```
#include <SPI.h>
#include <SdFat.h>
#include <SdFatUtil.h>
#include <SFEMP3Shield.h>
#include <MsTimer2.h>                  // タイマー割り込みを利用する為に必要なヘッダファイル

int MusicOn = 0;
int number = 1;
unsigned int set = 0;
unsigned long cnt1 = 0;
int stay = 2;
int stay2 = 1;
const int limit = 70;   //曲数+1

int Play;
int Stop;
int Up;
int Down;
int Pause;

int A0i;
int A1i;
int A2i;
int A3i;
int A4i;

//-----------------------
//定数の定義
//-----------------------
SdFat sd;                          //SdFat オブジェクトを宣言
SFEMP3Shield MP3player;          //MP3 プレーヤーオブジェクトを宣言

void Switch() {   //スイッチ対応
  Play = digitalRead(A1);
  Stop = digitalRead(A2);
  Up = digitalRead(A3);
  Down = digitalRead(A4);
  Pause = digitalRead(A5);
}
```

```
void MP3piay() {   //シールドの状態
  if (MusicOn == 1) {
    MP3player.begin();   //MP3 プレーヤー開始
    MusicOn = 2;
  }

  else if (MusicOn == 2) {
    MP3player.playTrack(number);       //track001.mp3 を再生 SD カードに入っている曲番号を入力
  }
  else {
    MP3player.end();   //MP3 プレーヤー終了
  }
}

void NumberLimit() {   //再生可能曲の範囲
  if (number >= limit) {
    number = 1;
  }
  if (number <= 0) {
    number = (limit - 1);
  }
}

void MainPlay() { //メイン
  //cnt1++;
  switch (set) {
    case 0:   //スタンバイ
      if (Pause == LOW) {
        number = random(1, limit);
      }
      if (Play == LOW) {
        MusicOn = 1;
        set = 1;
        cnt1 = 0;
        break;
      }
      break;

    case 1:   //再生
      if (Stop == LOW && cnt1 >= stay2) {
```

```
      cnt1 = 0;
      set = 2;
      break;
    }
    if (Up == LOW && cnt1 >= stay) {
      MusicOn = 0;
      // MP3player.end();
      cnt1 = 0;
      set = 4;
      break;
    }
    if (Down == LOW && cnt1 >= stay) {
      MusicOn = 0;
      //MP3player.end();
      cnt1 = 0;
      set = 5;
      break;
    }
    break;

  case 2:  //一時停止
    if (MusicOn == 2) {
      MP3player.pauseDataStream();
    }
    if (Stop == LOW && cnt1 >= stay2) {//20
      cnt1 = 0;
      set = 3;
      break;
    }
    if (Up == LOW && cnt1 >= stay2) {
      MusicOn = 0;
      //MP3player.end();
      cnt1 = 0;
      set = 4;
      break;
    }
    if (Down == LOW && cnt1 >= stay2) {
      MusicOn = 0;
      // MP3player.end();
      cnt1 = 0;
```

```
      set = 5;
      break;
    }
    break;

case 3:   //再開
    if (MusicOn == 2) {
      MP3player.resumeMusic();
    }
    if (Stop == LOW && cnt1 >= stay2) {
      cnt1 = 0;
      set = 2;
      break;
    }
    if (Up == LOW && cnt1 >= stay) {
      MusicOn = 0;
      //MP3player.end();
      cnt1 = 0;
      set = 4;
      break;
    }
    if (Down == LOW && cnt1 >= stay) {
      MusicOn = 0;
      //MP3player.end();
      cnt1 = 0;
      set = 5;
      break;
    }
    break;

case 4:   //曲を 1 つ上げる
    // MP3player.begin();
    number++;
    cnt1 = 0;
    MusicOn = 1;
    set = 1;
    break;

case 5:   //曲を 1 つ下げる
    //MP3player.begin();
```

```
        number--;
        cnt1 = 0;
        MusicOn = 1;
        set = 1;
        break;
    }
}

// 割り込み時に処理される関数
void flash() {
    static boolean output = HIGH;   // プログラム起動前に 1 回だけ HIGH(1)で初期化される

    digitalWrite(13, output);         // 13 番ピン(LED)に出力する(HIGH>ON LOW>OFF)
    output = !output;                  // 現在の output 内容を反転(HIGH→LOW/LOW→HIGH)させ output
にセットする
    cnt1++;
}

void setup() {
    Serial.begin(9600);                    //シリアル通信開始

    sd.begin(SD_SEL, SPI_HALF_SPEED);   //SdFat 開始
    pinMode(A1, INPUT);
    pinMode(A2, INPUT);
    pinMode(A3, INPUT);
    pinMode(A4, INPUT);
    pinMode(A5, INPUT_PULLUP);

    pinMode(13, OUTPUT);               // 13 番ピンを出力に設定(13 番ピンに LED が接続されている為)

    MsTimer2::set(1000, flash);        // 500ms 毎に flash( )割込み関数を呼び出す様に設定
    MsTimer2::start();                 // タイマー割り込み開始
}

void loop() {   //基本的に void を別に作る
    Switch();
    serial();
    NumberLimit();
    MP3piay();
    MainPlay();
```

```
  //      MP3player.playTrack(1);      //track001.mp3 を再生 SD カードに入っている曲番号を入力
}

void serial() {   //状態をシリアルで確認
  Switch();
  if (Play == LOW) {
    A0i = 1;
  }
  else {
    A0i = 0;
  }
  if (Stop == LOW) {
    A1i = 1;
  }
  else {
    A1i = 0;
  }
  if (Up == LOW) {
    A2i = 1;
  }
  else {
    A2i = 0;
  }
  if (Down == LOW) {
    A3i = 1;
  }
  else {
    A3i = 0;
  }
  if (Pause == LOW) {
    A4i = 1;
  }
  else {
    A4i = 0;
  }
  Serial.print("A0");
  Serial.print(" ");
  Serial.print(A0i);
  Serial.print(" ");
  Serial.print("A1");
```

```
  Serial.print(" ");
  Serial.print(A1i);
  Serial.print(" ");
  Serial.print("A2");
  Serial.print(" ");
  Serial.print(A2i);
  Serial.print(" ");
  Serial.print("A3");
  Serial.print(" ");
  Serial.print(A3i);
  Serial.print(" ");
  Serial.print("A4");
  Serial.print(" ");
  Serial.print(A4i);
  Serial.print(" ");
  Serial.print("set");
  Serial.print(" ");
  Serial.print(set);
  Serial.print(" ");
  Serial.print("number");
  Serial.print(" ");
  Serial.print(number);
  Serial.print(" ");
  Serial.print("cnt1");
  Serial.print(" ");
  Serial.println(cnt1);
}
```