Embedded System Practice Lab 3

2016311821 한승하

우선 이번주 Exercise를 진행하기 위해 실습 수업에서 주어진 대로 환경 설정을 완료하였습니다.

이 때 제 컴퓨터 기준으로 일반 emulator를 사용한 경우 위의 사진과 같이 7segment에 아무런 반응이 없어 piazza를 참고하여 emulator64-x86을 사용하여 실습을 진행하였습니다.

```
han@han: ~
299     64      recvmmsg                sys_recvmmsg
300     common  fanotify_init           sys_fanotify_init
301     common  fanotify_mark           sys_fanotify_mark
302     common  prlimit64               sys_prlimit64
303     common  name_to_handle_at       sys_name_to_handle_at
304     common  open_by_handle_at       sys_open_by_handle_at
305     common  clock_adjtime           sys_clock_adjtime
306     common  syncfs                  sys_syncfs
307     64      sendmmsg                sys_sendmmsg
308     common  setns                   sys_setns
309     common  getcpu                  sys_getcpu
310     64      process_vm_readv        sys_process_vm_readv
311     64      process_vm_writev       sys_process_vm_writev
312     common  kcmp                    sys_kcmp
313     common  finit_module            sys_finit_module
# 314   common  sched_setattr           sys_sched_setattr
# 315   common  sched_getattr           sys_sched_getattr
# 316   common  renameat2               sys_renameat2
317     common  seccomp                 sys_seccomp
322     common  7seg_control            sys_7seg_control
#
# x32-specific system call numbers start at 512 to avoid cache impact
# for native 64-bit operation.
                                                        327,41-56     90%
```

7segment를 조작하기 위하여 sys_7seg_control을 추가하여 주었습니다. Code를 goldfish_segment.c 파일에 추가할 것이기 때문에 Makefile을 수정해 주지는 않았습니다.

```
asmlinkage long sys_7seg_control(int num)
{
        if(num>9999999)
        {
                GOLDFISH_SEGMENT_WRITE(data, SEGMENT0, 0x77);
                GOLDFISH_SEGMENT_WRITE(data, SEGMENT1, 0x77);
                GOLDFISH_SEGMENT_WRITE(data, SEGMENT2, 0x77);
                GOLDFISH_SEGMENT_WRITE(data, SEGMENT3, 0x77);
                GOLDFISH_SEGMENT_WRITE(data, SEGMENT4, 0x77);
                GOLDFISH_SEGMENT_WRITE(data, SEGMENT5, 0x77);
                GOLDFISH_SEGMENT_WRITE(data, SEGMENT6, 0x77);
                return 0;
        }
        int start_num = num;
        int temp;
        int idx;
        for(idx=6;idx>=0;idx--)
        {
                temp = start_num%10;
                start_num = start_num/10;
                switch(temp){
                        case 0: GOLDFISH_SEGMENT_WRITE(data, idx*0x4, 0x77);
                                break;
                        case 1: GOLDFISH_SEGMENT_WRITE(data, idx*0x4, 0x24);
                                break;
                        case 2: GOLDFISH_SEGMENT_WRITE(data, idx*0x4, 0x5d);
                                break;
                        case 3: GOLDFISH_SEGMENT_WRITE(data, idx*0x4, 0x6d);
                                break;
                        case 4: GOLDFISH_SEGMENT_WRITE(data, idx*0x4, 0x2e);
                                break;
                        case 5: GOLDFISH_SEGMENT_WRITE(data, idx*0x4, 0x6b);
                                break;
                        case 6: GOLDFISH_SEGMENT_WRITE(data, idx*0x4, 0x7a);
                                break;
                        case 7: GOLDFISH_SEGMENT_WRITE(data, idx*0x4, 0x27);
                                break;
                        case 8: GOLDFISH_SEGMENT_WRITE(data, idx*0x4, 0x7f);
                                break;
                        case 9: GOLDFISH_SEGMENT_WRITE(data, idx*0x4, 0x2f);
                                break;
                }
        }
        return 0;
}
```

