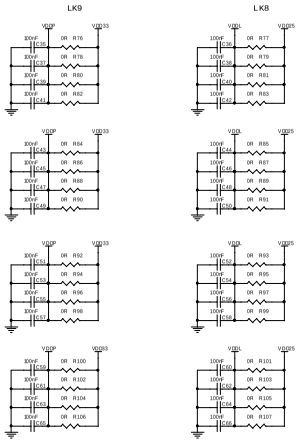


(c) INICORE Inc. 5600 Mowry School Rd. Suite 180 Newark, CA 94560
T: 510-445-1529 F: 510-656-0995 www.inicore.com

Title
PAD-Main : JTAG-Chain

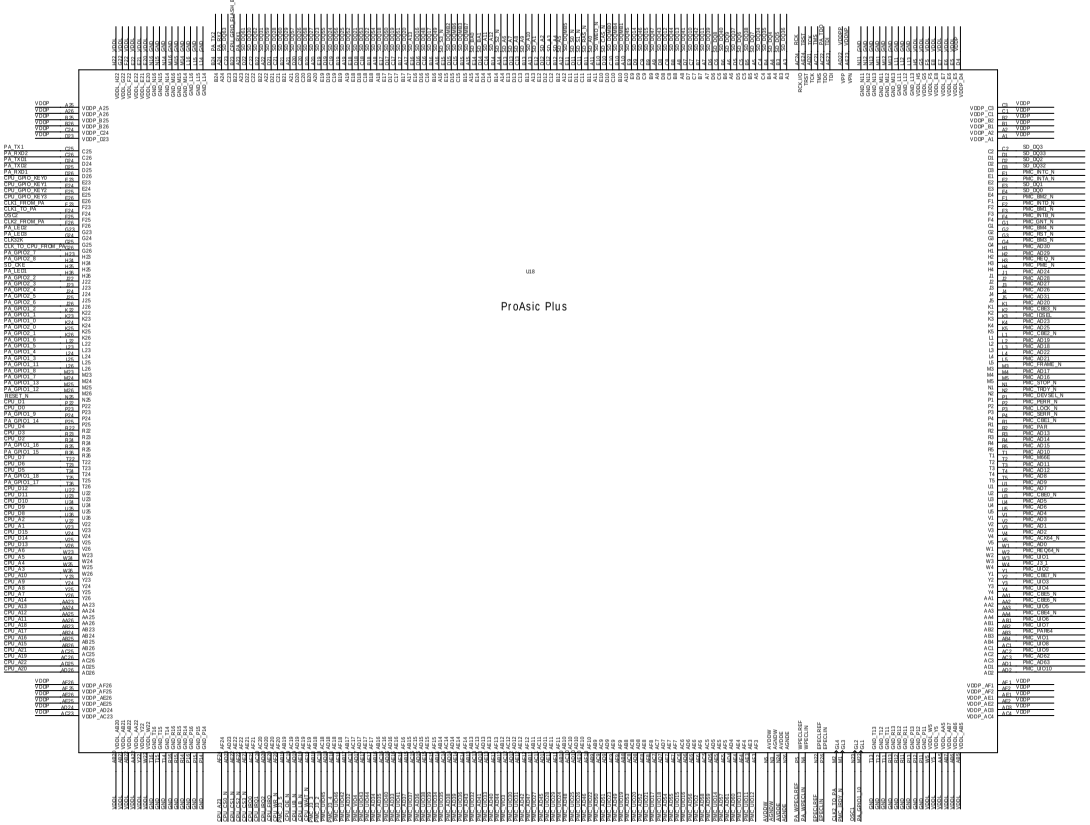
Size	Document Number	Rev
B	PADMA_SCH_01_0010	1.06

Date: Thursday, December 20, 2001 Sheet 4 of 10

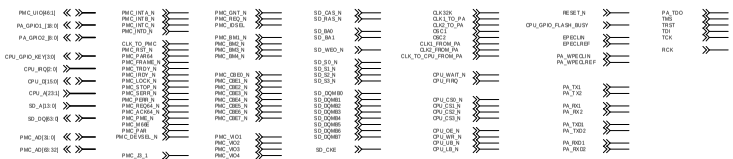


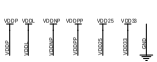
ProASIC Plus has two isolated power islands. One 2.5 Volt and one 3.3 Volt. Each of these power islands is connected with four 0R resistors and four 100nF caps on each side of ProASIC Plus to the main power supply.

These Filters should be placed close to the package pins and minimize inductance on capacitors, resistors and traces as much as possible.



