|  |  |  |
| --- | --- | --- |
| **개인 신상** | |  |
| 성 명 | : 권 민 희 |
| 생년월일 | : 1991.07.13 |
| 연 락 처 | : 010-2898-5880 |
| E-mail | : entwurf.k@gmail.com |

|  |  |
| --- | --- |
| **최종 학력사항** | |
| 상명대학교 | 문화예술경영 |

|  |  |  |
| --- | --- | --- |
| **자격 사항** | | |
| 자격증명 | 취득일 | 발행처 |
| 정보처리기사 | 2023.11.15 | 한국산업인력공단 |
|  |  |  |
|  |  |  |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| **교육이수사항** | | |
| 교육명 | 주요 내용 | 진행처 |
| 임베디드시스템 하이테크 10개월 과정 | * 임베디드 펌웨어 프로그래밍(AVR ATmega128, ARM Cortex-M4) * 영상처리, 머신비전 * HTML, CSS, JavaScript 웹 프로그래밍 * Java 및 Spring Boot * 데이터베이스 및 SQL | 한국폴리텍대학  융합기술교육원  임베디드시스템과 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **전문 설계 능력** | | | | |
| 사용 가능 툴 이름 | 버전 | 제조사 | 주요 사용 능력 | 숙련도 |
| Visual Studio |  | MS | C언어 프로그래밍 및 디버깅 | 상 |
| STM32CubeMX |  | STM | STM32F429(Cortex-M4)펌웨어 코드 생성 | 상 |
| SW4STM32(Eclipse기반) |  | AC6 | STM32F429 펌웨어 프로그래밍 및 디버깅 | 상 |
| Spring Tool Siute4 |  |  |  |  |
| MySQL |  |  |  |  |
| Visual Studio Code |  |  |  | 중 |
| Git |  |  |  |  |
| Ubuntu Linux |  |  |  |  |
| Source Insight |  | Source dynamics | 소스코드 분석 | 중 |
| Arduino IDE |  |  |  |  |
| KiCad |  |  |  |  |
| Source Tree |  |  |  |  |
| IAR Workbench |  |  |  |  |
| ST-Link |  |  |  |  |
| PuTTY |  |  |  |  |
| Notepad++ |  |  |  |  |

|  |  |  |
| --- | --- | --- |
| **직무관련 수상내역** | | |
| 수상명 | 수상내역 | 주관처 / 시행일 |
| 동상 | 2023 창의융합 인재양성을 위한  교육장비 개발 및 아이디어 경진대회 | (사)한국실천공학교육학회 /  2023.11.03 |
|  |  |  |
|  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **주요 교육 이수 내용** | | |  | |
| 과목명 | 시간 | 교육 장비 및 개발 환경 | | 내용 |
| 임베디드 하드웨어 기초 | 60 | 오실로스코프, 브레드보드, IC칩 | | 전자회로 기초, RLC회로, 트랜지스터 특성 이해 |
| 임베디드 하드웨어 설계 | 100 | OrCAD Capture, 브레드보드, 74시리즈IC | | 디지털 공학, 전자회로 설계 실습, AVR 보드 설계, 74시리즈를 이용한 디지털 시계 제작등 |
| 임베디드 특성화 C언어 | 100 | MS Visual Studio 2015 | | C언어 기초, 제어문, 배열, 포인터, 구조체, 파일 제어등 |
| AVR 프로세서 기초 및 설계 | 100 | ATmega128 보드, AVR Studio 4.0 | | AVR 펌웨어 프로그래밍, GPIO, Interrupt, Timer, ADC, UART, EEPROM, I2C 제어 |
| ARM Cortex-M 펌웨어 프로그래밍 | 120 | STM32F429ZI 보드, STMCubeMX,OpenSTM32,Keil MDK-ARM v5 | | ARM Cortex-M 구조, CubeMX를 통한 펌웨어 생성, GPIO, UART, Timer, Interrupt, ADC를 통한 센서 제어 |
| 머신비전 | 120 | MS Visual Studio 6.0 & 2015  OpenCV v3.3 | | 디지털 영상 기초, 픽셀단위 영상처리, 히스토그램, 필터 처리, 기하학적 변환, 모폴로지, opencv활용 영상처리 |
| PCB회로설계 | 50 | Cadence Allegro v16.04 | | PCB 보드 설계 |
| IoT 플랫폼 실무 | 100 | Intel Edison보드, Arduino보드, Ubuntu Linux, Arduino IDE v1.8 | | Intel Edision을 통한 클라우드 연결, 각종 센서 제어, Ubuntu Linux 설치 및 C언어 프로그래밍, 스케치 프로그래밍 |
| 스마트 팩토리 기초 | 64 | Siemens TIA Portal v13, S7-1200, S7-1500, MPS장비 | | 시퀀스 이론 및 실습, 지멘스 PLC제어 및 프로그래밍, 공압, 센서 제어,HMI프로그래밍 |
| 임베디드 하드웨어 실무 | 120 | Xilinx ISE, Vivado , Model SIM | | 디지털 회로 설계 , Verilog 언어를 활용한 FPGA 설계 |

