

content_mgmt - Feature #1233

Перевод <https://xdevs.com/article/ddr4s/>

08/22/2015 01:58 PM - tin

Status:	Closed	Start date:	08/22/2015
Priority:	Normal	Due date:	09/23/2015
Assignee:	tin	% Done:	100%
Category:	Article	Estimated time:	1.00 hour
Target version:			

Description

DDR DIMM SPD адаптер для Raspberry Pi

Сегодня практически каждый модуль DDR памяти имеет в своем составе небольшой объем I2C EEPROM памяти для хранения информации о производителе, спецификации, конфигурации и прочих параметров модуля. Наличие этой конфигурационной информации позволяет компьютеру автоматически определять тип памяти, задавать корректные параметры скорости и напряжений интерфейса памяти, в соответствии со спецификацией производителя модуля памяти. Такой механизм работы основан на стандарте JEDEC [Serial presence detection specification](#). Когда нужно изменить информацию SPD используют программные утилиты для модификаций параметров памяти в составе системы. Данные утилиты очень редко доступны разработчику для использования, а также нередко поддерживают лишь ограниченный круг системных плат.

Для решения данной проблемы поставим задачу разработать автономный программатор для чтения/записи SPD.

В качестве базовой платформы выбираем Raspberry Pi. Применение данный процессорного модуля - один из сотни возможных путей решения задачи. Причины выбора модуля: широко распространенная Linux платформа, легкая в освоении и настройке, позволяющая инженерам реализовывать новые возможности встраиваемых ARM платформ. Низкая стоимость и универсальность модуля также позволит использовать его в других проектах, без необходимости дополнительных затрат.

Дополнительное требование к работе программатора: изменения в SPD должны производиться без риска повреждения модуля механически или в результате пайки. Для удовлетворения этого требования необходимо применить стандартный разъем для DIMM. Такой подход позволяет реализовать быстрое подключение к SPD без риска повреждения и без паяного неразрывного соединения.

Основные требования к проекту

- Проект с открытым кодом, под управлением [dev.xdevs.com](#) и ["Mercurial DVCS"](#)
- Полноразмерные DDR3/DDR4 UDIMM/RDIMM разъемы совместимые со стандартными модулями памяти
- Разъем для подключения к порту GPIO Raspberry Pi
- I2C EEPROM интерфейс подключен к интерфейсу I2C1 Raspberry Pi
- Напряжение питания микросхемы SPD +3.3 В с порта Raspberry Pi
- Переключатель питания модулей памяти для замены без отключения питания Raspberry Pi
- Переключатель выбора адреса I2C для подключения нескольких адаптеров к Raspberry Pi
- МК NXP LPC11U37FBD64- в составе адаптера для реализации дополнительных функций, автономного программирования без Raspberry Pi (например, для автоматизации процесса)

Структура проекта

Все проектные файлы, а так же сопутствующая документация отслеживается Распределенной Системой Контроля Версий Mercurial¹.

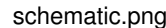
Для работы с проектом вы можете получить дерево проекта, выполнив команду:

```
pre. hg clone http://dev.xdevs.com/hg/ddr4s
```

Также можно просмотреть отдельные фалы в [дереве проекта](#).

Разработка аппаратной части

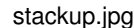
Основа адаптера - простой дизайн с небольшим количеством компонентов. Потратив несколько часов, получаем схему адаптера:



Доступна схема в варианте [PDF-документа](#).

Часть с микроконтроллером не является обязательной и не требуется для работы с RPI. Рисунок печатной платы делался для 4-х слоев, для стандартной печатной платы с использованием материала FR4. Внутренние слои предназначены для питания. Для простоты сборки шелкография компонентов наносится с обеих сторон платы. Большая часть компонентов - для технологии поверхностного монтажа, за исключением разъемов, переключателей и электролитических конденсаторов. Исходные файлы для изготовления печатной платы доступны для загрузки[2]

Внешний вид стека платы:



Для отдельных приложений DIY сборка платы может быть упрощена.



На схеме показаны резисторы подтяжки I2C к питанию RPI, а также переключатель питания. Имеется индикаторный светодиод для отображения наличия питания.