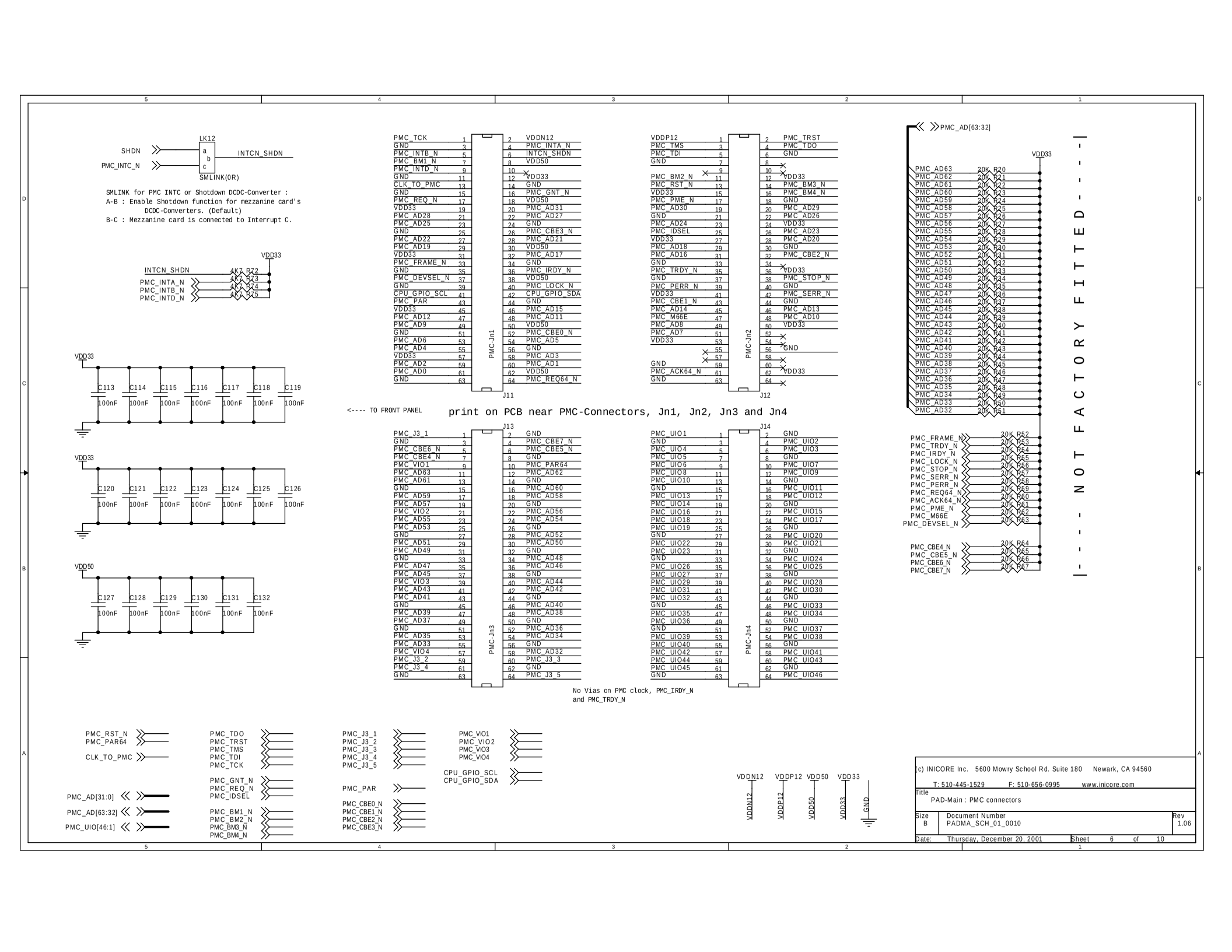
BGA456 Socket for Prototypes.
Place ProASIC close to PMC-Connectors.
No Vias on PMC _clock, PMC_IRDY_N and PMC_TRDY_N.

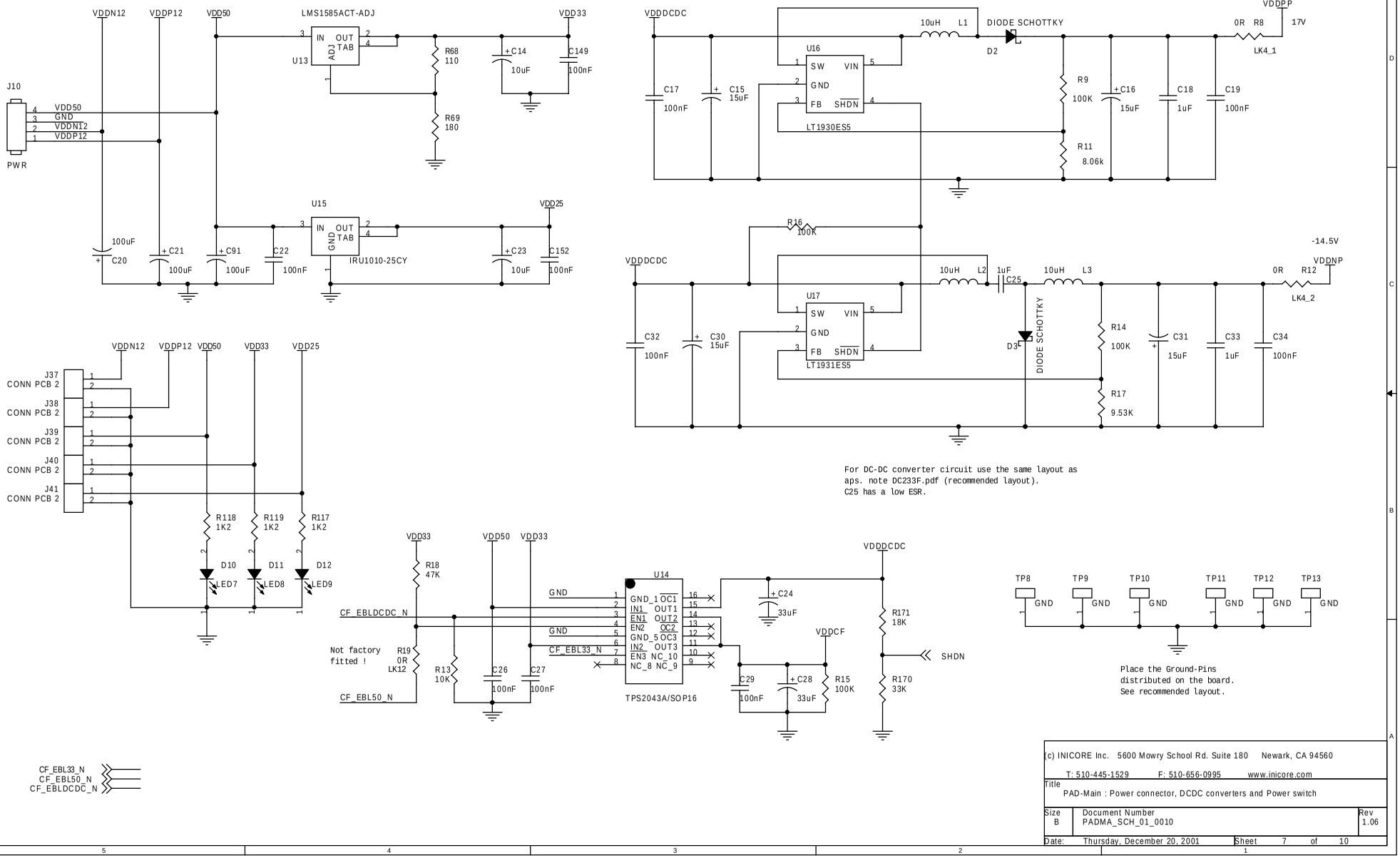




(c) INCORE Inc. 5600 Mowry School Rd. Suite 180 Newark, CA 94560
T: 510.445.3529 F: 510.666.0995 www.incore.com

