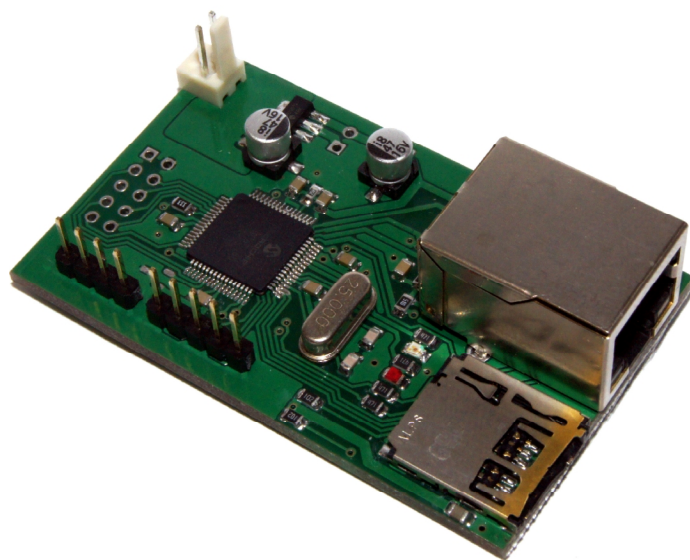


Описание

Отладочная плата **TRT-Ethernet** предназначена для прототипирования устройств с **Ethernet**-интерфейсом или конечного использования в качестве Ethernet/UART-мостов, WEB-терминалов и т.д.

Плата разработана на базе контроллера **PIC18F67J60** со встроенными MAC и PHY уровнями. В качестве прошивки используется адаптированная под топологию платы библиотека **TCP/IP Stack**. Исходные коды, принципиальная схема и разводка платы распространяются бесплатно на сайте <http://www.trt.ru>.



Характеристики:

- Контроллер: PIC18F67J60-I/PT
 - § 128 Кб flash
 - § 3808 RAM
 - § 8 Кб Ethernet-buffer
- Интерфейсы:
 - § Ethernet 10BASE-T
 - § UART до 112500 б/с
 - § слот для microSD-карт на SPI
 - § 8 GPIO
- Поддерживаемые протоколы:
 - § ARP
 - § IP
 - § ICMP
 - § UDP
 - § TCP
 - § DHCP
 - § SNMP
 - § HTTP
 - § FTP
 - § TFTP
- Габариты: 61 x 33 x 17 мм
- Питание:
 - § $U_{пит} = 4 - 9 В$
 - § $I_{пот} < 200 мА$

Работа с платой

Для работы платы в исходном варианте требуется наличие micro-SD-карты в слоте. В исходной прошивке первой инструкцией основного цикла программы стоит бесконечная проверка наличия карты. Для тестирования работы платы при отсутствии micro-SD-карты требуется убрать из исходного проекта строчку проверки карты:

```
while (!MDD_MediaDetect());           // Stops here if SD card can't initialize
```

Строчка располагается в функции *main* модуля *MainDemo.c*.

Поддерживаются карты вплоть до 2 гб (без приставки HC - High Capacity).

В корневой директории также располагается файл тестовой прошивки для проверки работоспособности платы при отсутствии карты micro-SD: *trt-ethernet_no_SD.hex*. После прошивки платы этим файлом при подключении питания и Ethernet-шнура будет определяться новое подключение, плата будет отвечать на ping-запросы по адресу 192.168.0.2.

При наличии в слоте карты памяти и соответствии питания требованиям на плате начинает мигать светодиод. При соединении платы с компьютером должно определяться новое сетевое подключение.

Плата поставляется со следующим функционалом: при получении от браузера запроса GET/ плата отправляет файл *index.htm* с micro-SD-карты. Аналогично отправляются остальные файлы, запрашиваемые браузером.

Исходные сетевые параметры такие:

IP: 192.168.0.2

Subnet Mask: 255.255.255.0

NetBIOS name: trt (т.е. в адресной строке нужно вводить <http://trt>)

Работа с проектом

В архиве *Ethernet.rar* Вы найдете следующие директории:

- *Ethernet* – файлы проекта
- *Server* – демонстрационная веб-страница для *micro-SD*-карты
- *MPFS2* – конфигурационная веб-страница для работы через *MPFS2*

Файл проекта находится в директории Ethernet/TCP/IP Demo App.