|  |  |  |  |
| --- | --- | --- | --- |
| **보유기술** | | | |
| 범위 | 분류 | 개발 환경 | 기술 내용 |
| 소프트웨어 | C언어 프로그래밍 | MS Visual Studio 2015 | * 기본 C언어 기반 콘솔 응용 프로그램 개발 * 디버거 기능을 활용한 디버깅 * 동적 메모리할당, 포인터, 구조체, 파일 입출력등 C언어 고급 기술 활용 가능 * C코드 분석 가능 |
| 버전관리 | Visual SVN  Tortoise SVN | * Subversion툴을 활용한 소스 코드 버전 관리 * Project import, check in, check out등 |
| C코드 분석 | Source Insight | * Project 내 함수 원형 찾기, 전역 변수 찾기 등 |
| 펌웨어 | ATmega128 펌웨어 프로그래밍 | ATmega128보드  AVR Studio v4.0  오실로스코프 | * ATmega128 레지스터 프로그래밍 * Interrupt 프로그래밍 (External , Timer, ADC interrupt등) * GPIO, Interrupt, Timer, ADC, UART, EEPROM, I2C 제어 |
| STM32F429 (ARM Cortex-M4)프로그래밍 | Nucleo-F429보드  STMCubeMX  OpenSTM32  Keil MDK-ARM  오실로스코프 | * STMCubeMX를 활용한 펌웨어 생성 * STM 라이브러리 함수를 이용한 내부 레지스터 제어 * Interrupt및 callback함수 활용 * Clock셋팅, GPIO제어, Timer셋팅, ADC, UART, I2C, PWM 제어 * ST-Link 를 활용한 JTAG 디버깅 * 오실로스코프를 이용한 I2C신호등 디버깅 가능 |
| Nuvoton Mini51FDE  (ARM Cortex-M0)프로그래밍 | Mini51FDE보드  Keil MDK-ARM | * Mini51FDE 내부 레지스터 제어 * ADC를 활용한 센서 모니터링 * PWM신호를 활용한 서보 모터 제어 * 릴레이 제어 |
| 사물인터넷 프로그래밍 | 인텔 edison 보드  Arduino IDE  Ubidots cloud | * 기본적인 리눅스 명령어 사용가능 * 리눅스 보드(edison 보드)에 네트워크 설정 * Arduino sketch코드를 활용한 센서 모니터링 * Ubidots cloud를 연결하여 센서 원격 모니터링 및 스위치 원격 제어 |
| 하드웨어 | 전자 회로 설계 | OrCAD Capture | * AVR 보드 설계 * 부품 라이브러리 제작 * DRC 체크를 통한 설계 오류 보완 가능 |
| PCB보드 설계 | OrCAD Allegro | * AVR 보드 PCB보드 설계 * Gerber 파일 제작 * Silk Top, Soldermask Top/Bottom, Top, Bottom, Drill Chart, 도면 인쇄작업 * SMD 타입과 DIP타입의 특성 고려한 배선 |
