

eRun3.5 사용설명서

User's Engineering Maual of eRun Solution



제이피플 주식회사

JPeople Corporation



개정이력.

내용

제 1 장 eRun 소개.....	8
1.1 SCADA / HMI	8
1.2 eRun SCADA.....	8
1.3 적용분야	8
1.4 특징 및 장점	9
제 2 장 eRun 설치하기	10
2.1 시스템 요구사항.....	10
2.2 설치시작	10
제 3 장 eRun 시작하기	13
3.1 eRun Studio	13
3.2 런타임 실행하기 (eRun.exe).....	14
3.3 런타임 종료하기	16
3.4 서버관리자 실행하기.....	17
제 4 장 프로젝트 만들기	23
4.1 프로젝트 시작	23
4.2 프로젝트 생성	23
4.3 실행환경 설정	26
4.4 단축키 설정	28
4.5 로고설정	29
4.6 HSMS 설정	29
4.7 VISION 설정	33
4.7.1 소개 및 특징.....	33
4.7.2 eRun Vision 개발절차	35
4.8 프로젝트 열기 및 저장	45
4.8.1 프로젝트 열기.....	45
4.8.2 프로젝트 다른 이름으로 저장	45
4.8.3. 프로젝트 버전변환	46
4.8.4 프로젝트 불러오기	46
4.9 프로젝트 관리메뉴	47
4.9.1 프로젝트 실행 (F5)	47
4.9.2 프로젝트 배포	48
4.9.3 프로젝트 제거	48
4.9.4 프로젝트 활성화	49
4.9.5 프로젝트 폴더 바로가기	49
4.9.6 배포폴더 바로가기	49
4.9.7 프로젝트 속성	49

4.9.8 태그이름 일괄 변경하기.....	49
제 5 장 뷰 페이지 만들기	51
5.1 뷰 페이지 구성.....	51
5.1.1 오브젝트 그려 넣기	51
5.1.2 그리기 오브젝트 도구바.....	52
5.1.3 윈도우 오브젝트 도구바.....	58
5.1.4 오브젝트 정렬 도구바	59
5.1.5 글씨체 도구바.....	67
5.1.6 팝업메뉴.....	68
5.1.7 표현식 속성	71
5.1.8 오브젝트와 스크립트 함수연결.....	73
5.2 뷰 페이지 생성	74
5.3 뷰페이지 불러오기	77
5.4 뷰페이지 삭제	77
5.5 페이지명 변경	77
5.6 시작페이지 설정	78
5.7 페이지 속성	78
5.8 다른이름으로 추가	79
5.9 일반 오브젝트 속성설정	79
5.9.1 기본설정.....	80
5.9.2 표시	87
5.9.3 이동	91
5.9.4 확대축소.....	93
5.9.5 채우기	95
5.9.6 슬라이더.....	100
5.9.7 회전	104
5.9.8 입력	106
5.10 시계 오브젝트	118
5.11 라인그래프 오브젝트	119
5.12 그리드 오브젝트	125
5.13 경보 오브젝트	128
5.14 애니메이션 오브젝트	132
5.15 날짜콘트롤 오브젝트	134
5.16 콤보박스 오브젝트.....	137
5.17 리스트박스 오브젝트	138
5.18 리스트콘트롤 오브젝트	140
5.19 푸시버튼 오브젝트.....	144
5.20 체크박스 오브젝트.....	146
5.21 라디오버튼 오브젝트	147

5.22 차트 오브젝트	148
5.23 ACTIVE-X 오브젝트	151
5.24 비전 오브젝트	154
 제 6 장 스크립트.....	156
6.1 소개 및 특징	156
6.2 스크립트 프로세스 및 활용	156
6.3 스크립트 생성 및 편집	159
6.4 스크립트 파일삭제	160
6.5 스크립트 파일명 변경	160
6.6 스크립트 빌드	160
6.7 스크립트 편집메뉴 구성	161
6.8 함수의 구성	164
6.8.1 함수형 정의	164
6.8.2 변수의 정의	166
6.8.2 연산자 종류	167
6.8.3 예약어 및 기본문법	171
 제 7 장 태그관리.....	175
7.1 소개 및 특징	175
7.2 태그그룹 생성하기	175
7.3 태그종류	178
7.3.1 디지털태그	179
7.3.2 아날로그 태그	180
7.3.3 문자열 태그	182
7.3.4 경보태그	183
7.3.5 시스템 태그	185
7.4 실태그 어드레스 설정방법	185
7.5 태그편집 도구바	187
7.6 태그 일괄생성 하기	189
7.7 태그편집 팝업메뉴	189
7.8 태그명 일괄변경	191
7.9 태그리스트 엑셀파일 구성	193
 제 8 장 I/O 디바이스	196
8.1 디바이스 생성하기	197
8.2 디바이스 통신시험	199
8.3 다른 이름으로 추가하기	201
8.4 디바이스 내보내기(Export) / 불러오기(Import)	202
8.5 디바이스 관련 스크립트 함수	203
8.6 아스키코드(ASCII) 테이블	204



제 9 장 데이터베이스	205
9.1 소개 및 특징	205
9.2 데이터베이스 생성하기	206
9.3 데이터베이스 삭제	209
9.4 SQL 테이블 생성하기	209
9.5 SQL 브라우저 활용하기	211
제 10 장 보고서	215
10.1 보고서 모델 생성하기	215
10.2 보고서 삭제	218
10.3 보고서 모델 다른이름으로 추가	218
10.4 MS-EXCEL 보고서 처리용 스크립트 함수	219
제 11 장 사용자 관리	226
11.1 사용자 정보 추가하기	226
11.2 사용자 삭제	227
11.3 프로젝트에서 사용자 정보 활용	228
제 12 장 실행함수	230
12.1 실행함수 추가하기	230
12.2 실행함수 편집	231
12.3 속성창 설명	231
제 13 장 심볼관리	233
13.1 정적 라이브러리	233
13.2 동적 라이브러리(애니메이션)	235
13.3 애니메이션 파일 생성하기	237
제 14 장 프로젝트 실행	241
14.1 런타임 메뉴	242
제 15 장 다중 사용자접속	248
15.1 기능 및 특징	248
15.2 JOIN 실행	248
제 16 장 디바이스 드라이버	251
16.1 시리얼 통신이란?	251
16.2 아스키코드 테이블	257
16.3 MODBUS RTU/TCP	258
16.4 MELSEC-Q Ethernet	267
16.5 XGTFEnet	274
16.6 OMRON PLC – EtherNet/IP	279

16.7 USERNET	285
16.8 MODBUS SERVER	295
16.9 SIEMENS PLC S7-300	302

제 1 장 eRun 소개

1.1 SCADA / HMI

SCADA(Supervisory Control And Data Acquistion)는 중앙제어 및 데이터수집을 의미하며 , HMI(Human Machine Interface)는 사람과 기계의 대화를 도와주는 장치를 의미합니다. 여기서는 제조, 생산, 발전, 정제 및 기타 설비 제어 및 모니터링에 직관적인 구성도구와 강력한 엔지니어링 기능을 제공함으로써 폭발적으로 늘어나는 규모의 데이터를 효과적으로 관리하고 그 상태를 파악하도록 보여주거나 기록하는 등의 기능을 갖는 컴퓨터, 프로그램, 주변 장치를 통틀어서 표현하는 말입니다. 공항, 철도, 선박 및 우주 정거장과 같은 시설의 시설 프로세스조차도 SCADA를 사용하여 다양한 프로세스를 모니터링하고 제어합니다.

1.2 eRun SCADA

우수한 성능의 감시제어 및 데이터 수집 소프트웨어 솔루션입니다. 직관적인 UI 구성도구와 강력한 엔지니어링 기능을 제공하며, 폭발적으로 늘어나는 실시간 데이터를 효과적으로 수집하고, 수집된 BIG DATA를 통해 제조/생산 성능을 업그레이드 합니다.

eRun Platform은 사물인터넷, 빅데이터, 클라우드 컴퓨팅 기술을 접목, 활용하며 생산설비의 실시간 현장 모니터링은 물론 모바일을 통한 분석데이터를 언제든지 제공하여 완전한 지능형 공장(Smart Factory) 구축에 많은 도움이 됩니다.

특히 개방형 시스템 구조로 시스템구축 이후에 사후관리나 유지보수가 원활하게 이루어짐으로써, 관리비용 절감효과가 탁월하다고 할 수 있습니다.

eRun 으로 거의 모든 현장의 모니터링과 자동화기기 제어가 가능합니다.

1.3 적용분야

- 다양한 계측센서에 대한 제어 및 모니터링
- 장비제어 및 자동화 시스템
- 반도체, LCD FAB의 UTILITY 감시시스템
- 실험실 및 연구실에서 사용하는 다양한 계측장비 인터페이스
- BAS(Building Automation System) 구축시스템
- 생산정보화 시스템
- 기계, 정유, 화학공장 UTILITY 감시 모니터링
- 중앙 감시제어 SCADA 시스템
- 기타 모니터링 및 자동화기기 제어

1.4 특징 및 장점

- 자동화 화면구성을 위한 사용자UI 뷰페이지 편집기능
- 다양한 공정제어 구현과 사용자 요구사항의 확장성을 위한 내장 스크립트 탑재
- 가상태그 및 디바이스 태그의 무제한 사용 및 태그별 제어기능
- 각종 통신드라이버를 통한 디바이스 인터페이스 설정기능
- 프로젝트별 실행 로그파일 자동생성
- 다양한 데이터베이스 연결제공 (eRunDB, Oracle, MS-SQL, MY-SQL, MariaDB 등)
- 보고서 스크립트를 이용해서 사용자가 원하는 형태의 보고서를 생성
- 자체 양식 (SF-Form)과 MS-EXCEL 형태의 보고서 관리 기능
- 다양한 자동화 화면 및 현장감을 더해줄 수 있는 그래프 및 애니메이션 제공
- 시스템의 보안 및 사용자 권한 설정으로 시스템 접근 권한 제한기능
- 폭넓은 사용자층을 겨냥한 제품 설계
- 간단하고 깔끔한 사용자 인터페이스 구현
- 가격대비 폭넓은 적용범위와 신속한 시스템 구축지원
- 대부분의 사용자 운영체제인 마이크로소프트사 윈도우즈 OS 최적화
- 인터넷, 인트라넷 환경에서 사내, 사외 구분 없이 시스템관리
- 260 여개의 시스템 내부 함수를 제공하여 다양한 시스템기능 구현가능

제 2 장 eRun 설치하기

2.1 시스템 요구사항

원활한 프로젝트 성능 위해서는 아래와 같은 시스템을 필요로 합니다.

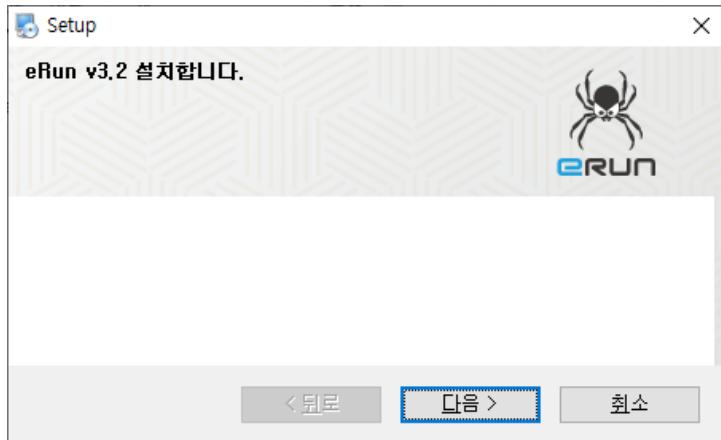
시스템	사양
CPU	Intel i3 이상
OS	MS Window 10 / 11 MS Window 2008 Server
RAM	8GB 이상 (16GB 권장)
HDD	300MB 이상의 여유공간
USB PORT	최소 1 개 (Dongle Key lock 용)

2.2 설치시작

설치 CD를 CD-ROM 드라이버에 삽입합니다.

eRun3.x.exe(32bit)파일을 실행하여 프로그램을 설치합니다.

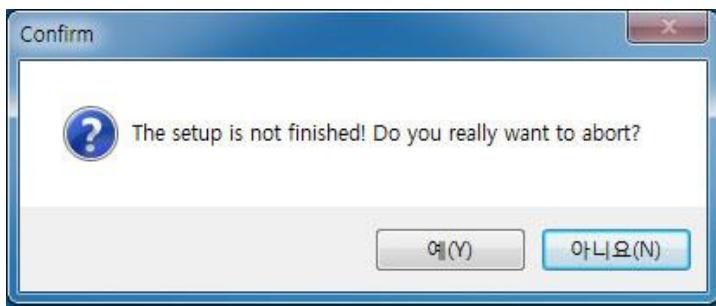
[설치화면 #01]



다음 버튼을 클릭하여 다음단계로 진행을 합니다.

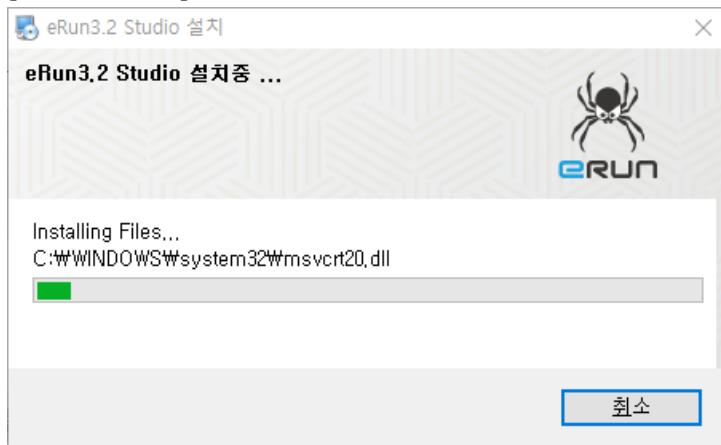
취소 버튼을 클릭하면 아래 그림과 같이 설치취소 메시지 박스가 표시됩니다.

[설치 취소 화면]



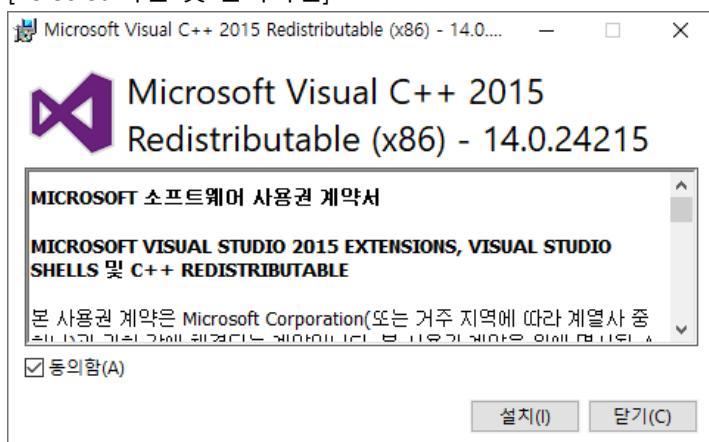
설치 취소화면에서 예(Y)를 선택하면 설치가 취소되고, 아니오(N)를 선택하면 설치를 계속 진행합니다.

[설치화면 #02]

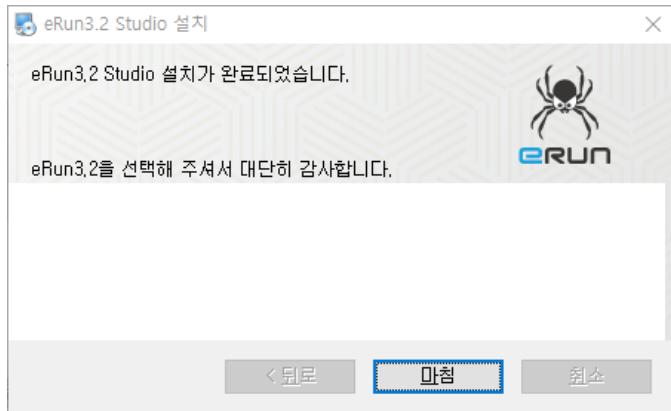


Microsoft 라이브러리 배포버전 vcredist를 자동 확인하여 최신버전이 아니거나 vcredist가 없는 경우 vcredist를 먼저 자동 설치합니다

[vcredist 확인 및 설치화면]



[설치 화면 #03 (설치완료)]



마침 버튼을 클릭하면 프로그램 설치가 완료 됩니다.

제품키 설치 확인

제품키가 없으면 eRun은 DEMO Version으로 구동이 됩니다.

DEMO Version에서는 한 개의 프로젝트에 I/O 드라이버는 1개만 등록사용 가능합니다.

UI화면 좌측상단에 "DEMO Version" 표시가 됩니다.

제 3 장 eRun 시작하기

3.1 eRun Studio

eRun Studio는 프로젝트 편집용 프로그램으로 신규 프로젝트 생성을 하고 감시화면 UI를 작성하고 실행을 하기 위한 프로그램입니다. 화면구성은 기본적으로 기본메뉴, 도구바, 프로젝트 관리자, 출력 메시지창으로 구성되어 있습니다.

■ 바탕화면에서 실행

eRun이 설치가 되면 바탕화면에 아래와 같이 스튜디오 실행아이콘이 설치가 됩니다.

마우스 더블 클릭으로 eRun Studio 프로그램을 실행합니다.



eRunStudio.exe 아이콘

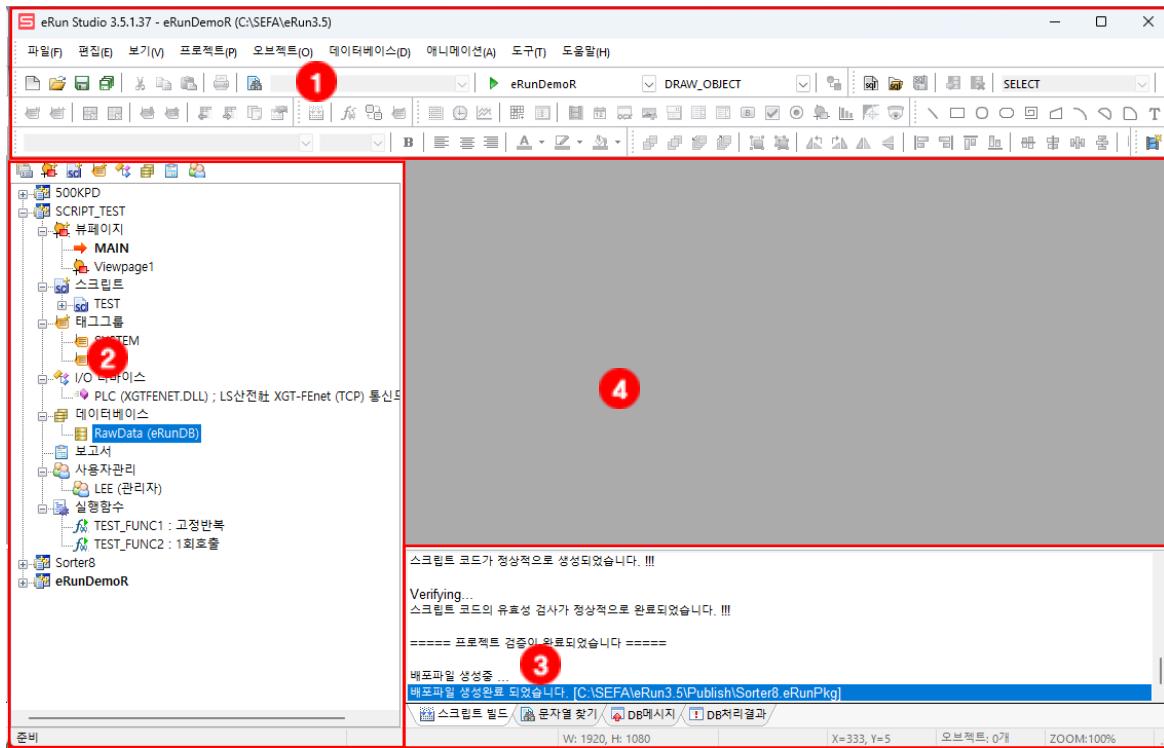
■ 시작메뉴에서 실행

[윈도우]-[시작메뉴]에서 eRun Studio 실행



편집기 실행하면 다음과 같이 실행화면이 나타납니다.

타이틀바에 보면 버전번호가 표시되고 정품 라이선스인지 표시됩니다.



① 기본메뉴 및 도구바 영역

② 프로젝트 구성 관리자 영역입니다. 프로젝트 생성, 삭제, 뷰페이지, 스크립트, 태그, IO디바이스, 데이터베이스, 보고서, 사용자관리, 실행함수 인터페이스 영역입니다.

③ 스크립트 빌드 내용표시, 문자열 찾기 결과표시, SQL 브라우저 결과메시지, SQL 쿼리결과 표시영역입니다.

④ 편집작업 영역입니다.

3.2 런타임 실행하기 (eRun.exe)

eRun은 eRun Studio에서 편집된 프로젝트를 실행하는 런타임 엔진입니다. eRun에서는 등록된 I/O 디바이스와의 통신을 통해 현장의 상태를 데이터 베이스로 저장하며, 경보, 트렌드 및 보고서 등의 기능을 실행하며, 현장의 상태를 나타냅니다.

사용자의 편의를 위하여 다양한 방법으로 eRun을 실행할 수 있습니다. 바탕화면, 윈도우 시작 메뉴에서 프로그램을 동작시킬 수 있고, 프로그램내에서 프로젝트를 바로 실행할 수도 있습니다.

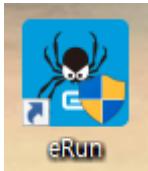
■ 바탕화면에서 실행

eRun이 설치가 되면 바탕화면에 아래와 같이 실행아이콘이 설치가 됩니다.

마우스 더블 클릭으로 eRun 런타임 프로그램을 실행합니다.



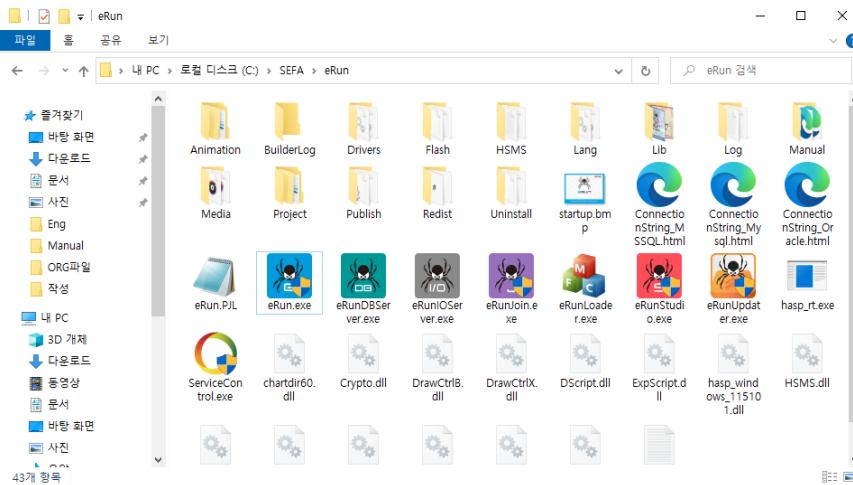
제 3 장 eRun 시작하기



바탕화면에서 eRun.exe 아이콘 더블클릭

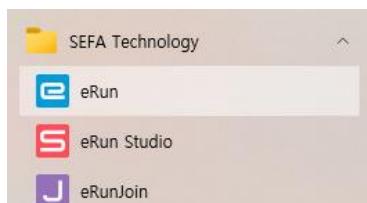
■ 폴더에서 실행

eRun이 설치경로는 "C:\SEFA\WeRun" 입니다. eRun.exe 파일을 마우스 더블 클릭하여 실행합니다.



■ 시작메뉴에서 실행

[윈도우]-[시작메뉴]에서 실행



■ 스튜디오에서 실행

Studio에서 상단 중앙에 녹색으로 된 화살표 (하단 그림 참조)를 클릭하거나, Studio에서 F5키를 클릭하여 eRun을 실행합니다.



※ eRun 실행하기 위해 프로젝트명과 뷰페이지명(상단 이미지 : 프로젝트명 – STUFFING, 시작페이지 – 메인페이지)을 확인하셔야 합니다.

※ 프로젝트가 등록이 안되었거나 (미표기) 뷰페이지가 등록이 안된경우(미표기) [7장 프로젝트] 편을 참고하시기 바랍니다.

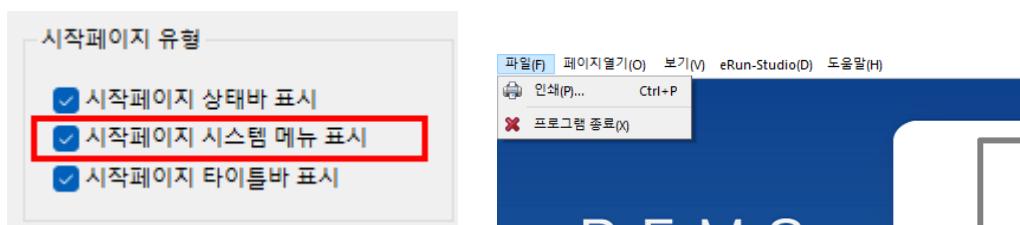
3.3 런타임 종료하기

eRun을 실행 후 종료하는 방법은 몇 가지가 있습니다. eRun Runtime을 종료한다는 것은 프로젝트를 종료하는 것과 같습니다.

■ 메뉴에서 종료

프로젝트 실행화면에 있는 메인메뉴의 “프로그램 종료” 선택해서 프로젝트를 종료합니다.

※ 메뉴에서 종료를 하기위해서는 스튜디오에서 [프로젝트구성관리자 – 프로젝트명 마우스 우클릭 – 프로젝트 속성 – 실행환경 – 시작페이지 유형 – 시작페이지 시스템 메뉴표시] 항목이 아래 적색박스 부분과 같이 체크되어 있어야 합니다.



■ 종료버튼을 이용해서 종료

프로젝트 실행화면에 캡션바의 X버튼을 클릭해서 프로젝트 종료합니다.

※ 스튜디오에서 [프로젝트구성관리자 – 프로젝트명 마우스 우클릭 – 프로젝트 속성 – 실행환경 – 시작페이지 유형 – 시작페이지 타이틀바표시] 항목이 체크 되어있어야 합니다.



■ 팝업메뉴에서 종료

프로젝트 실행화면에서 마우스 우클릭하면 팝업메뉴가 나타납니다. 여기서 제일 밑에 “프로그램 종료”메뉴를 클릭해서 프로젝트 종료합니다.

※ 메뉴에서 종료를 하기위해서는 Studio에서 [프로젝트구성관리자 – 프로젝트명 마우스 우클릭 – 프로젝트 속성 – 실행환경 – 팝업메뉴 표시] 항목이 아래 그림과 같이 설정이 되어있어야 합니다.



■ 스크립트 함수호출 종료

프로젝트 실행중 스크립트함수 실행 또는 오브젝트 클릭으로 프로젝트를 종료할 수 있습니다.
eRun 사용설명서 [일반오브젝트 – 입력 - 페이지닫기] 또는 스크립트 일반함수 _ClosePage() 참고하시기 바랍니다.

3.4 서버관리자 실행하기

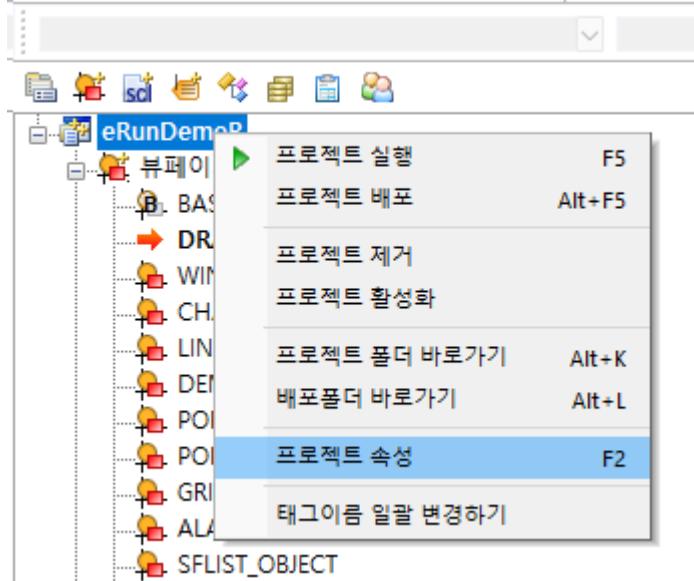
eRun 프로젝트가 시작이 되면 2개의 서비스 서버프로그램도 함께 실행이 됩니다. 그 두개의 서비스 서버프로그램은 I/O서버, D/B서버 입니다. 이 두개의 프로그램을 간단하게 설명하면, I/O서버는 디바이스 통신을 담당하고, D/B서버는 태그 값들을 파일에 저장하고 검색을 담당합니다. 뒤에서 보다 자세한 설명을 하도록 하겠습니다.

보통 프로젝트 실행시 서비스 프로그램을 자동으로 실행되게 하지만, 프로젝트 편집중에 필요에 따라서 엔지니어가 수동으로 실행하거나 중지시킬 수 있습니다.

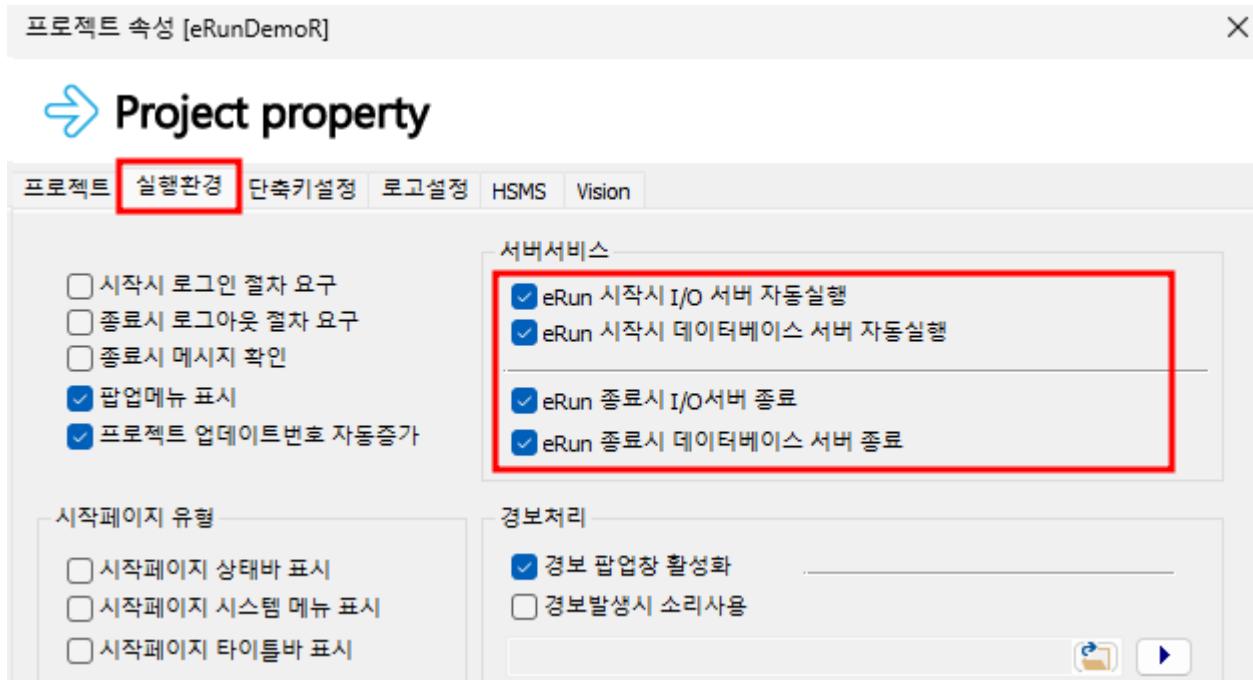
3.4.1 자동으로 시작

eRun 프로젝트 시작할 때 I/O서버, DB서버를 서비스에 등록하고 I/O서버 실행하고 D/B서버 실행합니다. 자동으로 실행되게 하기 위해서는 프로젝트 속성 - 실행환경 탭에서 다음과 같이 설정해주어야 합니다.

우선 프로젝트명에서 마우스 우클릭하면 다음과 같이 프로젝트 팝업메뉴가 나타납니다.



팝업메뉴에서 프로젝트 속성(F2) 클릭하고 실행환경 탭으로 들어가서 적색박스 영역에 있는 체크항목을 선택해 주어야 합니다.



eRun 시작시 I/O서버 자동실행은 반드시 체크해주어야 합니다.

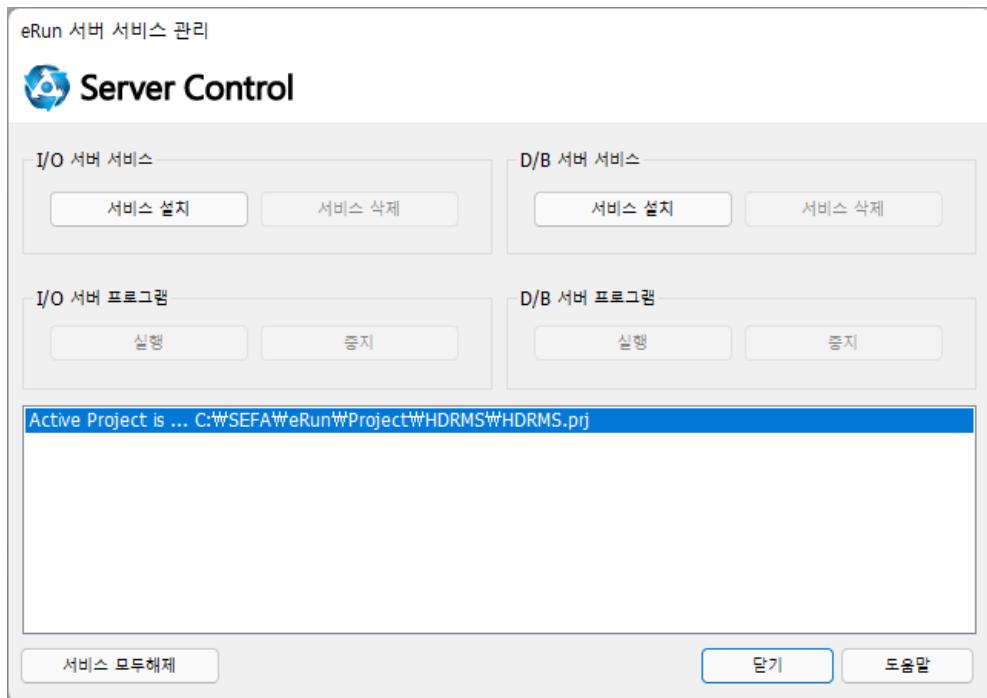


제 3 장 eRun 시작하기

데이터베이스 서버는 프로젝트에 등록된 eRunDB 모델이 있으면 체크해주고 없으면 체크해 줄 필요 없습니다.

