Граффити смарт WIFI-SDK Описание

# История версий:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| издание | Запись / Просмотр Примечания | Ревизионные люди | Дата пересмотра | замечание |
| 1.0.0 | Создание документов | Лю Канг | 20160630 |  |
| 1.0.1 | Оптимизация схемы последовательности операций | Лю Канг | 20160715 |  |
| 1.0.2 | Изменение некоторых входных параметров и возвращаемых значений, описанных | Хусайн | 20160729 |  |
| 1.0.3 | 1. Увеличение распределения низкой сети при условии 2. Обновите основную библиотеку драйверов и записать бен файл | Лю Канг | 20170330 |  |
| 1.0.4 | 1. устанавливать Интерфейс конфигурации WiFi Описание 2. прибавление Структура каталога SDK 3. Увеличение функции Описание | Лю Канг | 20180412 |  |

* **Структура каталога:**

1. ├── app
2. │ ├── tuya\_common
3. │ │ └── include
4. │ │ └── system
5. │ └── tuya\_user
6. ├── bin
7. │ └── upgrade
8. ├── extra\_include
9. ├── include
10. ├── ld
11. ├── lib
12. └── tools

каталог приложений

app/tuya\_user

Ссылки каталог для файлов заголовка

app/tuya\_common/include 和 app/tuya\_common/include/system

скомпилированный каталог бен

bin/upgrade

# Описание функции:

1. Установить отличительное имя и программно-аппаратные функции обратного вызова версии встроенного программного обеспечения заводской калибровки

VOID set\_firmware\_tp (IN OUT СИМ \* firm\_name, IN OUT СИМ \* firm\_ver)

{

strcpy(firm\_name,APP\_BIN\_NAME); strcpy(firm\_ver,USER\_SW\_VER); return;

}

1. инициализации приложения обратного вызова может быть множество Режим конфигурации WiFi

VOID app\_init (VOID)

{

app\_cfg\_set (WCM\_OLD, NULL);

}

1. GPIO функция тестирования обратного вызова

BOOL gpio\_func\_test (VOID)

{

вернуть TRUE;

}

1. Функция входа приложения

OPERATE\_RET Device\_Init (VOID)

{

// инициализируем рамки

// зарегистрировано PSM сектор

// таймер и другие функции, ключевые для создания

}

# Основная структура интерфейса (доступ граффити облако необходимости использования) НЕ АКТУАЛЬНО ЕЩЁ

* + 1. **получить Номер версии SDK**

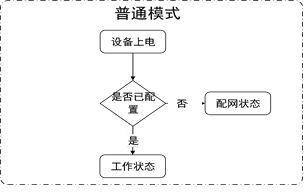
|  |  |
| --- | --- |
| **прототип функции** | CHAR \* tuya\_get\_sdk\_ver (VOID) |
| **Функциональное описание** | Получить SDk номер версии |
| **Входные параметры** | нет |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | SDK номер версии, например, «1.0.0» |
| **замечание** | нет |

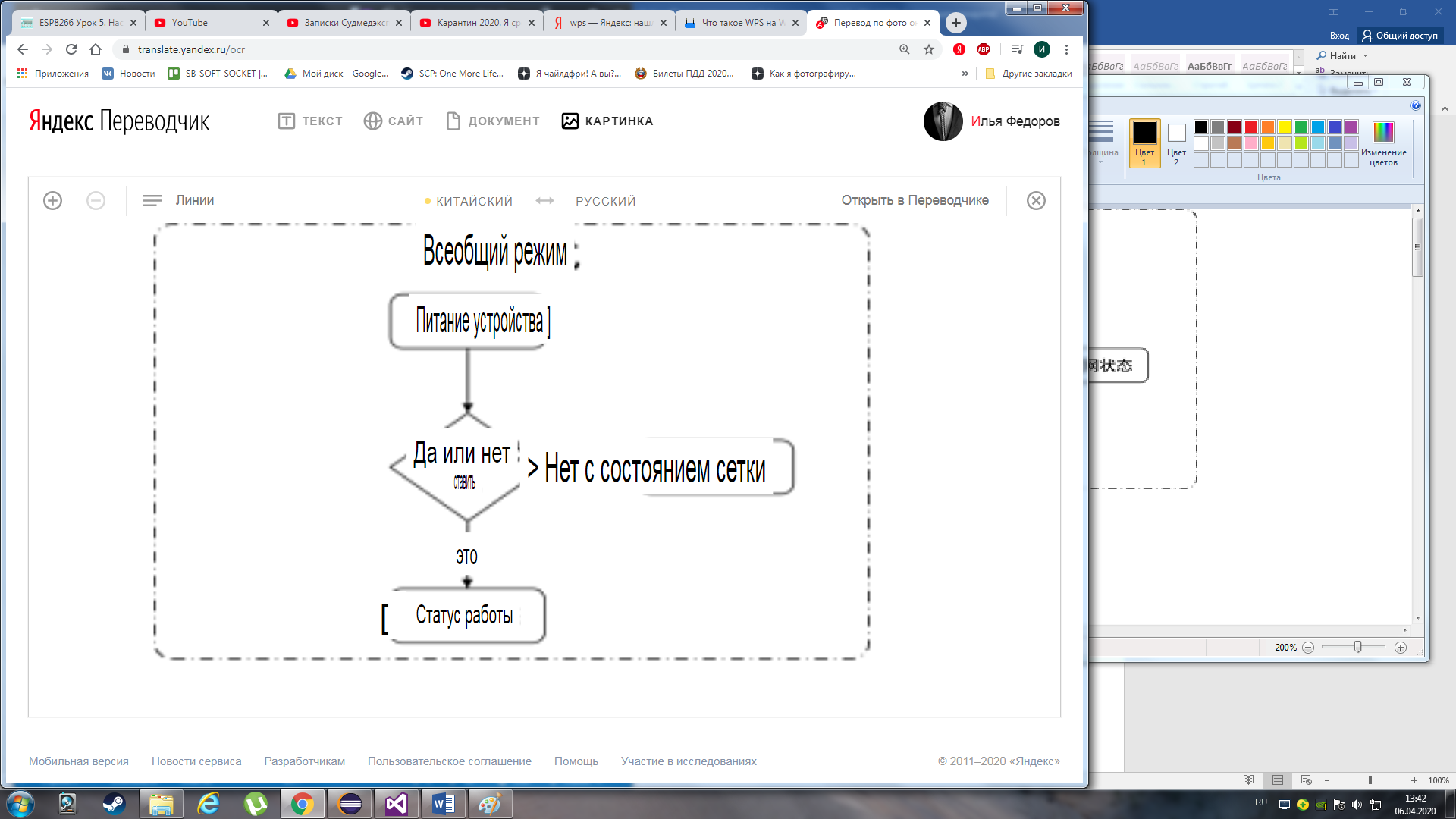
* + 1. **устанавливать Режим конфигурации WiFi**

|  |  |
| --- | --- |
| **прототип функции** | VOID app\_cfg\_set (ВО ФИКСИР WF\_CFG\_MTHD\_SEL воспр, APP\_PROD\_CB обратного вызова) |
| **Функциональное описание** | Настройка режима конфигурации Wi-Fi |
| **Входные параметры** | воспр  WCM\_OLD Нормальный режим (не низкий мощность) WCM\_LOW\_POWER Режим малой мощности (С малой мощности) WCM\_SPCL\_MODE Режим SPECIAL СЕТИ (специальный режим с низким мощность) WCM\_OLD\_CPT Режим совместимости |
|  | Обратный вызов  тест функции обратного вызова Описание продукции:   1. Нормальный режим, нет теста производства обратного вызова, вызов методы   app\_cfg\_set (WCM\_OLD, NULL);   1. Режим малой мощности, должен быть установлен, чтобы получить измеренный обратный вызов   app\_cfg\_set (WCM\_LOW\_POWER, prod\_test);   1. Специальное распределение сетевого режима должен быть установлен обратным вызовом Измеренного выхода   app\_cfg\_set (WCM\_SPCL\_MODE, prod\_test);   1. Режим совместимости   app\_cfg\_set (WCM\_OLD\_CPT, prod\_test); |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | нет |
| **замечание** | Нормальный режим по умолчанию, вы должны позвонить в app\_init |

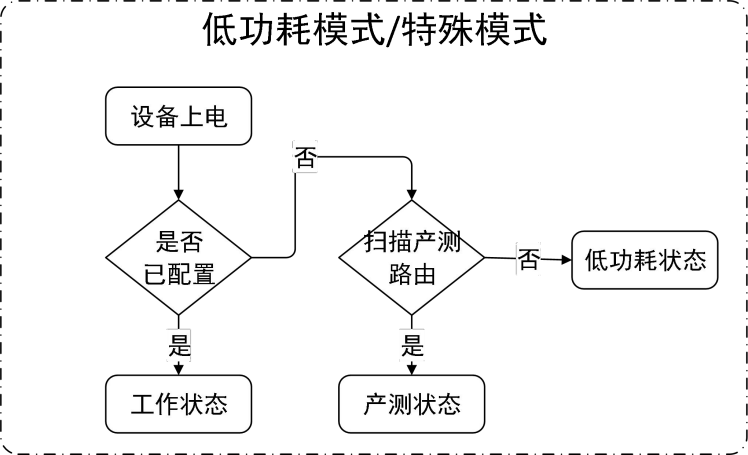
Различные режимы рабочего состояния процесса переключения:

