Implementing Immediate files in MINIX 3.1.8

To implement immediate files we:

* Create a flag that marks the file as immediate (specifically so we can tell with a ls –l)
* allow for files to be opened as immediate
* restrict the size of an immediate file to 40 bytes
* Alter the reading and writing process to accommodate immediate files.
* Integrate Immediate files with other commands (ls, stat,cat)

*/usr/src/commands/ls/ls.c*

Modify the character array that is used for the rwx field to include an ‘i’ where we want it.

*/usr/src/commands/stat/stat.c*

Add an S\_ISIMM check to the byte[0] to set the character to ‘i’ if the file is immediate

*/usr/src/include/minix/const.h*

*/usr/include/minix/const.h*

Define I\_IMMEDIATE for support of immediate files, we choose 50000.

*/usr/src/include/minix/fcntl.h*

*/usr/include/fcntl.h*

Define O\_IMM for support of immediate files, we choose 5000.

*/usr/src/include/stat.h*

*/usr/include/stat.h*

Define a macro for S\_ISIMM similar to the S\_ISREG to be included in the same areas as S\_ISREG for support of other system calls.

*/usr/src/servers/mfs/read.c*

Add writing of immediate files.

Ensures we cannot read more bytes than are available.

Makes sure position will not go over 40.

Make a special case for immediate:

If reading, send file data from inode to vfs using sys\_safecopyto

If writing, store data from vfs into the inode using sys\_safecopyfrom

Add nrbytes to position and cum\_io and set nrbytes to 0

*/usr/src/servers/vfs/open.c*

If O\_IMM is in oflags then set mode to I\_IMMEDIATE

Known bugs:

* O\_APPEND does not write to the end of the previous written content.
* ‘cp’ does not make the new file immediate.

Testing cases:

* Single write
* Single read
* try to read/write past the file size
* nano read/edit
* ls -l
* less
* more
* stat
* cat
* multiple writes
* multiple reads