Realtek rtl8195a crypto engine

This document provide guideline for crypto engine.

Table of Contents

找不到目錄項目。

1 Hardware Crypto Engine Support

Hardware Crypto Engine can support the following Hash and Crypto functions:
Hash (include HMAC): MD5 / SHA1 / SHA2 (224,256)
Crypto : AES (CBC/ECB/CTR), 3DES (CBC/ECB), DES(CBC/ECB)

The prototype of function is shown below:

```
// Crypto Engine
  extern int rtl_cryptoEngine_init(void);
// md5
  extern int rtl_crypto_md5(IN const u8* message, IN const u32 msglen, OUT u8* pDigest);
  extern int rtl_crypto_md5_init(void);
  extern int rtl_crypto_md5_process(IN const u8* message, const IN u32 msglen, OUT u8* pDigest);
// sha1
  extern int rtl_crypto_sha1(IN const u8* message, IN const u32 msglen, OUT u8* pDigest);
  extern int rtl_crypto_sha1_init(void);
  extern int rtl_crypto_sha1_process(IN const u8* message, IN const u32 msglen, OUT u8* pDigest);
// sha2
  extern int rtl_crypto_sha2(IN const SHA2_TYPE sha2type,
                                        IN const u8* message, IN const u32 msglen, OUT u8* pDigest);
  extern int rtl_crypto_sha2_init(IN const SHA2_TYPE sha2type);
  extern int rtl_crypto_sha2_process(IN const u8* message, IN const u32 msglen, OUT u8* pDigest);
// HMAC-md5
  extern int rtl_crypto_hmac_md5(IN const u8* message, IN const u32 msglen,
  extern int rtl_crypto_hmac_md5_init(IN const u8* key, IN const u32 keylen);
  extern int rtl_crypto_hmac_md5_process(IN const u8* message, IN const u32 msglen, OUT u8*
pDigest);
// HMAC-sha1
  extern int rtl_crypto_hmac_sha1(IN const u8* message, IN const u32 msglen,
                          IN const u8* key, IN const u32 keylen, OUT u8* pDigest);
  extern int rtl_crypto_hmac_sha1_init(IN const u8* key, IN const u32 keylen);
  extern int rtl_crypto_hmac_sha1_process(IN const u8* message, IN const u32 msglen, OUT u8*
pDigest);
// HMAC-sha2
  extern int rtl_crypto_hmac_sha2(IN const SHA2_TYPE sha2type, IN const u8* message, IN const u32
msglen, IN const u8* key, IN const u32 keylen, OUT u8* pDigest);
  extern int rtl_crypto_hmac_sha2_init(IN const SHA2_TYPE sha2type, IN const u8* key, IN const u32
keylen);
  extern int rtl_crypto_hmac_sha2_process(IN const u8* message, IN const u32 msglen, OUT u8*
pDigest);
```

```
//
// Cipher Functions
//
// AES - CBC
  extern int rtl_crypto_aes_cbc_init(IN const u8* key, IN const u32 keylen);
  extern int rtl_crypto_aes_cbc_encrypt(
      IN const u8* message, IN const u32 msglen,
      IN const u8* iv, IN const u32 ivlen, OUT u8* pResult);
  extern int rtl_crypto_aes_cbc_decrypt(
      IN const u8* message, IN const u32 msglen,
      IN const u8* iv, IN const u32 ivlen, OUT u8* pResult);
// AES - ECB
  extern int rtl_crypto_aes_ecb_init(IN const u8* key, IN const u32 keylen);
  extern int rtl_crypto_aes_ecb_encrypt(
      IN const u8* message, IN const u32 msglen,
      IN const u8* iv, IN const u32 ivlen, OUT u8* pResult);
  extern int rtl_crypto_aes_ecb_decrypt(
      IN const u8* message, IN const u32 msglen,
      IN const u8* iv, IN const u32 ivlen, OUT u8* pResult);
// AES - CTR
  extern int rtl_crypto_aes_ctr_init(IN const u8* key, IN const u32 keylen);
  extern int rtl_crypto_aes_ctr_encrypt(
      IN const u8* message, IN const u32 msglen,
      IN const u8* iv, IN const u32 ivlen, OUT u8* pResult);
  extern int rtl_crypto_aes_ctr_decrypt( IN const u8* message, IN const u32 msglen,
      IN const u8* iv, IN const u32 ivlen, OUT u8* pResult);
// 3DES - CBC
  extern int rtl_crypto_3des_cbc_init(IN const u8* key, IN const u32 keylen);
  extern int rtl_crypto_3des_cbc_encrypt(
      IN const u8* message, IN const u32 msglen,
      IN const u8* iv, IN const u32 ivlen, OUT u8* pResult);
  extern int rtl_crypto_3des_cbc_decrypt(
      IN const u8* message, IN const u32 msglen,
      IN const u8* iv, IN const u32 ivlen, OUT u8* pResult);
// 3DES - ECB
  extern int rtl_crypto_3des_ecb_init(IN const u8* key, IN const u32 keylen);
  extern int rtl_crypto_3des_ecb_encrypt(
      IN const u8* message, IN const u32 msglen,
      IN const u8* iv, IN const u32 ivlen, OUT u8* pResult);
  extern int rtl_crypto_3des_ecb_decrypt(
      IN const u8* message, IN const u32 msglen,
      IN const u8* iv, IN const u32 ivlen, OUT u8* pResult);
// DES - CBC
  extern int rtl_crypto_des_cbc_init(IN const u8* key, IN const u32 keylen);
  extern int rtl_crypto_des_cbc_encrypt(
      IN const u8* message, IN const u32 msglen,
      IN const u8* iv, IN const u32 ivlen, OUT u8* pResult);
```

```
 extern int rtl_crypto_des_cbc_decrypt(
      IN const u8* message, IN const u32 msglen,
      IN const u8* iv, IN const u32 ivlen, OUT u8* pResult);
// DES - ECB
 extern int rtl_crypto_des_ecb_init(IN const u8* key, IN const u32 keylen);
 extern int rtl_crypto_des_ecb_encrypt(
      IN const u8* message, IN const u32 msglen,
      IN const u8* iv, IN const u32 ivlen, OUT u8* pResult);
 extern int rtl_crypto_des_ecb_decrypt(
      IN const u8* message, IN const u32 msglen,
      IN const u8* iv, IN const u32 ivlen, OUT u8* pResult);
```