United Power wrist watch SDK interface document

version	Modify Content	author	date
V1.0	init	YingChen Hou	2021-04-23
V1.1	1. add startup and self start interface 2. add automatic reconnection interface upon startup 3. Add message push settings interface	Tinggang Fu	2023-10-26

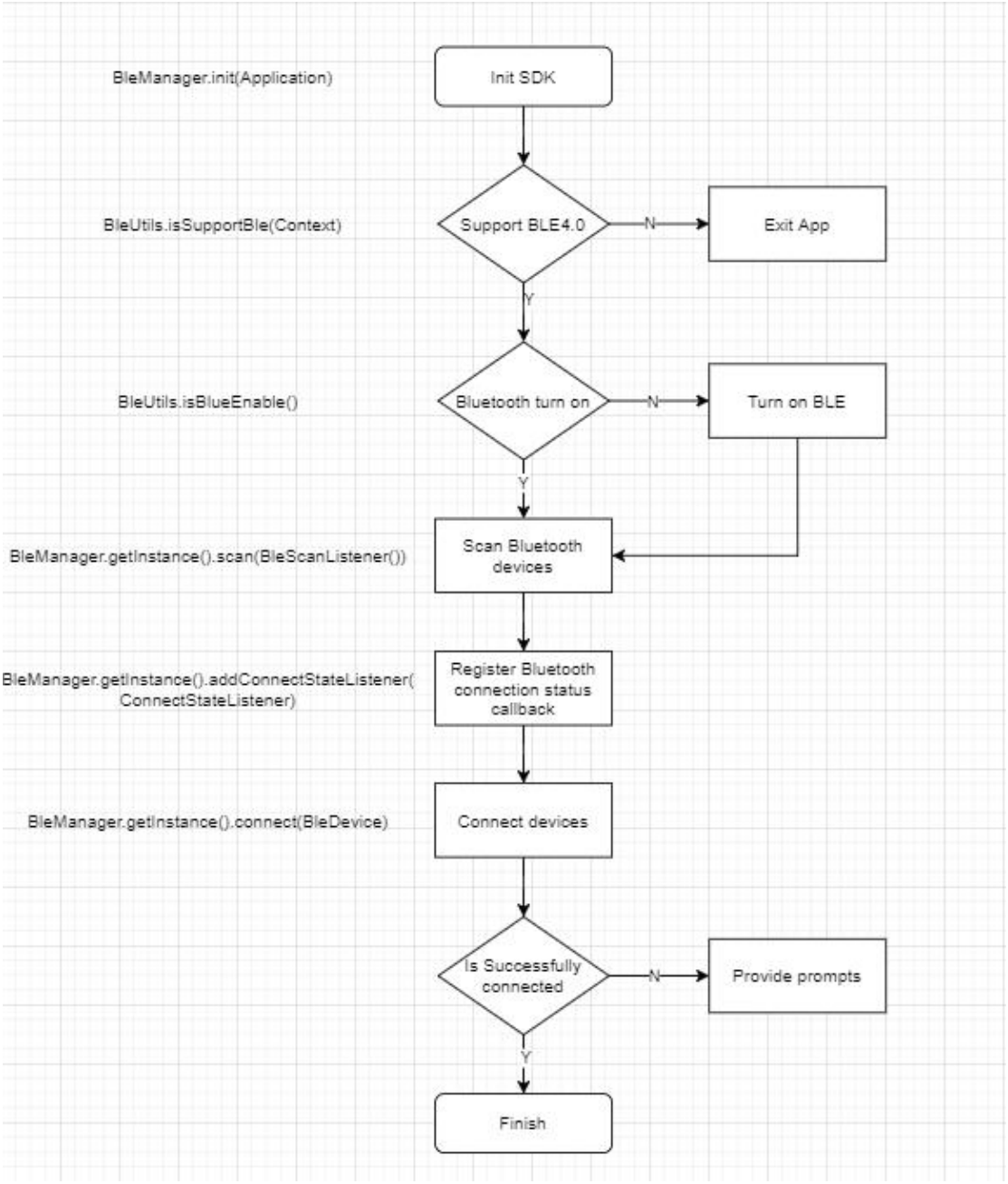
Contents

1. SDK Required permissions

<uses-permission android:name="android.permission.BLUETOOTH" />
<uses-permission android:name="android.permission.BLUETOOTH_ADMIN" />
<uses-permission android:name="android.permission.INTERNET" />
Note: Systems after Android 6.0 require adding location permissions
<uses-permissionandroid:name="android.permission.ACCESS_COARSE_LOCATION" />

