

Техническое задание

на разработку программного обеспечения

для устройства “WiFi метка” на базе процессора PIC18F26K22

Данное ТЗ не является полным и законченным и предоставляется для предварительной оценки объема работ и определения сроков разработки.

Предмет разработки:

Исполнителю предлагается разработать программное обеспечение для процессора PIC18F26K22 для обеспечения необходимого функционала работы устройства согласно существующей принципиальной схеме.

Язык программирования:

Итогом работы будет являться исходный код программы процессора на языке программирования “C/C++” с использованием стандартных или общедоступных подключаемых библиотек, без использования в коде вставок на других языках программирования, стабильно работающий на прототипе устройства и не приводящий к периодическим частичным или полным отказам функционала, а так же отвечающий остальным требованиям технического задания.

Технология изготовления:

Данным ТЗ не предусматривается разработка технологии изготовления устройства. Принципиальная схема, используемые элементы, разводка платы, корпус устройства, остаются за рамками данного проекта.

Питание устройства:

Предполагается портативное питание устройства от литиевого аккумулятора с периодической беспроводной зарядкой по стандарту Qi. Исходный код программы процессора необходимо оптимизировать для минимального энергопотребления в режимах простоя с переводением датчиков в режим энергосбережения.

Основные функции выполняемые процессором:

1. Конфигурация и опрос периферии датчиков по шине i2c
 - 1.1 Акселерометр ADXL345BCCZ
 - 1.2 Контроллер литиевой батареи LC709203FQH
2. Взаимодействие с WiFi модулем HF-LPT200 (дискретные сигналы)
 - 2.1 Опрос состояния выходов модуля nReady, nLink
 - 2.2 Управление входами модуля nReload, Reset
3. Взаимодействие с WiFi модулем HF-LPT200 по UART
 - 3.1 Перевод модуля в командный режим (AT команды)
 - 3.2 Запрос информации о текущем подключении и видимых AP
 - 3.3 Запрос версии ПО WiFi модуля (AT команды)
 - 3.4 Интерпретация поля версия ПО. В этом поле будет передаваться некоторая служебная информация, период сканирования WiFi, период опроса, более подробная информация будет позже.
 - 3.5 Перевод модуля в режим прозрачного Serial канала (AT команды)
4. Опрос состояния входов процессора
 - 4.1 SW_PB1 – интерфейсная кнопка устройства
 - 4.2 Charge – сигнал режима заряда
 - 4.3 Fall – сигнал детектирования падения от акселерометра
 - 4.4 V_mes – аналоговый сигнал напряжения
5. Управление выходами процессора
 - 5.1 LED_PB1 – световая индикация подсветки кнопки
 - 5.2 LED1_on – световая индикация режима Link (10ms ON / 3000ms OFF)
 - 5.3 LED_u – световая индикация режима Charge
 - 5.4 LED_g – световая индикация режима Charge_end
 - 5.5 Buzer – управление пьезокерамическим сигнализатором (ШИМ, профиль режима работы уточнится позже)
 - 5.6 Viibro – управление вибромотором (профиль режима работы уточнится позже)
6. Формирование пакета посылки на сервер в формате JSON
7. Передача пакета посылки на сервер по UART через WiFi модуль
8. Прием ответа сервера в формате JSON по UART
9. Интерпретация ответа сервера, управление режимами.
 - 9.1 Режим “Вызов Диспетчера”
 - 9.2 Вызов подтвержден (долгое нажатие кнопки)
 - 9.3 Режим Panic от метки (долгое нажатие кнопки в обычном режиме)
 - 9.4 Подтверждение сигнала Panic от Диспетчера
10. Конфигурационный режим устройства
 - 10.1 Вход в конфигурационный режим при наличии сигналов Charge + SW_PB1 (3s)

- 10.2 В конфигурационном режиме необходимо перевести WiFi модуль в режим точки доступа AP+STA (AT команды)
- 10.3 В конфигурационном режиме передаются параметры настройки WiFi модуля посредством специального конфигуратора, подключенного к WiFi модулю по беспроводному интерфейсу, в том числе поле версия ПО с нужными параметрами режимов работы.
- 10.4 Выход из режима конфигурации по пропаданию сигнала Charge
- 11. Обновление прошивки через разъем CN1 (PGC, PGD, RST)
- 12. Обработка UART_2 через разъем CN1 (PGC, PGD, RST)
 - 12.1 Прием посылки, упаковка в JSON, отправка на сервер по WiFi
 - 12.2 Прием пакета с сервера, распаковка, отправка в UART_2

Пример пакета передачи данных на сервер в формате JSON:

```
{
  "MAC": "00:00:00:00:00:01",
  "TagType": "WiFi_tag",
  "ConnectedAP": {
    "BSSID": "00:00:00:00:00:02",
    "SSID": "base-net",
    "Channel": 6,
    "Security": "WPA2"
    "Signal": 65
  },
  "BatteryLife": 75,
  "Status": 0,
  "Temperature": 22.8,
  "ScannedAPs": [
    {
      "BSSID": "00:00:00:00:00:03",
      "SSID": "base-net",
      "Channel": 6,
      "Security": "WPA2",
      "Signal": 65
    },
    {
      "BSSID": "00:00:00:00:00:04",
      "SSID": "base-net",
      "Channel": 6,
      "Security": "WPA2",
      "Signal": 65
    }
  ]
}
```

Описание полей пакета:

Name	Type	Description
MAC	String	The MAC address of the tag.
TagType	String	The tag type (i.e. WT-300, WT-400, etc.). WT-300 is personnel, and WT- 400 is vehicle.

ConnectedAP	Object	An object representing information about the AP that the tag is currently connected to. The AP object fields are defined as:
- BSSID	String	The Basic Service Set Identifier of the AP.
- SSID	String	The Service Set Identifier of the AP.
- Channel	Number	The WLAN channel used by the AP.
- Security	String	The security profile used by the AP.
- Signal	Number	The WIFI signal strength of the AP in % (0-100). Higher value represents a stronger signal.
BatteryLife	Number	The relative state of charge of the tag battery in % (0-100)
Status	Number	The status bytes containing the following bit fields: Bit 0: Panic alarm button activated Bit 1: Battery low Bit 2: Battery charging Bit 3: Charger connected Bit 4: Free-fall/Impact enabled/disabled Bit 5: Free-fall/Impact alarm activated Bit 6: Temperature low alarm enabled/disabled Bit 7: Temperature low alarm Bit 8: Temperature high alarm enabled/disabled Bit 9: Temperature high alarm Bit 10: Inactivity alarm enabled/disabled Bit 11: Inactivity alarm Bit 12: Double Tap enabled/disabled Bit 13: Scanned AP list enabled/disabled Bit 14: System alarm activated Bit 15: System alarm acknowledged Bit 16 - 19: Motion indication. Range 0 – F (0=Not moving, 1=Walking, 2=Running) Bit 20: Future use Bit 21: Future use Bit 22: Future use Bit 23: Future use
Temperature	Number	The tag temperature in °C
ScannedAPs	Array[Object]	The list of AP objects that were scanned in range of the tag. The format of the objects is identical to object format defined above for the ConnectedAP field.

Далее будет более подробное описание каждой функции или режима.