**OTP Controller request feature**

Revision history

|  |  |  |  |
| --- | --- | --- | --- |
| Revision | Date | Description | Author |
| 0.1 | 2016-12-05 | Initial | SongPan |
| 0.2 | 2016-12-12 | Update AHB interface with secure/non-secure mode access  Remove HW read HDCP key interface  Add 16bit ECC | Song pan |
| 0.3 | 2016-12-19 | Add BISR function nice to have | Songpan |
| 0.4 | 2017-1-18 | Remove BISR  Add OTP\_busy/change AHB slave interface to AXI slave interface | Songpan |

# OTP Controller request feature

## Logic to control access to CPU debug ports depending on level set in fuses.

The password is stored into OTP.

2’b11: Locked

2’b10/01: password mode

2’b00: open

## ~~Private key interface to other on-chip security module to transfer content securely. For HDMI/DP block.~~

~~i.e. HDMI HDCP KEY 0..39 is 40\*56 bit and HDMI HDCP KEY SELECT is 40bit.~~

~~DP HDCP KEY 0..39 is 40\*56 bit and HDMI HDCP KEY SELECT is 40bit.~~

~~387byte key ???~~

## Different type of OTP stored data

2 type of data : configure type data and key type data

Chip ID information is automatically loaded by HW logic when system power on.

Secret KEY for HDMI HDCP /DP is loaded by HW logic when master issue, such as bank1 data

RSA KEY(2048bit) is loaded by CPU, such as bank32 data

SPERCIAL USER KEY is only loaded by security mode and not for user mode.

## Split 128Kb with 0-31 bank with 1Kb and bank32/bank33/bank34 with 32Kb. Bank0—debug and test control

Bank1—HDCP Keys/DP keys

Bank2—boot control

Bank3—chip identification

Bank4—feature selection

…

Bank32—public key store

Bank33/34—reserved

(need consider it more)

## Two programming method support: via IO pad or through register programming by system CPUs.

## It’s 1st IP to be taken out of reset when the system powered up, need generate one OTP\_ready signal to release the rest of chip reset signal.

## Add one output signal as OTP\_chip\_disable, when performing POR checks or when a problem is deteted.

## Two sets of flip-flop are used to latch the fuse data. After programmed, the value can be immediately updated in one of the sets. The content in the second set is used to store the configuration used by the system. It will not be updated with the new values until the next power-up.

## The whole OTP data need protected by ECC (16bit mode).

## ~~One APB32 slave interface for register access.~~

## ~~One HDMI HDCP/DP key programming interface~~

## ~~One AHB slave interface for register access, for Secure mode and non-secure mode.~~

## One AXI slave interface for register access with secure mode or non-secure mode.

## One programming interface from IO PAD

## Running clock speed is 100Mhz from PLL

## Async reset is only from POR. Negative level is valid.

## ~~BISR function only support 2 entry fix, nice to have feature. No need support.~~

## \*\*\*During POR checks, AXI interface can’t work, should be locked to avoid the unnecessary operation.

## Descript the data location which use redundant data protect, which is locked by HW.

## Three modes: POR check (~~need read by three times???~~), function mode (when programming data, need no more than 16 times), debug test mode (in this mode, can be access by off-chip external bus)

\*\*\*Password only can write once from CPU

## \*\*\*\*\*\*Add checksum feature for POR check auto read data block. Sum all data into 1 byte and the final result data should be 8’hFF. (Checksum algorithm: sum all data into 1byte and inverse the result data, then program into checksum area). Store the result into RO register. Don’t do any block control for OTP\_ready.

**\*\*secure debug password only program one time after POR. If compare result is failed, reset the chip is needed, which increasing crack password time.**

**Add internal read only register to show current secure debug level: no-password protect, password protect, locked.**

**Add internal read-only register to show current password compare result: no password set, failed, passed.**