|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **예 비 보 고 서** | | | | |
| 학 과 | 학 년 | 학 번 | 조 | 성 명 |
| 전자공학과 | 3 | 12191505 |  | 윤수연 |
| 실험 제목 | LCD | | | |
| 1. **자료조사**    1. **LCD의 소개**   LCD는 Liquid Crystal Display의 약자로 문자나 그래픽을 표시하는 장치를 말한다. 보통 화면 뒤에 함께 이 화면을 control하는 컨트롤러가 붙어 있어 ATmega128과는 완전히 독립적인 장치다. 이 때 Liquid Crystal은 거시적으로는 비정질의 성질을 가지고 미시적으로는 결정의 성질을 가지는 물질로, 액정이라고도 부른다. Display가 되는 자세한 원리는 실험에서 중요하지 않으므로 아래 동영상으로 대체한다.  20x2 LCD 디스플레이 화면 LC2023 대신 HD44780 WH2002A LMB202DFW|display  hd44780|display 20x2display lcd - AliExpressLCD 구조 | 삼성 디스플레이 PID  장치의 외관을 살펴보면 총 14개의 단자(PIN)이 있다. 또한 표시영역은 문자를 기준으로 14, 16, 20문자를 2라인이나 4라인 display하는 lcd가 일반적이다. 각 핀의 역할은 이후 연결되는 내용에서 소개할 것이다.  라즈베리파이 - LCD제어] : 네이버 블로그   * 1. **LCD의 모듈의 제어**   LCD는 외부 장치이기 때문에 직접 접근이 불가하고 ATmega128 프로그래밍을 통해 레지스터를 수정하고 메모리를 이용하여 간접적으로 접근해야 한다. 이 후 소개할 LCD컨트롤러에 내장된 레지스터에도 마찬가지로 간접 접근해야 한다. 내장된 메모리는 세가지 종류, 레지스터는 두 가지 종류가 있는데, 다음과 같다. 아래 이미지는 LCD컨트롤러의 내부 구조를 나타낸 이미지다. 크게 보면 입출력 버퍼를 통과하여 각 레지스터에 연결되고 RAM과 ROM통해 실제 LCD판넬까지 data가 전달되는 구조다.  AVR(ATmaga128)로 텍스트LCD 장난하기   1. 명령 레지스터(IR)   LCD 모듈에 작업 명령을 내리기 위한 레지스터. 환경을 어떻게 사용할 것인지 설정한다.   1. 데이터 레지스터(DR)   ATmega128을 통해서 LCD컨트롤러에 명령어 자체를 넣어주거나 출력할 데이터를 넣어주는 실질적인 데이터를 담는 레지스터다. 반대로 데이터를 읽어 올 때도 필요하다. LCD모듈에 글자를 나타내기 위한 데이터 값을 기록한다. 즉 실제로 사용되는 내용을 저장하며, input, output되는 내용을 담고 있다.   1. DDRAM   출력문자를 저장하는 메모리다.   1. CGROM   아스키문자가 어떤 dot으로 구성되는지 정보를 담고 있는 메모리다. 사전에 정의한다. 이번 실험 주차에서는 건드리지 않아도 된다.   1. CGRAM   사전에 정의되지 않은 문자를 저장하는 메모리다. 커스텀 메모리라고 볼 수 있다. 이번 실험 주차에서는 사용하지 않는다.   * 1. **LCD 컨트롤러의 각 pin의 역할**   LCD 컨트롤러에는 총 14개의 모듈 단자와 Back-light 전원 control을 위한 2 단자가 있다. 이중 우리는 14개의 모듈단자에 집중하여 그 역할을 알아보자.  Arduino] LCD Control   * 핀 1~3: 전원을 control하는 핀들. 