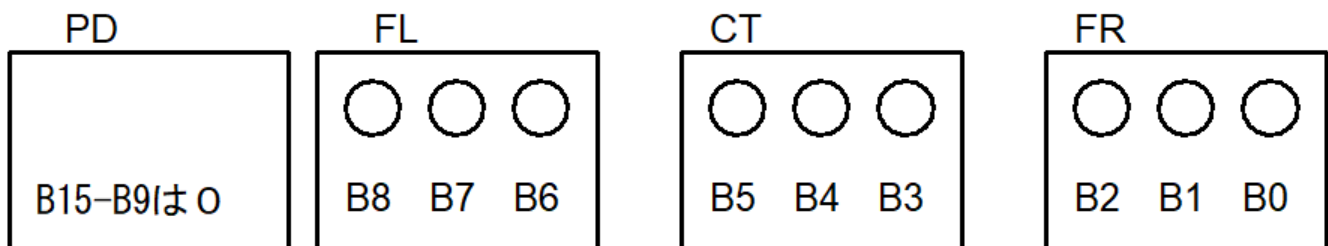


練習問題

- ロボットのフィールドセンサーが9個あり以下の図のように配置されている。各ブロックは3個のセンサーを3bitでアクセスしたい。プログラムからFL CT FR としてアクセスし、B0-B15を 符号なし16ビットとして一括アクセスしたい。
- 共用体・構造体の定義または、宣言をしなさい。

16bitアクセスはsensor.word 各ブロックは sensor.bit.FL

同様に.CT .FR のメンバーを持つものとする。B9ーB15は0でパディングされている。



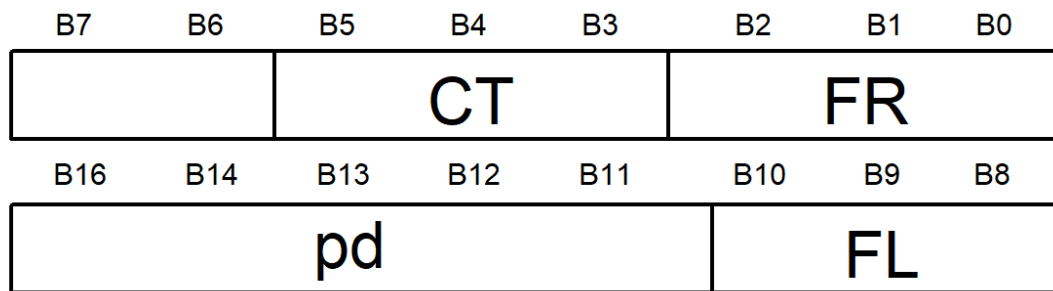
19

```
union Bit {  
    unsigned int word;  
    struct {  
        unsigned char FR:3;  
        unsigned char CT:3;  
        unsigned char FL:3;  
        unsigned char pd:7;  
    }  
} sensor;
```

