ADS1262IPW A large number of checksum errors were found

Note: One day was normal yesterday. Special exception for today and the day before yesterday. The phenomenon is as follows Code for reading the data: In principle, the fifth 14 of the SPI \_ RX \_ Buff is a checksum. However, the checksum of 0F 3A 27 08 is calculated as: 49. So after the last discovery, a lot of checksum errors were found: According to figures 1 and figure 2: SPI \_ RX \_ Buff display: The status is: 08 Reading books is: 0F 3A 27 08 The CRC is: 14 But the CRC result of the 0F 3A 27 08 calculation is49

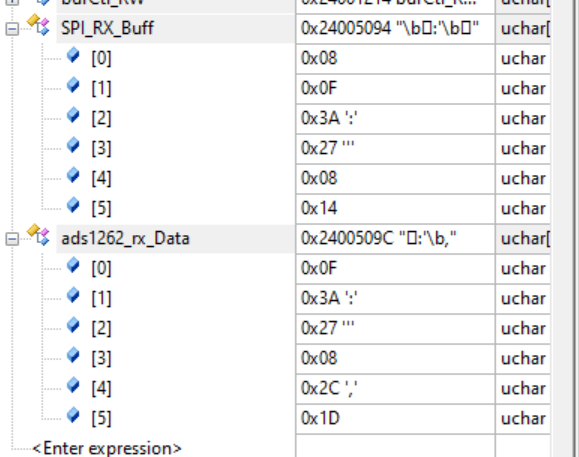


图1

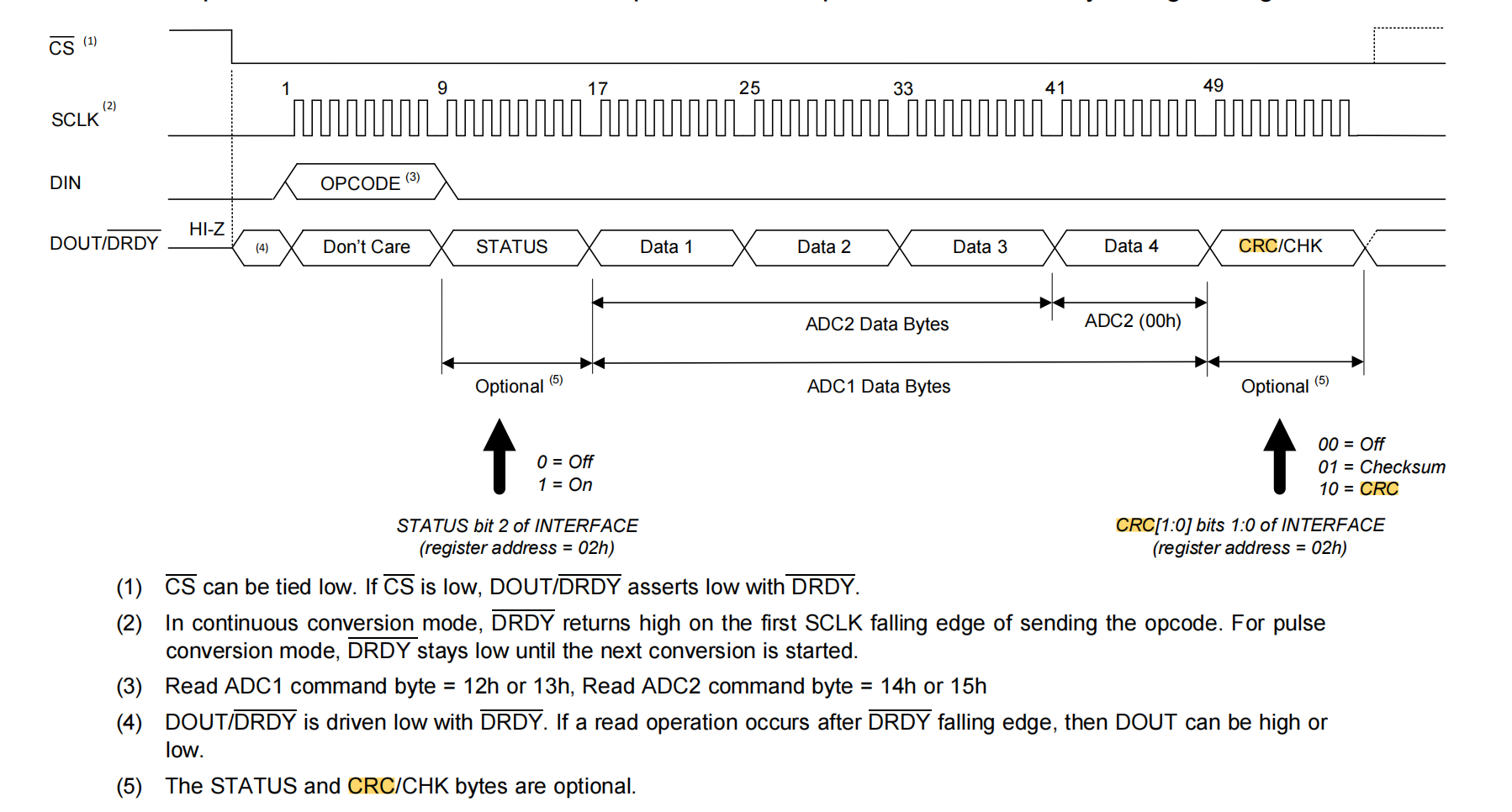


图2

Code for reading the data:

void ReadFrom\_ADS1262\_B(float \*mv)

{

volatile char ads1262\_rx\_Data[6];

volatile char SPI\_RX\_Buff[6];

int i;

float volt\_V=0;

Dev\_ADS1262\_C.SPI\_RX\_Buff\_Count = 0;

if(HAL\_GPIO\_ReadPin(DRDY2\_GPIO\_Port,DRDY2\_Pin)==GPIO\_PIN\_RESET)

{

Dev\_ADS1262\_C.SPI\_RX\_Buff\_Ptr=ads1262\_Read\_Data\_B(); Dev\_ADS1262\_C.Responsebyte = 1 ;

}

if(Dev\_ADS1262\_C.Responsebyte == 1)

{

for(i = 0; i <6; i++)

{

SPI\_RX\_Buff[Dev\_ADS1262\_C.SPI\_RX\_Buff\_Count++]= \*(Dev\_ADS1262\_C.SPI\_RX\_Buff\_Ptr + i);

}

Dev\_ADS1262\_C.Responsebyte = 0;

