\Firmware - готовый пакет MGFToolsдля программирования образа системы в плату

\App – файлы прикладного web интерфейса

# Программирование платы EV-imx287-nano-x-mbпри помощи USB

Для загрузки П.О. (включая все - ядро и rootfs ) Вам необходимо:

1. Подключить верхний USB порт платы (USB0) к компьютеру с ОС Windowsc помощью USB кабеля.

2. Проводком, закоротить контакт RE на модуле EV-imx287-nano на землю. Еслисмотретьнамодультакчто mini PCI гнездослева, токонтакт RE - этосквозноеотверстие в платемодулярасположенное в нижнемправомуглу, рядом с крепежнымболтом. В качестве земли я выбрал штырек 2 коннектора UEXT1 обозначен GND на принципиальной схеме модуля.

3. Подать питание на плату.

4. Удерживая RE в нуле, нажать кнопку RST на плате

5. Windows в этот момент должна определить новое USBHID устройство

6. Проводок замыкающий RE на землю можно убрать.

7. Запустить программу Mfgtool.exe (в корне MFG\_copy)

8. Убедитесьчтовыбраноправильноедействие, в главномменю Options -> Configuration, в Tab Profiles в таблицесправа в колонке Options должнобыть NAND-Write Linux V3&rootfs. Каждый раз этого делать не нужно будет.

9. Нажать кнопку Start

10. Дождаться сообщения Operationscomplete!,тулбар меняет цвет с синего на зеленый.

11. Выйти из программы, отключить USB кабель и перезагрузить плату

После программирования плата имеет адрес 192.168.0.100

# Расположение файлов

/var/www -HTML/JSфайлы webсервера

# Простые web-тесты на производительность

Тесты для web-сервера lighttpd лежат в отдельной папке: /var/www/public/test и доступны по адресу http://ip\_of\_dev\_board/test

в данной папке размещается весь тестовый контент:  
- page1.html - содержит немного статический текстовой информации.  
- page2.html - содержит немного статический текстовой информации + картинка (File1.bmp).   
- page3.php - PHP-скрипт, который выполняется в FastCGI PHP-движком. Данный скрипт только генерит немного текстовой информации для отображения в браузере  
- page4.php - PHP-скрипт, который открывает файл базы данных SQLITE (production.sqlite), извлекает из таблицы migrations все записи и отображает на странице.  
- page5.php - PHP-скрипт, который открывает файл базы данных SQLITE (production.sqlite), помещает в таблицу migrations одну запись (рендомная строка + число).

Замер производительности выполнения каждого теста делается с помощью утилит wget и time, объеденных в скрипт test.sh. wget выполняет закачку страницы, включая картинки и файлы требуемые для данной страницы (CSS-файлы, JS-файлы и т.д). А утилита time замерет время выполнения программы wget. wget выполняет обращение к web-серверу lighttpd точно также как это делает браузер, но только в пределах сервера).

Чтобы инициировать из браузера выполнение test.sh, была создана страничка index.php, которая содержит ссылки для выполнения всех 5-ти тестов. При клике на ссылку, выполняется нужный тест на сервере, замеряется время работы сервера и затем отображается на странице браузера.  
  
Чтобы браузер и сервер не кешировали страницы для тестов, был подправлен файл lighttpd.conf. Для тестов выполняющихся на стороне сервера это уже не так актуально, но может быть полезным при отладке вместе с браузером.

Утилита wget на основе busybox - очень проста и не годится для тестов. C использованием buildroot была скомпилирована другая (полноценная) утилита wget и положена в папку /usr/local/bin. 