



**XILINX**

ALL PROGRAMMABLE™



Embedded Linux  
Conference Europe

## **U-Boot: Verified RSA** ***Boot on ARM target***

JagannadhaSutradharudu Teki

U-Boot Mini Summit- Edinburgh, 2013 Oct

# Agenda

- Zynq U-Boot
- SPI Custodianship
- Verified Boot
- RSA Concept
- U-boot Verified RSA Boot
- Current u-boot state(Simon's support)
- U-boot needs
- Demo run
- TODO
- References

# Zynq U-Boot



<https://github.com/Xilinx/u-boot-xlnx/commits/xilinx-v14.6.01>

GitHub This repository Search or type a command Explore Features Enterprise Blog Sign up Sign in

Xilinx / u-boot-xlnx Star 12 Fork 19

tag: xilinx-v14.6.01 u-boot-xlnx / Commits

**Jul 04, 2013**

**sf: Correct typo mistake on bank\_boun in dual parallel** ... 0f6dbff16b  
Jagannadha Sutradharudu Teki authored 4 months ago  
→ michalsimek committed 4 months ago Browse code

**Jun 18, 2013**

**zynq: ddr: Change memory size when 16bit mode is used** ... 471ec625b2  
michalsimek authored 4 months ago Browse code

**Jun 17, 2013**

**fpga: zynqpl: Add support for zc7100 device.** ... 1d51b81e18  
michalsimek authored 4 months ago Browse code

**zynq: Add new ddr: driver for ECC support** ... 90cc1a8ae2  
michalsimek authored 4 months ago Browse code

**Jun 16, 2013**

**fpga: zynqpl: Clear loopback mode during device init** ... 2ed81b9ef6  
sorenb-xlnx authored 4 months ago  
→ michalsimek committed 4 months ago Browse code

**Jun 14, 2013**

**sf: Compute the dual parallel write chunk\_len based on offset** ... 3cee697241  
Jagannadha Sutradharudu Teki authored 4 months ago  
→ michalsimek committed 4 months ago Browse code

- Good customer support till now – feature additions SPI/QSPI, support new boards, d-caches and bug fixes
- ~75% of u-boot-xlnx code is in ML, rest will push soon.

# SPI Custodianship

[\[u-boot.git\]](#) / [doc](#) / [SPI](#) / [status.txt](#)

