

基本地址+0x2C	读	0x0000_0000	AES_Tx_DIN_04	AES Tx 数据输入寄存器 04
Rx 基本地址=0x7D10_0000				
Tx 基本地址=0x7D50_0000				

表 13-6 DES/TDES 寄存器映射

地址	读/写	复位值	名称	描述
Rx-DES/3DES 寄存器映射 (RX 部分)				
基本地址+0x00	读/写	0x0000_0040	TDES_Rx_CTRL	DES/3DES Rx 控制和状态寄存器
基本地址+0x10	读/写	0x0000_0000	TDES_Rx_KEY1_0	DES/3DES 密钥输入寄存器 1_0
基本地址+0x14	读/写	0x0000_0000	TDES_Rx_KEY1_1	DES/3DES 密钥输入寄存器 1_1
基本地址+0x18	读/写	0x0000_0000	TDES_Rx_KEY2_0	DES/3DES 密钥输入寄存器 2_0
基本地址+0x1C	读/写	0x0000_0000	TDES_Rx_KEY2_1	DES/3DES 密钥输入寄存器 2_1
基本地址+0x20	读/写	0x0000_0000	TDES_Rx_KEY3_0	DES/3DES 密钥输入寄存器 3_0
基本地址+0x24	读/写	0x0000_0000	TDES_Rx_KEY3_1	DES/3DES 密钥输入寄存器 3_1
基本地址+0x40	读/写	0x0000_0000	TDES_Rx_INPUT_0	DES/3DES 数据输入寄存器 0
基本地址+0x44	读/写	0x0000_0000	TDES_Rx_INTPUT_1	DES/3DES 数据输入寄存器 1
基本地址+0x48	读	0x0000_0000	TDES_Rx_OUTPUT_0	DES/3DES Rx 数据输出寄存器 0
基本地址+0x4C	读	0x0000_0000	TDES_Rx_OUTPUT_1	DES/3DES Rx 数据输出寄存器 1
基本地址+0x50	读/写	0x0000_0000	TDES_Rx_IV_0	TDES Rx IV 输入寄存器 0
基本地址+0x54	读/写	0x0000_0000	TDES_Rx_IV_1	TDES Rx IV 输入寄存器 1
Tx-DES/3DES 寄存器映射 (TX 部分)				
基本地址+0x48	读	0x0000_0000	TDES_Tx_OUTPUT_0	DES/3DES Tx 数据输出寄存器 0
基本地址+0x4C	读	0x0000_0000	TDES_Tx_OUTPUT_1	DES/3DES Tx 数据输出寄存器 1
Rx 基本地址=0x7D20_0000				
Tx 基本地址=0x7D60_0000				

表 13-7 SHA-1/PRNG 寄存器映射

地址	读/写	复位值	名称	描述
Rx-SHA-1/PRNG 寄存器映射 (RX 部分)				
基本地址+0x00	读/写	0x0000_0000	HASH_CONTROL	Hash 引擎控制寄存器
基本地址+0x04	读/写	0x0000_0000	HASH_DATA	Hash 数据或 HMAC 密钥输入寄存器
基本地址+0x08	读/写	0x0000_0000	HASH_DATA	Hash 数据或 HMAC 密钥输入寄存器
...
基本地址+0x20	读/写	0x0000_0000	HASH_DATA	Hash 数据或 HMAC 密钥输入寄存器
基本地址+0x08	读/写	0x0000_0000	SEED_DATA_01	PRNG 种子数据[31:0]
基本地址+0x0C	读/写	0x0000_0000	SEED_DATA_02	PRNG 种子数据[63:32]
基本地址+0x10	读/写	0x0000_0000	SEED_DATA_03	PRNG 种子数据[95:4]
基本地址+0x14	读/写	0x0000_0000	SEED_DATA_04	PRNG 种子数据[127:96]
基本地址+0x18	读/写	0x0000_0000	SEED_DATA_05	PRNG 种子数据[159:128]
基本地址+0x1C	读/写	0x0000_0000	SEED_DATA_06	PRNG 种子数据[191:160]
基本地址+0x20	读/写	0x0000_0000	SEED_DATA_07	PRNG 种子数据[223:192]
基本地址+0x24	读/写	0x0000_0000	SEED_DATA_08	PRNG 种子数据[255:224]
基本地址+0x28	读/写	0x0000_0000	SEED_DATA_09	PRNG 种子数据[287:256]
基本地址+0x2C	读/写	0x0000_0000	SEED_DATA_10	PRNG 种子数据[319:288]
基本地址+0x30	读	0x0000_0010	HASH_STATUS	状态检测
基本地址+0x34	读	0x0000_0000	HASH_OUTPUT_01 PRNG_OUTPUT_01	HASH(PRNG)输出 (h0)
基本地址+0x38	读	0x0000_0000	HASH_OUTPUT_02 PRNG_OUTPUT_02	HASH(PRNG)输出 (h1)
基本地址+0x3C	读	0x0000_0000	HASH_OUTPUT_03 PRNG_OUTPUT_03	HASH(PRNG)输出 (h2)
基本地址+0x40	读	0x0000_0000	HASH_OUTPUT_04 PRNG_OUTPUT_04	HASH(PRNG)输出 (h3)
基本地址+0x44	读	0x0000_0000	HASH_OUTPUT_05 PRNG_OUTPUT_05	HASH(PRNG)输出 (h4)