Title:	PAD-Main : ProASIC Plus
A-B:	Document Number: PADMAI_SCH_1L_0010
D:	Rev: 1.06
Date:	Thursday, December 16, 2004 Sheet 5 of 10


LK12
SHDN
PMC_INTC_N
INTCN_SHDN
SMLINK(OR)
SMLINK for PMC INTC or Shotdown DCDC-Converter :
A-B : Enable Shotdown function for mezzanine card's DCDC-Converters. (Default)
B-C : Mezzanine card is connected to Interrupt C.
VDD33
INTCN_SHDN
PMC_INTA_N
PMC_INTB_N
PMC_INTD_N
4K7 R72
4K7 R73
4K7 R74
4K7 R75
PMC_TCK 1
GND 3
PMC_INTB_N 5
PMC_BM1_N 7
PMC_INTD_N 9
GND 11
CLK_TO_PMC 13
GND 15
PMC_REQ_N 17
VDD33 19
PMC_AD28 21
PMC_AD25 23
GND 25
PMC_AD22 27
PMC_AD19 29
VDD33 31
PMC_FRAME_N 33
GND 35
PMC_DEVSEL_N 37
GND 39
CPU_GPIO_SCL 41
PMC_PAR 43
VDD33 45
PMC_AD12 47
PMC_AD9 49
GND 51
PMC_AD6 53
PMC_AD4 55
VDD33 57
PMC_AD2 59
PMC_AD0 61
GND 63
2 VDDN12
4 PMC_INTA_N
6 INTCN_SHDN
8 VDD50
10
12 VDD33
14 GND
16 PMC_GNT_N
18 VDD50
20 PMC_AD31
22 PMC_AD27
24 GND
26 PMC_CBE3_N
28 PMC_AD21
30 VDD50
32 PMC_AD17
34 GND
36 PMC_IRDY_N
38 VDD50
40 PMC_LOCK_N
42 CPU_GPIO_SDA
44 GND
46 PMC_AD15
48 PMC_AD11
50 VDD50
52 PMC_CBE0_N
54 PMC_AD5
56 GND
58 PMC_AD3
60 PMC_AD1
62 VDD50
64 PMC_REQ64_N
PMC-Jn1
J11
VDDP12 1
PMC_TMS 3
PMC_TDI 5
GND 7
9
PMC_BM2_N 11
PMC_RST_N 13
VDD33 15
PMC_PME_N 17
PMC_AD30 19
GND 21
PMC_AD24 23
PMC_IDSEL 25
VDD33 27
PMC_AD18 29
PMC_AD16 31
GND 33
PMC_TRDY_N 35
GND 37
PMC_PERR_N 39
VDD33 41
PMC_CBE1_N 43
PMC_AD14 45
PMC_M66E 47
PMC_AD8 49
PMC_AD7 51
VDD33 53
55
57
GND 59
PMC_ACK64_N 61
GND 63
2 PMC_TRST
4 PMC_TDO
6 GND
8
10
12 VDD33
14 PMC_BM3_N
16 PMC_BM4_N
18 GND
20 PMC_AD29
22 PMC_AD26
24 VDD33
26 PMC_AD23
28 PMC_AD20
30 GND
32 PMC_CBE2_N
34
36 VDD33
38 PMC_STOP_N
40 GND
42 PMC_SERR_N
44 GND
46 PMC_AD13
48 PMC_AD10
50 VDD33
52
54
56 GND
58
60
62 VDD33
64
PMC-Jn2
J12
PMC_AD[63:32]
VDD33
PMC_AD63 20K R20
PMC_AD62 20K R21
PMC_AD61 20K R22
PMC_AD60 20K R23
PMC_AD59 20K R24
PMC_AD58 20K R25
PMC_AD57 20K R26
PMC_AD56 20K R27
PMC_AD55 20K R28
PMC_AD54 20K R29
PMC_AD53 20K R30
PMC_AD52 20K R31
PMC_AD51 20K R32
PMC_AD50 20K R33
PMC_AD49 20K R34
PMC_AD48 20K R35
PMC_AD47 20K R36
PMC_AD46 20K R37
PMC_AD45 20K R38
PMC_AD44 20K R39
PMC_AD43 20K R40
PMC_AD42 20K R41
PMC_AD41 20K R42
PMC_AD40 20K R43
PMC_AD39 20K R44
PMC_AD38 20K R45
PMC_AD37 20K R46
PMC_AD36 20K R47
PMC_AD35 20K R48
PMC_AD34 20K R49
PMC_AD33 20K R50
PMC_AD32 20K R51
VDD33
C113 100nF
C114 100nF
C115 100nF
C116 100nF
C117 100nF
C118 100nF
C119 100nF
<---- TO FRONT PANEL
print on PCB near PMC-Connectors, Jn1, Jn2, Jn3 and Jn4
PMC_J3_1 1
GND 3
PMC_CBE6_N 5
PMC_CBE4_N 7
PMC_VIO1 9