pdにアクセスしないなら省略可
ビットポジションを明示するためあったほうが良い

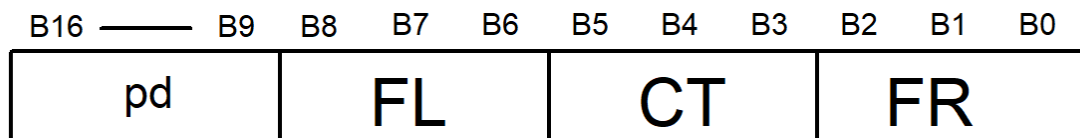
20

unsigned char で定義



B7, B6は自動的にパディングさ、アクセスできない

unsigned int で定義

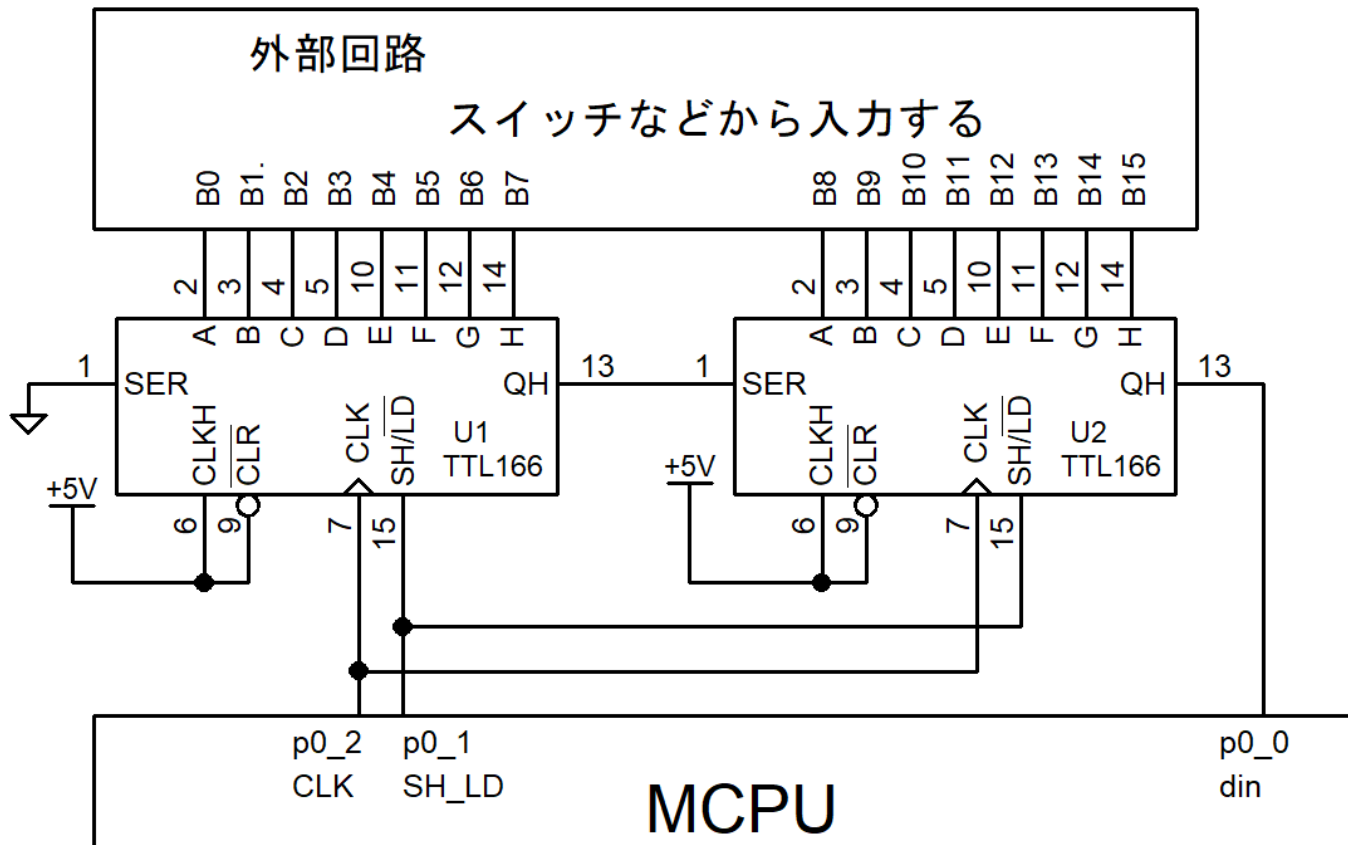


ビットアライメントが16bitになる

21

```
union   {  
    unsigned int word;  
    struct   {  
        unsigned int FR:3;  
        unsigned int CT:3;  
        unsigned int FL:3;  
        unsigned int pd:7;  
    } Bit;  
} sensor;
```

22



シフトレジスタは SH_LDが1の時シフト動作
0の時クロックの立ち上がりでA-Hを読む

23

応用例 プログラムの実装前に確認

• データのシフト

```
a=a<<1    // 変数aを1ビットシフトアップする。
a=a<<n    // 変数aをnビットシフトアップする。
同様に >>とするとシフトダウンできる
```

• 1bitのアクセス

```
SH_LD=0;
click();
SH_LD=1; volatile
```

```
sensor.word = sensor.word <<1 ;
sensor.bit.b0 = din;
click();           //シフトレジスタをシフトアップ
```

```
volatile void click(void){ //単に1パルス出力する
    CLK=1;
    CLK=0;
}
```

// volatile は最適化無効の宣言

24