В настройках компилятора требуется включить использование расширенного набора команд и *treat "char" as unsigned*. Также необходимо прописать актуальный путь к директориям **h**, **lib**, **lkr** в папке компилятора и указать путь к файлам проекта, отмеченным звездочкой, если таковые обнаружатся.

TCP/IP Stack

В данном разделе будет рассмотрена структура проекта и описаны внесенные изменения. Из всего дерева файлов проекта отметим только наиболее важные:

1. *TCPConfig.h*

Проект построен по модульной архитектуре и подключение/отключение модулей осуществляется в этом файле. Например, комментируя или убирая комментарий со строки «`#define STACK_USE_UART`», можно отключать/включать предстартовую инициализацию по UART'у.

Основные модули:

- `STACK_USE_UART2TCP_BRIDGE` – мост *UART/TCP*
- `STACK_USE_ICMP_SERVER` – пингуем другие адреса
- `STACK_USE_ICMP_CLIENT` – сами отвечаем на пинг
- `STACK_USE_HTTP_SERVER` – веб-сервер с *SD*-карты
- `STACK_USE_HTTP2_SERVER` – веб-сервер с образа в памяти программ (но более совершенный)
- `STACK_USE_SMTP_CLIENT` – шлем *e-mail*
- `STACK_USE_SD_CARD` – подключаем библиотеку *MDD File System* (необходимо для *HTTP*-сервера)

Ниже в этом файле следуют дополнительные настройки, параметры соединения (*IP*, *Subnet Mask*, *NetBIOS name* и т.д.) и распределение памяти по сокетам различных служб. Их также можно подстраивать под пользовательские задачи.

2. *HardwareProfile.h*

В данном файле задаются аппаратные константы и переменные. Файл целиком переписан под данную конкретную плату.

3. *MainDemo.c*

Главный исполнительный файл с main-функцией и основными функциями инициализации.

4. *HTTP.c*

Файл протокола HTTP, ответственный за обработку запросов от браузера и передачу данных и файлов. Изначально работал с образами файлов, хранящихся во флэш-памяти, теперь обращается к SD-карте посредством функций библиотеки **MDD File System**.

5. *FTP.c*

Файл представлен в исходном варианте для ознакомления с технологией работы по FTP.

Помимо перечисленных многие другие файлы проекта подверглись небольшим изменениям, связанным со структурой платы (убраны все ссылки на кнопки, внешний EEPROM и др.). В остальном же проект идентичен оригинальному **TCP/IP Stack v.4.50** и для него справедлива документация данной библиотеки.

Дополнительные возможности

Здесь описаны те файлы и функции, которых не было в оригинальной библиотеке.

1. Уже упомянутая **MDD File System**, которая позволяет считывать и записывать файлы на micro-SD-карту в привычной файловой системе FAT16 или FAT32. В дереве проекта представлена следующими файлами:

- *FS/FSIO.c*
- *FS/FS-SPI.c*
- *FS/FSIO.h*
- *FS/FS-SPI.h*

Подключается библиотека в файле *TCPIPConfig.h* директивой «`#define STACK_USE_SD_CARD`».

2. **Data EEPROM Emulation** – эмуляция EEPROM в памяти программ как альтернатива SPI-флэшу. Используется для энергонезависимого хранения основных настроек (IP, MAC, host name и т.д.), которые могут быть изменены из программы. В линкер-скрипте для хранения EEPROM-данных выделено 1024 байта (CODEPAGE NAME=deeprom). Данные записываются кластерами по 64 байта друг за другом (при этом валидным является только один из них – последний), а при заполнении сектора он целиком стирается. Это обеспечивает сокращение износа ячеек памяти в 16 раз по сравнению со статической записью во флэш. Включает в себя файлы:

- *deeprom.c*
- *deeprom.h*

Включается в файле *TCPIPConfig.h* директивой «`#define PFS_USE_DEEPROM`».

3. Программный выбор TCP порта для режима моста UART/TCP. Демонстрационная веб-страничка (загружаемый образ при включенном HTTP2) позволяет настраивать основные параметры платы, в том числе и порт TCP для режима моста, который раньше был константой. Этот параметр сохраняется при отключении платы (при включенном DEEPROM).

Техническая поддержка

Все вопросы, ошибки, комментарии и предложения просьба направлять по адресу support@trt.ru.