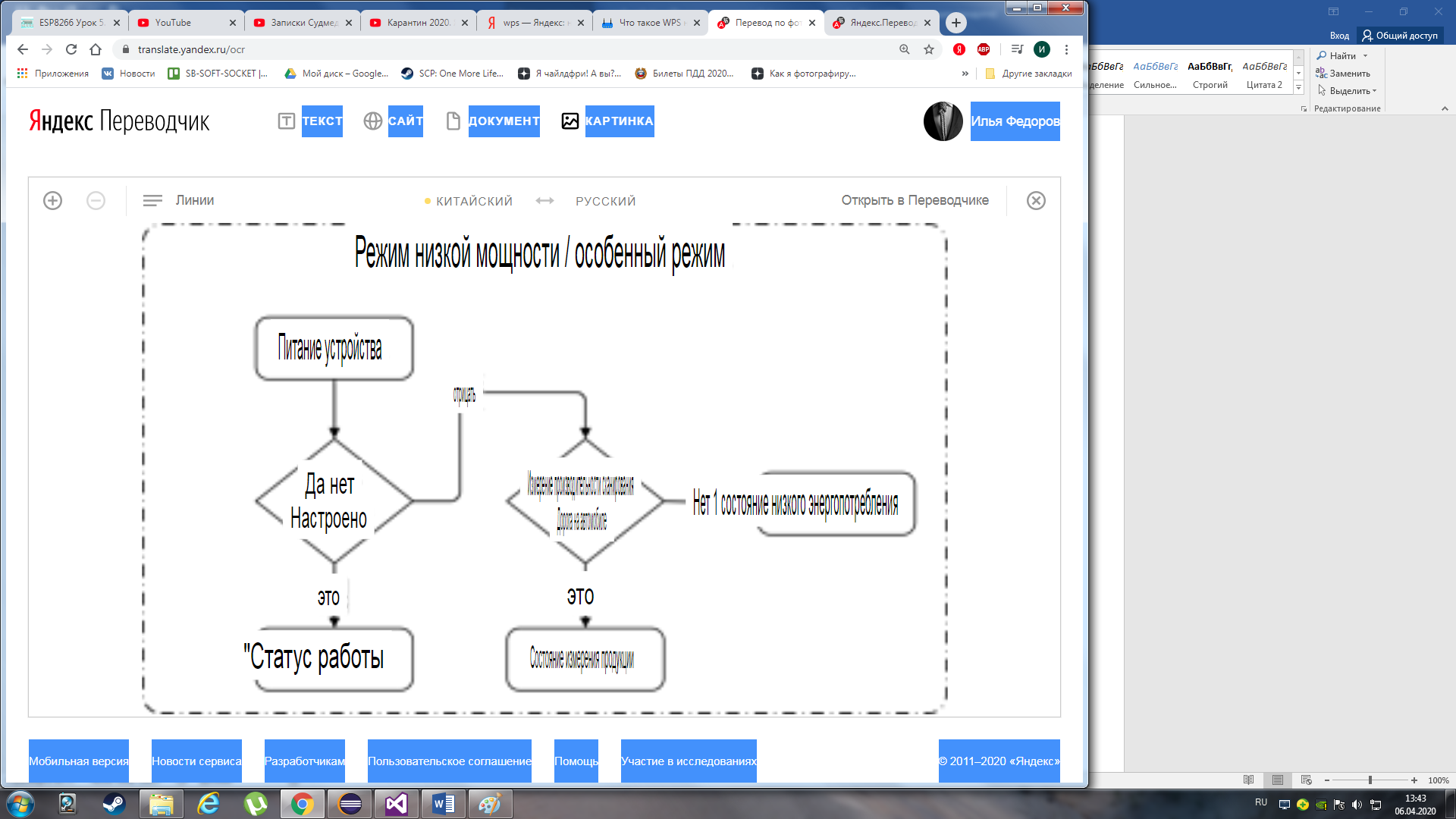
1. Нормальный режим



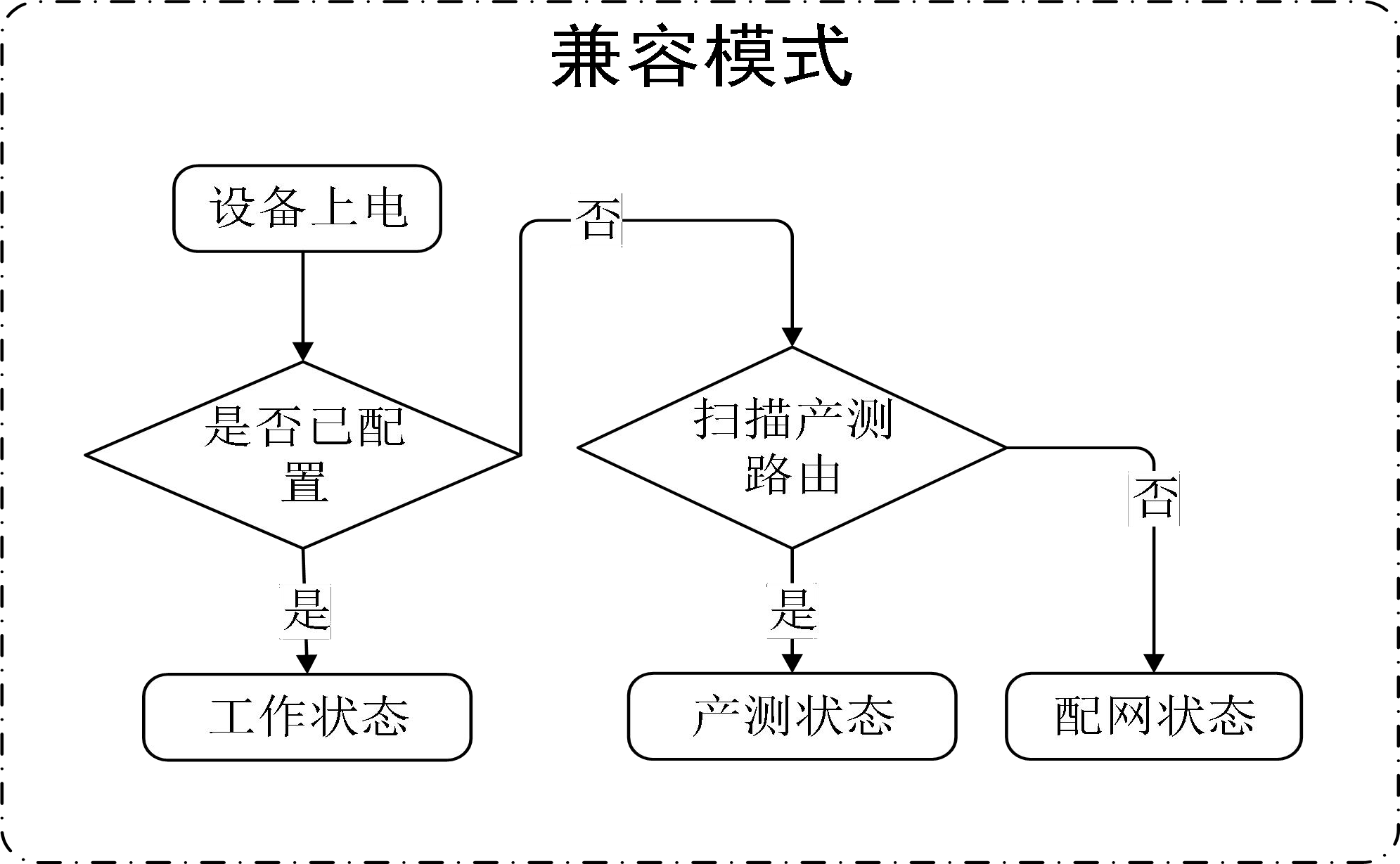


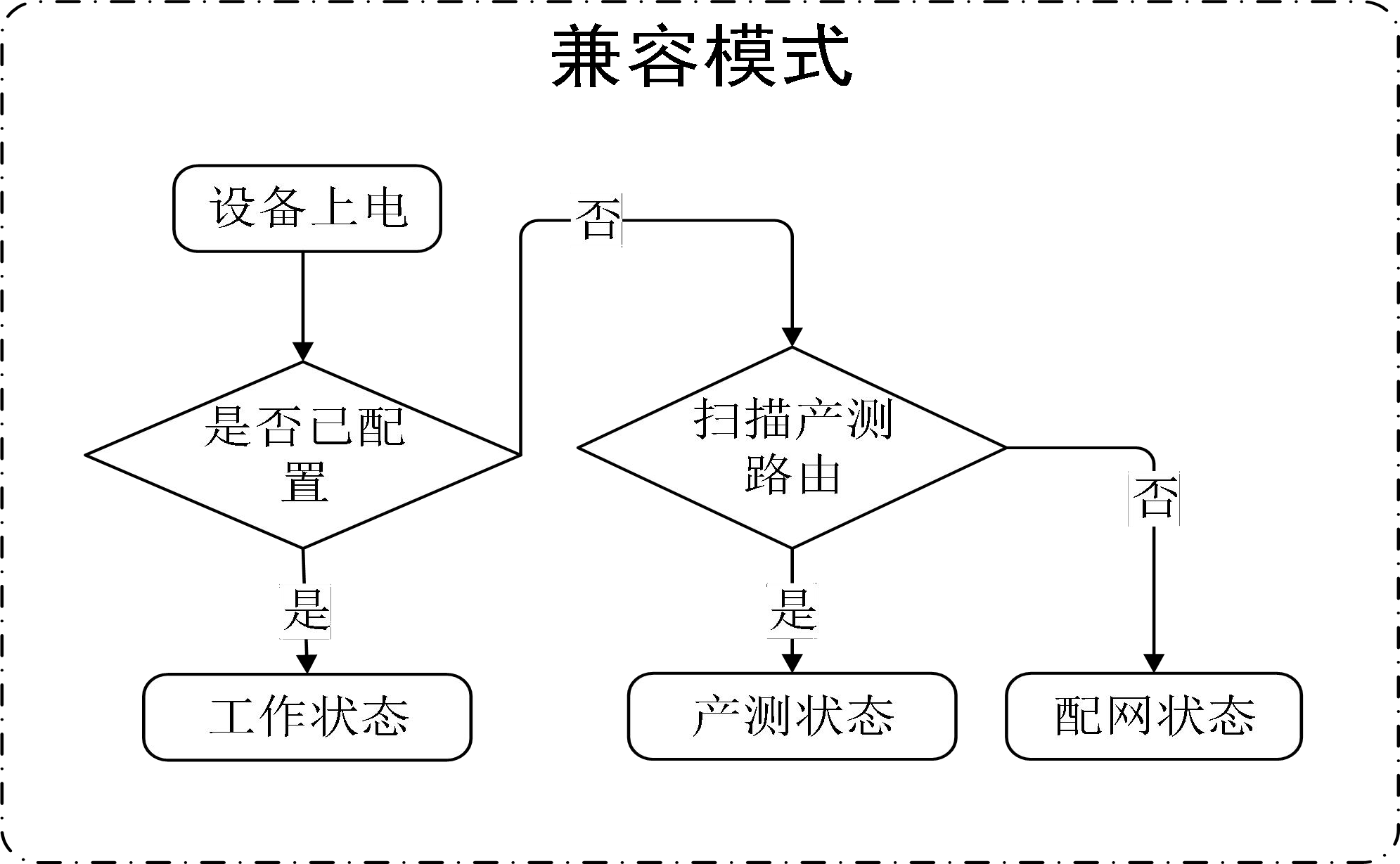
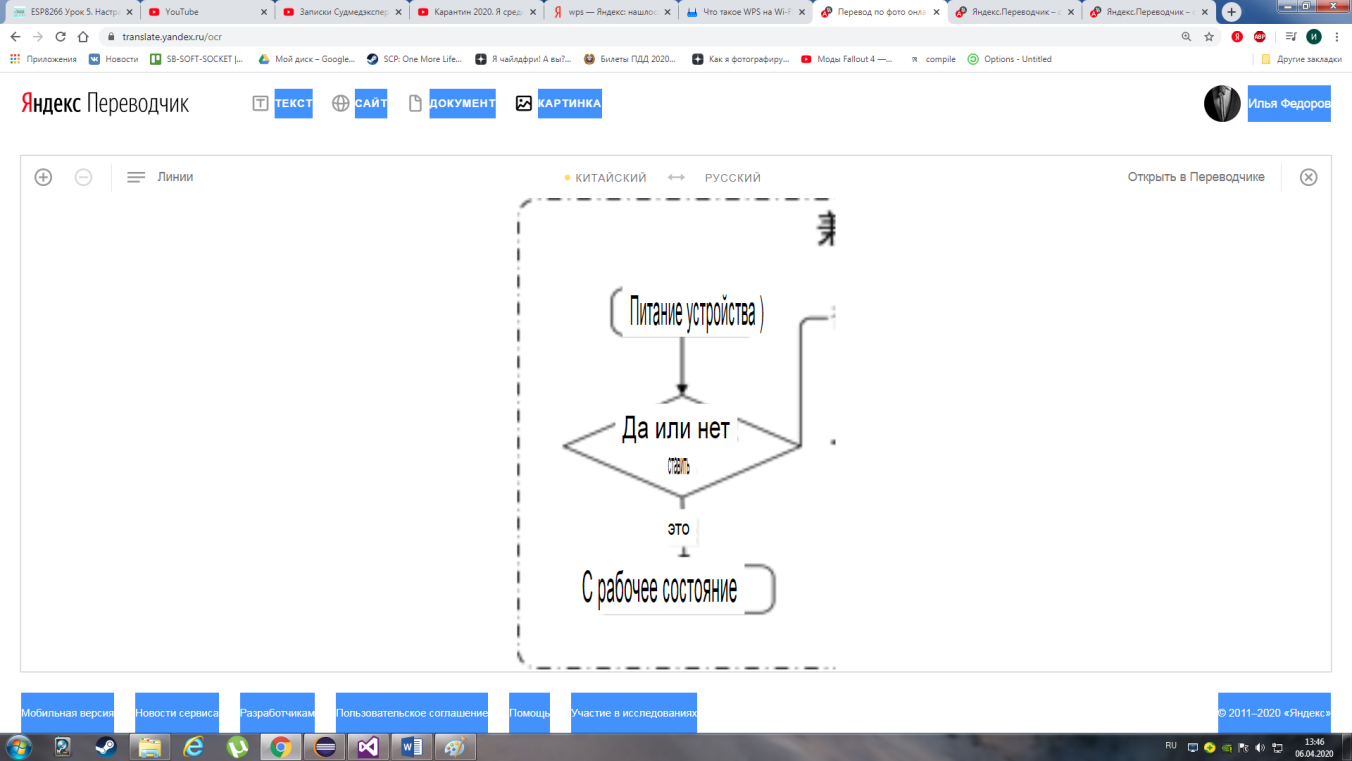
1. Режим малой мощности / специальный режим распределительной сети





1. Режим совместимости





# устройство инициализации

|  |  |
| --- | --- |
| **прототип функции** | OPERATE\_RET tuya\_device\_init (В ПОСТ CHAR \* product\_id,  В ПОСТ SMART\_FRAME\_CB центибар, ПОСТ CHAR \* app\_ver); |
| **Функциональное описание** | Зарегистрироваться функции обработки данных и номер версии приложения |
| **Входные параметры** | product\_id ID продукта  центибар телефон функция App Команда обратного вызова указатель,  VOID (\*) (SMART\_CMD\_E CMD, cJSON \* корень)  <1> CMD тип команды   1. Представляет сетевые издавал приказы местной области 2. Он представляет собой команду, выданную внешней сети   <2> Данные команды корня  Пример, { "1": 100, "2": 200}, в котором«1» и «2» в качестве идентификатора данных (ДПИП) чисел 100 и 200 соответствующих DPID  значение  app\_ver приложение номер версии, например, «1.0.0» |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | См возвращает список значений |
| **замечание** | Инициализация устройства, обработчик данных регистрации |

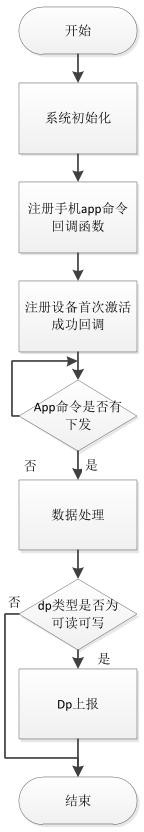


Рисунок 1-1 Процесс инициализации устройства

# Включите устройство для успеха обратного вызова первого раза

|  |  |
| --- | --- |
| **прототип функции** | VOID tuya\_active\_reg (ВО ФИКСИР SYN\_DATA\_CB обратного вызова) |

|  |  |
| --- | --- |
| **Функциональное описание** | Особенности регистрации продукта |
| **Входные параметры** | Функция обратного вызова обратного вызова (для устройства синхронизации состояния)  Обратный вызов определение функции ЬурейеЕ VOID (\* SYN\_DATA\_CB) (VOID); |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | нет |
