**Embedded System Software Project**

**(프로젝트 수행 결과 보고서)**

**과목명: [CSE4116] 임베디드시스템소프트웨어**

**담당교수: 서강대학교 컴퓨터공학과 박 성 용**

**학번 및 이름: 20161577, 김인호**

**개발 기간: 2020. 06. 26. – 2020. 06. 28**

**최 종 보 고 서**

1. **개발 목표**

Android 프로그래밍과 JNI 개발을 포함한 한 학기 동안 배운 강의 내용을 적용하여 전체적으로 다시 학습을 하고 Android 앱부터 Linux 커널까지의 여러 layer에서의 개발 및 구현. 클래식 게임 중 하나인 Hangman Game을 주제로 구현.

1. **개발 범위 및 내용**
2. **개발 범위**

* 디바이스 드라이버: hangman game 기능이 구현된 디바이스 드라이버를 하나의 모듈 형태로 제공
* 타이머: hangman game의 시간과 관련된 기능 구현을 위한 kernel timer 관리
* FPGA: 사용된 FPGA 디바이스에 대한 출력을 간소화하기 위한 interface 제공
* Interrupt: hangman game 조작에 사용되는 버튼들에 대한 interrupt handling
* Android 응용프로그램: Android NDK와 JNI를 통해 hangman game 커널 모듈과 interact하는 응용프로그램
* 로그: 디바이스 드라이버가 수행 중 유용한 정보나 FPGA 디바이스에 출력되는 데이터를 minicom으로 확인할 수 있도록 하는 logging 제공

1. **개발 내용**
   * 디바이스 드라이버

디바이스 드라이버는 hangman game의 기능을 모듈 형태로 담고 있으며 Linux 커널에 설치될 수 있다. 디바이스 파일에 수행되는 operation인 open(), close(), ioctl() 그리고 read() 각각에 대응되는 함수 포인터와 각 함수들이 구현되어 있다. 또한 hangman game의 핵심 기능들도 구현되어 있고 초기화와 state 관리를 담당한다.

* + 타이머

Hangman game에는 여러 시간과 관련된 기능이 있는데 이 모두 kernel timer를 이용해서 구형된다. 스위치 입력에 대한 timer, 점수 부여에 대한 timer 그리고 종료 기능에 대한 timer가 있다.

* + FPGA

Hangman game 구현에 쓰이는 FPGA 디바이스를 사용하기 위한 IO-mapping 작업과 출력 interface를 구현한다. Interface를 이용하여 디바이스에 데이터를 넘겨야 하는 방식에 대한 로직을 간소화할 수 있다.

* + Interrupt

Hangman game 조작에 사용되는 여러 버튼이 있다. 각 버튼으로부터 발생하는 interrupt처리를 구현하고 interrupt context에서의 실행을 처리한다. Home, Vol+그리고 Vol-에 대한 interrupt가 처리된다.

* + Android 응용프로그램

Android 프로그램 부분이 이번 프로젝트와 기존 과제가 가장 차이 많이 나는 부분이다. 응용 레벨에서의 프로그램을 만들어 개발한 커널 모듈과 소통하고 JNI를 활용하여 자바 코드 수행 중 C로 작성된 native 코드를 사용한다.

* + 로그

로그는 개발 과정에 도움을 주고 에러 해결과 디버깅을 하는 데에 유용하게 쓰인 기능이다. 입력과 출력에 대한 정보를 실시간으로 볼 수 있고 hangman game이 진행되면서 점수, 맞춰야 하는 단어 등 여러 state에 대한 정보를 확인할 수 있다.

1. **추진 일정 및 개발 방법**
2. **추진 일정**
   * 6월 26일 (금): 아이디어 구상, 프로젝트 구조와 주제 정의
   * 6월 27일 (토): Hangman game 커널 모듈 구현
   * 6월 28일 (일): Android 애플리케이션과 JNI 프로그래밍
3. **개발 방법**
   * 디바이스 드라이버

먼저 디바이스 드라이버와 관련된 함수들이다. 아래는 해당 디바이스 드라이버가 구현된 모듈을 Linux에서 `insmod`명령어와 `rmmod`명령어로 설치 및 제거를 할 때 실행되는 함수이다. 올바른 수행과 종료를 위한 초기화 작업을 담당한다.

