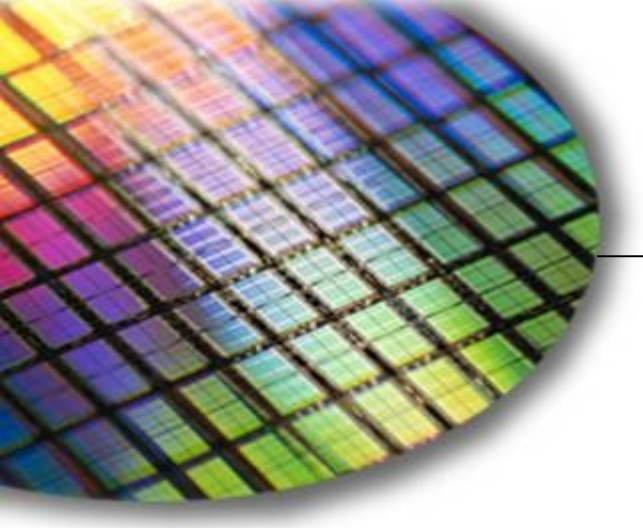


The World Leader in High Performance Signal Processing Solutions

Code Security Techniques

Presented by Dan Ledger, DSP FAE

February 22nd, 2005

ANALOG DEVICES

Agenda

- Why is Code Security Important?
- Introduction to Code Security Techniques
  1. Litigation Assistance
  2. Preventative Protection Techniques
- Code Security on Current ADI Processors
  - ADSP-BF53x Family
  - ADSP2126x/36x Family
  - ADSP-TS20x Family
- Code Security on Future ADI Processors
  - ADSP-BF54x Family

Why is Code Security More Important?

- IP theft more prevalent now than ever.
- Product features and value defined more by software than hardware -> customers' "special sauce" is the software.
- ADI is targeting more high-volume consumer applications.

No Way to Guarantee 100% Code Security

You can add an alarm to your car to deter a thief but nothing is going to stop someone with a flatbed truck from getting your vehicle.

With enough time and money, any software or hardware security technique can be circumvented.

...and companies specialize in circumventing security

- By delaminating a semiconductor, it is presently possible to remotely probe the contents of RAM, ROM and FLASH on most devices.
- There are companies scattered about the world that offer the above service at very reasonable prices.
- According to ADI sources, there is a company in Poland that will perform this service for around $2,000 USD.
- We have a contract with this company to notify us if anyone contracts them to perform this service on ADI parts.

Code/IP Security Techniques : 2 Categories

1. **Litigation Assistance**

   With enough time and money, any hardware or software protection scheme can be broken. Adding "dead code" signatures to the code base that don't get called, or are not functional can be used in a court case to prove code stealing.

1. **Preventative Protection**

   While no security technique is 100% effective, there are many techniques that can be employed to protect against invasive and non-invasive attacks.

Famous Litigation Case

- A famous computer “Company X” in the 1970s manufactured terminals.
- IBM accused “Company X” of stealing its font generation software.
- Company X denied these allegations and the case went to trial.
- IBM demonstrated that the lower-case ‘f‘ in their character set was missing a pixel due to an error in the rendering software.
- IBM also demonstrated that company X’s font rendering software also incorrectly generated an ‘f’ in the same manner.

Preventative Protection

1. **System-Level Protection**
   - **Mutual Watchdog**

     A small microcontroller and processor provide mutual watchdog timer services. If either processor is halted, the other processor
   - **2nd Stage Decrypting Boot Loader**
1. **Mechanical Protection**
   - **Board Layout**

     Package selection and layout techniques can make it very difficult to probe critical signals.
   - **Masking**

     Covering sensitive areas of the board with adhesives (Epoxy) can make it very difficult to probe critical signals. Labeling on devices should be removed.
   - **Self-Destruct Circuitry**

     Self-destruct circuits can be used to destroy critical devices when the board is tampered with.
1. **Chip-Level Protection**
   - **On-chip ROM or FLASH**