Внешний вид платы адаптера (ручная сборка):



Список компонентов:

- Выключатель питания SW4 (двухпозиционный)
- Резисторы подтяжки интерфейса I2C R10,R9 (2.2 кΩ, 0603 5%)
- вспомогательный резистор R7 (0-10 Ом, 0603 5%)
- конденсаторы развязки C3,C2 (0.1 мкФ 0603 X7R)
- Конденсатор фильтра питания C4 (100-560 мкФ 6.3-16V 7 мм диаметр, шаг выводов 4 мм)
- 2.54mm 26-контактный разъем для RPI порта расширения P1.
- разъем DDR3 240-pin DIMM
- разъем DDR4 288-pin DIMM (при необходимости)

Все компоненты расположены на верхней стороне платы. На нижней стороне платы расположены декоративные светодиоды 0603.



Питание, I2C, SPI и UART подключены к 26 контактному разъему. подключение к порту расширения Raspberry Pi производится с помощью плоского кабеля.



Для подключения нескольких адаптеров необходимо установить адрес с помощью двухпозиционного переключателя с индикацией светодиодами. Такая настройка типична для линий A0, A1, A2 выводов I2C EEPROM.



Микроконтроллер NXP LPC11U37FBD64 оснащен интерфейсом SWD для записи и отладки программного обеспечения. Для работы интерфейсов UART и USB микроконтроллера необходим кварцевый резонатор 12.000 МГц.



Внешний вид комплекта для работы адаптера:



- Модуль памяти DDR3 Corsair UDIMM
- DDR плата адаптера
- Raspberry Pi Модель B с 2GB SD card и 26-контактным 2.54 мм кабелем

Подключение DDR3 модуля памяти к адаптеру:



Подключение DDR4 модуля памяти к адаптеру:



Одновременное подключение обоих типов модулей памяти не рекомендуется, из-за конфликта адресации I2C шины.

Разработка программной части

Рассмотрим пошаговую инструкцию настройки Raspberry Pi для использования в данном проекте. За основу возьмем минимальный образ ОС, для обмена данными с платой будем использовать сетевое подключение с поддержкой ssh.

Образ ОС может быть получен с <http://www.linuxsystems.it/raspbian-wheezy-armhf-raspberry-pi-minimal-image/> или с зеркала http://dev.xdevs.com/projects/rpi/repository/entry/os/raspbian_wheezy_20130923.img.7z

Объем образа 120MB, Для записи на SD card можно использовать, например: "Win32 Disk imager" от gruemaster, tuxinator2009 или любым аналогичным ПО для записи образа на носитель.

Пароль root - "raspberrypi".

Настройка программного обеспечения

Для подключения к RPi по сети необходимо правильно настроить сетевой интерфейс (eth0). В случае если локальная сеть использует DHCP достаточно добавить строку конфигурации в /etc/network/interfaces

```
iface eth0 inet dhcp
```

Если размер SD card более 1ГБ, то может понадобиться дополнительная разметка дискового пространства: Создаем новые разделы, используя fdisk, таблица разделов может выглядеть следующим образом:

```
pre.. root@raspberrypi:~# fdisk /dev/mmcblk0
Command (m for help): p
Disk /dev/mmcblk0: 32.1 GB, 32127320064 bytes
...
Device Boot      Start         End      Blocks   Id  System
/dev/mmcblk0p1    2048         104447    51200    b   W95 FAT32
/dev/mmcblk0p2    104448       1742847   819200   83   Linux
/dev/mmcblk0p3    1742848     1945599   101376   82   Linux swap / Solaris
/dev/mmcblk0p4    1945600     62748671 30401536 83   Linux
```

Для нового раздела задаем тип файловой системы ext4.

```
pre.. root@raspberrypi:~# mkfs.ext4 /dev/mmcblk0p4 -L ummspi2
mke2fs 1.42.5 (29-Jul-2012)
Filesystem label=ummspi2
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
....
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done
```

