RunBMC - A Modular BMC Mezzanine Card
BUV - Bring Up Vehicle For BMC Mezzanine

Eric Shobe & Jared Mednick
Hardware Engineer - Salesforce
RunBMC
A Modular BMC Mezzanine

Eric Shobe & Jared Mednick, HW at Salesforce
eshobe@salesforce.com, jmednick@salesforce.com
Speaker Intro

Eric Shobe
Hardware Engineer
eshobe@salesforce.com

Jared Mednick
Hardware Engineer
jmednick@salesforce.com
Overview

Motivation

RunBMC Spec

Status
Good News

- **Security**: OpenBMC allows users to control security issues for out of band management (faster time to patch, custom implementations of open source software, secure boot)

- **Commonality**: Hardware BMC systems generally share commonality
  - A path to a low cost open hardware platform is possible

- **Adoption**: OpenBMC management stack is expanding!
Opportunities to Improve Out of Band Management

- **Implementations**: No two out of band server management systems are exactly the same. Some implementations are proprietary, both at the chipset and the hardware level.

- **Security**: Security issues require waiting on third parties to implement patches, even though many of these solutions exist today in open source.

- **Manageability**: These differences require engineering efforts to adapt existing management infrastructures for new server generations, which adds delays in delivering capacity.
RunBMC Benefits

Create an standard for hardware management

- Interface Consistency between vendors and implementations
- HW Security
- Supply Chain
- Can run OpenBMC management stack!
- An adaptable software platform
Baseboard Management Controller - Software

Software Stack
• Different software stacks in a BMC (Serial-Over-Lan, Sensors, Power Control, FRU’s, NC-SI, LAN Software, IPMI, USB, SNMP traps, Fan Speed/Control, Event Logs, etc)

Benefits of control and standardization, some examples
• Cost Savings to control your code
• Sensor Data
• Security
• Leverage your base code
• Increase of OpenBMC contributions
Baseboard Management Controller - Hardware

OCP and ODM/OEM systems share a common subsystem for the BMC.

Typical Hardware Feature set:
• SoC
• DDR4 (256 or 512MB), PCIe Gen2
• 2x SPI NOR Flash for BMC FW (32MB), 2x for BIOS FW
• 1x eSPI bus (Communication w/ CPLD or PCH)
• Multiple I2C bus (FRU, sensors), Multiple UARTs (OOB access), Multiple USB 2.0
• JTAG/LPC, GPIOs/TACH/PWM, VGA (sometimes)
• RGMII for 1GbT Ethernet, RMII for NC-SI or 100Mb
Baseboard Management Controller
Platform Analysis

- In depth analysis motivated the modular concept
- Next slides compare the different design topologies
## BaseBoard Management Controller - Platform Analysis

<table>
<thead>
<tr>
<th></th>
<th>GPIOs</th>
<th>IRQs</th>
<th>Total (GPIOs/IRQs/Others)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Server #1</td>
<td>58</td>
<td>32</td>
<td>103</td>
</tr>
<tr>
<td>Server #2</td>
<td>71</td>
<td>20</td>
<td>98</td>
</tr>
<tr>
<td>Server #3</td>
<td>92</td>
<td>8</td>
<td>111</td>
</tr>
</tbody>
</table>

Much of this signal functionality is typically shared with the PCH (for example, control of Host voltage supply)
Switch BMC Subsystem
OCP Server BMC Subsystem #1
OCP Server BMC Subsystem #2
OCP Server BMC Subsystem #3
Overview
RunBMC Specification Details

Proposal
- Modular Design
- Standardize BMC to HW interface
- Standardize the HW footprint

