

Project 2

F2FS-optimized FTL

Joohyung Park(joohyungpark@csl.skku.edu)

[Computer Systems Laboratory](#)

Sungkyunkwan University

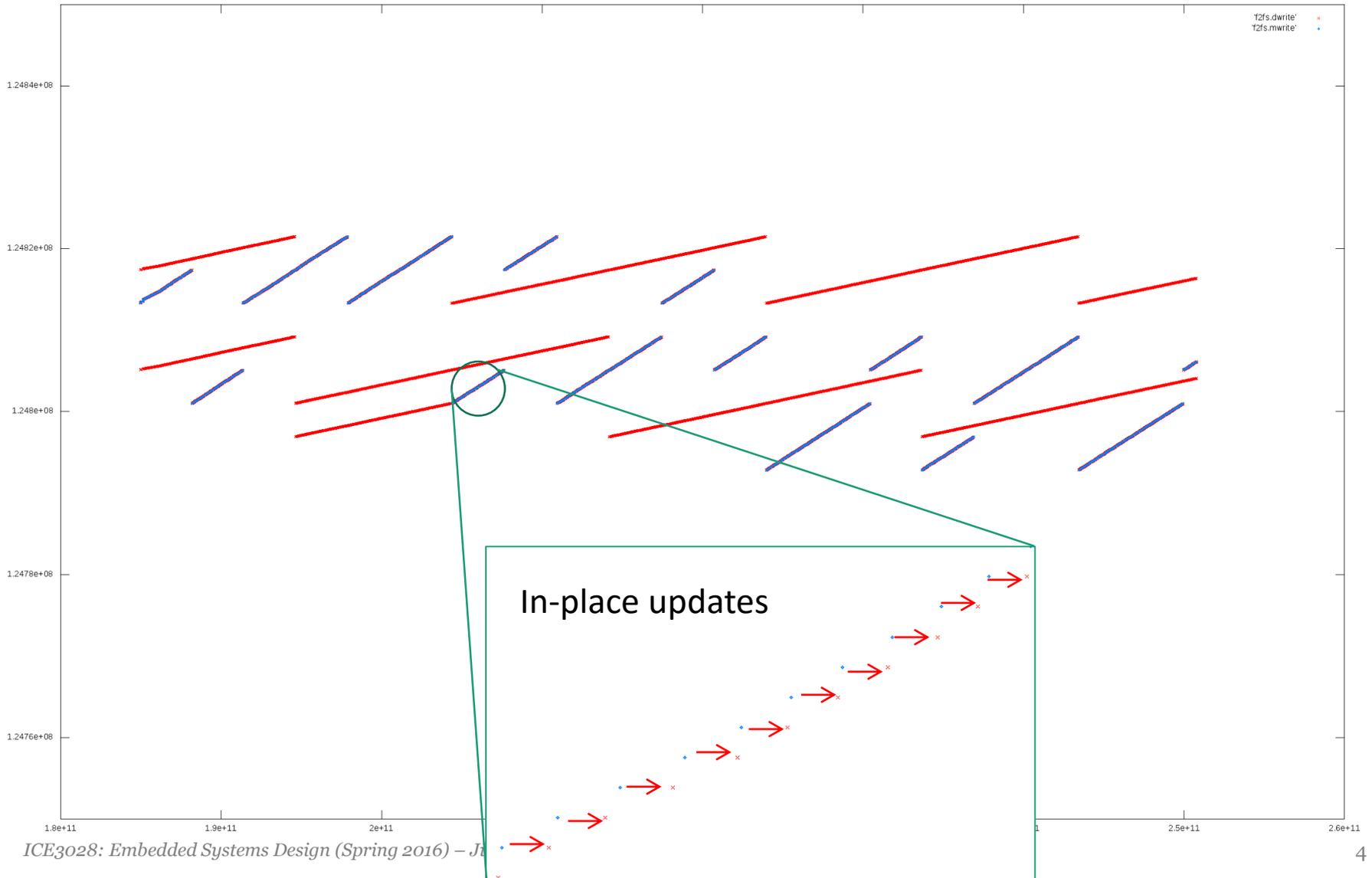
Project Goal

- Do not implement extra features of FTL
 - Ex) Trim, NCQ, DRAM buffering
- Focus on
 - Page allocation policy
 - Mapping unit and policy
 - GC management policy
 - Valid copy control
- Fit your own FTL into given I/O pattern for
 - Less memory footprint, better performance
 - I/O pattern: SQLite 35K tuples inserted onto F2FS
 - An fsync per an insertion