PMC_AD63 11
PMC_AD61 13
GND 15
PMC_AD59 17
PMC_AD57 19
PMC_VIO2 21
PMC_AD55 23
PMC_AD53 25
GND 27
PMC_AD51 29
PMC_AD49 31
GND 33
PMC_AD47 35
PMC_AD45 37
PMC_VIO3 39
PMC_AD43 41
PMC_AD41 43
GND 45
PMC_AD39 47
PMC_AD37 49
GND 51
PMC_AD35 53
PMC_AD33 55
PMC_VIO4 57
PMC_J3_2 59
PMC_J3_4 61
GND 63
2 GND
4 PMC_CBE7_N
6 PMC_CBE5_N
8 GND
10 PMC_PAR64
12 PMC_AD62
14 GND
16 PMC_AD60
18 PMC_AD58
20 GND
22 PMC_AD56
24 PMC_AD54
26 GND
28 PMC_AD52
30 PMC_AD50
32 GND
34 PMC_AD48
36 PMC_AD46
38 GND
40 PMC_AD44
42 PMC_AD42
44 GND
46 PMC_AD40
48 PMC_AD38
50 GND
52 PMC_AD36
54 PMC_AD34
56 GND
58 PMC_AD32
60 PMC_J3_3
62 GND
64 PMC_J3_5
PMC-Jn3
J13
PMC_UIO1 1
GND 3
PMC_UIO4 5
PMC_UIO5 7
PMC_UIO6 9
PMC_UIO8 11
PMC_UIO10 13
GND 15
PMC_UIO13 17
PMC_UIO14 19
PMC_UIO16 21
PMC_UIO18 23
PMC_UIO19 25
GND 27
PMC_UIO22 29
PMC_UIO23 31
GND 33
PMC_UIO26 35
PMC_UIO27 37
PMC_UIO29 39
PMC_UIO31 41
PMC_UIO32 43
GND 45
PMC_UIO35 47
PMC_UIO36 49
GND 51
PMC_UIO39 53
PMC_UIO40 55
PMC_UIO42 57
PMC_UIO44 59
PMC_UIO45 61
GND 63
2 GND
4 PMC_UIO2
6 PMC_UIO3
8 GND
10 PMC_UIO7
12 PMC_UIO9
14 GND
16 PMC_UIO11
18 PMC_UIO12
20 GND
22 PMC_UIO15
24 PMC_UIO17
26 GND
28 PMC_UIO20
30 PMC_UIO21
32 GND
34 PMC_UIO24
36 PMC_UIO25
38 GND
40 PMC_UIO28
42 PMC_UIO30
44 GND
46 PMC_UIO33
48 PMC_UIO34
50 GND
52 PMC_UIO37
54 PMC_UIO38
56 GND
58 PMC_UIO41
60 PMC_UIO43
62 GND
64 PMC_UIO46
PMC-Jn4
J14
PMC_FRAME_N 20K R52
PMC_TRDY_N 20K R53
PMC_IRDY_N 20K R54
PMC_LOCK_N 20K R55
PMC_STOP_N 20K R56
PMC_SERR_N 20K R57
PMC_PERR_N 20K R58
PMC_REQ64_N 20K R59
PMC_ACK64_N 20K R60
PMC_PME_N 20K R61
PMC_M66E 20K R62
PMC_DEVSEL_N 20K R63
PMC_CBE4_N 20K R64
PMC_CBE5_N 20K R65
PMC_CBE6_N 20K R66
PMC_CBE7_N 20K R67
- - - NOT FACTORY FITTED - - -
VDD50
C120 100nF
C121 100nF
C122 100nF
C123 100nF
C124 100nF
C125 100nF
C126 100nF
VDD50
C127 100nF
C128 100nF
C129 100nF
C130 100nF
C131 100nF
C132 100nF
No Vias on PMC clock, PMC_IRDY_N and PMC_TRDY_N
PMC_RST_N
PMC_PAR64
CLK_TO_PMC
PMC_AD[31:0]
PMC_AD[63:32]
PMC_UIO[46:1]
PMC_TDO
PMC_TRST
PMC_TMS
PMC_TDI
PMC_TCK
PMC_GNT_N
PMC_REQ_N
PMC_IDSEL
PMC_BM1_N
PMC_BM2_N
PMC_BM3_N
PMC_BM4_N
PMC_J3_1
PMC_J3_2
PMC_J3_3
PMC_J3_4
PMC_J3_5
PMC_PAR
PMC_CBE0_N
PMC_CBE1_N
PMC_CBE2_N
PMC_CBE3_N
PMC_VIO1
PMC_VIO2
PMC_VIO3
PMC_VIO4
CPU_GPIO_SCL
CPU_GPIO_SDA
VDDN12
VDDP12
VDD50
VDD33
GND
(c) INICORE Inc. 5600 Mowry School Rd. Suite 180 Newark, CA 94560
T: 510-445-1529 F: 510-656-0995 www.inicore.com
Title
PAD-Main : PMC connectors
Size B
Document Number PADMA_SCH_01_0010
Rev 1.06
Date: Thursday, December 20, 2001
Sheet 6 of 10