Collaboration!
- Work together to refine spec
<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
<th>J</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.3V</td>
<td>3.3V</td>
<td>5.3V</td>
<td>3.3V</td>
<td>3.3V</td>
<td>3.3V</td>
<td>3.3V</td>
<td>3.3V</td>
<td>3.3V</td>
<td>3.3V</td>
</tr>
<tr>
<td>2</td>
<td>1.3V</td>
<td>3.3V</td>
<td>5.3V</td>
<td>3.3V</td>
<td>3.3V</td>
<td>3.3V</td>
<td>3.3V</td>
<td>3.3V</td>
<td>3.3V</td>
<td>3.3V</td>
</tr>
<tr>
<td>3</td>
<td>GND</td>
<td>GND</td>
<td>GND</td>
<td>GND</td>
<td>GND</td>
<td>GND</td>
<td>GND</td>
<td>GND</td>
<td>GND</td>
<td>GND</td>
</tr>
<tr>
<td>4</td>
<td>ADC11</td>
<td>PHY_LINK1</td>
<td>GND</td>
<td>GND</td>
<td>GND</td>
<td>GND</td>
<td>GND</td>
<td>GND</td>
<td>GND</td>
<td>GND</td>
</tr>
<tr>
<td>5</td>
<td>ADC12</td>
<td>GND</td>
<td>LPC_CLK</td>
<td>GPION17</td>
<td>PWIM0</td>
<td>GPION5</td>
<td>I2C_SCL</td>
<td>GND</td>
<td>PERICLK</td>
<td>GND</td>
</tr>
<tr>
<td>6</td>
<td>ADC13</td>
<td>PHY_LINK2</td>
<td>LPC_CIIFM1_N</td>
<td>GPION0</td>
<td>PWIM1</td>
<td>GPION6</td>
<td>I2C_SDA</td>
<td>UART_TXD0</td>
<td>GND</td>
<td>PHY_LINK_N0</td>
</tr>
<tr>
<td>7</td>
<td>GND</td>
<td>GPION2</td>
<td>LPC_CTXIOQ</td>
<td>GPION1</td>
<td>PWIM2</td>
<td>GPION7</td>
<td>GND</td>
<td>UART_RXD0</td>
<td>PERI</td>
<td>GND</td>
</tr>
<tr>
<td>8</td>
<td>GPION4</td>
<td>GPION3</td>
<td>LPC_A0</td>
<td>GND</td>
<td>GND</td>
<td>GND</td>
<td>GND</td>
<td>GND</td>
<td>I2C_SCL</td>
<td>GND</td>
</tr>
<tr>
<td>9</td>
<td>PECVD</td>
<td>GPION4</td>
<td>LPC_A1</td>
<td>PWIM2</td>
<td>GND</td>
<td>GND</td>
<td>GND</td>
<td>UART_RXD0</td>
<td>PERI</td>
<td>PHY_LINK_P1</td>
</tr>
<tr>
<td>10</td>
<td>GPION5</td>
<td>GPION0</td>
<td>LPC_A2</td>
<td>GPION3</td>
<td>PWIM3</td>
<td>GPION8</td>
<td>SIOC_K</td>
<td>UART_TXD1</td>
<td>PERI</td>
<td>PHY_LINK_N1</td>
</tr>
<tr>
<td>11</td>
<td>GPION6</td>
<td>GPION1</td>
<td>LPC_A3</td>
<td>GPION4</td>
<td>PWIM4</td>
<td>GPION9</td>
<td>GPION10</td>
<td>UART_TXD1</td>
<td>PERI</td>
<td>PHY_LINK_P2</td>
</tr>
<tr>
<td>12</td>
<td>GPION7</td>
<td>GND</td>
<td>GPION0</td>
<td>PWIM5</td>
<td>GPION11</td>
<td>SYMCS0</td>
<td>SIOC</td>
<td>UART_RXD1</td>
<td>PERI</td>
<td>PHY_LINK_N2</td>
</tr>
<tr>
<td>13</td>
<td>GND</td>
<td>GPION2</td>
<td>SP2_CI3_N</td>
<td>GPION1</td>
<td>PWIM1</td>
<td>SYMCS0</td>
<td>GND</td>
<td>UART_RXD1</td>
<td>GND</td>
<td>GND</td>
</tr>
<tr>
<td>14</td>
<td>ADC14</td>
<td>TACH1</td>
<td>LTA</td>
<td>GPION2</td>
<td>PWIM2</td>
<td>GND</td>
<td>UART_RXD1</td>
<td>UART</td>
<td>UART</td>
<td>GND</td>
</tr>
<tr>
<td>15</td>
<td>ADC15</td>
<td>TACH2</td>
<td>LTA2</td>
<td>GPION3</td>
<td>PWIM3</td>
<td>GND</td>
<td>UART_RXD1</td>
<td>UART</td>
<td>UART</td>
<td>GND</td>
</tr>
<tr>
<td>16</td>
<td>GND</td>
<td>GPION3</td>
<td>PWIM4</td>
<td>GND</td>
<td>GND</td>
<td>GND</td>
<td>UART_RXD1</td>
<td>UART</td>
<td>UART</td>
<td>GND</td>
</tr>
<tr>
<td>17</td>
<td>SP2_SC1</td>
<td>LPC_RST_N</td>
<td>LTA2_TCK</td>
<td>GND</td>
<td>GND</td>
<td>GND</td>
<td>UART_RXD1</td>
<td>UART</td>
<td>UART</td>
<td>GND</td>
</tr>
<tr>
<td>18</td>
<td>GND</td>
<td>GPION2</td>
<td>PWIM5</td>
<td>GND</td>
<td>GND</td>
<td>GND</td>
<td>UART_RXD1</td>
<td>UART</td>
<td>UART</td>
<td>GND</td>
</tr>
<tr>
<td>19</td>
<td>GND</td>
<td>GPION3</td>
<td>PWIM6</td>
<td>GND</td>
<td>GND</td>
<td>GND</td>
<td>UART_RXD1</td>
<td>UART</td>
<td>UART</td>
<td>GND</td>
</tr>
<tr>
<td>20</td>
<td>MMC1_PCIE1</td>
<td>GND</td>
<td>GND</td>
<td>GND</td>
<td>GND</td>
<td>GND</td>
<td>UART_RXD1</td>
<td>UART</td>
<td>UART</td>
<td>GND</td>
</tr>
<tr>
<td>21</td>
<td>GPION5</td>
<td>PHY_LINK1</td>
<td>TACH1</td>
<td>VSA5</td>
<td>GPOIT5</td>
<td>UARTST1</td>
<td>UART</td>
<td>UART</td>
<td>GND</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>MMC1_PCIE2</td>
<td>GPION0</td>
<td>TACH1</td>
<td>VSA5</td>
<td>GPOIT5</td>
<td>UARTST2</td>
<td>UART</td>
<td>UART</td>
<td>GND</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>MMC1_PCIE3</td>
<td>GPOIT2</td>
<td>TACH2</td>
<td>GPOIT4</td>
<td>UARTST2</td>
<td>UART</td>
<td>UART</td>
<td>UART</td>
<td>GND</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>MMC1_PCIE4</td>
<td>GPOIT2</td>
<td>TACH3</td>
<td>GPOIT4</td>
<td>UART</td>
<td>UART</td>
<td>UART</td>
<td>UART</td>
<td>GND</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>MMC1_PCIE5</td>
<td>GPOIT2</td>
<td>TACH4</td>
<td>GPOIT4</td>
<td>UART</td>
<td>UART</td>
<td>UART</td>
<td>UART</td>
<td>GND</td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>GPOIT1</td>
<td>GPOIT4</td>
<td>TACH1</td>
<td>GPOIT4</td>
<td>UART</td>
<td>UART</td>
<td>UART</td>
<td>UART</td>
<td>GND</td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>GND</td>
<td>GPION5</td>
<td>TACH6</td>
<td>UART</td>
<td>UART</td>
<td>UART</td>
<td>UART</td>
<td>UART</td>
<td>GND</td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>USB2A_DP</td>
<td>GND</td>
<td>TACH1</td>
<td>UART</td>
<td>UART</td>
<td>UART</td>
<td>UART</td>
<td>UART</td>
<td>GND</td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>USB2A_DP</td>
<td>GND</td>
<td>TACH1</td>
<td>UART</td>
<td>UART</td>
<td>UART</td>
<td>UART</td>
<td>UART</td>
<td>GND</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>GPOIT6</td>
<td>TACH9</td>
<td>UART</td>
<td>UART</td>
<td>UART</td>
<td>UART</td>
<td>UART</td>
<td>UART</td>
<td>GND</td>
<td></td>
</tr>
</tbody>
</table>
RunBMC - Reference Board

