

Project Schedule (Deliverable Due Dates)

May 26 – TFTP Proof of Concept

June 18 – TFTP Client

June 30 – TFTP Daemon

June 30-July 7 – Flash Memory Code Review Period

July 7 – Flash Memory Interface Proof of Concept

August 11 – Flash Memory Interface Completion

August 18 – Summer of Code Summary

TFTP Proof of Concept:

First verify that UDP datagrams are sent and received as expected from a higher level of abstraction. This will involve a brief amount of tests with various size packets.

Second, show that given the datagram devices work as intended it is possible to implement a TFTP service. Based on prior research involving implementing network oriented services as well as some future research showing that specifically TFTP can be implemented a document will be turned in with this research.

Lastly, layout a more specific timeline which will be geared towards the implementation of the TFTP services. This will correspond with the general timeline that is within this application and will note any potential changes to the general timeline. The timeline will breakdown between receiving and sending as well as client versus daemon implementations.

TFTP Client & TFTP Daemon

TFTP client and daemon are two separate deliverables, each of which is heavily based on checking finalized code into the repository that is properly documented. The code check-in will be accompanied by a short write up that covers any problems, hindrances, or changes made from the original plan for the TFTP client or daemon implementation. Also as part of the write up will be the summary of various tests that have been implemented and run to show that the client or the daemon works as intended and needed. Tests may include verification of sending and receiving files as well as verification that the files maintain their integrity.

Flash Memory Interface Proof of Concept & Code Review

First, this document will contain findings from past and present research which allow the assumption that a flash memory interface implementation is possible.

Secondly, this document will be an outline of a code review for the current rudimentary implementation of the flash interface. The code review will determine which parts of the implementation work as intended and conform to predetermined syntax standards. In addition the code review will easily show which parts of the implementation are more rudimentary than others and need to be fleshed out. Lastly, the code review will show which parts of the wanted flash memory interface are not yet implemented.

Lastly, this document will contain a projected timeline for implementation of the various parts of the flash memory interface. This timeline will note any changes to the general timeline and will be broken down by the parts of the flash memory interface that need fixing, addition, or new implementation.

Flash Memory Interface Completion

The flash memory interface completion deliverable is a code turn-in deliverable that will be accompanied by a short write-up that notates any changes or problems encountered with the implementation of the flash memory interface. In addition to these notations will be a written summary of tests to prove that the various parts of the interface work as intended.

Summer of Code Summary

This document will contain an overview of what took place the whole summer, summary of tests for proof that implementations work as intended or show which fail. The final test of the summer will be to use the TFTP implementation to acquire a file and then write it into flash memory in addition to reading a file from flash memory and sending it out via the TFTP implementation. Also this document will contain suggestions as to future research areas and features to implement that take advantage of the systems put into place by the project.

Working Deliverables

SVN

A subversion branch will be created at the beginning of the summer. This branch will contain all changes to the code base that I make throughout the summer. Currently the repository is only accessible to those working on XINU, but upon request it can be changed to allow others to view a specific branch of interest or the main trunk.

Wiki

A wiki page will be added to the Embedded XINU wiki site. The page will be continually updated over the summer to contain finished portions of the project, portions to be implemented, portions in progress, problems that have been encountered, as well as many other things.