Front desk service permissions:<uses-permission android:name="android.permission.FOREGROUND_SERVICE" />

2. Startup flowchart



3. Specific classes and their corresponding API method interfaces

3.1 BleManager SDK Operation management class

method	public static void init(Application app);
return value	
parameter description	app: Application
description	SDK Initialization method, it is recommended to perform initialization operations in the Application.
method	public static BleManager getInstance()
return value	BleManager obtain an instance of this class
parameter description	
description	
method	public void scan(BleScanListener callback);
return value	
parameter description	BleScanListener: Scan listening
description	To start scanning devices, Bluetooth permission is required, and location permission is also required for devices above 6.0. At the same time, GPS is enabled, After 7.0, it is not allowed to scan and stop more than 5 times within 30 seconds, otherwise the device will not be scanned.
method	public void stopScan();
return value	

parameter description	
description	Stop scanning devices
method	public BluetoothGatt connect(String mac);
return value	BluetoothGatt
parameter description	Device bluetooth Mac
description	Connect device
method	public BluetoothGatt connect(BleDevice bleDevice);
return value	BluetoothGatt
parameter description	BleDevice scanned Bluetooth devices
description	Connect device
method	public void addConnectStateListener(ConnectStateListener listener);
return value	
parameter description	ConnectStateListener connection status listening
description	Add ConnectStateListener
method	public void removeConnectStateListener(ConnectStateListener listener);
return value	
parameter description	ConnectStateListener connection status listening
description	Remove ConnectStateListener
method	public boolean isBlueEnable();

return value	boolean
parameter description	
description	Get whether Bluetooth is currently enabled
method	public void enableBle();
return value	
parameter description	
description	Enable Bluetooth
method	public boolean isConnect();
return value	boolean
parameter description	
description	Is the device connected
method	public boolean isScanning();
return value	boolean
parameter description	
description	Is scanning devices
method	public boolean autoStart();
return value	
parameter description	
description	Turn on the app's startup function.
note	It is necessary to activate the self start permission and notification bar

	permission accordingly.
method	public void setAutoReconnect(boolean autoReconnect)
return value	
parameter description	autoReconnect: is need to set automatic reconnection? Default：false
description	Set automatic reconnection.
method	public void setAppPush(boolean isPush)
return value	
parameter description	isPush: Enable reminder message push
description	Set message push.
note	It is necessary to set corresponding settings to keep the app background running and turn off power saving mode. Prevent the system from mistakenly killing processes and causing notification message push failures.

3.2 BleDataProcessing data processing class

method	public static BleDataProcessing getInstance()
return value	BleDataProcessing obtain an instance of this class
parameter description	
description	BleDataProcessing instance.
method	public void setOnAlarmClockChangeListener(OnAlarmClockChangeListener listener)
return	

value	
parameter description	
description	Add alarm information to change listening.
method	public void setOnSettingInfoChangeListener(OnSettingInfoChangeListener listener)
return value	
parameter description	
description	Add setting information to change listening.
method	public void setOnRemoteCameraListener(OnRemoteCameraListener onRemoteCameraListener)
return value	
parameter description	
description	Add remote camera monitoring.
method	public void setOnFindPhoneListener(OnFindPhoneListener onFindPhoneListener)
return value	
parameter description	
description	Add mobile phone monitoring.
method	public void setOnRemoteCameraListener(OnRemoteCameraListener onRemoteCameraListener)

return value	
parameter description	
description	Add remote camera monitoring.
method	public void setOnMusicControlListener(OnMusicControlListener onMusicControlListener)
return value	
parameter description	
description	Add a bracelet to control music playback and monitoring.
method	public void setOnMeasureExitListener(OnMeasureExitListener onMeasureExitListener)
return value	
parameter description	
description	Add measurement to exit listening.
method	public void setOnDataChangeListener(OnDataChangeListener onDataChangeListener)
return value	
parameter description	
description	Add data upload monitoring.