....

```
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done
```

Убедимся, что изменения внесены в /etc/fstab:

```
pre.. root@raspberrypi:~# cat /etc/fstab
```

```
1. <file system> <mount point> <type> <options> <dump> <pass>
proc /proc proc defaults 0 0
/dev/mmcblk0p1 /boot vfat defaults 0 0
/dev/mmcblk0p3 none swap sw 0 0
/dev/mmcblk0p4 /repo ext4 defaults 0 0
root@raspberrypi:~#
```

Теперь у нас есть раздел /repo тип файловой системы ext4. Монтируем его для использования.

```
pre. root@raspberrypi:~# mount -a
```

Диск готов к использованию.

```
pre.. root@raspberrypi:~# df -h
Filesystem      Size  Used Avail Use% Mounted on
rootfs          788M  566M  222M  72% /
/dev/root       788M  566M  222M  72% /
devtmpfs        212M   0  212M   0% /dev
tmpfs           44M   180K  44M   1% /run
tmpfs           5.0M   0  5.0M   0% /run/lock
tmpfs           108M   0  108M   0% /run/shm
/dev/mmcblk0p1  50M  19M  32M  37% /boot
/dev/mmcblk0p4 29G 172M 27G 1% /repo
```

Хорошим правилом считается - обновление основных пакетов ОС перед установкой нового ПО. Убедитесь в наличии достаточного количества дискового пространства для данных обновлений.

```
pre.. root@raspberrypi:~# aptitude update
Get: 1 http://archive.raspbian.org wheezy Release.gpg [490 B]
Get: 2 http://archive.raspbian.org wheezy Release [14.4 kB]
Get: 3 http://archive.raspbian.org wheezy/main Sources [6262 kB]
70% [3 Sources 4375 kB/6262 kB 70%].....79.8 kB/s 23s
...
Current status: 33 updates [+33], 77 new [+77].
```

Запускаем процесс обновления. Ниже представлен пример.

```
root@raspberrypi:~# aptitude dist-upgrade
The following packages will be upgraded:
 apt apt-utils base-files dmsetup dpkg gnupg gpgv initscripts libapt-inst1.5 libapt-pkg-dev libap
t-pkg4.12 libc-bin libc-dev-bin libc6 libc6-dev libdevmapper1.02.1 libgnutls26
 libssl1.0.0 linux-libc-dev locales lsb-base multiarch-support openssl-client openssl-server perl
 perl-base perl-modules ssh sysv-rc sysvinit sysvinit-utils tzdata wget
The following packages are RECOMMENDED but will NOT be installed:
 bcc gcc gcc-4.4 gcc-4.5 gcc-4.6 gcc-4.7 gnupg-curl libldap-2.4-2 manpages-dev psmisc
33 packages upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
Need to get 35.5 MB of archives. After unpacking 234 kB will be freed.
Do you want to continue? [Y/n/?] Y
...
Setting up ssh (1:6.0p1-4+deb7u1) ...
Setting up perl (5.14.2-21+rpi2+deb7u1) ...
Current status: 0 updates [-33].
```

Автор статьи предпочитает использовать в качестве системного редактора mcedit. Производим замену редактора по умолчанию nano на mcedit:

```
root@raspberrypi:~# update-alternatives --config editor
There are 3 choices for the alternative editor (providing /usr/bin/editor).
 Selection Path Priority Status
 0 /bin/nano 40 auto mode
 1 /bin/nano 40 manual mode
 2 /usr/bin/mcedit 25 manual mode
 3 /usr/bin/vim.tiny 10 manual mode
Press enter to keep the current choice[*], or type selection number: 2
update-alternatives: using /usr/bin/mcedit to provide /usr/bin/editor (editor) in manual mode
```