| **замечание** | Включите устройство в первый раз будет вызывать функцию обратного вызова |

* + 1. **Получить устройство ID**

|  |  |
| --- | --- |
| **прототип функции** | CHAR \* tuya\_get\_devid (VOID) |
| **Функциональное описание** | Получить идентификатор устройства |
| **Входные параметры** | нет |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | Идентификатор устройства |
| **замечание** | нет |

* + 1. **Представление данных**

|  |  |
| --- | --- |
| **прототип функции** | OPERATE\_RET tuya\_obj\_dp\_report (В ПОСТ CHAR \* данные) |
| **Функциональное описание** | Представление данных |
| **Входные параметры** | Данные представили данные, например, { "1": 100, "2": 200} |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | Ссылаясь возвращает список значений |
| **замечание** | Этот интерфейс вызывается микропрограммное состоянием каждого DP будет сохранять данные, такие как данные повторно DP закачанного и сохраненное состояние  Же игнорируется загрузку, рекомендуется использовать эту функцию, сделать APP, облака, микропрограммное трехсторонний лучшую производительность |

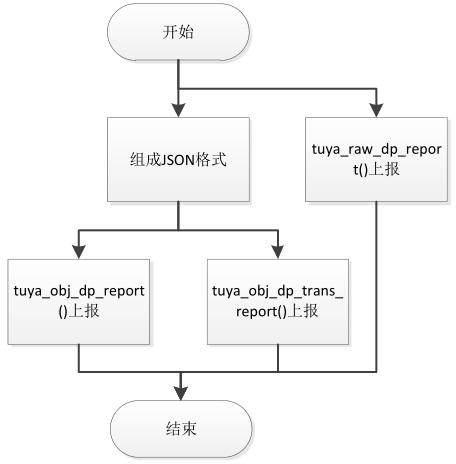


Рисунок 1-2 процесс представления данных

# Представление данных (проходной)

|  |  |
| --- | --- |
| **прототип функции** | OPERATE\_RET tuya\_obj\_dp\_trans\_report (В ПОСТ CHAR \* данные) |
| **Функциональное описание** | Представление данных (проходной) |
| **Входные параметры** | Данные представили данные, например, { "1": 100, "2": 200} |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | См возвращает список значений |
| **замечание** | Через данные на сервер, а не внутреннее состояние обработки прошивки |