基本地址+0x48	读	0x0000_0000	HASH_OUTPUT_06	PRNG 输出
基本地址+0x4C	读	0x0000_0000	HASH_OUTPUT_07	PRNG 输出
基本地址+0x50	读	0x0000_0000	HASH_OUTPUT_08	PRNG 输出
基本地址+0x54	读	0x0000_0000	HASH_OUTPUT_09	PRNG 输出
基本地址+0x58	读	0x0000_0000	HASH_OUTPUT_10	PRNG 输出
基本地址+0x5C	读	0x0000_0000	HASH_MIDOUT_01	HASH_MIDOUT[159:128]
基本地址+0x60	读	0x0000_0000	HASH_MIDOUT_02	HASH_MIDOUT[127: 96]
基本地址+0x64	读	0x0000_0000	HASH_MIDOUT_03	HASH_MIDOUT[95: 64]
基本地址+0x68	读	0x0000_0000	HASH_MIDOUT_04	HASH_MIDOUT[63: 32]
基本地址+0x6C	读	0x0000_0000	HASH_MIDOUT_05	HASH_MIDOUT[31: 0]
基本地址+0x70	写	0x0000_0000	HASH_IV_01	HASH 初始值 01
基本地址+0x74	写	0x0000_0000	HASH_IV_02	HASH 初始值 02
基本地址+0x78	写	0x0000_0000	HASH_IV_03	HASH 初始值 03
基本地址+0x7C	写	0x0000_0000	HASH_IV_04	HASH 初始值 04
基本地址+0x80	写	0x0000_0000	HASH_IV_05	HASH 初始值 01
基本地址+0x84	写	0x0000_0000	PRE_MSG_LENGT_1	HASH 长度[63:32]
基本地址+0x88	写	0x0000_0000	PRE_MSG_LENGT_2	HASH 长度[31:0]
Tx-SHA-1/PRNG 寄存器映射 (TX 部分)				
基本地址+0x34	读	0x0000_0000	HASH_OUTPUT_01 PRNG_OUTPUT_01	HASH (PRNG) 输出 (h0)
基本地址+0x38	读	0x0000_0000	HASH_OUTPUT_02 PRNG_OUTPUT_02	HASH (PRNG) 输出 (h1)
基本地址+0x3C	读	0x0000_0000	HASH_OUTPUT_03 PRNG_OUTPUT_03	HASH (PRNG) 输出 (h2)
基本地址+0x40	读	0x0000_0000	HASH_OUTPUT_04 PRNG_OUTPUT_04	HASH (PRNG) 输出 (h3)
基本地址+0x44	读	0x0000_0000	HASH_OUTPUT_05 PRNG_OUTPUT_05	HASH (PRNG) 输出 (h4)

