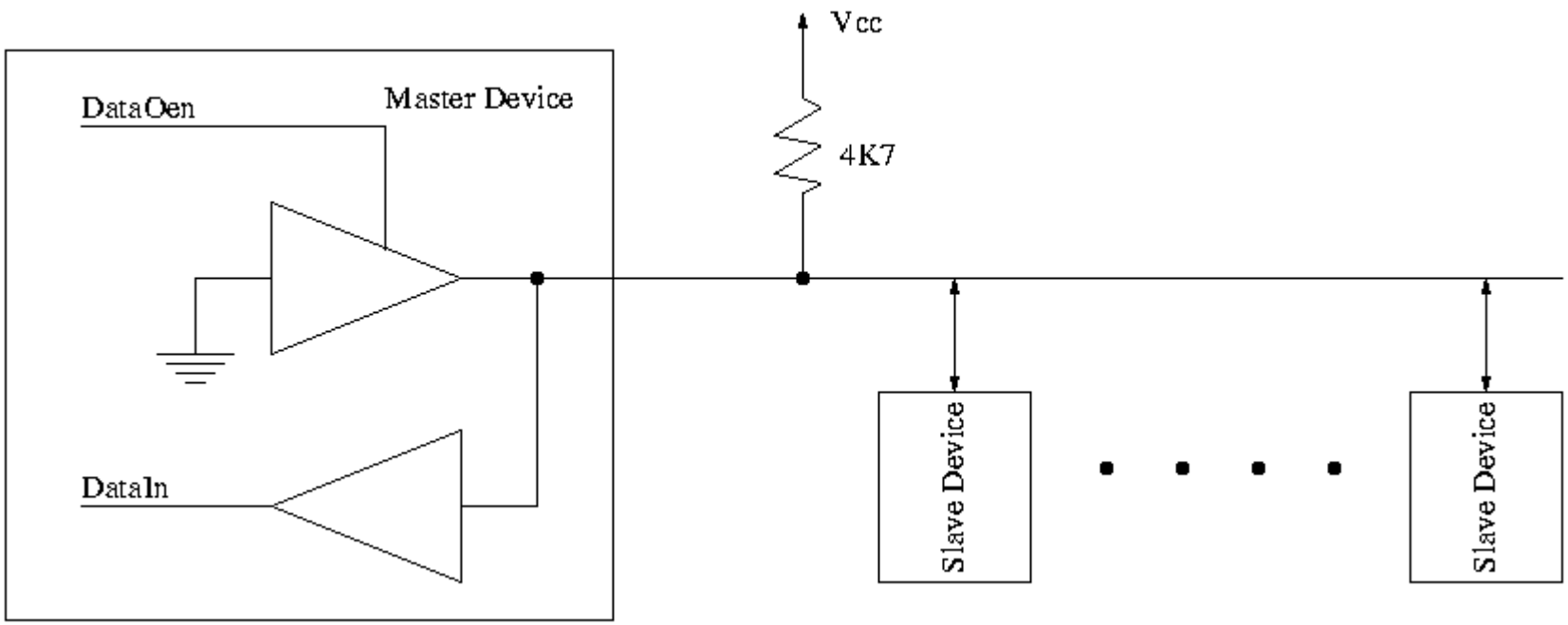
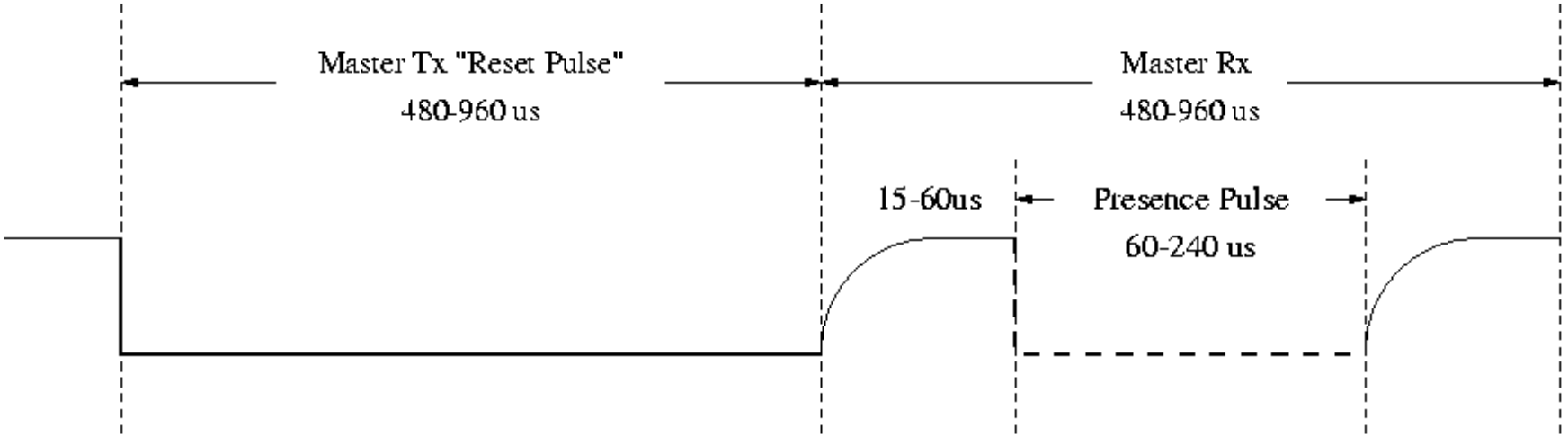