3.4.2 수동으로 시작

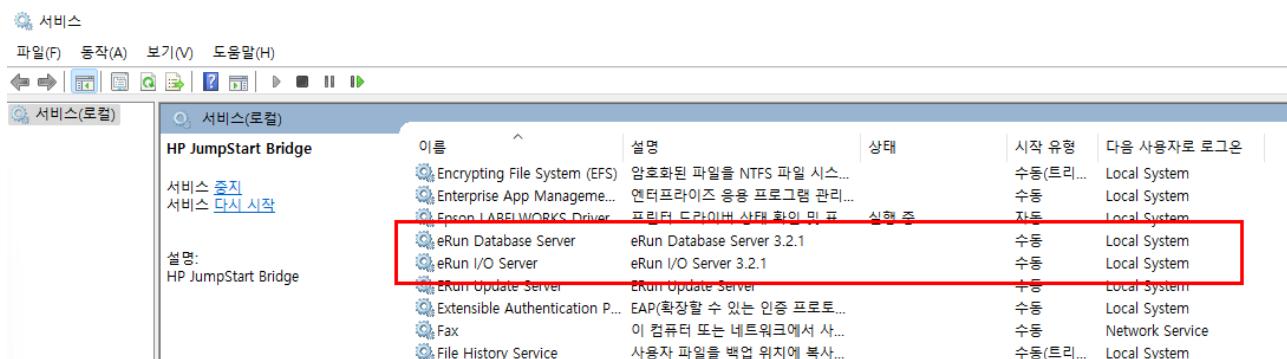
I/O서버, D/B서버를 수동으로 실행하고 종료하기 위해서 [eRun 스튜디오-도구메뉴-서버관리]를 클릭하면 아래 그림과 같이 서버관리자 창이 나타납니다. 이것은 eRun을 설치하면 같이 설치되는 ServiceControl.exe 외부 프로그램입니다.



■ I/O 서비스 설치/실행

I/O서버를 윈도우 서비스관리자에 등록합니다. 등록된 상태를 보시려면 윈도우 검색창에 “서비스”를 치면 아래와 같이 윈도우 서비스 관리자 프로그램 화면이 나타납니다.

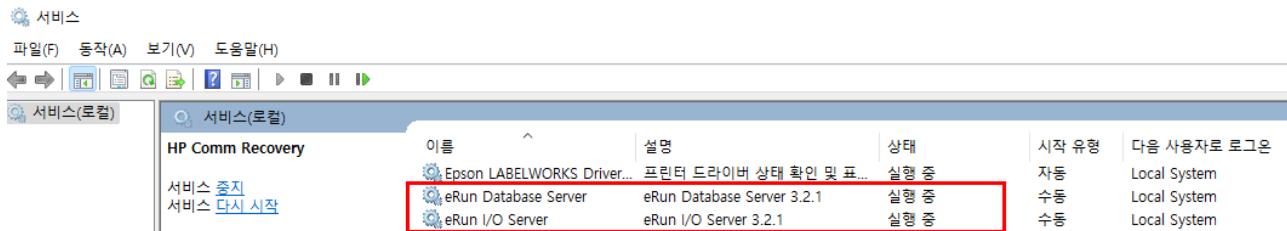
적색 박스부분에 eRun I/O Server (eRun I/O Server 3.2.1) 서비스 항목이 등록된 상태라는 것을 보여줍니다. 그러나 서비스 실행은 되지 않았기 때문에 실행상태 항목에는 공란입니다.



I/O 서비스 프로그램 “실행”을 클릭하면 I/O서버 서비스 프로그램이 실행이 되고 실행상태도 아래 그림과 같이 실행중으로 표시가 됩니다.



제 3 장 eRun 시작하기



■ D/B 서비스 설치/실행

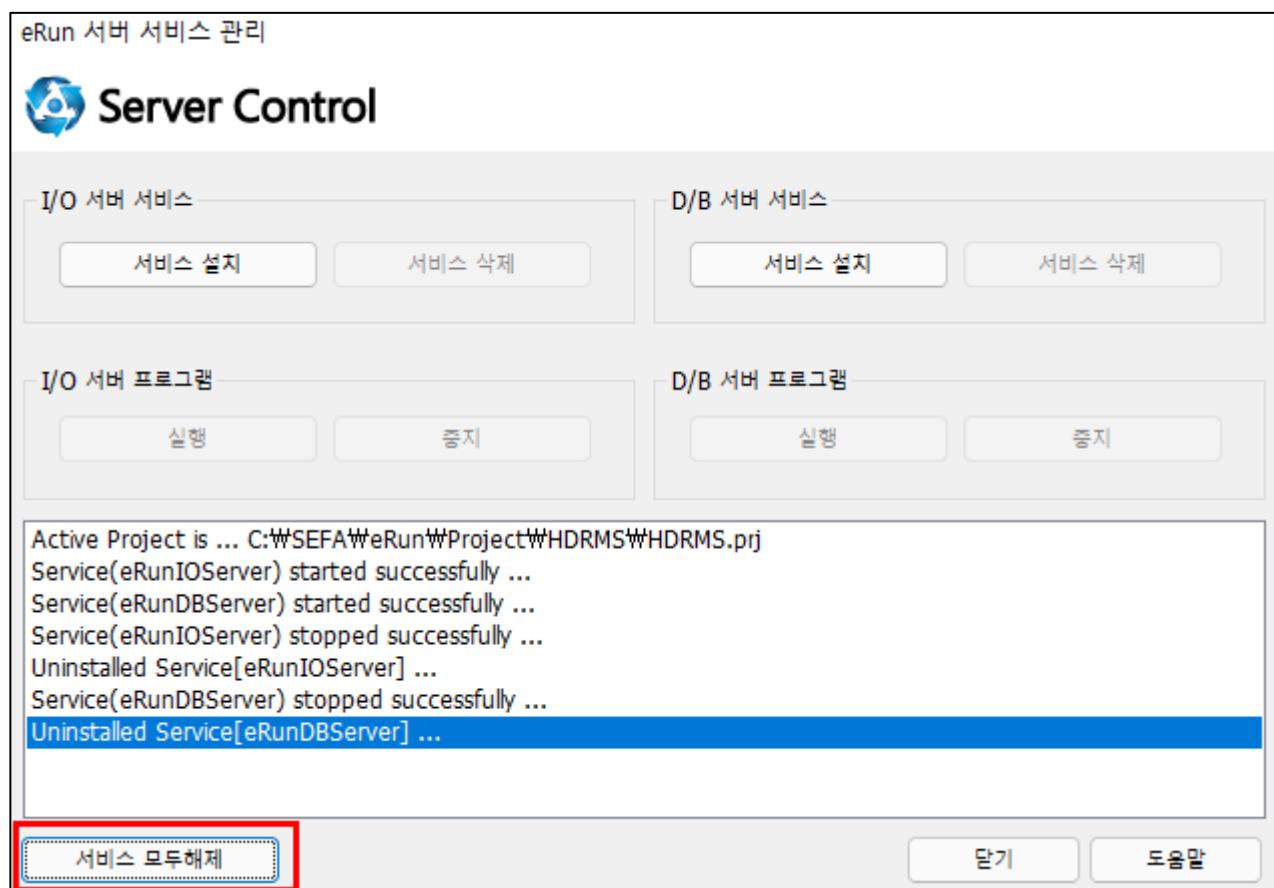
D/B서버를 윈도우 서비스관리자에 등록합니다. 등록된 상태를 보시려면 윈도우 검색창에 “서비스”를 치 위 그림에서 보면 eRun Database Server(eRun Database Server 3.2.1) 서비스로 등록된 것을 보실 수 있습니다. (I/O 서비스 설치부분 참고)

■ 서비스 삭제

I/O서버, DB서버를 서비스 관리자에서 등록을 해제합니다.

■ 서비스 모두해제

I/O서버, D/B서버 서비스 모두 실행 중지하고 서비스 등록을 해제합니다.





제 4 장 프로젝트 만들기

4.1 프로젝트 시작

eRun SCADA 시스템을 만들기 위해서 기본적으로 프로젝트를 제작합니다. 사용자 운영화면, 태그, 데이터베이스, I/O 디바이스, 보고서 등으로 구성된 파일단위의 프로젝트가 지정된 폴더에 생성이 되고 저장이 됩니다. 스튜디오에서는 동시에 최대 16개의 프로젝트를 생성하고 관리를 할 수 있어서, 이전에 작업했던 프로젝트를 복사해 오고 활용할 수 있어서 편리합니다.

4.1.1 프로젝트 구성

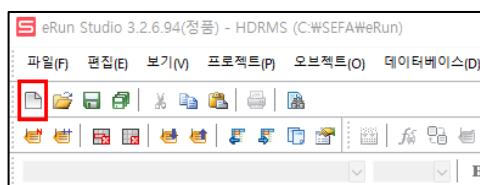
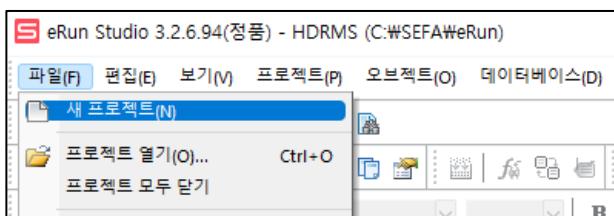
프로젝트를 생성하면 폴더는 C:\SEFA\WeRun\Project 이고, 다음과 같이 파일이 저장됩니다.

프로젝트 폴더를 임의로 변경하면 정상적으로 실행할 수 없으니 주의 하시기 바랍니다.

항목	설명
Database	데이터베이스 폴더로 DB의 기본 저장 경로입니다
Hsms	SECS/GEM 통신에 필요한 설정파일 폴더입니다.
IODevice	통신 디바이스 설정값이 저장됩니다
Report	보고서 모델설정 파일들이 있는 폴더이며, 파일저장시 기본저장 폴더입니다.
ReportTemplate	보고서에 사용되는 엑셀양식 파일을 여기에 저장합니다
Script	사용자가 작성한 Script 파일이 있는 폴더입니다
Sound	Wav (소리) 파일이 저장되는 폴더입니다. 특정소리를 넣은경우 프로젝트 복사를 할 때 사용하면 편리합니다.
Tag	태그는 파일단위(그룹)로 저장되며, 파일이 있는 폴더입니다.
Viewpage	사용자 운영화면 파일이 저장됩니다.
Vision	eRun Vision 프로젝트 제작시 셜록 비전 라이브러리에서 생성된 Inspection(.ivs)파일을 저장하는 폴더입니다
Dcode.bin	컴파일된 스크립트 바이너리 파일입니다.

4.2 프로젝트 생성

스튜디오의 주메뉴에서 [파일]-[새 프로젝트]를 선택하거나, 도구바에 있는 [새 프로젝트] 아이콘을 선택해서 프로젝트를 생성할 수 있습니다.



다음 그림과 같이 프로젝트의 세부사항을 설정하고 밑에 있는 “생성하기” 버튼을 눌르면 생성이 됩니다.

프로젝트 속성 []

X

Project property

프로젝트

속성항목	설정값
프로젝트	
업데이트 번호	1
프로젝트 이름	
프로젝트 경로	C:\SEFA\ERUN\Project
설치고객	
생성일자	2025년 02월 24일 11시 59분 01초
최종수정일자	2025년 02월 24일 11시 59분 01초
프로젝트 설명	
암호설정	False
프로젝트 암호	
로그저장(일)	10
프로젝트에 추가	True
벤더코드	1
벤더코드 검사여부	True
이미지 용량제한(KBytes)	256
I/O 서버	
IP주소	127.0.0.1
서비스포트	7001

<input type="button" value="생성하기"/>	
<input type="button" value="적용"/> <input type="button" value="취소"/> <input type="button" value="도움말"/>	

“생성하기” 버튼을 눌러서 정상적으로 프로젝트가 생성이 되면 아래와 같이 나머지 “실행환경”, “단축키설정”, “로고설정”, “HSMS”, “Vision” 탭이 나타납니다. 자세한 설명은 아래부분을 참조하시면 됩니다.

프로젝트 속성 [eRunDemo]

X

Project property

프로젝트

실행환경 **단축키설정** **로고설정** **HSMS** **Vision**

■ 프로젝트 기본속성에 대한 설명입니다

항목	설명
업데이트 번호	프로젝트가 수정된 횟수를 표시하며, 수정은 불가능합니다.

프로젝트 이름	프로젝트 이름을 설정합니다. 영문으로 입력하여야 하며, 특수문자 (%/.~!@#\$%^&*)와 스페이스는 사용할 수 없습니다.
프로젝트 경로	프로젝트 폴더명이 기본적으로 표시되며, 경로 수정은 불가능합니다.
설치고객	설치 고객 또는 현장이름을 입력합니다.
생성일자	프로젝트가 생성된 날짜입니다. 수정 불가능합니다.
최종수정일자	프로젝트가 최종적으로 수정된 날짜입니다. 수정 불가능합니다.
프로젝트 설명	프로젝트 내용을 간략하게 입력합니다.
암호설정	프로젝트 편집암호를 설정합니다. (True = 설정, False = 미설정)
프로젝트 암호	프로젝트 암호를 입력합니다. 프로젝트를 보호하려면 암호를 설정해 둡니다. 다시 열려고 할 때 암호가 맞지 않으면 편집이 불가능하니 주의하시기 바랍니다.
로그저장(일)	I/O서버, DB서버, Runtime 실행로그 파일의 보관기간을 입력합니다. (“0”이면 로그파일을 자동으로 삭제하지 않습니다.)
프로젝트에 추가	프로젝트를 생성할 때 프로젝트 관리자에 추가하는 방법을 설정합니다. True : 현재 관리자에 추가, False : 관리자리스트를 비우고 추가.
벤더코드	프로젝트 사용권한에 대한 벤더코드를 입력해서 프로젝트를 보호합니다.
벤더코드 검사여부	프로젝트 실행시 사용권한이 있는지 검사하도록 합니다. True : 검사, False : 무시
이미지 용량제한	배경이미지 또는 이미지 오브젝트 추가시 파일의 크기를 제한합니다.

■ I/O서버 속성에 대한 설명입니다

항목	설명
IP주소	Runtime, D/B서버, Join에서 사용되는 IP주소입니다. 일반적으로 127.0.0.1을 설정합니다. 단, Join을 사용할 경우에는 공인IP를 설정합니다. 127.0.0.1이 아닌 상태로 배포를 하게 되면 프로젝트 실행을 못하니 주의하시기 바랍니다.
서비스 포트	I/O 서버 서비스 포트번호를 입력합니다. (I/O서버, DB서버, Runner에서 공통으로 사용합니다. (기본값 : 7001)
접속 허용수	eRun SCADA 프로젝트를 여러 PC에서 공유할 경우 허용되는 수입니다. 이것은 라이선스에 포함된 값이며 수정은 불가합니다.
다중접속 사용여부	Join의 사용을 허용할 경우 True 설정합니다.
I/O 밸런스 조정값	디바이스 통신과 태그 값 밸런스를 조정함으로써 디바이스 반응속도를 설정합니다. (기본값 : 256) 이 값이 작을수록 반응속도는 빨라집니다. 그러나 PC의 리소스 사용량이 증가하므로 주의해서 설정하시기 바랍니다.

■ D/B서버 속성에 대한 설명입니다.

eRun SCADA에서 사용되는 태그값 저장용 eRunDB에 대한 접속정보 속성입니다.

항목	설명
IP주소	eRun replay 사용시 접속되는 IP주소입니다.
서비스 포트	서버 서비스 포트번호를 입력합니다. (기본값 : 7002) Runner에서 사용합니다.
접속 허용수	동시접속자 수를 제한합니다.
D/B저장 밸런스 조정값	태그값 저장처리 성능을 최적화하기 위한 값을 설정합니다. (기본값 512) 이 값이 작을수록 반응속도는 빨라집니다. 그러나 PC의 리소스 사용량이 증가하므로 주의해서 설정하시기 바랍니다.

■ 기본접속자 속성에 대한 설명입니다

프로젝트에 사용자관리 구성에 있는 사용자중에 선택해서 입력해줍니다.

항목	설명
사용자명	Runner에서 로그인 절차를 사용하지 않을 경우 I/O서버에 접속하기 위해서 사용합니다. 프로젝트 사용자에 등록된 사용자명을 입력합니다.
사용자 암호	기본 접속자 암호를 입력합니다.

4.3 실행환경 설정

프로젝트 시작 루틴을 변경하고 I/O서버, DB서버의 시작과 종료방법을 설정합니다.

프로젝트 속성 [eRunDemoR]



Project property

프로젝트 **실행환경** **단축키설정** **로그설정** **HSMS** **Vision**

서버서비스

시작시 로그인 절차 요구
 종료시 로그아웃 절차 요구
 종료시 메시지 확인
 팝업메뉴 표시
 프로젝트 업데이트번호 자동증가

eRun 시작시 I/O 서버 자동실행
 eRun 시작시 데이터베이스 서버 자동실행

eRun 종료시 I/O서버 종료
 eRun 종료시 데이터베이스 서버 종료

시작페이지 유형

시작페이지 상태바 표시
 시작페이지 시스템 메뉴 표시
 시작페이지 타이틀바 표시

커서 숨기기

리플레이

기능사용
 90 일 이전내용 자동삭제

경보처리

경보 팝업창 활성화
 경보발생시 소리사용

시스템 사운드

사운드#1	
사운드#2	
사운드#3	
사운드#4	
사운드#5	

적용 **취소** **도움말**

항목	설명
로그인 절차 요구	프로젝트 실행할 때 사용자이름 및 암호를 요구합니다. 프로젝트 사용자 목록에 등록된 정보입니다.
로그아웃 절차 요구	프로젝트 종료할 때 현재 사용자 암호를 요구합니다.
종료시 메시지 확인	프로젝트 종료시 종료할 것인지 확인메시지 박스창을 띄웁니다.
팝업메뉴 표시	사용자화면 임의의 위치에서 마우스 우클릭하면 팝업메뉴창을 표시할 지 선택합니다.

프로젝트 업데이트 번호 자동 증가	스튜디오에서 프로젝트 배포작업을 할 때마다 1씩 자동으로 증가하도록 합니다. 다중 사용자 PC환경에서는 반드시 해주어야 합니다.
시작페이지 유형	<p>프로젝트 실행할 때 메인페이지 형태를 선택할 수 있습니다.</p>  <p>(1) 타이틀바 영역 – 프로젝트 제목 정보표시합니다 (2) 시스템 메뉴 표시 (3) 상태바 영역표시 – ON LINE상태(I/O서버 연결)</p>
커서 숨기기	프로젝트 실행할 때 마우스커서를 안보이게 합니다.
eRun 시작시 I/O서버 자동실행	프로젝트 실행할 때 I/O서버를 자동으로 실행할 것인지 선택합니다. I/O서버 프로그램을 서비스모드로 등록하고 실행을 시킵니다.
eRun 시작시 D/B서버 자동실행	프로젝트 실행할 때 D/B서버를 자동으로 실행할 것인지 선택합니다. D/B서버 프로그램을 서비스모드로 등록하고 실행을 시킵니다.
eRun 종료시 I/O서버 종료	프로젝트 종료할 때 I/O서버 프로그램 실행을 끝냅니다. 서비스 등록해제는 하지 않습니다.
eRun 종료시 D/B서버 종료	프로젝트 종료할 때 D/B서버 프로그램 실행을 끝냅니다. 서비스 등록해제는 하지 않습니다.
경보 팝업창 활성화	경보태그에 한해서 경보가 발생 또는 해제된 경우에 경보메시지 창을 표시할 것인지 설정합니다.
경보발생시 소리사용	경보 팝업창 활성화된 경우에 소리발생 하도록 합니다.
시스템 사운드	프로젝트에서 사용할 음원파일을 등록합니다. 프로젝트 백업할 경우 필요합니다. WAV 포맷의 음원만 등록 가능합니다.

4.4 단축키 설정

프로젝트 속성 [eRunDemoR] ×

Project property

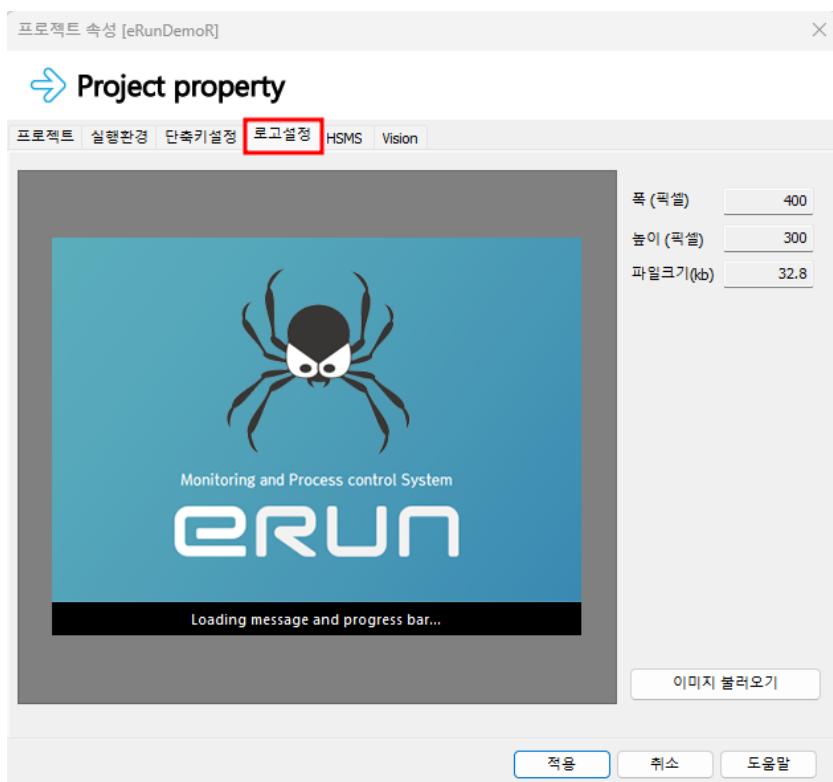
프로젝트 실행환경 단축키설정 로고설정 HSMS Vision

번호	단축키	실행함수	설명
1	Alt + F1	CHKPWD()	
2	Alt + F2	DISPLAY_TIMER_TIME()	

프로젝트 실행할 때 사용자 편의상 단축키로 특정 스크립트 함수를 호출하는 기능을 설정합니다. 단축키 조합은 ALT+F1 ~ ALT+F12 (12개), CTRL+F1 ~ CTRL+F12 (12개), Shift+F1 ~ Shift+F12 (12개)으로 최대 36까지 하나의 프로젝트에서 등록 사용이 가능합니다.

4.5 로고설정

프로젝트 실행할 때 프로젝트 로딩을 하면서 나타나는 로고창 이미지를 설정합니다. 고객사 이미지 또는 프로젝트 성격에 맞는 이미지를 등록해 주면 됩니다. 창의 오른쪽 하단의 [이미지 불러오기] 선택해서 이미지 파일을 선택합니다. 너무 큰용량의 이미지는 프로젝트 성능을 저하 시키므로 대략 가로,세로 400x300크기에 512kb 이하의 파일용량 이미지를 선택하는 것이 좋습니다.

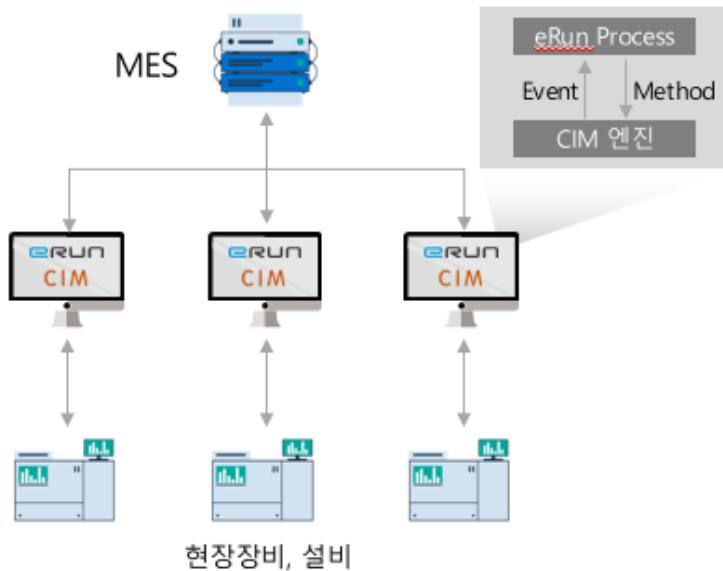


4.6 HSMS 설정

eRun CIM은 반도체와 OLED 생산공정에서 장비를 감시 제어를 하고 실시간으로 상위 HOST로의 생산보고를 통해서 효율적인 생산인프라를 구축할 수 있도록 합니다. 검증된 SECS/GEM 통신 프로토콜 라이브러리를 탑재하고 있으며, 하나의 eRun 라이선스로 공정제어, 모니터링을 통합해서 구현할 수 있는 시스템입니다.

주로 반도체공정 장비간 상위 통신에 사용되는 SECS/GEM 프로토콜 처리용 프로젝트 작업을 할 경우 기본적인 라이브러리 설정을 할 때 사용합니다. eRun SCADA에서는 그동안 현장에서 검증된 링크제네시스(주)사의 xGem300Pro 통신 드라이버를 기반으로 하고 있습니다. 다음은 eRun CIM의 특징입니다.

- 반도체 공정을 위한 SEMI 표준 프로토콜(SECS)을 쉽게 개발할 수 있도록 지원
- 간격하고 직관적인 Method 제공
- 제공되는 Configuration Tool을 이용하여 CIM Spec변경 및 적용 변경 기능제공
- eRun Studio를 활용한 레이아웃 변경 및 교체, UI확장 기능제공
- 통신모듈의 확장 및 다양성 제공
- eRun플랫폼을 이용한 다양한 HW 통합제어 가능



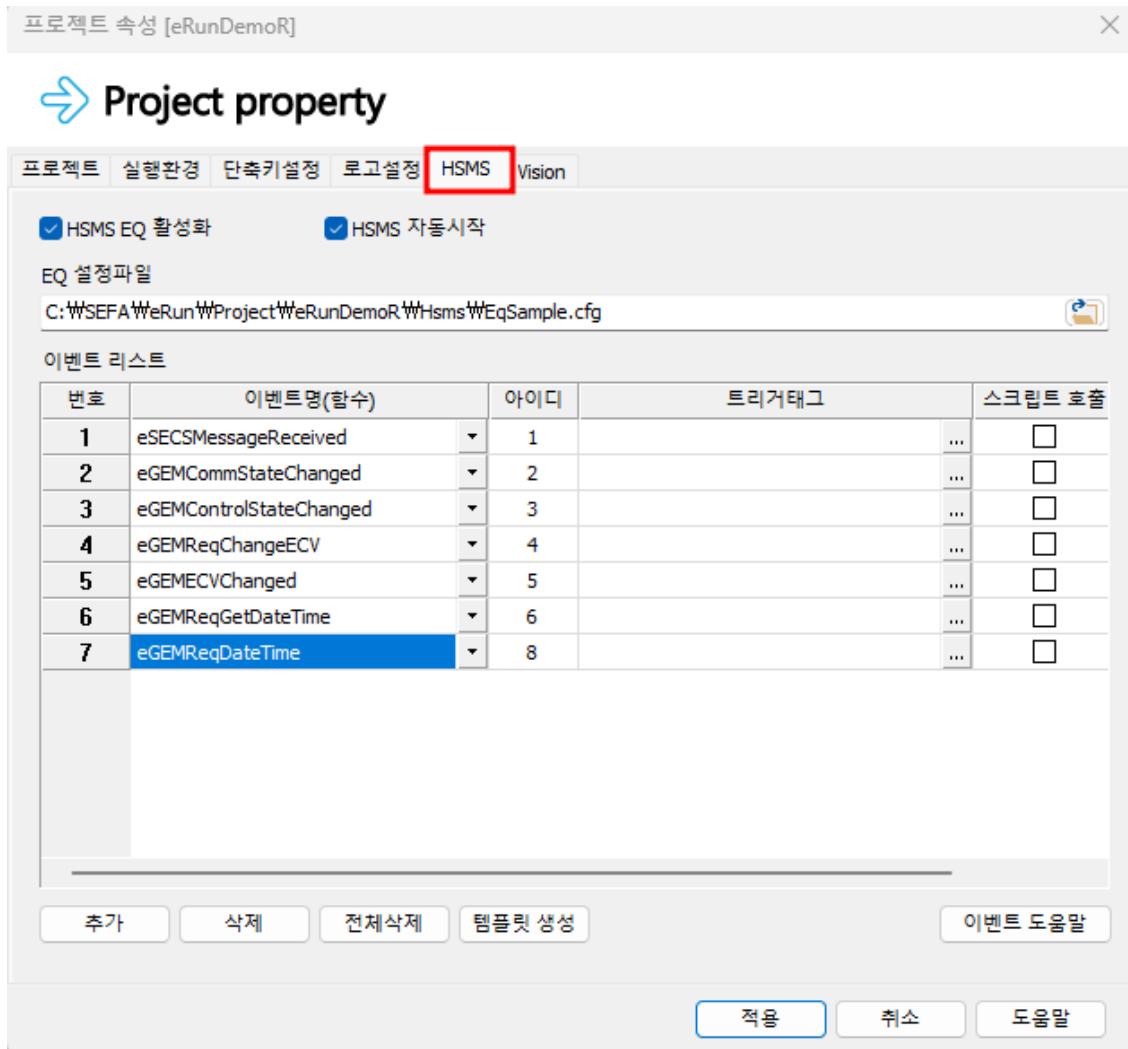
[eRun CIM 시스템 구성]

일반적인 CIM 프로그램 개발 방법은 아래와 같은 순서를 따릅니다.

1. EQ의 특성에 따라 HOST와 통신하는 시나리오를 정리합니다.
2. EQ에서 표출할 수 있는 각종 이벤트(CEID)와 데이터(VID)를 정리합니다.
3. HOST가 요구하는 Stream and Function을 정리합니다.
4. 링크제니시스사의 Configuration Tool을 이용하여 앞서 정리한 내용을 작성합니다.
5. eRun에서 위의 결과파일을 로딩합니다.
6. 작화작업 및 이벤트와 메써드들을 작성합니다.
7. 최종 결과파일을 작성합니다.

본 문서는 위의 내용중 5, 6, 7에 해당하는 후반부 작업에 대해 기술 합니다. 기타 사항은 드라이버 제조사에 문의하시기 바랍니다.

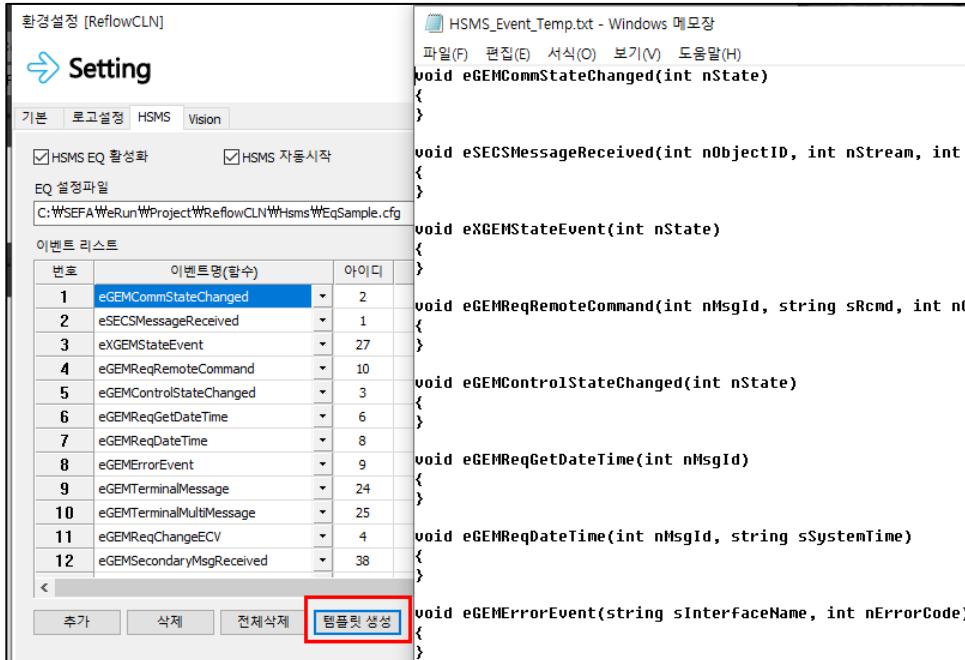
- EQ설정파일을 로딩하고 사용할 이벤트들을 추가합니다.



