Настройка вебкамеры для работы с freebsd

Автор: vp.
Оригинал : http://www.lissyara.su/articles/freebsd/tuning/webcam/

Задача - подключить камеру к freebsd. Для начала нам необходимо просто подключить камеру к freebsd и заставить ее работать.

Итак, на сервере используется freebsd 7.0.
После подключения камеры:# usbdevs -v
Controller /dev/usb4:
addr 1: full speed, self powered, config 1, UHCI root hub(0x0000), Intel(0x0000), rev 1.00
port 1 addr 2: full speed, power 100 mA, config 1, product 0x08d7(0x08d7), vendor 0x046d(0x046d), rev 1.00

dmesg | grep 0x046d
ugen0: <vendor 0x046d product 0x08d7, class 0/0, rev 1.10/1.00, addr 2> on uhub4

vendor 0x046d - говорит о производителе камеры(Logitech)
product 0x08d7 - это модель камеры

Поиск в сети решения для работы с камерой дало следующие результаты,
* существует порт graphics/spcaview, который позволяет без лишних наворотов и зависимостей снимать с камеры jpeg изображения, а так же работать с mjpeg форматом(потоковый jpgeg)
* graphics/qcamview - не подошел по причине отсутствия драйвера для конкретной модели и привязки к X11
* multimedia/pwcbsd - требует video4linux и совместимости с linux'ом

Из-за того, что qcamview и pwcbsd требую слишком много стороннего софта(в частности X11 и/или поддержки linux на бинарном уровне) для всего-лишь снятия изображения с камеры, решено было использовать graphics/spcaview

Т.к. мы поставили целью избавиться от зависимости от X11, перед установкой порта необходимо убедиться, что в /etc/make.conf содержиться строка вида "WITHOUT_X11=yes", что позволит собирать программы без библиотек X11(например, полезно для GD)# cat /etc/make.conf | grep 'WITHOUT_X11=yes' || echo "WITHOUT_X11=yes" >> /etc/make.conf

Устанавливаем порт, но не через # pkg_add -r graphics/spcaview

т.к. как выяснилось в последствии, поддержки текущей камеры нет в программе, а компилируем порт из исходников:# cd /usr/ports/graphics/spcaview
make && make install && rehash

При работе freebsd с камерами создается специальное устройство ugenX(например, /dev/ugen0)
Проверяем поддержку камеры# spca5shot -d /dev/ugen0 -i
Not found SPCA50x based usb camera, or Permission denied

Поддержки камеры нет, увы. Но мы ее добавим! Находим файл с данными о камерах и просматриваем его на предмет наличия поддержки камеры производителя 0x046d:# less /usr/ports/graphics/spcaview/work/spca5xx-20060402/drivers/usb/spca5xx.c | grep 0x046d
{USB_DEVICE(0x046d, 0x0960)}, /* Logitech Inc. ClickSmart 420 */
{USB_DEVICE(0x046d, 0x0901)}, /* Logitech Inc. ClickSmart 510 */
{USB_DEVICE(0x046d, 0x0890)}, /* Logitech QuickCam traveler */
{USB_DEVICE(0x046d, 0x0900)}, /* Logitech Inc. ClickSmart 310 */
{USB_DEVICE(0x046d, 0x0905)}, /* Logitech ClickSmart820 */
{USB_DEVICE(0x046d, 0x08a2)}, /* Labtec Webcam Pro */
{USB_DEVICE(0x046d, 0x0921)}, /* Labtec Webcam */
{USB_DEVICE(0x046d, 0x0920)}, /* QC Express */
{USB_DEVICE(0x046d, 0x0928)}, /* Logitech QC Express Etch2 */
{USB_DEVICE(0x046d, 0x092a)}, /* Logitech QC for Notebook */
{USB_DEVICE(0x046d, 0x08a0)}, /* Logitech QC IM */
{USB_DEVICE(0x046d, 0x08a1)}, /* Logitech QC IM 0x08A1 +sound */
{USB_DEVICE(0x046d, 0x08a3)}, /* Logitech QC Chat */
{USB_DEVICE(0x046d, 0x08b9)}, /* Logitech QC IM ??? */
{USB_DEVICE(0x046d, 0x0929)}, /* Labtec Webcam Elch2 */
{USB_DEVICE(0x046d, 0x092c)}, /* Logitech QC chat Elch2 */
{USB_DEVICE(0x046d, 0x08ae)}, /* Logitech QuickCam for Notebooks */
{USB_DEVICE(0x046d, 0x092b)}, /* Labtec Webcam Plus */
{USB_DEVICE(0x046d, 0x08ad)}, /* Logitech QCCommunicate STX */
{USB_DEVICE(0x046d, 0x08a9)}, /* Logitech Notebook Deluxe */
{USB_DEVICE(0x046d, 0x08aa)}, /* Labtec Webcam Notebook */
{USB_DEVICE(0x046d, 0x08a6)}, /* Logitech QCim */
case 0x046d: /* Logitech Labtec */

Как можно заметить, поддержки нашей камеры нет в файле
Поиск в сети по 0x08d7 модели камеры показал, что такая поддержка имеется в драйвере Logitech видеокамер gspcav1(http://mxhaard.free.fr/spca5xx.html). gspcav1 - это общий файл поддержки видеокамер для *nix систем, однако в скомпилированном виде он требует бинарной совместимости с linux, что нам не подходит.