method	public void setSyncStateListener(SyncStateListener syncStateListener)
return value	
parameter description	
description	Add data transmission status monitoring.
method	public void setOnSettingResultListener(OnSettingResultListener onSettingResultListener)
return value	
parameter description	
description	Add bracelet settings to monitor changes.
method	public void setOnReadBatteryListener(OnReadBatteryListener onReadBatteryListener)
return value	
parameter description	
description	Add reading bracelet battery monitoring.
method	public void setOnBindDeviceListener(OnBindDeviceListener onBindDeviceListener)
return value	
parameter description	

description	Add binding status monitoring.

3.3 BleScanListener device scan callback listening

method	void onScanStarted(boolean success);
return value	
parameter description	success：Is scanning enabled successfully
description	Enable Bluetooth scanning function callback
method	void onScanFinished();
return value	
parameter description	
description	Callback after Bluetooth scanning ends
method	void onLeScan(BleDevice bleDevice);
return value	
parameter description	BleDevice: The scanned Bluetooth device can obtain the Bluetooth name, Bluetooth address, and broadcast content.
description	When Bluetooth scans for device callbacks, it should be noted that the same device may have multiple callbacks, which requires developers to perform filtering processing.

3.4 ConnectStateListener Device connection status callback monitoring

method	void onStartConnect();
return value	
parameter description	

description	Start connecting device callback
method	void onConnectSuccess();
return value	
parameter description	
description	Callback for successful device connection
method	void onConnectFail(BleException exception);
return value	
parameter description	You can obtain the code and information of the connection failure from the exception.
description	Callback for device connection failure.
method	void onDisConnected();
return value	
parameter description	
description	Callback for device disconnection

3.5 OnBindDeviceListener device binding callback

method	void onBind(boolean isFirst);
return value	
parameter description	Is it the first time the device is bound? If it is the first time, it is recommended to clear the data of the day to avoid discrepancies with the device data.
description	Callback for device consent binding
method	void onRefuse();
return value	

parameter description	
description	Device rejected callback for binding.
method	void onUnbindSuccess();
return value	
parameter description	
description	Device unbinding successful callback

3.6 OnDataChangeListener Device data reporting and monitoring

method	void onDayStepChange(DayStepBean dayStepBean);
return value	
parameter description	Daily step data, which is calculated hourly and can be obtained based on the time in the HourDataBean.
description	Daily step data upload callback
method	void onHRChange(List<HeartRateBean> hrList);
return value	
parameter description	A list of heart rate data, each with time and heart rate values.
description	Heart rate data upload callback.
method	void onBPChange(List<BloodPressureBean> bpList);
return value	
parameter description	Blood pressure data list, each data will have time and high and low pressure values.
description	Blood pressure data upload callback.

method	void onSleepChange(SleepInfoBean sleepInfoBean);
return value	
parameter description	
description	Sleep data upload callback.
method	void onRealTimeStepChange(int step, int distance, int calorie);
return value	
parameter description	Step:Real time step counting data is uploaded on the same day, distance: distance in meters , calorie:calories in calories.
description	Real time step data upload and callback for the day.
method	void onBOChange(List<BloodOxygenBean> boList);
return value	
parameter description	Blood oxygen data list, each data will have time and blood oxygen value.
description	Blood oxygen data upload callback.

3.7 OnAlarmClockChangeListener Alarm clock data Listener

method	void onChange(List<AlarmClockBean> infoList);
return value	
parameter description	
description	The device returns the saved alarm clock.

3.8 OnFindPhoneListener device searching for mobile phone monitoring

method	void onFindPhone();

return value	
parameter description	
description	The device sends instructions to find the phone, and the app can control the phone's ringing to remind users of the location of the phone.
method	void onFindPhoneEnd();
return value	
parameter description	
description	The device ends searching for a phone.