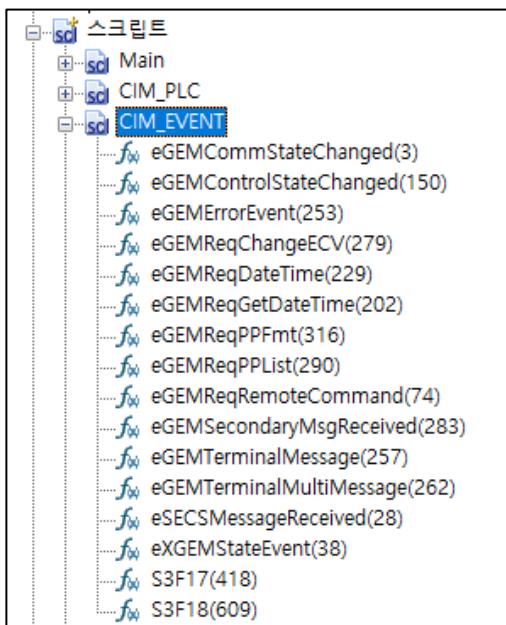
항목	설명
HSMS EQ 활성화	프로젝트 실행할 때 HSMS 라이브러리 생성할지 선택합니다. HSMS 프로젝트 구축할 경우에는 반드시 체크합니다.
HSMS 자동시작	HSMS 통신을 자동으로 시작합니다.
EQ 설정파일	HOST와의 통신에 대한 EQ 라이브러리 설정파일을 지정합니다. EQ설정파일은 링크제네시스에서 제공하는 설정프로그램으로 생성된 파일입니다.
이벤트리스트	<ul style="list-style-type: none"> ■ 이벤트명(함수) 상대 HOST측으로부터 이벤트 수신 했을 때 처리방법을 입력합니다. 이벤트 목록은 이미 정의되어 있습니다. ■ 아이디 (수정불가) 이벤트에 대한 고유 아이디번호. (수정불가) ■ 트리거 태그 이벤트 수신했을 때 트리거 신호를 설정할 태그명을 등록합니다. ■ 스크립트 호출 이벤트 수신했을 때 해당 이벤트 스크립트 함수를 실행할 것인지 설정합니다.

[추가] 사용할 이벤트를 추가합니다.

[템플릿 생성] 이벤트 수신했을 때 호출되어야 하는 스크립트 함수를 자동으로 생성해줍니다.
수행내용은 직접 작성해줍니다.



아래의 그림처럼 스크립트파일을 생성한후 템플릿으로 생성된 코드를 붙여 넣기합니다.



생성된 이벤트의 내용을 작성합니다.

```

201 //S2F17
202 void eGEMReqGetDateTime(int nMsgId)
203 {
204     string strYYYY;
205     string strMM;
206     string strDD;
207     string strHH;
208     string strmm;
209     string strSS;
210     string strDateTime;
211     string strLog;
212
213     strYYYY = _FormatString("%04d", @SYSTEM.N_YEAR) ;
214     strMM = _FormatString("%02d", @SYSTEM.N_MONTH) ;
215     strDD = _FormatString("%02d", @SYSTEM.N_DAY) ;
216     strHH = _FormatString("%02d", @SYSTEM.N_HOUR) ;
217     strmm = _FormatString("%02d", @SYSTEM.N_MIN) ;
218     strSS = _FormatString("%02d", @SYSTEM.N_SEC) ;
219
220     strDateTime = _FormatString("%s%s%s%s%5s", strYYYY, strMM, strDD, strHH, strmm, strSS);
221
222     WriteLog("[H->E] S2F17");
223
224     _HSMSMethod("GEHRspGetDateTime", nMsgId, strDateTime);
225     WriteLog("[H<-E] S2F18");
226 }
227
228 //S2F31
229 void eGEMReqDateTime(int nMsgId, string sSystemTime)
230 {
231     string strLog;
232     string strYYYY, strMM, strDD, strHH, strmm, strSS, strcc;
233     string strDateTime;
234
235     WriteLog("[H->E] S2F31");
236
237     _HSMSMethod("GEHRspDateTime", nMsgId, 0); //0: ACK, 0<: NAK
238     WriteLog("[H<-E] S2F32");
239 }
```

이때 실제 SECS통신의 Stream and Function에 해당하는 이벤트 목록은 링크제시시스(주)의 메뉴얼을 참고하시기 바랍니다.

■ eRun에서 SECS통신 METHOD 사용방법

각 이벤트의 응답에 필요한 Method를 작성합니다.

SECS통신은 훌수 번호와 짹수번호가 하나의 쌍으로 이루고 있으며,

훌수 번호로 요청을 하면, 짹수번호로 규약된 메시지 형태로 응답을 해야 합니다.

이때 매써드를 사용해야 합니다.

자세한 사항은 예제 프로젝트를 참고하시기 바랍니다.

4.7 VISION 설정

4.7.1 소개 및 특징

eRun Vision은 검증된 영상처리 라이브러리를 탑재하여 사용자의 Needs에 맞게 모션제어와 영상처리 그리고 HMI를 하나의 솔루션으로 통합한 머신비전 시스템입니다.

eRun Vision과 100% 호환되는 비전 알고리즘 라이브러리는 아래와 같이 지원합니다.

- 1) Teledyne Company의 Sherlock7 Machine vision library 7 (32bit/64bit)



- 2) Euresys Open eVision library 22.12 (64bit)



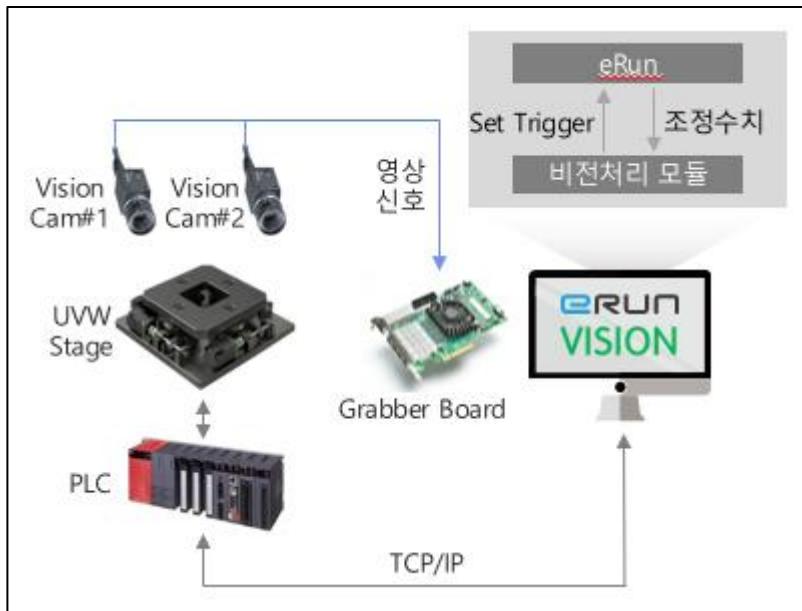
2종류의 라이브러리를 eRun에 적용시키는 방법은 약간 다릅니다.

먼저, Sherlock7 studio에서 Vision 검사설정을 파일을 생성해서, eRun에서 Plugin 형태로 Sherlock Vision Engine을 사용하게 됩니다. Vision Object를 뷰 페이지에 그려 넣고 엔진을 연결하고 event함수 및 결과태그를 연결 설정하면 기본적인 머신비전 프로젝트는 완성됩니다.

Euresys eVision에서는 일반 측정등과 같은 Rule base 방식을 사용이 가능하지만 eRun에서 사용하려면 직접 DLL 파일을 생성해야 하는 부분이 있습니다.

그러나 Deep Learning을 적용할 경우는 Euresys eVision Deep Learning 스튜디오를 이용해서 검사설정파일을 생성하고 그 생성된 파일을 eRun과 연결시켜 주면 프로젝트는 완성됩니다.

eRun에서 Vision 프로젝트를 작업할 경우에 규칙적인 검사가 필요한 경우는 Sherlock 라이브러리를 사용하고, Deep Learning으로 불규칙적인 검사가 필요한 경우는 Euresys 라이브러리를 사용할 것을 권장합니다.



eRun Vision 시스템의 특징은 다음과 같습니다.

- 사용자가 쉽게 사용할 수 있는 놀라운 영상처리 기술
- 다양한 최신 비전 카메라와 결합이 가능하며, 효율적인 비용으로 비전 솔루션을 제공
- 빠른 연동 및 규격화 제공
- 엔지니어가 쉽게 설치, 사용, 유지할 수 있고, 시스템을 이용하는데 전통적인 프로그래밍이나 비전시스템 기술에 대한 별도 경험 필요 없음
- 복잡한 머신 비전 어플리케이션을 쉽게 만들고 빠르게 입증할 수 있음
- 단일 연동 솔루션 하나에서 모든 검사 분야를 위한 기능 제공
- 기록 연동, 영상 저장 및 보안기능 내장

4.7.2 eRun Vision 개발절차

eRun Vision을 이용하여 개발하는 일반적인 절차는 아래와 같습니다.

1. Camera 세팅
2. Sherlock을 이용한 비전 알고리즘 선정
3. eRun과 데이터를 공유하기 위한 변수설정.
4. eRun에서 작화 작업 진행
5. eRun과 Sherlock간의 변수 연결
6. 최종 파일 생성

1. Camera 세팅

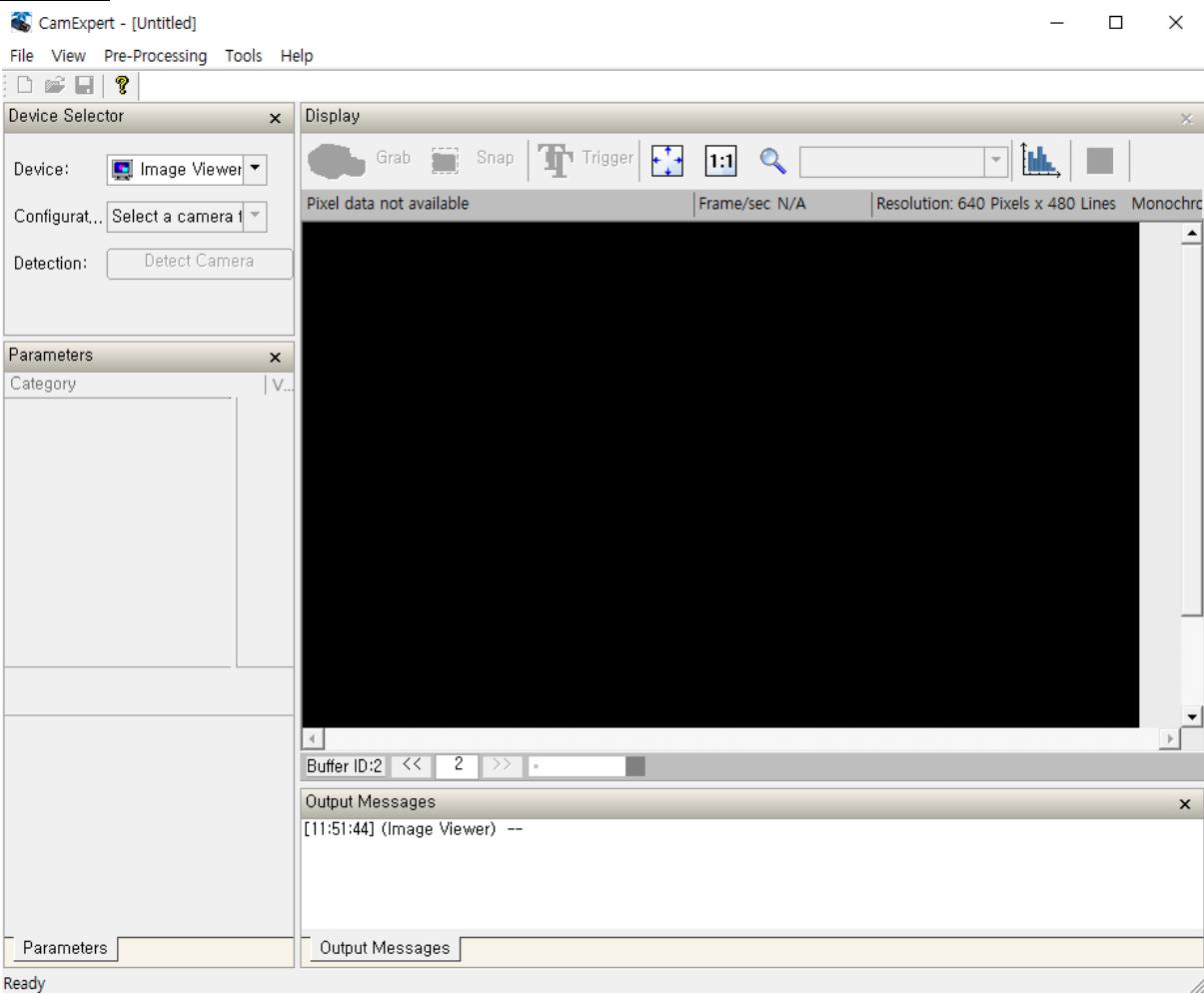
GigE 카메라의 네트워크 설정을 아래의 그림과 같이 Network Configurator를 이용하여 설정합니다. -자세한 사항은 카메라 제조사에 문의 바랍니다-



아래의 그림과 같이 Sapera CamExpert를 이용하여 카메라의 각종 파라미터들을 설정하여, 영상을 취득 테스트를 합니다. -자세한 사항은 카메라 제조사에 문의 바랍니다-

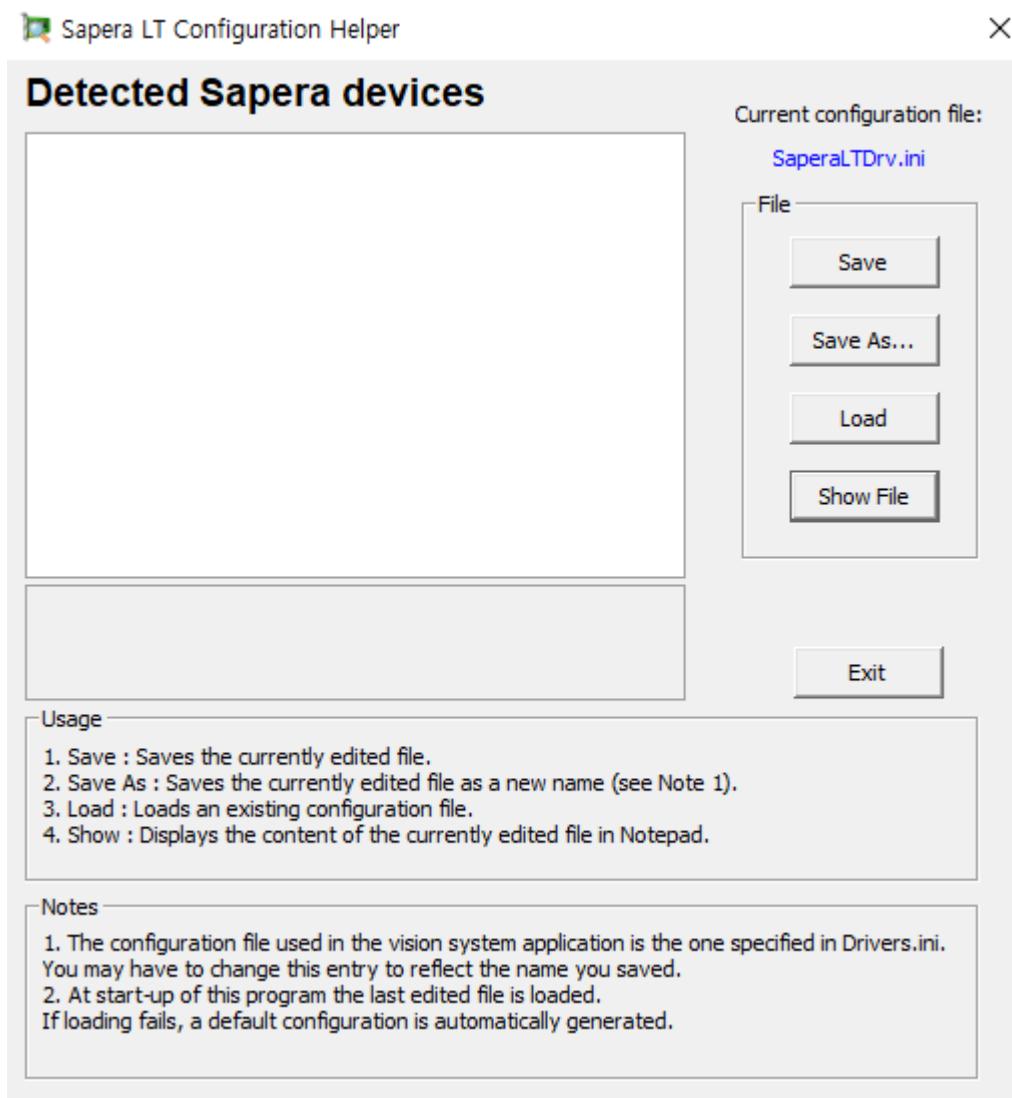


제 4 장 프로젝트 만들기



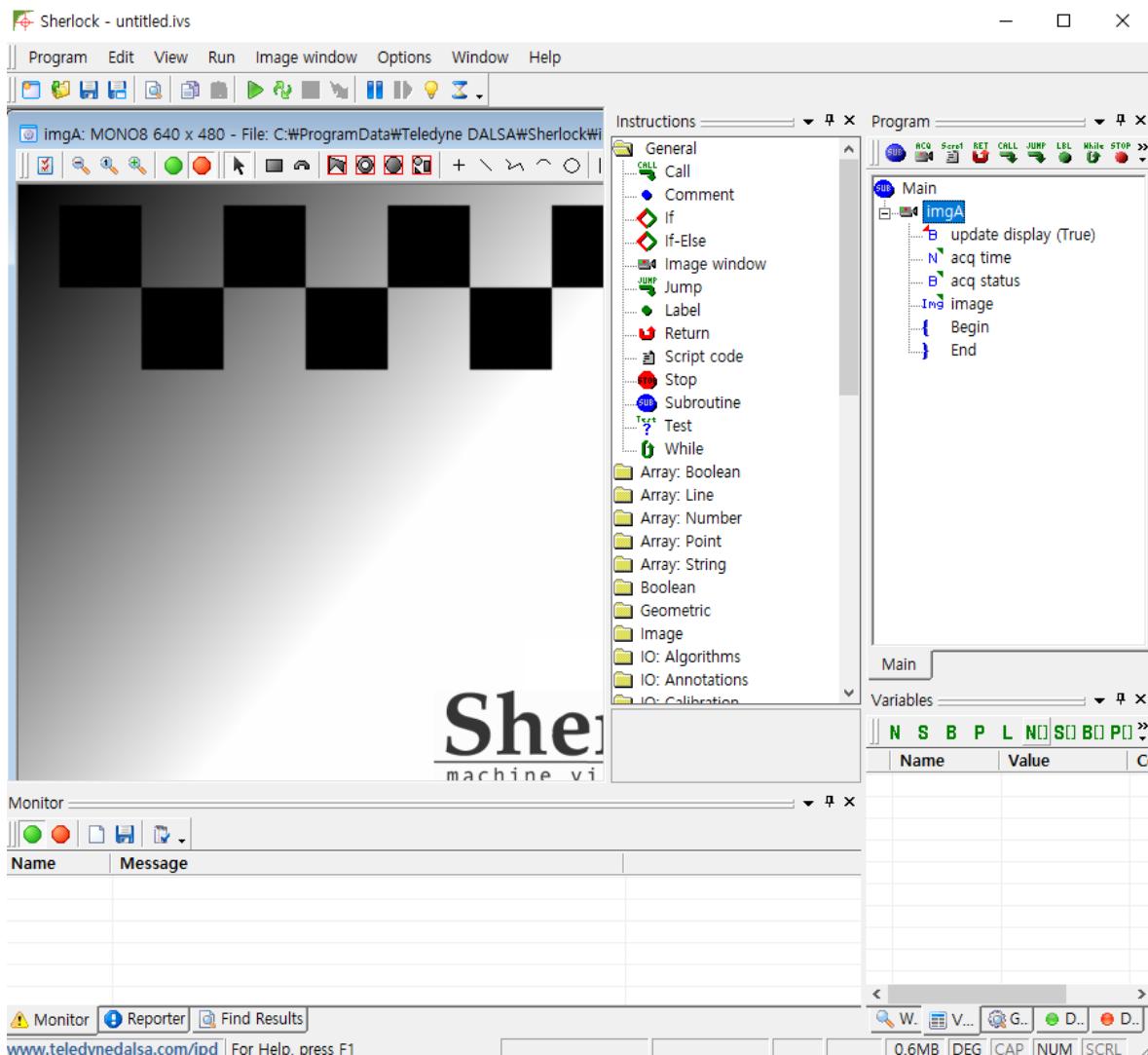
설정된 상태를 파일(.ccf)로 저장을 합니다.

Sapera Acquisition Wizard을 이용하여 Sherlock에서 사용할 카메라를 설정합니다. -자세한 사항은 카메라 제조사에 문의 바랍니다-



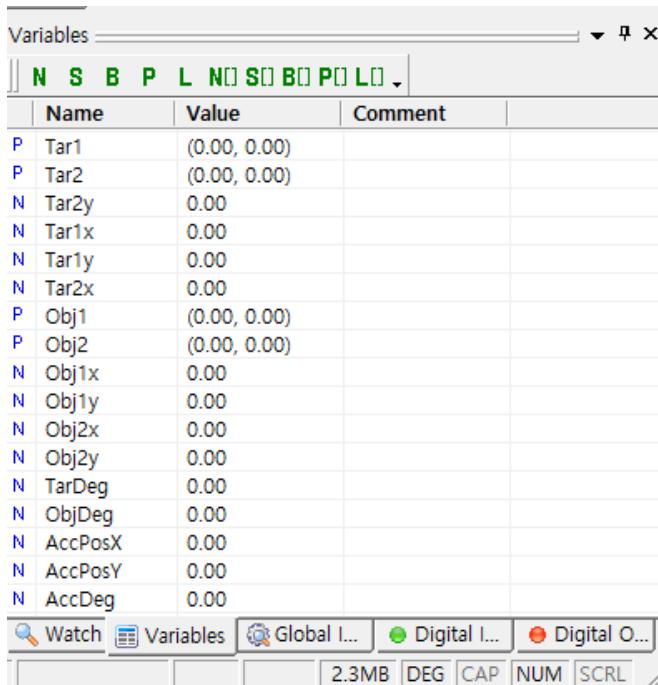
2. Sherlock을 이용한 비전 알고리즘 설정

아래의 그림과 같이 Sherlock 스튜디오 프로그램을 이용하여 비전 알고리즘 작업을 합니다.



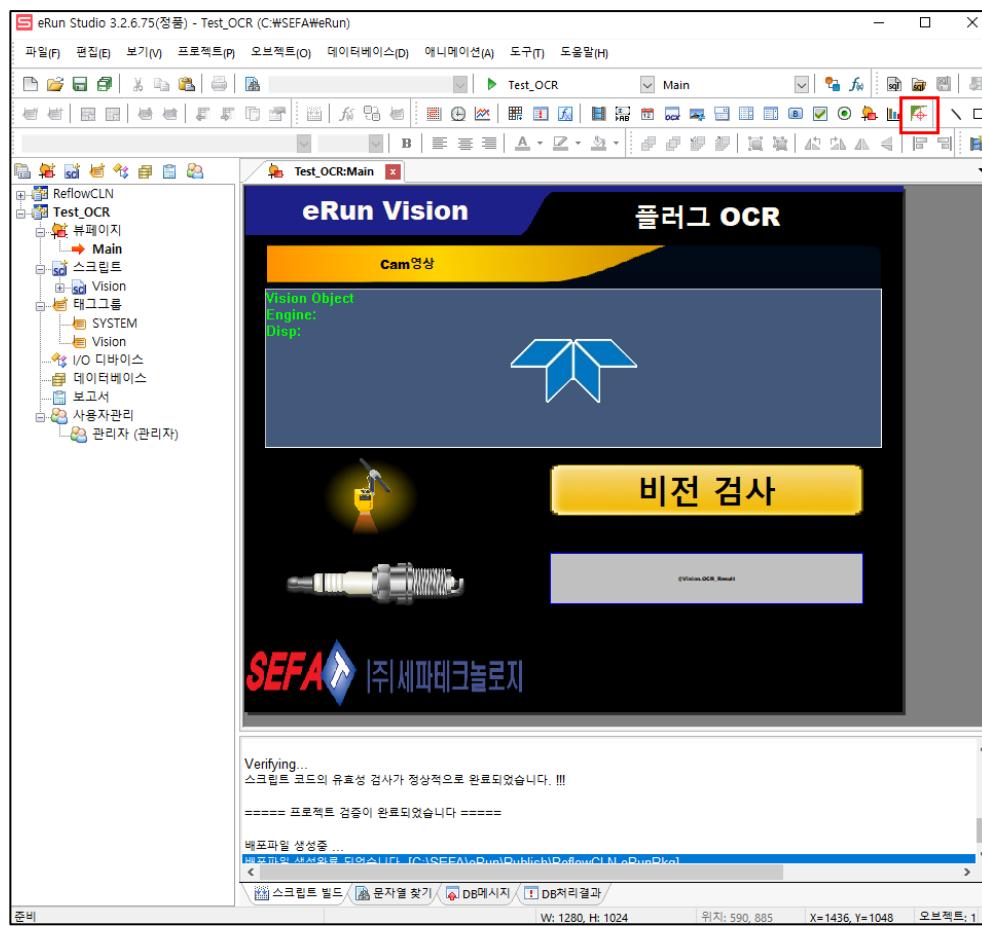
3. eRun과 데이터 공유를 위한 변수 설정

아래의 그림과 같이 데이터 타입을 고려하여 변수들을 설정합니다.

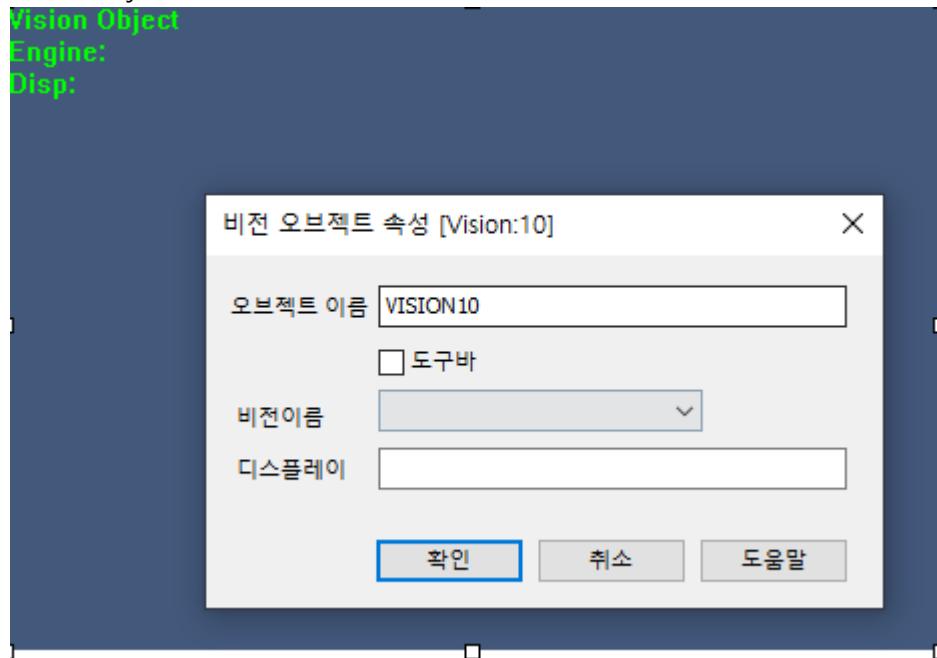


4. eRun에서 작화 작업 수행

eRun Studio를 이용하여 작화 작업을 합니다. 이때 영상 출력 및 비전 알고리즘 수행을 위한 eRunVision Object를 이용합니다.

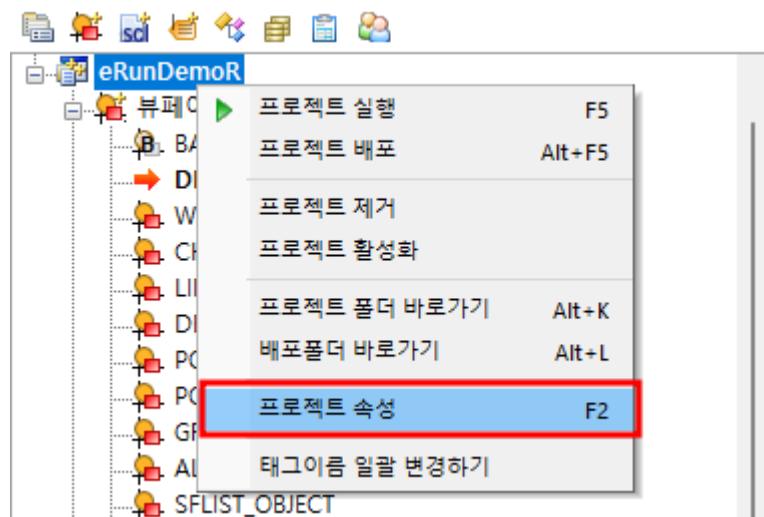


Vision Object를 더블 클릭하여 속성을 설정합니다.



5. eRun과 Sherlock간의 변수 연결

아래의 그림과 같이 프로젝트 관리자에서 [프로젝트명] – [프로젝트 속성] – [Vision] 탭을 누르면 설정할 수 있습니다.



프로젝트 속성 [eRunDemoR] ×

Project property

프로젝트 실행환경 단축키설정 로고설정 HSMS **Vision**

Sherlock

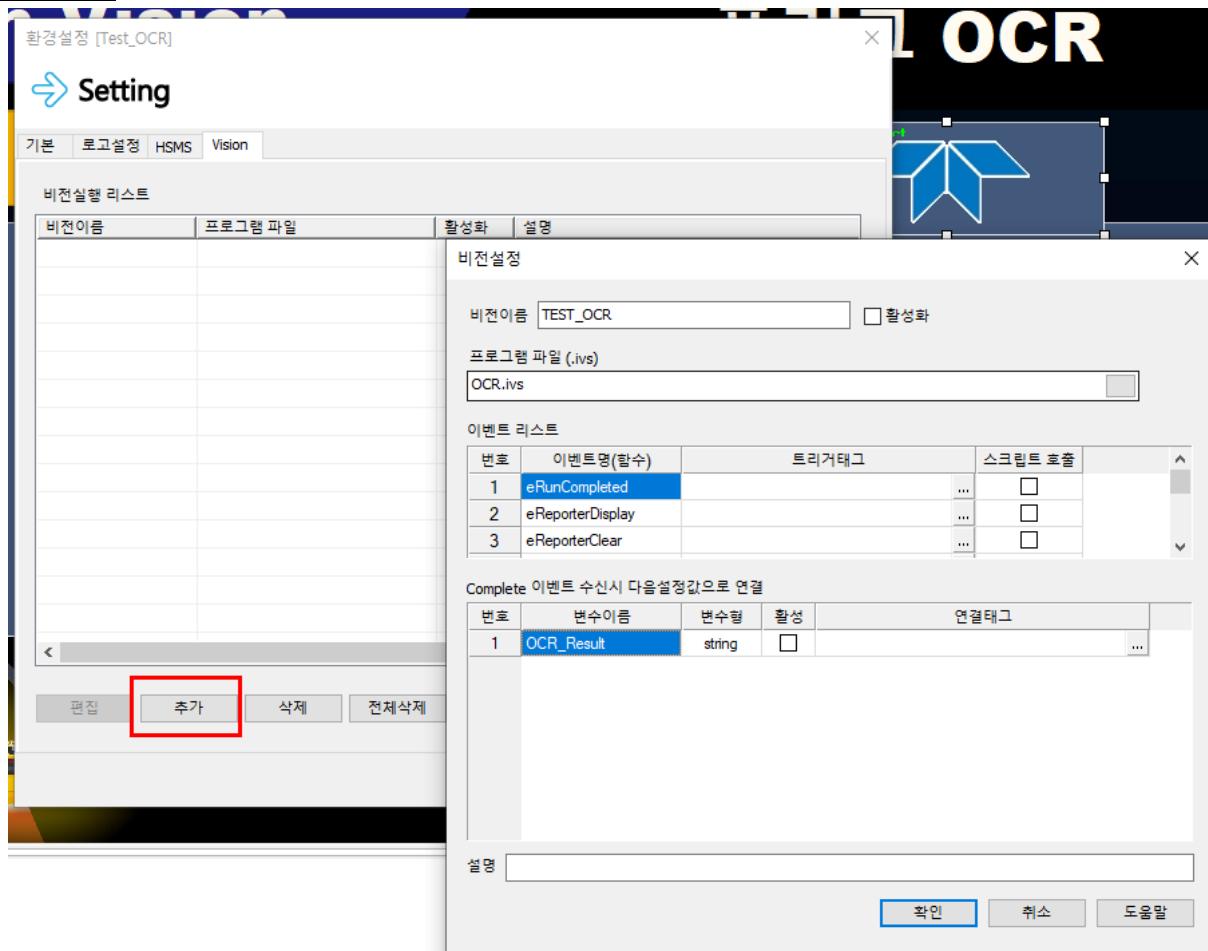
비전실행 리스트

비전이름	프로그램 파일	활성화	설명
OCR	OCR.ivs	비활성	
Inspect	Inspect.ivs	비활성	
2DMatrix	2DMatrix.ivs	비활성	
Measure	Measure.ivs	비활성	

편집 추가 삭제 전체삭제 템플릿 생성 도움말

적용 취소 도움말

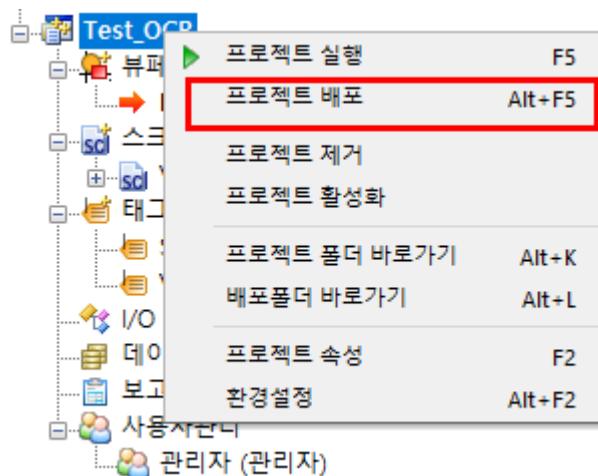
VISION 탭을 선택한 후 앞서 작성한 ivs파일을 로딩합니다.