基本地址+0x48	读	0x0000_0000	HASH_OUTPUT_06	PRNG 输出
基本地址+0x4C	读	0x0000_0000	HASH_OUTPUT_07	PRNG 输出
基本地址+0x50	读	0x0000_0000	HASH_OUTPUT_08	PRNG 输出
基本地址+0x54	读	0x0000_0000	HASH_OUTPUT_09	PRNG 输出
基本地址+0x58	读	0x0000_0000	HASH_OUTPUT_10	PRNG 输出
基本地址+0x5C	读	0x0000_0000	HASH_MIDOUT_01	HASH_MIDOUT[159:128]
基本地址+0x60	读	0x0000_0000	HASH_MIDOUT_02	HASH_MIDOUT[127: 96]
基本地址+0x64	读	0x0000_0000	HASH_MIDOUT_03	HASH_MIDOUT[95: 64]
基本地址+0x68	读	0x0000_0000	HASH_MIDOUT_04	HASH_MIDOUT[63: 32]
基本地址+0x6C	读	0x0000_0000	HASH_MIDOUT_05	HASH_MIDOUT[31: 0]
Rx 基本地址=0x7D20_0000				
Tx 基本地址=0x7D60_0000				

13.3 DMA 和中断控制模版

安全子系统 DMA 和中断寄存器

寄存器	地址	读/写	描述	复位值
Dnl_Cfg	0x7D00_0000	读/写	DMA 与中断配置控制寄存器和状态寄存器	0x0000_0000

Dnl_Cfg	位	描述	初始状态
WrPrivMismatch	[31]	SFR 的写访问权限不匹配状态位。如果设置' 1' , 发生 SFR 的写访问权限不匹配状态。	0b
RdPrivMismatch	[30]	SFR 的读访问权限不匹配状态位。如果设置' 1' , 发生 SFR 的读访问权限不匹配状态。	0b
Reserved	[29:23]	保留	0x00
SHA_intr_status	[22]	SHA-1/PRNG 中断状态和悬挂位。当读操作发生时, 此位被清除。	0b
DES_intr_status	[21]	DES/3DES 中断状态和悬挂位。当读操作发生时,	0b

		此位被清除。	
AES_intr_status	[20]	AES 中断状态和悬挂位。当读操作发生时，此位被清除。	0b
Reserved	[19:18]	保留	00b
FTX_intr_status	[17]	FIFO-TX 中断状态和悬挂位。当读操作发生时，此位被清除。	0b
FRX_intr_status	[16]	FIFO-RX 中断状态和悬挂位。当读操作发生时，此位被清除。	0b
Reserved	[15]	保留	00b
SHA_intr_En	[14]	结束操作以后 SHA-1/PRNG 中断状态使能。	0b
DES_intr_En	[13]	结束操作后 DES/3DES 中断状态使能	0b
AES_intr_En	[12]	结束操作后 AES 中断状态使能	0b
Reserved	[11: 10]	保留	00b
FTx_intr_En	[9]	结束操作以后 FIFO-Tx 中断状态使能。	0b
FRx_intr_En	[8]	结束操作后 FIFO-Rx 中断状态使能	0b
TxTrgLevel	[7:5]	Tx 部分触发电平设置 000=1 个字节 001=4 个字节 010=8 个字节 011= 12 个字节 100= 16 个字节 101=20 个字节 110=24 个字节 111=28 个字节	000b
TxDmaEnb	[4]	Tx 部分 DMS 使能位 (1: 使能)	0b
RxTrgLevel	[3:1]	Rx 部分触发电平设置 000=1 个字节 001=4 个字节 010=8 个字节 011= 12 个字节 100= 16 个字节 101=20 个字节 110=24 个字节 111=28 个字节	000b
RxDmaEnb	[0]	Tx 部分 DMS 使能位 (1: 使能)	0b

13.4 安全子系统 RX FIFO 模块

1. FIFO-RX 控制寄存器

寄存器	地址	读/写	描述	复位值
FRx_Ctrl	0x7D40_0000	读/写	FIFO-Rx 控制/状态寄存器（只可以读取 16 位 数据）	0x0420_0000