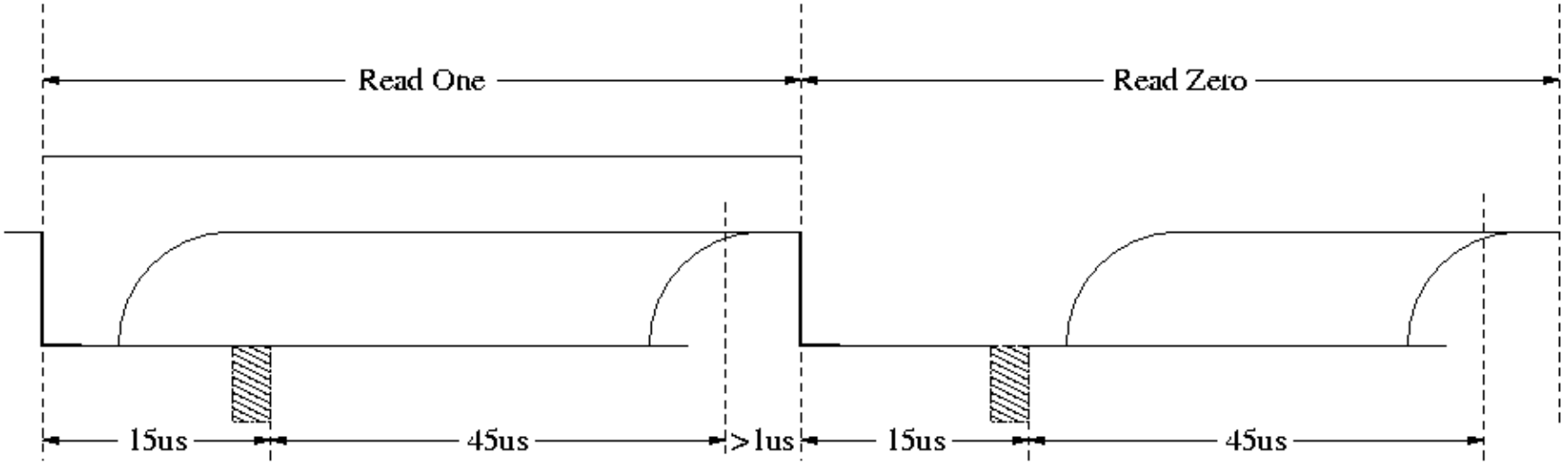
One Wire Bus