3.9 OnMeasureExitListener device exit measurement Listener

method	void onHRExit();
return value	
parameter description	
description	The device has exited heart rate measurement.
method	void onBPExit();
return value	
parameter description	
description	The device has exited blood pressure measurement.
method	void onTemperatureExit();
return value	
parameter description	

description	The device has exited temperature measurement.
method	void onBOExit();
return value	
parameter description	
description	The device has exited blood oxygen measurement.

3.10 OnMusicControlListener device control, phone playback, music Listener

method	void onMusicPlay();
return value	
parameter description	
description	The device sends a start playing command, and the app controls the player to start playing.
method	void onMusicPause();
return value	
parameter description	
description	The device sends a pause playback command, and the app controls the player to pause playback.
method	void onMusicLast();
return value	
parameter description	
description	The device sends a command to play the previous song, and the app controls the player to play the previous song.

method	void onMusicNext();
return value	
parameter description	
description	The device sends a command to play the next song, and the app controls the player to play the next song.

3.11 OnReadBatteryListener Read battery Listener

method	void onReadBattery(int battery);
return value	
parameter description	battery: Device battery percentage。
description	Read battery percentage

3.12 OnRemoteCameraListener Remote camera command Listener

method	void onOpenCamera();
return value	
parameter description	
description	The remote photo taking behavior initiated by the device side, and the app opens the photo taking function.
method	void onTakePicture();
return value	
parameter description	
description	The device sends a photo command, and the app responds and takes a photo.

method	void onCloseCamera();
return value	
parameter description	
description	The device sends a command to turn off the camera, and the app turns off the photography function.

3.13 OnSettingInfoChangeListener Monitoring for changes in device settings information

method	onSedentaryInfoChange(boolean enable, boolean isLunchBreakEnable, int step, int sedentaryTime, int startTime, int endTime, Int repeat);
return value	
parameter description	enable: Is the function enabled isLunchBreakEnable: Is the lunch break uninterrupted? The lunch break is from 12:00 to 14:00 step: Threshold Steps for Long Sittings edentaryTime: Sedentary time startTime: Start time endTime: End time repeat: The recurrence period, from low to high, represents Monday to Sunday, with 1 being a repeat for that day and 0 not repeating.
description	Long sitting reminder setting information callback
method	void onNotifySettingChange(boolean isTelNotify, boolean isSmsNotify,

	boolean isWechatNotify, boolean isQQNotify, boolean isFacebookNotify, boolean isTwitterNotify, boolean isSkypeNotify, boolean isLineNotify, boolean isWhatsappNotify, boolean isKakaoTalkNotify, boolean isInstagramNotify);
return value	
parameter description	isTelNotify: Is call notification enabled isSmsNotify: Is SMS notification enabled isWechatNotify: Is WeChat notification enabled isQQNotify:Is QQ notification enabled isFacebookNotify: Is Facebook notification enabled isTwitterNotify: Is Twitt notification enabled isSkypeNotify: Is Skype notification enabled isLineNotify:Is Line notification enabled isWhatsappNotify: Is Whatsapp notification enabled isKakaoTalkNotify: Is KakaoTalk notification enabled isInstagramNotify: Is Instagram notification enabled
description	Device notification setting information callback
method	void onSleepMonitorChange(boolean enable, int startHour, int startMin, int endHour, int endMin);
return value	

parameter description	enable: Is the function enabled startHour: Hours of start time startMin: Minutes of start time endHour: Hours of end time endMin: Minutes of end time
description	Sleep monitoring setting information callback
method	void onWearHandChange(boolean isLeft);
return value	
parameter description	isLeft: Is it worn with the left hand
description	Wearing method information callback
method	void onLanguageChange(int languageCode);
return value	
parameter description	languageCode: Current device language
description	Device side language information callback
method	void onVibrationChange(boolean enable);
return value	
parameter description	enable: Is the device vibration function enabled
description	Device vibration function setting callback.
method	void onRaiseBrighten(boolean enable, int startHour, int startMin, int endHour,