| 지멘스 PLC프로그래밍 | Siemens TIA-Portal  Siemens S7-1200  Siemens S7-1500 |  |
| FPGA설계 | Xilinx ISE Design Suite  Vivado | Verilog HDL |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **실습 과제 수행 사항** | | |  | |
| 기간 | 과목 | 실습 재료 및 개발 환경 | | 실습 과제 내용 |
| 2017.03 ~ 2017.04 | 임베디드 하드웨어 설계 | Bread board, 74시리즈 IC, wire, 인두기, 인두용 납 | | IC칩을 활용한 디지털 시계 제작 |
| 2017.05 ~ 2017.07 | AVR 회로 설계 프로젝트 | AVR 보드, ATmega128, RLC부품, DC모터, 적외선센서등  OrCAD Capture, Allegro | | 가정용 냉장고 회로 제작 및 제어  엘레베이터 회로 제작 및 제어  AVR 보드 회로 설계 및 PCB artwork |
| 2017.07 ~ 2017.08 | ARM Cortex-M 펌웨어 프로그래밍 | Nucleo-F429ZI보드, STMCubeMX, SW4STM32(Eclipse기반), Keil사의 MDK-ARM v5 | | STM32F429를 활용한 디지털 시계 제작 (Timer 인터럽트, GPIO제어를 통한 FND제어), 다양한 센서 제어 프로그래밍 |
| 2017.07 ~ 2017.09 | 스마트팩토리 기초 | TIA portal v13, S7-1200, S7-1500PLC, MPS장비 | | Siemens S7-1500을 활용한 불량품 분류기 제작 |
| 2017.08 ~ 2017.08 | 임베디드 하드웨어 실무 | Spartan보드, ATRIX-7보드, Model SIM, Xilinx ISE, Vivado | | FPGA를 활용한 디지털 시계 구현 (ATRIX-7보드에 vivado 개발 환경에서 Verilog HDL사용) |
| 2017.09 ~ 2017.09 | 사물인터넷 기반 프로그래밍 | Nuvoton사의 Mini54FDE보드, JTAG보드, bread board, Sound Sensor, 온도 센서, Soil Moisture Sensor | | Sound Sensor, 온도 센서, Soil Moisture Sensor등 센서 제어 프로그래밍  Ultrasonic Sensor와 Servo Motor를 활용한 주사위 만들기 |
| 2017.08 ~ 2017.09 | IoT 플랫폼 실무 | 인텔 에디슨 보드, Arduino IDE, Ubidots cloud환경 | | 인텔 에디슨 보드를 활용한 사물인터넷 제어 프로그래밍 (인텔 에디슨 보드에 센서와 스위치등을 장착한후 ubidots 클라우드 환경과 연결하여 원격 모니터링 및 제어) |
|  |  |  | |  |
|  |  |  | |  |
|  |  |  | |  |
|  |  |  | |  |