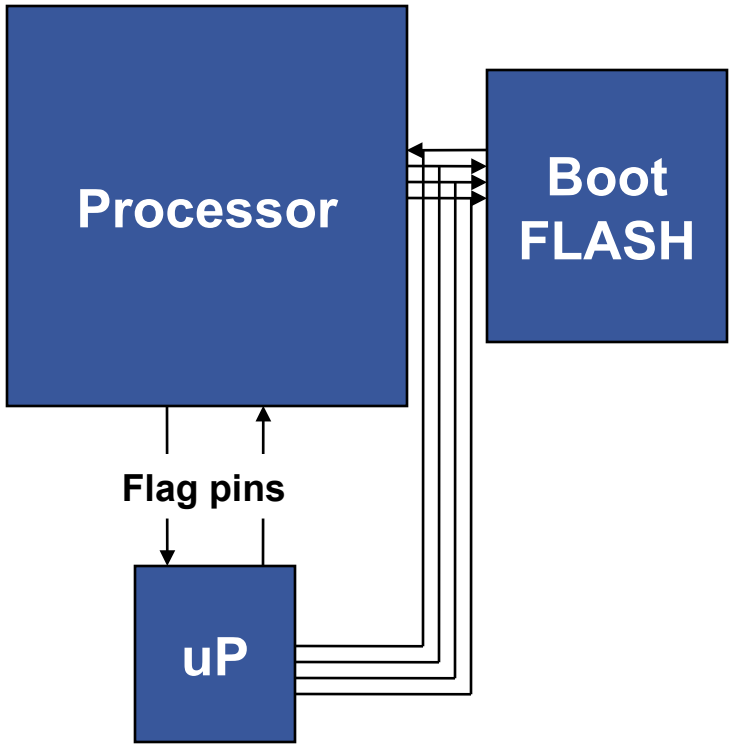
     On-chip memory is obviously much harder to access or probe than an external ROM or FLASH
   - **On-chip fuses**

     On-chip fuses can be blown to disable peripherals that would allow access to the internal memory and state of the processor such as a JTAG port or a host port.
   - **Encryption**

     A one-time programmable user decryption key allows contents of external boot FLASH or host boot stream to be encrypted. The stream is decrypted internally when the part boots.

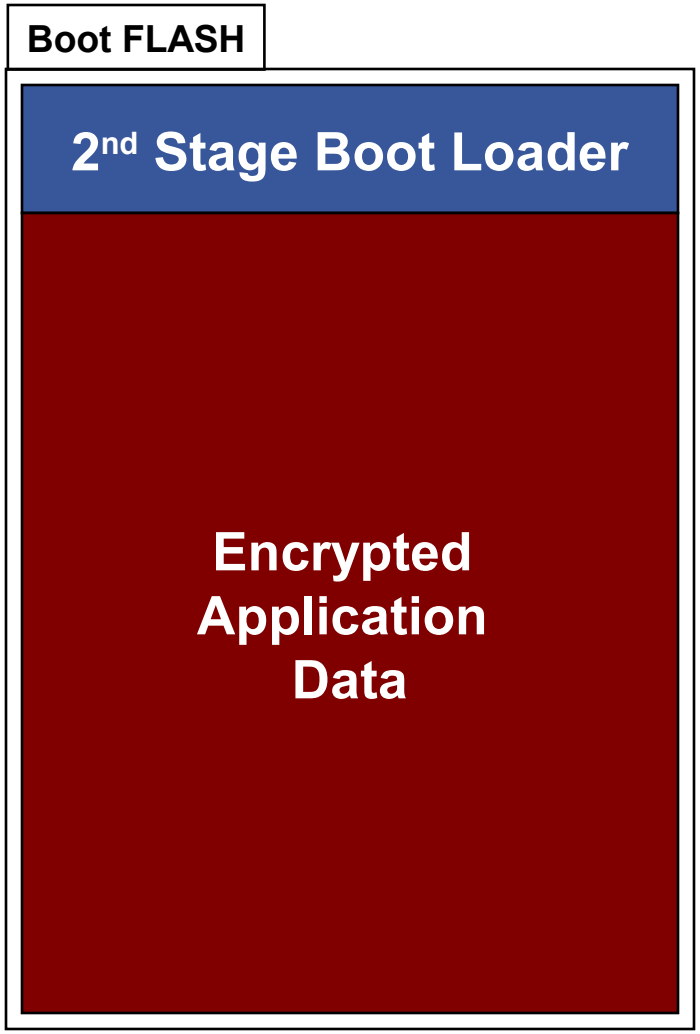
System-Level Protection : Mutual Watchdog

- Requires small microprocessor
- Both processor and microprocessor implement watchdog timer using each other.
- If one processor stops responding (i.e. halted via JTAG, removed from board), the other processor performs countermeasure.
  - Erase on-board FLASH
  - Remove power to other processor
  - Destroy processors and FLASH



