



C# & Robot

1) 2) 로봇...



진행 목차

•4. 오픈지그웨어 특수 기능 사용하기

- 타이머 클래스 사용
- 파일 관리
- 시리얼 통신 / 소켓통신
- 조이스틱 제어
- 모터제어
- 매니플레이터 제어



진행 목차

- 소프트웨어 다운로드 (다이나믹 셀 워저드 2.0)

- http://www.robotis.com/service/downloadpage.php?ca_id=10

- E 매뉴얼

- <http://emanual.robotis.com/docs/kr/>

-

2. 2. EEPROM 영역

주소	크기(Byte)	명칭	접근	기본값	범위	단위
0	2	Model Number	R	1,060	-	-
2	4	Model Information	R	-	-	-
6	1	Firmware Version	R	-	-	-
7	1	ID	RW	1	0 ~ 253	-
8	1	Baud Rate	RW	1	0 ~ 7	-
9	1	Return Delay Time	RW	250	0 ~ 254	2 [μsec]
10	1	Drive Mode	RW	0	0 ~ 1	-
11	1	Operating Mode	RW	3	0 ~ 16	-
12	1	Secondary(Shadow) ID	RW	255	0 ~ 252	-
13	1	Protocol Type	RW	2	1 ~ 2	-
20	4	Homing Offset	RW	0	-1,044,479 ~ 1,044,479	1 [pulse]
24	4	Moving Threshold	RW	10	0 ~ 1,023	0.229 [rev/min]
31	1	Temperature Limit	RW	72	0 ~ 100	1 [°C]
32	2	Max Voltage Limit	RW	140	60 ~ 140	0.1 [V]
34	2	Min Voltage Limit	RW	60	60 ~ 140	0.1 [V]
36	2	PWM Limit	RW	885	0 ~ 885	0.113 [%]
44	4	Velocity Limit	RW	265	0 ~ 1,023	0.229 [rev/min]
48	4	Max Position Limit	RW	4,095	0 ~ 4,095	1 [pulse]
52	4	Min Position Limit	RW	0	0 ~ 4,095	1 [pulse]
63	1	Shutdown	RW	52	-	-

2. 3. RAM 영역

주소	크기(Byte)	명칭	접근	기본값	범위	단위
64	1	Torque Enable	RW	0	0 ~ 1	-
65	1	LED	RW	0	0 ~ 1	-
68	1	Status Return Level	RW	2	0 ~ 2	-
69	1	Registered Instruction	R	0	0 ~ 1	-
70	1	Hardware Error Status	R	0	-	-
76	2	Velocity I Gain	RW	1,000	0 ~ 16,383	-
78	2	Velocity P Gain	RW	100	0 ~ 16,383	-
80	2	Position D Gain	RW	4,000	0 ~ 16,383	-
82	2	Position I Gain	RW	0	0 ~ 16,383	-
84	2	Position P Gain	RW	640	0 ~ 16,383	-
88	2	Feedforward 2nd Gain	RW	0	0 ~ 16,383	-
90	2	Feedforward 1st Gain	RW	0	0 ~ 16,383	-
98	1	Bus Watchdog	RW	0	1 ~ 127	20 [msec]
100	2	Goal PWM	RW	-	-PWM Limit(36) ~ PWM Limit(36)	-
104	4	Goal Velocity	RW	-	-Velocity Limit(44) ~ Velocity Limit(44)	0.229 [rev/min]
108	4	Profile Acceleration	RW	0	0 ~ 32,767 0 ~ 32,737	214.577 [rev/min ²] 1 [ms]
112	4	Profile Velocity	RW	0	0 ~ 32,767	0.229 [rev/min]
116	4	Goal Position	RW	-	Min Position Limit(52) ~ Max Position Limit(48)	1 [pulse]
120	2	Realtime Tick	R	-	0 ~ 32,767	1 [msec]
122	1	Moving	R	0	0 ~ 1	-
123	1	Moving Status	R	0	-	-
124	2	Present PWM	R	-	-	-
126	2	Present Load	R	-	-1,000 ~ 1,000	0.1 [%]
128	4	Present Velocity	R	-	-	0.229 [rev/min]
132	4	Present Position	R	-	-	1 [pulse]
136	4	Velocity Trajectory	R	-	-	0.229 [rev/min]
140	4	Position Trajectory	R	-	-	1 [pulse]
144	2	Present Input Voltage	R	-	-	0.1 [V]
146	1	Present Temperature	R	-	-	1 [°C]
168	2	Indirect Address 1	RW	224	64 ~ 661	-
170	2	Indirect Address 2	RW	225	64 ~ 661	-
172	2	Indirect Address 3	RW	226	64 ~ 661	-
...	-	-