For DC-DC converter circuit use the same layout as aps. note DC233F.pdf (recommended layout).
C25 has a low ESR.

(c) INICORE Inc. 5600 Mowry School Rd. Suite 180 Newark, CA 94560
T: 510-445-1529 F: 510-656-0995 www.inicore.com

Title: PAD-Main : Power connector, DCDC converters and Power switch

Size	Document Number	Rev
B	PADMA_SCH_01_0010	1.06

Date: Thursday, December 20, 2001 Sheet 7 of 10

When fitting 128Kx8 SRAM parts, align pin 1 of the SRAM to where
pin 2 would be if a 512K part were fitted.
Default is 128Kx8 SRAM.

J19

GND	1	85	GND
SD_DQ0	2	86	SD_DQ32
SD_DQ1	3	87	SD_DQ33
SD_DQ2	4	88	SD_DQ34
SD_DQ3	5	89	SD_DQ35
VDD33	6	90	VDD33
SD_DQ4	7	91	SD_DQ36
SD_DQ5	8	92	SD_DQ37
SD_DQ6	9	93	SD_DQ38
SD_DQ7	10	94	SD_DQ39
SD_DQ8	11	95	SD_DQ40
GND	12	96	GND
SD_DQ9	13	97	SD_DQ41
SD_DQ10	14	98	SD_DQ42
SD_DQ11	15	99	SD_DQ43
SD_DQ12	16	100	SD_DQ44
SD_DQ13	17	101	SD_DQ45
VDD33	18	102	VDD33
SD_DQ14	19	103	SD_DQ46
SD_DQ15	20	104	SD_DQ47
	21	105	
	22	106	
GND	23	107	GND
	24	108	
	25	109	
VDD33	26	110	VDD33
SD_WEO_N	27	111	SD_CAS_N
SD_DQMB0	28	112	SD_DQMB4
SD_DQMB1	29	113	SD_DQMB5
SD_S0_N	30	114	SD_S1_N
	31	115	SD_RAS_N
GND	32	116	GND
SD_A0	33	117	SD_A1
SD_A2	34	118	SD_A3
SD_A4	35	119	SD_A5
SD_A6	36	120	SD_A7
SD_A8	37	121	SD_A9
SD_A10	38	122	SD_BA0
SD_BA1	39	123	SD_A11
VDD33	40	124	VDD33
VDD33	41	125	CLK_SD1
CLK_SD1	42	126	SD_A12
GND	43	127	GND
	44	128	SD_CKE
SD_S2_N	45	129	SD_S3_N
SD_DQMB2	46	130	SD_DQMB6
SD_DQMB3	47	131	SD_DQMB7
	48	132	SD_A13
VDD33	49	133	VDD33
	50	134	
	51	135	
	52	136	
	53	137	
GND	54	138	GND
SD_DQ16	55	139	SD_DQ48
SD_DQ17	56	140	SD_DQ49
SD_DQ18	57	141	SD_DQ50
SD_DQ19	58	142	SD_DQ51
VDD33	59	143	VDD33
SD_DQ20	60	144	SD_DQ52
	61	145	
	62	146	
SD_CKE	63	147	O_REGE
GND	64	148	GND
SD_DQ21	65	149	SD_DQ53
SD_DQ22	66	150	SD_DQ54
SD_DQ23	67	151	SD_DQ55
GND	68	152	GND
SD_DQ24	69	153	SD_DQ56
SD_DQ25	70	154	SD_DQ57
SD_DQ26	71	155	SD_DQ58
SD_DQ27	72	156	SD_DQ59
VDD33	73	157	VDD33
SD_DQ28	74	158	SD_DQ60
SD_DQ29	75	159	SD_DQ61
SD_DQ30	76	160	SD_DQ62
SD_DQ31	77	161	SD_DQ63
GND	78	162	GND
CLK_SD2	79	163	CLK_SD2
	80	164	
	81	165	GND
SD_SDA	82	166	GND
SD_SCL	83	167	GND
VDD33	84	168	VDD33

SDRAM-SOCKET

BANK 0 , LOW BYTE

CPU_A18	1	36	
CPU_A1	2	35	CPU_A17
CPU_A2	3	34	CPU_A16
CPU_A3	4	33	CPU_A15
CPU_A4	5	32	CPU_A14
BANK_SEL0	6	31	CPU_OE_N
CPU_D0	7	30	CPU_D7
CPU_D1	8	29	CPU_D6
VDD33	9	28	GND
GND	10	27	VDD33
CPU_D2	11	26	CPU_D5
CPU_D3	12	25	CPU_D4
CPU_WR_N	13	24	CPU_A13
CPU_A5	14	23	CPU_A12
CPU_A6	15	22	CPU_A11
CPU_A7	16	21	CPU_A10
CPU_A8	17	20	CPU_A9
CPU_A19	18	19	

U8 — A17 NC_36; A0 A16; A1 A15; A2 A14; A3 A13; CE OE; D0 D7; D1 D6; VCC GND_28; GND_10 VCC_27; D2 D5; D3 D4; WE A12; A4 A11; A5 A10; A6 A9; A7 A8; A18 NC_19

512Kx8_SRAM/SOJ36

BANK 1 , LOW BYTE

CPU_A18	1	36	
CPU_A1	2	35	CPU_A17
CPU_A2	3	34	CPU_A16
CPU_A3	4	33	CPU_A15
CPU_A4	5	32	CPU_A14
BANK_SEL1	6	31	CPU_OE_N
CPU_D0	7	30	CPU_D7
CPU_D1	8	29	CPU_D6
VDD33	9	28	GND
GND	10	27	VDD33
CPU_D2	11	26	CPU_D5
CPU_D3	12	25	CPU_D4
CPU_WR_N	13	24	CPU_A13
CPU_A5	14	23	CPU_A12
CPU_A6	15	22	CPU_A11
CPU_A7	16	21	CPU_A10
CPU_A8	17	20	CPU_A9
CPU_A19	18	19	

U9 — A17 NC_36; A0 A16; A1 A15; A2 A14; A3 A13; CE OE; D0 D7; D1 D6; VCC GND_28; GND_10 VCC_27; D2 D5; D3 D4; WE A12; A4 A11; A5 A10; A6 A9; A7 A8; A18 NC_19

512Kx8_SRAM/SOJ36

BANK 0 , HIGH BYTE

CPU_A18	1	36	
CPU_A1	2	35	CPU_A17
CPU_A2	3	34	CPU_A16
CPU_A3	4	33	CPU_A15
CPU_A4	5	32	CPU_A14
BANK_SEL0	6	31	CPU_OE_N
CPU_D8	7	30	CPU_D15
CPU_D9	8	29	CPU_D14
VDD33	9	28	GND
GND	10	27	VDD33
CPU_D10	11	26	CPU_D13
CPU_D11	12	25	CPU_D12
CPU_UB_N	13	24	CPU_A13
CPU_A5	14	23	CPU_A12
CPU_A6	15	22	CPU_A11
CPU_A7	16	21	CPU_A10
CPU_A8	17	20	CPU_A9
CPU_A19	18	19	

U11 — A17 NC_36; A0 A16; A1 A15; A2 A14; A3 A13; CE OE; D0 D7; D1 D6; VCC GND_28; GND_10 VCC_27; D2 D5; D3 D4; WE A12; A4 A11; A5 A10; A6 A9; A7 A8; A18 NC_19