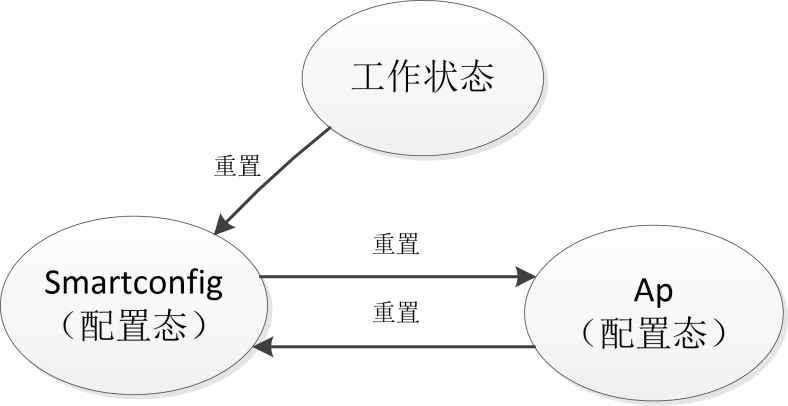
* + 1. **RAW Представление данных**

|  |  |
| --- | --- |
| **прототип функции** | OPERATE\_RET tuya\_raw\_dp\_report (В КОНСТАНТА BYTE ДПИД, В ПОСТ BYTE \* данные, IN  ПОСТ UINT LEN) |
| **Функциональное описание** | представление данных RAW |
| **Входные параметры** | Ссылка точечный дп DPID  необработанные данные двоичные данные  Длина Len данных |
| **Выходные параметры** | нет |

|  |  |
| --- | --- |
| **Возвращаемое значение** | См возвращает список значений |
| **замечание** | Через данные на сервер, а не внутреннее состояние обработки прошивки |

* + 1. **Восстановление заводских настроек**

|  |  |
| --- | --- |
| **прототип функции** | VOID tuya\_dev\_reset\_factory (VOID) |
| **Функциональное описание** | Восстановление заводских настроек |
| **Входные параметры** | нет |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | нет |
| **замечание** | Четкое распределение сеть и информация об устройстве  <1> Устройство активируется, то устройство будет сброшено к информации распределительных сетей активации называют smartconfig и ясно  <2> Устройство не включается, вызывает эту функцию повторно вызывает устройства для переключения вперед и назад smartconfig, ап распределительных сетей |



состояние переключения фиг фиг.1-3 Wi-Fi

# Восстановление заводских настроек и переключатель в указанном государстве

|  |  |
| --- | --- |
| **прототип функции** | VOID tuya\_dev\_reset\_select (режим NW\_CFG\_MODE\_E) |
| **Функциональное описание** | Восстановление заводских настроек и переключатель в указанном государстве |
| **Входные параметры** | состояние конфигурации режима  <1> NW\_SMART\_CFG EZ состояние  <2> NW\_AP\_CFGAP состояние |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | нет |
| **замечание** | Четкое распределение сеть и информация об устройстве |

* + 1. **Получить статус устройства**

|  |  |
| --- | --- |
| **прототип функции** | GW\_STAT\_E tuya\_get\_gw\_status (VOID) |
| **Функциональное описание** | Получить статус устройства |
| **Входные параметры** | нет |

|  |  |
| --- | --- |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | ЬурейеЕ перечисление {  UN\_INIT = 0, // не инициализирована, например, производственная информация не записывается  PROD\_TEST, // выход продукт режим измерения UN\_ACTIVE, // неактивный ACTIVE\_RD, // готовое состояние активации STAT\_WORK, // режим нормальной работы  } GW\_STAT\_E; |
| **замечание** | нет |

* + 1. **получить WIFI рабочее состояние**

|  |  |
| --- | --- |
| **прототип функции** | GW\_WIFI\_STAT\_E tuya\_get\_wf\_status (VOID) |
| **Функциональное описание** | WIFI статус получить работу |
| **Входные параметры** | нет |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | ЬурейеЕ перечисление {  STAT\_LOW\_POWER = 0 // состояние низкого энергопотребления (РФ выкл) STAT\_UNPROVISION = 0, // EZ Государство (не устроило)  STAT\_AP\_STA\_UNCONN, // AP состояние (не расположен) STAT\_AP\_STA\_CFG\_UNC, // AP и STA смешивание, STA ненагруженное состояние (Зарезервировано) STAT\_AP\_STA\_CONN, // AP STA и смешанные, подключенное состояние ГНА (Зарезервировано) STAT\_STA\_UNCONN, // СТО несвязанный  STAT\_STA\_CONN, // СТО Connected состояние  } GW\_WIFI\_STAT\_E; |
| **замечание** | нет |

* + 1. **получить Сила сигнала WIFI**

|  |  |
| --- | --- |
| **прототип функции** | OPERATE\_RET tuya\_get\_wf\_rssi (ВО UCHAR \* SSID, OUT СИМ \* RSSI) |
| **Функциональное описание** | Получить уровень сигнала WIFI |
| **Входные параметры** | Имя точки доступа SSID |
| **Выходные параметры** | Сила сигнала RSSI (дБм) |
| **Возвращаемое значение** | См возвращает список значений |
| **замечание** | Процесс сканирования будет заблокирован, не следует использовать в функции инициализации |

* + 1. **Получает статус соединения облака**

|  |  |
| --- | --- |
| **прототип функции** | BOOL tuya\_get\_cloud\_stat (VOID) |

|  |  |
| --- | --- |
| **Функциональное описание** | Получает статус соединения облака |
| **Входные параметры** | нет |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | TRUE, FALSE Не подключен Подключен |
| **замечание** | нет |

* + 1. **Получить статус обновления устройств**

|  |  |
| --- | --- |
| **прототип функции** | BOOL tuya\_get\_ug\_stat (VOID) |
| **Функциональное описание** | Получить статус обновления устройств |
| **Входные параметры** | нет |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | TRUE, FALSE не обновление обновления |
| **замечание** | нет |

* + 1. **Получить местное время**

|  |  |
| --- | --- |
| **прототип функции** | OPERATE\_RET tuya\_get\_local\_time (OUT структура тм \* st\_time) |
| **Функциональное описание** | Получить местное время |
| **Входные параметры** | нет |
| **Выходные параметры** | st\_time временная структура |
| **Возвращаемое значение** | Ссылаясь возвращает список значений |
| **замечание** | Вы можете использовать устройство для активации |

* + 1. **регистрация PSM имя модуля и имя сектора**

|  |  |
| --- | --- |
| **прототип функции** | OPERATE\_RET tuya\_psm\_register\_module (В ПОСТ CHAR \* module\_name,  В ПОСТ CHAR \* partition\_key) |
| **Функциональное описание** | Вход PSM имя модуля и имя сектора |
| **Входные параметры** | module\_name Имя модуля  partition\_key название сектора |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | OPRT\_PSM\_E\_EXIST сектор был зарегистрирован, OPRT\_OK успешно, другой для отказа |
| **замечание** | Каждый сектор сэкономить до 4K данных |

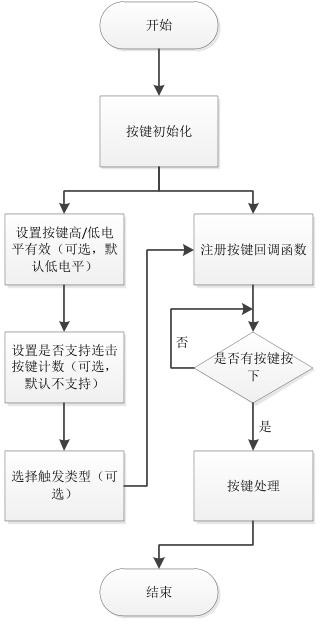
* + 1. **Написать данные PSM**

|  |  |
| --- | --- |
| **прототип функции** | OPERATE\_RET tuya\_psm\_set\_single (В ПОСТ CHAR \* модуль, В ПОСТ CHAR  \* Переменная, В ФИКСИР СИМ \* значение) |
| **Функциональное описание** | Пишет данные PSM |
| **Входные параметры** | module\_name Имя модуля  переменная имена переменных  значение значения переменных |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | Ссылаясь возвращает список значений |
| **замечание** | Поддерживается только тип строки |

* + 1. **Читать данные PSM**

|  |  |
| --- | --- |
| **прототип функции** | OPERATE\_RET tuya\_psm\_get\_single (В ПОСТ CHAR \* модуль,  В ФИКСИР СИМ \* переменная, OUT СИМ \* значение, В ФИКСИР без знака max\_len) |
| **Функциональное описание** | PSM чтения данных |
| **Входные параметры** | module\_name Имя модуля  переменная имена переменных  max\_len Размер буфера |
| **Выходные параметры** | значение значения переменных |
| **Возвращаемое значение** | Ссылаясь возвращает список значений |
| **замечание** | Поддерживается только тип строки |

1. **интерфейс Button**



Фиг обрабатывает с помощью клавиш 2-1

# Ключ инициализации

|  |  |
| --- | --- |
| **прототип функции** | OPERATE\_RET tuya\_kb\_init (VOID) |
| **Функциональное описание** | Ключ инициализации |
| **Входные параметры** | нет |

|  |  |
| --- | --- |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | Ссылаясь возвращает список значений |
| **замечание** | нет |

* + 1. **Обеспечено поддержка заблокированного теста (т.е. типа SEQ\_KEY)**

|  |  |
| --- | --- |
| **прототип функции** | VOIDtuya\_set\_kb\_seq\_enable (IN BOOL включить) |
| **Функциональное описание** | При условии поддержки кнопка подсчета Тесто Тип является SEQ\_KEY |
| **Входные параметры** | включить  <1> ИСТИНА поддержки бьющего ключа счетчика событий, тип события может вызвать SEQ\_KEY  <2> FALSE не поддерживает тесто заблокированные, только поддерживает запуск NORMAL\_KEY, событие LONG\_KEY |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | нет |
| **замечание** | Если вы не вызываете этот интерфейс по умолчанию всей ключевых функций подсчета Теста ключи поддержки обработки |

* + 1. **Активный ключ высокого значения**

|  |  |
| --- | --- |
| **прототип функции** | VOIDtuya\_set\_kb\_detect\_high\_valid (BOOL is\_high) |
| **Функциональное описание** | Активный ключ высокого значения |
| **Входные параметры** | is\_high  <1> ИСТИНА Active High  <2> FALSE, активный низкий |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | нет |
| **замечание** | нет |

