Play Music

Play Music
- Device connection
  - Hardware connection
  - Software connection
- Buzzer control
  - Control principle
  - Control pins
  - Code
- Experimental results

Control the passive buzzer on the Robduino expansion board to play a short piece of music "Twinkle Twinkle Little Star".

Device connection

Hardware connection

Use Type-B data cable to connect Arduino Uno and computer.

Software connection

Open the "Arduino IDE" software and select the model and serial port number corresponding to the development board.

Buzzer control

Buzzers are divided into two types: active buzzers and passive buzzers.

Active buzzers: require external working voltage to emit fixed frequency sound;

Passive buzzers: require external circuit to provide driving signal, and control the pitch and loudness of the sound by changing the frequency and amplitude of the signal.

Control principle

Arduino's tone library can be used to control the digital pin on the Arduino board to output a square wave signal of a specified frequency, thereby driving the buzzer connected to the pin to play a specified tone.

Control pins

Peripheral module	Arduino Uno
Buzzer	10

Code

Here we will only briefly introduce the code content. For detailed code, please refer to the corresponding code file, which is provided in the download area!

- Define BUZZER control pins

```
// 定义蜂鸣器控制引脚 Define BUZZER control pins
#define BUZZER_PIN 10
```

- Play tones of a specific frequency and time

```
/**
 * @brief 播放具体频率和时长的音调 Play tones of a specific frequency and time
 * @param frequency: 音调频率 Tone frequency
 * @param duration: 持续时间 time of duration
 * @retval 无 None
 */
void playMusic(int frequency, int duration) {
  tone(BUZZER_PIN, frequency);
  delay(duration);
  noTone(BUZZER_PIN);
}
```

- Initialization Code

```
void setup() {
  pinMode(BUZZER_PIN, OUTPUT);  // 设置蜂鸣器引脚输出模式 Set the buzzer pin output mode
}
```

- Looping code

```
void loop() {
  // Twinkle, twinkle, little star
  playMusic(523, 500);
  delay(100);
  playMusic(523, 500);
  delay(100);
  playMusic(587, 500);
  delay(100);
  playMusic(587, 500);
  delay(100);
  playMusic(659, 500);
  delay(100);
  playMusic(659, 500);
  delay(100);
  playMusic(587, 500);
  delay(1000);

  // How I wonder what you are
  playMusic(523, 500);
  delay(100);
  playMusic(493, 500);
  delay(100);
  playMusic(440, 500);
  delay(100);
  playMusic(440, 500);
  delay(100);
  playMusic(493, 500);
  delay(100);
  playMusic(493, 500);
  delay(1000);

  // Up above the world so high
  playMusic(523, 500);
  delay(100);
  playMusic(587, 500);
  delay(100);
  playMusic(659, 500);
  delay(100);
  playMusic(659, 500);
  delay(100);
  playMusic(587, 500);
  delay(100);
  playMusic(523, 500);
  delay(1000);
}
```

Experimental results

After compiling the program successfully, upload the code to the Arduino Uno development board.

After the program is started, the buzzer will play a short piece of music from the song "Twinkle Twinkle Little Star" in a loop!

```
The burning program cannot use other programs to occupy the serial port or an external serial communication module (for example: WiFi camera module), otherwise the program cannot be burned or an error message will be prompted!
```