자바 프로그램, 웹 프로그래밍

|  |
| --- |
| 실습 과제명 : STM32 보드를 활용한 블루투스 알람시계 |
| 수행기간 2023.11.10 ~ 2023.11.24 |
| 참가인원 1명 |
| 담당업무 회로 연결 및 소스코드 작성 |
| 수행목표 - Normal, Time Setting, Alarm Setting, Music Setting 4가지 모드 구현  - 조이스틱 사용하여 모드, 시간, 알람노래 설정 제어  - 블루투스 모듈 사용하여 휴대폰으로도 모드, 시간, 알람노래 설정 제어 |
| 사용기능 GPIO, UART, ADC, I2C, External Interrupt, Timer, RTC, DMA, Flash Memory |
| 소스 저장소 https://github.com/Minhee713 |

개발환경 : STM32Cube IDE 1.13.2

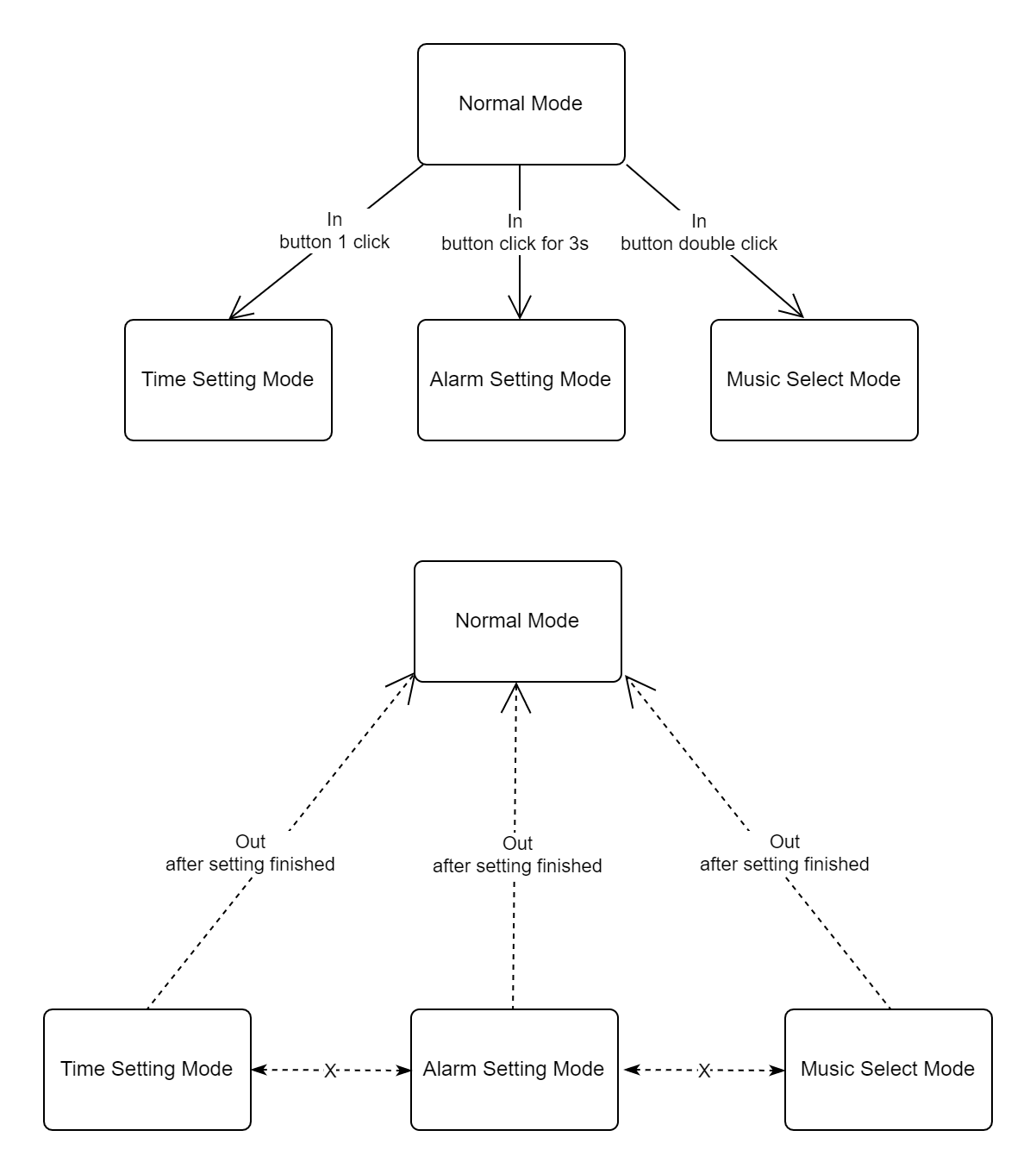
개발보드 : Nucleo-144(Nucleo-F429)

MCU: STM32F429ZI

기능 개요

1. 설정 시간 LCD에 출력
2. 알람 시간 및 노래 설정
3. 블루투스로 연결된 휴대폰으로 시간 및 알람 설정
4. Normal모드에서 현재 시간 출력. 설정된 시간부터 시간이 흐름
5. Time Setting, Alarm Setting 모드에서 각각 시간 및 알람시간 설정
6. Music Select 모드에서 알람 노래 설정. 노래 선택마다 해당 멜로디 출력
7. 모드 전환은 Normal 상태에서만 가능. Time, Alarm, Music 모드 간 전환 불가
8. 지난 세팅에서 설정한 시간, 알람시간, 노래가 저장
9. 전원을 껐다가 켜면 지난 세팅에서 설정한 내용이 불러오기 됨