A screenshot of a cell phone

Description automatically generated A screen shot of a computer

Description automatically generated

그 다음으로는 `open()`과 `close()`해 대응되는 함수들이다. `open()`을 하면 hangman game의 상태를 초기화 시키고 interrupt를 등록시킨 후 hangman game 기능을 시작한다. 그리고 `close()`를 하면 등록했던 interrupt를 해제하고 FPGA를 초기화하고 타이머를 삭제한다.

A screenshot of a cell phone

Description automatically generated

A screenshot of a cell phone

Description automatically generated

또한 `read()`와 `ioctl()`에 대한 함수도 구현되어 있다. 이 두 함수는 FPGA의 switch 입력을 받기 위해 사용하는데 Android 어플리케이션에서 먼저 `ioctl()`을 하면 커널 모듈이 해당 thread를 sleep 시키고 커널 타이머를 활용하여 유효한 switch 입력이 있는지 주기적으로 확인한다. 유효한 switch 입력이 들어오면 이를 저장했다가 sleep 시켰던 Android thread를 다시 wake 하고 Android thread는 `read()`해서 저장했던 switch 입력을 읽어 들인다. `read()`를 처리하면서 해당 switch 입력으로 선택한 글자가 현재 hangman 게임의 단어의 일부인지, 아닌지 또는 종료 조건을 만족했는지 확인한다.

A screenshot of a cell phone

Description automatically generated A screenshot of a cell phone

Description automatically generated

A screenshot of a cell phone

Description automatically generated

그 다음으로는 hangman game의 핵심 함수들이다. Switch로 선택한 글자를 저장하는 함수가 있고 게임을 다음 단어로 옮기는 함수가 구현되어 있다.

A screenshot of a cell phone

Description automatically generated A screenshot of a cell phone

Description automatically generated

또한 아래와 같이 현재 맞추고 있는 단어를 skip하는 함수가 있고 보너스 점수와 관련된 함수와 게임을 종료시키는 함수가 있다. 그리고 마지막으로 선택한 글자가 알맞은 선택지였는지 확인하는 함수가 있다.

A screenshot of a cell phone

Description automatically generatedA screenshot of text

Description automatically generated

* + 타이머

Hangman game에서는 총 3가지의 커널 타이머를 관리한다. 그 중 하나는 switch 입력을 주기적으로 확인하기 위한 타이머인데 아래와 같이 구현되어 있다. 1/7 초마다 expire 되는 타이머를 이용한다.

A screenshot of a cell phone

Description automatically generated A screenshot of a cell phone

Description automatically generated

그 다음은 score timer이다. 단어에 속한 글자의 개수에 따라 보너스 점수가 있는데 이를 첫 몇 초 내에 맞춰야만 받을 수 있다. 따라서 단어가 바뀔 때 마다 등록되는 타이머이며 각 글자마다 5초의 시간이 주어진다.

A screenshot of a cell phone

Description automatically generated

마지막 타이머는 exit timer이다. 이는 과제3에서 구현한 종료 기능과 유사하며 Vol- 버튼을 3초 이상 누르고 있을 때에 종료되도록 구현하였다.

A screenshot of a cell phone

Description automatically generated

* + Android 응용프로그램

먼저 JNI 부분은 다음과 같다. 총 3개의 method가 native code로 구현되었고 아래와 같다. 디바이스 파일을 open하고 close하는 함수가 있고 ioctl()과 read()까지 핵심 부분을 담당하는 함수도 있다.

A picture containing bird

Description automatically generatedA screenshot of a cell phone

Description automatically generated

A screenshot of a cell phone

Description automatically generated

저렇게 native code에서 Payload이라는 object에 모듈로부터 받은 데이터를 넣고 아래와 같은 방식으로 main thread에 알린다.

A screenshot of a cell phone

Description automatically generated

1. **연구 결과**

이번 프로젝트를 통화 성공적으로 어플리케이션으로부터 커널 영역까지의 개발을 할 수 있었다. 사이사이의 데이터 흐름이 어떻게 흘러가는지 더 정확히 파악할 수 있었다.



1. **기타**

개발 환경에 어려움이 있었지만 더 나은 환경을 찾아 마련하는 과정도 의미 있는 과정이었다.