FRx_Ctrl	位	描述	初始状态
FRx_WrPrivError	[31]	如果写访问 FIFO-Rx 导致特权错误此位设置为 1。（主机和命令不允许访问 FIFO-Rx）	0b
FRx_RdPrivError	[30]	如果读访问 FIFO-Rx 导致特权错误此位设置为 1。（主机和命令不允许访问 FIFO-Rx）	0b
Reserved	[29;28]	保留	0b
FRx_Full	[27]	如果 FIFO-Rx 缓冲区满时设置此位为 1	0b
FRx_Empty	[26]	如果 FIFO-Rx 缓冲区空时设置此位为 1	0b
FRx_Done	[25]	如果 FIFO-Rx 完成向目标的数据 FRx_Mlen 字节的转换，设置此位为 1	0b
FRx_Running	[24]	如果 FIFO-Rx 完成目标的数据转换设置此位为 1，或者准备好目标输入缓冲等待时，设置此位为 1。当 FRx_Start 位复位为 0 时，设置此位为 1。	0b
FRx_Wd2Write	[23: 16]	可以写入 FIFO 存储器的字节数（FRx_WrBuf）	0x00
FRx_Wd2Read	[15: 8]	可以从 FIFO 存储器内读取的字节数（FRx_WrBuf）	0x00
FRx_Dest_Module	[7: 6]	选择目标模块 (00: AES, 01:DES/3DES, 10:SHA-1/PRNG, 11:保留)	0b
FRx_Host_Rd_En	[5]	主机可以从 FRx_Ctrl[31:16]和 FRx_MlenCnt 内读取数据。	0b
FRx_Host_Wr_En	[4]	主机可以 FRx_WrBuf 内写入数据。	0b
FRx_Sync_Tx	[3]	当 enabled, FIFO-Rx 等待 FIFO-Tx 在向目标转换数据之前找回远模块的输出数据。	0b

FRx-Reset	[2]	停止当前的 FIFO-Rx 转换器，复位 FSM 和所有寄存器。	0b
FRx_ERROR_En	[1]	当不允许通过 FRx_Ctrl[4]或[5]访问 FIFO-Rx 而主机试图进行访问时，通过 HRESP 端口可以进行错位提示。	0b
FRx_Start	[0]	FIFO-Rx 转换器开始位。当内部 FSM 开始转换数据时复位为 0。	

2. FIFO-RX 信息长度寄存器

寄存器	地址	读/写	描述	复位值
FRx_Mlen	0x7D40_0004	读/写	FIFO-Rx 信息长度寄存器	0x0000_0000

FRx_Mlen	位	描述	初始状态
FRx_Mlen	[31: 0]	当设置 FRx_Ctrl 寄存器的 FRx_Reset 位设置时，将复位值重设。	0x0000_0000

3. FIFO_RX 模块尺寸寄存器

寄存器	地址	读/写	描述	复位值
FRx_Blksz	0x7D40_0008	读/写	FIFO-Rx 保密数据模块尺寸寄存器	0x0000_0000

FRx_Blksz	位	描述	初始状态
Reserved	[31: 18]	保留	0x0000
LastValidByte	[17: 16]	向 SHA-1/PRNG 模块转换的最后字的最后有效字节。 SHA 文本字节的末尾 00: 第一字节 01: 第二字节 10 第三字节 11: 第四字节 当 FRx_Ctrl 寄存器的 FRx-Reset 被设置时，从新设置复位值。	00b
Blksz	[15:0]	目标模块的模块尺寸。FIFO-Rx 转换 FRx_Blksz 字，目标模块会被触发，并开始运行。通过 FRx_Ctrl 内的 FRx_Dest_Module 选择目标模块。 当 FRx_Ctrl 寄存器的 FRx-Reset 被设置时，从新设置复	0x0000

		位值。	
--	--	-----	--

4. FIFO_Rx 目标地址寄存器

寄存器	地址	读/写	描述	复位值
FRx_DestAddr	0x7D40_000C	读/写	FIFO-Rx 输入缓冲地址寄存器	0x0000_0000

SKEY_IDx	位	描述	初始状态
DestAddr	[31: 0]	目标输入缓冲的内部存储器地址。FIFO_Rx 可以向此地址转换数据。当设置 FRx_Ctrl 寄存器的 FRX_Reset 位设置时，将复位值重设。	0x0000_0000

5. FIFO_Rx 信息长度计数器

寄存器	地址	读/写	描述	复位值
FRx_MlenCnt	0x7D40_0010	读/写	FIFO-Rx 信息长度计数寄存器	0x0000_0000

FRx_MlenCnt	位	描述	初始状态
FRx_MlenCnt	[31: 0]	剩下用于转换的字的数目。	0x0000_0000

6. FIFO_Rx 写缓冲区

寄存器	地址	读/写	描述	复位值
FRx_WrBuf	0x7D40_0040 ~ 0x7D40_007C	写	FIFO-Rx 写缓冲区 (32 字) 注: 这个地址是 CPU 访问地址	0x0000_0000
	0x7D90_0040 ~ 0x7D90_007C		FIFO-Rx 写缓冲区 (32 字) 注: 这个地址是 SDMA1 的访问地址。可以在此地址用 SDMA1 进行存储器到 FRx_WrBuf 的数据转换。	