* + 1. **Установить кнопку типа триггера**

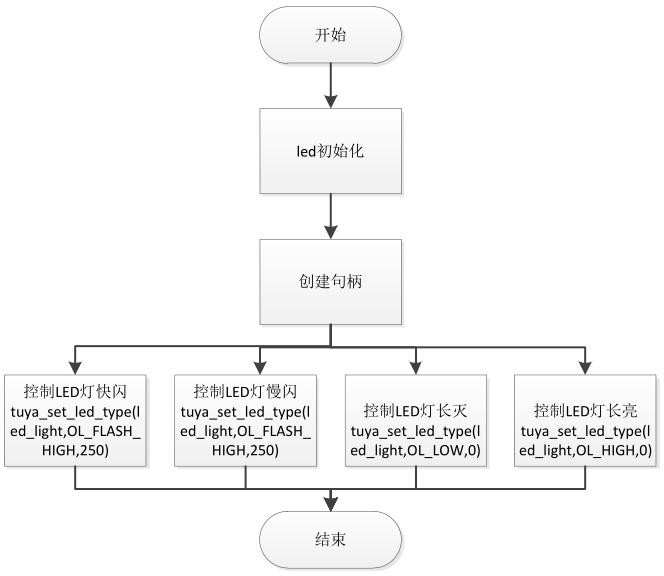
|  |  |
| --- | --- |
| **прототип функции** | VOID tuya\_set\_kb\_trig\_type (В ПОСТ INT gpio\_no,  В ПОСТ KEY\_TRIGGER\_TP\_E trig\_ty, В ПОСТ BOOL down\_trig\_cont) |
| **Функциональное описание** | Ключ инициализации |
| **Входные параметры** | gpio\_no: 0-16 соответствуют Io0-IO16 trig\_ty:  <1> KEY\_UP\_TRIG триггер отказов  <2> KEY\_DOWN\_TRIG Нажмите на спусковой крючок, только поддержка NORMAL\_KEY и LONG\_KEY down\_trig\_cont:  <1> действует только в том случае, когда trig\_ty == KEY\_DOWN\_TRIG  <2> TURE Если пользователи нажав трюм, интервал будет 400ms начиная событие NORMAL\_KEY снова  <3> FALSE пользователи отжимало трюм, только один раз вызвать |

|  |  |
| --- | --- |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | нет |
| **замечание** | По умолчанию, если вы не вызываете эту функцию, ключевые события являются тип триггера отскока |

* + 1. **Регистр обратного вызова функциональных клавиш**

|  |  |
| --- | --- |
| **прототип функции** | OPERATE\_RET tuya\_kb\_reg\_proc (В ПОСТ INT gpio\_no,  В ФИКСИР INT long\_key\_time, В ФИКСИР KEY\_CALLBACK call\_back); |
| **Функциональное описание** | Регистр обратного вызова функциональных клавиш |
| **Входные параметры** | gpio\_no: 0-16, соответственно, что соответствует Io0-IO16  long\_key\_time: длинный ключ время запуска (мс), 0 устанавливаются в качестве маски длиной ключа  call\_back функции обратного вызова ключи,  VOID (\*) (INT gpio\_no, тип PUSH\_KEY\_TYPE\_E, INT CNT) gpio\_no GPIO No.  Кнопки типа Типы  <1> NORMAL\_KEY общий ключ  <2> SEQ\_KEY  Описание: бисквитные ключи, ключ подпрыгивать тип триггера только эффективные, такие как тип параметра обратного вызова нажмите дважды  == SEQ\_KEY, CNT == 2 (CNT представляют собой непрерывные щелчки)  <3> LONG\_KEY Нажимать кнопку  CNT количество нажатий клавиш |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | Ссылаясь возвращает список значений |
| **замечание** | нет |

1. **Интерфейс LED**



3-1 LED с использованием способа по фиг.

# создать LED ручка управления

|  |  |
| --- | --- |
| **прототип функции** | OPERATE\_RET tuya\_create\_led\_handle (В ПОСТ INT gpio\_no,  OUT LED\_HANDLE \* ручка) |
| **Функциональное описание** | Создание LED рукоятки управления |
| **Входные параметры** | gpio\_noGPIO Нет. |
| **Выходные параметры** | handleLED Ручка |
| **Возвращаемое значение** | Ссылаясь возвращает список значений |
| **замечание** | нет |

* + 1. **Контроль LED**

|  |  |
| --- | --- |
| **прототип функции** | VOID tuya\_set\_led\_type (В ПОСТ LED\_HANDLE ручка, тип В ПОСТ LED\_LT\_E,  В ПОСТ USHORT flh\_mstime) |
| **Функциональное описание** | Ключ инициализации |
| **Входные параметры** | handleLED Ручка |

|  |  |
| --- | --- |
|  | тип Тип управления  <1> OL\_LOWLED низкий  <2> OL\_HIGHLED высокая  <3> OL\_FLASH\_LOWLED низкий мигающий  <4> OL\_FLASH\_HIGH светодиод мигает максимум  flh\_mstime мигающий интервал |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | нет |
| **замечание** | нет |

1. **УАПП интерфейс**



Рисунок 4-1 использовать последовательный поток

# Настройка печати серийного номера

|  |  |
| --- | --- |
| **прототип функции** | недействительный print\_port\_init (UART\_Port uart\_no) |
| **Функциональное описание** | Настройка печати серийного номера |
| **Входные параметры** | uart\_no серийный номер |
| **Выходные параметры** | нет |

|  |  |
| --- | --- |
| **Возвращаемое значение** | нет |
| **замечание** | Скорость передачи данных по умолчанию 74880 печатей  Uart\_io == UART1, информация о печати из выходного порта IO2, tysdk всей информации о печати от значения по умолчанию  Выходной порт IO2, пользователь при написании коды приложения через интерфейс реконфигурируемого |

* + 1. **Печать параметров последовательных портов**

|  |  |
| --- | --- |
| **прототип функции** | недействительным print\_port\_full\_init (UART\_Port uart\_no, UART\_BautRate bit\_rate,  UART\_WordLength data\_bits, четность UART\_ParityMode, UART\_StopBits stop\_bits); |
| **Функциональное описание** | Печать параметров последовательных портов |
| **Входные параметры** | uart\_no серийный номер  bit\_rate Скорость передачи данных (300-3686400) data\_bits биты данных  соотношение паритет  stop\_bits Стоп-биты |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | нет |
| **замечание** | Uart\_io == UART1, информация о печати из выходного порта IO2, tysdk всей информации о печати от значения по умолчанию  Выходной порт IO2, пользователь при написании коды приложения через интерфейс реконфигурируемого |

* + 1. **Читать уровень GPIO**

|  |  |
| --- | --- |
| **прототип функции** | INT tuya\_read\_gpio\_level (USHORT gpio\_no) |
| **Функциональное описание** | Чтение уровня GPIO |
| **Входные параметры** | gpio\_no: 0-16, соответственно, что соответствует Io0-IO16 |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | 1 Высокий 0 Низкий |
| **замечание** | нет |

* + 1. **UART0 Последовательная начальная конфигурация**

|  |  |
| --- | --- |
| **прототип функции** | недействительным user\_uart\_full\_init (UART\_BautRate bit\_rate, UART\_WordLength data\_bits, \  UART\_ParityMode четности, UART\_StopBits stop\_bits); |
| **Функциональное описание** | УАПП0 Последовательная начальная конфигурация |
| **Входные параметры** | bit\_rate Скорость передачи данных (300-3686400) data\_bits биты данных  соотношение паритет  stop\_bits Стоп-биты |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | нет |

|  |  |
| --- | --- |
| **замечание** | Если инициализация интерфейса UART0, I015 схемы необходимость специальная обработка, бетон и ощутимый Лексин оригинал документ. Преимущества использования последовательного порта связи, который начался после того, как 8266, чтобы избежать дефолта по выводу информации  Контроль пользователя помех данные панели |

* + 1. **UART0 Последовательная начальная конфигурация (по умолчанию УАПП0)**

|  |  |
| --- | --- |
| **прототип функции** | недействительным user\_uart\_raw\_full\_init (UART\_BautRate bit\_rate, UART\_WordLength data\_bits, \  UART\_ParityMode четности, UART\_StopBits stop\_bits); |
| **Функциональное описание** | УАПП0 Последовательная начальная конфигурация (по умолчанию УАПП0) |
| **Входные параметры** | bit\_rate Скорость передачи данных (300-3686400) data\_bits биты данных  соотношение паритет  stop\_bits Стоп-биты |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | нет |
| **замечание** | esp8266 по умолчанию IO порт UART0 приемопередатчика |

* + 1. **УАПП0 инициализации последовательного порта (esp8266 приемопередатчика порт I015: TX IO13: RX)**

|  |  |
| --- | --- |
| **прототип функции** | недействительный user\_uart\_init (UART\_BautRate bit\_rate) |
| **Функциональное описание** | УАПП0 инициализации последовательного порта (esp8266 приемопередатчика порт I015: TX IO13: RX) |
| **Входные параметры** | bit\_rate Скорость передачи данных (300-3686400) |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | нет |
| **замечание** | По умолчанию data\_bits == 8, четности == нет, stop\_bits == 0  Если инициализация интерфейса UART0, I015 схемы необходимость специальная обработка, бетон и ощутимый Лексин оригинал документ. Преимущества использования последовательного порта связи является информацией по умолчанию для вывода панели вмешательства данных управления пользователя можно избежать после 8266 старта. |