System-Level Protection : 2nd Stage Decrypting Boot Loader

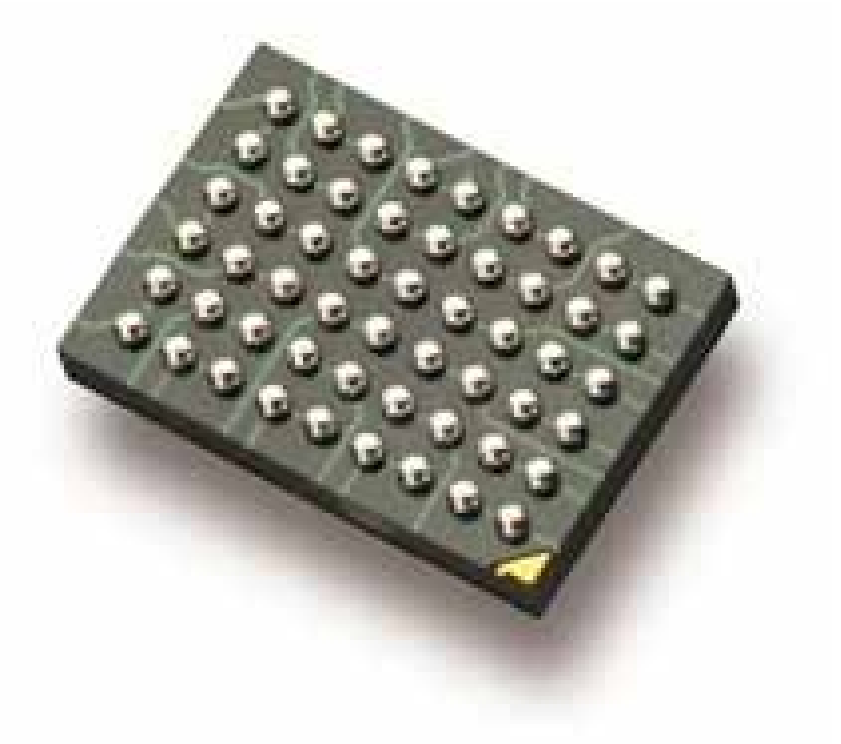
- First stage boot loader brings in custom second stage-loader
- Application data is encrypted and stored in non-linearly in external FLASH or ROM.
- Second stage loader decrypts and loads application from boot FLASH.



Mechanical Protection : Board Layout Techniques

Use BGA packages for both the DSP and boot FLASH

- It is very difficult to probe BGA signals without drilling through the PCB.
- Select packages where critical signals are not on outer-most row of balls.
- Smaller pitched BGAs are even more difficult to probe.



Mechanical Protection : Board Layout Techniques

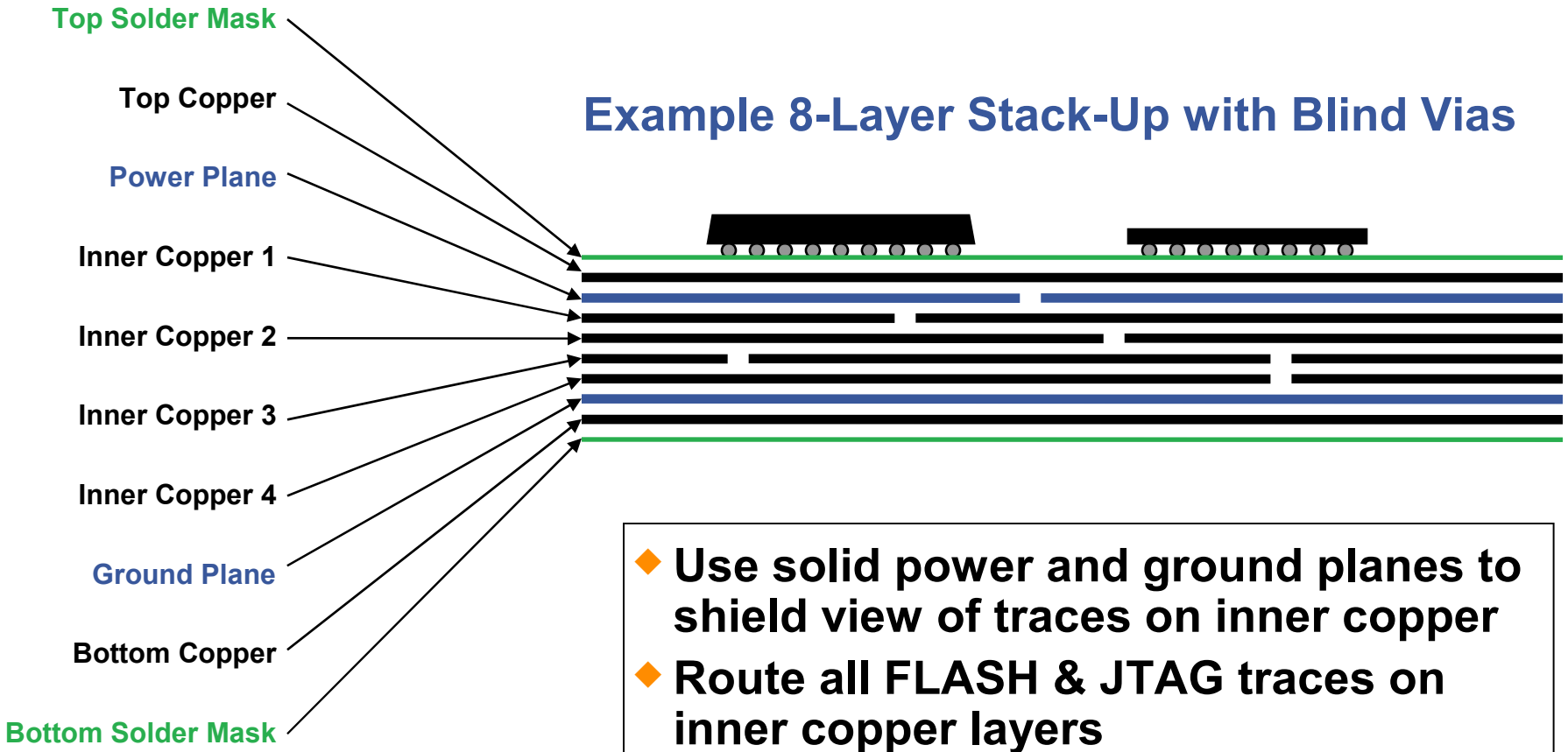
Place the packages as close together as possible

