SSD Firmware Implementation Project - Lab. #8-

Sang-Phil Lim (lsfeel0204@gmail.com)
SKKU VLDB Lab.

2011-06-02
<table>
<thead>
<tr>
<th>Lab.</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>#1</td>
<td>FTL Simulator Development Guide</td>
</tr>
<tr>
<td>#2</td>
<td>FTL Simulation Guide</td>
</tr>
<tr>
<td>#3</td>
<td>Project 1 Presentation</td>
</tr>
<tr>
<td>#4</td>
<td>Jasmine OpenSSD platform tutorial #1</td>
</tr>
<tr>
<td>#5</td>
<td>Jasmine OpenSSD platform tutorial #2</td>
</tr>
<tr>
<td>#6</td>
<td>FTL Porting Guide</td>
</tr>
<tr>
<td>#7</td>
<td>Firmware Debugging Guide</td>
</tr>
<tr>
<td>#8</td>
<td><strong>SSD Performance Evaluation Guide</strong></td>
</tr>
<tr>
<td>#9</td>
<td>Project 2 Presentation</td>
</tr>
</tbody>
</table>
FTL Evaluation Guide

1. Before benchmarking, you have to verify the FTL logics first

2. Microscopic analysis with measuring the response time & Code optimization
   – Processing overhead
   – Garbage collection overhead

3. Performance test with commercial/well-known storage benchmark tools
void ftl_test(void)
{
    ...io_cnt = 500;
    num_sectors = 4;
    ...

    // STEP 1 – write
    for (i = 0; i < io_cnt; i++){
        wr_buf_addr = WR_BUF_PTR(g_ftl_write_buf_id) +
            ((lba % SECTORS_PER_PAGE) * BYTES_PER_SECTOR);
        for (j = 0; j < num_sectors; j++){
            mem_set_dram(wr_buf_addr, data, BYTES_PER_SECTOR);
            wr_buf_addr += BYTES_PER_SECTOR;
            if (wr_buf_addr >= WR_BUF_ADDR + WR_BUF_BYTES) {
                wr_buf_addr = WR_BUF_ADDR;
            }
            data++;
        }
        ftl_write(lba, num_sectors);
        lba += num_sectors;
    }
    ...
}
// STEP 2 - read and verify
for (i = 0; i < io_cnt; i++) {
    rd_buf_addr = RD_BUF_PTR(g_ftl_read_buf_id) +
        ((lba % SECTORS_PER_PAGE) * BYTES_PER_SECTOR);
    ftl_read(lba, num_sectors);
    flash_finish();

    for (j = 0; j < num_sectors; j++) {
        UINT32 sample = read_dram_32(rd_buf_addr);

        if (sample != data) {
            uart_print("ftl test fail...");
            led_blink();
        }
        rd_buf_addr += BYTES_PER_SECTOR;

        if (rd_buf_addr >= RD_BUF_ADDR + RD_BUF_BYTES) {
            rd_buf_addr = RD_BUF_ADDR;
        }
        data++;
    }
    lba += num_sectors;
}
...
FTL Logic Test Guide (contd.)

• POR(Power-Off Recovery) Test
  – Enable `OPTION_FTL_TEST` in `jasmine.h`
  – Toggle Power-On/Off Switch after finishing `ftl_test()`
FTL Logic Optimization

- Performance fluctuation normalization
  - IO Response time & Throughput

![Graphs showing response time and throughput over time.](attachment:graphs.png)
Performance Evaluation: Introduction

• SSD Benchmarking
  – PCMark05
  – Iometer benchmark tool
• c.f.) IO bandwidth limitation
  – In current Jasmine version, only two channels enabled (2CH/4WAY)
    • BANK_BMP(A0,B0, A1,B1, A4,B4, A5,B5)
  – Raw device
    • 256KB Sequential read: < 100MB/s
    • 256KB Sequential write: < 85~90MB/s
Performance Evaluation: Introduction

1. Install new firmware
   - Enable `OPTION_REDUCED_CAPACITY` in `jasmine.h`
   - Reduce SSD Storage size 64GB to 8GB
2. Format Jasmine (erase all blocks)
3. w/ `lometer`
   - To evaluate firmware perf. before/after GC
   - Raw device test
4. w/ `PCMark05`
   - To evaluate overall test under the file system
   - Windows XP startup, Application loading, Virus scan, File write, etc.
Performance Evaluation: Iometer

- Iometer
  1. 256KB Sequential write (30sec)
  2. 256KB Sequential read (30sec)
  3. 8KB Random write (30sec)
  4. 8KB Random read (30sec)
  5. Aging (8KB Random write) (30min~)
  6. 8KB Random write (1min)
  7. 8KB Random read (30sec)
  8. 256KB Sequential write (1min)
  9. 256KB Sequential read (30sec)

- Sequential -> MB/s
- Random -> IOPS

Before GC

After GC
Performance Evaluation: PCMark05

• PCMark05
  – Download a basic edition (v.1.2.0)
    http://www.futuremark.com/download/pcmark05/
Performance Evaluation: PCMark05

• Run PCMark05

Test configuration
Performance Evaluation: PCMark05

• Test configuration (HDD Test Suite)
Performance Evaluation

• Benchmarking & See benchmark score
Evaluation Result

• Summarize your evaluation results
  – w/ Iometer
  – w/ PCMark05

• Plotting results
  – Bar/Graph, etc.
Any Questions?