* + 1. **UART0 последовательный порт инициализации(Esp8266 приемопередатчик порта io3: RX IO1: Техас)**

|  |  |
| --- | --- |
| **прототип функции** | недействительный user\_uart\_raw\_init (UART\_BautRate bit\_rate) |
| **Функциональное описание** | УАПП0 инициализации последовательного порта (esp8266 приемопередатчика порт io3: RX IO1: Техас) |
| **Входные параметры** | bit\_rate Скорость передачи данных (300-3686400) |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | нет |
| **замечание** | По умолчанию data\_bits == 8, четности == нет, stop\_bits == 0, esp8266 по умолчанию IO порт трансивера UART0 |

* + 1. **Последовательный размер данных для чтения буфера**

|  |  |
| --- | --- |
| **прототип функции** | uint16 user\_uart\_read\_size (недействительными) |
| **Функциональное описание** | Последовательный размер данных для чтения буфера |
| **Входные параметры** | нет |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | Чтение Длина буфера данных |
| **замечание** | нет |

* + 1. **Читайте последовательные данные**

|  |  |
| --- | --- |
| **прототип функции** | uint16 user\_uart\_read\_data (uint8 \* вне, uint16 out\_len) |
| **Функциональное описание** | Читайте последовательные данные |
| **Входные параметры** | out\_len размер буфера |
| **Выходные параметры** | Данные считываются |
| **Возвращаемое значение** | Считывание данных длины |
| **замечание** | нет |

* + 1. **Написать последовательные данные**

|  |  |
| --- | --- |
| **прототип функции** | недействительным user\_uart\_write\_data (uint8 \* в, uint16 in\_len) |
| **Функциональное описание** | Написать последовательные данные |
| **Входные параметры** | в данных, подлежащих записи  Длина данных in\_len |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | нет |
| **замечание** | нет |

1. **Таймер Интерфейс**
   * 1. **Добавление системного таймера**

|  |  |
| --- | --- |
| **прототип функции** | OPERATE\_RET sys\_add\_timer (В ПОСТ P\_TIMER\_FUNC pTimerFunc,  В ПОСТ PVOID pTimerArg, OUT TIMER\_ID \* p\_timerID) |
| **Функциональное описание** | Добавление системного таймера |
| **Входные параметры** | pTimerFunc: обработчик таймера  pTimerArg: Параметры обработки Таймер |
| **Выходные параметры** | p\_timerID: Таймер идентификационный номер |
| **Возвращаемое значение** | Ссылаясь возвращает список значений |

|  |  |
| --- | --- |
| **замечание** | нет |

* + 1. **Остановка таймера**

|  |  |
| --- | --- |
| **прототип функции** | OPERATE\_RET sys\_stop\_timer (ВО ФИКСИР TIMER\_ID timerID) |
| **Функциональное описание** | Остановка таймера |
| **Входные параметры** | timerID: Таймер идентификационный номер |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | Ссылаясь возвращает список значений |
| **замечание** | нет |

* + 1. **Таймер работает**

|  |  |
| --- | --- |
| **прототип функции** | BOOL IsThisSysTimerRun (ВО ФИКСИР TIMER\_ID timerID) |
| **Функциональное описание** | Таймер работает |
| **Входные параметры** | timerID: Таймер идентификационный номер |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | TRUE, FALSE работает не работает |
| **замечание** | нет |

* + 1. **Запуск таймера**

|  |  |
| --- | --- |
| **прототип функции** | OPERATE\_RET sys\_start\_timer (В ПОСТ TIMER\_ID timerID,  В ПОСТ TIME\_MS timeCycle, В ПОСТ TIMER\_TYPE timer\_type) |
| **Функциональное описание** | Запуск таймера |
| **Входные параметры** | timerID: Таймер ID ID timeCycle: Период времени (в миллисекундах) timer\_type: Тип таймера  <1> TIMER\_ONCE одиночное исполнения  цикл выполнения <2> TIMER\_CYCLE |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | Ссылаясь возвращает список значений |
| **замечание** | нет |

1. **слоистая интерфейс**
   * 1. **система сна**

|  |  |
| --- | --- |
| **прототип функции** | VOID SystemSleep (ВО ФИКСИР TIME\_MS msTime) |
| **Функциональное описание** | система сна |
| **Входные параметры** | msTime: время сна (в миллисекундах) |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | нет |
| **замечание** | нет |

* + 1. **Перезапуске системы**

|  |  |
| --- | --- |
| **прототип функции** | VOID SystemReset (VOID) |
| **Функциональное описание** | Перезапуске системы |
| **Входные параметры** | нет |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | нет |
| **замечание** | нет |

* + 1. **Сравнивает две строки равны**

|  |  |
| --- | --- |
| **прототип функции** | INT strcasecmp (Const символ \* s1, Const символ \* s2) |
| **Функциональное описание** | Сравнивает две строки равны |
| **Входные параметры** | s1: String s1  s2: строка s2 |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | 0 Non-строки равны Строка в диапазоне от 0 |
| **замечание** | нет |

* + 1. **ASCII переключатель HEX код**

|  |  |
| --- | --- |
| **прототип функции** | недействительный ascs2hex (неподписанный символ \* шестигранная, неподписанный символ \* ИССЫ, INT srclen) |
| **Функциональное описание** | ASCII переключатель HEX код |
| **Входные параметры** | ИСС: ASCII-код  srclen: ASCII длина кода |

|  |  |
| --- | --- |
| **Выходные параметры** | шестигранные: шестигранные данные после преобразования |
| **Возвращаемое значение** | нет |
| **замечание** | нет |

1. **интерфейс Task**
   * 1. **Создать задачу**

|  |  |
| --- | --- |
| **прототип функции** | OPERATE\_RET CreateAndStart (OUT THRD\_HANDLE \* pThrdHandle \  В ПОСТ P\_THRD\_FUNC pThrdFunc, \ IN ФИКСИР PVOID pThrdFuncArg, \  В ПОСТ STACK\_SIZE stack\_size, \ В ПОСТ TRD\_PRI ПОИ, \  В ПОСТ CHAR \* thrd\_name); |
| **Функциональное описание** | Создать задачу |
| **Входные параметры** | pThrdHandle ручка задачpThrdFunc Tasking функцияpThrdFuncArg Параметры задачstack\_size Укажите размер стека задачиПОИ приоритетная задача  thrd\_name Имя задачи |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | Ссылаясь возвращает список значений |
| **замечание** | нет |

* + 1. **Удалить задачу**

|  |  |
| --- | --- |
| **прототип функции** | OPERATE\_RET ThrdJoin (В ПОСТ THRD\_HANDLE thrdHandle,  OUT VOID \*\* ppThrdRet); |
| **Функциональное описание** | Удалить задачу |
| **Входные параметры** | thrdHandle ручка задач |
| **Выходные параметры** | ppThrdRet код выхода Task |
| **Возвращаемое значение** | Ссылаясь на имя возвращаемого значения |
| **замечание** | нет |

1. **Семафор интерфейс**
   * 1. **Семафор создание**

|  |  |
| --- | --- |
| **прототип функции** | SEM\_HANDLE CreateSemaphore (VOID) |
| **Функциональное описание** | Семафор создание |
| **Входные параметры** | нет |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | Семафор ручка |
| **замечание** | нет |

* + 1. **Семафор инициализация**

|  |  |
| --- | --- |
| **прототип функции** | OPERATE\_RET InitSemaphore (В ПОСТ SEM\_HANDLE semHandle,  В ПОСТ UINT semCnt, В ПОСТ UINT sem\_max) |
| **Функциональное описание** | Семафор инициализация |
| **Входные параметры** | semHandle Семафор ручка  semCnt Начальное значение семафора  sem\_max Максимальное значение семафора |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | Ссылаясь на имя возвращаемого значения |
| **замечание** | нет |

* + 1. **Суммы сигналов, передаваемых**

|  |  |
| --- | --- |
| **прототип функции** | OPERATE\_RET PostSemaphore (ВО ФИКСИР SEM\_HANDLE semHandle) |
| **Функциональное описание** | Суммы сигналов, передаваемых |
| **Входные параметры** | semHandle Семафор ручка |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | Ссылаясь на имя возвращаемого значения |
| **замечание** | нет |

* + 1. **Прием семафор**

|  |  |
| --- | --- |
| **прототип функции** | OPERATE\_RET WaitSemaphore (ВО ФИКСИР SEM\_HANDLE semHandle) |
| **Функциональное описание** | Прием семафор |

|  |  |
| --- | --- |
| **Входные параметры** | semHandle Семафор ручка |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | Ссылаясь на имя возвращаемого значения |
| **замечание** | нет |

1. **Mutex интерфейс**
   * 1. **Создание мьютекса**

|  |  |
| --- | --- |
| **прототип функции** | OPERATE\_RET CreateMutexAndInit (OUT MUTEX\_HANDLE \* pMutexHandle) |
| **Функциональное описание** | Создание мьютекса |
| **Входные параметры** | нет |
| **Выходные параметры** | pMutexHandle мьютекс ручка |
| **Возвращаемое значение** | Ссылаясь на имя возвращаемого значения |
| **замечание** | нет |

* + 1. **блокировка мьютекса**

|  |  |
| --- | --- |
| **прототип функции** | OPERATE\_RET MutexLock (ВО ФИКСИР MUTEX\_HANDLE mutexHandle) |
| **Функциональное описание** | блокировка мьютекса |
| **Входные параметры** | pMutexHandle мьютекс ручка |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | Ссылаясь на имя возвращаемого значения |
| **замечание** | нет |

* + 1. **разблокировка мьютекс**

|  |  |
| --- | --- |
| **прототип функции** | OPERATE\_RET MutexUnLock (ВО ФИКСИР MUTEX\_HANDLE mutexHandle) |
| **Функциональное описание** | разблокировка мьютекс |
| **Входные параметры** | pMutexHandle мьютекс ручка |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | Ссылаясь на имя возвращаемого значения |
| **замечание** | нет |