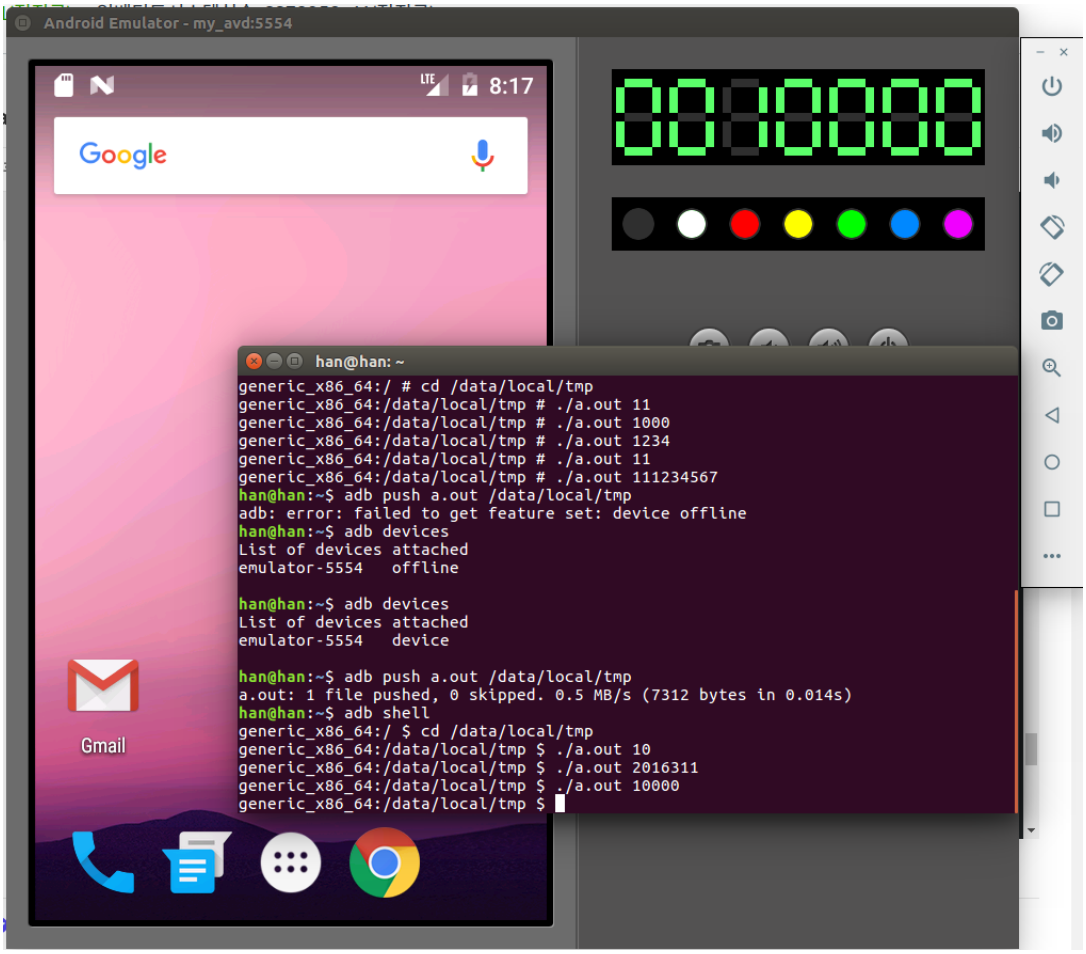
위 코드는 제가 사용한 코드입니다. 9999999이상의 입력이 들어온 경우 모든 Segment가 0이 출력되도록 처리해 주었으며, 이외의 경우 각 자리 수 마다 Segment가 제대로 표시될 수 있게 해 주었습니다.