Основные данные проекта хранятся в локальном репозитории. Для доступа к ним - установим Mercurial:

```
apt-get install mercurial
root@raspberrypi:~# apt-get install mercurial
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
 ca-certificates mercurial-common openssl python python-minimal
Suggested packages:
 qct wish vim emacs kdiff3 kdiff3-qt kompare meld xxdiff tkcvs mgdiff python-mysqldb python-pygme
nts python-openssl python-doc python-tk
The following NEW packages will be installed:
 ca-certificates mercurial mercurial-common openssl python python-minimal
0 upgraded, 6 newly installed, 0 to remove and 0 not upgraded.
Need to get 3514 kB of archives.
After this operation, 10.3 MB of additional disk space will be used.
Do you want to continue [Y/n]? Y
Get:1 http://archive.raspbian.org/raspbian/ wheezy/main python-minimal all 2.7.3-4+deb7u1 [42.8 kB]
...
...
Creating config file /etc/mercurial/hgext.rc with new version
Processing triggers for ca-certificates ...
Updating certificates in /etc/ssl/certs... 158 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d...done.
```

[Краткая статья по использованию Mercurial][<https://xdevs.com/hg-intro/>], [официальная документация][<http://mercurial.selenic.com/guide/>]. //снести вниз статьи, в справочные материалы

Настраиваем доступ к аппаратному интерфейсу I2C
Изменения в /etc/modules

```
i2c-bcm2708
i2c-dev
```

Если используется ядро Linux 3.18 или более новое, то в /boot/config.txt необходимо добавить параметры.

```
dtparam=i2c_arm=on
dtparam=i2c1=on
dtparam=i2c0=on
dtparam=spi=on
```

2. `sudo apt-get install python-smbus`

```
root@raspberrypi:~# apt-get install python-smbus
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  i2c-tools
Suggested packages:
  libi2c-dev
The following NEW packages will be installed:
  i2c-tools python-smbus
0 upgraded, 2 newly installed, 0 to remove and 32 not upgraded.
Need to get 71.0 kB of archives.
After this operation, 317 kB of additional disk space will be used.
Do you want to continue [Y/n]? Y
Get:1 http://archive.raspbian.org/raspbian/ wheezy/main i2c-tools armhf 3.1.0-2 [59.5 kB]
Get:2 http://archive.raspbian.org/raspbian/ wheezy/main python-smbus armhf 3.1.0-2 [11.5 kB]
Fetched 71.0 kB in 1s (36.0 kB/s)
Selecting previously unselected package i2c-tools.
(Reading database ... 22286 files and directories currently installed.)
Unpacking i2c-tools (from ../i2c-tools_3.1.0-2_armhf.deb) ...
Selecting previously unselected package python-smbus.
Unpacking python-smbus (from ../python-smbus_3.1.0-2_armhf.deb) ...
Processing triggers for man-db ...
Setting up i2c-tools (3.1.0-2) ...
Setting up python-smbus (3.1.0-2) ...
```

3. `sudo apt-get install i2c-tools`

```
root@raspberrypi:~# apt-get install i2c-tools
Reading package lists... Done
Building dependency tree
Reading state information... Done
i2c-tools is already the newest version.
i2c-tools set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 32 not upgraded.
```

Теперь мы можем подключить адаптер и проверить работу интерфейса I2C

```
i2cdetect -y 1
```

Если модуль памяти не установлен, либо установлен неправильно, устройства на шине обнаружены не будут:

```
root@raspberrypi:~# i2cdetect -y 1
   0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
10:  -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
20:  -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
30:  -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
40:  -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
50:  -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
60:  -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
70:  -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
```

5. Минимальный образ linux может не иметь установленный компилятор gcc, установим компилятор перед началом использования кода программ.

```
apt-get install gcc
apt-get install make
```

В случае ошибки вида:

```
sudo: unable to resolve host raspberrypi
```

Добавим строку /etc/hosts

```
127.0.0.1 raspberrypi
```

Если библиотека wiringPi не установлена - ее необходимо установить:

```
pre.. root@raspberrypi:/rpd/4s/fw/wiringPi# ./build
wiringPi Build script =====
WiringPi Library
[UnInstall]
[Compile] wiringPi.c
...
[Compile] wiringPiSPI.c
[Compile] wiringPiI2C.c
...
GPIO Utility
[Compile] gpio.c
gpio.c:85:12: warning: 'decodePin' defined but not used [-Wunused-function]
[Compile] extensions.c
[Compile] readall.c
[Link]
[Install]
All Done.
NOTE: This is wiringPi v2, and if you need to use the lcd, Piface,
Gertboard, MaxDetext, etc. routines then you must change your
compile scripts to add -lwiringPiDev
```

