SHA256 本质上是一种哈希函数，而哈希函数的功能与散列函数相同。
散列函数把消息或数据压缩成摘要，使得数据量变小，将数据的格式固定下来。该函数将数据打乱混合，重新创建一个叫做散列值（哈希值）的指纹。散列值通常用一个短的随机字母和数字组成的字符串来代表。
对于任意长度的消息，SHA256 都会产生一个 256bit 长的哈希值，称作消息摘要。
分为初始化数值（8 个哈希初值和 64 个哈希常量）

```
h0 := uint32(0x6a09e667)
h1 := uint32(0xbb67ae85)
h2 := uint32(0x3c6ef372)
h3 := uint32(0xa54ff53a)
h4 := uint32(0x510e527f)
h5 := uint32(0x9b05688c)
h6 := uint32(0x1f83d9ab)
h7 := uint32(0x5be0cd19)

k := [64]uint32{
        0x428a2f98, 0x71374491, 0xb5c0fbcf, 0xe9b5dba5, 0x3956c25b, 0x59
        0xd807aa98, 0x12835b01, 0x243185be, 0x550c7dc3, 0x72be5d74, 0x80
        0xe49b69c1, 0xefbe4786, 0x0fc19dc6, 0x240calcc, 0x2de92c6f, 0x4a
        0x983e5152, 0xa831c66d, 0xb00327c8, 0xbf597fc7, 0xc6e00bf3, 0xd5
        0x27b70a85, 0x2e1b2138, 0x4d2c6dfc, 0x53380d13, 0x650a7354, 0x76
        0xa2bfe8a1, 0xa81a664b, 0xc24b8b70, 0xc76c51a3, 0xd192e819, 0xd6
        0x19a4c116, 0x1e376c08, 0x2748774c, 0x34b0bcb5, 0x391c0cb3, 0x4e
        0x748f82ee, 0x78a5636f, 0x84c87814, 0x8cc70208, 0x90befffa, 0xa4
```

信息预处理（在被处理的报文后添加比特与固定长度，达到结构要求）

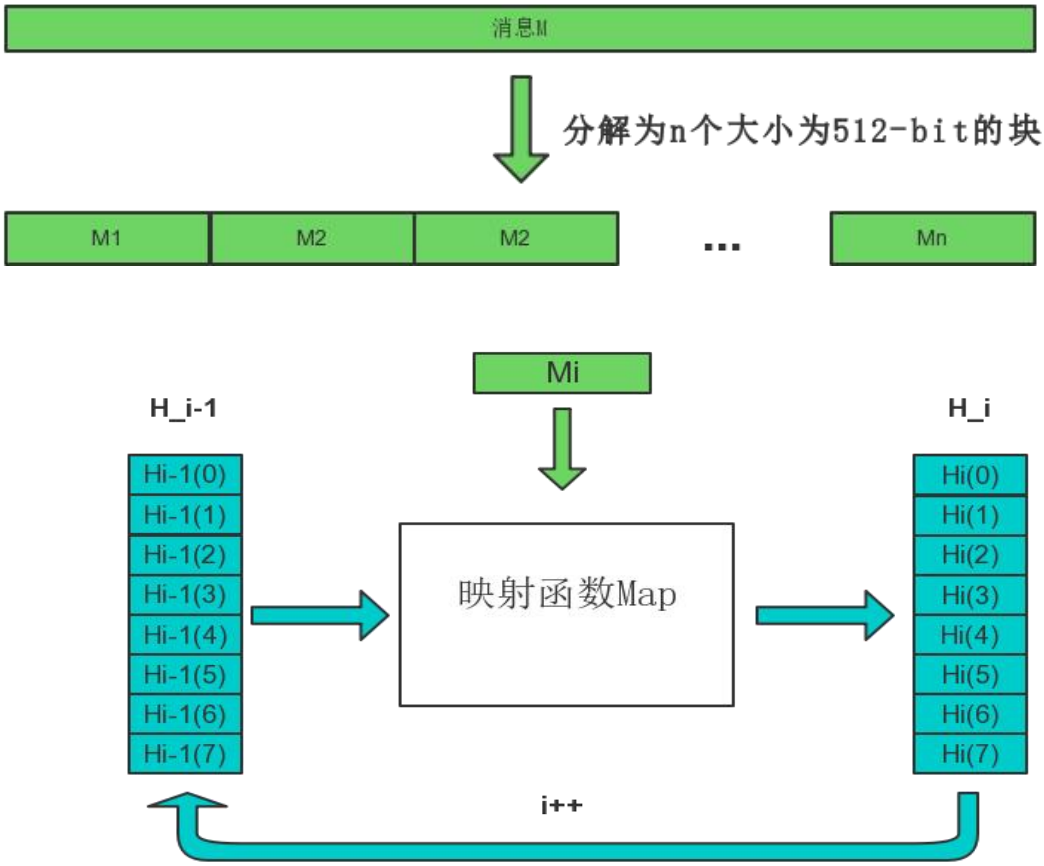
在报文末尾进行填充（附加填充比特），使报文长度在对 512 取模以后的余数是 448；再将原始数据的长度信息补到已经进行了填充操作的消息后面（附加长度值）

```
h0 := uint32(0x6a09e667)
h1 := uint32(0xbb67ae85)
h2 := uint32(0x3c6ef372)
h3 := uint32(0xa54ff53a)
h4 := uint32(0x510e527f)
h5 := uint32(0x9b05688c)
h6 := uint32(0x1f83d9ab)
h7 := uint32(0x5be0cd19)

k := [64]uint32{
        0x428a2f98, 0x71374491, 0xb5c0fbcf, 0xe9b5dba5, 0x3956c25b, 0x59
        0xd807aa98, 0x12835b01, 0x243185be, 0x550c7dc3, 0x72be5d74, 0x80
        0xe49b69c1, 0xefbe4786, 0x0fc19dc6, 0x240calcc, 0x2de92c6f, 0x4a
        0x983e5152, 0xa831c66d, 0xb00327c8, 0xbf597fc7, 0xc6e00bf3, 0xd5
        0x27b70a85, 0x2e1b2138, 0x4d2c6dfc, 0x53380d13, 0x650a7354, 0x76
        0xa2bfe8a1, 0xa81a664b, 0xc24b8b70, 0xc76c51a3, 0xd192e819, 0xd6
        0x19a4c116, 0x1e376c08, 0x2748774c, 0x34b0bcb5, 0x391c0cb3, 0x4e
        0x748f82ee, 0x78a5636f, 0x84c87814, 0x8cc70208, 0x90befffa, 0xa4
```

以及对分组循环加密

```
func iToB(i uint32) []byte {
        bs := make([]byte, 4)
        binary.BigEndian.PutUint32(bs, i)
        return bs
}
```


消息M
分解为n个大小为512-bit的块
M1
M2
M2
...
Mn
Mi
H_i-1
H_i
Hi-1(0)
Hi-1(1)
Hi-1(2)
Hi-1(3)
Hi-1(4)
Hi-1(5)
Hi-1(6)
Hi-1(7)
映射函数Map
Hi(0)
Hi(1)
Hi(2)
Hi(3)
Hi(4)
Hi(5)
Hi(6)
Hi(7)
i++