

```

/**
 * 演習課題6: アナログ距離センサ(SHARP 2Y0A21, 測距モジュール), ブザー使用.
 * 障害物が適当な距離にある時, ブザーを鳴らす
 */

// defineはコンパイル時にマクロ変換される
#define SH_2Y0A21 2 // アナログ測距センサ接続ピン
#define BZ 9 // ブザー接続ピン

/**
 * 関数名: setup
 * 引数: なし
 * 処理: 各初期設定のため初回のみ実行
 * 返回值: なし
 */
void setup () {
    // put your setup code here, to run once:

    Serial.begin ( 9600 ); // シリアル通信の初期化
}

int count = 5; // 測定回数
float bf = 5000.0; // 基準周波数
float set_d = 20.0; // ブザーを鳴らす距離[cm]

/**
 * 関数名: loop
 * 引数: なし
 * 処理: 無限ループ. 距離センサからの出力を取得, 距離に応じてブザーを鳴らす
 * 返回值: なし
 */
void loop () {
    // put your main code here, to run repeatedly:

    float ain = 0; // A/D変換値, 平均値

    // 平均A/D変換値を出すため, 総和する
    for ( int i = 0; i < count; i++ ) {
        ain += analogRead ( SH_2Y0A21 );
    }
    ain = ain / count; // 平均A/D変換値

    // 障害物までの距離[cm]
    float dcm = ( 6787.0 / ( ain - 3.0 ) ) - 4.0;

    // 設定した距離より近くに障害物がある場合
    if ( dcm < set_d ) {

```

```
float f = bf / dcm; // 周波数決定

// tone関数が複数回実行されるのを防ぐため、音止める
noTone ( BZ );
tone ( BZ, f );      // 音鳴らす

} else {

    noTone ( BZ );      // 音消す
}

// 障害物までの距離を表示
Serial.print ( dcm );
Serial.println ( " cm" );

delay ( 200 );          // 遅延[ms]
}
```