XL430-W250

AX-12+ /12A

2. 2. EEPROM 영역

주소	크기(Byte)	명칭	의미	접근	초기값
0	2	Model Number	모델 번호의 바이트	R	12
2	1	Firmware Version	펌웨어 버전 정보	R	-
3	1	ID	다이나믹셀 ID	RW	1
4	1	Baud Rate	다이나믹셀 통신 속도	RW	1
5	1	Return Delay Time	응답 지연 시간	RW	250
6	2	CW Angle Limit	시계 방향 한계 각도 값의 바이트	RW	0
8	2	CCW Angle Limit	반시계 방향 한계 각도 값의 바이트	RW	1023
11	1	Temperature Limit	내부 한계 온도	RW	70
12	1	Min Voltage Limit	최저 한계 전압	RW	60
13	1	Max Voltage Limit	최고 한계 전압	RW	140
14	2	Max Torque	토크 한계 값의 바이트	RW	1023
16	1	Status Return Level	응답 레벨	RW	2
17	1	Alarm LED	알람용 LED 기능	RW	36
18	1	Shutdown	알람용 셧 다운(Shut down) 기능	RW	36

2. 3. RAM 영역

주소	크기(Byte)	명칭	의미	접근	초기값
24	1	Torque Enable	토크 켜기	RW	0
25	1	LED	Status LED On/Off	RW	0
26	1	CW Compliance Margin	CW Compliance Margin	RW	1
27	1	CCW Compliance Margin	CCW Compliance Margin	RW	1
28	1	CW Compliance Slope	CW Compliance Slope	RW	32
29	1	CCW Compliance Slope	CCW Compliance Slope	RW	32
30	2	Goal Position	목표 위치 값의 바이트	RW	-
32	2	Moving Speed	목표 속도 값의 바이트	RW	-
34	2	Torque Limit	토크 한계 값의 바이트	RW	ADD 14\&15
36	2	Present Position	현재 위치 값의 바이트	R	-
38	2	Present Speed	현재 속도 값의 바이트	R	-
40	2	Present Load	현재 하중 값의 바이트	R	-
42	1	Present Voltage	현재 전압	R	-
43	1	Present Temperature	현재 온도	R	-
44	1	Registered	Instruction의 등록 여부	R	0
46	1	Moving	움직임 유무	R	0
47	1	Lock	EEPROM 잠금	RW	0
48	2	Punch	Punch 값의 바이트	RW	32

exam1

- Printf, printf_Error
- 조건문 :
 - if ~ else if ~ else
- 반복문 :
 - for
- 입력된 글자의 숫자/문자 판별 :
 - Ojw.CConvert.IsDigit(문자)


exam2 - 파라미터 저장, 가상 키보드

The screenshot shows a Windows Form titled "Form1" with standard minimize, maximize, and close buttons. The form contains the following controls:

- Two single-line text boxes on the left, containing the numbers "3" and "2" respectively.
- A multi-line text box below them containing the text "4", "5", "6", and "hh" on separate lines.
- Two checkboxes on the right, both labeled "checkBox1". The top checkbox is unchecked, and the bottom checkbox is checked.
- A dropdown menu below the checkboxes, currently displaying "111".
- Three radio buttons at the bottom right, all labeled "radioButton1". The top radio button is selected (indicated by a filled circle).