	int endMin)
return value	
parameter description	enable: Is the function enabled startHour: Hours of start time startMin: Minutes of start time endHour: Hours of end time endMin: Minutes of end time
description	Hands up and screen up setting information callback
method	void onHrAutoTestChange(boolean enable, boolean enableSleepAssist, int startHour, int startMin, int endHour, int endMin, int cycle);
return value	
parameter description	enable: Is the function enabled enableSleepAssist: Is the heart rate assisted sleep measurement function enabled startHour: Hours of start time startMin: Minutes of start time endHour: Hours of end time endMin: Minutes of end time cycle: Number of minutes of recurrence cycle
description	Callback of heart rate automatic measurement setting information.
method	void onDNDSettingChange(boolean enable, int startHour,

	int startMin, int endHour, int endMin);
return value	
parameter description	enable: Is the function enabled startHour: Hours of start time startMin: Minutes of start time endHour: Hours of end time endMin: Minutes of end time
description	Device side do not disturb mode setting information callback.
method	void onTemperatureAutoTestChange(boolean enable, int startHour, int startMin, int endHour, int endMin, int cycle);
return value	
parameter description	enable: Is the function enabled startHour: Hours of start time startMin: Minutes of start time endHour: Hours of end time endMin: Minutes of end time cycle: Number of minutes of recurrence cycle
description	Device side automatic temperature measurement information callback.

3. 14 OnSettingResultListener Set result callback listening

method	void onSettingResult(int key, boolean isSuccess);
return value	

parameter description	key:The return can be determined based on the key value, and the key value information can refer to SetCallbackStatus; isSuccess: Set the success flag.
description	App Set device side information callback

3.15 SyncStateListener Synchronization status monitoring

Attention: During the process of uploading data on the device, please do not perform any other operations. Wait until the data upload is completed before proceeding to avoid causing data transmission errors.

method	void onSyncStart();
return value	
parameter description	
description	Callback for device end to start uploading data
method	void onSyncEnd();
return value	
parameter description	
description	Callback for device end uploading data.

3.16 CustomInfoChangeListener Custom message return listening

method	void onCustomInfoChange(byte[] info);
return value	
parameter description	
description	Customized information returned by the device.

3.17 WriteCommandToBle Send Command Class

method	public static void syncAllHistory();

return value	
parameter description	
description	Send all historical data instructions to synchronize the device end to the device
method	public static void syncTodayHistory();
return value	
parameter description	
description	Send the command to synchronize the device's current historical data to the device.
method	public static void setRaiseBrighten(boolean enable, int startHour, int startMin, int endHour, int endMin);
return value	
parameter description	enable: Is the function enabled startHour: Hours of start time startMin: Minutes of start time endHour: Hours of end time endMin: Minutes of end time
description	Send the command to raise the hand and light up the screen to the device.
method	public static void setLanguage(int language);
return value	
parameter description	language:The language code must be a language code supported by the device.

description	Send language change instructions to the device.
method	public static void getDeviceBattery();
return value	
parameter description	
description	Send command to obtain power to device.
method	public static void setAlarmClock(List<AlarmClockBean> alarmClockList);
return value	
parameter description	alarmClockList: Alarm Data List
description	Sending alarm data to the device.
method	public static void setProfile(int gender,int age,int height,int weight);
return value	
parameter description	gender: gender，0：female，1：male age: age height: Height (cm) weight: Weight (kg)
description	Synchronize user information to devices.
method	public static void setTargetStep(int targetStep);
return value	
parameter description	targetStep: The target number of steps set by the user, with a valid value of 1000-20000 and an interval of 1000
description	Synchronize motion targets to devices.
method	public static void setNotify(boolean isTelNotify,