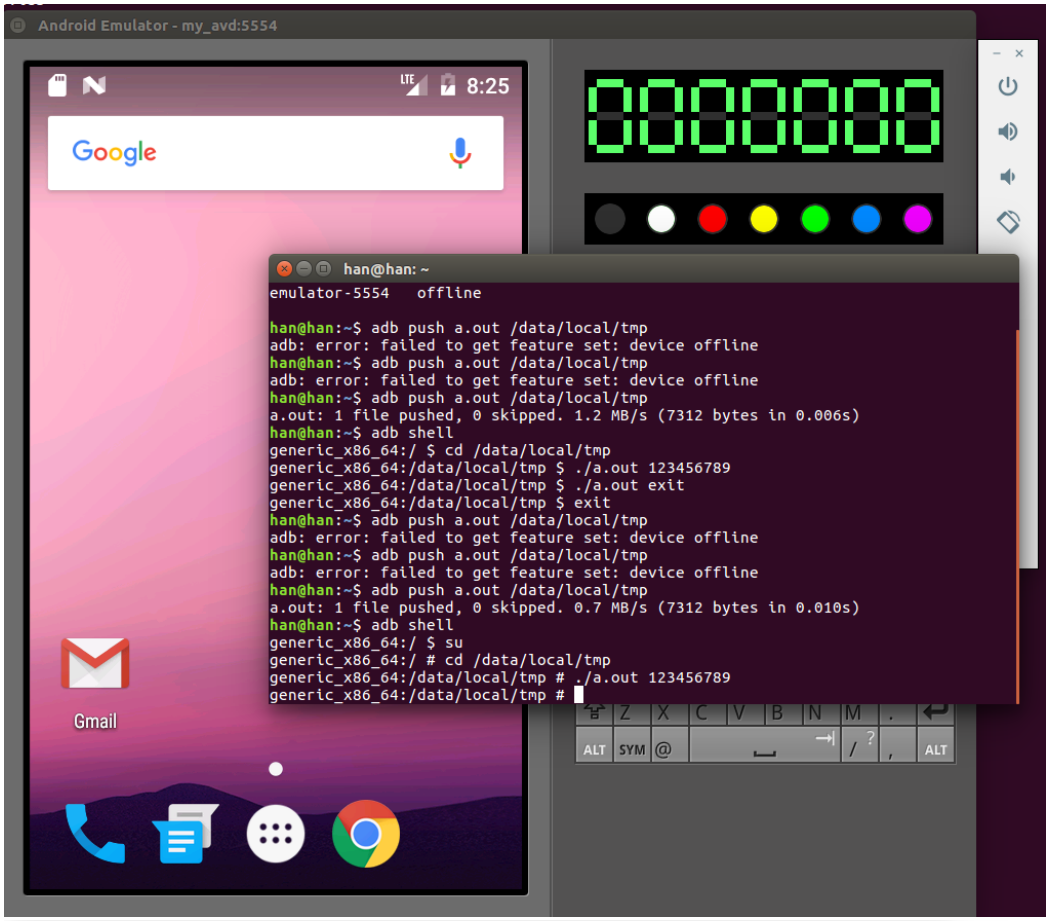
```
struct goldfish_segment_data *data;
```

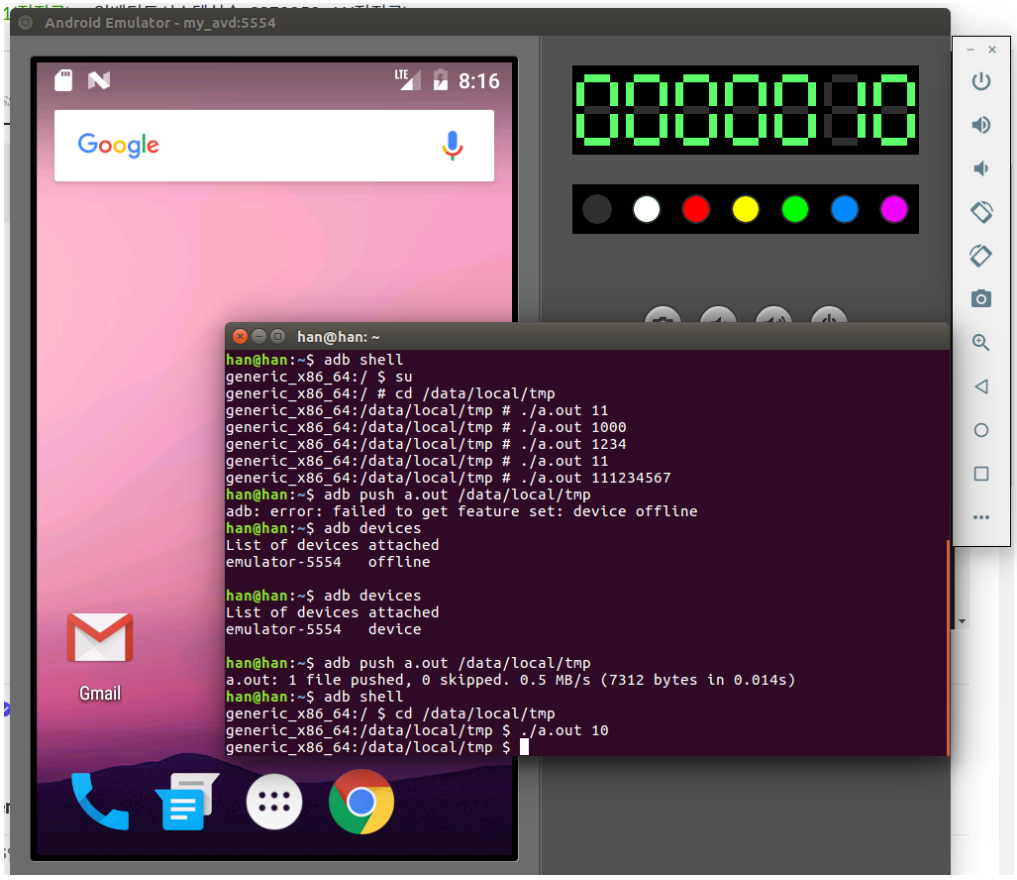
이때 data변수를 사용하기 위해 해당 변수를 전역변수로 바꾸어 주었습니다.

