

```
183 - ^/
184 void EXTI2 3 IRQHandler(void)
185 - {
186
       /* USER CODE BEGIN EXTI2 3 IRQn 0 */
187
       if(( EXTI->PR & 0xlul<<2)==(0xlul<<2)) {
         printf("PB.2 int.\n\r");
188
189
          songNumber = Song 5;
190
         EXTI->PR |=(0xlul<<2);
191
        } else if(( EXTI->PR & 0xlul<<3)==(0xlul<<3)) {</pre>
         printf("PB.3 int.\n\r");
192
193
         songNumber = Song 3;
         EXTI->PR \mid = (0xlu1 << 3);
194
195
196
```

```
stm32f0xx_it.c melody_control.c* melody_control.h startup_stm32f072xb.
main.c
 129
             break;
 130
 131
          case Song 4:
 132
            printf("Play song 4\r\n");
 133
            beatValue=500;
            for(int i=0; i < sizeof(note_basic); i++) {</pre>
 134
              toneValue = note basic[i];
              selectTone(toneValue);
136
 137
               Delay tim6(beatValue);
 138
               disablePWM();
139
               Delay tim6(soundStop);
 140
                 if(songNumber != Song_1)
 141
                   break;
 142
               enablePWM();
143
 144
 145
           case Song 5: // New Song: "萍聚"
 146
147
               printf("Play Song 5\n\r");
 148
               for (int i = 0; i < sizeof(note_hitorigoto); i++) {</pre>
 149
                  toneValue = note hitorigoto[i];
 150
                   if(toneValue==0){
151
                       disablePWM();
 152
                       beatValue = beat_hitorigoto[i];
 153
                       Delay_tim6(beatValue);
 154
                       enablePWM();
155
                 }else{
156
                       toneValue = toneValue + 7;
                       selectTone(toneValue);
 157
 158
                       beatValue = beat hitorigoto[i];
159
                       Delay tim6(beatValue);
 160
                       disablePWM();
 161
                       Delay_tim6(soundStop);
 162
                       if(songNumber != Song 5)
163
                         break:
164
                       enablePWM();
 165
 166
 167
             break;
 168
 169
          case Song off:
 170
          default:
 171
            disablePWM();
 172
            break;
        1//end of emitch
```

```
main.c stm32f0xc_it.c melody_control.c melody_control.h erretarget_io.c startup_stm32f072xb.s bluetooth.h
uint32_t ToneArr[]={1,764,681,607,573,510,454,405,382,340,303,286,255,227,202,191,170,152,143,128,114,101};
  // Happy Birthday Song const uint8_t note_happy_song[]={5,5,6,5,1+7,7,0,5,5,6,5,2+7,1+7,5,5,5+7,3+7,1+7,7,6,0,4+7,4+7,3+7,1+7,2+7,1+7};
  main.c stm32f0xx_it.c melody_control.c* melody_control.h startup_stm32f072xb.s bluetooth.h
   139
          enablePWM();
          Delay_tim6(3000);
   140
   141
          disablePWM():
       // Delay_tim6(1000);
   142
   143
   144
        /* USER CODE END 2 */
   145
        /* Infinite loop */
   146
         /* USER CODE BEGIN WHILE */
   147
   148
        while (1)
   149
   150
   151 🖨
          switch(songNumber) {
   152
           case Song_1:
   153
             playSong(Song_1);
   154
             break;
   155
           case Song_2:
   156
             playSong(Song_2);
   157
   158
             break;
   159
   160
            case Song 3:
             playSong(Song 3);
   161
   162
              break;
   163
   164
            case Song_4:
   165
             playSong(Song_4);
   166
              break;
   167
            case Song_5:
   168
   169
             playSong(Song_5);
   170
              break;
   171
            case Song_off:
   172
   173
            default:
   174
              playSong(Song_off);
              break;
   176
          }//end of switch
   177
   178
          /* USER CODE END WHILE */
   179
   180
          /* USER CODE BEGIN 3 */
   181
         /* USER CODE END 3 */
```

182 183