Vcc
4K7
DataOen
Master Device
DataIn
Slave Device
Slave Device

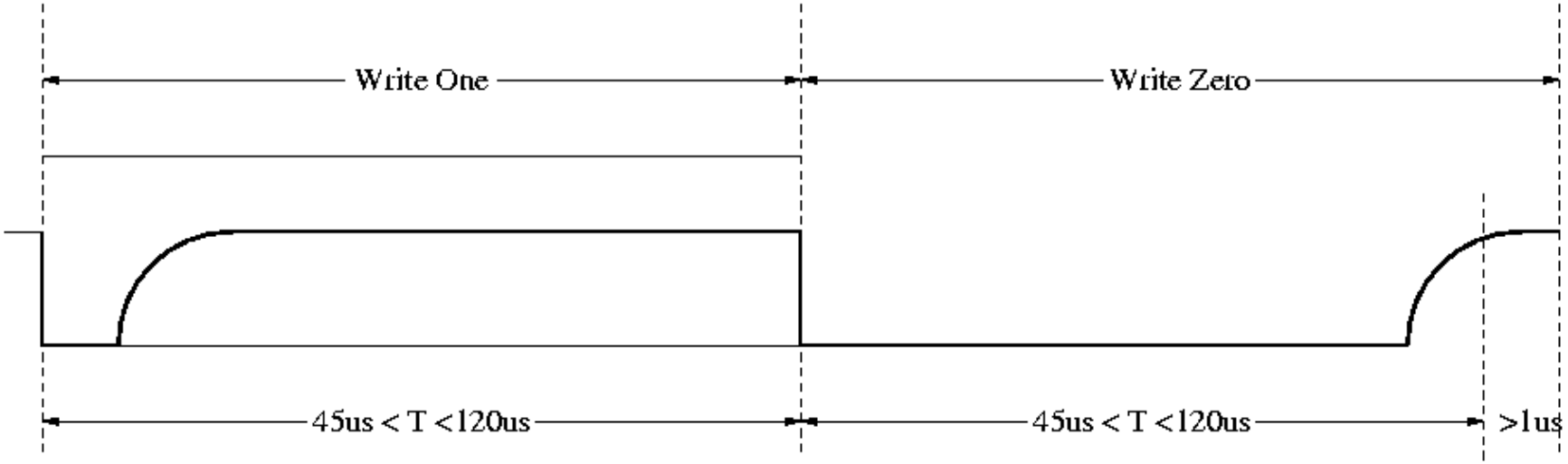
Reset Operation



Read Operation



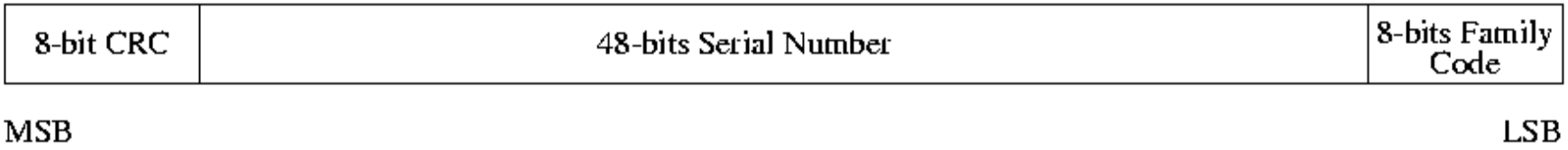
Write Operation



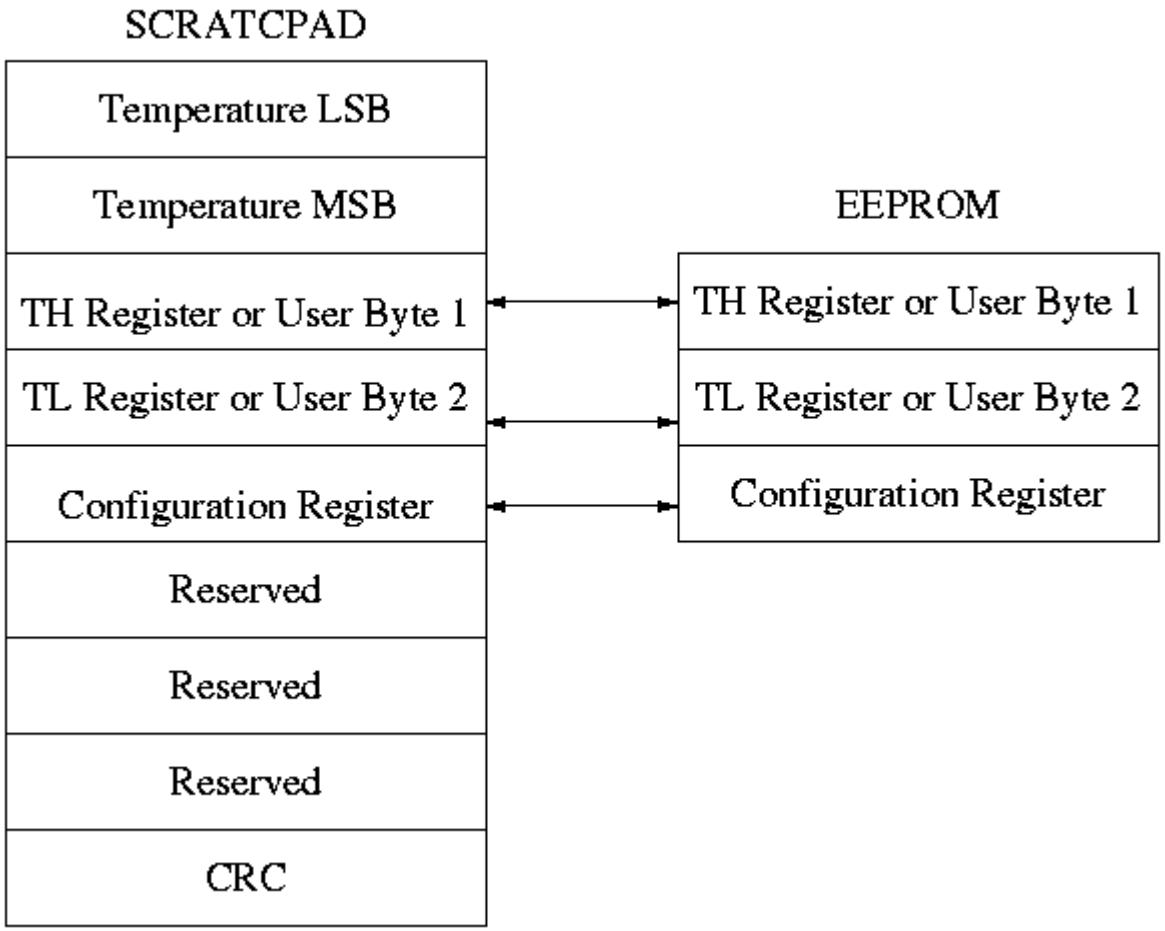
DS18B20 Digital Thermometer

- Unique 1-wire interface requires only one pin for communication
- Each device has a unique 64 bit serial code stored in an on-chip ROM
- Requires no extra components
- Power supply range is 3.0V to 5.5V
