

\* bit内の数字は例えです

		15bit	14bit	13bit	12bit	11bit	10bit	9bit	8bit	7bit	6bit	5bit	4bit	3bit	2bit	1bit	0bit
data[0]	下位bit									1	1	0	1	1	1	0	1
data[1]	上位bit								0	0	0	0	0	0	1	0	0

①上位ビットを8ビット左にシフトさせてOR(|)で足し合わせている。

		15bit	14bit	13bit	12bit	11bit	10bit	9bit	8bit	7bit	6bit	5bit	4bit	3bit	2bit	1bit	0bit
data[0]	下位bit									1	1	0	1	1	1	0	1
data[1]	上位bit	0	0	0	0	0	1	0	0								
raw		0	0	0	0	0	1	0	0	1	1	0	1	1	1	0	1

左に8bitシフト  
ORで足し合わせる

②チップによって初期の分解能が異なるらしいのでif分岐している。

古いDS18B20のチップは9bitの分解能なので12bitの計算式に当てはめるために3ビット左にシフトしている

③すいません、、、ちょっとわからんす

```
if (data[7] == 0x10) {
    // "count remain" gives full 12 bit resolution
    raw = (raw & 0xFFF0) + 12 - data[6];
}
```

システム予約の7byteが10hだったらrawをFFFF0hのアンドをとって12を足してシステム予約の6byteを引く？  
わかる人がいたら教えてほしいです。たぶん125(1111101)以上の値の時の処理？

④4byte目のデータから分解能分岐

		8bit	7bit	6bit	5bit	4bit	3bit	2bit	1bit	0bit
data[4]		0	1	1	1	1	1	1	1	1
60h		0	1	1	0	0	0	0	0	0
cfg		0	1	1	0	0	0	0	0	0

&で比較

例)データメモリ4byte

	分解能	8bit	7bit	6bit	5bit	4bit	3bit	2bit	1bit	0bit
cfg	9bit	0	0	0	0	0	0	0	0	0
	10bit	0	0	1	0	0	0	0	0	0
	11bit	0	1	0	0	0	0	0	0	0

AND-NOT(&~)で分解能ごとに不定部分を0とする

④摂氏・華氏計算

		符号		正の整数								少数					
		15bit	14bit	13bit	12bit	11bit	10bit	9bit	8bit	7bit	6bit	5bit	4bit	3bit	2bit	1bit	0bit
raw	int16_t	0	0	0	0	0	1	0	0	1	1	0	1	1	1	0	1
raw	float	1245															
celsius	/16	77.8125		摂氏													
fahrenheit	*1.8+32	172.0625		華氏													