xflash_image_check Application

IN THIS DOCUMENT

1 Summary

The XMOS xflash_image_check application has been provided to aid customers identify xflash images that contain image tags on sector boundaries that are not valid images. It does this by scanning a given xflash generated binary file for image tags and if determined to be on a potential sector boundary, it performs a CRC check of the first page of the image and compares this against the expected first page CRC in the image table. If the CRC compares then it is a valid image. If the CRC does not compare then it is possible that DFU could fail as described in the tools advisory.

Publication Date: 2017/9/19 Document Number: XM-UNKNOWNA

2 Help

```
xflash_image_check --help
Options:
  -h [ --help ]                           Display this information
  -f [ --file ] arg (=flash_image.bin)  XFLASH binary file to be checked
  -v [ --image-tag-version ] arg (=tools_13)
                                          Specify which version of the image
                                          header tag is being used.
                                          If XFLASH binary file generated
                                          with xTIMEcomposer version 12.X
                                          or below use tools_12.
                                          If XFLASH binary file generated
                                          with xTIMEcomposer version 13.X
                                          or above use tools_13.
  -t [ --flash-device-type ] arg (=QSPI)
                                          Specify what type of flash device
                                          is being used.  SPI or QSPI
  -s [ --sector-size ] arg (=4096)        Size of the sectors of intended
                                          flash device.  Refer to the data
                                          sheet for your device for the
                                          sector size.

For bug reporting instructions, please see:
http://www.xmos.com/support
```

3 Example Use

For xTIMEcomposer versions greater than or equal to 13.0.0 using QSPI:

```
xflash_image_check --file=flash_image.bin --image-tag-version=tools_13
--flash-device-type=QSPI --sector-size=4096
```

For xTIMEcomposer versions greater than or equal to 13.0.0 using SPI:

```
xflash_image_check --file=flash_image.bin --image-tag-version=tools_13
--flash-device-type=SPI --sector-size=4096
```

For xTIMEcomposer versions less than 13.0.0 using SPI:

```
xflash_image_check --file=flash_image.bin --image-tag-version=tools_12
--flash-device-type=SPI --sector-size=4096
```

4 Output

The xflash_image_check application will output the following:

If a valid image is found on a sector boundary:

Valid Image found on sector boundary @32768

If an invalid image is found on a sector boundary:

Invalid Image found on sector boundary @32768 Warning this could cause DFU to fail if factory image was compiled with xTIMEcomposer 14.3.0 of previous

If no images were found:

No image tags found in file

Check that you are using the correct version of the image tag and that you have specified the correct type of flash device. It may also be possible that the file given does not contain any upgrade images.

5 Return Code

The application xflash_image_check will return 0 if valid images were found. It will return 1 if invalid program options were provided, if no images were found or if invalid images were found. An accompanying error message will be provided on the console with more specific details of the error.

6 Notes

- It is not possible for a binary file generated with tools 12 or below to use a QSPI device.
- Sector size can vary from device to device. Please refer to the data sheet for your chosen flash device to use the correct sector size.
- Some SPI devices may also have irregular sector size. In this case it is sensible to use the smallest of those sector sizes.
- Only upgrade images are found on sector boundaries. The factory image is always located immediately after the second stage bootloader. This may, on rare occasion be coincidentally on a sector boundary but will be CRC checked to validate.