Т.к. порт graphics/spcaview основан именно на этом драйвере, скачиваем исходники

драйвера для анализа http://mxhaard.free.fr/spca50x/Download/gspcav1-20071224.tar.gz. Нас интересует файлы gspca_core.c и gspca.h

Анализ исходника gspca_core.c дает следующий код по модели 0x08d7:
{USB_DEVICE(0x046d, 0x08d7)}, /* Logitech QCam STX */

и далее case 0x08d7:
spca50x->desc = LogitechQCCommunicateSTX;
spca50x->bridge = BRIDGE_ZC3XX;
spca50x->sensor = SENSOR_HV7131C;
break;

Осталось добавить эти данные в драйвер spca5xx.c из нашего порта graphics/spcaview и, возможно, появиться поддержка камеры. Но прежде, нужно не забыть стереть уже скомпилированные данные:# cd /usr/ports/graphics/spcaview
make clean

Теперь наша задача внести изменения в исходник драйвера и откомпилировать его. Применяем патчи но не компилируем исходник# make patch

Далее редактируем /usr/ports/graphics/spcaview/work/spca5xx-20060402/drivers/usb/spca5xx.c любимым редактором и вносим следующие изменения:
Добавляем в секцию устройства 0x046d {USB_DEVICE(0x046d, 0x08d7)}, /* Logitech QCam STX */

После чего исходный текст будет выглядеть примерно так: {USB_DEVICE(0x046d, 0x08ad)}, /* Logitech QCCommunicate STX */
{USB_DEVICE(0x046d, 0x08a9)}, /* Logitech Notebook Deluxe */
{USB_DEVICE(0x046d, 0x08d7)}, /* Logitech QCam STX */
{USB_DEVICE(0x046d, 0x08aa)}, /* Labtec Webcam Notebook */

Теперь нам необходимо добавить конфигурацию камеры, находим в исходном файле секцию камер Logitech: case 0x046d: /* Logitech Labtec */
case 0x041E: /* Creative cameras */
switch (product) {

Далее идет перечисление всех известных камер Logitech, нам необходимо создать новую секцию и добавить в spca5xx.c данные из драйвера gspca_core.c: case 0x08d7:

```
 spca50x->desc = LogitechQCCommunicateSTX;
 spca50x->bridge = BRIDGE_ZC3XX;
 spca50x->sensor = SENSOR_HV7131C;
 break;
```

Но это еще не все. Исходный текст spca5xx.c показывает, что для freebsd используются более расширенные настройки! Для добавления необходимой информации ищем какую-нибудь запись с сенсором SENSOR_HV7131C, например: case 0x08a6:

```
 spca50x->desc = LogitechQCim;
 spca50x->bridge = BRIDGE_ZC3XX;
 spca50x->sensor = SENSOR_HV7131C;
 spca50x->header_len = SPCA50X_OFFSET_DATA;
 spca50x->i2c_ctrl_reg = SPCA50X_REG_I2C_CTRL;
 spca50x->i2c_base = 0;
 spca50x->i2c_trigger_on_write = 0;
 spca50x->cameratype = JPGH;
 info("USB SPCA5XX camera found. Logitech QC IM ");
 break;
```

И копируем недостающие данные в нашу секцию, внеся небольшие изменения в название: case 0x08d7:

```
 spca50x->desc = LogitechQCCommunicateSTX;
 spca50x->bridge = BRIDGE_ZC3XX;
 spca50x->sensor = SENSOR_HV7131C;
 spca50x->header_len = SPCA50X_OFFSET_DATA;
 spca50x->i2c_ctrl_reg = SPCA50X_REG_I2C_CTRL;
 spca50x->i2c_base = 0;
 spca50x->i2c_trigger_on_write = 0;
 spca50x->cameratype = JPGH;
 info("USB SPCA5XX camera found. Logitech QCam STX ");
 break;
```

Все! Осталось скомпилировать и установить драйвер:# cd /usr/ports/graphics/spcaview
make && make install && rehash

Теперь проверим работу нашего драйвера# spca5shot -i -d /dev/ugen0
USB SPCA5XX camera found. Logitech QCam STX

[spca5xx_probe:8279] Camera type JPEG
[zc3xx_config:530] Find Sensor HV7131R(c)
[spca5xx_getcapability:2242] maxw 640 maxh 480 minw 176 minh 144

--- video capability ---
name: Logitech QuickCam Communicate Stype: 1
channels: 1
maxwidth: 640
maxheight: 480
minwidth: 176
minheight: 144
--- size and format ---
size: 640x480 352x288 320x240 176x144 160x120
native input format: JPEG
output format: 24bit RGB
--- video picture (initial value) ---
brightness: 32768
hue: 0
colour: 0
contrast: 32768
whiteness: 0
depth: 24
palette: 4

Done.

Ура! Работает! Теперь можно делать снимки с камеры используя команду spca5shot, так же можно настроить доступ к устройству /dev/ugen0 не из-под рута и модифицировать программу онлайн-просмотра камеры, но об этом в другой раз.
PS Так же можно подключить отдельные камеры, которых нет в исходниках, однако, возможно, потребуется эксперементировать с настройками.