# 1、Introduction

This document will provide a comprehensive introduction to the application development interface instructions to help application developers to better secondary development

# 2、API interface

## 2.1、osl\_app\_init

# osl\_app\_init

### Interface functions

Initialize system

### Prototype

|  |
| --- |
| - int osl\_app\_init(int flag); |

#### Parameter

| **Name** | **Type** | **Description** |
| --- | --- | --- |
| flag | int | Reserved parameters |

#### Return

int Reserved

## 2.2、osl\_system\_start

# osl\_system\_start

### Interface functions

Start application main task

### Prototype

|  |
| --- |
| - void osl\_system\_start(void \* pfun); |

#### Parameter

| **Name** | **Type** | **Description** |
| --- | --- | --- |
| pfun | void\* | Task callback interface |

#### Return

void

## 2.3、osl\_task\_create

# osl\_task\_create

### Interface functions

Create tasks

### Prototype

|  |
| --- |
| - void osl\_task\_create(void \* pfun, int prio); |

#### Parameter

| **Name** | **Type** | **Description** |
| --- | --- | --- |
| pfun | void\* | Task callback interface |
| prio | int | Task priority |

#### Return

void

## 2.4、osl\_tts\_play

# osl\_tts\_play

### Interface functions

Play voice file

### Prototype

|  |
| --- |
| - int osl\_tts\_play(char \* sndfile); |

#### Parameter

| **Name** | **Type** | **Description** |
| --- | --- | --- |
| sndfile | char \* | voice file |

#### Return

| **Type** | **Description** |
| --- | --- |
| int | 0 succed， -1 fail |

## 2.5、osl\_set\_tts\_sound

# osl\_set\_tts\_sound

### Interface functions

Set volume level

### Prototype

|  |
| --- |
| - void osl\_set\_tts\_sound(int volume); |

#### Parameter

| **Name** | **Type** | **Description** |
| --- | --- | --- |
| volume | int | Volume size |

#### Return

void

## 2.6、osl\_get\_tts\_sound

# osl\_get\_tts\_sound

### Interface functions

Get volume size

### Prototype

|  |
| --- |
| - int osl\_get\_tts\_sound(); |

#### Return

| **Type** | **Description** |
| --- | --- |
| int | return volume size |

## 2.7、osl\_Sleep

# osl\_task\_create

### Interface functions

Task sleep

### Prototype

|  |
| --- |
| - void osl\_Sleep(int ms); |

#### Parameter

| **Name** | **Type** | **Description** |
| --- | --- | --- |
| ms | int | Sleep time， Unit millisecond |

#### Return

void

## 2.8、osl\_CheckTimeover

# osl\_task\_create

### Interface functions

Verify timeout

### Prototype

|  |
| --- |
| - int osl\_CheckTimeover(unsigned int tick1 , unsigned int timeover); |

#### Parameter

| **Name** | **Type** | **Description** |
| --- | --- | --- |
| tick1 | unsigned int | current tick |
| timeover | unsigned int | timeout time |

#### Return

| **Type** | **Description** |
| --- | --- |
| int | 1 timeout， 0 NO timeout |

## 2.9、osl\_GetTick

# osl\_GetTick

### Interface functions

Get system tick

### Prototype

|  |
| --- |
| - unsigned int osl\_GetTick(void); |

#### Parameter

void

#### Return

| **Type** | **Description** |
| --- | --- |
| int | system tick |

## 2.10、osl\_getTerminalID

# osl\_getTerminalID

### Interface functions

Read Sequence Number

### Prototype

|  |
| --- |
| - const char\* osl\_getTerminalID( char \*outdata ,int nMaxLen ); |

#### Parameter

| **Name** | **Type** | **Description** |
| --- | --- | --- |
| outdata | char \* | Terminal ID buff |
| nMaxLen | int | Buff size |

#### Return

| **Type** | **Description** |
| --- | --- |
| char \* | Terminal ID buff |

## 2.11、gui\_key\_event

# gui\_key\_event

### Prototype

|  |
| --- |
| - typedef struct \_\_st\_gui\_key\_event {  int keycode;  int status;  }gui\_key\_event; |

#### Parameter

| **Name** | **Type** | **Description** |
| --- | --- | --- |
| keycode | int | Key value |
| status | int | 1 down,0 up |

## 2.12、gui\_key\_getevent

# gui\_key\_getevent

### Interface functions

Get key information

### Prototype

|  |
| --- |
| - int gui\_key\_getevent(gui\_key\_event \*key\_event); |

#### Parameter

| **Name** | **Type** | **Description** |
| --- | --- | --- |
| gui\_key\_event | [gui\_key\_event](enum.md#gui_key_event) | Get key information pointer |

