Firmware-Based Validation Module for NextGeneration Mobile Processors

Sponsor: Intel Corporation
Mentors: Dan Iparraguirre, Christian Le, Rahul Khanna
Jordan Anderson, Abdulhakim Giydan, Calvin Odell, Spencer Schuler

Overview
The project aims to automate product validation and analysis for the high-volume manufacturing of mobile processors by creating a cloud-based system that analyzes large portions of data. The final product is to allow the OEMs to access the cloud network and review the validation data of their products. In addition, the OEMs would be able to enable a variety of selected tests for different products.

Our tasks
Develop firmware packages, modify drivers, and transfer collected data in the UEFI:
- RMT
- Metadata
- eMMC Margin Testing
- Optimize Runtime of Margin Testing

System Overview

Project Details
RMT
- A tool used to run specialized tests on the memory units of a processor
- Developed firmware packages that used Intel data structures to transfer margining results from the UEFI to OS
- Debugged incorrect values in the margining results and implemented a method to repair these values in the UEFI

Metadata
- Used to define specified metadata that requires to read registers from the platform
- The metadata includes the customer ID, product family, and product revision
- Used to define which OEMs platform data was sent from, allowing data from different OEMs to be grouped accordingly in the cloud

eMMC Margin Testing
- Test eMMC signals at increasing and decreasing conditions until failure
- Transmit (Tx) and Receive (Rx) voltage and timing tested, with high and low limits as results
- Results stored into a specialized structure for access at later UEFI stages in addition to the OS

<table>
<thead>
<tr>
<th>Test Number</th>
<th>Frequency (MHz)</th>
<th>Data Rate</th>
<th>Test Mode</th>
<th>Highside Margin</th>
<th>Lowside Margin</th>
<th>Training Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>200</td>
<td>0</td>
<td>1</td>
<td>7</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>200</td>
<td>0</td>
<td>0</td>
<td>10</td>
<td>-2</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>50</td>
<td>0</td>
<td>1</td>
<td>33</td>
<td>-7</td>
<td>7</td>
</tr>
<tr>
<td>3</td>
<td>50</td>
<td>0</td>
<td>0</td>
<td>11</td>
<td>-20</td>
<td>20</td>
</tr>
<tr>
<td>4</td>
<td>50</td>
<td>1</td>
<td>1</td>
<td>9</td>
<td>-3</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>50</td>
<td>1</td>
<td>0</td>
<td>17</td>
<td>-12</td>
<td>14</td>
</tr>
</tbody>
</table>

Optimize Runtime of Margin Testing
- Include functions to allow for accurate timing of RMT loops
- Store high/low results from preliminary margining done in MRC
- Extract and utilize stored results to provide more refined starting point in RMT
- Develop function to ensure RMT margining does not begin past a fail point
- Decrease in runtime beneficial to overall system efficiency

Impact Considerations
Economic Impact
- Reduced runtime over large scale manufacturing translates to product’s shorter time to market and potentially bigger profits
- Cost of validation cycle significantly diminished, potential price drop for consumers

Validation Time
- Improved validation schemes allow more units tested in shorter amount of time
- Increase productivity at both client and stakeholder level due to automation

Customization
- Allows each module to have its own verification set
- Initial time to create validation firmware is potential issue
- Need to ensure firmware uses good practices to allow easier transition to new platforms

New Platforms
- Not everything will be able to be tested against bugs so tradeoffs will need to be made
- Time to create and update will be significantly less than current implementations
- Decisions on modularity versus functionality for current platform

Glossary
UEFI: Unified Extensible Firmware Interface is a standard firmware interface for PCs, designed to replace BIOS (basic input/output system)
eMMC: Embedded Multi Media Card, a solid-state storage device commonly found on mobile platforms
PCD: A setting established during the time that the platform UEFI/Bootloader is built
Metadata: A way of replicating hardware registers without the need of changing the hardware itself, using PCDs

Acknowledgment
We would like to thank the following people for their support and contributions: Christian Le, Peter Tan, Rahul Khanna, and Varalakshmi Prabhala.

Team Leopard