Radiation Hardened Very Low Power ASICs for Satellite Command Control and Data Handling (C&DH) and Sensor Integration

Sasan H Ardalan, Donald Elkins
Micro-RDC
2100 Airpark Rd. SE, Suite 120; Albuquerque, NM 87106 (719)531-0805
sasan.ardalan@micro-rdc.com

ABSTRACT

To enable missions in harsh radiation environments and for long term missions, we have developed a family of low power radiation hardened ASICs on the IBM 90nm CMOS 9LP Process. The chips include System on a Chip (SoC) designs ranging from the 32 bit OpenRISC, the 8 bit 8051XC and an 8-bit RISC architecture. For two SoCs (8 bit and 32 bit) we have incorporated two OpenCores 16550 compatible UARTs in addition to six 32 bit OpenCores Pulse Width Modulation and Timers (PTCs). The PTC’s can be used for Servo Motor Control, Timing events, and clock sources. All SoC’s support a dedicated SPI Slave for firmware download as well as a SPI Master for firmware boot up from external SPI Non Volatile Memory (NVM). A SPI Slave to Parallel Non-Volatile Memory (NVM) ASIC has been developed that converts Parallel NVM to a SPI Slave interface. This allows for both boot up from external Parallel NVM and programming external Parallel NVM up to 16Mbytes. The SoC’s all run at 50MHz clock rates. All sequential logic is implemented with Temporal Latch® technology which makes the designs SEU immune up to 60 LET. The chips are TID hard to 1 MRad.

INTRODUCTION

The Structured ASIC program at the IBM 90nm CMOS 9LP Process technology node funded by Air Force Research Laboratory (AFRL) has as an objective to provide RTL to ASIC tapeout in a rapid development cycle. The ASIC program supports Radiation Hardened by Design (RHBD) techniques at the 90nm 9LP Technology Node to achieve TID hardness of greater than 1 MRAD. In addition, the sequential logic supports the Temporal Latch® technology in which SEU immunity is achieved up to 60 LET. Previously, an 8051XC based SoC ASIC was taped out and fully verified in Silicon [1] as well as a very low power small 8 Bit RISC based SoC (Mini-PnP) [2] which was also fully verified in Silicon. Both chips use the Temporal Latch® [3] for SEU immunity. Through a joint project between Micro-RDC and GE Global Research funded by AFRL [4], BGA packages were developed for the ASIC chips. See Figure 1. The BGA packaging uses GE Global Research’s HDI process to achieve a very small package for each Die. The 8051XC SoC (7x7mm Die) uses a 16x16 BGA with 0.8 mm pitch. The Mini-PnP (3x3mm Die) uses a 10x10 BGA with 0.8mm pitch. All signals are brought out to the BGA pins with no compromises in functionality. The HDI process with the BGA’s can withstand up to a 100,000 g’s acceleration.

This paper will describe new ASICs that were designed and taped out on the IBM 90nm CMOS 9LP process Structured ASIC program. A revised version of the 8051XC SoC was also taped out with enhanced capability but maintaining the same pinout. With the new ASIC chips, we have a complete family of advanced high performance, very low power and with the HDI process ultra small BGA’s to address the requirements for Radiation Hardened CubeSats. In the next sections we will describe the ASICs and also an example of using the 32bit OpenRISC SoC ASIC and 8051XC SoC ASIC in a CubeSat design where the 32bit OpenRISC SoC is the C&DH. Furthermore, using the Parallel Non Volatile Memory to SPI Slave Interface Conversion ASIC (Support Chip) that we designed and taped out, we can support TID Hard up to 1 MRAD SPI Non Volatile Memory (up to 16Mbytes).

The new ASIC chips support a large number of cores and peripheries for chip to chip communications and sensor interfacing. The Super Mini-PnP ASIC is based on the original Mini-PnP [2] but supports twice the code space and also multiple cores. Table 1 shows the
cores for the 32 bit OpenRISC SoC and Super Mini-PnP ASICs.

8051XC SOC

The 8051XC Based SoC ASIC can be used in a stand-alone mode providing control and sensor interfaces in a small satellite. It can also be integrated with the 32 bit OpenRISC SoC ASIC to provide the more powerful 32 bit SoC with additional capabilities. The 8051XC CPU is an enhanced version of the popular 8051 microcontroller. The ASIC clock speed is 50 MHz. The RTL for the CPU and some associated peripherals were developed by Evatronix now part of Cadence. We designed the 8051XC SoC ASIC to include SpaceWire/USB and a 16550 compatible UART from Open Cores. The ASIC is fully described in [1]. The SoC has been designed so that CPU instructions run from on-chip fast Block SRAM (64Kx8) at 50MHz. We have the capability to download firmware into the on-chip instruction RAM though a SPI Slave interface. We also support a boot up Core which is a SPI Master that can copy the instructions from an external SPI EEPROM. Both methods can be used to scrub the Block SRAM. The Block SRAM is TID Hard and supports EDAC with interleaving. The chip also supports 128Kx8 external ROM.