여기에 전원공급 회로를 연결하여 사용해야 한다. * 핀 4~14: ATmega128의 핀들과 1:1 연결하여 레지스터를 제어하는 핀들. 4~6번 핀의 3개의 핀이 control pin으로 LCD를 control하고 7~14번 핀의 8개의 핀이 data pin으로 8bit data송수신을 위한 핀으로 동작한다. 이 핀들은 ATmega128의 핀들 중 GPIO로 사용할 핀과 연결하여 통신한다. 이 때 ATmega128에는 총 53개의 I/O port가 있으므로 그 중 아무거나 사용해 연결해주면 된다. * 핀 4: 핀 RS. 제어할 레지스터를 설정한다. H/L로 IR과 DR을 선택함. * 핀 5: 핀 R/W. 선택한 레지스터에서 데이터를 읽을지 쓸지 선택함. * 핀 6: 핀 E. E는 Enable의 약자로 말 그대로 LCD모듈의 통신 허가를 선택함. 아래 DB로 시작하는 핀들을 제어하려면 Enable시켜야한다. 반대는 Disable이다. * 핀 7~14: number순서대로 핀 DB0부터 핀 DB7이라고 함. 8개의 핀으로 8bit data를 사용할 수 있으며 지정된 값을 쓰거나 읽을 때 사용한다. 이 때 4bit data를 이용할 때에는 핀 11~14, 즉 DB4~ EB7의 핀을 사용한다.   1. **LCD 컨트롤러의 사용**   지금까지 LCD 컨트롤러의 각 핀의 역할에 대해 알아보았다. 구체적으로 LCD 모듈의 구동방법은 아래 이미지와 같다. 이 때 커서라는 개념을 알아야 한다. 컴퓨팅에서 커서(cursor)는 컴퓨터 모니터 등의 디스플레이 장치 위에서 입력을 기다리거나 위치를 가리키는 물체, 곧 포인터다. 모든 입력은 커서 위치에서 추가되고, DDRAM에 접근하여 주소를 설정, 즉 커서위치를 설정한다. 이 때 주의할 점은 주소체계가 7bit 체계라는 점이다.  Table을 보면 0과 1 이 아니라 알파벳으로 제어 명령이 표시된 것을 볼 수 있는데 어렵게 생각할 것이 아니라 기능의 on/off를 나타낸다. 예를 들어 display on/off control기능의 DB0pin을 보면 B라고 적혀 있다. 이는 Blink(깜빡임)을 설정하는 bit로 0은 Blink 기능을 off, 1은 on을 나타낸다.     1. **실험**    1. **실험 예상**       1. **실습 1단계**   LCD 1-라인에 자신의 이름을 출력  제공되는 1단계 코드를 보니 cmd제어 부분이 많다.   * + 1. **실습 2단계**   LCD 1-라인에 최대 길이를 넘어가는 긴 문자열을 자동 shift 되어 가며 출력  • 회로도는 1단계와 동일  • ABCDEFGHIJKLMNOPQRSTUVWXYZ 출력  • ABCDEFGHIJKLMNOP 부터 출력을 시작하여 1초에 오른쪽으로 한 칸씩 shift  • KLMNOPQRSTUVWXYZ 까지 출력되고 나면 1초 뒤에 다시ABCDEFGHIJKLMNOP부터 출력 반복  문자가 흘러가듯 자동으로 shift되는 출력을 display하는 단계다. 자동으로 shift한다는 점에서 반복문을 넣고 1초의 시간을 delay 함수를 이용하여 delay를 준 후, if 조건문을 이용하여 문자열을 끝까지 shift한 후 문자열의 첫번째로 다시 돌아오는 코드를 작성하면 될 것 같다.   * + 1. **실습 3단계**   10 x 40 텍스트 정보를 바탕으로 스위치 입력 (상/하/좌/우)을 받아 그에 맞는 위치의 내용을 2-라인 LCD를 통해 출력    4개의 각 스위치 가 상, 하, 좌, 우 닌텐도의 버튼처럼 동작하는 단계다. 간단한 e-book을 구현하는 내용인데, 가장 어려운 단계다. 커서의 위치를 그 때 그 때 바꾸는 것이 관건이기 때문에 커서의 움직임을 잘 이해하고 구현해야 할 것이라고 생각한다. | | | | |
|  | | | | |