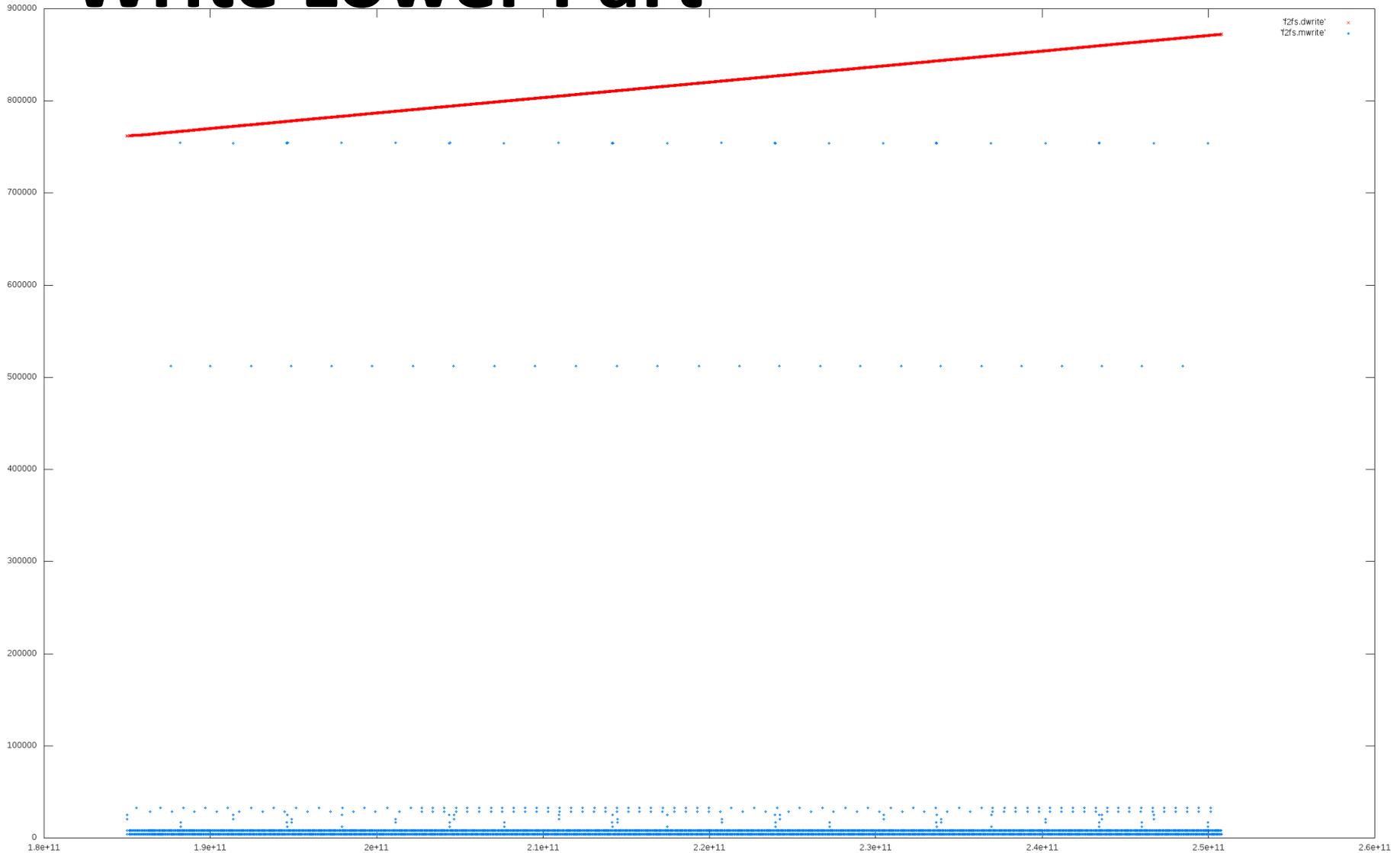
Given I/O Pattern



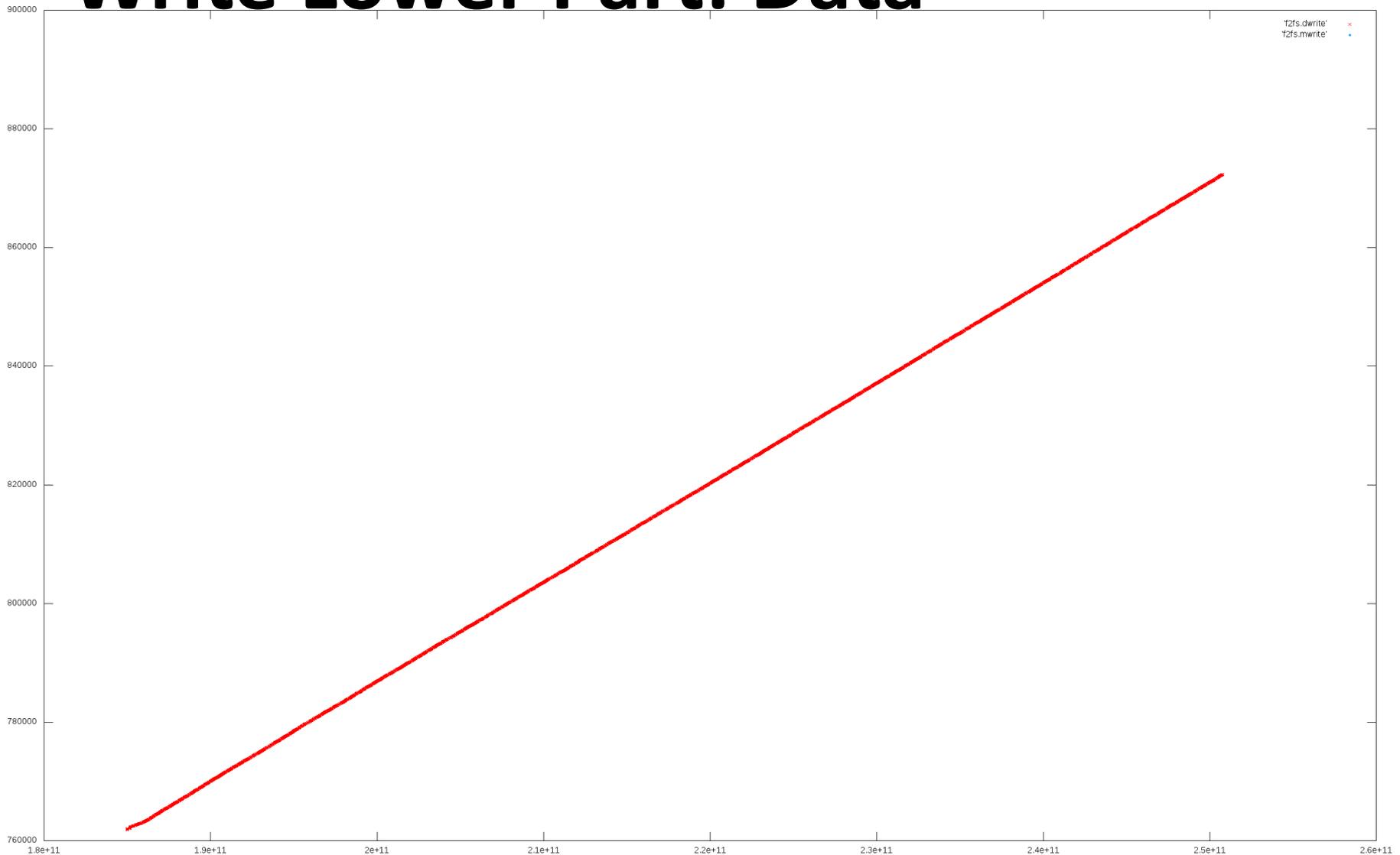
Write Upper Part



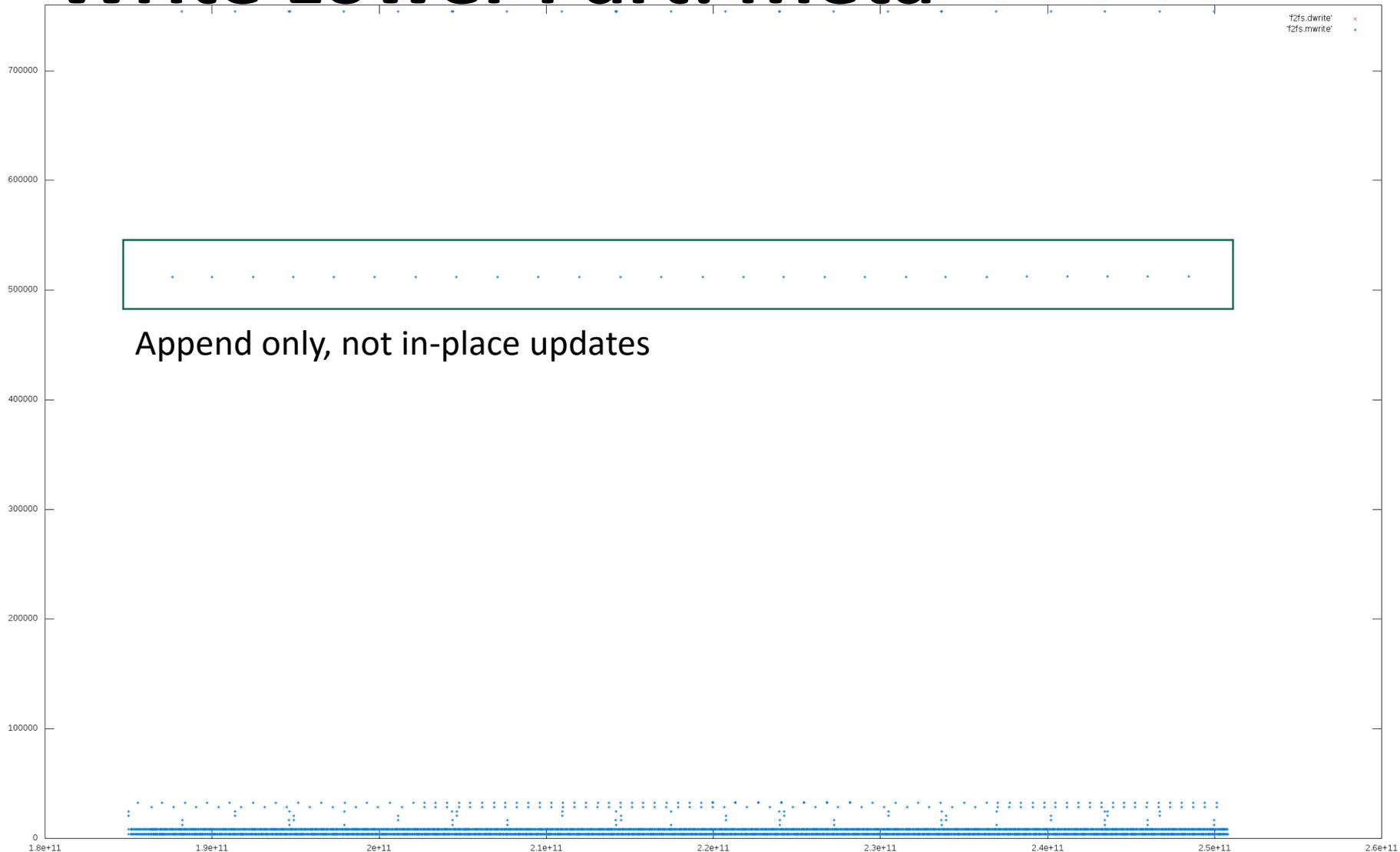
Write Lower Part



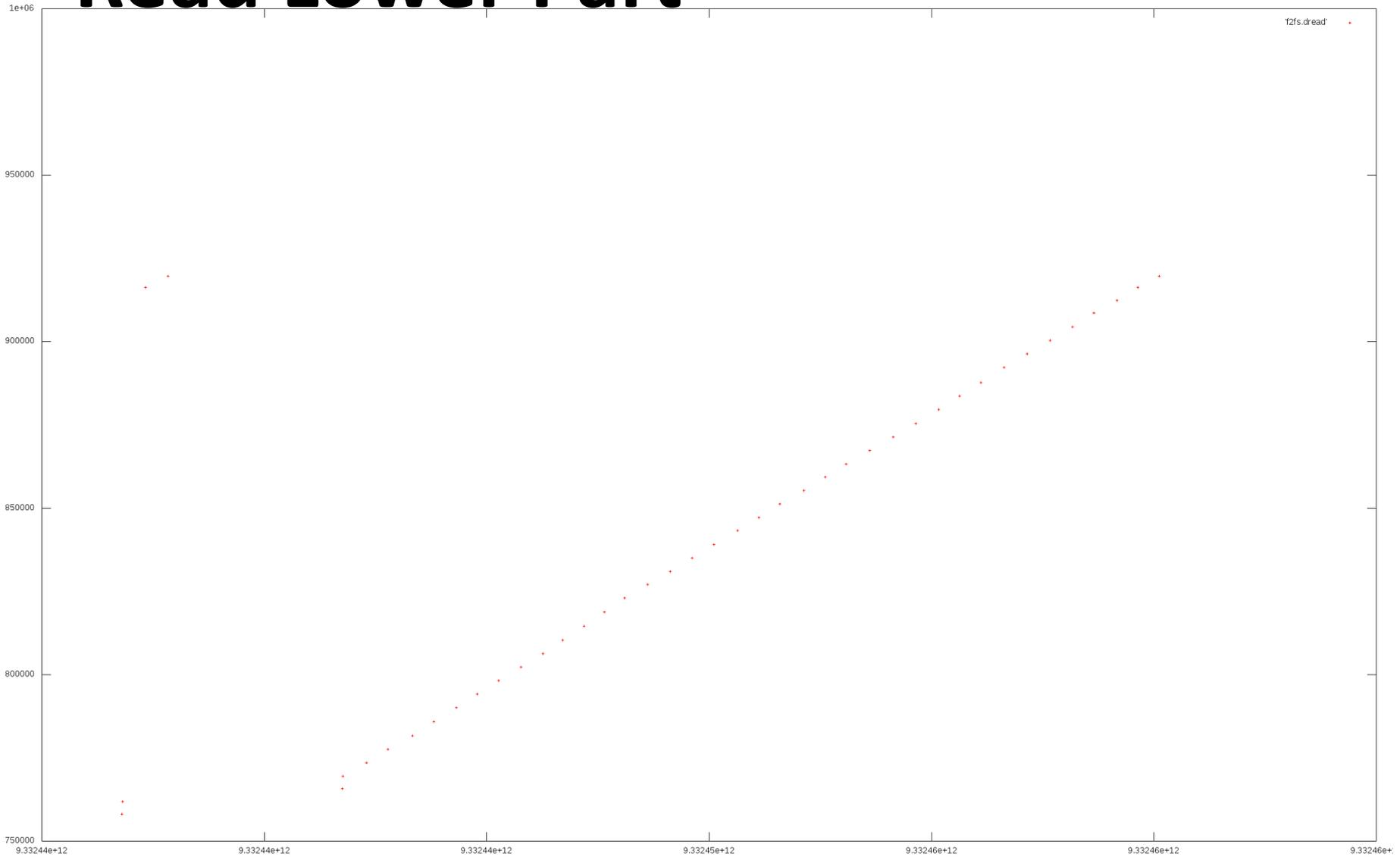
Write Lower Part: Data



Write Lower Part: Meta



Read Lower Part



Testing Process

- Make sure the Jasmine to be included in /dev/sd*
 - lsblk
- Make a partition to test
 - fdisk /dev/sd*
 - 'n' for a new partition creation
 - 'w' to commit the modification