7. Проверяем работу программ. Raspberry Pi подключена к терминалу Putty:

```
soft.jpg
```

8. Использование программ:

- /comp.sh* – скрипт запуска gcc компилятора с spd_prog_both.c для получения выполняемого файла spd_prog (*nix)
- /check.sh* – скрипт запуска gcc компилятора с checksum.c для получения выполняемого файла checksum (*nix)
- /spd_prog* – запуск приложения для чтения / записи DDR3/DDR4 SPD. Приложение работает только с бинарными файлами.
- /checksum* – запуск утилиты расчета контрольной суммы и проверки CRC16 полей в двоичном файле DDR3/DDR4. Утилита запускается с параметром -f [filename]

9. Пример расчета контрольной суммы:

DDR3 дамп пример:

```
pre.. $:/repo/4s/fw/Pi$ ./checksum -f ddr3.bin
/***** xDevs.com DDR4s Project *****/
DDR3/DDR4 dump checksum check tool | calculates CRC16 for input binary
Usage: checksum -f [dump_filename]
/*****
File DDR3 CRC16 = 600E | byte 126 = E | byte 127 = 60
Calculated DDR3 CRC16 = 600E | byte 126 = E | byte 127 = 60
CRC in dump is correct
```

DDR4 дамп пример:

```
pre.. $:/repo/4s/fw/Pi$ ./checksum -f ddr4.bin
/***** xDevs.com DDR4s Project *****/
DDR3/DDR4 dump checksum check tool | calculates CRC16 for input binary
Usage: checksum -f [dump_filename]
/*****
+ DDR4 mode enabled, 512 bytes parsed
File DDR4 CRC16 = E752 | byte 126 = 52 | byte 127 = E7
```

File vendor section DDR4 CRC16 = 0 | byte 382 = 0 | byte 383 = 0
Calculated DDR4 CRC16 = E752 | byte 126 = 52 | byte 127 = E7
CRC in dump is correct
tin@tin-lap:/repo/ddr4s/fw/Pi\$

Справочная литература:

[JEDEC Standard No. 21C : Serial Presence Detect \(SPD\), General Standard](#)
[JEDEC Registration - DDR4 DIMM PTH 288 Pin Socket Outline, 0.85 mm Pitch, SKT](#)
[JEDEC 288 PIN DDR4 DIMM, 0.85 mm PITCH](#)
[JEDEC Definitions of the EE1004-v 4 Kbit Serial Presence Detect \(SPD\) EEPROM and TSE2004av 4 Kbit SPD EEPROM with Temperature Sensor \(TS\) for Memory Module Applications](#)
[JEDEC DDR4 UDIMM Design Specification Annex D](#)
[JEDEC Proposed Annex for DDR4 UDIMM, R/C B](#)
[JEDEC Standard No. 21C : Annex L: Serial Presence Detect \(SPD\) for DDR4 SDRAM Modules](#)
[JEDEC Standard No. 21C : Serial Presence Detect \(SPD\) for DDR3 SDRAM Modules DDR3 SPD Document Release 6](#)
[JEDEC Standard No. 21C : Memory Module Nomenclature](#)
[JEDEC Standard No. 21C : Definition of the TSE2002av Serial Presence Detect \(SPD\) EEPROM with Temperature Sensor \(TS\) for Memory Module Applications](#)

Версия данного документа:

Rev 10, Initial release to public, added low-cost option schematics
Rev 11, Added extra settings for Kernel 3.18+ to enable I2C operation
Rev 12, Добавлена версия на русском языке, стилистические правки, добавлено оглавление

¹ Это бесплатная и хорошо поддерживаемая система контроля версий с поддержкой синтаксиса, аналогично GIT и SVN

²[RS274X GERBER-файлы для изготовления печатной платы](#)

History

#1 - 08/24/2015 04:42 AM - tin

неупаль?

#2 - 08/26/2015 04:14 AM - tin

- Priority changed from Normal to Low

#3 - 08/27/2015 07:22 PM - tin

и неопсал.

