

deJovi Presents:

SOS

Simple Operating System

A CPSC-321 Project

Devon Harker
Josh Haskins
Vincent Tennant

Project Description

Our initial goal was to implement a file system within the SAM4S. We wanted to integrate the sd card reader from the I/O1 Xplained Pro expansion board. We would have allowed the user to add applications to the sd card, load them into the board's RAM, and execute them using the GUI. This proved troublesome and due to time constraints we reverted to sticking with what we knew. Our plan changed and we implemented all hardware calls through system calls, changed all processes to user mode, and created some example processes. We started by implementing all hardware calls into system calls to prevent application from accessing the hardware directly. Everything from button presses, turning lights on and off, writing to the screen, and delaying execution were implemented in system calls. We moved all non-operating system processes into user mode once the board is initialized. Finally we used C structures to represent multiple processes running on the same system.

Solution

Supervisor calls were used for all hardware related functions. The number of arguments for the supervisor calls ranged from zero to two, depending on what was required.

Threads were used to separate concepts; for example, one thread was dedicated to the menu/GUI, one for everything to do with the temperature sensor, and another thread for everything to do with the light sensor. All of the threads execute in unprivileged mode.

A simple GUI allows the user to determine what information should be displayed. That is, the user can chose to see the current temperature, the current light value, both, or neither. The threads for temperature and light are designed to reduce the frequency of crashes by not pinging their respective sensors at every opportunity.

To schedule the threads, we used the FIFO/RR hybrid that we learned as part of

the labs. All threads are stored in a queue and each thread is given a short time to execute before control is passed to the next thread in the queue.

Execution

To use this project an ATSAM4S-XSTK MCU board is required in addition to the I/O1 Xplained Pro and the OLED1 Xplained Pro expansion boards. Atmel Studio is required to load onto the board, our current project was created in version 6.1 update 2.0 (build 2730). The main program is simply called *main.c*.

To use the operating system on the SAM4S, simply follow the on screen instructions displayed on the lower line of the screen on device. When on the appropriate screen, displaying the current light and temperature can be toggled by clicking middle button. For an unknown reason, the system occasionally ignores single and double button pushes. Triple clicking has mostly consistent success.

Atmel Studio is available for download from [here](#).

Possible Issues/Glitches

During execution, the system may randomly freeze, we have been unable to reproduce this on a regular basis. We believe this has to do with the scheduler, as sometimes it will unfreeze and allow execution to continue as normal.

Also the light value displayed on screen may not refresh for up to 3 seconds at random intervals. We believe this likely has to do with the thread being starved.

Division of Work

We found it difficult to split up the work due to the nature of the project. Ideas for. This was mainly due to the fact that there was one board and that Atmel Studio required Windows and was somewhat finicky to setup. What we ended up doing was

sitting as a group at a table and coding on one computer. Due to the way the project was behaving, having multiple persons working at the same time would have resulted in more bugs and delays. the project were decided in a group setting and had many revisions made.

Experience

Overall the project was a great learning experience and we did enjoy it. Rafael did provide a lot of help to the groups that asked for it, including us. If there was a lab, or a working sample of a file system on the SAM4S this would have aided us and most likely all other groups in properly implementing a file system in the projects.