- Measures temperature from -55C to +125C .5C accurancy from -10C to +85C
- Thermometer resolution is user selectable from 9 to 12 bits
- Converts temperature to 12-bits digital word in 750 ms
- User- define nonvolative alarm setting

DS18B20 Lasered ROM Code



DS18B20 Memory Map

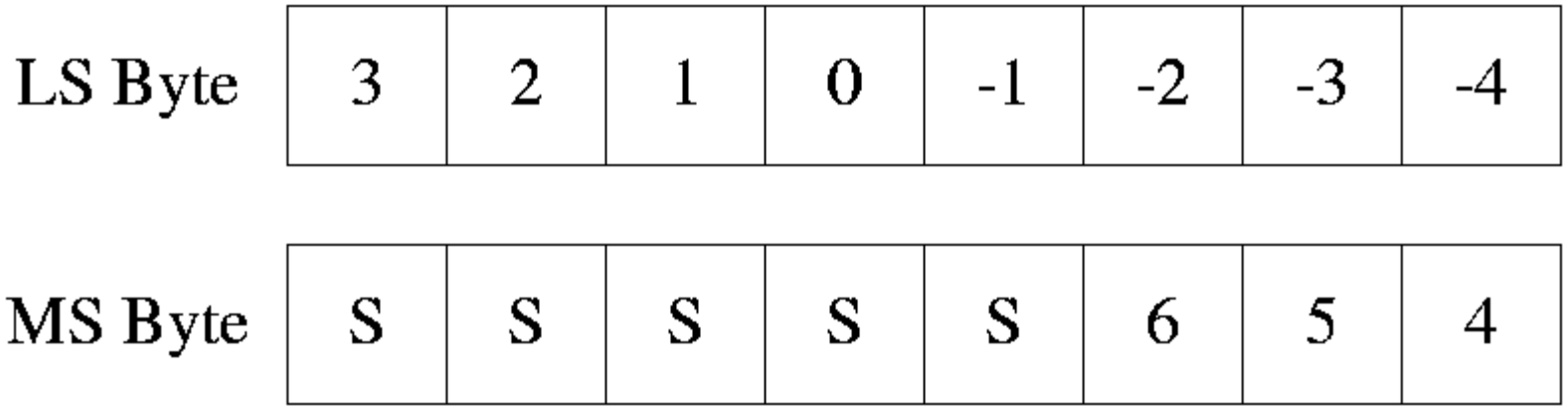


DS18B20 Configuration Register

0	R1	R0	1	1	1	1	1

R1	R0	Resolution	Max Convertion Time	
0	0	9-bit	93.75ms	tconv/8
0	1	10-bit	187.5ms	tconv/4
1	0	11-bit	375ms	tconv/2
1	1	12-bit	750ms	tconv