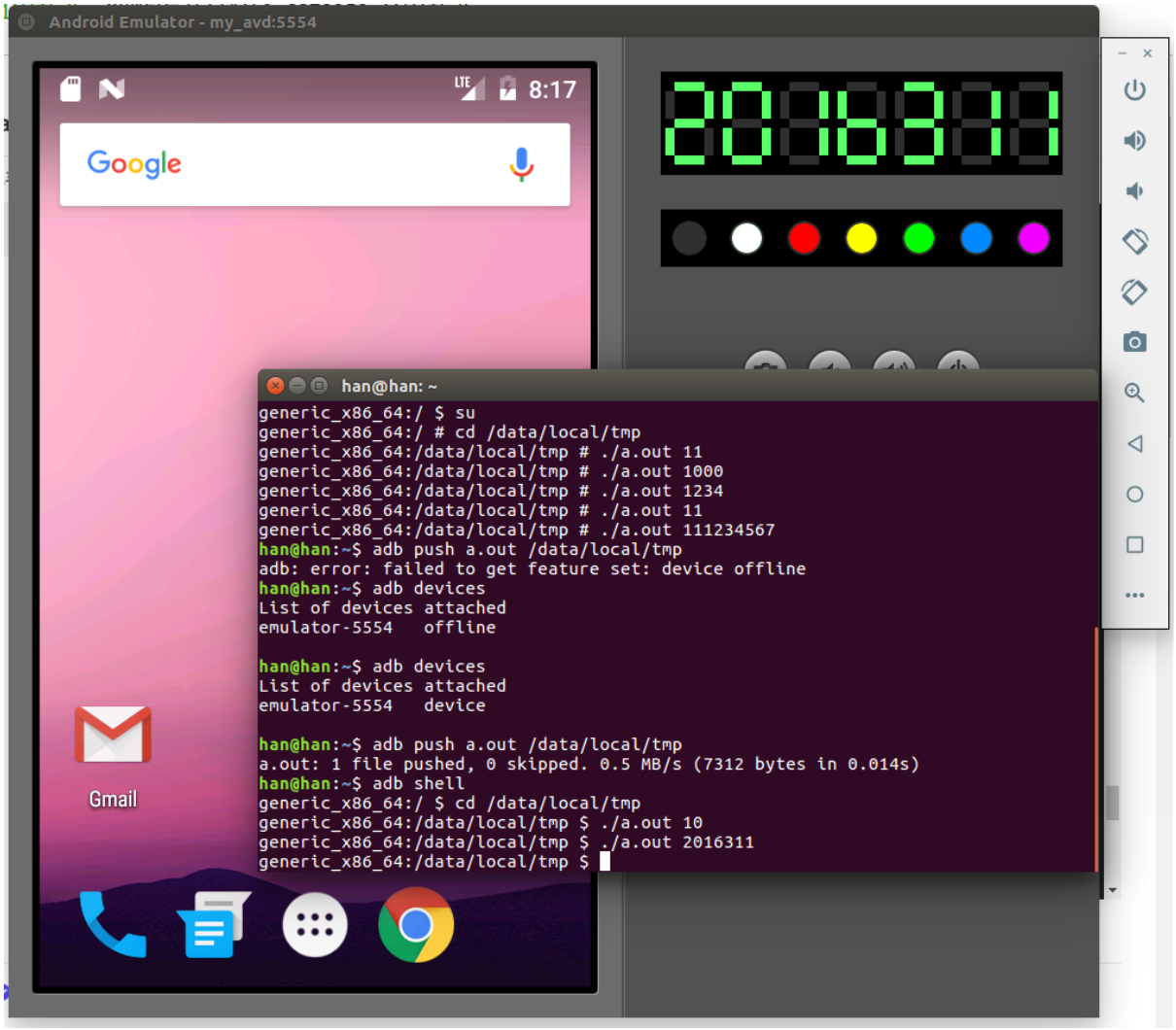
```
han@han: ~
#include <unistd.h>
#define __NR_7seg_control 322

int main(int argc, char **argv)
{
        int num = atoi(argv[1]);
        syscall(__NR_7seg_control,num);
        return 0;
}
~
~
~
~
~
~
~
~
~
~
~
~
~
~
"userspace.c" 9L, 157C                                1,1           All
```