Features

- SOC
- DDR4 (512MB)
- 300 pin 1.27mm pitch connector
- 1Gb Phy for RGMII
- Power Delivery standardized
- 2x SPI NOR for BMC FW
- Local I2C
- 3x UARTs, 2x USB, JTAG, LPC
- 2x SPI
- Tach/PWM
- 12x I2C
- VGA
- 1GbT and NC-SI
BUV - Bring Up Vehicle

Our Solution

- Bring-up board allows easy development cycles w/out hardware overhead. Access to network interfaces, UART, some basic parts for testing. Can be used at your desk, universities, etc.

Features

- RMII to 10/100 PHY
- 2x RJ45 for 1Gb and 100Mb
- Micro-USB access for console and power
- LPC, JTAG headers
- 12x I2C headers
- GPIO/ADC headers
- 7 seg display
- 3x console
- JTAG/LPC
- TPM
- 2x SPI bus
- TACH/PWM
- I2C devices (TPM, Temp, Current, expander)
- VGA
- Clocks
Overview

Motivation

RunBMC Spec

Status
Goal is to create a stand-alone BMC mezzanine (RunBMC) + bring-up board. Low risk project compared to full platform port (stage II). Success is defined by complete bring-up of board, booting linux, and ssh’ing into BMC.

<table>
<thead>
<tr>
<th>RunBMC mezzanine board</th>
<th>A re-usable daughter card containing the BMC chip, common subsystems (i2c, power, ddr, clocks, eeproms, flash), and a small pitch connector for any hardware platform to use (switches, servers, etc). Only the platform SW changes.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bring-up vehicle board</td>
<td>RunBMC mezzanine and Bring-up board allows easy development cycles w/out hardware overhead. Access to network interfaces, UART, some basic parts for testing. Can be used at your desk, universities, etc</td>
</tr>
<tr>
<td>Re-use</td>
<td>RunBMC mezzanine would allow faster turnaround from a hardware development life-cycle, earlier platform software development, and higher software adoption from developers (using the development bring-up board)</td>
</tr>
</tbody>
</table>
RunBMC - Bring-Up Success!
Integrate RunBMC mezzanine with ODM server platform of choice. Riskier port due to interactions with CPU, PCH, CPLD/FGPA subsystems, power-on, and deeper software integration. However - lots of code will be re-used.

<table>
<thead>
<tr>
<th>RunBMC mezzanine board</th>
<th>The re-usable daughter card defined in Stage I.</th>
</tr>
</thead>
<tbody>
<tr>
<td>ODM server platform</td>
<td>Our ODM platform modified (schematic/layout) to use the BMC mezzanine board.</td>
</tr>
<tr>
<td>Software Integration</td>
<td>Deeper integration to SFDC tools (agents on the BMC)</td>
</tr>
</tbody>
</table>
Citations

- Server Layout Images:
Questions?
THANK YOU