	boolean isSmsNotify, boolean isWechatNotify, boolean isQQNotify, boolean isFacebookNotify, boolean isTwitterNotify, boolean isSkypeNotify, boolean isLineNotify, boolean isWhatsappNotify, boolean isKakaoTalkNotify, boolean isInstagramNotify);
return value	
parameter description	isTelNotify: Is call notification enabled isSmsNotify: Is SMS notification enabled isWechatNotify: Is WeChat notification enabled isQQNotify:Is QQ notification enabled isFacebookNotify: Is Facebook notification enabled isTwitterNotify: Is Twitt notification enabled isSkypeNotify: Is Skype notification enabled isLineNotify:Is Line notification enabled isWhatsappNotify: Is Whatsapp notification enabled isKakaoTalkNotify: Is KakaoTalk notification enabled isInstagramNotify: Is Instagram notification enabled
description	Send whether to enable notification command to the device.
method	public static void setStepAutoUpdate(boolean isAutoUpdate);
return value	
parameter description	isAutoUpdate: Automatic upload step switch
description	Send real-time upload instructions for step count to the device. When the app

	is in the background, it can be turned off to automatically upload step count and save power. This function is disabled by default.
method	public static void requestAlarmClock();
return value	
parameter description	
description	Send request alarm data command to device.
method	public static void requestSettingInfo();
return value	
parameter description	
description	Send request setting information instruction to device.
method	public static void setVibrationEnable(boolean enable);
return value	
parameter description	enable: Turning vibration on or off
description	Send vibration switch to device.
method	public static void setTemperatureAuto(boolean enable, int startHour, int startMin, int endHour, int endMin, int cycle);
return value	
parameter	enable: true is enable， false is disable

description	startHour: Start Hours startMin: Start Minutes endHour: End Hours endMin: End Minutes cycle: Recurrence period (three values are 60,90,120 respectively)
description	Set automatic temperature measurement information Note: The start and end times cannot be the same.
method	public static void setDNDSetting(boolean enable, int startHour, int startMin, int endHour, int endMin);
return value	
parameter description	enable: true is on, false is off startHour: Start Hours startMin: Start Minutes endHour: End Hours endMin: End Minutes
description	Set Do Not Disturb Mode Note: The start and end times cannot be the same.
method	public static void setSleepMonitor(boolean enable, int startHour, int startMin, int endHour, int endMin);
return value	
parameter description	enable: Sleep monitoring function switch startHour: Monitoring start hours startMin: Monitoring start minutes endHour: Monitoring end hours

	endMin: Monitoring end minutes
description	Send sleep monitoring information to the device end. Note: When the device is in a sleep monitoring state, setting monitoring information will fail. The start and end times cannot be the same.
method	public static void setDeviceTime();
return value	
parameter description	
description	Synchronize the time of the watch Note: After the device is connected, the SDK will automatically synchronize the time of the watch once.
method	public static void restoreFactory();
return value	
parameter description	
description	Restore factory settings.
method	public static void setWearHand(boolean isLeft);
return value	
parameter description	isLeft: Is it worn with the left hand
description	Set left and right hand wear.
method	public static void setHeartAuto(boolean enable, int startHour, int startMin, int endHour, int endMin, int cycle);
return value	

parameter description	enable: true is on, false is off startHour: Start hours startMin: Start minutes endHour: End hours endMin: End minutes cycle: Recurrence period (three values are 60,90,120 respectively)
description	Set automatic heart rate measurement information Note: The start and end times cannot be the same.
method	public static void setSedentaryRemind(boolean enable, boolean isLunchBreakEnable, int step, int sedentaryTime, int startTime, int endTime, int repeat);
return value	
parameter description	enable： Long sitting switch, true for on and false for off isLunchBreakEnable： Lunch break is uninterrupted, true is on, false is off step：Judgment of sedentary steps, less than this value is considered sedentary, with a minimum value of 15 sedentaryTime：Long sitting time, with 15 minute intervals, with a minimum of 45 minutes and a maximum of 120 minutes startTime： start time endTime： end time repeat： repetition period
description	Set up reminders for prolonged sitting.
method	public static void setHourUnit(boolean is24);
return value	
parameter description	is24： Whether to use 24-hour display time
description	Set the time display.