이때 ivs파일의 변수들이 자동으로 등록되며, eRun에서 사용할 태그와 연결합니다.

6. 최종 파일 생성

아래의 그림과 같이 배포파일을 만들어 최종 파일을 생성합니다.





제 4 장 프로젝트 만들기

기타 자세한 사항은 예제 프로젝트를 참고하시기 바랍니다.

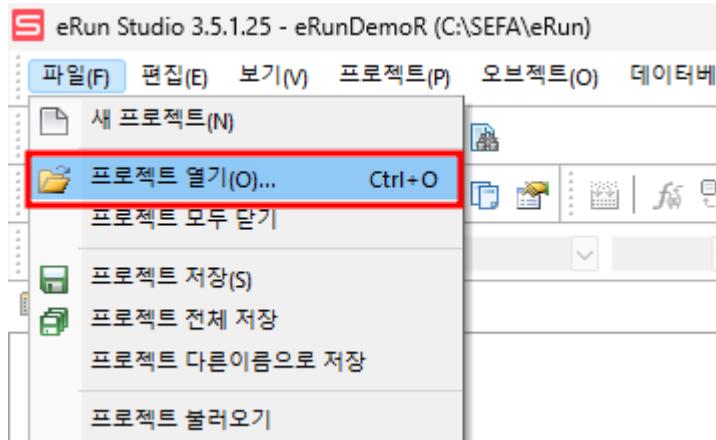
4.8 프로젝트 열기 및 저장

기존에 만들었던 프로젝트를 관리자에 추가하고 프로젝트를 편집합니다.

프로젝트를 복사하거나 배포시에 저장 폴더에 주의하시기 바랍니다. (C:\SEFA\WeRun\Project)

4.8.1 프로젝트 열기

상단의 주메뉴에서 [파일]-[프로젝트 열기] 선택해서 프로젝트 선택창을 불러옵니다.



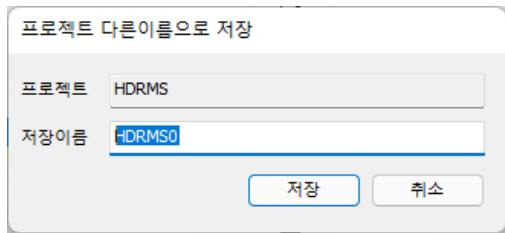
정상적인 eRun SCADA에서 만들어진 프로젝트만 다음과 같이 나타납니다.

프로젝트를 선택하고 하단의 [열기] 버튼을 클릭하거나, 마우스 더블클릭을 하면 프로젝트를 불러옵니다.



4.8.2 프로젝트 다른 이름으로 저장

현재 활성화 된 프로젝트의 이름을 바꾸어서 복사할 경우 사용합니다.

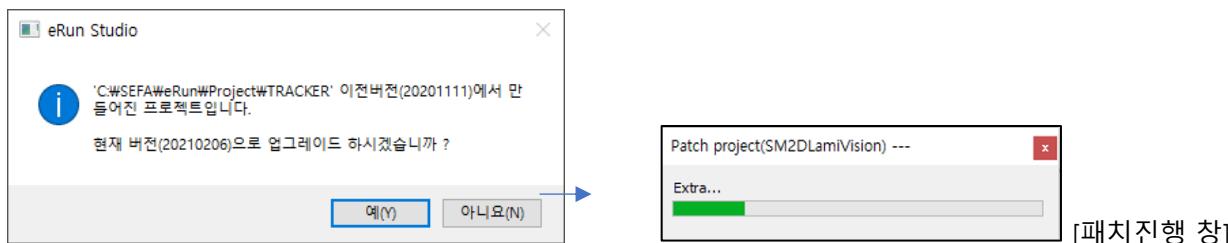


프로젝트 이름을 다른 방법(탐색기의 이름 바꾸기, 폴더이름 변경 등)으로 바꾸게 되면 정상적으로 프로젝트를 실행할 수 없게 됩니다.

또한 기존에 작업했던 프로젝트를 다른 프로젝트에 적용할 경우에 사용하는 기능입니다.

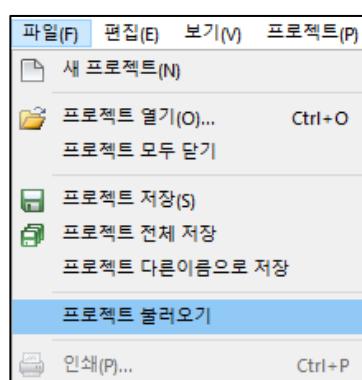
4.8.3. 프로젝트 버전변환

스튜디오 업데이트 이후 이전버전에서 제작된 프로젝트를 불러오려고 할 때 버전체크를 하고 업그레이드를 진행합니다. 버전 변환 작업중 오류가 발생하면 변환작업이 중단됩니다. 이럴 경우 당사에 문의주시기 바랍니다.



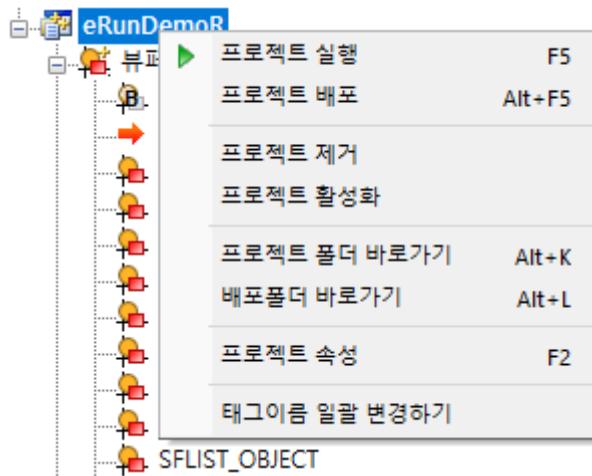
4.8.4 프로젝트 불러오기

프로젝트 열기는 eRun 스튜디오에서 고정된 프로젝트 폴더의 리스트만 보여지기 때문에 다른 폴더의 프로젝트를 열수가 없습니다. [프로젝트 불러오기]는 프로젝트를 인터넷으로 다운로드 받은 경우 그 프로젝트를 스튜디오 폴더로 복사하고 엽니다. 만일 스튜디오 프로젝트 폴더에 같은 이름의 프로젝트가 있는 경우 덮어쓸지 물어보고 진행합니다.



4.9 프로젝트 관리메뉴

스튜디오 왼쪽 프로젝트 관리자 영역에서 프로젝트명을 선택하고 마우스 우클릭을 하면 프로젝트 관리팝업메뉴가 나타납니다.



4.9.1 프로젝트 실행 (F5)

선택한 프로젝트를 실행하는 메뉴입니다.

활성화된 프로젝트가 아닌 경우 활성화 프로젝트로 변경되고 실행을 시작합니다.

보통은 활성화 되어있는 프로젝트에 대해서 키보드 F5를 눌러서 빠르게 실행할 수 있습니다.

프로젝트 실행 프로세스는 간단하게 다음과 같습니다.

- ① 프로젝트 편집내용을 저장합니다.
- ② 스크립트 빌드를 하고 오류 검증을 합니다.
- ③ 사용자 화면 오브젝트 오류 검증을 합니다.
- ④ 배포파일을 생성합니다.
- ⑤ I/O서버 서비스 등록을 합니다. (이미 등록 되어있으면 넘어감)
- ⑥ D/B서버 서비스 등록을 합니다. (이미 등록 되어있으면 넘어감)
- ⑦ I/O서버 실행.
- ⑧ D/B서버 실행.
- ⑨ I/O서버 사용자접속.
- ⑩ 사용자 화면 나타냅니다.
- ⑪ 운영.

4.9.2 프로젝트 배포

프로젝트의 소스를 보호하기 위해서 프로젝트 패키지 파일을 배포파일로 만듭니다. 여기서 생성된 배포파일을 최종실행 PC로 복사해서 UI실행합니다.

배포파일 생성과정은 위에서도 설명했듯이 프로젝트 실행과정과 동일합니다.

다만 배포파일 생성만 하고 프로젝트 실행은 하지 않습니다.

다음 그림은 스튜디오 하단에 스크립트 빌드창에 나타나는 배포생성 과정 창입니다.

```
Compiling...DATABASE.scl
Compiling...REPORT_AUTO.scl

Linking...
링크오류가 0개 있습니다...
스크립트 코드가 정상적으로 생성되었습니다. !!!

Verifying...
스크립트 코드의 유효성 검사가 정상적으로 완료되었습니다. !!!

===== 프로젝트 검증이 완료되었습니다 =====

배포파일 생성중 ...
배포파일 생성완료 되었습니다. [C:\SEFA\WeRun\Publish\HDRMS.eRunPkg]

[ 스크립트 빌드 / 문자열 찾기 / DB메시지 / DB처리결과 ]
```

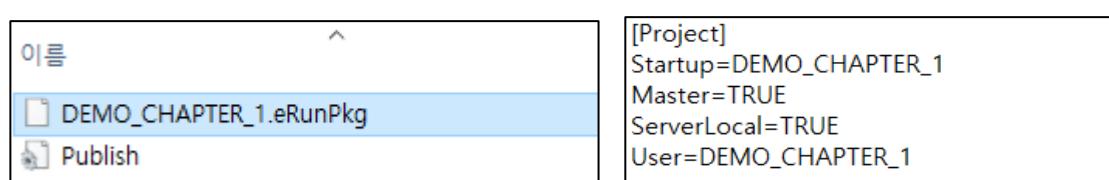
정상적으로 배포파일 생성이 되면 아래와 같이 팝업창이 나타납니다.



배포파일이 저장되는 폴더는 팝업메뉴의 [배포 폴더 바로가기] 선택을 하면 폴더가 열립니다.

C:\SEFA\WeRun\Publish 폴더에 아래와 같이 2개의 파일이 배포파일로 생성됩니다.

오른쪽은 Publish.ini 파일내용입니다.

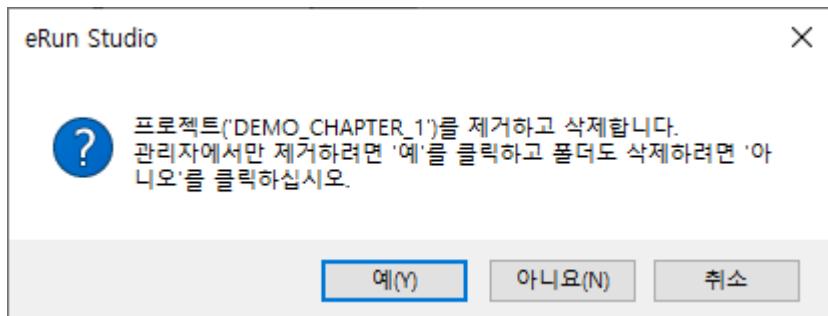


설치 PC에 동일한 폴더에 복사해주고 eRun.exe 실행을 시키면 됩니다.

4.9.3 프로젝트 제거

해당 프로젝트를 프로젝트 관리창에서 제거하고 필요하면 폴더까지 삭제합니다.

아래 메시지 창에서 '예'를 누르면 프로젝트 관리자 항목에서만 제거되고 폴더에 파일은 그대로 있습니다. '아니오'를 누르면 폴더내용가지 모두 삭제됩니다. 주의하시기 바랍니다.



4.9.4 프로젝트 활성화

해당 프로젝트를 활성화 합니다.

프로젝트 관리에서 여러 개의 프로젝트가 있는 경우 해당 프로젝트를 활성화하며, 메뉴 도구바에서 프로젝트와 시작페이지를 변경합니다.



4.9.5 프로젝트 폴더 바로가기

해당 프로젝트 폴더를 탐색기로 오픈합니다.

4.9.6 배포폴더 바로가기

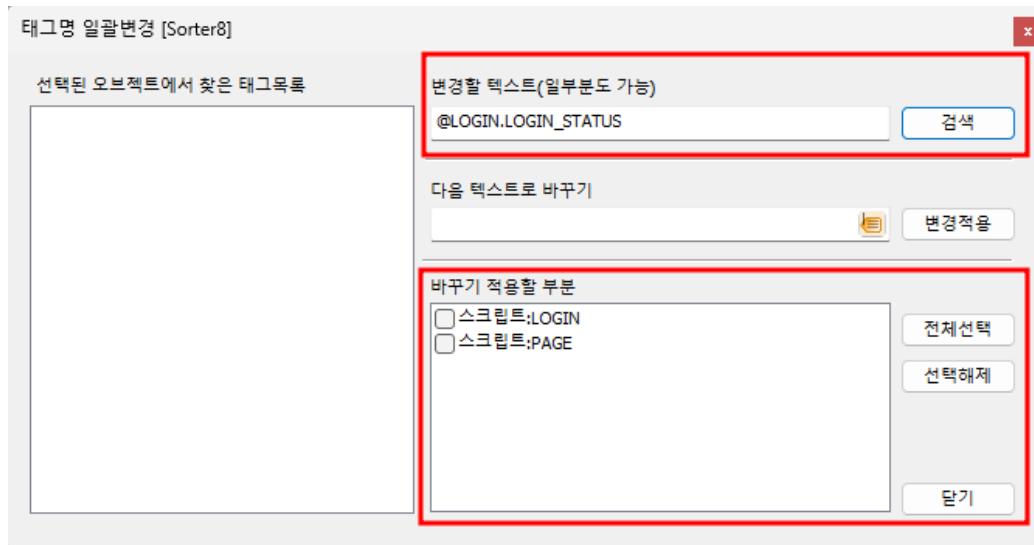
배포파일이 있는 파일탐색기를 오픈합니다.

4.9.7 프로젝트 속성

속성을 보여줍니다. (프로젝트 생성부분 참고바랍니다)

4.9.8 태그이름 일괄 변경하기

프로젝트내에 등록된 태그이름을 일괄적으로 바꾸거나, 선택한 오브젝트에 사용된 태그이름을 바꾸는 기능입니다. 변경할 텍스트를 포함하고 있는 매니저를 보여주고 선택한 매니저에만 적용해서 태그이름을 바꿉니다



- ① 변경할 텍스트는 완전한 태그이름을 넣거나, 일부의 문자열을 넣어줍니다.
- ② 다음텍스트로 바꾸기에서 변경할 텍스트를 입력합니다.
- ③ 바꾸기 적용할 부분은 변경할 텍스트가 있는 매니저가 나타나고 선택해주면 선택된 영역만 변경됩니다.

제 5 장 뷰 페이지 만들기

eRun SCADA 프로젝트를 구성할 때 사용자 뷰 페이지를 통하여 현장 장비의 상태를 감시하거나 제어권한을 부여해서 시스템을 운영하도록 합니다. 프로젝트를 생성하고 뷰 페이지를 추가해서 그 페이지에 다양한 그리기 오브젝트를 그려서 속성을 설정하고 태그명을 연결해서 실시간 태그 값의 변화에 따라 직관적인 화면감시를 할 수 있게 하기위한 사용자 화면을 **뷰 페이지**라고 합니다.

- 베이스뷰 기능으로 편리하고 빠르게 다양한 사용자 페이지 편집가능
- 라인, 박스, 원, 이미지, 텍스트등의 그리기 오브젝트로 직관적으로 감시화면 편집
- 특수목적 표현을 위한 윈도우 오브젝트
- 오브젝트 상,하,좌,우 맞춤 정렬
- 그룹화 및 오브젝트 표시 순서 정렬
- 오브젝트 선택위주의 속성설정으로 간결하게 설정가능
- 런타임시 뷰 페이지 화면크기를 자유자재로 변경
- 움직이는 이미지 오브젝트 지원
- 윈도우에서 지원하는 이미지 오브젝트 모두 지원
- 오브젝트의 다양한 표시제어 속성
- 스크립트 함수로 윈도우 오브젝트 연동으로 유연성 제공

5.1 뷰 페이지 구성

뷰 페이지는 라인, 박스, 원, 이미지, 텍스트, 애니메이션등의 그리기 오브젝트와 그래프, 차트, 그리드, 버튼, 리스트, 콤보, 비전등의 윈도우 오브젝트로 구성이 됩니다. eRun SCADA에서는 뷰 페이지를 구성하는 표시객체를 오브젝트라고 표현합니다. 다음은 그리기 오브젝트와 윈도우 오브젝트의 차이점입니다.

비교항목	그리기 오브젝트	윈도우 오브젝트
그리기 방식	동일	동일
속성설정	속성페이지가 같음	오브젝트마다 속성페이지 다름
제어방식	태그 값	태그 값, 스크립트 함수
표시형태	다중 속성설정이 가능하므로 표현을 여러가지로 할 수 있음	오브젝트별 표시방식이 다름

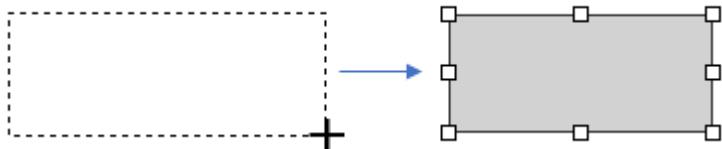
뷰 페이지 편집화면에 선택을 하면 그리기 오브젝트, 윈도우 오브젝트, 정렬, 글씨체 이렇게 4개의 도구바가 활성화 되며 오브젝트 아이콘을 선택하고 뷰 페이지에 그려 넣으면 됩니다. 다음은 그려 넣는 편집방법에 대해서 설명하겠습니다.

5.1.1 오브젝트 그려 넣기

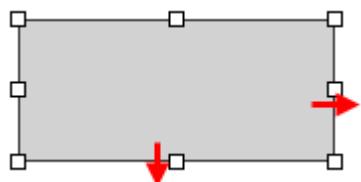
그리기 오브젝트 도구바, 윈도우 오브젝트 도구바에서 선택을 하거나, 스튜디오 상단 메뉴의 [오브젝트] 메뉴에

서 아이콘을 선택하면 마우스 커서가 +로 변경되고 그려주려는 위치에 마우스 왼쪽 버튼을 클릭하고 이동하면서 적당한 크기로 왼쪽버튼을 놓아주면 선택한 오브젝트의 형태가 그려집니다.
마우스 버튼을 놓기 전에 키보드의 ESC 버튼을 누르면 그리기가 취소됩니다.

[사각형 오브젝트 선택후 그려넣기]



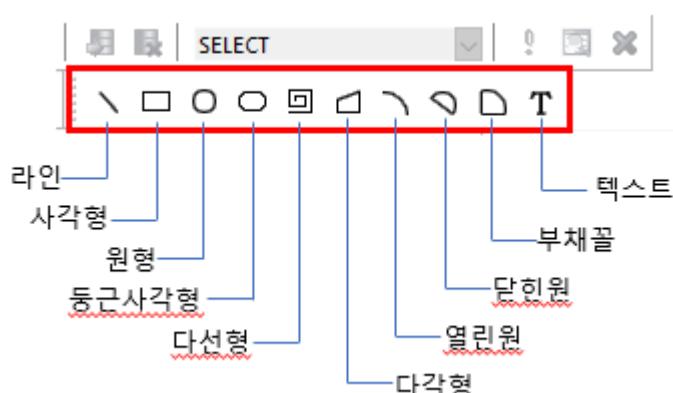
[Shift + 방향키 눌러서 아래로 또는 오른쪽으로 크기 키우기]



[오브젝트 복사]

한 개 이상의 오브젝트 선택 후 CTRL+C 누르고 마우스 커서이동 후 적당한 붙여넣기 위치에서 CTR+V 를 누르면 마우스 커서 위치에 복사됩니다. 또는 CTR+V 대신에 마우스 왼쪽버튼을 클릭하면 그 자리에 복사됩니다.

5.1.2 그리기 오브젝트 도구바



	항목	설명
	라인	라인 / 파이프 오브젝트
	사각형	사각형 오브젝트

○	원형	원형 오브젝트
○	둥근사각형	모서리가 둥근 사각형 오브젝트
回	다선형	다선형 오브젝트
□	다각형	다각형 오브젝트
~	열린원	열린 원형 오브젝트
○	닫힌원	닫힌 원형 오브젝트
□	부채꼴	부채꼴 오브젝트
T	텍스트	고정 문자열 또는 태그값 표시용

그리기 오브젝트에서 다선형과 다각형 오브젝트를 제외하고는 편집 점이 8개로 동일합니다. 그래서 다선형과 다각형 편집방법에 대해서 설명하겠습니다.

가. 다선형 / 다각형 생성

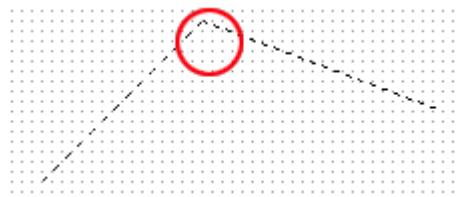
다선형은 시작점과 끝점 연결이 되지 않습니다. 주로 배관 파이프라인을 표현할 때 사용합니다. 다각형은 시작점과 끝점이 자동 연결되며 닫힌 도형이 됩니다. 주로 오일탱크등의 수위표시용으로 사용합니다.
다선형과 다각형은 일반 오브젝트처럼 마우스로 클릭 후 마우스를 드래그해서 영역을 설정해서 그리는 것이 아닌 변곡점을 마우스로 선택(선이 깨이는 부분)해서 다선형, 다각형 오브젝트를 그립니다.

시작점(원)을 마우스 왼쪽버튼을 클릭하면 아래 그림과 같이 다음 변곡점을 선택하기 전까지 직선으로 실선이 그려집니다.



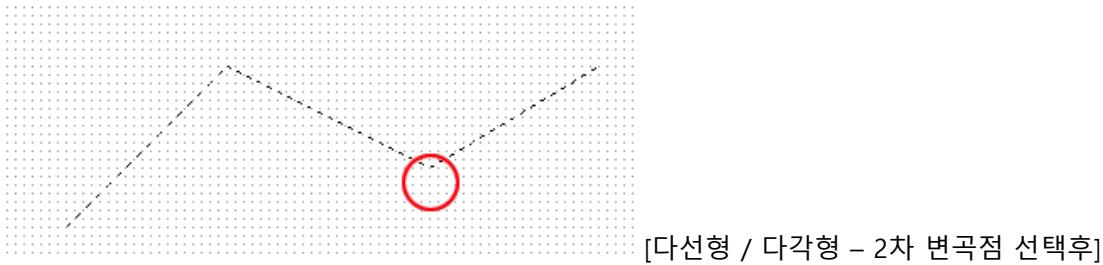
[다선형 / 다각형 – 시작점 선택 후]

첫번째 변곡점 (원)을 마우스 왼쪽버튼을 클릭하면 아래 그림과 같이 클릭한 지점부터 다음 변곡점을 선택하기 전까지는 직선으로 실선이 그려집니다.

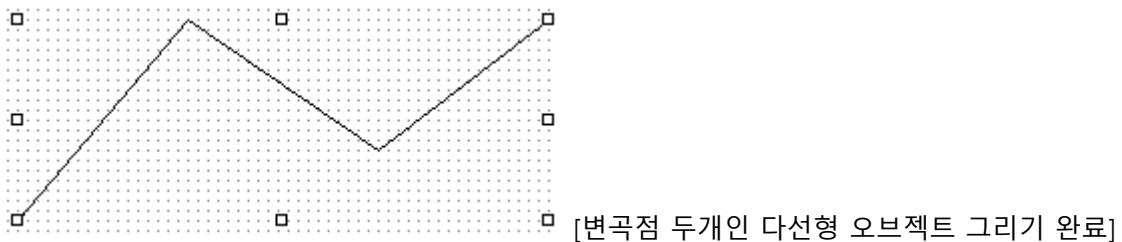


[다선형 / 다각형 – 1차 변곡점 선택 후]

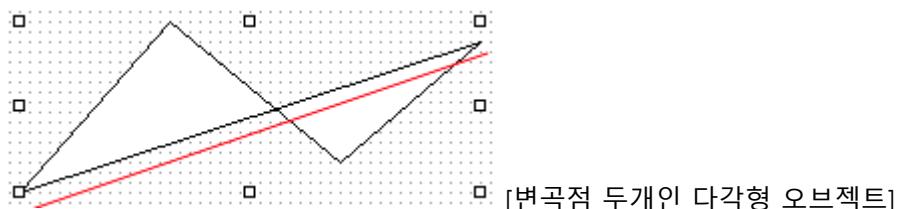
두번째 변곡점 (원)을 마우스 왼쪽 버튼을 클릭하면 아래 그림과 같이 클릭한 지점부터 다음 변곡점을 선택하기 전까지는 직선으로 실선이 그려집니다.



변곡점을 추가 후 더 이상 변곡점을 추가하지 않고 그리기를 종료하기 위해서는 마지막 원하는 위치에서 마우스 왼쪽 버튼을 클릭하면 아래 그림과 같이 오브젝트가 그려집니다.



다각형의 경우 마지막 위치에서 마우스 왼쪽버튼을 더블클릭을 하면 아래 그림과 같이 마지막 위치와 처음 위치(선)를 선으로 연결을 시킵니다.



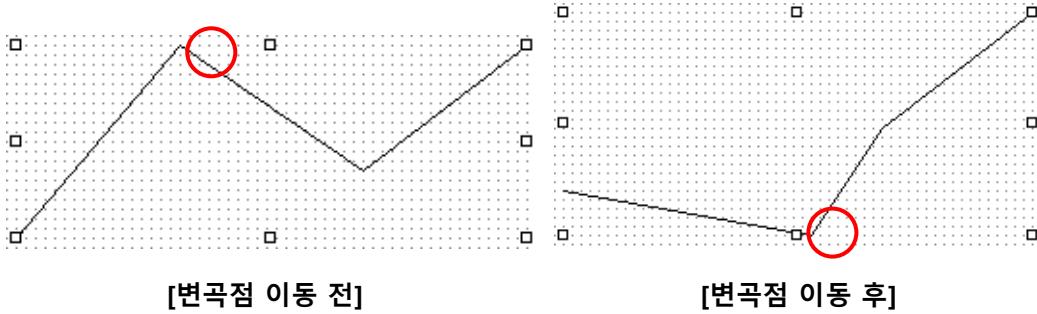
이처럼 다선형과 다각형은 아래 사용자가 원하는 위치를 선택해서 다각형과 다선형 그리기를 할 수 있습니다.

나. 다선형 / 다각형 점편집

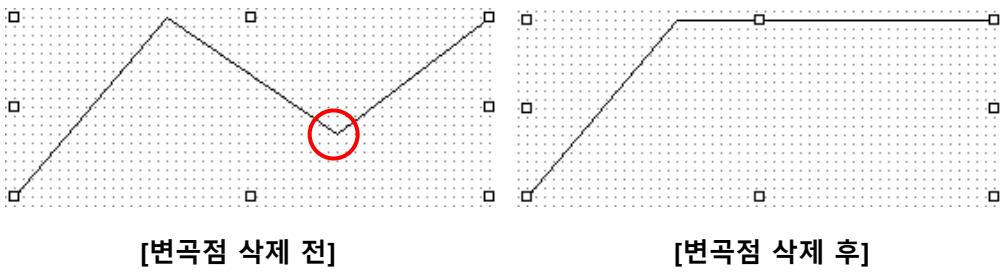
다각형 또는 다선형 오브젝트에 대해서 점편집 (변곡점)을 할 수 있습니다.

점편집이라 하는 것은 아래 그림 예와 같이 변곡점을 선택하여 위치를 이동시키거나 변곡점을 삭제 또는 변곡점을 추가하여 오브젝트의 모형을 변경하는 것을 말합니다.

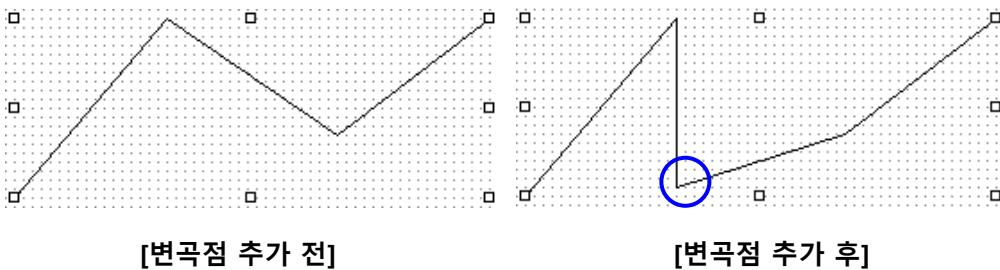
[변곡점 이동]



[변곡점 삭제]



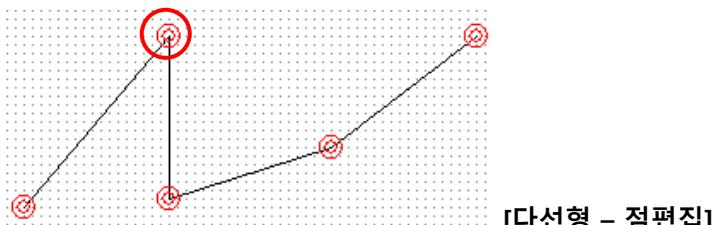
[변곡점 추가]



다. 점편집 변곡점 삭제

마우스로 다선형 또는 다각형 오브젝트를 선택합니다.

다선형 또는 다각형 오브젝트를 선택 후 메뉴에서 **점편집**을 선택하면 아래 그림과 같이 변곡점에 빨간색 원이 그려져서 편집할 수 있게 화면에 표시됩니다.

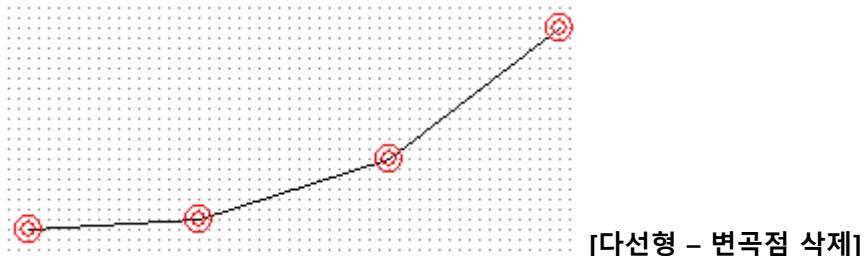


특정 변곡점을 삭제하기 위해서는 키보드의 왼쪽 CTRL 키를 누른 상태에서 삭제하고자 하는 변곡점에 마우스가 위치하게 되면 마우스 커서가 모양으로 변경됩니다.

마우스 커서가 변경된 상태에서 마우스 왼쪽버튼을 클릭하면 아래 그림과 같이 특정 변곡점이 삭제되고 떨어져

있는 두 변곡점을 연결합니다.

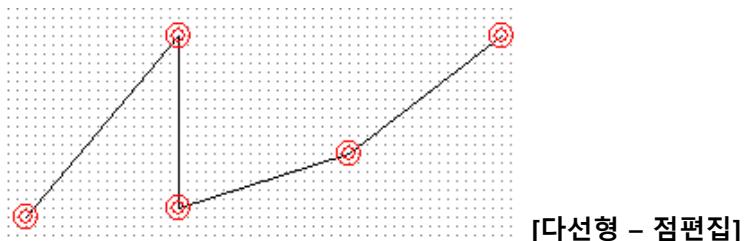
여기서는 위의 그림의 원에 있는 변곡점을 삭제한 것에 대한 결과가 아래 그림과 같습니다.



라. 점편집 변곡점 이동

마우스로 다선형 또는 다각형 오브젝트를 선택합니다.

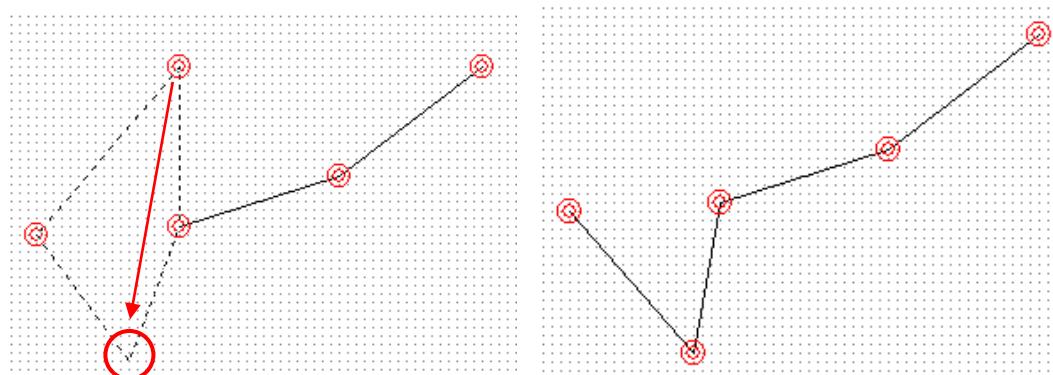
다선형 또는 다각형 오브젝트를 선택하고 뷰페이지 팝업메뉴에서 “점편집”을 선택하면 아래 그림과 같이 변곡점에 빨간색 원이 그려져서 편집할 수 있게 화면에 표시됩니다.



변곡점을 이동하기 위해서는 이동을 하고자 하는 변곡점에 마우스가 위치하게 되면 마우스 커서가 모양으로 변경됩니다.

커서가 변경 된 상태에서 마우스의 왼쪽버튼을 누르면서 드래그를 하면 다음 그림과 같이 표시됩니다. 드래그가 되는 동안에는 이동되는 변곡점에 대해서는 실선으로 표시 됩니다. 원하는 위치에서 드래그를 멈추고 마우스 왼쪽버튼을 놓으면 다음 그림과 같이 마우스의 현재 위치로 변곡점을 이동시켜서 오브젝트를 다시 그립니다.