512Kx8_SRAM/SOJ36

BANK 1 , HIGH BYTE

CPU_A18	1	36	
CPU_A1	2	35	CPU_A17
CPU_A2	3	34	CPU_A16
CPU_A3	4	33	CPU_A15
CPU_A4	5	32	CPU_A14
BANK_SEL1	6	31	CPU_OE_N
CPU_D8	7	30	CPU_D15
CPU_D9	8	29	CPU_D14
VDD33	9	28	GND
GND	10	27	VDD33
CPU_D10	11	26	CPU_D13
CPU_D11	12	25	CPU_D12
CPU_UB_N	13	24	CPU_A13
CPU_A5	14	23	CPU_A12
CPU_A6	15	22	CPU_A11
CPU_A7	16	21	CPU_A10
CPU_A8	17	20	CPU_A9
CPU_A19	18	19	

U12 — A17 NC_36; A0 A16; A1 A15; A2 A14; A3 A13; CE OE; D0 D7; D1 D6; VCC GND_28; GND_10 VCC_27; D2 D5; D3 D4; WE A12; A4 A11; A5 A10; A6 A9; A7 A8; A18 NC_19

512Kx8_SRAM/SOJ36

VDD33

R7 2K5

U10 — FLASH/TSOP_I_56

CPU_A22	1	56	NC56
	2	55	CPU_WR_N
CPU_A21	3	54	CPU_OE_N
CPU_A20	4	53	CPU_GPIO_FLASH_BUSY
CPU_A19	5	52	CPU_D15
CPU_A18	6	51	CPU_D7
CPU_A17	7	50	CPU_D14
CPU_A16	8	49	CPU_D6
VDD33	9	48	GND
CPU_A15	10	47	CPU_D13
CPU_A14	11	46	CPU_D5
CPU_A13	12	45	CPU_D12
CPU_A12	13	44	CPU_D4
CPU_CS0_N	14	43	VDD33
VDD33	15	42	GND
RESET_N	16	41	CPU_D11
CPU_A11	17	40	CPU_D3
CPU_A10	18	39	CPU_D10
CPU_A9	19	38	CPU_D2
CPU_A8	20	37	VDD33
GND	21	36	CPU_D9
CPU_A7	22	35	CPU_D1
CPU_A6	23	34	CPU_D8
CPU_A5	24	33	CPU_D0
CPU_A4	25	32	CPU_LB_N
CPU_A3	26	31	FLASH_BM_N
CPU_A2	27	30	CPU_A23
CPU_A1	28	29	

NC56 WE OE STS DQ15 DQ7 DQ14 DQ6 Vss DQ13 DQ5 DQ12 DQ4 VCCQ Vss_42 DQ11 DQ3 DQ10 DQ2 Vcc_37 DQ9 DQ1 DQ8 DQ0 A0 BYTE A23 CE2

TSOP56 Socket

Print on PCB the position of a,b and c

VDD33 — LK3 (a b c) — FLASH_BM_N — SMLINK(0R)

SMLINK for Flash-mode selection
fitted with 400K resistor:
A-B : Flash in 16-bit mode (Default).
B-C : Flash in 8-bit mode.

VDD33

1 capacitor for each power pin of Flash and SRAM

C9 C10 C11 C12 C13 C133 C134 C135 C136 C137 C138 C139 C140 — 100nF

VDD33 — Place C141 near SRAMs — + C141 10uF

VDD33

C142 C143 C144 C145 C146 C147 C148 C150 — 100nF

Place this Caps near SDRAM Connector (J19).

SMLINKs for SRAM configuration :
Config(0) / Config(1)
A-B / A-B : 128Kx8,(Default)
A-B / B-C : 256Kx8,
B-C / A-B : 512x8.

VDD33 — LK7_1 (a b c) — SRAM_CONFIG0 — SMLINK(0R)
LK7_2 (a b c) — SRAM_CONFIG1 — SMLINK(0R)

Print on PCB the position of a,b and c

VDD33 — LK5 (a b c) — J_REGE — SMLINK(0R)

SMLINK for SDRAM mode selection :
A-B : Registered DIMM mode.
B-C : Buffered DIMM mode.
Not factory fitted

Print on PCB the position of a,b and c

SD_A[13:0]
SD_DQ[63:0]
CPU_D[15:0]
CPU_A[23:1]
SD_CAS_N
SD_RAS_N
SD_BA0
SD_BA1
SD_WEO_N
RESET_N

SD_S0_N
SD_S1_N
SD_S2_N
SD_S3_N
SD_DQMB0
SD_DQMB1
SD_DQMB2
SD_DQMB3
SD_DQMB4
SD_DQMB5
SD_DQMB6
SD_DQMB7
CPU_GPIO_FLASH_BUSY

CPU_GPIO_SDA
CPU_GPIO_SCL
SD_SDA
SD_SCL
CPU_CS0_N
CPU_OE_N
CPU_WR_N
CPU_UB_N
CPU_IB_N
BANK_SEL0
BANK_SEL1
CLK_SD2
CLK_SD1
SD_CKE