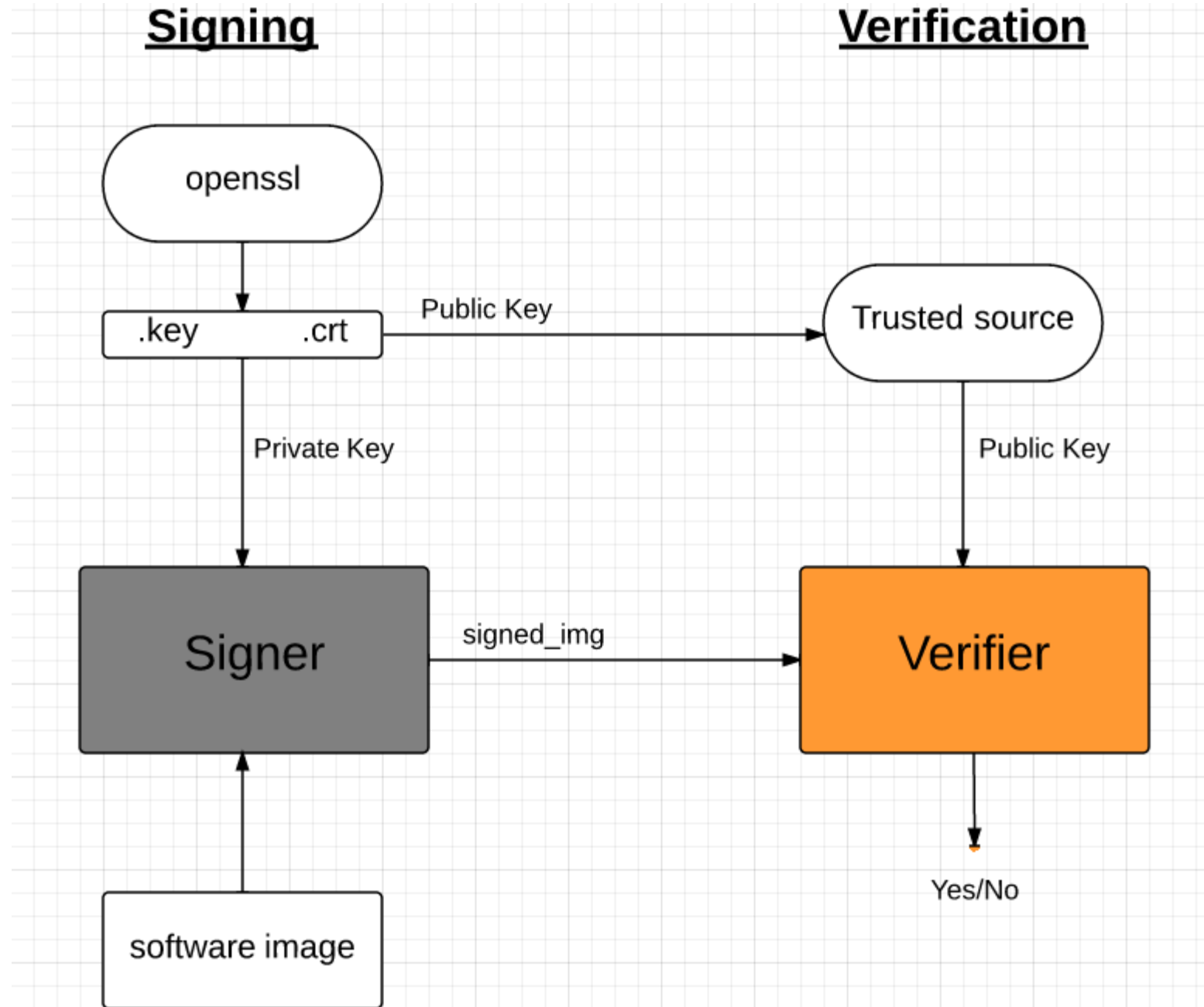
---

```
1 Status on SPI subsystem:
2 =====
3
4 SPI COMMAND (common/cmd_sf, cmd_spi):
5 -
6
7 SPI FLASH (drivers/mtd/spi):
8 - sf_probe.c: SPI flash probing code.
9 - sf_ops.c: SPI flash operations code.
10 - sf.c: SPI flash interface, which interacts controller driver.
11 - Bank Address Register (Accessing flashes > 16Mbytes in 3-byte addressing)
12 - Added memory_mapped support for read operations.
13 - Common probe support for all supported flash vendors except, ramtron.
14
15 SPI DRIVERS (drivers/spi):
16 -
17
18 TODO:
19 - Runtime detection of spi_flash params, SFDP(if possible)
20 - Add support for multibus build/accessing.
21 - Extended read commands support(dual read, dual IO read)
22 - Quad Page Program support.
23 - Quad Read support(quad fast read, quad IO read)
24 - Dual flash connection topology support(accessing two spi flash memories with single cs)
25 - Banking support on dual flash connection topology.
26 - Need proper cleanups on spi_flash and drivers.
27
```