[점편집 – 변곡점 이동]

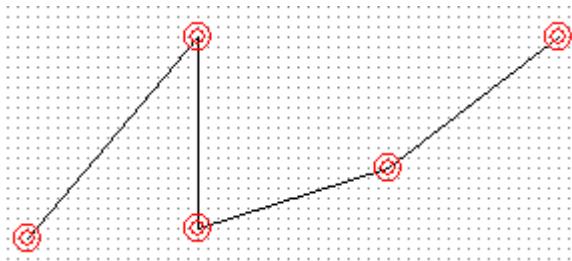


변곡점 이동을 해도 점편집 상태는 해제되지 않습니다.

마. 점편집 변곡점 추가

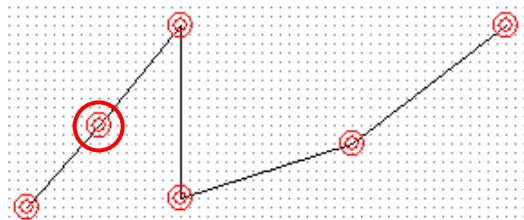
마우스로 다선형 또는 다각형 오브젝트를 선택합니다.

다선형 또는 다각형 오브젝트를 선택하고 난 후 마우스의 메뉴에서 **점편집**을 선택하면 아래 그림과 같이 변곡점에 빨간색 원이 그려져서 편집할 수 있게 화면에 표시됩니다.



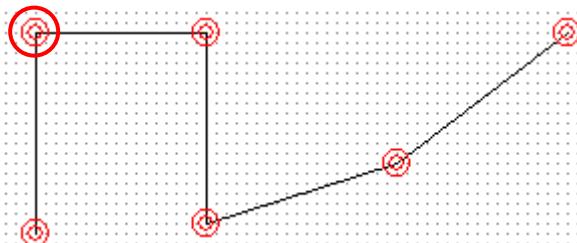
[다선형 – 점편집]

변곡점 추가는 현재 선택한 다선형/다각형 오브젝트의 영역이 아닌 선에서만 추가할 수 있습니다. 변곡점을 추가하기 위해서 마우스를 오브젝트의 선위에 위치하게 되면 마우스 커서가  모양으로 변경됩니다. 마우스 모양이 변경된 상태에서 마우스의 왼쪽버튼을 클릭하면 아래 그림과 같이 변곡점이 추가됩니다.



[다선형 – 점편집 – 변곡점 추가]

추가된 변곡점의 위치를 아래 그림과 같이 원하는 위치로 이동시켜서 원하는 오브젝트 형태로 만듭니다.



[다선형 – 점편집 – 변곡점 추가 후 이동]

점편집을 종료하려면 키보드의 ESC를 누르거나 마우스로 뷰페이지의 빈 영역을 클릭하면 점편집이 종료됩니다.

바. 텍스트 오브젝트

문자열 오브젝트는 특정 문자를 고정으로 표기를 하거나 설정된 태그값 또는 표현식 결과값을 표시할 때 사용하

는 오브젝트입니다.

문자열 오브젝트를 추가하기 위해서는 뷰 도구 툴바에서 **T** 버튼을 클릭하거나 [메뉴]–[오브젝트]–[그리기]–[문자열 입력]을 선택하면 마우스 커서가 +로 변경됩니다.

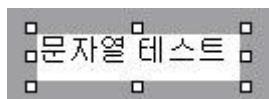
키보드의 ESC 버튼을 누르면 그리기가 취소됩니다.

마우스로 문자열 입력을 원하는 위치에서 클릭한 상태에서 드래그하여 영역을 설정한 후 클릭을 해제하면 아래 그림과 같이 문자열 입력창이 표시가 되고 기본으로 "TEXT..." 문구가 표시됩니다.

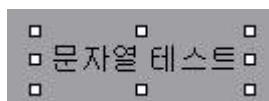


문자 입력란에 작성할 문자를 아래 그림과 같이 작성합니다.

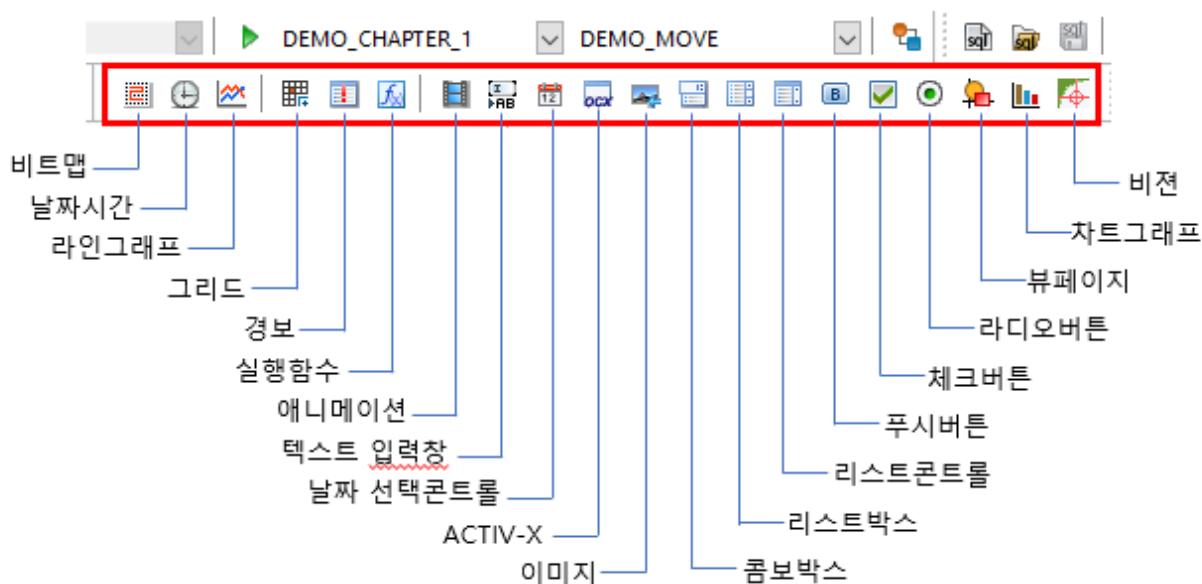
문자 입력란에 작성하는 동안 문자열에 대한 정렬은 적용되지 않습니다.



문자 입력을 하고 엔터키를 치면 아래 그림과 같이 기본 문자열이 입력됩니다.



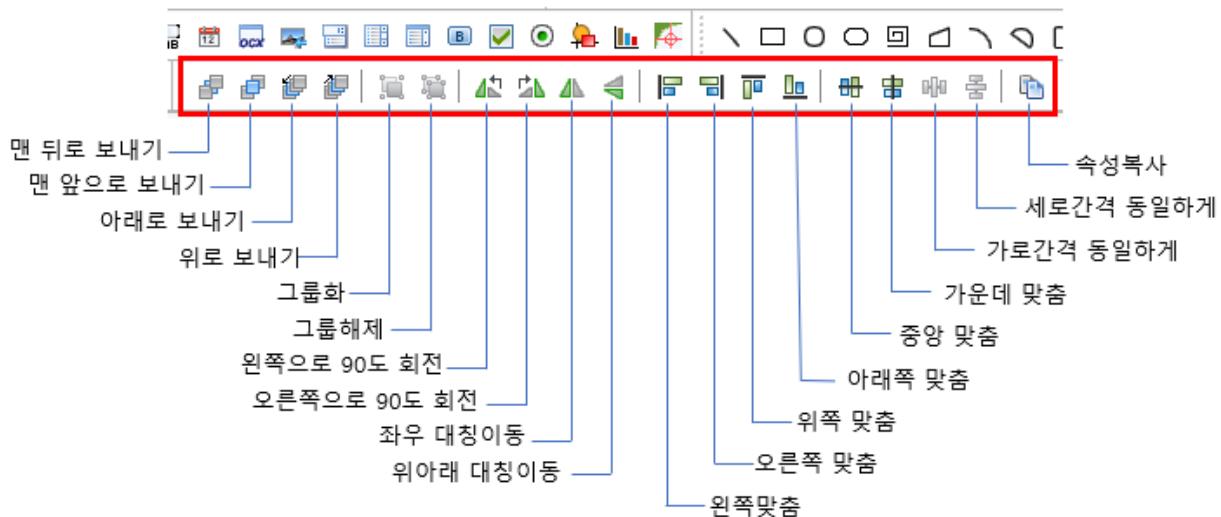
5.1.3 원도우 오브젝트 도구바



아이콘	항목	설명
	비트맵	간단한 비트맵을 생성하여 뷰페이지에 추가할때 사용
	시계	현재 시스템에 시간을 원하는 형태로 표시
	라인그래프	실시간 / 히스토리 그래프를 표시
	그리드	사용자가 시트에 다양한 값을 입력하여 처리하는 경우 사용 레시피 편집시 주로 사용
	경보	현재 발생한 경보리스트 표시 또는 경보이력 검색 표시용
	애니메이션	움직이는 이미지 표시
	날짜콘트롤	날짜 / 시간을 다양한 형식의 문자열로 표시
	ACTIVE-X	윈도우 운영체제에서 제공되는 OCX 3rd 라이브러리
	이미지	PNG/GIF/BMP/JPG 이미지
	콤보박스	여러 항목을 선택할 수 있고, 입력도 가능한 DROP DOWN 오브젝트
	리스트박스	여러 항목을 선택할 수 있도록 하는 오브젝트.
	리스트콘트롤	그리드 오브젝트와 비슷하지만, 입력기능 없음.주로 DATABASE 연계표시용으로 사용
	푸시버튼	누름단추
	체크박스	텍스트가 포함된 ON / OFF 선택용 오브젝트
	라디오버튼	여러 개의 버튼중 취사선택이 가능하도록 하는 오브젝트
	뷰페이지	다른 뷰페이지를 서브페이지 오브젝트로 삽입할 때 사용
	차트	BAR 그래프 / 원형차트 / 메터표시
	비전	eRun Vision 프로젝트 제작시 사용. Sherlock / Euresys 머신비전 연계
	CCTV	IP Camera 오브젝트

5.1.4 오브젝트 정렬 도구바

선택한 오브젝트의 표시순서, 그룹 설정 및 해제, 회전 및 대칭이동, 가로정렬, 세로정렬을 할 때 사용하는 도구바입니다

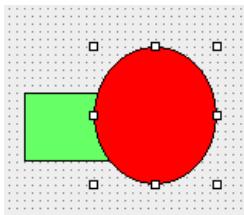


가. 맨뒤로 보내기

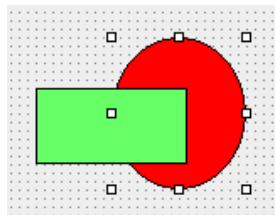
현재 선택한 오브젝트를 맨 뒤로 보냅니다.

아래 좌측 그림과 같이 우선 순서를 조절할 오브젝트를 선택 후 “맨뒤로 보내기 버튼”을 클릭하면 아래 우측 그림과 같이 현재 선택한 오브젝트를 맨 뒤로 보냅니다.

※ 오브젝트가 1개이상 선택되어야 툴바 버튼이 활성화됩니다.



[순서 조절전]



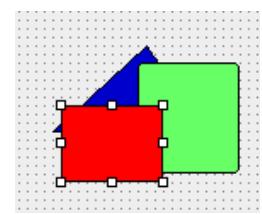
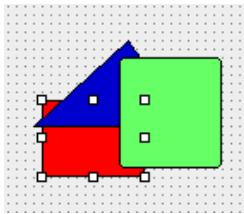
[순서 조절후]

나. 맨앞으로 보내기

현재 선택한 오브젝트를 맨 앞으로 보냅니다.

아래 좌측 그림과 같이 우선 순서를 조절할 오브젝트를 선택 후 “맨앞으로 보내기 버튼”을 클릭하면 아래 우측 그림과 같이 현재 선택한 오브젝트를 맨 앞으로 보냅니다.

※ 오브젝트가 1개이상 선택되어야 툴바 버튼이 활성화됩니다.



[순서 조절전]

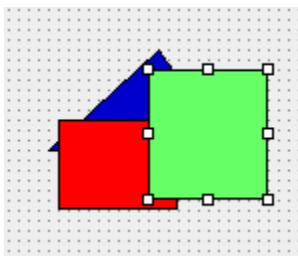
[순서 조절후]

다.  뒤로

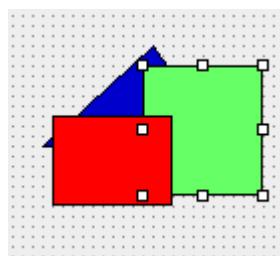
현재 선택한 오브젝트를 하나 뒤로 보냅니다.

아래 좌측 그림과 같이 우선 순서를 조절할 오브젝트를 선택 후 “뒤로 버튼”을 클릭하면 아래 우측 그림과 같이 선택한 오브젝트를 하나 뒤로 보냅니다.

※ 오브젝트가 1개이상 선택되어야 툴바 버튼이 활성화됩니다.



[순서 조절전]



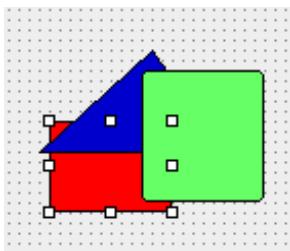
[순서 조절후]

라.  앞으로

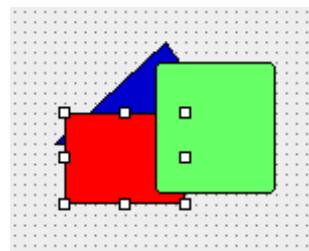
현재 선택한 오브젝트를 하나 앞으로 보냅니다.

아래 좌측 그림과 같이 순서를 조절할 오브젝트를 선택 후 “앞으로 버튼”을 클릭하면 아래 우측 그림과 같이 현재 선택한 오브젝트를 하나 앞으로 보냅니다.

※ 오브젝트가 1개이상 선택되어야 툴바 버튼이 활성화됩니다.



[순서 조절전]



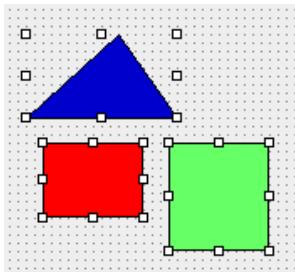
[순서 조절후]

마.  그룹 설정

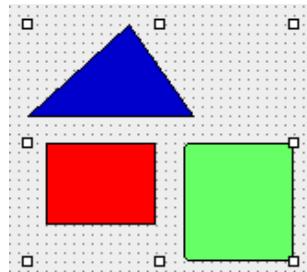
한 개 이상의 선택된 오브젝트를 하나의 그룹으로 설정합니다.

아래 좌측 그림과 같이 오브젝트 세 개를 다중 선택 후 “그룹 설정 버튼”을 클릭하면 아래 우측 그림과 같이 그룹이 설정됩니다.

※ 오브젝트가 1개이상 선택되고, 원도우 오브젝트가 포함되지 않아야 활성화됩니다.



[그룹설정 조절전]

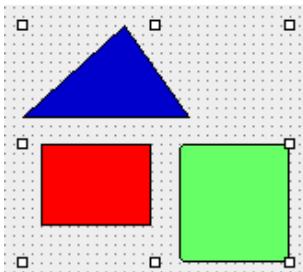


[그룹설정 조절후]

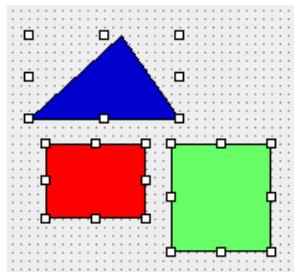
바. 그룹 해제

현재 선택한 그룹을 해제합니다.

아래 좌측 그림과 같이 그룹 오브젝트를 선택 후 “그룹 해제 버튼”을 클릭하면 아래 우측 그림과 같이 그룹이 해제되고 다중 선택이 됩니다.



[그룹 해제 전]



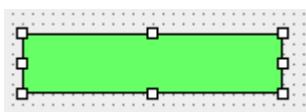
[그룹 해제 후]

사. 좌측으로 회전

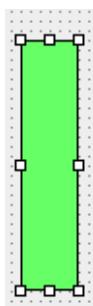
현재 선택한 오브젝트 또는 그룹오브젝트를 좌측으로 90도 회전합니다.

아래 좌측 그림과 같이 회전시킬 오브젝트를 선택 후 “좌측으로 회전 버튼”을 클릭하면 아래 우측 그림과 같이 오브젝트가 회전됩니다.

※ 윈도우 오브젝트는 회전이 안됩니다. 오브젝트를 다중선택시 윈도우 오브젝트가 포함되어 있으면 툴바가 비활성화 됩니다.



[회전 전]



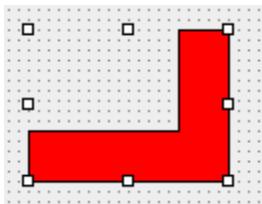
[회전 후]

아. 우측으로 회전

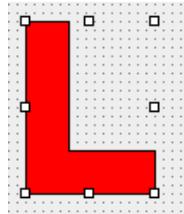
현재 선택한 오브젝트 또는 그룹오브젝트를 우측으로 90도 회전합니다.

아래 좌측 그림과 같이 회전시킬 오브젝트를 선택 후 “**우측으로 회전 버튼**”을 클릭하면 아래 우측 그림과 같이 오브젝트가 회전됩니다.

※ 윈도우 오브젝트는 회전이 안됩니다. 오브젝트를 다중선택시 윈도우 오브젝트가 포함되어 있으면 툴바가 비활성화 됩니다.



[회전 전]



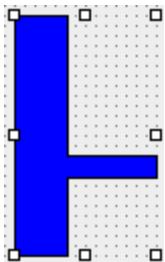
[회전 후]

차. 수평 대칭

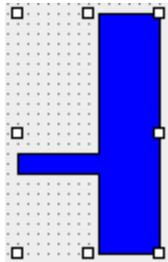
현재 선택한 오브젝트 또는 그룹오브젝트를 수평 대칭합니다.

아래 좌측 그림과 같이 좌우 대칭시킬 오브젝트를 선택 후 “**수평 대칭 버튼**”을 클릭하면 아래 우측 그림과 같이 오브젝트가 대칭됩니다.

※ 윈도우 오브젝트는 수평대칭이 안됩니다. 오브젝트를 다중선택시 윈도우 오브젝트가 포함되어 있으면 툴바가 비활성화 됩니다.



[대칭 전]



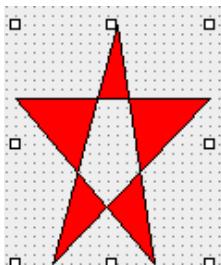
[대칭 후]

차. 수직 대칭

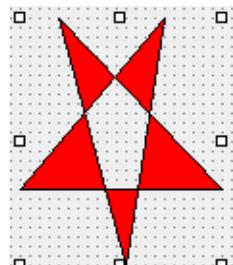
현재 선택한 오브젝트 또는 그룹오브젝트를 수직 대칭합니다.

아래 좌측 그림과 같이 상하 대칭시킬 오브젝트를 선택 후 “**수직 대칭 버튼**”을 클릭하면 아래 우측 그림과 같이 오브젝트가 대칭이 됩니다.

※ 윈도우 오브젝트는 수직대칭이 안됩니다. 오브젝트를 다중선택시 윈도우 오브젝트가 포함되어 있으면 툴바가 비활성화 됩니다.



[대칭 전]



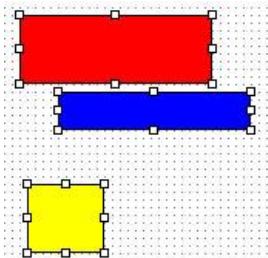
[대칭 후]

카. 왼쪽 정렬

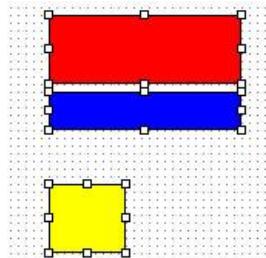
현재 선택한 오브젝트 또는 그룹오브젝트를 왼쪽으로 정렬합니다.

아래 좌측 그림과 같이 정렬시킬 오브젝트를 선택 후 “왼쪽정렬 버튼”을 클릭하면 아래 우측 그림과 같이 왼쪽 정렬이 됩니다.

※ 오브젝트를 한개만 선택되어 있으면 뷰페이지를 기준으로 뷰페이지의 왼쪽에 위치하게 됩니다.



[정렬 전]



[정렬 후]

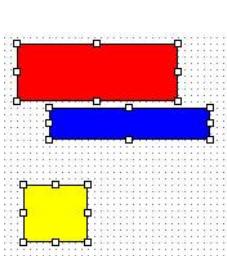
타. 오른쪽 정렬

현재 선택한 오브젝트 또는 그룹오브젝트를 오른쪽으로 정렬합니다.

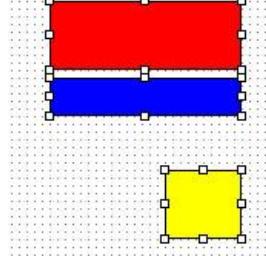
아래 좌측 그림과 같이 정렬시킬 오브젝트를 선택 후 “오른쪽정렬 버튼”을 클릭하면

아래 우측 그림과 같이 오른쪽 정렬이 됩니다.

※ 오브젝트를 한개만 선택되어 있으면 뷰페이지를 기준으로 뷰페이지의 오른쪽에 위치하게 됩니다.



[정렬 전]



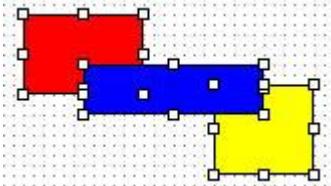
[정렬 후]

파. 위쪽 정렬

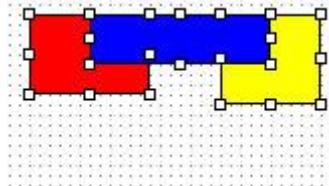
현재 선택한 오브젝트 또는 그룹오브젝트를 위쪽으로 정렬합니다.

아래 좌측 그림과 같이 정렬시킬 오브젝트를 선택 후 “**위쪽정렬 버튼**”을 클릭하면 아래 우측 그림과 같이 위쪽 정렬이 됩니다.

※ 오브젝트를 한개만 선택되어 있으면 뷰페이지를 기준으로 뷰페이지의 가장 위쪽에 위치하게 됩니다.



[정렬 전]



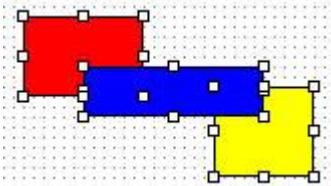
[정렬 후]

하. 아래쪽 정렬

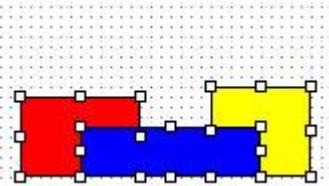
현재 선택한 오브젝트 또는 그룹오브젝트를 아래쪽으로 정렬합니다.

아래 좌측 그림과 같이 정렬시킬 오브젝트를 선택 후 “**아래쪽정렬 버튼**”을 클릭하면 아래 우측 그림과 같이 아래쪽 정렬이 됩니다.

※ 오브젝트를 한개만 선택 되어 있으면 뷰페이지를 기준으로 뷰페이지의 가장 아래쪽에 위치하게 됩니다.



[정렬 전]



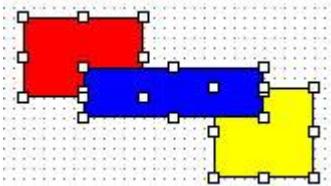
[정렬 후]

가. 수평중앙정렬

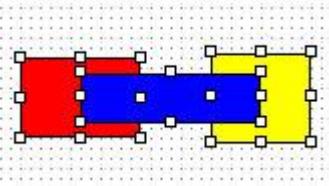
현재 선택한 오브젝트 또는 그룹오브젝트를 수평중앙으로 정렬합니다.

아래 좌측 그림과 같이 정렬시킬 오브젝트를 선택 후 “**수평중앙정렬 버튼**”을 클릭하면 아래 우측 그림과 같이 수평중앙으로 정렬이 됩니다.

※ 오브젝트를 한개만 선택되어 있으면 뷰페이지를 기준으로 뷰페이지의 수평 중앙에 위치하게 됩니다.



[정렬 전]



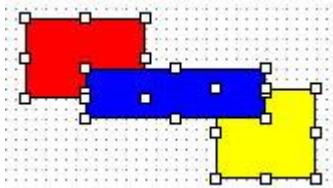
[정렬 후]

나. 수직중앙정렬

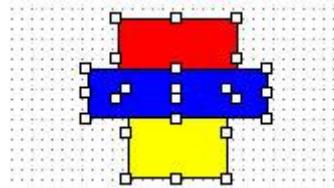
현재 선택한 오브젝트 또는 그룹오브젝트를 수직중앙으로 정렬합니다.

아래 좌측 그림과 같이 정렬시킬 오브젝트를 선택 후 “수직중앙정렬 버튼”을 클릭하면 아래 우측 그림과 같이 수직중앙으로 정렬이 됩니다.

※ 오브젝트를 한개만 선택되어 있으면 뷰페이지를 기준으로 뷰페이지의 수직 중앙에 위치하게 됩니다.



[정렬 전]



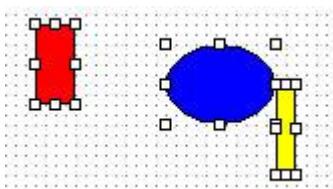
[정렬 후]

다. 수평균일정렬

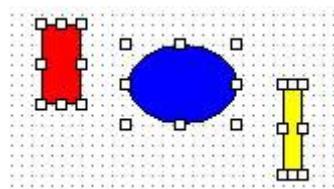
현재 선택한 오브젝트 또는 그룹오브젝트를 수평으로 균일하게 정렬합니다.

아래 좌측 그림과 같이 정렬시킬 오브젝트를 선택 후 “수평균일정렬 버튼”을 클릭하면 아래 우측 그림과 같이 수평으로 균일하게 정렬이 됩니다.

※ 오브젝트를 한개이상 선택시 툴바 버튼이 활성화됩니다.



[정렬 전]



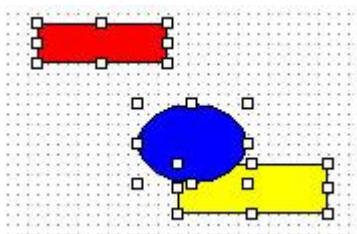
[정렬 후]

라. 수직균일정렬

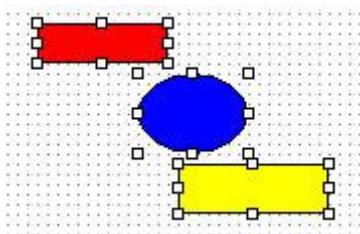
현재 선택한 오브젝트 또는 그룹오브젝트를 수직으로 균일하게 정렬합니다.

아래 좌측 그림과 같이 정렬시킬 오브젝트를 선택 후 “수직균일정렬 버튼”을 클릭하면 아래 우측 그림과 같이 수직으로 균일하게 정렬이 됩니다.

※ 오브젝트를 한개이상 선택시 툴바 버튼이 활성화됩니다.



[정렬 전]



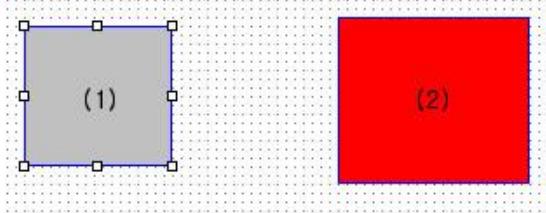
[정렬 후]

마. 속성복사

현재 선택된 오브젝트의 속성을 다른 오브젝트에도 동일하게 적용합니다.

아래 그림과 같이 오브젝트 속성을 복사할 오브젝트를 선택 후 “속성복사 버튼”을 클릭하고 오브젝트 속성을 복사될 오브젝트를 선택하면 복사한 오브젝트의 속성이 복사될 오브젝트에 동일하게 적용이 됩니다.

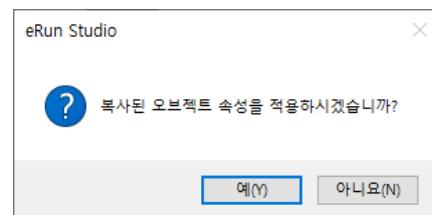
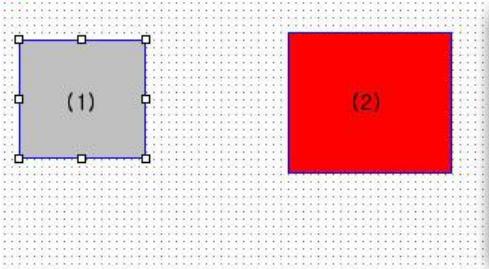
① 복사할 오브젝트 (1) 선택



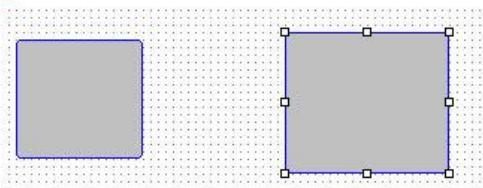
② 정렬툴바에서 속성복사 버튼 선택



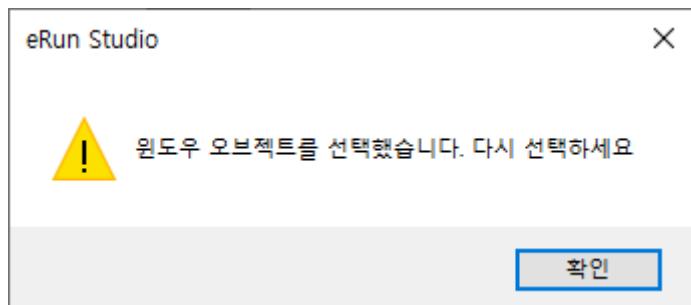
③ 복사될 오브젝트 (2) 선택



④ 오브젝트 속성 복사 완료



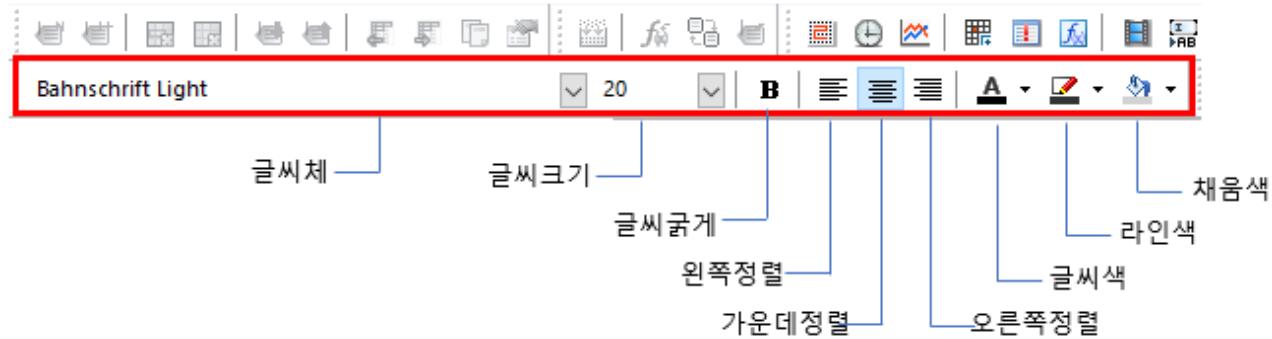
속성복사는 윈도우 오브젝트에는 적용되지 않습니다.



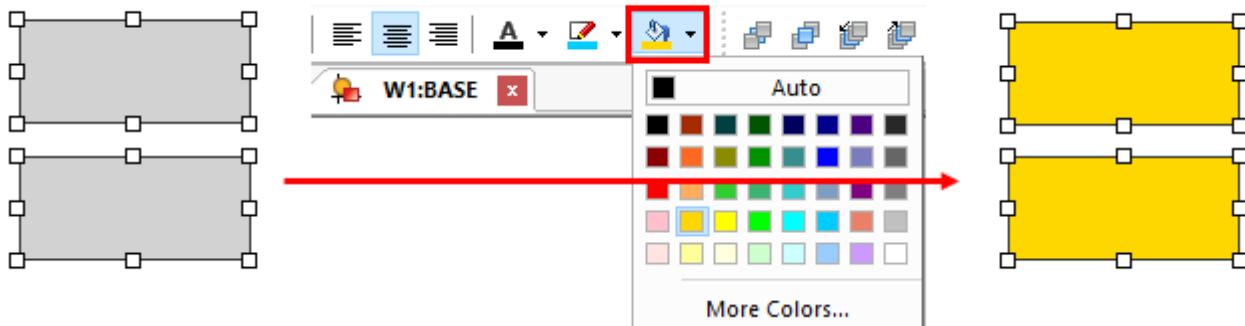
5.1.5 글씨체 도구바

한 개 또는 그 이상의 텍스트 오브젝트에 대해서 글씨체 변경, 크기, 굵기, 좌우정렬, 색깔 지정할 때 사용합니다. 그리고 라인, 사각형, 둥근사각형, 원형 오브젝트를 한 개 이상 선택하고 라인색, 채움색을 일괄적으로 적용할 때

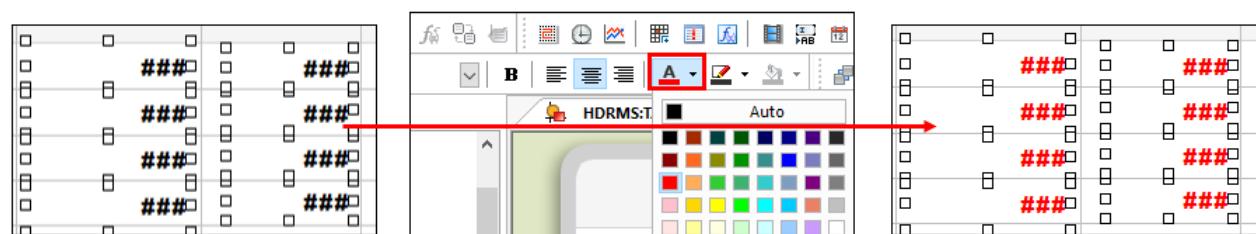
사용합니다.