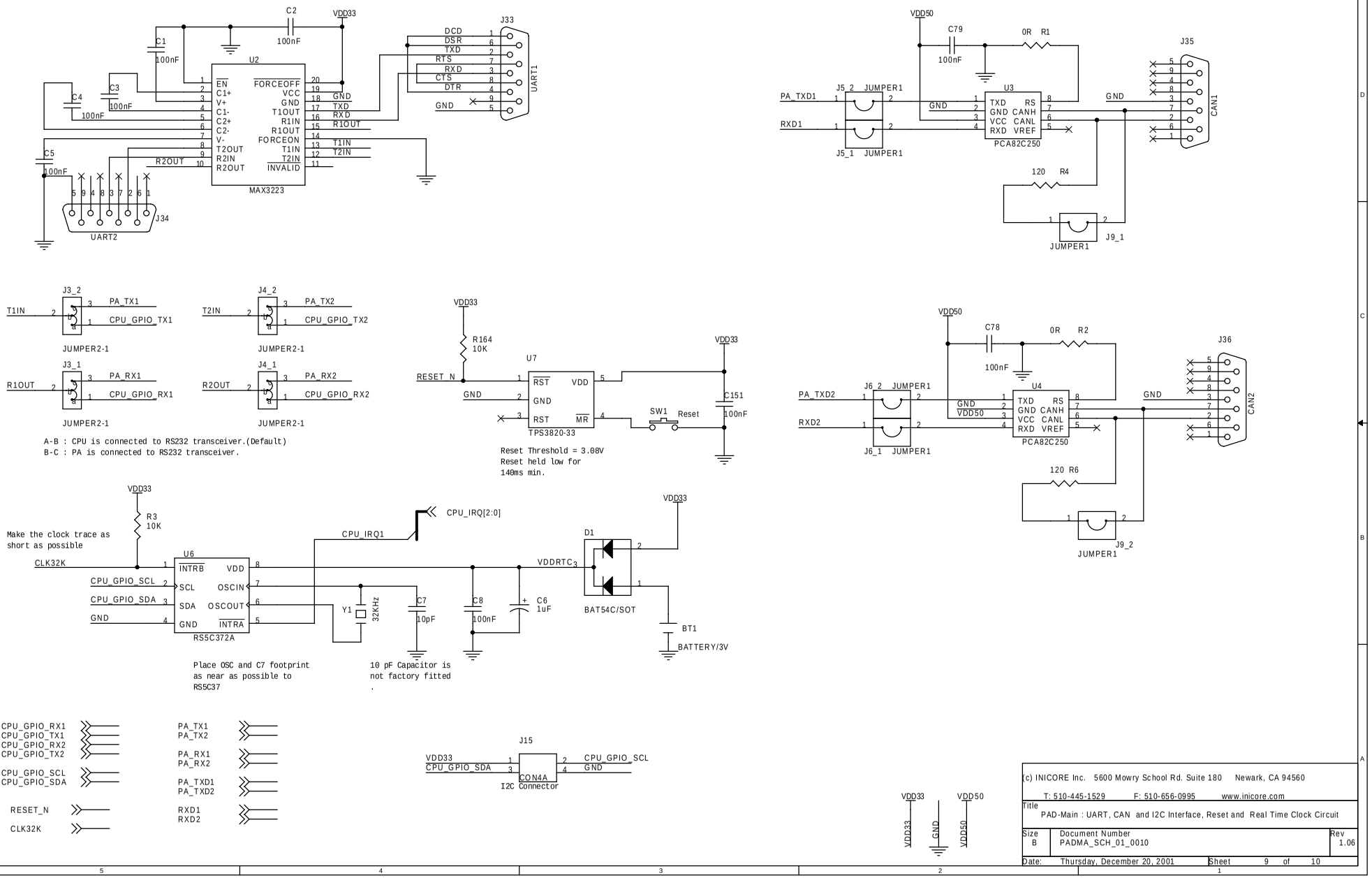
VDD33 VDD33 GND

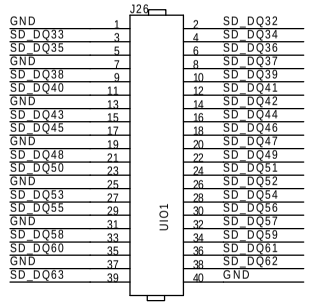
 5600 Mowry School Rd. Suite 180 Newark, CA 94560
T: 510-445-1529 F: 510-656-0995 www.inicore.com

Title: PAD-Main : SRAM, SDRAM and Flash-Memory

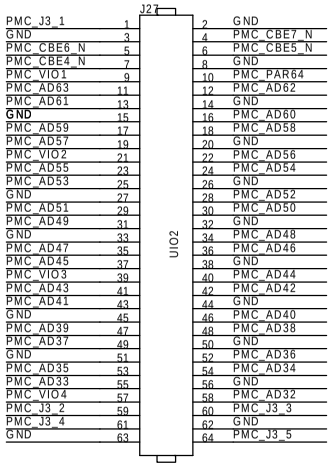
Size	Document Number	Rev
B	PADMA_SCH_01_0010	1.06