The 8051XC SoC supports a Deep Sleep mode and an Idle mode for power management. In Idle mode all peripherals are turned off (clocks disabled). In this mode, however, an interrupt (for example on the I2C core) will take the processor out of Idle mode. In deep sleep mode the CPU clock is also disabled. Only a reset or an interrupt on interrupt_0 will wake up the CPU and take it out of deep sleep mode. Figure 2 shows the power consumption in Deep Sleep and Idle mode of the actual Silicon for the 8051XC SoC ASIC. The Real Time Clock (RTC) described below can also take the CPU out of deep sleep mode.

Table 1 Cores and Peripherals for the 32 bit OpenRISC SoC and Super Mini-PnP SoC ASICs

<table>
<thead>
<tr>
<th>Chip</th>
<th>UART_1</th>
<th>UART_2</th>
<th>ID_C</th>
<th>ID_D</th>
<th>ID_E</th>
<th>ID_F</th>
<th>SPI Master</th>
<th>SPI Download</th>
</tr>
</thead>
<tbody>
<tr>
<td>32 bit Open RISC</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Super Mini PnP</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chip</th>
<th>PTC_0</th>
<th>PTC_1</th>
<th>PTC_2</th>
<th>PTC_3</th>
<th>PTC_4</th>
<th>PTC_5</th>
<th>Watch Dog Timer</th>
<th>EEPROM Bootup</th>
</tr>
</thead>
<tbody>
<tr>
<td>32 bit Open RISC</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Super Mini PnP</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
</tbody>
</table>

Figure 1 BGA Packaging with 0.8mm Pitch for the 7x7mm and 3x3mm ASIC Die [4]
A key feature of the 8051XC is the support for a Real Time Clock (RTC) Unit.

From Evatronics Design Guide on Real Time Clock for 8051XC:

The Real Time Clock realizes a real time count with a resolution of 1/256th second. It allows the user to read seconds, minutes, hours, day of the week and the date. The date is represented by a 16-bit number, which value is interpreted by the user software. The RTC enables a count of 179 years. The RTC features an alarm function which may be used to generate interrupts at a given time during a day or periodically. These interrupts may be used to resume operation from the IDLE/DEEP SLEEP mode at a given time.

We have fully verified the RTC in Silicon. We have also verified that it can take the CPU out of Deep Sleep mode upon wake up.

32 BIT OPENRISC SOC ASIC

The Radiation Hardened 90nm SASIC System on a Chip based on the OpenRISC 32 bit RISC CPU with multiple cores to support SpaceWire/ I2C/ SPI/ UARTS/ PWM/Timers and GPIO was designed and taped out. Figure 3 shows the bondout of the chip to the LGA-484 package. The Die is 7x7mm. The chip includes 256Kx8 Block SRAM with EDAC and Interleaving.

For details on the history of the OpenCores 32 bit RISC CPU see [5]. Also for a review of the architecture and future developments see [5]. We have designed the 32 bit OpenRISC SoC ASIC based on the OpenCores OR1200 CPU. We developed a Wishbone Bus (switched) to support all the cores in Table 1. We also developed test benches for RTL verification including Post Place and Route gatelevel verification.

The 32 bit OpenRISC SoC has been designed such that the on-chip Instruction RAM can be organized as either 64Kx32 for executing code or 256Kx8 for bootup from
external SPI EEPROM (NVM) or SPI Firmware Download. In this fashion, the 32bit OpenRISC SoC ASIC is compatible with all tools and hardware for firmware and bootup of the 8-bit SoC ASICs.

All cores of the 32 bit OpenRISC ASIC and the CPU itself have been verified in Silicon. We are currently testing the SpaceWire core.

Figure 5 shows the 32 bit OpenRISC SoC ASIC Core (Vdd 1.2V) current consumption versus clock frequency. To achieve very low power we can scale the clock frequency down. This is a very low power 32 bit SoC ASIC. The results are also tabulated in Table 2.

8-BIT RISC SOC (SUPER MINI-PNP)

The Super Mini-PnP is a major enhancement of the 3x3mm Mini-PnP ASIC reported in [2]. The code space has been expanded to 8Kx14. The supported cores and peripherals are tabulated in Table 1. The Super Mini-PnP die is 5x5mm. It shares the very low power consumption of the Mini-PnP design. It operates at a 50 MHz clock speed. It is currently being tested.