- Run the given script
 - ./F2FS.sh \$DEVICE_MINOR \$PARTITION_PATH

Results

- *.sqlite3.db
 - All I/O events on Jasmine
- *.*read / *.*write
 - Prefix d = data, m = meta
 - nanosec | starting | length
- *.info
 - Total events num | total amount of I/O in sectors
- *.png
 - Visualized results
- *.*runtime
 - Elapsed time info

```
> csl@JHlnx: ~/FI0test
csl@JHlnx:~/FI0test$ ls
block_bio_remap.dat  f2fs.dread          f2f
block_rq_issue      f2fs.dread.info    f2f
f2fs_bottom_data.png f2fs.dwrite         f2f
f2fs_bottom_meta.png f2fs.dwrite.info    F2F
f2fs_bottom.png     f2fs.mread         f2f
f2fs.dat            f2fs.mread.info    f2f
csl@JHlnx:~/FI0test$ cat f2fs.dwrite.info
]27596|276992
csl@JHlnx:~/FI0test$ head f2fs.dwrite
48515351232974|124813312|8
48515351504977|124805128|8
48515351775274|124813320|8
48515351789653|124817416|8
48515352895088|124813328|8
48515353079759|761856|16
48515353337930|124813336|8
48515353618210|761872|8
48515353855517|761872|16
48515354107175|124813344|8
csl@JHlnx:~/FI0test$ █
```

Grading Policies

- *Document Grade* $\times \frac{dram_org}{dram_yours} \times \frac{wrt_org}{wrt_yours} \times \frac{rrt_org}{rrt_yours}$
- Document requirements
 - DRAM layout
 - All of custom defined structures
 - All of global variables
 - Objective/usage of layout, structures, and variables
 - What is the problem?
 - Why should you argue your approach?
 - Expected gain / disadvantage of your approach
 - Specification to be uploaded