만들어 보기

- 입력을 2개 받는다.
 - `Ojw.scanf()` 이용
 - 변수 2개에 각각 값을 넣어둔다.
 - `String strValue0; // → 작은 값`
 - `String strValue1; // → 큰 값`
- 각 값을 숫자인지 확인하여(`if` 조건문, `Ojw.IsDigit()` 활용) 숫자라면 `int` 변수를 만들어 각각 변환하여 저장한다.
 - `int nValue0 = Ojw.Cconvert.StrToInt(strValue0)`
 - `int nValue1 = Ojw.Cconvert.StrToInt(strValue1)`
- 두 숫자간에 존재하는 모든 값들을 더해서 출력한다.
 - `for (int i = nValue0; i <= nValue1; i++) { } 활용`



반복문을 돌리는 동안 프로그램은 죽는다. 그럼?

- Application.DoEvent
- Ojw.CTimer.Wait(1);


난 그래도 1부터 100까지 글자를 천천히 찍어보고 싶어...
프로그램이 죽지 않게...

- **Timer** 컴포넌트

- 일정 시간마다 이 컴포넌트의 함수가 자동으로 실행된다.
- 다른 컴포넌트의 제어가 가능하다.
- 정확한 시간으로 동작하지는 않는다.

- **Thread**(따라하기 없음)

- 백그라운드에서 무한정 동작하는 것이 가능하다.
- 다른 컴포넌트는 사용 못한다.(일반적으로는...)
 - **Invoke** 라는 방법이 가능하나... 어렵다.
 - 예외 : **Ojw.printf** 는 사용 가능
- 함수의 동작이 빨라 보다 정확한 시간의 흐름으로 동작이 가능하다.

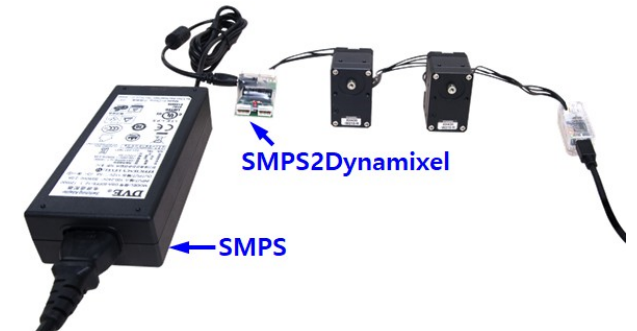
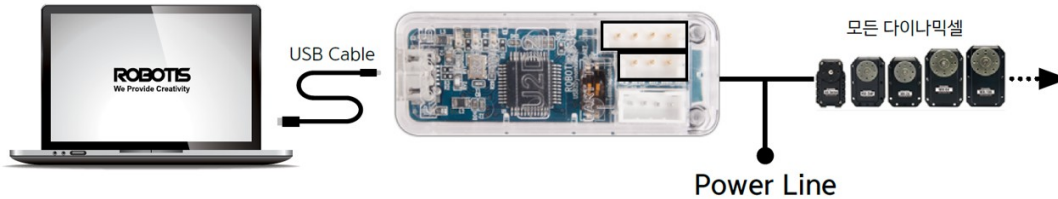


exam4 – Timer

- **Timer** 컴포넌트를 활용해 시계를 만들어 보자.