상태 다이어그램



결과물

|  |  |
| --- | --- |
| 하드웨어 연결 |  |
| Normal Mode |  |
| Time Setting Mode |  |
| Alarm Setting Mode |  |
| Music Select Mode |  |

하드웨어 인터페이스 구성

|  |  |
| --- | --- |
| GPIO  (External Interrupt) | Button |
| UART | Bluetooth,  ST-Link Debugger |
| ADC | Joystick |
| I2C | 16X2 character LCD |
| PWM | Buzzer |

핀 설정

|  |  |  |
| --- | --- | --- |
| 모듈명 | 모듈 위치? 핀? | 핀 번호 |
| LCD |  |  |
| Joystick |  |  |
| Bluetooth |  |  |
| UART |  |  |

주요 함수 구현 내용

1. LCD 출력을 위해 lcd.c라는 예제 파일 사용
2. 시간 출력을 위해 STM32 보드 내부 RTC 사용
3. 조이스틱 값은 DMA 방식으로 읽어 들임
4. 모드 간 전환은 조이스틱 버튼으로 제어

* GPIO Interrupt 를 사용해 버튼을 한 번 누르면 Time Setting 모드로 진입
* 3초이상의 Long와 Double Click을 구현하기 위해 HAL\_GetTick함수 사용
* HAL\_GetTick함수로 interval이라는 시간을 계산하여, 버튼을 누른 상태에서 경과한 시간을 확인함으로써 Long, Double Click 상태를 판별
* 버튼을 3초이상 눌러 Long 상태가 되면 Alarm Setting 모드로 진입
* 버튼을 Double Click 상태로 인식하면 Music Select 모드로 진입

1. GPIO Interrupt CallBack 함수에서 모드가 정해지면 main함수로 돌아와 각 모드에 맞는 함수가 호출됨

* NORMAL\_STATE: get\_time() 함수 호출
* TIME\_SETTING: time\_set\_mode() 함수 호출
* ALARM\_TIME\_SETTING: alarm\_set\_mode() 함수 호출
* MUSIC\_SELECT: music\_set\_mode() 함수 호출

1. get\_time() 함수

* Normal모드일 때 main함수의 while반복문 안에서 계속 호출 됨
* HAL 드라이브의 RTC\_GetTime 함수를 계속 호출함으로써 시간이 흐르도록 함
* LCD 모듈에 시간을 출력하는 showCurrentTime() 함수 호출

1. time\_set\_mode() 함수

* joystick\_btn\_chk()함수로 조이스틱의 값을 확인하여 UP, DOWN, LEFT, RIGHT의 버튼상태를 확인함
* joystick\_btn\_chk():
  + y축 값이 4000 이상이면 UP 상태로 인식하여 UP 반환
  + y축 값이 1000 이하이면 DOWN 상태로 인식하여 DOWN 반환
  + x축 값이 1000 이하이면 RIGHT 상태로 인식하여 RIGHT 반환
  + x축 값이 4000 이상이면 LEFT 상태로 인식하여 LEFT 반환
  + x축 값이 2000 이상 2100 이하이면 아무 제어도 하지 않는 NOTHING 상태로 인식
* am/pm, 시분초의 커서 위치를 position이라는 변수를 통해 판별
* position 변수의 값은 0~3까지의 총 4개의 값을 가지며 각 position별로 버튼 상태에 따른 동작을 함
* position 0일 때,

컴포넌트 다이어그램? (함수 변수 상관관계)

리팩토링 과정(코드 가독성, 전역변수 최소화, 변수 이름 수정, 기능 추가 등등)

~~티스토리 블로그~~

~~깃허브 블로그~~

~~깃허브~~

여러 프로젝트 / 프로젝트 하나

하나의 프로젝트를 깊이 ???

프로젝트를 통해 배운 점

Nv items

조이스틱 제어 : DMA VS ADC + Timer