PARALLEL NVM TO SPI CONVERSION SUPPORT CHIP ASIC

The Parallel EEPROM to SPI Conversion chip is critical for the deployment of Radiation Hardened System on a Chips. It provides for the interface of 1 MRAD Parallel Non Volatile Memories to the SoC chips using the SPI interface. All SoC chips that we have designed support bootup and scrubbing using a SPI interface. The chip also has a full radiation hardened Mux equivalent to the 74HC157 and also tristate buffers for I2C bus interface eliminating the need for external NPN or MOSFET transistors.

Figure 6 shows the Bondout of the 3x3mm Support Chip ASIC die to the LGA-484 package for Post Silicon Verification. The chip has been designed so that the 10x10 BGA with 0.8mm pitch as shown in Figure 1 can be used to package the part. In this case, CAD work and layout of the HDI design is unchanged. Figure 7 shows the toplevel of the Support Chip. In Figure 8 we show how the Support Chip ASIC can be used to convert a Parallel EEPROM to a SPI Slave interface. This allows all the SoC ASICs to bootup from and read data from an external Parallel EEPROM. The Support Chip can also be used as a SPI port expander (where the address lines, control lines and bidirectional 8 bit bus are used as GPIO or digital outputs).

Figure 5 32 bit OpenRISC SoC ASIC Core Current Consumption versus Clock Frequency

Table 2 32 bit OpenRISC SoC ASIC Core Current Consumption versus Clock Frequency

<table>
<thead>
<tr>
<th>Frequency(MHz)</th>
<th>Core Current(mA)</th>
</tr>
</thead>
<tbody>
<tr>
<td>75</td>
<td>54</td>
</tr>
<tr>
<td>50</td>
<td>40</td>
</tr>
<tr>
<td>25</td>
<td>21</td>
</tr>
<tr>
<td>20</td>
<td>18</td>
</tr>
<tr>
<td>15</td>
<td>14</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>2.5</td>
<td>3.5</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>
The objective of these tests were to demonstrate:

1- Program the Parallel EEPROM with the Support Chip ASIC SPI Slave Interface.

2- Read back the burned in data using the Support Chip ASIC SPI Slave Interface.

Figure 9 shows the test setup. In the Test Setup, the Support Chip interfaces to an ATMEL Parallel EEPROM as shown in Figure 10. A microcontroller with a SPI Master (an SoC) is connected to the Support Chip SPI Slave.

Summary of Silicon Test Results:

- Rad Hard Support Chip ASIC designed so that System on a Chip (SoC) ASICs can Bootup from Rad Hard Parallel EEPROMS or Read Data from Parallel EEPROM.

- The chip provides a Slave SPI Interface to Parallel EEPROMs.

- We have verified that we can program a parallel EEPROM via the SPI Slave interface of the Support Chip.

- Support for 16 MBytes or 128 MBits of Storage.

- We have demonstrated that the Rad Hard Parallel EEPROMs in space can be reprogrammed using the Support Chip via a SPI interface.
In the tests, the SoC firmware was developed to support the setup of the ATMEL EEPROM for writing (burning) using industry standard Software Data Protection (SDP) protocol.

Figure 9 Test Setup for The Support Chip to Provide SPI Slave Interface to ATMEL Parallel EEPROM for Reading and Writing.

Figure 10 Close Up View of Daughter Card for ATMEL EEPROM interface through 100 MHz Hirose FX-2 Connector to Support Chip ASIC.

Figure 11 Test Results of the Support Chip ASIC used to Read and Write to the ATMEL EEPROM

32BIT OPENRISC SOC AND SUPPORT CHIP INTEGRATION (ROBOTIC ARM CONTROL)

In order to fully verify the 6 PWM cores, and the two I2C cores and the SPI Master core in the 32 bit OpenRISC SoC, we have interfaced the PWM(PTC) Cores to a research laboratory Robot ARM. One of the intended applications of the OpenRISC SoC is to control Robots in space or any servos that use PWM for control. One standard way to control servos by Hitech, used in Robotics, is to use a PWM signal with a frequency of 50 Hz and a pulse width between 700uS and 2600uS. To develop this test, the firmware for the 32 bit OpenCores CPU is developed on Linux with the GNU tool chain (32 bit Assembler in this case). The firmware is downloaded into on-chip Block SRAM (64Kx32) using the on-chip SPI Slave and a USB to SPI Bridge device (Diolan DLN). Using this approach many iterations of the software were developed and tested. Once the software was fully functional it was “burned” into the SPI 1Mbit ATMEL EEPROM on the test Board. It is then booted into the on-chip RAM using the on-chip SPI Bootup core. Once this is achieved, the system is independent of the Windows/Linux environment and the USB/SPI bridge device. It is can be deployed in Space.