- Placing processor and external FLASH as close together as possible makes it more difficult locate and gain access to critical signals.

ANALOG DEVICES

Mechanical Protection : Board Layout Techniques

Use blind vias and place power and ground planes towards top and bottom of layer stack



- Use solid power and ground planes to shield view of traces on inner copper
- Route all FLASH & JTAG traces on inner copper layers

Mechanical Protection : Masking Compounds

Cover sensitive areas of the board with black Epoxy

- Hides chips' location, orientation and labeling
- Effective means of hiding pins packages with exposed pins (LQFP, DIP, etc)
- Effective means of preventing access to outer balls in BGA packages
- Cover critical spots or the entire board

Mechanical Protection : Remove Labels

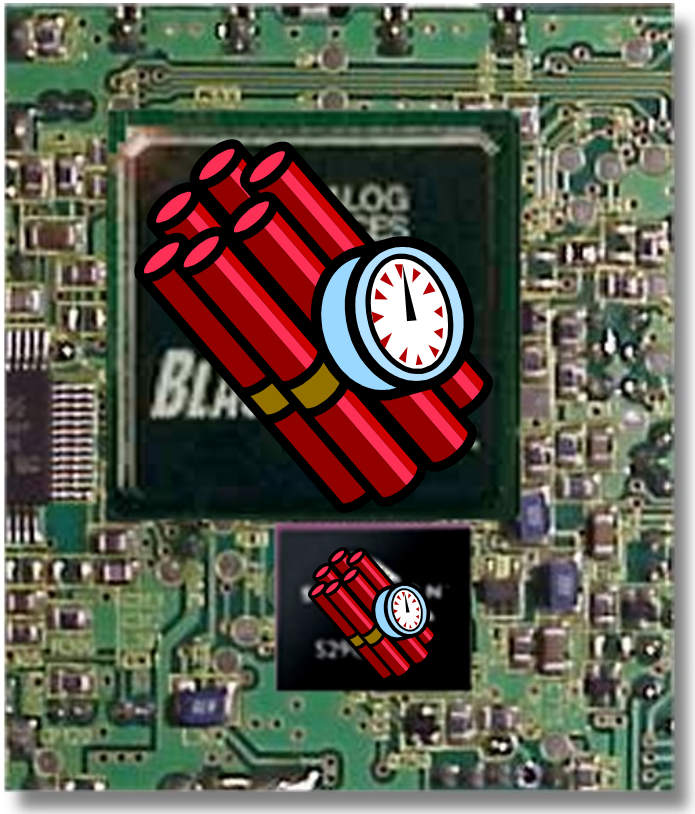
Removing the labels hides vendor and part number

- Labels can be easily removed chemically or mechanically.
- Many semiconductor devices come in standard packages making it difficult to identify the device without opening the package.
- Parts can be re-labeled to appear to be custom ASICs.