#4 - 09/07/2015 02:21 AM - tin

Ping

#5 - 09/21/2015 12:13 AM - Izerg

- Description updated

- Due date changed from 08/23/2015 to 09/23/2015

- Status changed from New to In Progress

- Priority changed from Low to Normal

- % Done changed from 0 to 10

- Estimated time set to 1.00 h

#6 - 09/21/2015 12:41 AM - Izerg

- Description updated

#7 - 09/21/2015 12:46 AM - Izerg

- Description updated

#8 - 09/21/2015 12:50 AM - Izerg

- Description updated

#9 - 09/21/2015 12:52 AM - Izerg

- Description updated

#10 - 09/21/2015 12:53 AM - Izerg

- Description updated

#11 - 09/21/2015 01:08 AM - Izerg

- Description updated

#12 - 09/21/2015 08:47 AM - Izerg

- Description updated

#13 - 09/22/2015 05:00 AM - tin

А дальше чего будем?

#14 - 09/27/2015 10:20 AM - Izerg

В воскресенье продолжу.

#15 - 09/28/2015 12:43 PM - tin

Вот и понедельник узакончился...

#16 - 09/30/2015 02:16 AM - tin

И вторник тоже :)

#17 - 10/01/2015 12:15 PM - tin

И даже четверг. Тушите свет, господа!

#18 - 10/05/2015 08:31 PM - Izerg

- Description updated

- % Done changed from 10 to 20

#19 - 10/05/2015 08:42 PM - Izerg

- Description updated

#20 - 10/05/2015 09:02 PM - Izerg

- Description updated

#21 - 10/05/2015 09:03 PM - Izerg

- Description updated

#22 - 10/05/2015 09:28 PM - Izerg

- Description updated

у программеров есть термин "код-спагетти"
тут статья шотландское одеяло - лоскуток на лоскутке. мосг от нее ломается. закончу "механический перевод" - нужно будет делать вычитку и форматирование, в таком виде как исходный вариант - оно не читаемое.

#23 - 10/06/2015 07:46 AM - tin

Можешь без картинок для начала. Обычно я пишу текст, а затем вставляю или фоткаю и затем вставляю картинки, иллюстрирующие текст. (исключение - обзор, там наоборот, что видим то и описываем, хотя тоже на вкус и цвет разные фломастеры).

Можешь код исходный постить включив в тег

```
<pre></pre>
```

Это значит что textile ничего парсить не будет внутри тега.

#24 - 10/07/2015 07:23 PM - Izerg

без картинок - не могу,
мне нужно видеть всю картину ... я ж перечитываю постоянно, смотрю - как оно получилось, что нужно править, а что на своем месте

pre - я пробовал - на нем как раз и спотыкается при вложенности

#25 - 10/08/2015 01:03 AM - tin

Пример в студию? Будем смотреть.

#26 - 10/11/2015 07:14 PM - Izerg

- *Description updated*

- % Done changed from 20 to 40

#27 - 10/11/2015 07:32 PM - Izerg

- *Description updated*

#28 - 10/11/2015 08:15 PM - Izerg

- *Description updated*

- *Status changed from In Progress to Feedback*

- *Assignee changed from Izerg to tin*

Статью нужно вычитывать, некоторые места - переписывать.
Прыгаешь с темы на тему, некоторые - не раскрыты, некоторые - лишние в данной статье (вынести в справочные материалы или под_статьи)

по факту - статья полезна но сильно выносит мозг.

по факту 2 - кто-то вконец устал писать :) начало - подробное , конец - очень сжатый.

Кликабельные картинки?

Работа с именованными ссылками?

#29 - 10/11/2015 08:15 PM - Izerg

- % Done changed from 40 to 100

#30 - 10/12/2015 04:27 AM - tin

100% :)

Кликабельные картинки : " ! ссылка на картинку которая будет отображаться !": ссылка на открываемую картинку или файл
Именованные ссылки:

В содержании

" название параграфа или ссылки " : #имя_параграфа

А сам параграф маркируется как

h2. `%{intro}` Введение%

#31 - 10/13/2015 10:10 AM - tin

1. Чтобы у нас не случилось "Длкаяльная сеть" стоит прогонять текст по F7 в ворде :)
2. Единицы измерения также переводим на русский, а то коряво получается читать +3.3V и 40 kOhm вместо +3.3 В и 40 кОм. Омегу можно вставлять

`Ω`;

так.