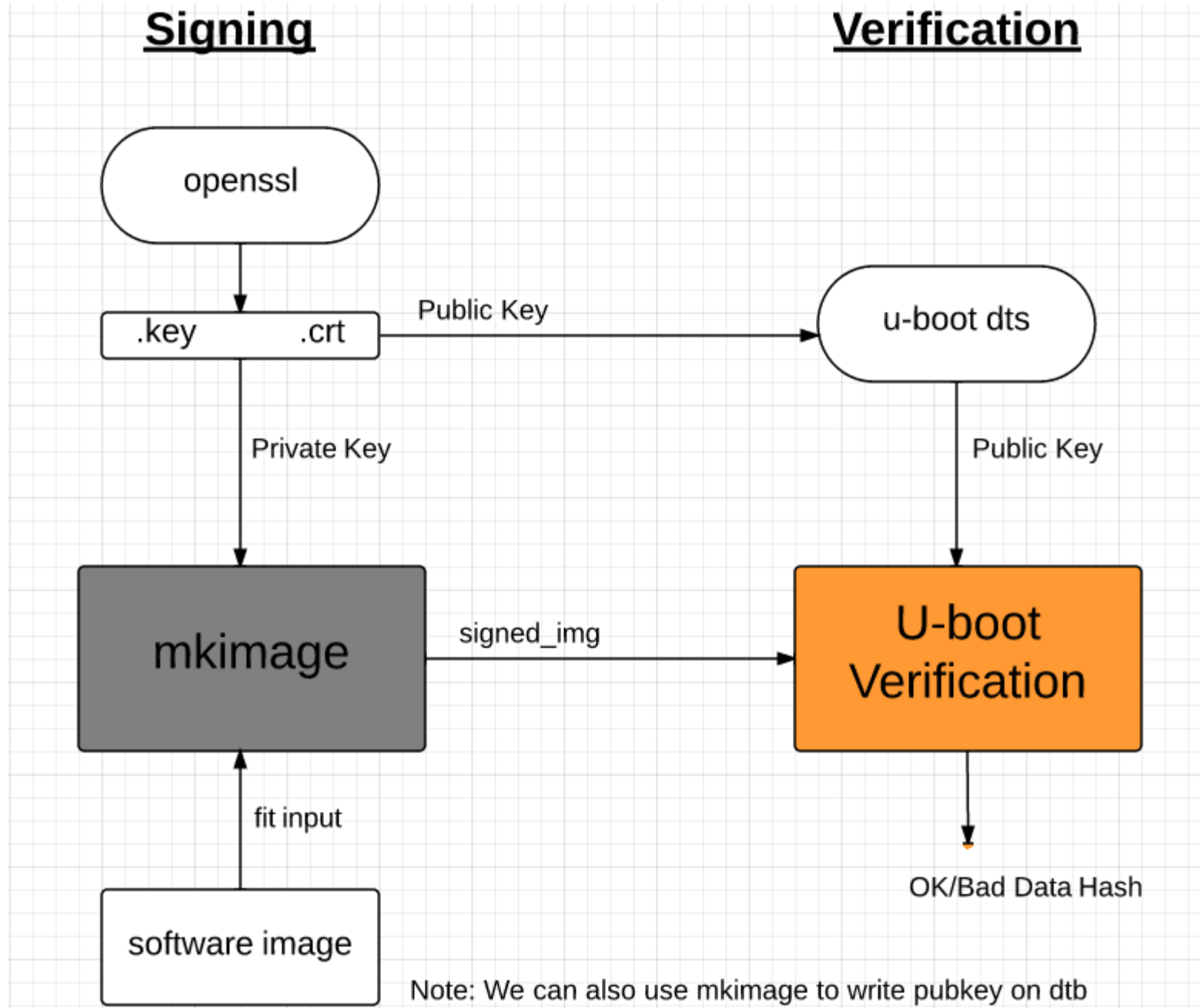
# Verified Boot

- Verified - Secure - Trusted boot
- Verify the loaded software to ensure that it is authorized during boot.
- Benefits:
  - Prevent from malware
  - Provide authorized read access
  - Machine safe - runs only signed software
  - Possible to filed-upgrade the software