Date: Thursday, December 20, 2001 Sheet 8 of 10

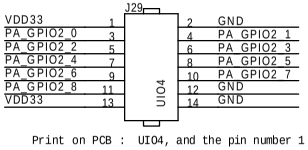

C2
VDD33
100nF
C1
100nF
U2
C3
100nF
C4
100nF
C5
100nF
R2OUT
EN
C1+
V+
C1-
C2+
C2-
V-
T2OUT
R2IN
R2OUT
FORCEOFF
VCC
GND
T1OUT
R1IN
R1OUT
FORCEON
T1IN
T2IN
INVALID
MAX3223
J33
DCD
DSR
TXD
RTS
RXD
CTS
DTR
GND
UART1
5 9 4 8 3 7 2 6 1
J34
UART2
VDD50
C79
100nF
0R R1
J5_2 JUMPER1
PA_TXD1
RXD1
J5_1 JUMPER1
U3
TXD RS
GND CANH
VCC CANL
RXD VREF
PCA82C250
GND
120 R4
J9_1
JUMPER1
J35
CAN1
J3_2
T1IN
PA_TX1
CPU_GPIO_TX1
JUMPER2-1
J4_2
T2IN
PA_TX2
CPU_GPIO_TX2
JUMPER2-1
J3_1
R1OUT
PA_RX1
CPU_GPIO_RX1
JUMPER2-1
J4_1
R2OUT
PA_RX2
CPU_GPIO_RX2
JUMPER2-1
A-B : CPU is connected to RS232 transceiver.(Default)
B-C : PA is connected to RS232 transceiver.
VDD33
R164
10K
RESET_N
U7
RST
GND
RST
VDD
MR
TPS3820-33
SW1
Reset
C151
100nF
Reset Threshold = 3.08V
Reset held low for
140ms min.
VDD50
C78
100nF
0R R2
J6_2 JUMPER1
PA_TXD2
RXD2
J6_1 JUMPER1
U4
TXD RS
GND CANH
VCC CANL
RXD VREF
PCA82C250
GND
VDD50
120 R6
J9_2
JUMPER1
J36
CAN2
VDD33
R3
10K
Make the clock trace as
short as possible
CLK32K
U6
CPU_GPIO_SCL
CPU_GPIO_SDA
GND
INTRB
SCL
SDA
GND
VDD
OSCIN
OSCOUT
INTRA
RS5C372A
CPU_IRQ1
CPU_IRQ[2:0]
Y1
32KHz
C7
10pF
C8
100nF
C6
1uF
VDDRTC
D1
BAT54C/SOT
VDD33
BT1
BATTERY/3V
Place OSC and C7 footprint
as near as possible to
RS5C37
10 pF Capacitor is
not factory fitted
.
CPU_GPIO_RX1
CPU_GPIO_TX1
CPU_GPIO_RX2
CPU_GPIO_TX2
CPU_GPIO_SCL
CPU_GPIO_SDA
RESET_N
CLK32K
PA_TX1
PA_TX2
PA_RX1
PA_RX2
PA_TXD1
PA_TXD2
RXD1
RXD2
J15
VDD33
CPU_GPIO_SDA
CON4A
I2C Connector
CPU_GPIO_SCL
GND
VDD33
GND
VDD50
(c) INICORE Inc. 5600 Mowry School Rd. Suite 180 Newark, CA 94560
T: 510-445-1529 F: 510-656-0995 www.inicore.com
Title
PAD-Main : UART, CAN and I2C Interface, Reset and Real Time Clock Circuit
Size B
Document Number
PADMA_SCH_01_0010
Rev 1.06
Date: Thursday, December 20, 2001
Sheet 9 of 10



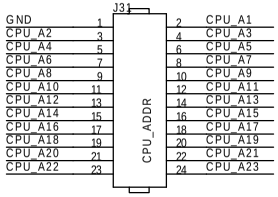
Print on PCB : UIO1 and the pin numbers 1, 11, 21 and 31.



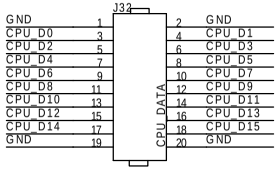
Print on PCB : UIO2 and the pin numbers 1, 11, 21, 31, 41, 51 and 61.



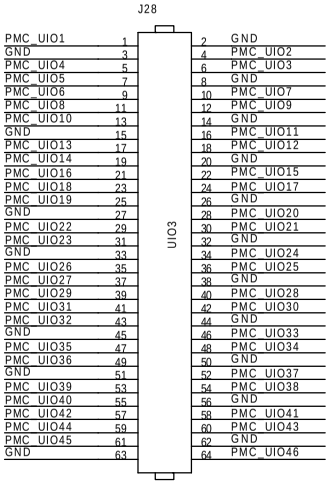
Print on PCB : UIO4, and the pin number 1



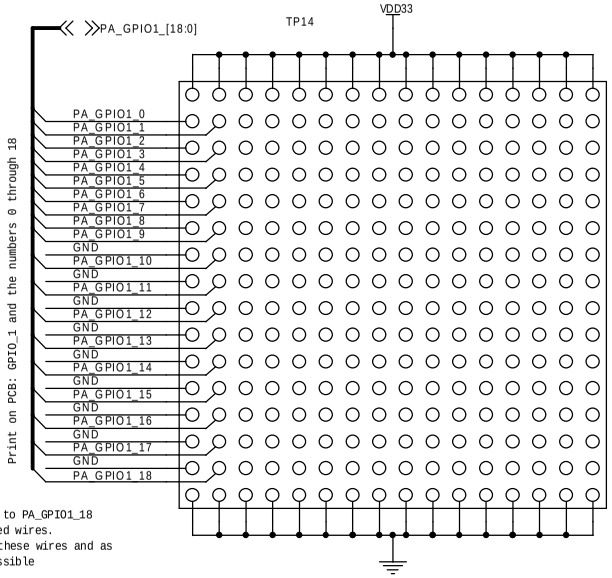
Print on PCB all signal names, without prefix "CPU_".



Print on PCB all signal names, without prefix "CPU_".

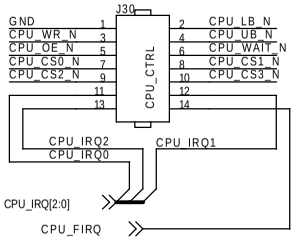


Print on PCB : UIO3 and the pin numbers 1, 11, 21, 31, 41, 51 and 61.



PA_GPIO1_10 to PA_GPIO1_18 are Highspeed wires. No vias on these wires and as short as possible

SOLDER POINTS/16X16



Print on PCB all signal names, without prefix "CPU_".

PA_GPIO1_[18:0]
PA_GPIO2_[8:0]
PMC_UIO[46:1]
PMC_AD[31:0]
PMC_AD[63:32]

CPU_A[23:1]
CPU_D[15:0]
CPU_IRQ[2:0]
SD_DQ[63:0]

CPU_WAIT_N

CPU_OE_N
CPU_WR_N
CPU_UB_N
CPU_LB_N

CPU_CS0_N
CPU_CS1_N
CPU_CS2_N
CPU_CS3_N

PMC_CBE4_N
PMC_CBE5_N
PMC_CBE6_N
PMC_CBE7_N

PMC_PAR64

PMC_VIO1
PMC_VIO2
PMC_VIO3
PMC_VIO4

PMC_J3_1
PMC_J3_2
PMC_J3_3
PMC_J3_4
PMC_J3_5

VDD33
VDD33
GND

(c) INICORE Inc. 5600 Mowry School Rd. Suite 180 Newark, CA 94560
T: 510-445-1529 F: 510-656-0995 www.inicore.com

Title
PAD-Main : USERI/Os and Prototyping Area

Size	Document Number	Rev
B	PADMA_SCH_01_0010	1.06

Date: Thursday, December 20, 2001 Sheet 10 of 10