3. Некоторые вещи не переводы, например Software design

4. Параграфы? Их нету... После каждого параграфа, пустая строки. Иначе как ты говоришь - вынос мозга. Текстиль тоже с ума сходит

5. На будущее - вместо < rge>ляляля пишем rge. однострочный блок, или если многострочный кусок то rge.. кусок, перенос, кусок, деренос, кусок, и после окончания - пустая строка и следующий параграф начинается с r. текст

6. Хедеры обозначаются h2. Текст, ну или h3. Текст пустые строки перед и после

7. Сноски - ляляля текст-текст-текст¹ и потом внизу пишем fn1. ляляля сноски

смотрим <https://xdevs.com/article/>

#32 - 10/13/2015 10:20 AM - tin

- *Description updated*

- *Assignee changed from tin to Izerg*

#33 - 10/13/2015 10:21 AM - tin

[смотрим дифф](#) тут

#34 - 10/13/2015 08:48 PM - Izerg

п. 1 очепятки - принято

п. 1 единицы измерения - принято

п. 1 принято, с оговоркой - было предупреждение о скелетности перевода

п. 1 принято, параграфы см. выше п.1

п.1 rge - принято на тестирование

п.1 принято

п.1 принято

финальный результат просмотрен, в общем - возражений не имеется, за исключением оформления ссылок.

Диф - просмотрен, готов оспорить половину стилистики, но учитывая скелетность перевода - могу этого не делать.

#35 - 10/14/2015 10:25 AM - tin

Мм-мэ...

:) Результат не совсем финальный финальный, это больше бета. Форматирование к стилистике отношения не имеет, хотя принимается замечание. Заметка на следующую тему статьи - предлагаю тебе вместо переводов сделать статью "Как использовать textile для написания статей". Почему тебе? Потому что я его уже более менее освоил, и вопросы "новичка" уже задать не смогу. А тебе в самый раз, мол выделяем так то, оформляем так то. И да, текстиль на сайте и текстиль в редмайне немного разные (в редмайне корявый, ибо редмайн свои еще теги парсит).

В качестве итога - будет темплейт для написания будущих статей, мол содержание, шапки, заголовки, картинки. Я для вставки картинкой например [давно уже питоновый генератор использую](#) что превращает вставку картинку в три клика мышкой.

По ссылкам - я бы их все же вставлял как есть, гипертекстом сразу в тексте по ходу дела. Мы же не журнальную статью на бумаге пишем, странно использовать бумажный подход и отказываться от интерактивных возможностей WEВа.

Например когда я сам читаю чужие статьи, часто приходится делать дополнительные телодвижения в гугл и вбивать ключевые слова или там даташит на описываемый чип гуглить, хотя если ссылка в тексте - нажал и читай себе.

#36 - 10/17/2015 10:44 AM - Izerg

а ты связь испортил,

вот - жду пока починишь

#37 - 10/22/2015 06:53 PM - Izerg

- *Status changed from Feedback to Resolved*

- *Assignee changed from Izerg to tin*

статья опубликована, иссуй не закрыт.

я плохо понимаю что происходит.

начинать текстиль и шаблон ?

#38 - 10/23/2015 01:32 AM - tin

Работа происходит, выходные завтра...что испортил, куда испортил тоже непонятно :)

#39 - 10/25/2015 08:58 AM - tin

- *Status changed from Resolved to Feedback*

#40 - 10/25/2015 09:00 AM - tin

Статья оформляется и готовится к публиковке.

Финальный вариант будет приведен в ссылке.

#41 - 10/25/2015 01:29 PM - tin

- *Status changed from Feedback to Closed*

#42 - 09/15/2016 09:46 AM - tin

- *Description updated*