#### Return

| **Type** | **Description** |
| --- | --- |
| int | 1 timeout， 0 NO timeout |

## 2.13、wifi\_remote\_set\_page

# wifi\_remote\_set\_page

### Interface functions

Terminal enters distribution network mode

### Prototype

|  |
| --- |
| - void wifi\_remote\_set\_page(); |

#### Parameter

void

#### Return

void

## 2.14、mf\_sock\_create

# mf\_sock\_create

### Interface functions

Sock creation

### Prototype

|  |
| --- |
| - int mf\_sock\_create(); |

#### Parameter

void

#### Return

| **Type** | **Description** |
| --- | --- |
| int | Less than 0 failed, other success |

## 2.15、mf\_sock\_connect

# mf\_sock\_connect

### Interface functions

Connect to the server

### Prototype

|  |
| --- |
| - int mf\_sock\_connect(int s,const char \* pIp, int nPort); |

#### Parameter

| **Name** | **Type** | **Description** |
| --- | --- | --- |
| s | int | sock index |
| pIP | char \* | sock ip |
| nPort | int | sock port |

#### Return

| **Type** | **Description** |
| --- | --- |
| int | -1 failure, other success |

## 2.16、mf\_sock\_send

# mf\_sock\_send

### Interface functions

send data

### Prototype

|  |
| --- |
| - int mf\_sock\_send(int s, unsigned char \* buff , int size); |

#### Parameter

| **Name** | **Type** | **Description** |
| --- | --- | --- |
| s | int | sock index |
| buff | char \* | send buffer |
| size | int | send size |

#### Return

| **Type** | **Description** |
| --- | --- |
| int | -1 failure, other success |

## 2.17、mf\_sock\_recv

# mf\_sock\_recv

### Interface functions

Receive data

### Prototype

|  |
| --- |
| - int mf\_sock\_recv(int s, unsigned char \* buff, int len, unsigned int timeover); |

#### Parameter

| **Name** | **Type** | **Description** |
| --- | --- | --- |
| s | int | sock index |
| buff | char \* | Receive buffer |
| len | int | Receive length |
| timeover | int | overtime time |

#### Return

| **Type** | **Description** |
| --- | --- |
| int | -1 failure, other success |

## 2.18、mf\_sock\_close

# mf\_sock\_close

### Interface functions

Disconnect the server

### Prototype

|  |
| --- |
| - int mf\_sock\_close(int s); |

#### Parameter

| **Name** | **Type** | **Description** |
| --- | --- | --- |
| s | int | sock index |

#### Return

Reserve

## 2.19、mf\_file\_open

# mf\_file\_open

### Interface functions

Open the file

### Prototype

|  |
| --- |
| - mf\_file\_open(const char \* name , int flag , int mode); |

#### Parameter

| **Name** | **Type** | **Description** |
| --- | --- | --- |
| name | char \* | File name |
| flag | int | Open the sign |
| mode | int | Open mode |

#### Return

| **Type** | **Description** |
| --- | --- |
| int | >=0 succed， |

## 2.20、mf\_file\_lseek

# mf\_file\_lseek

### Interface functions

File Location

### Prototype

|  |
| --- |
| - int mf\_file\_lseek(int fp , int pos , int flag); |

#### Parameter

| **Name** | **Type** | **Description** |
| --- | --- | --- |
| fp | int | File handle |
| pos | int | pos position |
| flag | int | flag sign |

#### Return

| **Type** | **Description** |
| --- | --- |
| int | Location of movement |

## 2.21、mf\_file\_write

# mf\_file\_write

### Interface functions

Writing Documents

### Prototype

|  |
| --- |
| - int mf\_file\_write(int fp ,const void \* buff , unsigned int size); |

#### Parameter

| **Name** | **Type** | **Description** |
| --- | --- | --- |
| fp | int | File handle |
| buff | char \* | Buffer |
| size | int | size length |

#### Return

| **Type** | **Description** |
| --- | --- |
| int | Written length |

## 2.22、mf\_file\_read

# mf\_file\_read

### Interface functions

read file

### Prototype

|  |
| --- |
| - int mf\_file\_read(int fp , void \* buff , unsigned int size); |

#### Parameter

| **Name** | **Type** | **Description** |
| --- | --- | --- |
| fp | int | File handle |
| buff | char \* | Buffer |
| size | int | size length |

#### Return

| **Type** | **Description** |
| --- | --- |
| int | Read Length |

## 2.23、mf\_file\_close

# mf\_file\_close

### Interface functions

Close files

### Prototype

|  |
| --- |
| - int mf\_file\_close(int fp ); |

#### Parameter

| **Name** | **Type** | **Description** |
| --- | --- | --- |
| fp | int | File handle |

#### Return

## 2.24、mf\_file\_unlink

# mf\_file\_unlink

### Interface functions

Delete files

### Prototype

|  |
| --- |
| - int mf\_file\_unlink(const char \* name); |

#### Parameter

| **Name** | **Type** | **Description** |
| --- | --- | --- |
| name | char \* | File name |