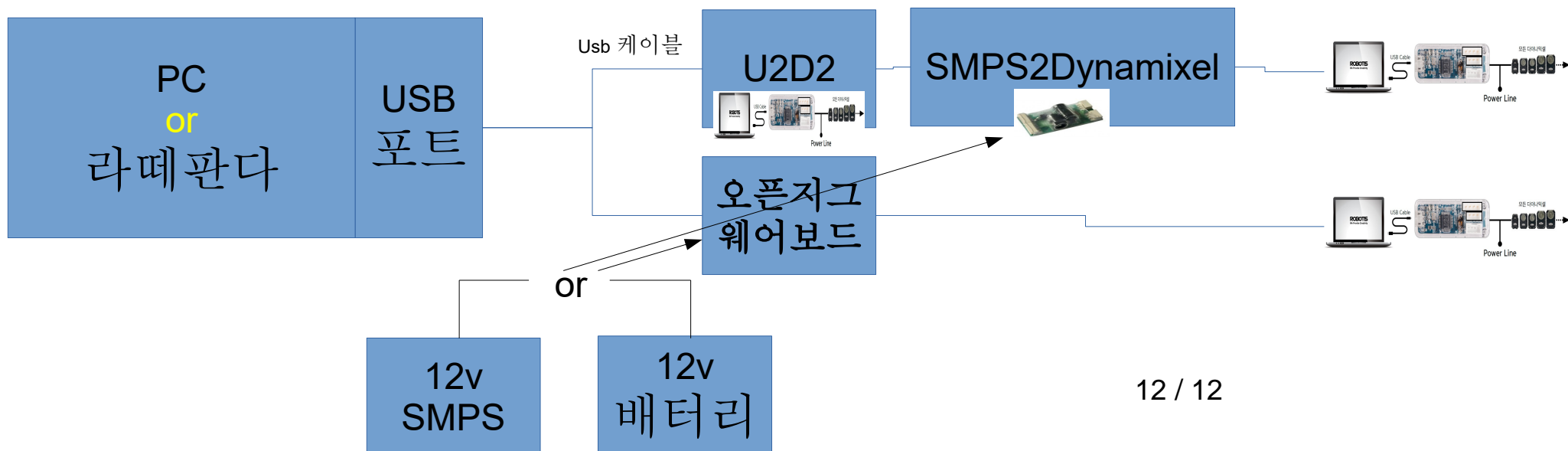
exam_Motor

• 모터의 연결



- 다양한 연결 방법

- <http://support.robotis.com/ko/product/auxdevice/inte>
- CMonster 클래스를 활용해 모터를 제어해 보자.



exam_Joystick

- Cjoystick

- 스틱

- 주의사항

- 값의 범위 : 0 ~ 1.0
 - 가운데 : 0.5 (정확히 오지 않을 수 있다. 어느정도 0.5에 가까우면 센터로 해 주는 부분이 필요)
 - 스틱에는 2개의 방향값이 같이 있다.
 - 모델마다 스틱의 값이 동일하지 않을 수 있다.

- 버튼

- IsDown_Event
 - IsDown
 - IsUp_Event
 - IsUp

- 실습 : 조이스틱으로 마우스 제어하기


주어진 환경의 매니플레이터 제어 맞보기(다음 시간을 위해)

- 3D 라이브러리 추가

- Tao ...

- 3D 그리기

- 제어...



```
1) Design
Ojw.CTools CTool = new Ojw.CTools();
CTool.ShowTools_Modeling(Ojw.CConvert.StrToFloat(txtRatio.Text));

2) Motion
Ojw.CTools_Motion CTool_Motion = new Ojw.CTools_Motion();
CTool_Motion.ShowTools(Ojw.CConvert.StrToFloat(txtRatio.Text));

3) Design + Motion
Ojw.CTools_Designer CTool_Designer = new Ojw.CTools_Designer();
CTool_Designer.ShowTools(1.0f);
```

1) Design

```
Ojw.CTools CTool = new Ojw.CTools();
CTool.ShowTools_Modeling(Ojw.CConvert.StrToFloat(txtRatio.Text));
```

2) Motion

```
Ojw.CTools_Motion CTool_Motion = new Ojw.CTools_Motion();
CTool_Motion.ShowTools(Ojw.CConvert.StrToFloat(txtRatio.Text));
```

3) Design + Motion

```
Ojw.CTools_Designer CTool_Designer = new Ojw.CTools_Designer();
CTool_Designer.ShowTools(1.0f);
```

오랫동안 꿈을 그리는 사람은

마침내 그 꿈을 닮아간다.

-앙드레 말로

감사합니다.