	Note: This feature requires device support and can be determined through BleDeviceConfig. deviceFun. isHave24Setting().
method	public static void setUnitSystem(boolean isMetric);
return value	
parameter description	isMetric: Is it in metric units
description	Set metric and English display.
method	public static void setTodayWeather(int weatherCode, int minTemperature, int maxTemperature, int currentTemperature, String cityName);
return value	
parameter description	weatherCode： Status 0: Clear, 1: Clear to cloudy, 2: Cloudy, 3: Cloudy, 4: Shower, 5: Rain, 6: Thunder shower, 7: Snow, 8: Haze currentTemperature：Current temperature maxTemperature：maximum temperature minTemperature：minimum temperature cityName：City Name
description	Set weather information for the day.
method	public static void findBand();
return value	
parameter description	
description	Send a device search command, and the device will ring and vibrate upon receiving it.
method	public static void setCameraOpen(boolean isOpen)
return value	

parameter description	isOpen: true is on, false is off
description	Set the handheld camera function switch.
method	public static void setHrTestEnable(boolean enable);
return value	
parameter description	enable: true is on, false is off
description	Send a heart rate measurement command, and the device will start/end a single heart rate measurement after receiving it.
method	public static void setTemperatureTestEnable(boolean enable);
return value	
parameter description	enable: true is on, false is off
description	Send a temperature measurement command, and the device will start/end the temperature measurement once it receives it.
method	public static void setBPTestEnable(boolean enable);
return value	
parameter description	enable: true is on, false is off
description	Send a blood pressure measurement command, and the device will start/end a single blood pressure measurement after receiving it.
method	public static void setBOTestEnable(boolean enable);
return value	
parameter description	

description	Send a blood oxygen measurement command, and the device will start/end a single blood oxygen measurement after receiving it.
method	public static void unbindDevice();
return value	
parameter description	
description	Send unbinding instructions.
method	public static void sendInCall(String name);
return value	
parameter description	name: Incoming call number or contact name
description	Send a call reminder to the device.
method	public static void sendMessage(int type, String content);
return value	
parameter description	type: 1.SMS, 2.QQ, 3.Wechat, 4.Facebook, 5.Twitter, 6.Skype, 7.Line, 8.Whatsapp, 9.KakaoTalk, 10.Instagram content：content
description	Push messages to devices
method	public static void sendCustomInfo(byte key, byte[] info)
return value	
parameter description	key：custom command instructions; info: The information to be sent.
description	Send custom information

3.18 SetCallbackStatus: Set and return the corresponding key class

```
public static final int SET_SYSTEM_TIME = 1; // set time
public static final int SET_ALARM_CLOCK = 2; // alarm clock
public static final int SET_TARGET_STEP = 3; // sport Target
public static final int SET_PROFILE = 4; // user information
public static final int SET_SEDENTARY_REMIND = 5; // long sitting reminder
public static final int SET_WEAR_HAND = 6; // wearing with both hands
public static final int SET_NOTIFY = 7; // notice
public static final int SET_VIBRATION = 8; // vibrate
public static final int SET_RAISE_BRIGHTEN = 9; // raise your hand to light up the screen
public static final int SET_FIND_BAND = 11; // find a bracelet
public static final int SET_PHOTO_OPEN = 12; // band photography function
public static final int SET_HEART_RATE_TEST = 13; // heart rate measurement
public static final int SET_BLOOD_PRESSURE_TEST = 14; // command for opening and closing blood pressure
public static final int SET_SLEEP_MONITOR = 15; // sleep monitoring settings
public static final int SET_IN_CALL = 17; // incoming push
public static final int SET_MSG_PUSH = 18; // message push
public static final int SET_DND_MODE = 20; // Do Not Disturb Mode
public static final int SET_LANGUAGE = 21; // Language settings
public static final int SET_HR_AUTO_TEST = 22; // automatic measurement of heart rate
public static final int SET_TEMPERATURE_AUTO_TEST = 24; // automatic temperature measurement
public static final int SET_TEMPERATURE_TEST = 25; // temperature on and off control commands
public static final int SET_HOUR_UNIT = 26; // time display switching
public static final int SET_UNIT_SYSTEM = 27; // unit system switching
public static final int SET_BLOOD_OXYGEN_TEST = 28; // blood oxygen on and off control command
public static final int SET_TODAY_WEATHER = 29; // set weather information
```

