Знакомство с Сименс

Сходили в подразделение компании Siemens, которое специализируется на встраиваемых устройствах, под управлением ОС Linux.

Здесь хочу описать там полученную информацию. Там нас встретили два прикольных чувака, одиного из которых звали Максим, а другого Дима.

Вопросы и ответы

Сначала Максим вкратце рассказал о Siemens, о подразделении Siemens в Санкт-Петербурге. Затем он описал что-такое SLIND и с чем его едят. (:

В процессе знакомства мы задавали некоторые вопросы, которые я попытался здесь вспомнить и написать

- Вопрос: зачем каждый раз перезаливать rootfs на устройство, не сделав возможности установить на него менеджер пакетов?
- Ответ: Устройство само по себе может быть ограничено в ресурсах. Например, на нем может быть read-only файловая система. И чтобы обновить rootfs, необходимо воспользоваться “перепрошивкой” и залить всю файловую-систему целиком

- Вопрос: Когда мы ставили SLIND мы не смогли установить кроссскомпилятор для архитектуры arm?
- Ответ: это нормально ;)

- Вопрос: Когда у вас будет кросс-компилятор для arm?
- Ответ: скорее всего он будет в конце следующей недели.

- Вопрос: Когда мы ставили SLIND мы не смогли поставить пакет maintainer-core, хотя в Вики на сайте он описан.
- Ответ: Это тоже нормально, он канул в лету (:

Процесс разработки для/в SLIND

Дима продемонстрировал нам процесс разработки для SLIND.

Он сделал следующие шаги:

1. Открыл Eclipse с CDT, в котором был создан проект SLIND[1)]. Данный проект содержал простейщий файлик с программой HelloWorld ((:
2. при помощи *каких-то* менюшек, которые предоставляются плагином, он выбрал для компилирования компилятор под mips, который стоит в rootfs SLIND-а, т.е. в том, который мы получили при помощи slindctl setup; slindctl chroot; apt-get install gcc-.... (для mips)
3. далее он откомпилировал программу в бинарник, при помощи чего-то (чего конкретно не заметил). Однако компилирование он запусил прямо в эклипсе
4. открыл консоль и набрал file hello. После чего увидел что файл hello – это исполняемый

файл elf32 для архитектуры mips.
5. сказал что у него где-то запущена QEmu и “проброщен” порт на локальную машину
6. руками скопировал файлик на целевую платформу (похоже что в QEmu)
7. запустил там GDB–Server и судя по всему как раз порт этого GDB–Server-а он и “пробросил” на локальную машину
8. вновь открыл Eclipse и настроил стандартный отладчик GDB, указав для него тот самый “проброшенный” порт
9. настроив отладчик Eclipse запустил отладку и радовался жизни, показывая в отладчике ассемблерный код, не похожий на i386 (:

Таким образом сейчас плагин для Eclipse может:

- создавать SLIND проект
- собирать (??) SLIND проект
- удаленно(на целевом утсройстве) отлаживать программы.

[1)]
был применен существующий плагин для Eclipse

From:
http://wiki.osll.ru/ - **Open Source & Linux Lab**

Permanent link:
http://wiki.osll.ru/doku.php/etc:users:zps:acquaintance_with_siemens

Last update: **2016/08/08 20:53**