[선택된 오브젝트 모두 채움색 변경하기]

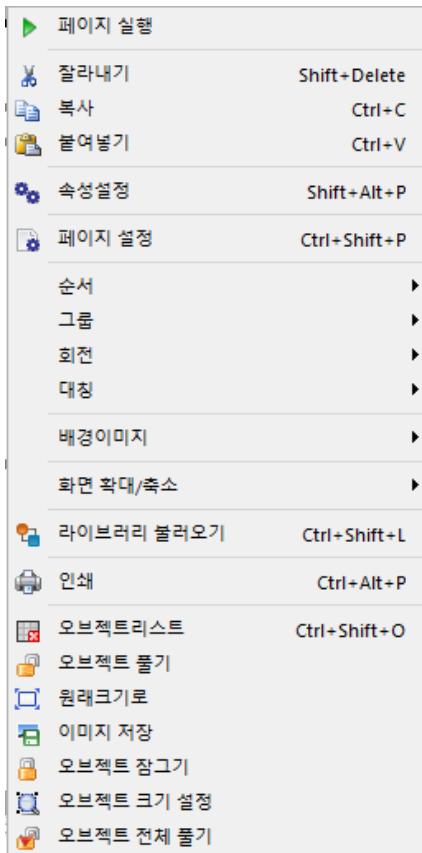


[선택된 오브젝트 모두 글씨체 빨강색으로 변경하기]

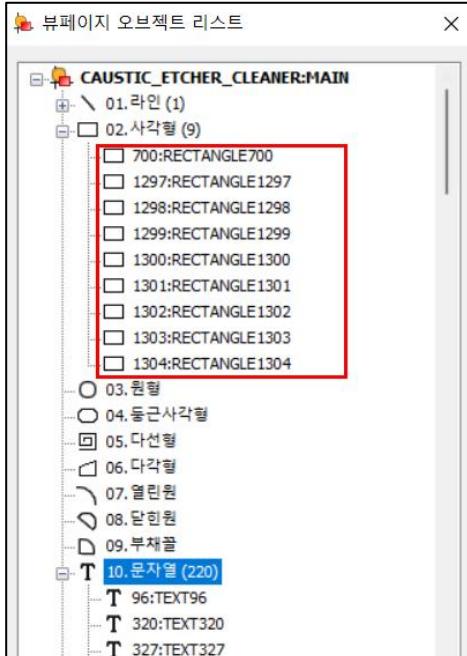


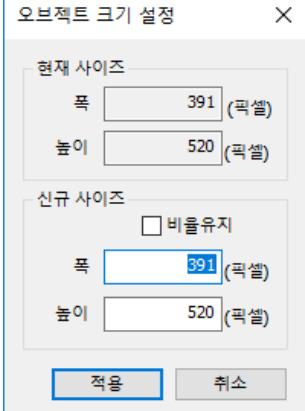
5.1.6 팝업메뉴

뷰 페이지 임의의 위치에서 마우스 오른쪽 버튼 클릭하면 팝업메뉴가 나타납니다.



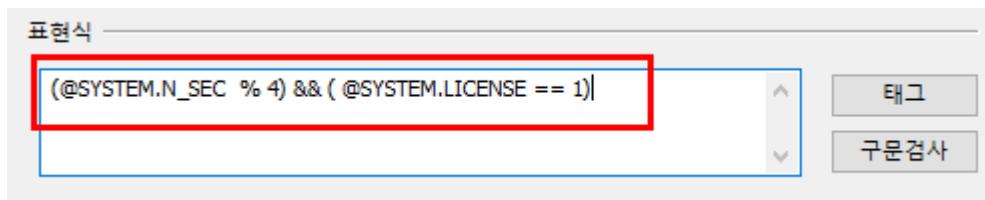
항목	설명
페이지 실행	현재의 뷰페이지를 시작페이지로 자동으로 등록하고, 프로젝트를 실행합니다.
잘라내기	현재 뷰페이지에서 선택한 오브젝트를 삭제하고 붙여넣기 할 수 있게 메뉴를 활성화 합니다.
복사	현재 선택한 오브젝트를 복사합니다.
붙여넣기	현재 위치에 복사/잘라내기한 오브젝트를 현재 마우스 위치에 붙여넣기 합니다.
속성설정	현재 선택한 오브젝트에 대한 속성을 설정합니다. 다중선택이 되면 비활성화 됩니다.
페이지 설정	현재 뷰페이지의 설정을 확인/수정합니다.
순서	정렬 도구바의 [순서]와 동일한 기능입니다. <ul style="list-style-type: none"> ■ 맨뒤로 보내기 : 현재 선택한 오브젝트를 맨뒤로 보냅니다. ■ 뒤로 보내기 : 현재 선택한 오브젝트를 하나 뒤로 보냅니다. ■ 앞으로 보내기 : 현재 선택한 오브젝트를 하나 앞으로 보냅니다. ■ 맨 앞으로 보내기 : 현재 선택한 오브젝트를 맨 앞으로 보냅니다.
그룹	정렬 도구바의 [그룹]과 동일한 기능입니다 <ul style="list-style-type: none"> ■ 그룹설정 : 현재 선택한 오브젝트들을 하나의 그룹으로 설정합니다. ■ 그룹해제 : 현재 선택한 그룹오브젝트를 그룹해제 합니다.
회전	정렬 도구바의 [회전]과 동일한 기능입니다 <ul style="list-style-type: none"> ■ 90 도 시계방향 회전

	<p>현재 선택한 오브젝트를 시계방향으로 90 도 회전합니다.</p> <ul style="list-style-type: none"> ■ 90 도 반시계방향 회전 <p>현재 선택한 오브젝트를 반 시계방향으로 90 도 회전합니다.</p>
대칭	<p>정렬툴바의 [대칭]과 동일한 기능입니다.</p> <ul style="list-style-type: none"> ■ 상하대칭 : 현재 선택한 오브젝트를 상하 대칭합니다. ■ 좌우대칭 : 현재 선택한 오브젝트를 좌우 대칭합니다.
배경이미지	<ul style="list-style-type: none"> ■ 설정 : 현재 뷰페이지에 배경이미지를 적용합니다. ■ 해제 : 뷰페이지 배경이미지를 제거합니다. ■ 저장 : 현재 뷰페이지의 배경이미지를 파일에 저장합니다.
확대 화면 축소	현재 뷰페이지의 화면을 확대/축소 (50% ~400%) 해서 편집 합니다.
라이브러리 불러오기	이미지 라이브러리 창에서 이미지 오브젝트를 선택하고 추가합니다.
인쇄	현재 뷰페이지를 인쇄합니다.
오브젝트 리스트	현재 뷰페이지에 그려진 오브젝트 리스트 창입니다. 적색 박스부분과 같이 사각형 오브젝트 리스트를 표시하고 있습니다. 이중 하나를 선택하면 선택커서가 해당 오브젝트로 변경됩니다. 오브젝트 찾을 때 사용합니다.
	
이미지 저장	선택한 이미지 오브젝트를 파일로 저장할 수 있습니다.
오브젝트 잠그기	<p>오브젝트를 선택이 안되도록 LOCK을 걸어두면 편집할 때 편리합니다.</p> <p>오브젝트 잠그기 상태는 저장이 됩니다.</p>
오브젝트 풀기	<p>현재 위치의 잠겨있는 오브젝트를 해제합니다.</p> <p>이 메뉴 항목은 잠겨있는 오브젝트에서만 활성화 표시됩니다.</p>
오브젝트 전체풀기	<p>현재 뷰 페이지에 잠겨있는 오브젝트를 모두 해제합니다.</p> <p>이 메뉴 항목은 현재 뷰페이지에 오브젝트가 한 개라도 잠겨있으면 활성화</p>

	표시됩니다.
오브젝트 크기설정	현재 선택된 오브젝트의 크기를 수치로 입력을 해서 원하는 크기로 변경할 수 있습니다. 비율유지를 체크한 후 폭과 높이를 조정하면 일정한 비율로 오브젝트 크기가 변경됩니다. 

5.1.7 표현식 속성

그리기 오브젝트의 동적인 움직임을 제어하는데 사용하는 속성이 표현식 부분입니다. 표현식에는 한 개 이상의 태그명을 입력해주거나, 산술연산식을 사용해서 값을 변환할 수 있습니다. 오브젝트의 움직임은 이 표현식의 계산 결과값으로 대부분 제어됩니다. 모든 오브젝트에는 표현식 속성창이 있으며 사용방식은 모두 동일합니다.

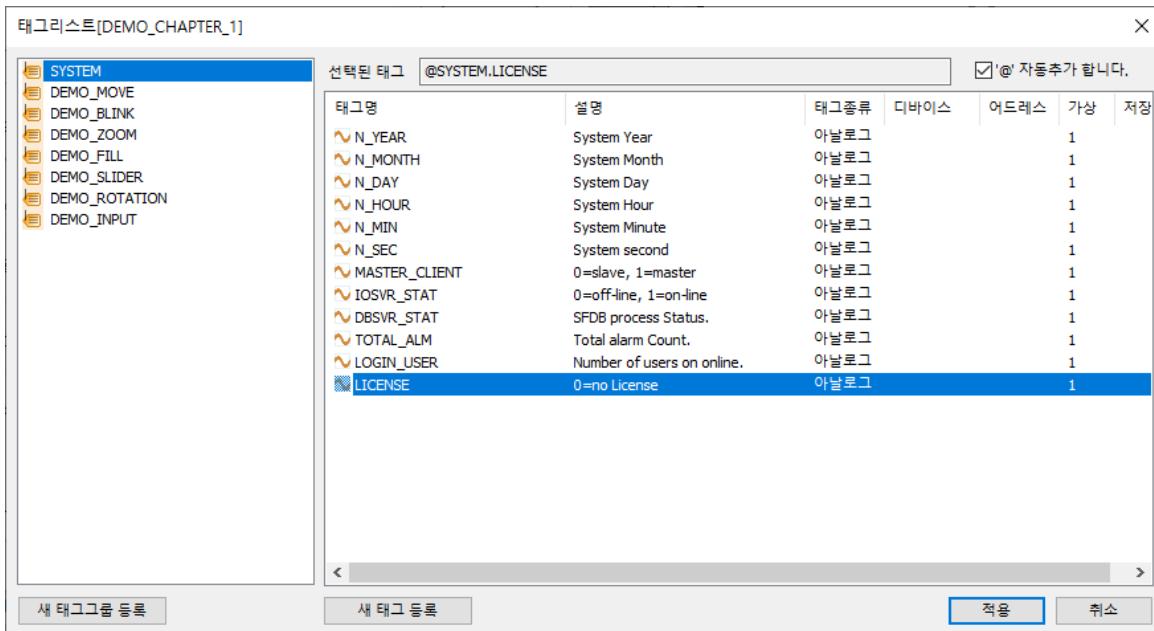


표현식에 사용할 수 있는 연산으로는 사칙연산(+, -, *, /) 및 논리연산(&, |, ^)을 할 수 있습니다. 그러나 스크립트에서 사용하는 조건, 반복 키워드와 같은 표현식은 사용할 수 없으며 구문분석을 통해서 구문오류 메시지를 띠웁니다.

결과값은 다음과 같이 계산식에 따라서 다른 형태로 나옵니다.

- ① 논리연산(AND, OR, NOT)이면 결과값은 0 또는 1
- ② 조건연산 (=, !=, >, <) 이면 결과값은 0 또는 1
- ③ 사칙연산 그래도 연산결과

[태그버튼]을 눌러서 프로젝트에 등록된 태그리스트 창을 띄워서 태그를 선택합니다.



태그명 앞에는 항상 태그값을 의미하는 문자 @를 붙여서 사용해야 합니다.

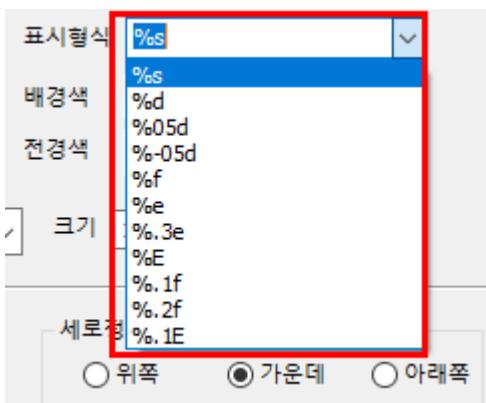
[구문분석]을 눌러서 구분에 오류가 있는지 확인할 수 있습니다.

[표현식 사용 예]

- ① @DEMO.VAL1 ; 태그(DEMO.VAL1)값 가져옴
- ② @DEMO.VAL1 !=0 ; 태그(DEMO.VAL1) 값이 0이 아니면 결과는 TRUE(1)
- ③ @DEMO.VAL1 + @DEMO.VAL2 ; 2개의 태그값 합산
- ④ @DEMO.VAL1 > 10 && @DEMO.VAL2 > 50 ; 두개의 태그값이 조건이 만족할 때

표현식의 결과값을 텍스트로 출력할 경우 자리수, 소수점, 문자열등의 표시형태를 다양하게 지정할 수 있습니다.

아래와 같이 예시 외에도 직접 입력해서 표시할 수 있습니다. 스크립트 함수에서 `_FormatString()` 포맷 파라미터와 동일하게 처리되므로 참고하셔도 됩니다.



표시형식	설명
%s	표현식의 값이 문자열인 경우 사용합니다. 아날로그 태그가 사용된 표현식일 경우 화면에는 알수없음 의미의 "???" 표시가 됩니다.
%d	아날로그 태그값을 찍을 때 사용합니다.
%f	소수점 표현합니다.
%.1f	소수점 한자리 값으로 표현합니다.
%E	지수형태의 값으로 표현합니다.

5.1.8 오브젝트와 스크립트 함수연결

뷰페이지를 구성하는 오브젝트 중의 하나로 원도우 운영체제에서 제공하는 원도우 오브젝트로 각각이 제어핸들을 가지고 있어서 스크립트 함수에서 제어가 가능합니다. eRun에서 제공하는 콤포넌트 오브젝트는 다음과 같습니다.

입력창, 콤보박스, 리스트박스, 리스트콘트롤, 날짜콘트롤, 그리드, 라인그래프, 현재경보창, ACTIVE-X

뷰페이지 오브젝트는 제어를 하기 위해서 모두 오브젝트 이름을 가지고 있는데, 일반 그리기 오브젝트는 연결되는 스크립트 함수가 없기 때문에 단순히 설명용으로 설정하면 됩니다. 그러나 콤포넌트 오브젝트는 오브젝트 이름을 설정해 주어야 스크립트 함수에서 제어가 가능합니다.

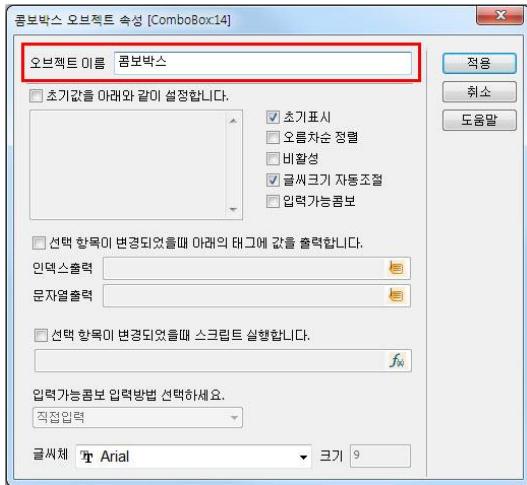
오브젝트 이름은 뷰페이지 내에서는 다르게 설정해야 합니다. 스크립트에서 콤포넌트 오브젝트에 접근할 경우 오브젝트 이름 설정방법은 아래와 같습니다.

1) “오브젝트이름@뷰페이지”

```

87 void ScanAlarmHistory1()
88 {
89     string strStart;
90     string strEnd;
91
92     strStart = _DTGetValue("DT33@06_경보상태", "%Y-%m-%d");
93     strEnd = _DTGetValue("DT34@06_경보상태", "%Y-%m-%d");
94
95     strStart = strStart + " 00:00:00";
96     strEnd = strEnd + " 23:59:59";
97
98     _AlarmScan("ALM17@06_경보상태", strStart, strEnd, "1");
99
100    //TraceEx("SearchAlarmHistory [%s]~[%s]", strStart, strEnd);
101 }
```

먼저 이름을 설정하기 위해 콤보박스 속성창을 열고, 적색부분에 영문, 한글, 숫자로 구성된 이름을 입력합니다. 여기서는 한글로 “콤보박스”로 입력하였습니다.



스크립트에서 사용할 때 다음과 같이 “**오브젝트 이름**”만 사용할 경우와 “**오브젝트이름@뷰페이지이름**”을 사용할 경우 2가지가 있는데,

(1) 아래 부분은 호출되는 뷰페이지에 “콤보박스” 오브젝트의 값을 가져올 때 이렇게 사용합니다.

```
_ComboGetValue("콤보박스", 0);
```

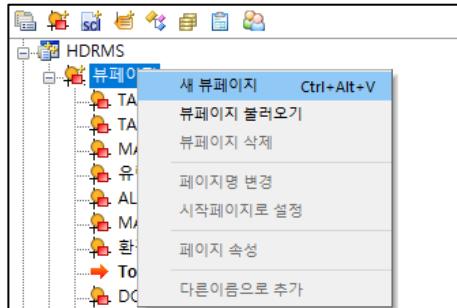
(2) 아래 부분은 뷰페이지 이름이 “메인페이지”이고 그 페이지에 있는 “콤보박스” 오브젝트의 값을 가져올 때 이렇게 사용합니다.

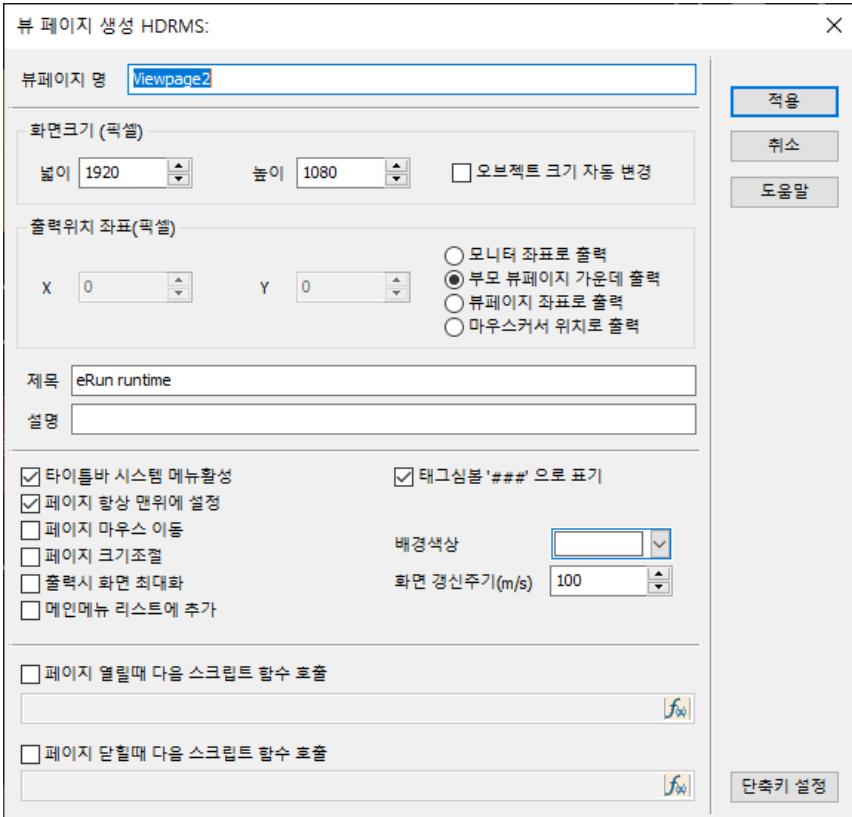
```
_ComboGetValue("콤보박스@메인페이지", 0);
```

이와 같이 함수가 호출되는 시점에 해당 뷰페이지에 있는 오브젝트를 접근하느냐 아니면 다른 뷰페이지 있는 오브젝트에 접근하느냐에 따라서 두가지 방법으로 사용합니다.

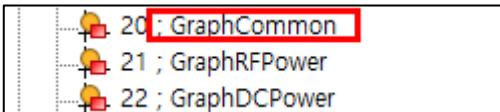
5.2 뷰 페이지 생성

프로젝트 관리창에서 [뷰페이지] 항목을 마우스 우클릭으로 선택하면 뷰페이지 관리 팝업메뉴가 나타납니다. 메뉴에서 [새 뷰페이지]를 선택하면 새로운 뷰 페이지를 생성하고 프로젝트에 추가합니다.



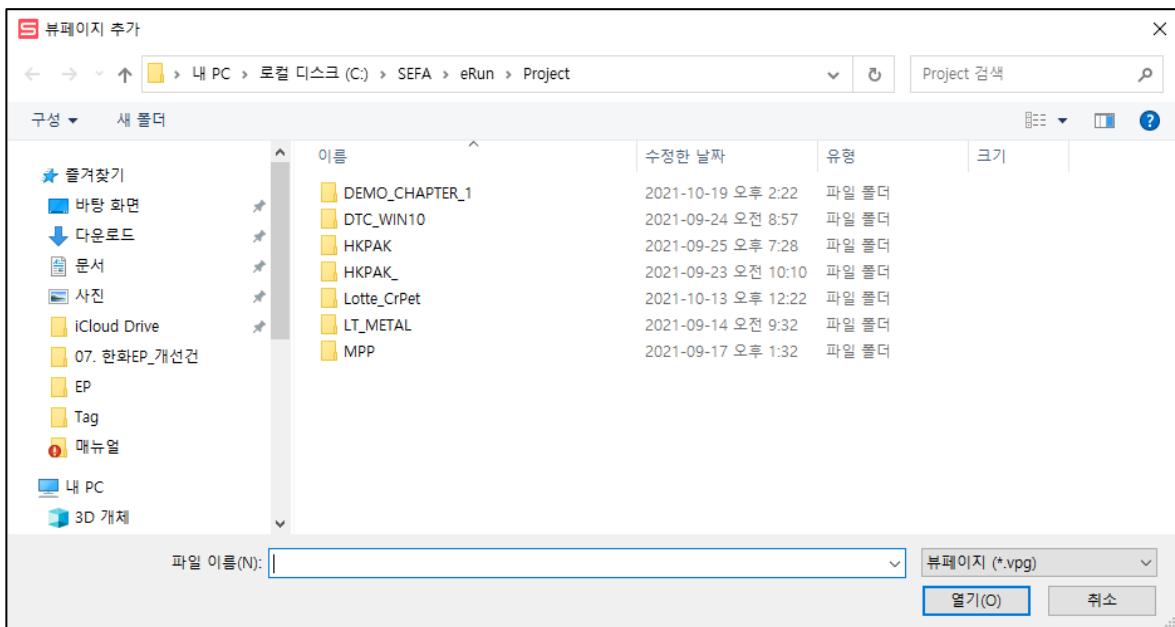


항목	설명
뷰페이지 명	뷰페이지 이름을 입력합니다. 영문, 한글, 숫자 사용 가능하나 특수문자는 사용불가입니다. (₩ / : * ? "<> ! @ # \$ % ^ & * ())
베이스뷰	뷰페이지 기본 디자인을 선택합니다.
화면크기 넓이	픽세단위의 넓이를 입력합니다. (기본 : 화면 해상도 넓이)
화면크기 높이	픽셀단위의 높이를 입력합니다. (기본 : 화면 해상도 높이)
오브젝트 크기 자동변경	페이지 크기 조절시 변경비율에 맞게 모든 오브젝트의 크기를 자동으로 조절 (스튜디오에서 사용)합니다.
프로젝트 업데이트 번호 자동증가	스튜디오에서 프로젝트 배포작업을 할 때마다 1씩 자동으로 증가하도록 합니다. 다중 사용자 PC환경에서는 반드시 해주어야 합니다.
출력위치 좌표	프로젝트 실행시 위치를 지정합니다. 모니터좌표로 출력 또는 뷰페이지 좌표로 출력을 선택한 경우 활성화됩니다.
모니터 좌표로 출력	eRun 실행시 설정된 좌표값에 해당되는 모니터의 좌표위치에 뷰페이지를 표시합니다. 다중모니터인 경우에 사용시 유용합니다.
부모페이지 가운데 출력	eRun 실행시 현재 뷰페이지가 팝업창으로 표기가 될 때 현재 뷰페이지 열기 속성이 있는 뷰페이지의 중앙에 뷰페이지를 표시합니다. 시작페이지인 경우에는 모니터 중앙에 표시합니다.
뷰페이지 좌표로 출력	eRun 실행시 설정된 좌표값에 해당되는 뷰페이지의 좌표위치에 표시합니다.

	모니터 좌표의 기준이 아닌 뷰페이지의 좌표가 기준이 됩니다. 뷰페이지의 좌상단이 (0,0)입니다.
마우스커서 위치로 출력	프로젝트 실행시 팝업창표시 할 때 사용이 됩니다. 일반적으로 팝업메뉴를 뷰페이지로 만드는 경우 사용시 유용합니다.
제목	프로젝트 실행시 메인화면의 타이틀바 제목을 입력합니다.
설명	프로젝트 관리자 창에 있는 뷰페이지 리스트에 설명을 표시하도록 합니다. “뷰페이지 이름 ; 설명” 
타이틀바 시스템메뉴 설정	eRun 실행시 타이틀바의 시스템 메뉴의 활성화 여부를 선택합니다.
페이지 항상 맨위에 설정	현재의 뷰페이지를 다른 뷰페이지 보다 앞에 표시할 수 있게 설정합니다.
페이지 마우스 이동	마우스로 페이지를 클릭과 드래그를 하여 페이지의 위치를 이동할 수 있습니다.
페이지 크기 조절	화면 지정 크기와는 상관없이 모니터의 화면 크기에 맞추어 최대화하여 표시합니다.
출력시 화면 최대화	화면 지정 크기와는 상관없이 모니터의 화면 크기에 맞추어 최대화하여 표시합니다.
메인메뉴 리스트에 추가	현재 뷰페이지를 메인메뉴 리스트에 추가할 것인지 선택합니다. 이 기능은 스크립트나 오브젝트의 속성을 사용하지 않고 메뉴를 선택해서 뷰페이지를 열 수 있습니다. 단, 메인 메뉴 리스트에 추가되는 뷰페이지는 시작페이지의 크기로 변경됩니다. 도구 - 환경설정의 시작페이지의 시스템 메뉴표시가 활성화되어 있어야 합니다.
태그심볼 '####'으로 표시	문자열 오브젝트나 오브젝트의 Text 속성에서 값 표시인 경우 기본으로 설정된 표현식이 표시됩니다. 이 설정된 표현식을 ### 표시로 변경합니다. (@DEMO.1 -> #####). 이 기능은 오브젝트 단위가 아닌 페이지 전체에 적용됩니다.
배경 색상	배경색상을 설정합니다. (기본 흰색)
화면갱신주기	뷰페이지 다시 그리기 갱신주기를 설정합니다. 단위는 m/s 입니다.
열릴 때 다음 스크립트 함수 호출	페이지 열릴 때 지정된 스크립트 함수 호출합니다
닫을 때 다음 스크립트 함수 호출	페이지 닫을 때 지정된 스크립트 함수 호출합니다.
단축키 설정	eRun에서 사용할 사용자 단축키를 설정합니다. 단축키를 눌러 사용자가 작성한 스크립트 함수를 호출할 수 있습니다

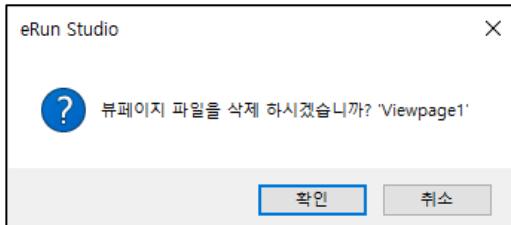
5.3 뷰페이지 불러오기

사용자가 이전에 만든 파일을 현재 프로젝트에 추가 복사할 때 사용합니다. 메뉴에서 “뷰페이지 불러오기”를 선택하면 아래 그림과 같이 뷰페이지 파일을 선택할 수 있는 파일 다이얼로그가 표시되고 추가하고자 하는 뷰페이지 파일이 현재 프로젝트에 추가됩니다.



5.4 뷰페이지 삭제

현재 선택된 뷰페이지를 삭제합니다.



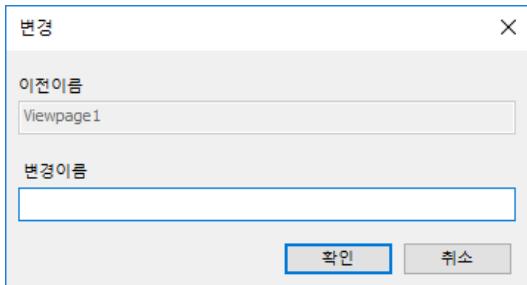
“확인”을 선택하면 페이지를 삭제하고, “취소”를 선택하면 삭제 취소합니다.

선택한 뷰페이지를 삭제하게 되면 해당 폴더에서 뷰페이지 파일을 삭제하게 됩니다.

삭제시 복구할 수 없으니 주의하시기 바랍니다.

5.5 페이지명 변경

메뉴에서 “페이지명 변경” 메뉴를 선택하면 선택한 뷰페이지의 이름을 새로운 이름으로 변경하는 팝업창이 표시됩니다.



변경 이름란에 변경하기를 원하는 페이지명을 넣으면 됩니다.

※ 주의 : 다음 문자(₩ / : * ? "<> | ! @ # \$ % ^ & * ())들은 페이지명에 포함시킬 수 없습니다

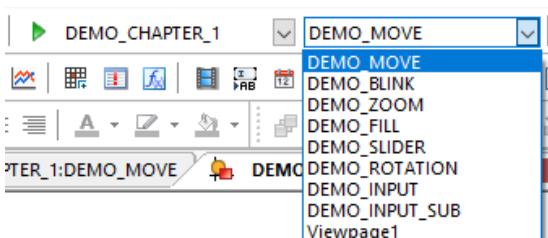
5.6 시작페이지 설정

메인 페이지라고도 하는데, 프로젝트 실행할 때 나타나는 사용자 화면 첫 페이지로 설정합니다.

■ 관리자 팝업메뉴에서 시작페이지로 설정



■ 상단 주메뉴 프로젝트 도구바에서 시작페이지로 설정



5.7 페이지 속성

뷰페이지의 크기 및 배경색, 타이틀명 등 편집을 합니다.

메뉴에서 “페이지 속성”을 클릭하거나 뷰페이지에서 마우스 오른쪽을 클릭하여 “페이지 설정”을 선택하면 뷰페이지 속성을 편집할 수 있는 페이지가 나옵니다.

※ 이 메뉴는 뷰페이지 편집기능이기 때문에 뷰페이지명을 수정할 수 없습니다.

속성에 대한 설명은 뷰페이지 생성 매뉴얼 설명서부분을 참고하세요.

5.8 다른이름으로 추가

뷰페이지 복사를 할 경우 사용하는 기능입니다. 선택한 뷰페이지를 다른이름으로 지정해서 복사 붙여넣기 합니다.

5.9 일반 오브젝트 속성설정

일반 오브젝트를 이용해서 감시화면을 효과적으로 운영하기위해서 태그를 설정하고 움직임에 대한 다양한 설정을 합니다. 속성설정은 그리기 오브젝트에 따라 약간씩 다르지만 대부분의 내용들이 유사합니다.

정적인 상태를 태그 값에 따라 동적임 움직임으로 직관적인 감시화면을 제공합니다.

속성의 종류는 기본설정, 표시방식, 이동, 확대, 축소, 채우기, 슬라이더, 회전, 입력속성으로 구성합니다. 설정창은 속성종류에 따라 보조설정창(서브 탭)이 바뀌며, 그 서브 탭 속성에 설정 값을 입력합니다. 다음 표는 오브젝트 유형에 따라서 서브 설정창이 다릅니다. ● 표시가 있는 오브젝트는 설정기능이 있는 경우를 의미합니다.