# RSA Concept



# U-boot Verified RSA Boot



# Current u-boot state (Simon's support)

2013-06-26	Dirk Behme	<b>spi: mxc_spi: Fix pre and post divider calculation</b>	<a href="#">commit</a>   <a href="#">commitdiff</a>   <a href="#">tree</a>   snapshot ( <a href="#">tar.gz</a> <a href="#">tar.bz2</a> )
2013-06-26	Simon Glass	<b>Add verified boot information and test</b>	<a href="#">commit</a>   <a href="#">commitdiff</a>   <a href="#">tree</a>   snapshot ( <a href="#">tar.gz</a> <a href="#">tar.bz2</a> )
2013-06-26	Simon Glass	<b>sandbox: config: Enable FIT signatures with RSA</b>	<a href="#">commit</a>   <a href="#">commitdiff</a>   <a href="#">tree</a>   snapshot ( <a href="#">tar.gz</a> <a href="#">tar.bz2</a> )
2013-06-26	Simon Glass	<b>image: Add support for signing of FIT configurations</b>	<a href="#">commit</a>   <a href="#">commitdiff</a>   <a href="#">tree</a>   snapshot ( <a href="#">tar.gz</a> <a href="#">tar.bz2</a> )
2013-06-26	Simon Glass	<b>libfdt: Add fdt_find_regions()</b>	<a href="#">commit</a>   <a href="#">commitdiff</a>   <a href="#">tree</a>   snapshot ( <a href="#">tar.gz</a> <a href="#">tar.bz2</a> )
2013-06-26	Simon Glass	<b>mkimage: Add -r option to specify keys that must be...</b>	<a href="#">commit</a>   <a href="#">commitdiff</a>   <a href="#">tree</a>   snapshot ( <a href="#">tar.gz</a> <a href="#">tar.bz2</a> )
2013-06-26	Simon Glass	<b>mkimage: Add -c option to specify a comment for key...</b>	<a href="#">commit</a>   <a href="#">commitdiff</a>   <a href="#">tree</a>   snapshot ( <a href="#">tar.gz</a> <a href="#">tar.bz2</a> )
2013-06-26	Simon Glass	<b>mkimage: Add -F option to modify an existing .fit file</b>	<a href="#">commit</a>   <a href="#">commitdiff</a>   <a href="#">tree</a>   snapshot ( <a href="#">tar.gz</a> <a href="#">tar.bz2</a> )
2013-06-26	Simon Glass	<b>mkimage: Add -K to write public keys to an FDT blob</b>	<a href="#">commit</a>   <a href="#">commitdiff</a>   <a href="#">tree</a>   snapshot ( <a href="#">tar.gz</a> <a href="#">tar.bz2</a> )
2013-06-26	Simon Glass	<b>mkimage: Add -k option to specify key directory</b>	<a href="#">commit</a>   <a href="#">commitdiff</a>   <a href="#">tree</a>   snapshot ( <a href="#">tar.gz</a> <a href="#">tar.bz2</a> )
2013-06-26	Simon Glass	<b>image: Add RSA support for image signing</b>	<a href="#">commit</a>   <a href="#">commitdiff</a>   <a href="#">tree</a>   snapshot ( <a href="#">tar.gz</a> <a href="#">tar.bz2</a> )
2013-06-26	Simon Glass	<b>image: Support signing of images</b>	<a href="#">commit</a>   <a href="#">commitdiff</a>   <a href="#">tree</a>   snapshot ( <a href="#">tar.gz</a> <a href="#">tar.bz2</a> )
2013-06-26	Simon Glass	<b>image: Add signing infrastructure</b>	<a href="#">commit</a>   <a href="#">commitdiff</a>   <a href="#">tree</a>   snapshot ( <a href="#">tar.gz</a> <a href="#">tar.bz2</a> )
2013-06-26	Simon Glass	<b>x86: config: Add tracing options</b>	<a href="#">commit</a>   <a href="#">commitdiff</a>   <a href="#">tree</a>   snapshot ( <a href="#">tar.gz</a> <a href="#">tar.bz2</a> )
2013-06-26	Simon Glass	<b>x86: Support tracing function</b>	<a href="#">commit</a>   <a href="#">commitdiff</a>   <a href="#">tree</a>   snapshot ( <a href="#">tar.gz</a> <a href="#">tar.bz2</a> )
2013-06-26	Simon Glass	<b>exynos: config: Add tracing options</b>	<a href="#">commit</a>   <a href="#">commitdiff</a>   <a href="#">tree</a>   snapshot ( <a href="#">tar.gz</a> <a href="#">tar.bz2</a> )
