IP Decryption Management

Jeff Fox
July 9, 2012

Scope of This Presentation

- Protection of plain text IP source code at any level of abstraction
- Control over authorization for EDA tools to decrypt
- Tool-specific permissions not addressed here
  - Each EDA, FPGA or ASIC vendor may support a large variety of controls over what may be done with encrypted and licensed IP
    - Example: Altera permissions are encoded in IP encryption header and authorized by license. Least restrictive wins for each right

Types of IP Decryption Authorization

- Open – no license required for specified EDA tools
  - Permissions granted by IP provider during encryption, embedded in encryption header
  - Altera uses this for most company owned IP
    - IP can be parameterized, simulated, synthesized, fit
    - FPGA can be programmed, but device will stop working when time limit expires
- License required
  - IP vendor must provide a license to grant permission to decrypt for specified EDA tools
  - In Altera's implementation, the IP vendor crypt key is in the Flexlm license in an encrypted format

Granularity of Permissions

- To provide as much control and flexibility to IP providers as possible, the 1735 spec should allow a range of decryption authorization choices from coarse to fine grained
  - Independent of whether the "open" or "licensed" permission models are used
- EDA vendors may choose the level of control that they will support

Examples of Decryption Authorization Options

- Vendor(s)
- List of product names or codes
- List of release numbers or date codes
  - Minimum
  - Exact
  - Maximum
  - No limit
- List of SW components in tools, including version numbers or date codes for each component
  - Would allow authorization for tools with common code bases