메인탭	서브탭	라인	사각	원형	둥근사각	다선	다각	열린원	닫힌원	부채꼴	텍스트	이미지	동적심볼
기본설정	기본	●	●	●	●	●	●	●	●	●	●	●	●
	텍스트		●	●	●		●						
	파이프					●							
표시	보이기숨기기	●	●	●	●	●	●	●	●	●	●	●	●
	깜빡임	●	●	●	●	●	●	●	●	●	●	●	●
이동	수평	●	●	●	●	●	●	●	●	●	●	●	●
	수직	●	●	●	●	●	●	●	●	●	●	●	●
확대축소	수평	●	●	●	●	●	●	●	●	●	●	●	●
	수직	●	●	●	●	●	●	●	●	●	●	●	●
채우기	ON/OFF		●	●	●		●		●	●			
	레벨		●	●	●		●		●	●			

	채우기조건		●	●	●		●		●	●			
슬라이 더	수평	●	●	●	●	●	●	●	●	●	●	●	●
	수직	●	●	●	●	●	●	●	●	●	●	●	●
회전	회전	●	●	●	●	●	●	●	●	●	●	●	●
입력	누르기	●	●	●	●	●	●	●	●	●	●	●	●
	놓기	●	●	●	●	●	●	●	●	●	●	●	●
	마우스오버	●	●	●	●	●	●	●	●	●	●	●	●

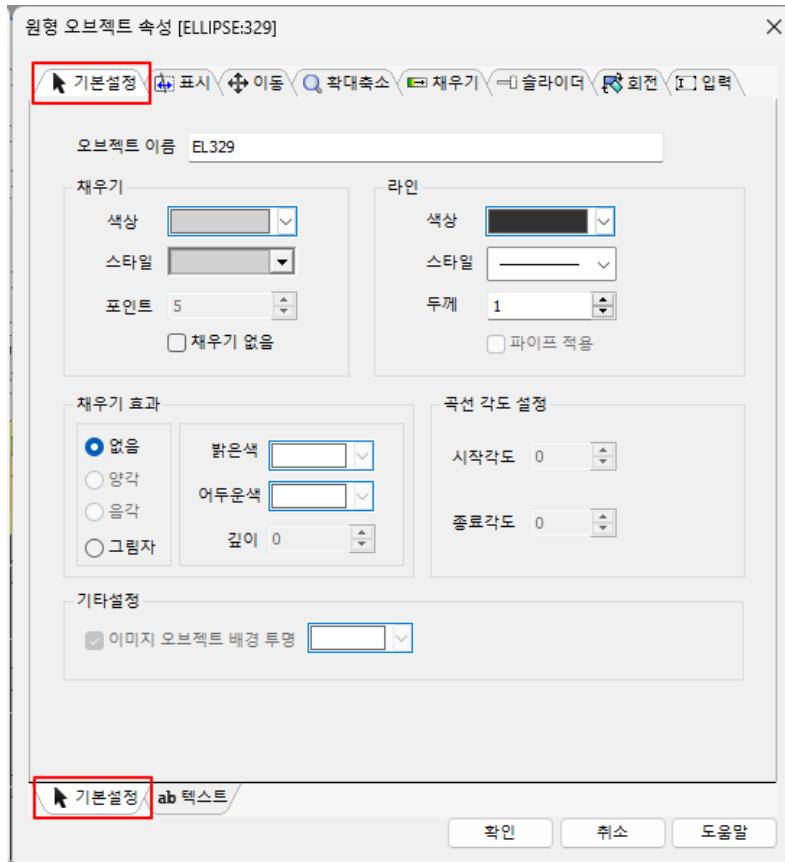
속성설정을 위해서는 오브젝트를 그리고 그 위에서 마우스 왼쪽버튼을 두번 클릭하면 설정창이 나타납니다.

5.9.1 기본설정

기본 설정창의 서브탭은 기본, 텍스트, 파이프 서브탭이 있습니다.

메인탭	서브탭	라인	사각	원형	둥근 사각	다선	다각	열린원	닫힌원	부채꼴	텍스트	이미지	동적 심볼
기본설정	기본	●	●	●	●	●	●	●	●	●	●	●	●
	텍스트		●	●	●		●						
	파이프					●							

5.9.1.1 기본설정 페이지

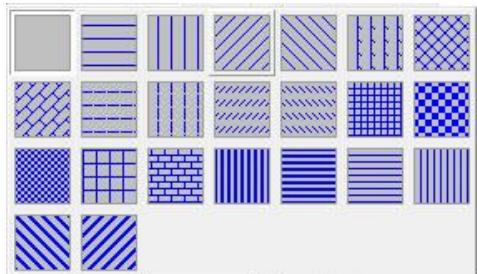


가. 오브젝트 이름 : 오브젝트의 이름을 설정합니다.

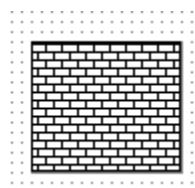
나. 채우기

색상 : 채우기 색 설정합니다.

스타일 : 채우기 효과를 낼 수 있게 아래 그림과 같이 표시됩니다.



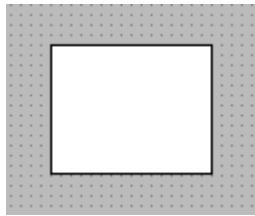
[채우기 스타일 설정]



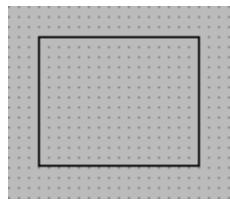
[수평벽돌무늬로 설정된 채우기 스타일]

포인트 : 둥근 사각형인 경우 활성화되는 속성입니다. 라운드의 값을 설정합니다.

채우기 없음 : 채우기 영역을 투명으로 설정합니다. 채우기 없음을 선택하면 오브젝트를 선택하기 위해서는 오브젝트의 영역이 아닌 오브젝트의 선을 선택해야 합니다.



[채우기 없음 미선택시]



[채우기 없음 선택시]

다. 라인

색 : 라인 색을 설정합니다.

스타일 : 라인의 종류 선택, 라인 스타일은 실선, 파선, 파선-점선, 긴 파선, 파선-점선-점선, 선없음으로 구성되어 있습니다.

두께 : 라인의 두께를 설정합니다.

파이프 적용 : 오브젝트가 다선형인 경우 활성화되는 속성으로 다선형을 파이프로 사용여부 설정합니다.

라. 채우기 효과 : 양각, 음각, 그림자 채우기 효과를 설정합니다.

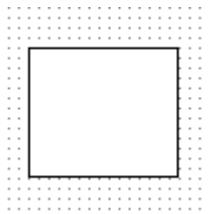


밝은색 : 오브젝트 상단과 좌측에 그려지는 색을 설정합니다.

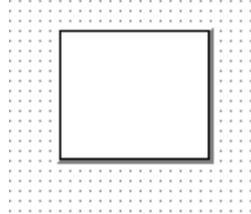
어두운색 : 오브젝트 우측과 하단에 그려지는 색을 설정합니다.

깊이 : 테두리의 두께를 설정합니다. 최대 32까지 설정할 수 있습니다.

[양각을 설정한 상태]

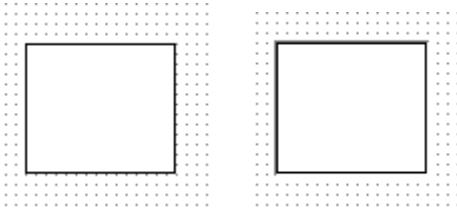


[양각 설정 전]



[양각 설정 후]

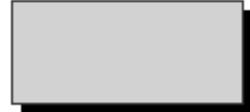
[음각을 설정한 상태]



[그림자 설정한 상태]



[그림자 설정 전]



[그림자 설정 후]

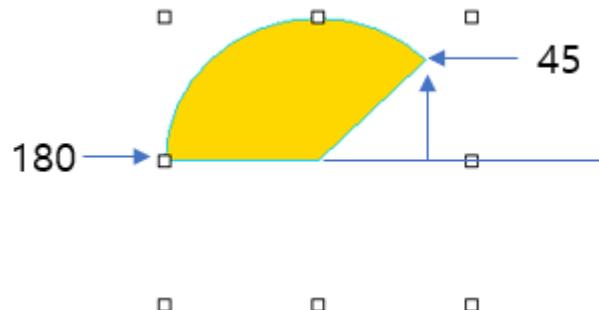
마. 곡선 각도 설정 : 열린원, 닫힌원, 부채꼴인 경우 활성화 됩니다.

시작 각도 : 시작각도 설정합니다.

종료 각도 : 종료 각도 설정합니다.

곡선의 각도 설정은 각도기의 설정 방법대로 반시계 방향으로 설정합니다.

0도 → 3시 방향 / 90도 → 12시방향 / 180도 → 9시방향 / 270도 → 6시방향



바. 기타설정

이미지 오브젝트 배경 투명 :

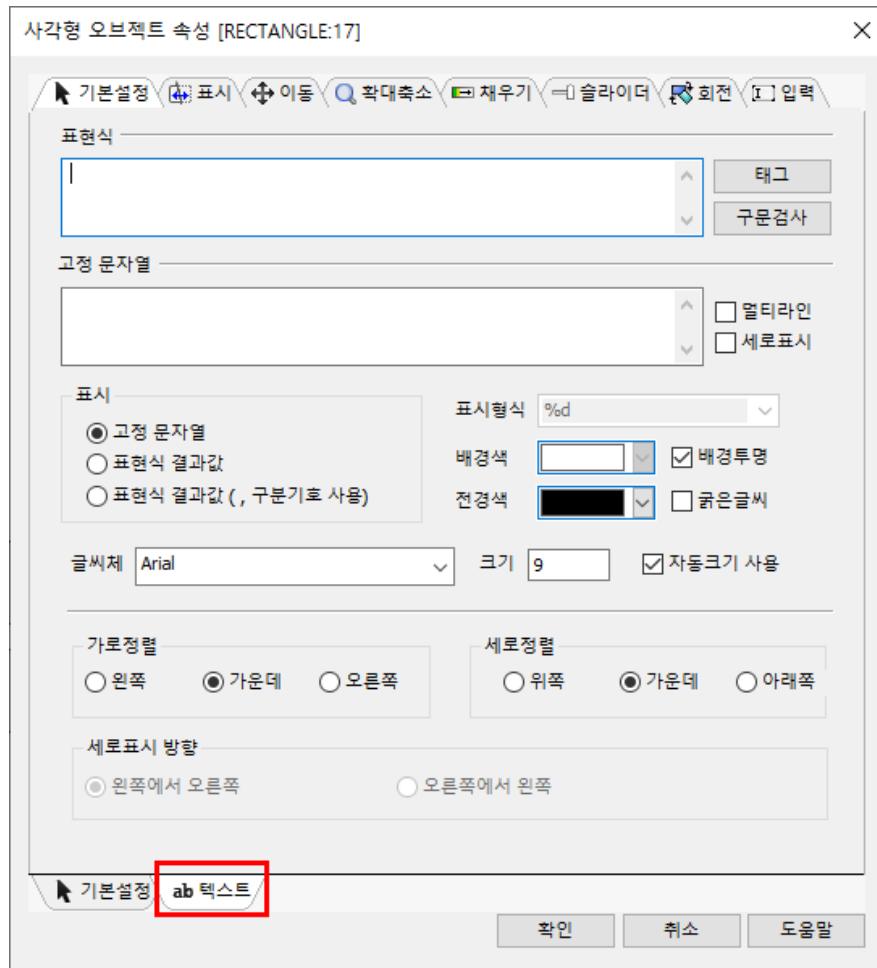
이미지 오브젝트와 비트맵 오브젝트에서 활성화 되는 속성으로 배경색상을 설정하면 설정된 색상을 제거하여 표시합니다.

그룹 오브젝트로 설정될 경우 현재 설정된 속성 사용 :

오브젝트가 그룹으로 설정이 되어도 기본으로 설정된 속성을 사용하고자 할 때 사용하는 속성입니다. 예를 들어 이미지나 문자열 오브젝트가 그룹으로 설정되고 문자열 오브젝트 보이기/숨기기가 설정되는 경우 사용합니다.

5.9.1.2 텍스트 페이지

사각형, 원형, 둥근사각형, 다각형에서 오브젝트에 고정 문자열 또는 태그값 포함 표현식 결과 값을 표시형식으로 화면에 표시합니다.



가. 표현식

표시유형이 고정 문자열이 아닌 표현식 결과값 출력유형인 경우에 사용합니다. 표현식 속성은 뷰페이지 구성의 표현식 속성 설명부분을 참고하세요.

나. 고정문자열 : 오브젝트에 고정으로 표시할 문자열을 설정합니다.

멀티라인 : 현재 입력된 문자열에 줄바꿈 문자가 있는 경우 설정이 되어 있으면 멀티라인으로 표시가 되고 그렇지 않으면 줄바꿈 문자를 포함해서 한 줄로 표시됩니다.

세로표시 : 문자열을 세로로 표시합니다.

다. 표시설정

고정문자열 : 고정된 제목등의 문자열을 표시합니다.

표현식 결과값 : 표현식의 결과값을 표시합니다.

표현식 결과값 (구분기호 사용) : 표현식의 결과값이 수치형일 때 천 단위 구분기호(,)를 표시합니다. ex)
12,345,000

라. 표시형식 : 표현식 결과값을 여러가지 형식으로 표시할 수 있습니다. 표현식 속성 설명부분을 참고하세요.

마. 글씨체 : 글씨체를 설정합니다.

크기 : 글씨체 크기를 설정합니다.

배경색 : 문자열의 배경의 색상을 지정합니다.

전경색 : 문자열의 색상을 지정합니다.

굵은글씨 : 글씨체를 볼드체로 설정합니다.

배경투명 : 문자열의 배경을 투명으로 설정합니다.

자동크기 사용 : 문자의 크기를 해당 오브젝트 크기에 맞추어 자동 조절기능을 설정합니다.

바. 가로정렬 : 문자열의 가로 정렬을 설정합니다.

세로정렬 : 문자열의 세로 정렬을 설정합니다.

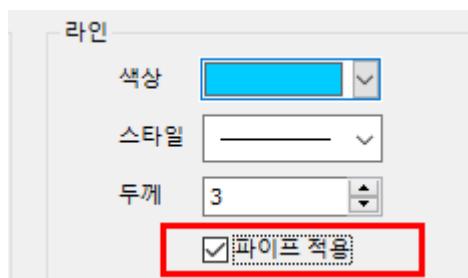
세로표시방향 : 문자열의 세로표시 선택시 활성화 됩니다.

왼쪽에서 오른쪽 : 문자열을 왼쪽에서 오른쪽으로 세로표시 합니다.

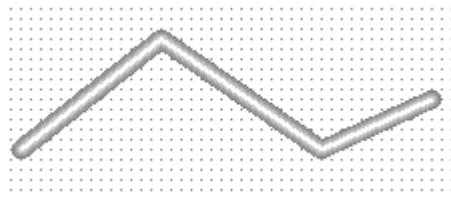
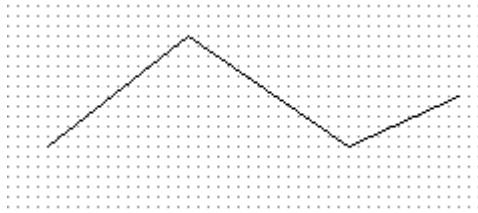
오른쪽에서 왼쪽 : 문자열을 오른쪽에서 왼쪽으로 세로표시 합니다.

5.9.1.3 파이프 속성설정

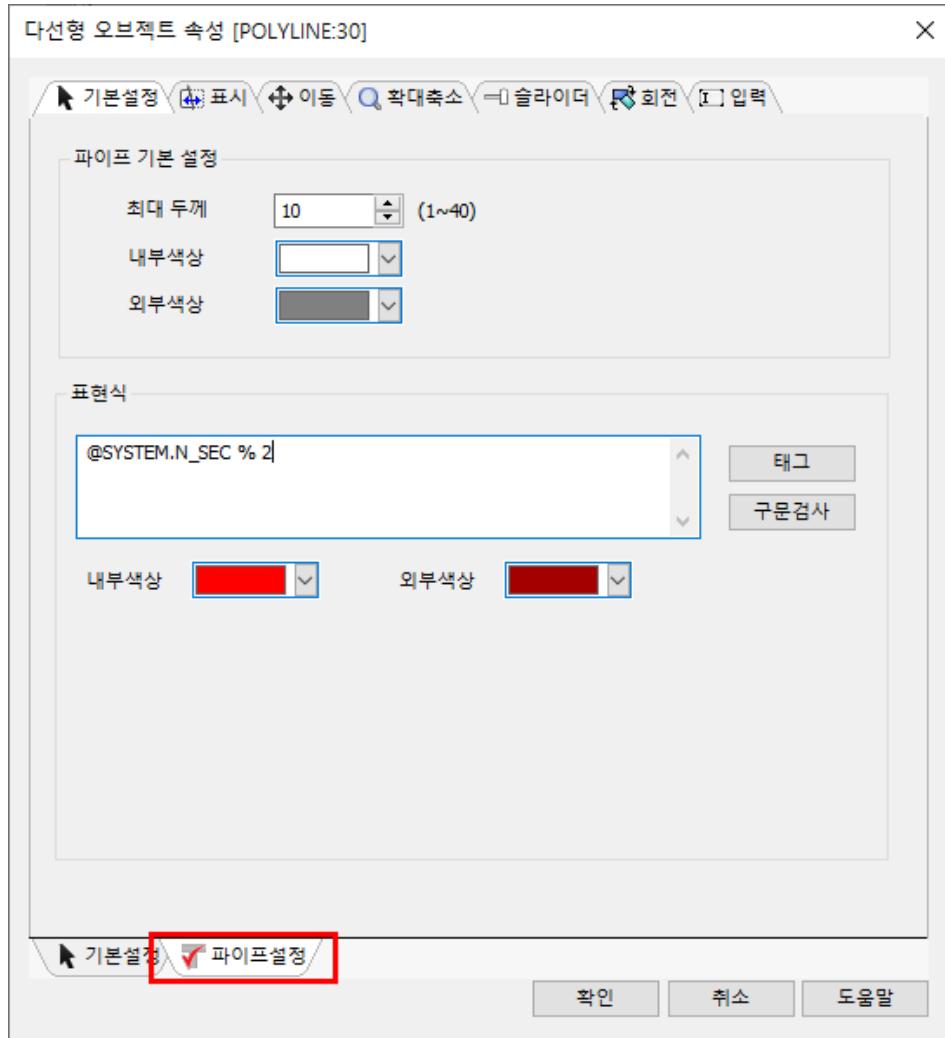
다선형 오브젝트에서만 사용가능한 속성으로 다선형을 배관 파이프 이미지의 형태로 사용할 때 설정하는 속성입니다. 파이프 속성을 설정하기 앞서 아래 그림과 같이 기본설정에서 파이프 적용을 먼저 설정합니다. ※ 파이프 적용이 되면 라인 두께의 값은 무시됩니다.



아래 다선형 오브젝트와 같이 왼쪽은 일반 다선형이고 오른쪽은 파이프 적용이 된 상태입니다.



다음은 파이프 적용속성 설정창입니다.



가. 파이프 기본설정

파이프 최대두께 : 파이프오브젝트의 두께를 설정합니다. 최대 40까지 설정할 수 있습니다.

내부색 : 파이프의 중앙에 그려질 색상을 설정합니다.

외부색 : 파이프의 바깥쪽에 그려질 색상을 설정합니다.

나. **표현식** : 표현식을 사용해서 태그 값이 0이 아닌경우 파이프의 색을 변경합니다. 예를 들어 경보 발생시나 이벤트 발생시 파이프를 특정색으로 변경합니다

내부색상 : 표현식 결과에 따라 동작하는 파이프의 중앙에 그려질 색상을 설정합니다.

외부색상 : 표현식 결과에 따라 동작하는 파이프의 외부에 그려질 색상을 설정합니다.

5.9.2 표시

표시속성에는 보이기/숨기기와 깜빡임 2가지의 속성이 있습니다.

메인탭	서브탭	라인	사각	원형	동근 사각	다선	다각	열린원	닫힌원	부채꼴	텍스트	이미지	동적 심볼
-----	-----	----	----	----	----------	----	----	-----	-----	-----	-----	-----	----------

표시	보이기/숨기기	●	●	●	●	●	●	●	●	●	●
	깜빡임	●	●	●	●	●	●	●	●	●	●

5.9.2.1 보이기/숨기기

이 기능은 오브젝트를 표현식의 결과 값에 따라 보이거나 숨기게 하는 기능입니다.



가. 표현식

표시유형이 고정 문자열이 아닌 표현식 결과값 출력유형인 경우에 사용합니다. 표현식 속성은 뷰페이지 구성의 표현식 속성 설명부분을 참고하세요.

표현식 결과 참인 경우 보이기 :

표현식 결과가 참인 경우(0 이 아닌경우) 오브젝트를 보이게 설정합니다. (이미지 오브젝트에서는 활성화 안됨)

나. 표현식 결과값 범위 설정 (이미지 오브젝트에서는 활성화 안됨)**범위사용 :**

표현식의 결과값에 대한 범위 설정, 결과값의 최소/최대값을 설정해서 표현식의 결과값이 최소/최대 범위에 있는 경우 표시상태 설정에 따라 화면에 표시 또는 숨기기 할 수 있습니다.

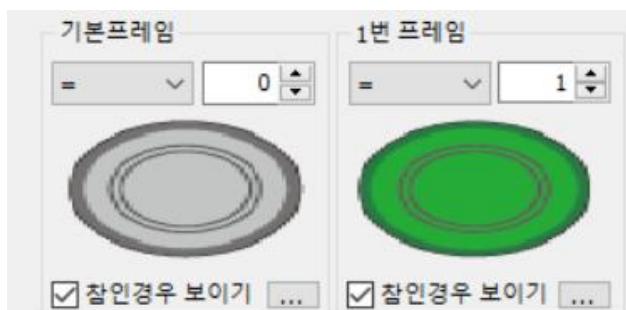
범위사용 값 범위 :

최소값: -999999999 ~ 999999999, 최대값: -999999999 ~ 999999999

이미지 오브젝트의 경우 표현식의 결과 값에 따른 이미지를 다르게 표시할 수 있습니다.

프레임 :

이미지 오브젝트의 속성으로만 사용이 되며 기본프레임부터 5개의 추가 프레임으로 구성이 됩니다. 표현식의 결과값을 각 프레임의 조건에 따라 표현됩니다.



프레임별 조건식은 $>$, \geq , $=$, \neq , $<$, \leq 6가지가 있으며 태그 값과 조건이 맞으면 참이 되면서 해당프레임이 표시됩니다. 당연하지만 각 프레임별 조건식은 서로 다르게 설정해 주어야 합니다.

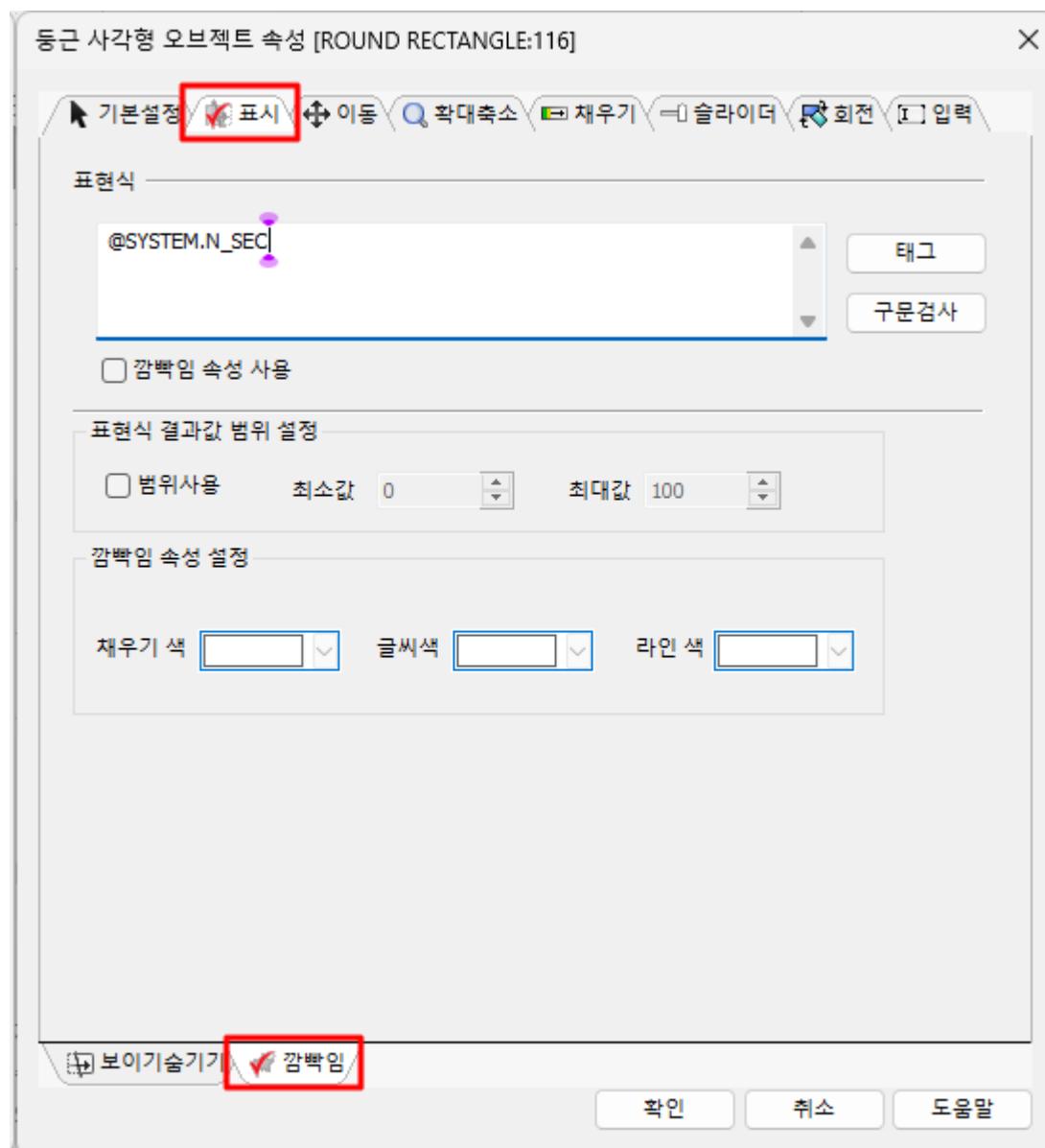


: 버튼을 클릭하여 프레임 이미지를 변경합니다.

이 기능은 채우기 속성의 조건속성과 비슷하지만 조건에 따른 해당 프레임 이미지를 표시합니다.

5.9.2.2 깜빡임

깜빡임 기능은 오브젝트를 화면에서 점멸 표시하는 기능입니다.

**가. 표현식 :**

오브젝트와 연결되는 태그 값. 이 값이 0이 아니면 오브젝트가 점멸 표시합니다.

나. 깜빡임 속성 사용 :

단순히 점멸만 하게 하려면 체크하지 않습니다.

깜빡임 속성을 사용 하지 않으면 보이기 / 숨기기와 동일한 기능을 합니다.

다. 표현식 결과값 범위 설정 :

범위 사용 : 표현식 결과값이 결과 값이 최소/최대 범위에 있는 경우 설정에 따라 화면에 표시 또는 숨기기 할 수 있습니다.

최소값: -999999999 ~ 999999999, 최대값: -999999999 ~ 999999999

라. 깜빡임 속성 설정

깜빡임 속성은 보이기/숨기기 속성에서 보이기 상태인 경우에만 속성이 적용됩니다. 숨기기 상태에서는 처리되지 않습니다.

깜빡임 속도 :

깜빡임 속도를 설정합니다. 기본은 100m/s 입니다. 깜빡임 속도가 뷰 페이지 속성의 화면 갱신주기보다 큰 경우 뷰 페이지 갱신주기에 의해 깜빡이게 됩니다.

채우기색 : 채우기색을 설정한 색으로 표시합니다.

글씨색 : 문자열 색을 설정한 색으로 표시합니다.

라인색 : 라인색을 설정한 색으로 표시합니다..

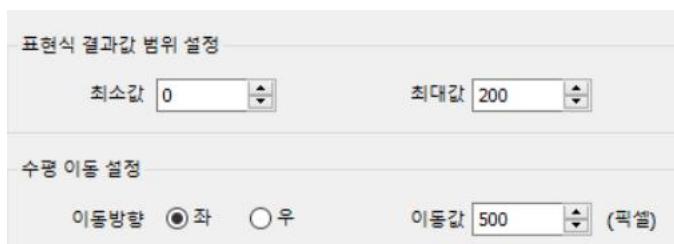
5.9.3 이동

오브젝트의 화면 표시 위치를 초기상태에서 왼쪽 또는 오른쪽으로 이동표시 합니다.. 일반 오브젝트 모두 사용이 가능한 속성입니다. 수평이동, 수직이동 2개의 서브 탭으로 구성됩니다.

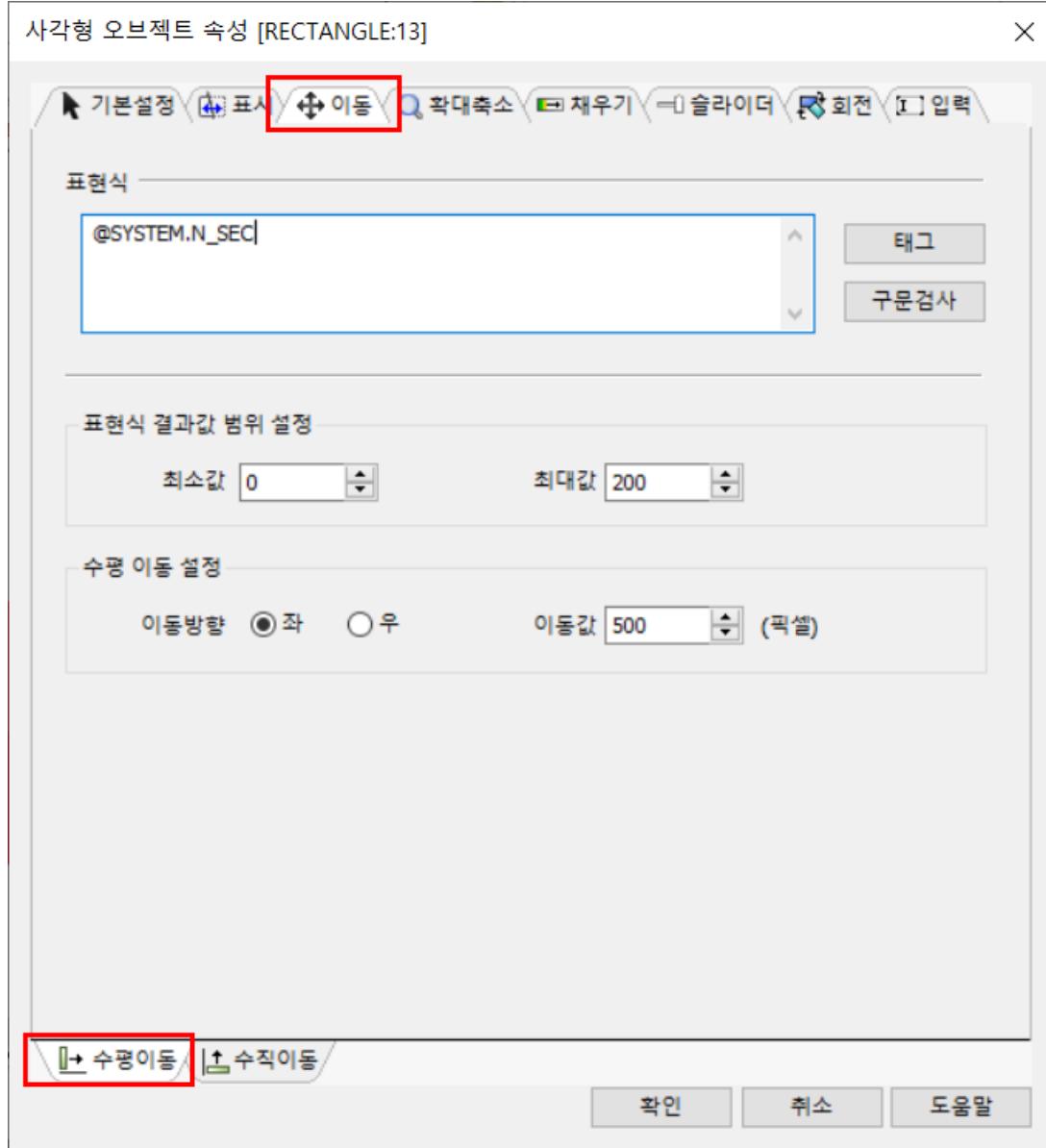
메인탭	서브탭	라인	사각	원형	등근 사각	다선	다각	열린원	닫힌원	부채꼴	텍스트	이미지	동적 심볼
이동	수평	●	●	●	●	●	●	●	●	●	●	●	●
	수직	●	●	●	●	●	●	●	●	●	●	●	●

5.9.3.1 수평이동

표현식(태그 값)을 입력 값으로 해서 수평 이동량(픽셀)에 대한 비례식으로 계산해서 오브젝트를 현재위치에서 왼쪽으로 또는 오른쪽으로 이동시킵니다.



$$\frac{\text{이동 값(픽셀)}}{(\text{표현식 최대값} - \text{표현식 최소값})} = \text{이동 값(픽셀)}$$



가. 표현식

오브젝트와 연결되는 태그 값. 자세한 사항은 표현식 설정부분 참고하세요.

나. 표현식 결과값 범위설정

결과값의 최소, 최대 입력 값을 설정합니다. 이 값이 이 해당 범위안에 있는 경우에만 오브젝트를 수평 이동합니다.

다. 수평 이동 설정

이동방향 : 오브젝트의 화면상의 현재 위치에서 왼쪽 또는 오른쪽으로 이동 방향설정 합니다.

이동값 : 표현식 결과값의 최대값에 대한 이동량을 픽셀단위로 입력합니다.

5.9.3.2 수직이동

표현식(태그 값)을 입력 값으로 해서 수직 이동량(픽셀)에 대한 비례식으로 계산해서 오브젝트를 현재위치에서 위로 또는 아래로 이동시킵니다.



$$\frac{\text{이동 값(픽셀)}}{(\text{표현식 최대값} - \text{표현식 최소값})} = \text{이동 값(픽셀)}$$

- 이동속성에 대한 설정은 수평이동 수직이동이 동일하기 때문에 추가설명은 생략합니다.