The Robotic ARM Control system block diagram is shown in the Figure 12.
In Figure 12 we show that the 32 bit OpenRISC SoC ASIC communicates with two 4 channel ADC I2C devices that are interfaced to Joysticks. An I2C Core is also interfaced to an I2C LCD display (Showing Bar Graphs and Text). In this design, the Robot arm can be fully controlled with the joysticks (or Potentiometers). The LCD shows bar graphs of the current position corresponding to the PWM pulse for all 6 servos. This demonstrates that the 32bit OpenRISC SoC is ready for the control of complex systems in Space.

To demonstrate chip to chip communication with the 32bit OpenRISC SoC as SPI Master and the Support Chip as a slave, the two chips communicated such that the 32 bit SoC can individually control an array of 32 LEDs (4x8) through the Support Chip (acting as a port expander). This has been done and the LEDs show the PWM position as the joysticks are moved.

**Robotic ARM Demonstration System**

Figures 13 and 14 shows the Demonstration Board with the 32 bit OpenRISC SoC ASIC and Support Chip interfaced to the Robotic ARM. We also show that in this case the Robot ARM is controlled with 6 Potentiometers. The I2C LCD display is also shown. The Chip to Chip communication is through a SPI Master on the 32bit OpenRISC SoC to the SPI Slave on the Support Chip ASIC. The bonded out (to LGA 484 Package) ASICs are shown in the E-Tec LGA-484 Sockets in Figures 15 and 16.

In Figure 17 we show the I2C LCD Display indicating the Robot ARM Servo Positions using Bar Graphs. This is controlled by the 32 bit OpenRISC SoC ASIC Firmware. The LED Matrix also shows the ARM servo positions. The LED matrix is driven by the Support Chip via a SPI interface controlled by the 32 bit OpenRISC SoC as the SPI Master.
OPENRISC SOC ASIC C&DH WITH 8051XC SOC ASIC INTEGRATION FOR CUBESATS

A reference design for radiation hardened Satellite electronics can include the 32 bit OpenRISC SoC as the C&DH with the 8051XC based SoC supporting a large number of GPIO and built in Real Time Clock (RTC). The 8051 XC supports deep sleep and idle modes. The RTC can wake the chip up from deep sleep after multiple years of operation. The 8051XC SoC can also provide the 32bit OpenRISC SoC with SpaceWire support. The reference design is shown in Figure 18.

In this design we only use one Parallel EEPROM (NVM) for both SoC ASICs. The Support Chip converts the Parallel EEPROM to a SPI Slave Interface. The 32 bit OpenRISC SoC code is booted up using the on-chip SPI Master Bootup Core. Once the code is transferred to on-chip instruction RAM for execution, the 32 bit OpenRISC SoC can read the code for the 8051XC SoC and other potential chips through its SPI Master which can also communicated with the Support Chip as shown. This mode is supported in Silicon where either the SPI Bootup Core or the SPI Master Core can communicate with the SPI EEPROM. Therefore, the 32 bit OpenRISC SoC can read the 8051XC SoC Code from the SPI EEPROM and using the 8051XC SoC SPI Slave for Firmware Download, transfer the code to the 8051XC SoC on-chip SRAM. In this way the C&DH (32 bit OpenRISC SoC) can change the functionality of the 8051XC SoC multiple times during a mission.

The C&DH can also scrub the Block SRAM. The 32 bit OpenRISC SoC, C&DH in this configuration, can communicate with 8051XC SoC through the same SPI Master. This way it can gain access to the large number of GPIO and serial ports available on the 8051XC SoC. See [1]. The SPI Master has multiple chip selects. Note that the SPI Slaves do not provide tri-state MISO (Master In Slave Out) signals. So multiple Muxes are used as shown in Figure 18. Note the coloring on the Muxes. These can be associated with the 32 bit OpenRISC SoC and Support Chip. These chips have integrated Muxes on chip. We have eliminated the need for external logic chips. Also the support chip supports converting CMOS I2C to a Tri-State I2C Bus interface.

All chips run off a common 50 MHz clock. They also have asynchronous (SEU Immune) resets. The Real Time Clock requires a 32.768 kHz clock.

This reference design can be used in long term deep space missions where the Real Time Clock can wake up the system after a deep sleep period. All chips provide
SEU immunity in the harshest environment. The TID hardness up to 1 MRAD is limited only by the Parallel Non Volatile Memory which are available up to 1 MRAD TID hardness.

Figure 18 Reference Rad Hard Cubesat design with 32bit OpenRISC SoC ASIC as C&DH.

Acknowledgments
This work was funded by the Air Force Research Laboratory (AFRL).

References