2013-06-26	Simon Glass	<b>exynos: Avoid function instrumentation for microsecond...</b>	<a href="#">commit</a>   <a href="#">commitdiff</a>   <a href="#">tree</a>   snapshot ( <a href="#">tar.gz</a> <a href="#">tar.bz2</a> )
2013-06-26	Simon Glass	<b>arm: Implement the 'fake' go command</b>	<a href="#">commit</a>   <a href="#">commitdiff</a>   <a href="#">tree</a>   snapshot ( <a href="#">tar.gz</a> <a href="#">tar.bz2</a> )
2013-06-26	Simon Glass	<b>Add a 'fake' go command to the bootm command</b>	<a href="#">commit</a>   <a href="#">commitdiff</a>   <a href="#">tree</a>   snapshot ( <a href="#">tar.gz</a> <a href="#">tar.bz2</a> )
2013-06-26	Simon Glass	<b>Refactor the bootm command to reduce code duplication</b>	<a href="#">commit</a>   <a href="#">commitdiff</a>   <a href="#">tree</a>   snapshot ( <a href="#">tar.gz</a> <a href="#">tar.bz2</a> )
2013-06-26	Simon Glass	<b>Clarify bootm OS arguments</b>	<a href="#">commit</a>   <a href="#">commitdiff</a>   <a href="#">tree</a>   snapshot ( <a href="#">tar.gz</a> <a href="#">tar.bz2</a> )
2013-06-26	Simon Glass	<b>Add a simple test for sandbox trace</b>	<a href="#">commit</a>   <a href="#">commitdiff</a>   <a href="#">tree</a>   snapshot ( <a href="#">tar.gz</a> <a href="#">tar.bz2</a> )
2013-06-26	Simon Glass	<b>sandbox: Support trace feature</b>	<a href="#">commit</a>   <a href="#">commitdiff</a>   <a href="#">tree</a>   snapshot ( <a href="#">tar.gz</a> <a href="#">tar.bz2</a> )
2013-06-26	Simon Glass	<b>Add proftool to decode profile data</b>	<a href="#">commit</a>   <a href="#">commitdiff</a>   <a href="#">tree</a>   snapshot ( <a href="#">tar.gz</a> <a href="#">tar.bz2</a> )
2013-06-26	Simon Glass	<b>Add trace support to generic board</b>	<a href="#">commit</a>   <a href="#">commitdiff</a>   <a href="#">tree</a>   snapshot ( <a href="#">tar.gz</a> <a href="#">tar.bz2</a> )
2013-06-26	Simon Glass	<b>Support tracing in config.mk when enabled</b>	<a href="#">commit</a>   <a href="#">commitdiff</a>   <a href="#">tree</a>   snapshot ( <a href="#">tar.gz</a> <a href="#">tar.bz2</a> )
2013-06-26	Simon Glass	<b>Add a trace command</b>	<a href="#">commit</a>   <a href="#">commitdiff</a>   <a href="#">tree</a>   snapshot ( <a href="#">tar.gz</a> <a href="#">tar.bz2</a> )
2013-06-26	Simon Glass	<b>Add trace library</b>	<a href="#">commit</a>   <a href="#">commitdiff</a>   <a href="#">tree</a>   snapshot ( <a href="#">tar.gz</a> <a href="#">tar.bz2</a> )
2013-06-26	Simon Glass	<b>Add function to print a number with grouped digits</b>	<a href="#">commit</a>   <a href="#">commitdiff</a>   <a href="#">tree</a>   snapshot ( <a href="#">tar.gz</a> <a href="#">tar.bz2</a> )
2013-06-26	Simon Glass	<b>bootstage: Correct printf types</b>	<a href="#">commit</a>   <a href="#">commitdiff</a>   <a href="#">tree</a>   snapshot ( <a href="#">tar.gz</a> <a href="#">tar.bz2</a> )
2013-06-26	Simon Glass	<b>Show stdout on error in fit-test</b>	<a href="#">commit</a>   <a href="#">commitdiff</a>   <a href="#">tree</a>   snapshot ( <a href="#">tar.gz</a> <a href="#">tar.bz2</a> )
2013-06-26	Simon Glass	<b>Fix missing return in do_mem_loop()</b>	<a href="#">commit</a>   <a href="#">commitdiff</a>   <a href="#">tree</a>   snapshot ( <a href="#">tar.gz</a> <a href="#">tar.bz2</a> )