}

if(Dev\_ADS1262\_C.SPI\_RX\_Buff\_Count >= 5)

{

ads1262\_rx\_Data[0]= (unsigned char)SPI\_RX\_Buff[1]; // read 4 bytes adc count

ads1262\_rx\_Data[1]= (unsigned char)SPI\_RX\_Buff[2];

ads1262\_rx\_Data[2]= (unsigned char)SPI\_RX\_Buff[3];

ads1262\_rx\_Data[3]= (unsigned char)SPI\_RX\_Buff[4];

char crc = (unsigned char)SPI\_RX\_Buff[5];

char crc2 = CRC8((uint8\_t\*)ads1262\_rx\_Data,4);

if(Dev\_ADS1262\_C.crcCount==0){Dev\_ADS1262\_C.crcErrCnt=0;Dev\_ADS1262\_C.crcCount++;}

if(crc==crc2){

Dev\_ADS1262\_C.uads1262Count = (signed long) (((unsigned long)ads1262\_rx\_Data[0]<<24)|((unsigned long)ads1262\_rx\_Data[1]<<16)|(ads1262\_rx\_Data[2]<<8)|ads1262\_rx\_Data[3]);

Dev\_ADS1262\_C.sads1262Count = (signed long) (Dev\_ADS1262\_C.uads1262Count);

Dev\_ADS1262\_C.resolution = (double)((double)VREF/2147483648.0);

volt\_V= (Dev\_ADS1262\_C.resolution)\*(float)Dev\_ADS1262\_C.sads1262Count;

\*mv= volt\_V\*1000.0f;

}else Dev\_ADS1262\_C.crcErrCnt++;

Dev\_ADS1262\_C.crcCount++;

}

}

crc校验程序：

uint8\_t CRC8(uint8\_t \*data, uint16\_t length)

{

uint8\_t i;

uint8\_t crc = 0;

while(length--)

{

crc ^= \*data++;

for ( i = 0; i < 8; i++ )

{

if ( crc & 0x80 ) //1号

crc = (crc << 1) ^ 0x07; //2号

else

crc <<= 1;

}

}

return crc;