FRx_WrBuf	位	描述	初始状态
FRx_WrBuf	[31: 0]	FIFO-Rx 写缓冲区 (32x32 位)	0x0000_0000

注：向 FRx_WrBuf 进行的写访问操作使 FIFO_Rx 忽略已给的地址向 FIFO 存储器内写入数据。这就意味着在 0x0040 和 0x0080 之间的任何地址将会触发 FIFO 存储器的读操作。这个性能用突发写操作向 FIFO-Tx 产生程序。

13.5 安全子系统 TX FIFO 模块

1. FIFO-TX 控制寄存器

寄存器	地址	读/写	描述	复位值
FTx_Ctrl	0x7D80_0000	读/写	FIFO-Tx 控制/状态寄存器（只可以读取 16 位 数据）	0x0400_0000

Ftx_Ctrl	位	描述	初始状态
WrPrivError	[31]	如果写访问 FIFO-Tx 导致特权错误，此位设置为 1。（主机和命令不允许访问 FIFO-Tx）	0b
RdPrivError	[30]	如果读访问 FIFO-Tx 导致特权错误，此位设置为 1。（主机和命令不允许访问 FIFO-Tx）	0b
Reserved	[29:28]	保留	0b
Full	[27]	如果 FIFO-Tx 缓冲区满时设置此位为 1	0b
Empty	[26]	如果 FIFO-Tx 缓冲区空时设置此位为 1	0b
Done	[25]	如果 FIFO-Tx 完成向目标的数据 FTx_Mlen 字节的转换，设置此位为 1	0b
Running	[24]	如果 FIFO-Tx 完成目标的数据转换设置此位为 1，或者准备好目标的输入缓冲等待时，设置此位为 1。当 FTx_Start 位复位为 0 时，设置此位为 1。	0b
Wd2 Read	[23: 16]	可以从 FIFO 存储器内读取的字节数（FTx_RdBuf）	0x00
Wd2 Write	[15: 8]	可以写入 FIFO 存储器的字节数（FTx_RdBuf）	0x00
Src_Module	[7: 6]	选择目标模块 (00: AES, 01:DES/3DES, 10:SHA-1/PRNG, 11:保留)	0b

Host_Rd_En	[5]	主机可以从 FTx_Ctrl[31:16]和 FTx_MlenCnt 内读取数据。	0b
Host_Wr_En	[4]	主机可以 FTx_WrBuf 内写入数据。	0b
Reset	[2]	停止当前的 FIFO-Tx 转换器，复位 FSM 和所有寄存器。	0b
ERROR_En	[1]	当不允许通过 FTx_Ctrl[4]或[5]访问 FIFO-Tx 而主机试图进行访问时，通过 HRESP 端口可以进行错位提示。	0b
Start	[0]	FIFO-Tx 转换器开始位。当内部 FSM 开始转换数据时复位为 0。	

2. FIFO-TX 信息长度寄存器

寄存器	地址	读/写	描述	复位值
FTx_Mlen	0x7D80_0004	读/写	FIFO-Tx 信息长度寄存器	0x0000_0000

FTx_Mlen	位	描述	初始状态
FTx_Mlen	[31: 0]	当设置 FTx_Ctrl 寄存器的 FTx_Reset 位设置时，将复位值重设。	0x0000_0000

3. FIFO_TX 模块尺寸寄存器

寄存器	地址	读/写	描述	复位值
FTx_BlkSz	0x7D80_0008	读/写	FIFO-Tx 保密数据模块尺寸寄存器	0x0000_0000

FTx_BlkSz	位	描述	初始状态
BlkSz	[31:0]	目标模块的模块尺寸。FIFO-Tx 转换 FTx_BlkSz 字，目标模块会被触发，并开始运行。通过 FTx_Ctrl 内的 FTx_Dest_Module 选择目标模块。 当 FTx_Ctrl 寄存器的 FTx-Reset 被设置时，从新设置复位值。	0x0000

4. FIFO_Tx 源地址寄存器

寄存器	地址	读/写	描述	复位值
-----	----	-----	----	-----