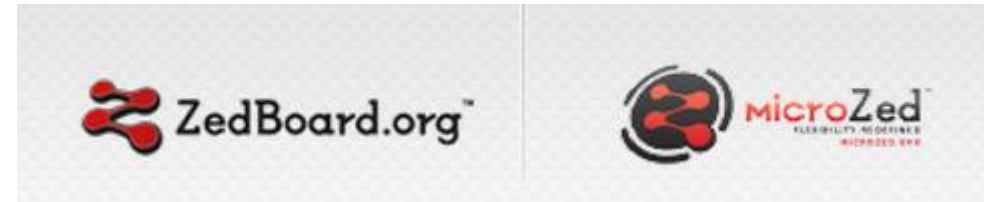


# U-boot needs

- Enable FIT
  - CONFIG\_FIT - enable support for the FIT uImage format
- Enable FDT
  - CONFIG\_OF\_CONTROL
  - CONFIG\_OF\_SEPARATE
- *Enable verified boot*
  - *CONFIG\_FIT\_SIGNATURE - enables signature verification of FIT images*
  - *CONFIG\_RSA - enables the RSA algorithm used for FIT image verification*

# Demo run...

- Build FDT u-boot
- Build rsa\_signed image
- Build FDT u-boot with public key
- Run rsa\_signed image



# Build FDT u-boot

➤ Setup the toolchain:

<http://www.wiki.xilinx.com/Zynq+Base+TRD+14.5#x-5> Building the U-boot Boot Loader

➤ Clone u-boot-spi.git

```
$ git clone git://git.denx.de/u-boot-spi.git
$ cd u-boot-spi
$ git checkout -b master-xlnx origin/master-xlnx
```

➤ U-boot build

```
$ make zynq_zed_config
$ make DEVICE_TREE=zynq-zed -j4
```

```

/dts-v1/;
/ {
    description = "Simple image with single Linux kernel, FDT blob and ramdisk";
    #address-cells = <0x1>;
    images {
        kernel@1 {
            description = "Zynq Linux kernel";
            data = /incbin/("./vmlinux.bin.gz");
            type = "kernel";
            arch = "arm";
            os = "linux";
            compression = "gzip";
            load = <0x8000>;
            entry = <0x8000>;
            hash@1 {
                algo = "sha1";
            };
            signature@1 {
                algo = "sha1,rsa2048";
                key-name-hint = "dev";
            };
        };
        fdt@1 {
            description = "ZED board Flattened Device Tree blob";
            data = /incbin/("./devicetree.dtb");
            type = "flat_dt";
            arch = "arm";
            compression = "none";
            hash@1 {
                algo = "sha1";
            };
            signature@1 {
                algo = "sha1,rsa2048";
                key-name-hint = "dev";
            };
        };
    };
};

ramdisk@1 {
    description = "Ramdisk Image";
    data = /incbin/("./ramdisk.image.gz");
    type = "ramdisk";
    arch = "arm";
    os = "linux";
    compression = "gzip";
    load = <0x00800000>;
    entry = <0x00800000>;
    hash@1 {
        algo = "sha1";
    };
    signature@1 {
        algo = "sha1,rsa2048";
        key-name-hint = "dev";
    };
};

};
configurations {
    default = "conf@1";
    conf@1 {
        description = "Boot Linux kernel, FDT blob and ramdisk";
        kernel = "kernel@1";
        fdt = "fdt@1";
        ramdisk = "ramdisk@1";
    };
};
};

```

# Build rsa\_signed

➤ RSA key generation:

- Create RSA key pair

```
$ openssl genrsa -F4 -out mykeys/dev.key 2048
```

- Create a certificate contains public key

```
$ openssl req -batch -new -x509 -key mykeys/dev.key -out mykeys/dev.crt
```

➤ Create dtb for existing u-boot dts

```
$ dtc -p 0x1000 board/xilinx/dts/zynq-zed.dts -O dtb -o zynq-zed.dtb
```

```
$ cp zynq-zed.dtb zynq-zed-pubkey.dtb
```

➤ Sign the images with mykeys

```
$ DTC_OPS="-I dts -O dtb -p 2000"
```

```
$ mkimage -D "${DTC_OPS}" -f rsa.its -K zynq-zed-pubkey.dtb -k mykeys -r rsa_signed.img
```