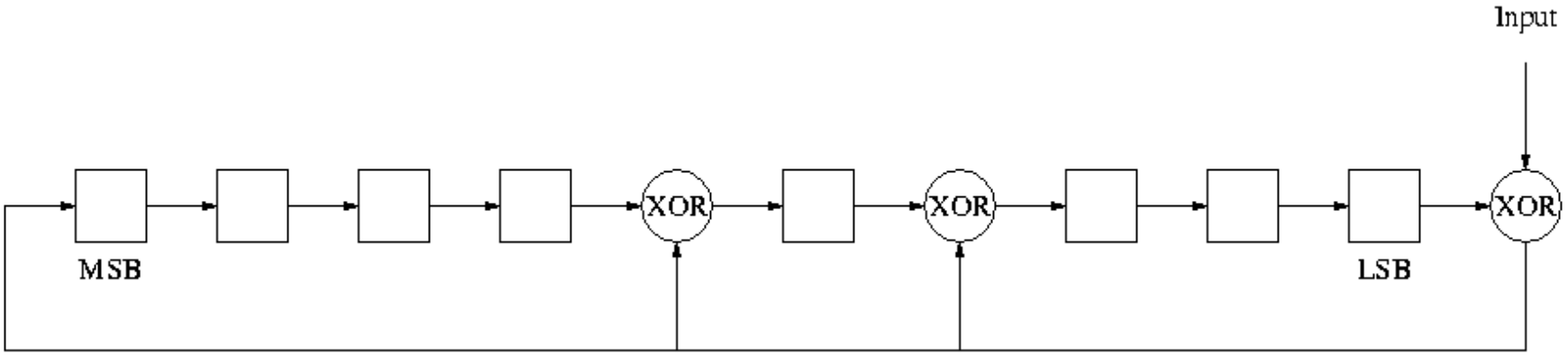
DS18B20 Temperature Register Format



DS18B20 TH and TL Registers Fomrat

S	6	5	4	3	2	1	0

CRC



DS18B20 Commands

- ROM Commands
  - Search ROM (0xF0)
  - Read ROM (0x33)
  - Match ROM (0x55)
  - Skip ROM (0xCC)
  - Alarm Search (0xEC)
- Function Commands
  - Convert (0x44)
  - Write Scratchpad (0xBE)
  - Read Scratchpad (0x4E)
  - Copy Scratchpad (0xB8)
  - Recall E^2 (0xB8)
  - Read Power Supply (0xB4)

Init

```
# define tRST   70
# define tSlot  35
# define tDrvz   4
# define tRead  35
  // Termometer Data Wire
sbit Wire = P3^0;
```

Reset Command

```
void RstCmd(void){
 int i;
    Wire = 0;
    for(i=0; i<tRST; i++);
    Wire = 1;
    for(i=0; i<tRST; i++);
}
```

Read Command

```
unsigned int RdCmd(void){
  unsigned char j, i, x;
  unsigned int Val, tmpVal;
     Val = 0;
     x = 0;
     for(j=0; j<16; j++){
        Wire = 0;
        Wire = 1;
        for(i=0; i<tDrvz; i++);
        tmpVal = 0x0001&Wire;
        for(i=0; i<tRead; i++);
        Val = Val|(tmpVal<<j);
     }
    return Val;
}
```

Write Command

```
void WrCmd(unsigned char Val){
  char j, i;
     for(j=0; j<8; j++){
        Wire = 0;
        if((Val&0x01)==1) Wire = 1;
        for (i=0; i<tSlot; i++);
        Wire = 1;
        Val = Val>>1;
     }
}
```

Read Temperature

```
unsigned int GetTemp(void){
  unsigned int Val;
  unsigned char i, j;
     RstCmd();              // Reset Thermometer
     WrCmd(0xCC);           // Skip Command
     WrCmd(0x44);           // Start Convertion
     j = 0;
     while(j!=7){           // Wait Convertion Completed
        Wire = 0;
        Wire = 1;
        for(i=0; i<tDrvz; i++);
        if(Wire==1) j++;
        else j =0;
        for(i=0; i<tRead; i++);
     }
     RstCmd();              // Reset Termometer
     WrCmd(0xCC);           // Skip Command
     WrCmd(0xBE);           // Read Command
     Val = RdCmd();         // Read Data
     RstCmd();              // Reset Termometer to stop Transmition
     return Val;
}
```