}

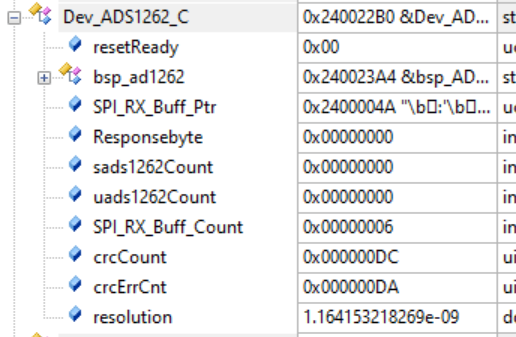
In our program, data with CRCj checksums errors are to be discarded. However, according to Figure 3, the total DC (220 data), and the wrong DA (218), is obviously problematic.

图3

According to Figure 1, the stutus status type: 08 is shown as the PGAout put low. According to Figure 4, our Mode2 register is worth 04, and this value feels like it should be correct. Not a true false alarm. ADS1262IPW Chip settings for:

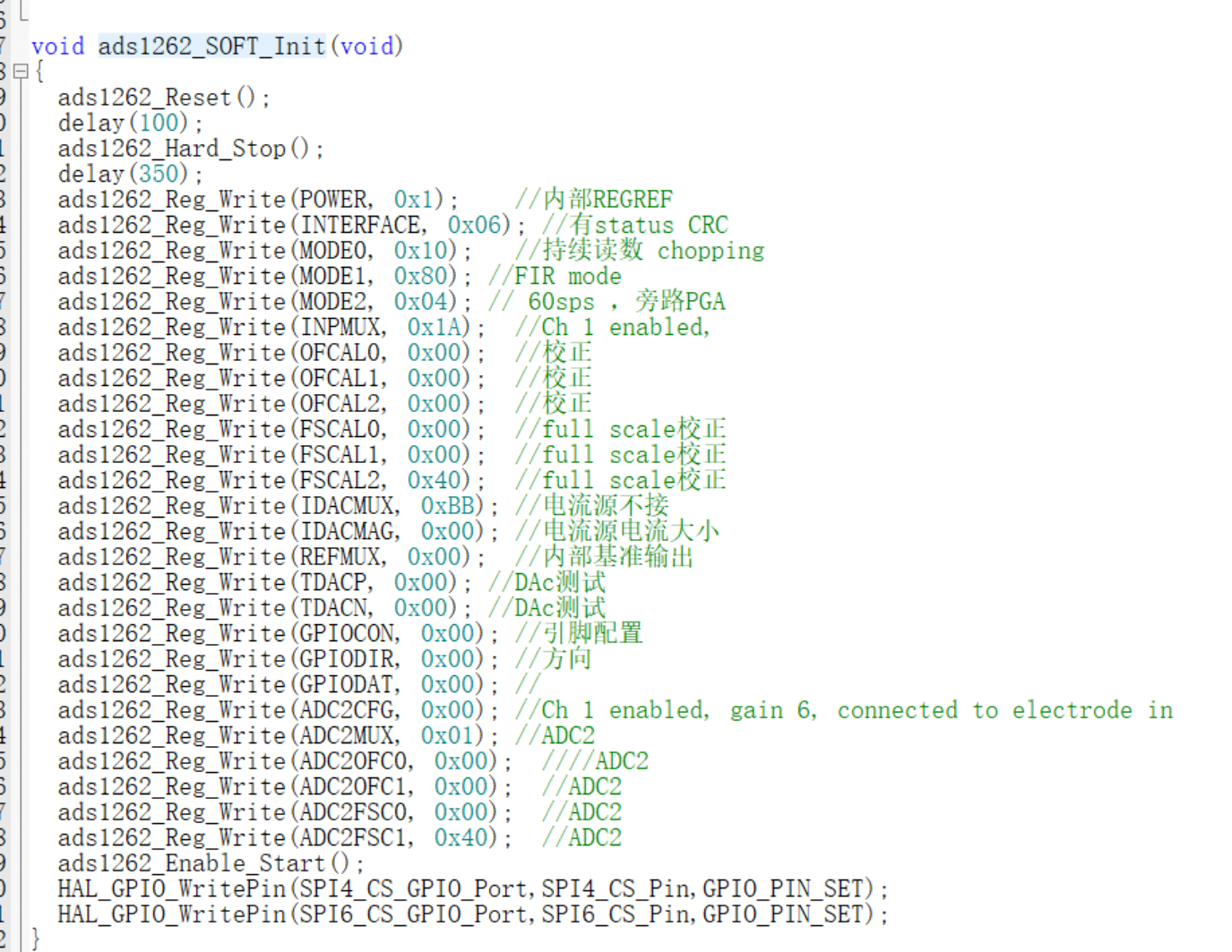


图4

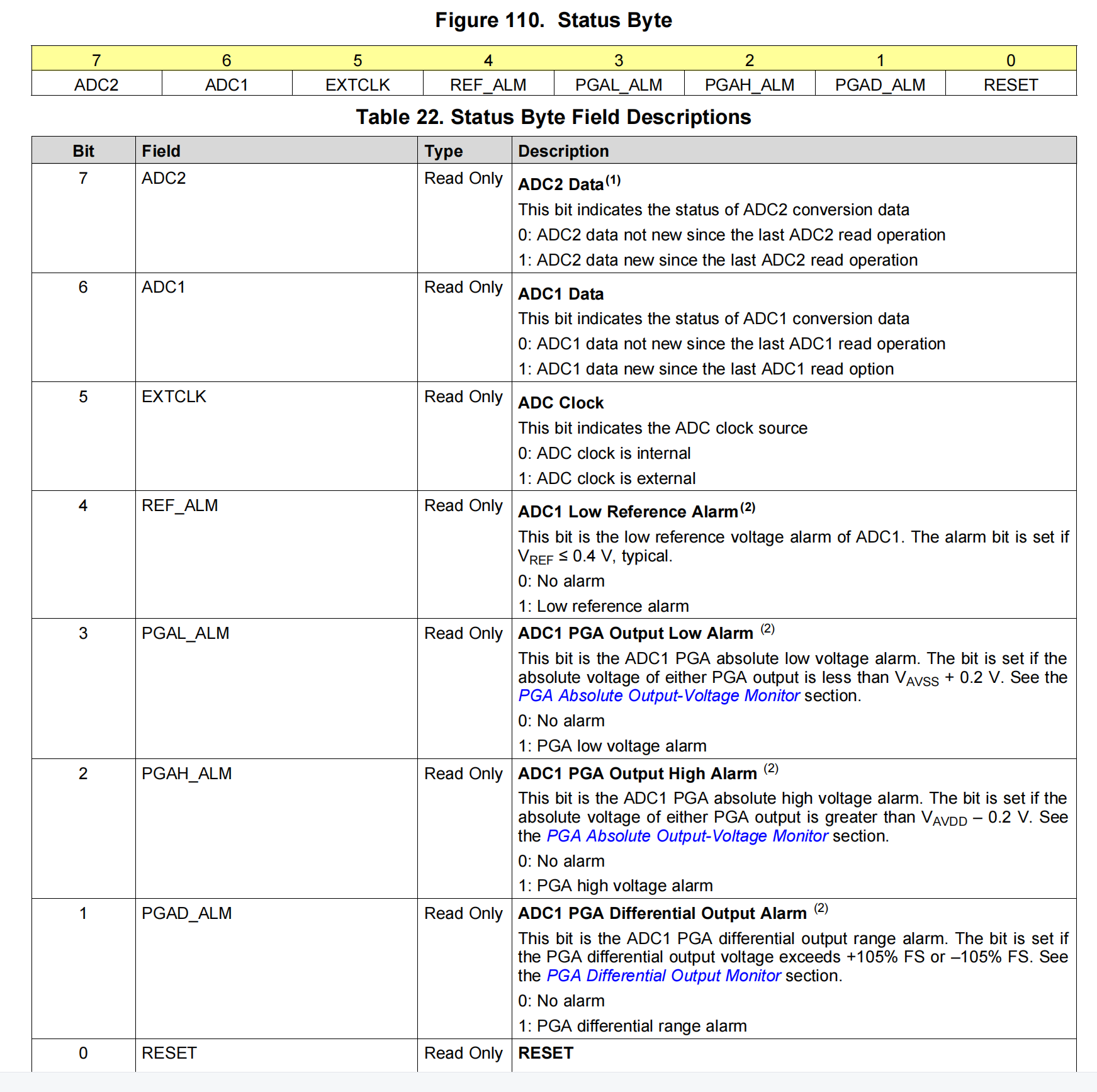


图5