# Build FDT u-boot with public key

- For building FDT u-boot with public key- externally

```
$ make DEV_TREE_BIN=./zynq-zed-pubkey.dtb
```

u-boot-dtb.bin -> Is final FDT u-boot image with public key on it, hence the pubkey will used in verification process.

```
zynq-uboot> bootm 0x2000000
## Loading kernel from FIT Image at 02000000 ...
Using 'conf@1' configuration
Verifying Hash Integrity ... OK
Trying 'kernel@1' kernel subimage
  Description: Zynq Linux kernel
  Type: Kernel Image
  Compression: gzip compressed
  Data Start: 0x020000f0
  Data Size: 2972178 Bytes = 2.8 MiB
  Architecture: ARM
  OS: Linux
  Load Address: 0x00008000
  Entry Point: 0x00008000
  Hash algo: md5
  Hash value: 3601aec79bd62a71a43e72880a41d24
  Hash algo: sha1
  Hash value: 5c10a3632e83939349d9ea6d42e3e9fa861d5193
  Sign algo: sha1,rsa2048:dev
  Sign value:1b63d3e6c0277836026779f8fa4bebaed46d97d4d3ce4ce4e39f10aff4e79da2a796c04619806e6a8d7ae17
65d670a934f21370a84af6ac1cf7cc74d66ee9c7a619b7a636508bc8cffe73dcb155dcd5b262c1cb9582e4d2cf05315c701dc53
a56ec93e56ddaeb5c7b334aedc73e13e75d45c5d9b9c2004683420378a0f9c34bbbab724256e9fac56c9a8b3375e0c8cd9334a6
4ed35f21b51306ae603e73802961c0e150d2aa8aa6c9b50d8f7447e1f1083dd2542231579f40aae89456d39bd09ab50bed8e8f6
43369426c60ab41be2aad89df5918a5a0802daca5a21313f40b22f54376d11ff4229c9507bedd99c7cc2bf440237b72372cec5793194c56c372d
  Verifying Hash Integrity ... sha1,rsa2048:dev+ md5+ sha1+ OK
## Loading ramdisk from FIT Image at 02000000 ...
Using 'conf@1' configuration
Trying 'ramdisk@1' ramdisk subimage
  Description: Ramdisk Image
  Type: RAMDisk Image
  Compression: gzip compressed
  Data Start: 0x022d7cf8
  Data Size: 3688961 Bytes = 3.5 MiB
  Architecture: ARM
  OS: Linux
  Load Address: 0x00000000
```

# TODO

- Possible TODO's @ doc/uImage.FIT/signature.txt
- ***Signed\_image creations support for bootable images (SPL) or FIT support in SPL ???***

```
images {
    spl@1 {
        description = "Zynq SPL";
        data = /incbin("./SPL.bin");
        type = "spl";
        arch = "arm";
        compression = "none";
        load = <0x0>;
        entry = <0x0>;
        hash@2 {
            algo = "sha1";
        };
        signature@1 {
            algo = "sha1,rsa2048";
            key-name-hint = "dev";
        };
    };
    u-boot@1 {
        description = "Zynq u-boot";
        data = /incbin("./u-boot.bin");
        type = "u-boot";
        arch = "arm";
        compression = "none";
        load = <0x4000000>;
        entry = <0x4000000>;
        hash@2 {
            algo = "sha1";
        };
        signature@1 {
            algo = "sha1,rsa2048";
            key-name-hint = "dev";
        };
    };
};
```



# References

➤ Zynq u-boot-xlnx.git repo

<https://github.com/Xilinx/u-boot-xlnx>

➤ For verified boot: doc/uImage.FIT/verified-boot.txt

➤ For signature: doc/uImage.FIT/signature.txt

➤ Sample sign its: doc/uImage.FIT/sign-configs.its

➤ Code for this demo run

<http://git.denx.de/?p=u-boot/u-boot-spi.git;a=shortlog;h=refs/heads/master-xlnx>

➤ Possible TODO's on doc/uImage.FIT/signature.txt

➤ Any questions - mail to [sjg@chromium.org](mailto:sjg@chromium.org) CC [u-boot@lists.denx.de](mailto:u-boot@lists.denx.de), [jagannadh.teki@gmail.com](mailto:jagannadh.teki@gmail.com)