1. **Очередь сообщений Интерфейс**
   * 1. **Создание очереди сообщений**

|  |  |
| --- | --- |
| **прототип функции** | OPERATE\_RET CreateMsgQueAndInit (OUT MSG\_QUE\_HANDLE \* pMsgQueHandle) |
| **Функциональное описание** | Создание очереди сообщений |
| **Входные параметры** | нет |
| **Выходные параметры** | Структура управления сообщениями pMsgQueHandle ручки |
| **Возвращаемое значение** | Ссылаясь на имя возвращаемого значения |
| **замечание** | нет |

* + 1. **Добавить сообщение**

|  |  |
| --- | --- |
| **прототип функции** | OPERATE\_RET AddMsgNodeToQueue (В ПОСТ MSG\_QUE\_HANDLE  msgQueHandle, В ПОСТ MSG\_ID MsgID, В ПОСТ P\_MSG\_DATA pMsgData, В ПОСТ MSG\_DATA\_LEN msgDataLen, В ФИКС Msg\_type MsgType) |
| **Функциональное описание** | Добавить сообщение |
| **Входные параметры** | Структура управления сообщениями pMsgQueHandle ручки  MsgID сообщение данных идентификатор сообщения pMsgData  длина данных сообщения msgDataLen  MsgType тип сообщения |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | Ссылаясь на имя возвращаемого значения |
| **замечание** | режим FIFO, система выполняет последовательность сообщений |

* + 1. **Получает указанный Сообщение идентификатор узла**

|  |  |
| --- | --- |
| **прототип функции** | OPERATE\_RET GetMsgNodeFromQueue (В ПОСТ MSG\_QUE\_HANDLE  msgQueHandle, В ФИКСИР MSG\_ID MsgID, выход P\_MSG\_LIST \* PpMsgListNode) |
| **Функциональное описание** | Получить указанный идентификатор узла сообщения |
| **Входные параметры** | Структура управления сообщениями pMsgQueHandle ручки  Идентификатор сообщения MsgID |
| **Выходные параметры** | pMsgListNode сообщение Node |
| **Возвращаемое значение** | Ссылаясь на имя возвращаемого значения |
| **замечание** | нет |

* + 1. **Приобретение узла цепи первого сообщения**

|  |  |
| --- | --- |
| **прототип функции** | OPERATE\_RET GetFirstMsgFromQueue (В ПОСТ MSG\_QUE\_HANDLE  msgQueHandle, выход P\_MSG\_LIST \* ppMsgListNode) |
| **Функциональное описание** | Приобретение узла цепи первого сообщения |
| **Входные параметры** | Структура управления сообщениями pMsgQueHandle ручки |
| **Выходные параметры** | pMsgListNode сообщение Node |
| **Возвращаемое значение** | Ссылаясь на имя возвращаемого значения |
| **замечание** | нет |

* + 1. **Получить общее количество сообщений в очереди узла**

|  |  |
| --- | --- |
| **прототип функции** | OPERATE\_RET GetMsgNodeNum (В ПОСТ MSG\_QUE\_HANDLE msgQueHandle,  OUT PINT pMsgNodeNum) |
| **Функциональное описание** | Получить общее количество сообщений в очереди узла |
| **Входные параметры** | Структура управления сообщениями pMsgQueHandle ручки |
| **Выходные параметры** | pMsgNodeNum общее количество узлов новостей |
| **Возвращаемое значение** | Ссылаясь на имя возвращаемого значения |
| **замечание** | нет |

* + 1. **Удалить сообщение из узла узла в памяти сообщений цепи и выпуска**

|  |  |
| --- | --- |
| **прототип функции** | OPERATE\_RET DelAndFreeMsgNodeFromQueue (В ПОСТ MSG\_QUE\_HANDLE  msgQueHandle, В ПОСТ P\_MSG\_LIST pMsgListNode) |
| **Функциональное описание** | Удалить сообщение из узла узла в памяти сообщений цепи и выпуска |
| **Входные параметры** | Структура управления сообщениями pMsgQueHandle ручки  pMsgListNode удаленные сообщения узлу |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | Ссылаясь на имя возвращаемого значения |
| **замечание** | нет |

* + 1. **Очередь сообщений Release занимает память**

|  |  |
| --- | --- |
| **прототип функции** | OPERATE\_RET ReleaseMsgQue (ВО ФИКСИР MSG\_QUE\_HANDLE msgQueHandle) |
| **Функциональное описание** | Очередь сообщений Release занимает память |
| **Входные параметры** | Структура управления сообщениями pMsgQueHandle ручки |
| **Выходные параметры** | нет |

|  |  |
| --- | --- |
| **Возвращаемое значение** | Ссылаясь на имя возвращаемого значения |
| **замечание** | нет |

* + 1. **доставка сообщений**

|  |  |
| --- | --- |
| **прототип функции** | OPERATE\_RET PostMessage (ВО ФИКСИР MSG\_QUE\_HANDLE msgQueHandle, В ФИКСИР MSG\_ID MsgID, В ФИКСИР P\_MSG\_DATA pMsgData,  В ПОСТ MSG\_DATA\_LEN msgDataLen) |
| **Функциональное описание** | доставка сообщений |
| **Входные параметры** | обработчик сообщений msgQueHandle  MsgID сообщение данных идентификатор сообщения pMsgData  длина данных сообщения msgDataLen |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | Ссылаясь на имя возвращаемого значения |
| **замечание** | Доставить сообщение для модуля (исполнение сообщения FIFO) |

* + 1. **Deliver срочное сообщение**

|  |  |
| --- | --- |
| **прототип функции** | OPERATE\_RET PostInstancyMsg (В ПОСТ MSG\_QUE\_HANDLE msgQueHandle, \ IN ФИКСИР MSG\_ID MsgID, В ПОСТ P\_MSG\_DATA pMsgData, \  В ПОСТ MSG\_DATA\_LEN msgDataLen) |
| **Функциональное описание** | Deliver срочное сообщение |
| **Входные параметры** | обработчик сообщений msgQueHandle  MsgID сообщение данных идентификатор сообщения pMsgData  длина данных сообщения msgDataLen |
| **Выходные параметры** | нет |
| **Возвращаемое значение** | Ссылаясь на имя возвращаемого значения |
| **замечание** | нет |

* + 1. **В ожидании новостей**

|  |  |
| --- | --- |
| **прототип функции** | OPERATE\_RET WaitMessage (В ПОСТ MSG\_QUE\_HANDLE msgQueHandle,  OUT P\_MSG\_LIST \* ppMsgListNode) |
| **Функциональное описание** | В ожидании новостей |
| **Входные параметры** | обработчик сообщений msgQueHandle |
| **Выходные параметры** | ppMsgListNode сообщение Node |

|  |  |
| --- | --- |
| **Возвращаемое значение** | Ссылаясь на имя возвращаемого значения |
| **замечание** | Успех WaitMessage будет называться, как сообщение было дело с называться  пресс-релиз DelAndFreeMsgNodeFromQueue |

**аппендикс**

1. **Возвращает список значений**

|  |  |  |
| --- | --- | --- |
| Определение макроса возвращаемого значения | Возвращаемое значение | описание |
| OPRT\_OK | 0 | Успешная реализация |
| OPRT\_COM\_ERROR | 1 | Общая ошибка |
| OPRT\_INVALID\_PARM | 2 | Недопустимый параметр в |
| OPRT\_MALLOC\_FAILED | 3 | Ошибка выделения памяти |
| OPRT\_INIT\_MUTEX\_ATTR\_FAILED | 4 | Не удалось инициализировать свойства синхронизации |
| OPRT\_SET\_MUTEX\_ATTR\_FAILED | 5 | Настройка свойства отказа синхронизации |
| OPRT\_DESTROY\_MUTEX\_ATTR\_FAILED | 6 | Разрушение сбоя синхронизации свойства |
| OPRT\_INIT\_MUTEX\_FAILED | 7 | Не удалось инициализировать семафор |
| OPRT\_MUTEX\_LOCK\_FAILED | 8 | отказ блокировки мьютекса |
| OPRT\_MUTEX\_TRYLOCK\_FAILED | 9 | блокировка мьютекса неудачной попытки |
| OPRT\_MUTEX\_LOCK\_BUSY | 10 | мьютекс занят |
| OPRT\_MUTEX\_UNLOCK\_FAILED | 11 | Не удалось разблокировать мьютекс |
| OPRT\_MUTEX\_RELEASE\_FAILED | 12 | Отказ релиза Мьютекса |
| OPRT\_INIT\_SEM\_FAILED | 13 | Не удалось инициализировать семафор |
| OPRT\_WAIT\_SEM\_FAILED | 14 | Ожидание семафора не удалось |
| OPRT\_POST\_SEM\_FAILED | 15 | Не удалось освободить семафор |
| OPRT\_THRD\_STA\_UNVALID | 16 | Тема незаконного статуса |
| OPRT\_THRD\_CR\_FAILED | 17 | Создание резьбы не удалось |
| OPRT\_THRD\_JOIN\_FAILED | 18 | нитьРЕГИСТРИРУЙТЕСЬ Функция терпит неудачу |
| OPRT\_THRD\_SELF\_CAN\_NOT\_JOIN | 19 | Не могу назвать себя ниткуРЕГИСТРИРУЙТЕСЬ функция |
| OPRT\_TIMERID\_EXIST | 20 | таймерID Уже существует |
| OPRT\_TIMERID\_NOT\_FOUND | 21 | Определяет таймер не найденID |
| OPRT\_TIMERID\_UNVALID | 22 | таймерID нелегальный |
| OPRT\_GET\_IDLE\_TIMERID\_ERROR | 23 | Получить таймер простояID ошибка |
| OPRT\_MSG\_NOT\_FOUND | 24 | Указано сообщение не найдено |
| OPRT\_MSG\_LIST\_EMPTY | 25 | цепь Сообщение пусто |
| OPRT\_PSM\_FLH\_RET\_ERR | 26 | PSM Стирание недостаточность |
| OPRT\_PSM\_FLH\_TM\_ERR | 27 | PSM Стирание Тайм-аут |
| OPRT\_PSM\_E\_INVAL | 28 |  |
| OPRT\_PSM\_E\_IO | 29 |  |

