DEVC API 函数说明

1. *u8 FDevcPs_init(FDevcPs_T *dev, u32 addr)*

描述	* This function initializes devc device
参数	* @param dev is devc handle. * @param addr is the base address of CSU.
返回值	* @return * - 0 if successful * - 1 not support

2. *u8 FDevcPs_fabricInit(FDevcPs_T *dev, u32 TransferType)*

描述	* This function initializes Device Configuration module, configure pcap interface work parameters
参数	* @param TransferType. * - FMSH_PCAP_READBACK * - FMSH_NON_SECURE_PCAP_WRITE * - FMSH_PCAP_LOOPBACK
返回值	* @return * - 0 if successful * - 1 if unsuccessful

3. *u8 FDevcPs_unLockCSU(FDevcPs_T *dev)*

描述	* This function unlocks CSU module
参数	* @param dev is devc handle.
返回值	* @return * - 0 if successful * - 1 if unsuccessful

4. *u8 FDevcPs_lockCSU(FDevcPs_T *dev)*

描述	* This function locks CSU module
参数	* @param dev is devc handle.
返回值	* @return * - 0 if successful * - 1 if unsuccessful

5. *u8 FDevcPs_KUPKEY(FDevcPs_T *dev, u32 *p, u32 len)*

描述	* This function configures KUPKEY * this key value maybe changed by different bitstream
参数	* @param dev is devc handle. * @param p is pointer to KEY buffer. * @param len is length of KEY, this should be fix as 8.
返回值	* @return * - 0 if successful * - 1 if unsuccessful

6. u8 FDevcPs_DEVKEY(FDevcPs_T *dev, u32 *p, u32 len)

描述	* This function configures DEVKEY * this key value maybe changed by different bitstream
参数	* @param dev is devc handle. * @param p is pointer to KEY buffer. * @param len is length of KEY, this should be fix as 8.
返回值	* @return * - 0 if successful * - 1 if unsuccessful

7. u8 FDevcPs_IV(FDevcPs_T *dev, u32 *p, u32 len)

描述	* This function configures fixed IV * this IV value maybe changed by different bitstream
参数	* @param dev is devc handle. * @param p is pointer to IV buffer. * @param len is length of IV, this should be fix as 4.
返回值	* @return * - 0 if successful * - 1 if unsuccessful

8. u8 FDevcPs_getConfigdata(FDevcPs_T *dev, u32 *DestinationDataPtr, u32 DestinationLength, u32 addr, u32 ConfigReg)

描述	* This function returns the value of the specified configuration register or bitstream.
参数	* @param dev is devc handle. * @param DestinationDataPtr contains a pointer to the destination memory

	* where the data is to be transferred to. * @param DestinationLength is the number of words (32 bit) to be transferred * for the Destination transfer. * @param addr is the value of the specified configuration * register. * @param ConfigReg is a constant which represents the configuration * register value to be returned. * if(ConfigReg == 0xaa55) radback bitstream
返回值	* @return * - 0 if successful * - 1 if unsuccessful

9. u8 FDevcPs_pcapLoadPartition(FDevcPs_T *dev, u32 *SourceDataPtr, u32 *DestinationDataPtr, u32 SourceLength, u32 DestinationLength, u32 SecureTransfer)

描述	* This function starts the DMA transfer. This function only starts the * operation and returns before the operation may be completed. * If the interrupt is enabled, an interrupt will be generated when the * operation is completed, otherwise it is necessary to poll the Status register * to determine when it is completed. It is the responsibility of the caller to * determine when the operation is completed by handling the generated interrupt * or polling the Status Register.
参数	* @param dev is devc handle. * @param SourceDataPtr contains a pointer to the source memory where the data * is to be transferred from. * @param SourceLength is the number of words (32 bit) to be transferred

	* for the source transfer. * @param DestinationDataPtr contains a pointer to the destination memory * where the data is to be transferred to. * @param DestinationLength is the number of words (32 bit) to be transferred * for the Destination transfer. * @param SecureTransfer * FMSH_NON_SECURE_PCAP_WRITE 0 * FMSH_SECURE_PCAP_WRITE 1 * FMSH_PCAP_READBACK 2
返回值	* @return * - 0 if successful * - 1 if unsuccessful

10. *u8 FDevcPs_keyRollingDownload(FDevcPs_T *dev, u8 alg_flag, u8 opkey_flag, u32 *srcPtr, u32 bitlen)*

描述	* This function is used to secure download different BITSTREAM
参数	* @param alg_flag： 0 -- AES 1 -- SM4 * @param opkey_flag： 0 -- no opkey 1 -- use opkey * @param bitlen: PL BITSTREAM LENGTH
返回值	* @return 0 if successful, otherwise 1.

11. *u8 FDevcPs_pollFpgaDone(FDevcPs_T *dev, u32 maxcount)*

描述	* This function polls FPGA done
参数	* @param dev is devc handle. * @param maxcount is the max poll times
返回值	* @return

	* - 0 if successful * - 1 if unsuccessful