3.19 BleDeviceConfig Equipment Information Class

public static String FIRMWARE_REVISION = ""; Firmware version

public static boolean IS_SUPPORT_CLASSIC = false; Does the device come with Bluetooth 3.0

public static boolean IS_SUPPORT_CONTACT = false; Does the device support the address book function

public static boolean IS_SUPPORT_TEMPERATURE = false; Does the device have a temperature function

public static boolean IS_SUPPORT_12_24_HOUR = false; Does the device support 12/24 hour switching

public static boolean IS_SUPPORT_IMPERIAL = false; Does the device support English units

public static boolean IS_SUPPORT_MULTI_LANGUAGE = false; Does the device support multiple languages

public static boolean IS_SUPPORT_BLOOD_OXYGEN = false; Does the device have blood oxygen function

public static boolean IS_SUPPORT_WEATHER = false; Does the device have weather function

public static List<Integer> LanguageCodeList; Device Language List

Which language can be obtained from the following languages based on the values in the device language list:

English, Simplified Chinese, Traditional Chinese, French, Spanish, Polish, Portuguese, Italian, German, Dutch, Turkish, Russian, Czech, Persian, Hungarian, Greek, Arabic, Filipino, Malay, Indonesian, Vietnamese, Thai, Burmese, Indian, Korean, Japanese, Swedish, Hebrew, Finland, Ukraine, Croatian, Cambodian

3.20 GlobalVariable Global Variable Class

public static final byte MONDAY = 0x01; // Monday

public static final byte TUESDAY = 0x02; // Tuesday

public static final byte WEDNESDAY = 0x04; // Wednesday

public static final byte THURSDAY = 0x08; // Thursday

public static final byte FRIDAY = 0x10; // Friday

public static final byte SATURDAY = 0x20; // Saturday

public static final byte SUNDAY = 0x40; // Sunday
public static final byte EVERYDAY = 0x7F; // Everyday

3.21 BleDevice Equipment Information Class

private BluetoothDevice mDevice; // Bluetooth Device
private int mRssi; // Signal strength
private byte[] mScanRecord; // Broadcast Information	

method	public String getName();
return value	
parameter description	
description	Get device name
method	public String getMac();
return value	
parameter description	
description	Obtain device address
method	public byte[] getScanRecord();
return value	
parameter description	
description	Get device broadcast information
method	public String getRssi();
return value	
parameter description	
description	Get device signal strength

3.22 BleScanState Scan Status Enumeration

```
STATE_IDLE(-1), // Scan is idle
STATE_SCANNING(0X01); // Scanning
```

3.23 BloodOxygenBean Blood oxygen data

```
private String time; // measure time
private int value; // data
```

3.24 BloodPressureBean Blood pressure data

```
private String time; // measure time
private int bpSystolic; // hypertension
private int bpDiastolic; // hypotension
```

3.25 DayStepBean HourDataBean Step data

```
private String date; // date
private List<HourDataBean> hourDataList; // Hourly step data

private int step; // step
private int distance; // Distance (meters)
private int calorie; // Calories (calories)
```

3.26 AlarmClockBean Alarm information

```
private int hour; // Reminder Hours
private int minute; // Reminder minutes
private int repeat = 0; // repetition period
```

3.27 HeartRateBean Heart rate data

```
private String time; // measure time
private int value; // value
```

3.28 SleepInfoBean Sleep data

```
private String date; // date yyyy-MM-dd
private int startTime; // start time
private int endTime; // end time
private int deepTime; // deep sleep time
private int lightTime; // light sleep time
```

private int remTime; // Rapid eye movement

3.29 TemperatureBean Temperature data

private String time; // measure time
private int value; // value

4. Disclaimers

All services provided by Shenzhen United Power Technology Co., Ltd. are aimed at assisting customers in accelerating the development progress of products. Any procedures, documents, test results, plans, support and other materials and information provided during the service process are for reference only, and customers have the right not to use them. Our company does not provide any guarantees of completeness, reliability, or other aspects. If special circumstances arise during the customer's use due to any reason Our company shall not be liable for any incidental or indirect losses.