|  |  |  |
| --- | --- | --- |
| OPRT\_PSM\_E\_EXIST | 30 |  |
| OPRT\_PSM\_E\_NOENT | 31 |  |
| OPRT\_PSM\_FAIL | 32 |  |
| OPRT\_PSM\_E\_NOSPC | 33 |  |
| OPRT\_PSM\_E\_METADATA\_CRC | 34 |  |
| OPRT\_PSM\_E\_CRC | 35 |  |
| OPRT\_WIFI\_SCAN\_FAIL | 36 |  |
| OPRT\_WF\_MAC\_SET\_FAIL | 37 |  |
| OPRT\_WF\_CONN\_FAIL | 38 |  |
| OPRT\_WF\_NW\_CFG\_FAIL | 39 |  |
| OPRT\_SET\_SOCK\_ERR | 41 |  |
| OPRT\_SOCK\_CONN\_ERR | 42 |  |
| OPRT\_CR\_MUTEX\_ERR | 43 |  |
| OPRT\_CR\_TIMER\_ERR | 44 |  |
| OPRT\_CR\_THREAD\_ERR | 45 |  |
| OPRT\_BUF\_NOT\_ENOUGH | 46 |  |
| OPRT\_URL\_PARAM\_OUT\_LIMIT | 47 |  |
| OPRT\_HTTP\_OS\_ERROR | 48 |  |
| OPRT\_HTTP\_PR\_REQ\_ERROR | 49 |  |
| OPRT\_HTTP\_SD\_REQ\_ERROR | 50 |  |
| OPRT\_HTTP\_RD\_ERROR | 51 |  |
| OPRT\_HTTP\_AD\_HD\_ERROR | 52 |  |
| OPRT\_HTTP\_GET\_RESP\_ERROR | 53 |  |
| OPRT\_HTTP\_AES\_INIT\_ERR | 54 |  |
| OPRT\_HTTP\_AES\_OPEN\_ERR | 55 |  |
| OPRT\_HTTP\_AES\_SET\_KEY\_ERR | 56 |  |
| OPRT\_HTTP\_AES\_ENCRYPT\_ERR | 57 |  |
| OPRT\_TY\_WS\_PART\_ERR | 58 |  |
| OPRT\_CR\_CJSON\_ERR | 59 |  |
| OPRT\_PSM\_SET\_ERROR | 60 |  |
| OPRT\_PSM\_GET\_ERROR | 61 |  |
| OPRT\_CJSON\_PARSE\_ERR | 62 |  |
| OPRT\_CJSON\_GET\_ERR | 63 |  |
| OPRT\_CR\_HTTP\_URL\_H\_ERR | 64 |  |
| OPRT\_HTTPS\_HANDLE\_FAIL | 65 |  |
| OPRT\_HTTPS\_RESP\_UNVALID | 66 |  |
| OPRT\_MEM\_PARTITION\_EMPTY | 67 |  |
| OPRT\_MEM\_PARTITION\_FULL | 68 |  |
| OPRT\_MEM\_PARTITION\_NOT\_FOUND | 69 |  |
| OPRT\_CR\_QUE\_ERR | 70 |  |
| OPRT\_SND\_QUE\_ERR | 71 |  |
| OPRT\_NOT\_FOUND\_DEV | 72 |  |
| OPRT\_NOT\_FOUND\_DEV\_DP | 73 |  |

|  |  |  |
| --- | --- | --- |
| OPRT\_DP\_ATTR\_ILLEGAL | 74 |  |
| OPRT\_DP\_TYPE\_PROP\_ILLEGAL | 75 |  |
| OPRT\_DP\_REPORT\_CLOUD\_ERR | 76 |  |
| OPRT\_NO\_NEED\_SET\_PRODINFO | 77 |  |
| OPRT\_NW\_INVALID | 78 |  |
| OPRT\_SELECT\_ERR | 79 |  |
| OPRT\_SELECT\_TM | 80 |  |
| OPRT\_SEND\_ERR | 81 |  |
| OPRT\_DEV\_NOT\_BIND | 82 |  |
| OPRT\_FW\_UG\_FAILED | 83 |  |
| OPRT\_VER\_FMT\_ERR | 84 |  |
| OPRT\_FW\_NOT\_EXIST | 85 |  |
| OPRT\_SEM\_CR\_ERR | 86 |  |
| OPRT\_SELECT\_TIMEOUT | 87 |  |
| OPRT\_GW\_MQ\_OFFLILNE | 88 |  |
| OPRT\_NOT\_SPT\_CLX\_DP | 89 |  |
| OPRT\_RECV\_ERR | 90 |  |
| OPRT\_UG\_PKG\_NOT\_ENOUGH | 91 |  |
| OPRT\_SCMA\_INVALID | 92 |  |
| OPRT\_PRODECT\_KEY\_NULL | 93 |  |
| OPRT\_DEVICE\_VER\_NULL | 94 |  |
| OPRT\_MSG\_OUT\_OF\_LMT | 95 |  |
| OPRT\_NOT\_FOUND\_AUTH\_SSID | 96 |  |
| OPRT\_SOCKET\_FAULT | 97 |  |
| OPRT\_MQ\_PUBLISH\_TIMEOUT | 98 |  |
| OPRT\_GW\_NOT\_EXIST | 99 |  |
| OPRT\_GW\_SCHEMA\_SIZE\_LMT\_OUT | 100 |  |
| OPRT\_DEV\_DP\_CNT\_INVALID | 101 |  |
| OPRT\_TOKEN\_OVERTIME OPRT\_CR\_TIMER\_FAILED  OPRT\_PUB\_NO\_PERMISSION | 102  103  104 |  |

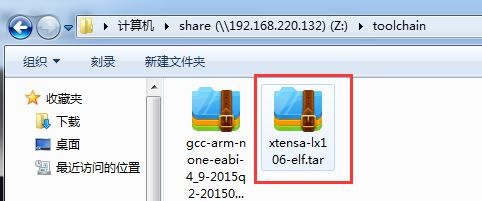
1. **Составитель среда для сборки**
2. **скачать кросскомпилятор**

Вы можете загрузить первый кросс-компилятор (также можно загрузить непосредственно под ununtu) под окнами, скачать адрес:

<http://bbs.espressif.com/viewtopic.php?f=57&t=2>

# Установка кросс-компилятор

Копирование в окнах после завершения загрузки кросс-компилятор Xtensa-lx106-elf.tar.bz2 к общему каталогу, я был специально создан новый каталог ToolChain разместить серию установочных файлов кросс-компилятор



# Распаковка кросс-компилятор

Переключение на терминал в Linux / дома / доли / самба / Набор инструментов каталога, декомпрессионной в Xtensa-lx106-elf.tar.bz2

В каталоге / USR / бен каталог, введите следующую команду:

Sudo смолы -jxvf Xtensa-lx106-elf.tar.bz2 -C / USR / бен

# Свойства каталога Изменить

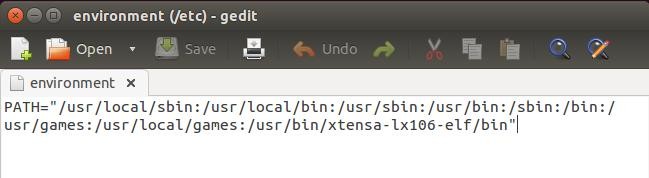
Этот шаг очень важен, в противном случае вам будет предложено во время компиляции не может быть скомпилирован

Судо CHMOD 777 -R / USR / бен / Xtensa-lx106-эльф

# Изменение переменной среды

Открыть / и т.д. / файл среды введите следующую команду:

Sudo Gedit / и т.д. / окружающая среда

Изменить / и т.д. / окружающая среда файл (я использовал для редактирования этого файла), добавить в конце файла: / USR / бен / Xtensa-lx106 эльф / бен к текущей переменных окружения

# выполнение источник / и т.д. / окружающая среда

Переменные окружения вступили в силу после вступления в силу, может повторить $ PATH Смотрите, если успешно

# Приложения, скомпилированные методы

1. вводить tysdk\_for\_esp8266 / каталог приложений
2. Compile выход журнала встроенного программного обеспечения, например, sample\_pk, версия 1.0.0 ш build\_app.sh sample\_pk 1.0.0
3. скомпилировать нет выхода прошивки журнала, например, sample\_pk, версия 1.0.0 ш build\_app\_release.sh sample\_pk 1.0.0
4. скомпилированный Описание файла бен

Составитель бен файл в / bin / каталог обновления

1. sample\_pk (1) \_1.0.0.bin в виде загружаемого файла прошивки от горящего адреса 0x1000
2. sample\_pk\_ug\_1.0.0.bin для файла обновления для использования в качестве обновления программного обеспечения
3. Описание встроенного программного обеспечения горения (1) файлы могут быть записаны адреса

|  |  |  |
| --- | --- | --- |
| Загрузочный файл | boot\_v1.4 (б1) .bin | 0x0000 |
| досье заявок | sample\_pk (1) \_1.0.0.bin | 0x1000 |
| Файл параметров системы | esp\_init\_data\_default.bin | 0xfc000 |
| файл Перетяжка | blank.bin | 0xfe000 |

1. рекомендуется файлы в целевом файле 1M для 1M / 2M / 4M байт вспышки, вы можете только записать размер 1M
2. 