Mechanical Protection : Self-Destruct

Develop a circuit that will destroy the devices containing sensitive information if enclosure is breached

- Use battery powered over-voltage circuit to destroy sensitive ICs if enclosure is breached even if power is disconnected.



Chip-Level Protection : On-Chip ROM or FLASH

- Some processors contain internal FLASH or ROM memory which cannot be read from outside the device. This can be a good place to hold protected code.
- Processor must boot from internal FLASH/ROM for this protection to be effective.
  - If not, hacker could load custom code from external memory to read out contents of internal FLASH/ROM.
- FLASH/ROM should be optionally protected from host or emulator accesses via fuses.
- FLASH/ROM can contain proprietary decrypting boot loader to load encrypted applications from FLASH.

Chip-Level Protection : On-chip fuses

- On-Chip fuses can be “blown” to disable certain features on the chip.
- On-Chip fuses are typically used to disable JTAG and/or force a processor to boot from internal ROM/FLASH if it’s available.
- Typically only useful with parts /w ROM/FLASH

Chip-Level Protection : Encryption

- Some processors contain a user-programmable ROM encryption key which can be used to decrypt a boot stream from external FLASH or a host processor.
- This is typically a 64-bit key which is stored in a OTP ROM location on the processor.
- The development tool then allows for the encryption of a boot loader file using this custom 64-bit key before it is programmed in FLASH memory.

A look at some ADI processors

- ADSP-BF53x Family
- ADSP-2126x and ADSP-2136x Families
- ADSP-TS20x Family
- ADSP-BF54x Family

ASDP-BF53x Family – Security Weaknesses

1. JTAG port cannot be disabled
   - Problem

     When the part is running, someone can connect an emulator to the JTAG port, halt the core and upload code and data.
   - Most Effective Protection Techniques

     "Mutual Watchdog", Mechanical Protections
1. Boot stream is not encrypted
   - Problem

     Signals between boot device and Blackfin can be tapped to recover source code.
   - Most Effective Protection Techniques

     "2nd Stage Decrypting Boot Loader", Mechanical Protections

ASDP-BF53x Family – Security Features

None

ASDP-2126x/36x Family – Security Weaknesses

1. **Boot stream is not encrypted**
   - **Problem**

     Signals between boot device and Blackfin can be tapped to recover source code.
   - **Most Effective Protection Techniques**

     Use custom ROM

     "2nd Stage Decrypting Boot Loader", Mechanical Protections

ASDP-2126x/36x Family – Security Features

1. **JTAG Port *Can* Be Disabled**
   - **An internal fuse can be blown which will disable the JTAG port on the device.**
     - preventing someone from halting the processor and uploading all of your code
   - **Requires unique 64-bit key to provided via emulator to open port.**
2. **Boot from Internal ROM**
   - **An internal fuse can be blown which causes the part to boot from internal ROM rather than from an external device.**
     - Proprietary decrypting boot loader can be placed in internal ROM so customers can safely store their application in external FLASH. This would allow the customer to still maintain field upgradeability.
     - Entire application can be placed in ROM.
     - The volumes would have to be large enough to meet requirements for custom masked ROM.

ASDP-TS201 Family – Security Weaknesses

1. JTAG port cannot be disabled
   - Problem

     When the part is running, someone can connect an emulator to the JTAG port, halt the core and upload code and data.
   - Most Effective Protection Techniques

     "Mutual Watchdog", Mechanical Protections
1. Boot stream is not encrypted
   - Problem

     Signals between boot device and Blackfin can be tapped to recover source code.
   - Most Effective Protection Techniques

     "2nd Stage Decrypting Boot Loader", Mechanical Protections

ASDP-TS201 Family – Security Features

None

BF54x Security Features : *Preliminary and Confidential!*

- **Many new features being considered for BF54x to facilitate good code security.**

- Blown fuse(s) will be used to protect silicon. Material removed via laser fuses cannot be reconstructed by hackers in order to place silicon back into an unprotected mode.
- A unique chip identification or user ID (equivalent to the user name), stored on the chip which is unalterable after entry
- Complete or partial disabling of emulation, test, download and most of JTAG features
- A 64-bit user-programmable ROM encryption key which is stored in a OTP ROM location on the processor will be used to decrypt a boot stream from external FLASH/host. Decryption algorithm (such as Rijngdael / AES or possibly RSA) will reside in ROM as Blackfin executable instruction code.
- Implement forced delay/timeout between subsequent key entries (2ms?). This is a form of security to extend the time it would take hackers from discovering OTP key code by inputting multiple key codes in brute force attack. Worst case crack time for 64-bit key : 1,169,884,834 Years!