User Code는 위와 같이 처리해 주었으며, 아래는 해당 실습의 결과창입니다.




Android Emulator - my_avd:5554
8:25
Google
Gmail
han@han: ~
emulator-5554 offline
han@han:~$ adb push a.out /data/local/tmp
adb: error: failed to get feature set: device offline
han@han:~$ adb push a.out /data/local/tmp
adb: error: failed to get feature set: device offline
han@han:~$ adb push a.out /data/local/tmp
a.out: 1 file pushed, 0 skipped. 1.2 MB/s (7312 bytes in 0.006s)
han@han:~$ adb shell
generic_x86_64:/ $ cd /data/local/tmp
generic_x86_64:/data/local/tmp $ ./a.out 123456789
generic_x86_64:/data/local/tmp $ ./a.out exit
generic_x86_64:/data/local/tmp $ exit
han@han:~$ adb push a.out /data/local/tmp
adb: error: failed to get feature set: device offline
han@han:~$ adb push a.out /data/local/tmp
adb: error: failed to get feature set: device offline
han@han:~$ adb push a.out /data/local/tmp
a.out: 1 file pushed, 0 skipped. 0.7 MB/s (7312 bytes in 0.010s)
han@han:~$ adb shell
generic_x86_64:/ $ su
generic_x86_64:/ # cd /data/local/tmp
generic_x86_64:/data/local/tmp # ./a.out 123456789
generic_x86_64:/data/local/tmp #


Android Emulator - my_avd:5554
8:16
Google
Gmail
han@han: ~
han@han:~$ adb shell
generic_x86_64:/ $ su
generic_x86_64:/ # cd /data/local/tmp
generic_x86_64:/data/local/tmp # ./a.out 11
generic_x86_64:/data/local/tmp # ./a.out 1000
generic_x86_64:/data/local/tmp # ./a.out 1234
generic_x86_64:/data/local/tmp # ./a.out 11
generic_x86_64:/data/local/tmp # ./a.out 111234567
han@han:~$ adb push a.out /data/local/tmp
adb: error: failed to get feature set: device offline
han@han:~$ adb devices
List of devices attached
emulator-5554 offline
han@han:~$ adb devices
List of devices attached
emulator-5554 device
han@han:~$ adb push a.out /data/local/tmp
a.out: 1 file pushed, 0 skipped. 0.5 MB/s (7312 bytes in 0.014s)
han@han:~$ adb shell
generic_x86_64:/ $ cd /data/local/tmp
generic_x86_64:/data/local/tmp $ ./a.out 10
generic_x86_64:/data/local/tmp $


Android Emulator - my_avd:5554
8:17
Google
Gmail
han@han: ~
generic_x86_64:/ $ su
generic_x86_64:/ # cd /data/local/tmp
generic_x86_64:/data/local/tmp # ./a.out 11
generic_x86_64:/data/local/tmp # ./a.out 1000
generic_x86_64:/data/local/tmp # ./a.out 1234
generic_x86_64:/data/local/tmp # ./a.out 11
generic_x86_64:/data/local/tmp # ./a.out 111234567
han@han:~$ adb push a.out /data/local/tmp
adb: error: failed to get feature set: device offline
han@han:~$ adb devices
List of devices attached
emulator-5554 offline
han@han:~$ adb devices
List of devices attached
emulator-5554 device
han@han:~$ adb push a.out /data/local/tmp
a.out: 1 file pushed, 0 skipped. 0.5 MB/s (7312 bytes in 0.014s)
han@han:~$ adb shell
generic_x86_64:/ $ cd /data/local/tmp
generic_x86_64:/data/local/tmp $ ./a.out 10
generic_x86_64:/data/local/tmp $ ./a.out 2016311
generic_x86_64:/data/local/tmp $