XinuPi: Porting a Lightweight Educational Operating System to the Raspberry Pi

Eric Biggers and Dr. Dennis Brylow (mentor)

**Motivation**
Operating systems education suffers from a lack of easily accessible platforms and the high complexity of modern operating systems such as Linux. As a result, courses often use textbooks or simulations rather than real operating systems and hardware; this is not as helpful to students.

**Raspberry Pi**
The Raspberry Pi is a tiny, $35 computer designed for educational use. The latest version has 512MB of RAM, an ARM CPU, a GPU, HDMI output, 2 USB ports, an Ethernet port, and SD card interface. It normally runs GNU/Linux. To use, users must insert an SD card, apply power, and connect other devices such as a keyboard and monitor.

**Embedded Xinu**
Embedded Xinu is a re-implementation of the classic "Xinu" design for modern RISC architectures. It is a simple UNIX-like operating system designed for educational use. Embedded Xinu contains several orders of magnitude fewer lines of code than Linux-based software stacks, which have millions of lines of code.

**USB Support**
Like many modern PCs, the Raspberry Pi relies heavily on USB in order to attach devices such as keyboards and mice. In addition, the Ethernet adapter on the Model B is internally attached to the USB. Although USB is very complicated, we have implemented a relatively simple USB subsystem for Embedded Xinu.

**USB Host Controller**
The primary challenge of supporting USB on the Raspberry Pi is driving the nonstandard, undocumented Synopsys USB Controller. However, we have implemented a working driver for this hardware that is ~20 times smaller than the corresponding Linux driver. We plan to fully document XinuPi on the Embedded Xinu wiki (http://xinu.mscs.mu.edu), including previously undocumented hardware, our code, and instructions for a lab setup.

**Future and Other Work**
Farzeen and Tyler’s posters describe other aspects of the XinuPi port. We also had to implement low-level functionality such as interrupt handling, context switching, and a UART driver. We plan to fully document XinuPi on the Embedded Xinu wiki (http://xinu.mscs.mu.edu), including previously undocumented hardware, our code, and instructions for a lab setup.

**Ethernet support**
The Raspberry Pi model B has an integrated USB Ethernet Adapter (SMSC LAN9512), which, although not documented, we wrote a driver for. As a result, networking is now supported on XinuPi and it can be used for networking courses. Alternatively, XinuPi can be set up as a back-end network bootloader that loads experimental student kernels in the same way as in the current Embedded Xinu lab (which uses Linksys wireless routers).

**USB Ethernet Adapters**
USB Ethernet adapters like the SMSC LAN9512 work by sending and receiving packets over the USB bus via USB bulk transfers.

---

USA: 134217728 bytes physical memory.
32768 bytes reserved system area.
3776384 bytes Xinu code.
65536 bytes stack space.
130343040 bytes heap space.

---

This work was supported by National Science Foundation grant CCF-1063041.