

ENSC 351 Final Project

Group 18

Tahsin Alam 301233675

Reese Erickson 301225675

Michael Chyziak 301170717

Jason Liu 301224956

Timer Driver

- Sets up ioctl, and async interrupt handling capability
 - Timer driver communication facilitated by ioctl()
 - **ioctl slower than mmap, but mmap() unnecessary because timer only reads/writes 4 bytes at a time**
 - Uses a timer_ioctl_data struct, with offset and data values
 - Define macros for register offsets (control, load, and timer registers)
 - Only two operations possible, READ and WRITE.
 - WRITE protection on TIMER_REG done by driver (attempting to write to the read-only register will fail)
- Wrappers in UART_interface.h/c create easier setup
- setup_timer_interval does all ioctls to the timer
 - set up asynchronous SIGIOs coming from the timer at a fixed interval in milliseconds of the specified argument (constantly).
 - exit_interface turns off the timer, should be used whenever user program wants to exit

VGA Driver

- 640 x 480 pixels for vga
- Pixels -- u32 with RGBA values each 2 bytes
- Mmap allows user access to virtual memory for vga physical memory
- Non-cached mmap -- not as safe, but a faster way to mmap, and faster than ioctl
- Reading frame 60 times/sec
- Memory allocated in kernel space
- Views of memory must be consistent
- Implement mmap--> kernel space frame buffer to user space app.
- Hardware block uses 1 register
- When frame buffer allocated, physical address written to register

VGA Console

- Wraps to new line when at end of line, screen clears when console fills
 - Allows backspace and enter keys to be used properly
 - Typing in console is fixed to console window -> movable, resizable
 - Cursor is drawn to identify the next character position
-
- ASCII subimages drawn from ASCII table raw file
 - Character row and column
 - Table of alpha values
 - Allows commands to be executed at any time
 - `\c`, `\q`, `\r`
 - Flag set when `'\'` is pressed
 - If no escape character identified, `'\'` is printed before next char

UART Performance Monitor

- Characters typed in Kermit are read and echoed
- Each time a character is typed, update relevant metrics
- Gets the start time of initialization using `CLOCK_MONOTONIC` (software clock)
- Displayed metrics measured in time are using this software clock
- Timer driver set to count down from a value, rollover, reset, and count down again
- Timer rolls over every 50 ms which sends an interrupt
- Every time an I/O interrupt occurs perform computations and display metrics

Key Design Decisions

- VGA Console and UART Performance Monitor merged together
- ASCII characters are drawn from subimages of memory mapped ASCII table
- Commands can be executed for clearing screen, resetting metrics, and quitting
- Frame Synchronization implemented for performance monitor
 - Large quantities were being drawn at the same time, introduced freezing
 - Screen only updates when all changes to draw are completed
 - Interrupts received during draw are simply ignored
 - Allows analysis of frame rate
- Bar graphs are drawn to quickly visualize various statistics we tracked
 - Frames per second, Words per minute, Keys pressed per second
- Console start and System start times shown in format HH:MM:SS

References

A lot of knowledge was acquired from reading Linux man pages, ENSC 351 slides, LogiCORE IP AXI Timer (v1.02a) Data Sheet, and StackOverflow.

Special thanks to ASCII Cat.

Questions?

Answers!