5.9.4 확대축소

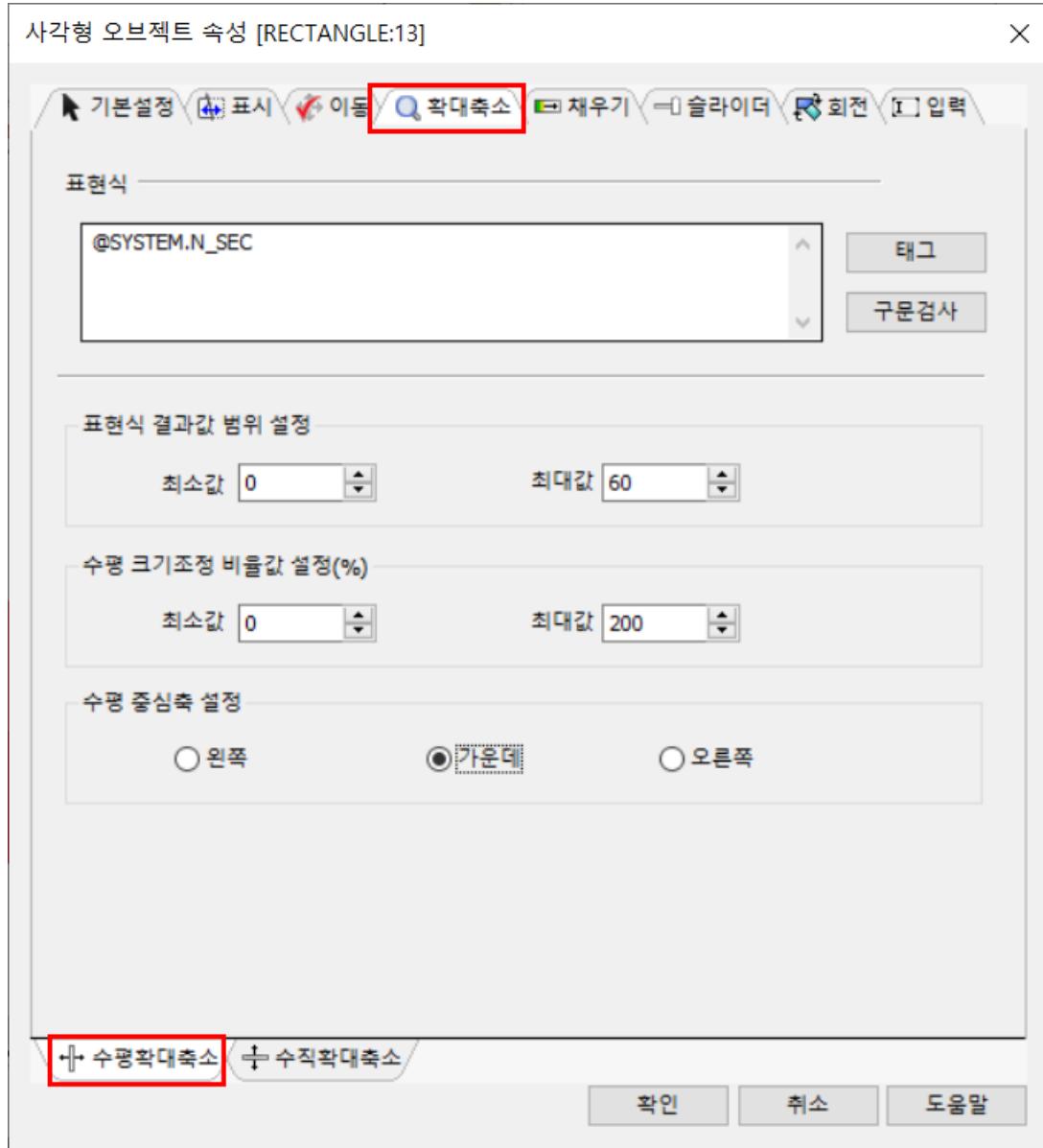
오브젝트의 화면 표시 크기를 초기상태에서 확대 표시하거나 축소 표시합니다. 일반 오브젝트 모두 사용이 가능한 속성입니다. 수평확대, 수직확대 2개의 서브 탭으로 구성됩니다.

메인 탭	서브탭	라인	사각	원형	등근 사각	다선	다각	열린 원	닫힌 원	부채 꼴	텍스트	이미지	동적심 볼
확대	수평	●	●	●	●	●	●	●	●	●	●	●	●
축소	수직	●	●	●	●	●	●	●	●	●	●	●	●

5.9.4.1 수평확대

표현식(태그 값)을 입력 값으로 해서 수평으로 확대에 대한 비례식으로 계산해서 오브젝트 표시 크기를 확대하거나 축소합니다. 초기 표시크기가 기준 값이 됩니다.

$$\frac{\text{변경 최대 비율 값} - \text{변경 최소비율 값}}{(\text{표현식 최대값} - \text{표현식 최소값})} = \text{변경 값 (초기크기에 대한 비율)}$$



가. 표현식 :

오브젝트와 연결되는 태그 값. 논리연산인 경우 결과가 0, 1 이기 때문에 정상적인 확대 축소는 되지 않습니다. 자세한 사항은 표현식 설정부분 참고하세요.

나. 표현식 결과값 범위설정

결과값의 최소, 최대 입력 값을 설정합니다. 이 값이 이 해당 범위안에 있는 경우에만 오브젝트를 수평 방향으로 확대하거나 축소합니다.

다. 수평 크기조정 비율 값

확대 축소의 비율 값을 입력합니다. 크기조정 비율 값의 단위는 %이며, 기준크기는 초기표시 크기가 100%입니다. 100%이하 입력은 축소가 되고 100% 이상 입력은 확대입니다.

라. 수평중심축 설정 :

확대 축소 방향을 설정합니다.

왼쪽 : 가장 왼쪽을 기준으로 오른쪽으로 확대/축소 됩니다.

가운데 : 가운데를 기준으로 왼쪽/오른쪽이 동일한 비율로 확대/축소가 됩니다.

오른쪽 : 가장 오른쪽을 기준으로 왼쪽으로 확대/축소 됩니다.

예시)

결과값 범위 설정 → 최소값:0, 최대값:200

확대 축소 값 범위 설정 → 최소값:0, 최대값:500

중심축: 왼쪽

→ 결과값이 0인 경우에는 0% 축소시키고, 결과값이 200인 경우에는 500% 확대, 왼쪽을 기준으로 오른쪽으로 확대 축소 시킵니다.

5.9.4.2 수직확대

표현식(태그 값)을 입력 값으로 해서 수직으로 확대에 대한 비례식으로 계산해서 오브젝트 표시 크기를 확대하거나 축소합니다. 초기 표시크기가 기준 값이 됩니다.

$$\frac{\text{변경 최대 비율 값} - \text{변경 최소비율 값}}{(\text{표현식 최대값} - \text{표현식 최소값})} = \text{변경 값 (초기크기에 대한 비율)}$$

- 확대/축소 속성에 대한 설정은 동일하기 때문에 추가설명은 생략합니다.

5.9.5 채우기

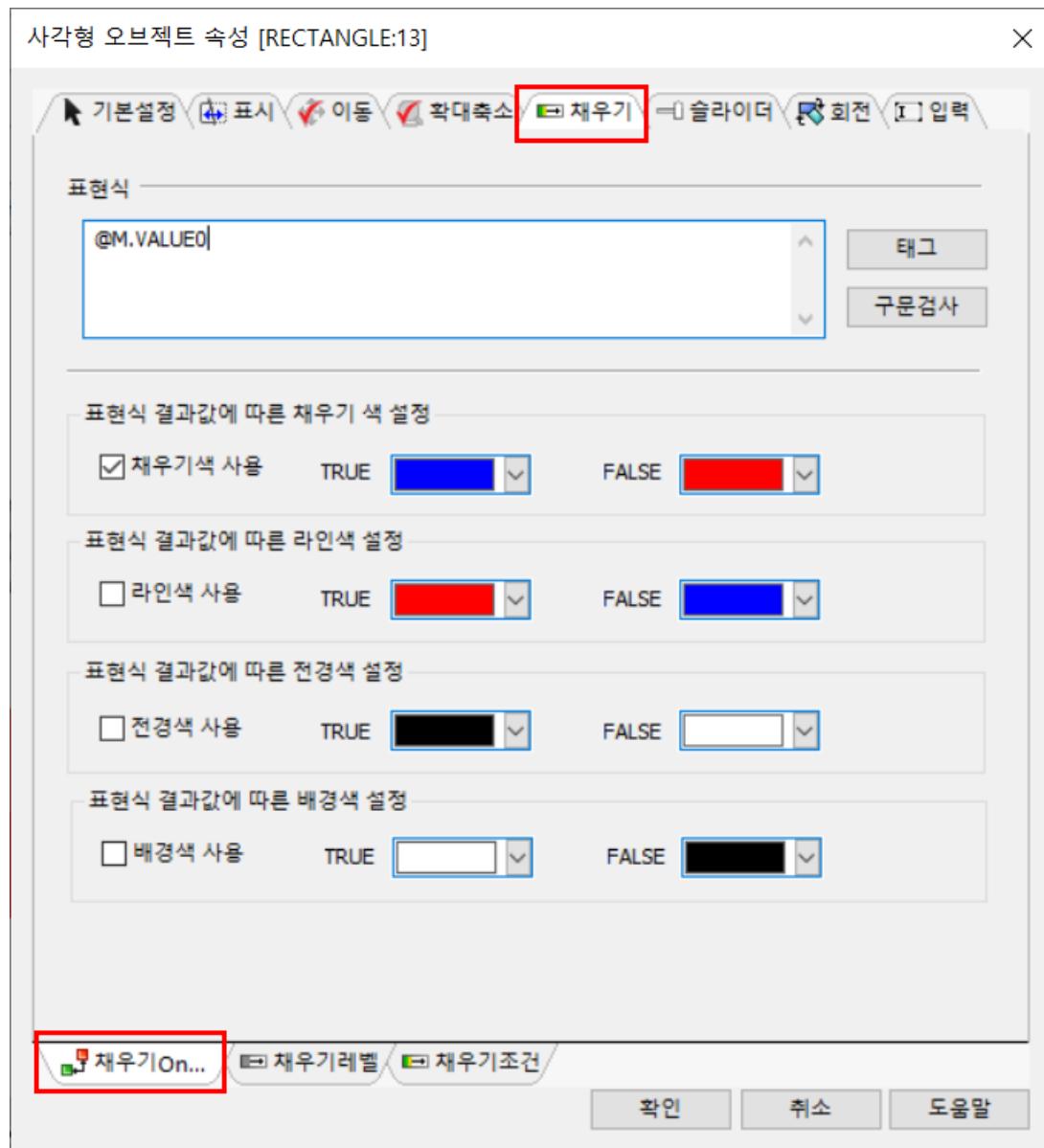
닫힌 도형 오브젝트의 채우기, 글씨색을 설정하는 속성이고 사용가능한 오브젝트는 다음 표와 같습니다. 채우기 ON/OFF, 채우기레벨, 채우기조건 3개의 서브 탭으로 구성됩니다.

메인탭	서브탭	라인	사각	원형	등근 사각	다선	다각	열린 원	닫힌 원	부채 꼴	텍스 트	이미 지	동적 심볼
채우기	ON/OFF		●	●	●		●		●	●			

	레벨	•	•	•	•	•	•	•		
	채우기조 건	•	•	•		•		•		

5.9.5.1 채우기 ON/OFF

채우기 On/Off는 표현식의 결과값이 0인 경우에는 해당 설정의 OFF에 설정된 색이 적용되고 0이 아닌 경우에는 ON에 설정된 색이 적용됩니다. 표현식 결과가 0이 아닌 값이 들어오면 TRUE에 설정한 색이 표시되고 0이면 FALSE에 설정한 색이 표시 됩니다.



가. 표현식 :

오브젝트와 연결되는 태그 값. 자세한 사항은 표현식 설정부분 참고하세요.

나. 표현식 결과에 따른 채우기 색 설정

태그 값이 0이 아닐 때 정상색(녹색)을 지정하고, 0일때는 멈춤색(회색)을 표시합니다. 기계의 동작상태를 표시할 때 주로 사용합니다.

다. 표현식 결과값에 따른 라인색 설정

오브젝트 라인색을 지정합니다.

라. 표현식 결과값에 따른 전경색 설정

문자열이 있는 오브젝트 사각형, 둥근 사각형, 타원, 다각형에만 해당하며 글씨색을 변경합니다.

마. 표현식 결과값에 따른 배경색 설정

문자열이 있는 오브젝트 사각형, 둥근 사각형, 타원, 다각형에만 해당하며 글씨 배경색을 변경합니다.

5.9.5.2 채우기 레벨

채우기 레벨은 오브젝트의 높낮이 표시를 하기위해 사용하는 오브젝트로 표현식의 결과값을 채움의 크기로 표시합니다. 주로 오일탱크 레벨, 저장소 수위를 표현할 때 사용합니다.

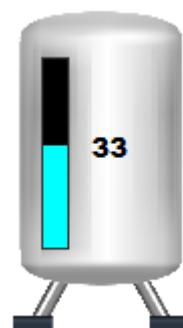


가. 표현식

오브젝트와 연결되는 태그 값. 자세한 사항은 표현식 설정부분 참고하세요.

나. 표현식 결과값 범위설정

표현식 결과값의 최소, 최대 입력 값 범위를 입력합니다.



표현식 결과값을 0에서 60으로 설정하고 아래서 위로 채움 방향을 설정하였습니다. 그림은 표현식 결과 값이 33 일 때 레벨표시 상태입니다.

다. 레벨색상 설정

레벨 표시 색을 설정합니다. 그라데이션은 레벨색상에서 초기 설정된 색으로 2가지 색으로 오브젝트 레벨 표시 합니다. 오브젝트에 회전속성 설정이 있는 경우에는 그라데이션 효과는 적용 되지 않습니다.

라. 레벨방향 설정

레벨 표시를 어디서부터 시작할 것인지 설정합니다.

5.9.5.3 채우기 조건

채우기 조건은 표현식 결과값에 따라 조건을 걸어서 특정색을 표시할 수 있습니다. 최소범위와 최대범위를 설정하여 결과값에 따라서 최대 10개까지 다른 색을 설정할 수 있습니다. 조건식은 (<, <=, =, >, >=) 입니다.



다음과 같은 설정은 표현식 결과값이 1~3이면 빨강색으로 채웁니다.





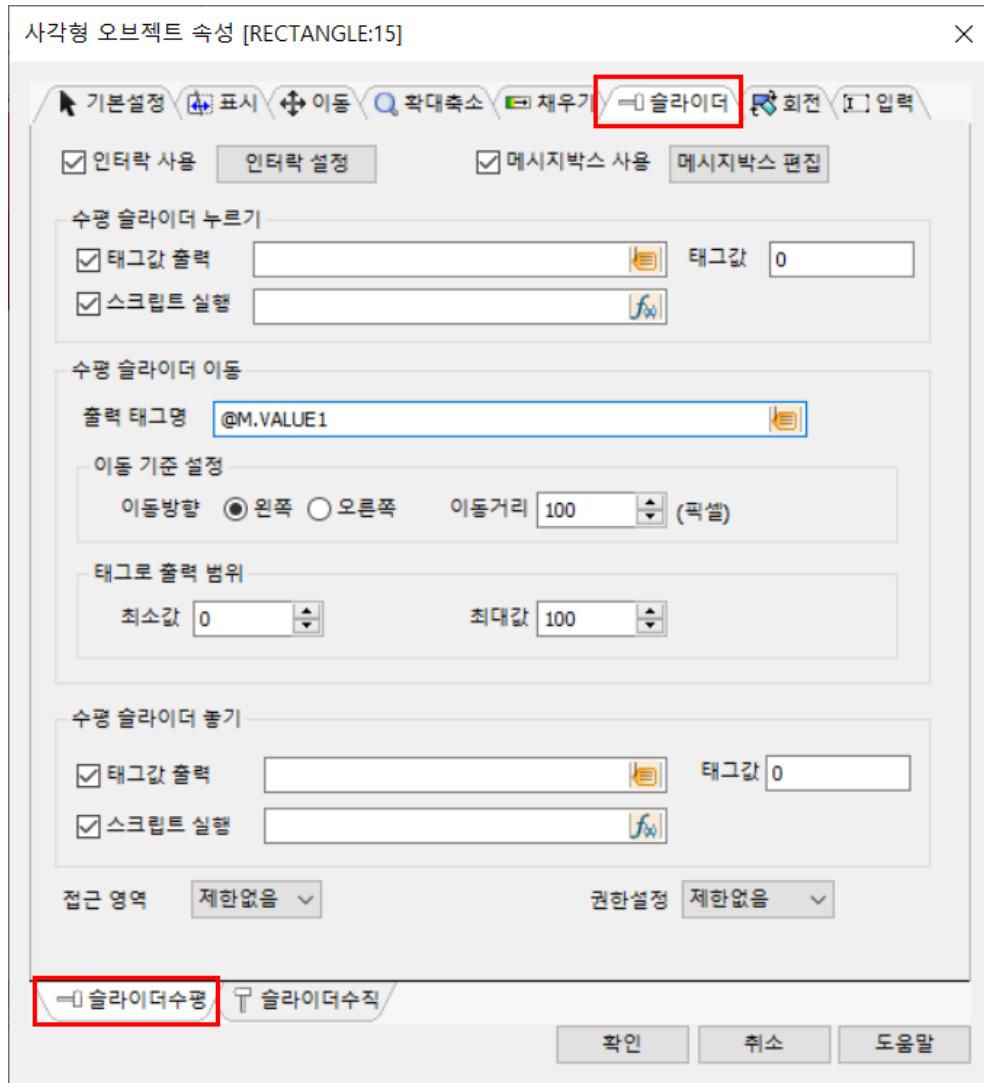
5.9.6 슬라이더

슬라이더는 마우스 버튼을 누른 상태에서 오브젝트를 이동시키면 이동한 양을 계산해서 출력태그에 그 값을 저장합니다. 슬라이더 특성은 입력과 출력 두가지 속성을 가지고 있는 오브젝트입니다. 다시 말해서 태그 값에 따라 오브젝트를 이동시키는 입력과 사용자가 마우스 이동으로 오브젝트 이동했을 때 태그에 그 값을 저장하는 출력기능 속성입니다.

슬라이더 기능이 사용 가능한 오브젝트는 다음 표와 같습니다. 수평, 수직 슬라이더 2개의 서브 탭으로 구성됩니다.

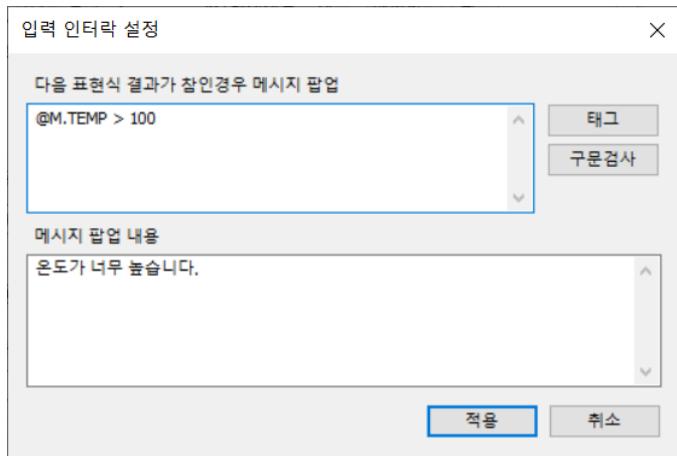
메인탭	서브탭	라인	사각	원형	둥근 사각	다선	다각	열린 원	닫힌 원	부채꼴	텍스트	이미지	동적 심볼
슬라이더	수평	●	●	●	●	●	●	●	●	●	●	●	●
	수직	●	●	●	●	●	●	●	●	●	●	●	●

5.9.6.1 수평 슬라이더



가. 인터락(INTERLOCK) 사용

슬라이더는 태그에 출력하는 기능이 있으므로 인터락 설정을 하는 기능이 있습니다. 인터락 사용을 체크하고 인터락 속성을 설정합니다. 인터락은 사용자가 기계 조작을 할 때 보안장치 또는 기계 상태 확인이 필요할 때 사용합니다.

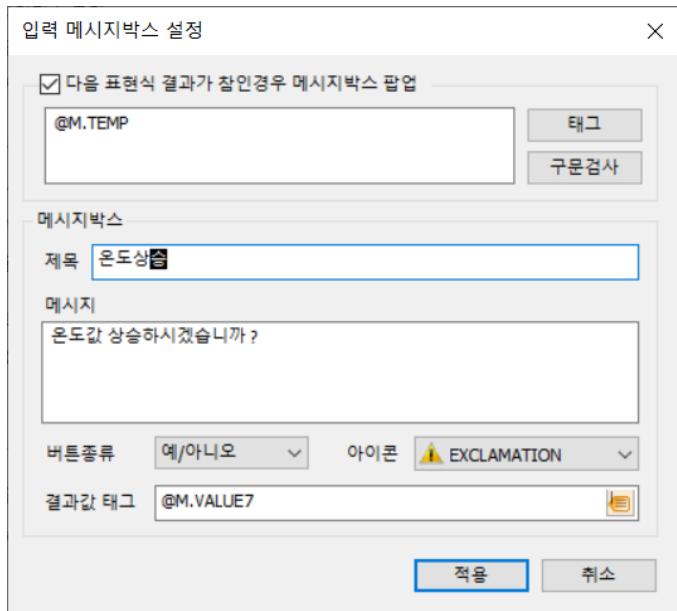


표현식에 인터랙 조건식을 입력하고, 메시지 내용을 입력하고 적용하면 됩니다.

위의 표현식은 슬라이더 이동을 위해 마우스 클릭했을 때 태그 @M.TEMP 값이 100보다 크면 메시지 팝업을 띠워서 표시해주고 슬라이더 이동을 실행하지 않습니다.

나. 메시지박스 사용

사용자가 입력 처리할 때 메시지를 확인을 하고 처리해야 하는 경우 메시지 창을 띠워서 사용자에게 메시지를 묻고 진행을 합니다.



메시지 박스 설정 내용은 입력속성 설정기능편을 참고해 주시기 바랍니다.

다. 수평슬라이더 누르기

오브젝트를 마우스로 클릭했을 때 설정한 태그 값 출력 그리고 스크립트 함수를 실행합니다.

태그값 출력 체크를 하면 태그값을 설정해줍니다. 스크립트 실행 체크를 하면 사용자 함수를 입력해줍니다.

라. 수평슬라이더 이동

오브젝트를 마우스로 이동할 때 출력 태그 값과 화면에서의 이동거리를 설정합니다.

출력 태그명 : 출력범위 해당하는 값을 저장할 태그를 지정합니다.

이동기준 설정 (화면상 이동량)

이동방향 : 오브젝트의 초기위치에서 이동 방향을 설정합니다.

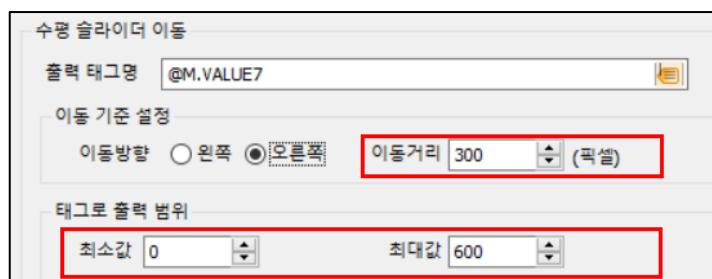
이동거리 : 페이지 뷰 화면상에서 오브젝트의 이동 거리 설정합니다.

태그로 출력범위 (이동량의 비례값)

최소값 : 오브젝트의 이동거리에 비례하는 최소 값을 설정합니다.

최대값 : 오브젝트의 이동거리에 비례하는 최대 값을 설정합니다.

예시로 다음과 같이 설정하였을 때 프로젝트 실행할 때 태그값의 변화를 보시기 바랍니다.



프로젝트 실행화면은 다음과 같이 동작합니다.



마. 수평슬라이더 놓기

수평슬라이더 누르기와 반대로 오브젝트를 이동하고 마우스를 놓았을 때 설정한 태그 값 출력 그리고 스크립트 함수를 실행합니다. 태그값 출력 체크를 하면 태그 값을 설정해줍니다. 스크립트 실행 체크를 하면 사용자 함수를 입력해줍니다.

5.9.6.2 수직 슬라이더

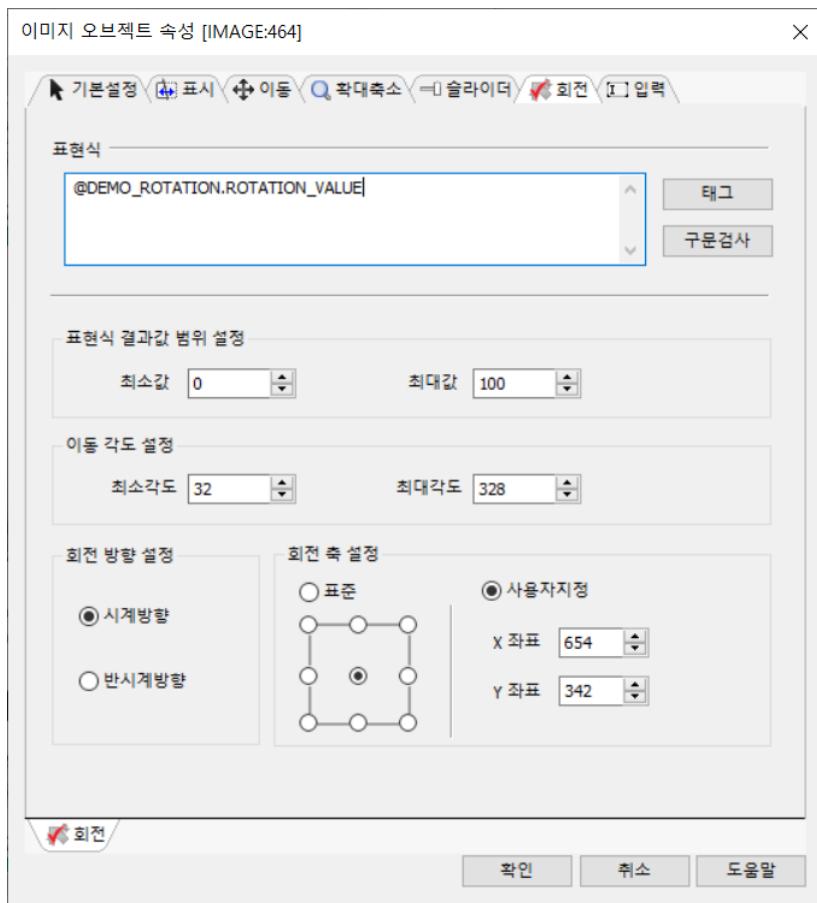
수평 슬라이더와 수직 슬라이더는 움직임 방향만 다르고 설정방법, 실행방법은 서로 동일하므로 추가적으로 설

명은 생략하겠습니다.

5.9.7 회전

표현식 결과값 범위에 따라 회전각도를 계산해서 오브젝트를 회전합니다. 회전속성은 아래 표와 같이 그리기 오브젝트 모두 사용이 가능합니다. 회전속성 서브 탭은 별도로 없습니다.

메인탭	서브탭	라인	사각	원형	둥근 사각	다선	다각	열린 원	닫힌 원	부채꼴	텍스트	이미지	동적 심볼
슬라이더	회전	●	●	●	●	●	●	●	●	●	●	●	●



가. 표현식

오브젝트와 연결되는 태그 값. 자세한 사항은 표현식 설정부분 참고하세요.

나. 표현식 결과값 범위설정

표현식 결과값이 해당 범위안에 들어와야 오브젝트가 회전합니다

다. 이동 각도 설정

표현식 결과값의 비례식으로 회전각도를 설정합니다.

라. 회전방향

오브젝트의 회전방향을 설정합니다. 기본방향은 시계방향으로 회전합니다.

마. 회전축 설정

오브젝트 회전의 기준이 되는 축을 선택합니다.

표준 : 9개의 위치 중 선택을 합니다.

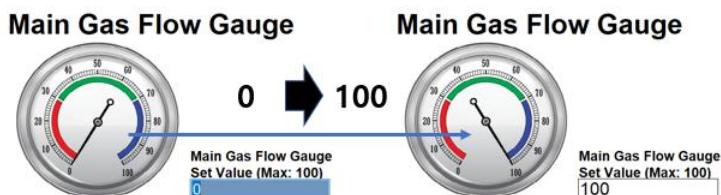
사용자 지정 : 회전축 중심 화면상의 x, y 좌표로 설정합니다.

뷰 페이지 상태바에 표시되는 마우스의 X, Y 좌표를 확인해서 X, Y 좌표를 입력하여 회전의 중심을 설정합니다.



표현식의 계산 값은 0에서 100의 범위 값을 취합니다. 이값을 이동 각도 설정의 비례식으로 변환합니다. 여기 오브젝트에서는 회전속성을 지시바늘 이미지에 설정하였습니다. 배경 이미지의 0의 각도와 100일 때 각도를 계산해서 입력해줍니다. 회전축은 사용자지정으로 뷰페이지에서 마우스로 위치값을 얻어서 입력해줍니다.

[프로젝트 실행한 화면]



5.9.8 입력

입력 속성은 오브젝트를 통하여 사용자로 하여금 마우스 클릭으로 감시화면을 제어하거나 태그 값을 내보내서 디바이스 제어를 하기 위한 설정 속성입니다. 다음 표와 같이 오브젝트 모두 입력 속성설정이 가능합니다. 누르기, 놓기, 마우스 오버 3가지의 서브 탭으로 구성됩니다. 마우스 오버 서브속성을 제외한 2가지(누르기, 놓기) 속성설정 방법 및 내용이 같습니다.

매뉴얼 설명에서는 누르기 속성만 설명하도록 하겠습니다.

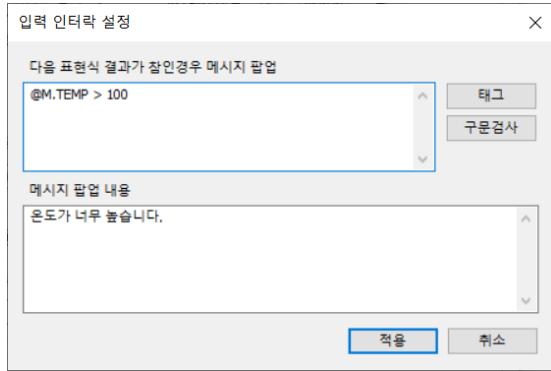
입력속성에는 다음과 같은 제어기능을 설정합니다.

메인 탭	서브탭	라인	사각	원형	등근사각	다선	다각	열린원	닫힌원	부채꼴	텍스트	이미지	동적심볼
입력	누르기	●	●	●	●	●	●	●	●	●	●	●	●
	놓기	●	●	●	●	●	●	●	●	●	●	●	●
	마우스 오버	●	●	●	●	●	●	●	●	●	●	●	●

- 인터락 검사
- 메시지 묻기
- 뷰 페이지 열기, 닫기, 전환
- 태그값 처리 (쓰기, 토플, 증가, 감소, 모멘터리)
- 스크립트 함수 실행
- 그래프 보기
- 입력창

5.9.8.1 인터락(INTERLOCK)

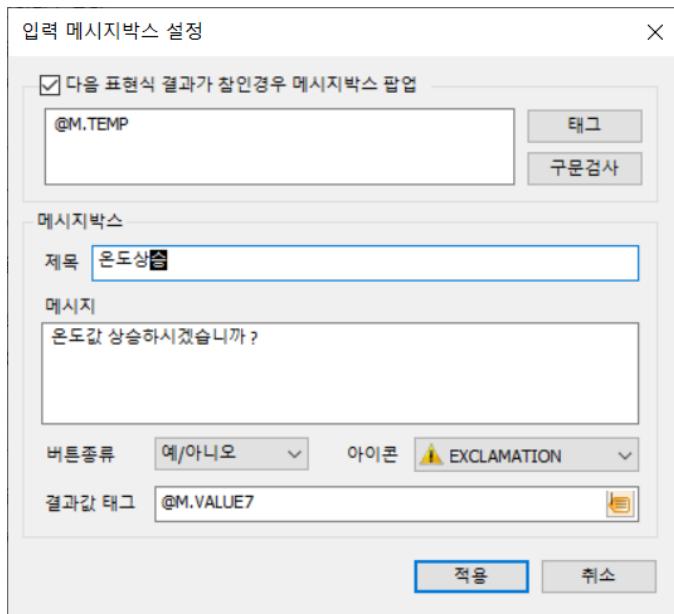
시스템을 운영하는데 안전운전을 위해서 보호장치가 반드시 필요합니다. 보호와 안전운전을 위해서 인터락 기능을 사용합니다.



입력 인터랙션 설정 창에서 표현식에 인터랙션 조건식을 입력하고, 메시지 내용을 입력하고 적용하면 됩니다. 위의 표현식은 오브젝트위에서 마우스 클릭했을 때 태그 @M.TEMP 값이 100보다 크면 메시지 팝업을 띄워서 표시해 주고 오브젝트 누르기는 실행되지 않습니다.

5.9.8.2 메시지박스 사용

사용자가 입력 처리할 때 메시지를 확인을 하고 처리해야 하는 경우 메시지 창을 띄워서 사용자에게 메시지를 묻고 진행을 합니다.



가. 표현식

표현식의 계산 결과가 0이 아니면 메시지 박스를 띄워서 진행을 묻습니다. 표현식의 구문은 표현식 설정 설명부를 참고하세요.

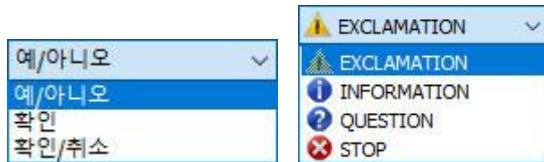
나. 메시지박스

제목 : 메시지박스 타이틀 텍스트를 입력합니다.

메시지 : 메시지 내용을 입력합니다.

다. 버튼종류 & 아이콘

예/아니오, 확인, 확인/취소 3가지의 버튼 형태가 있습니다. 아이콘은 메시지 옆에 나타나며 4종류가 있습니다.

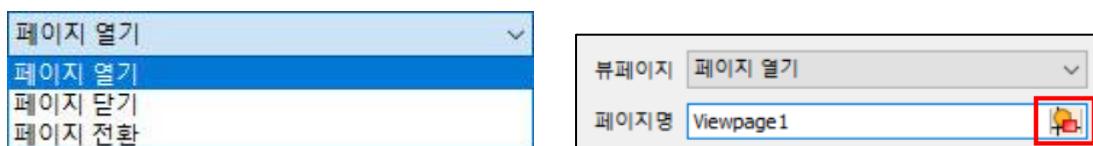


라. 결과값 태그

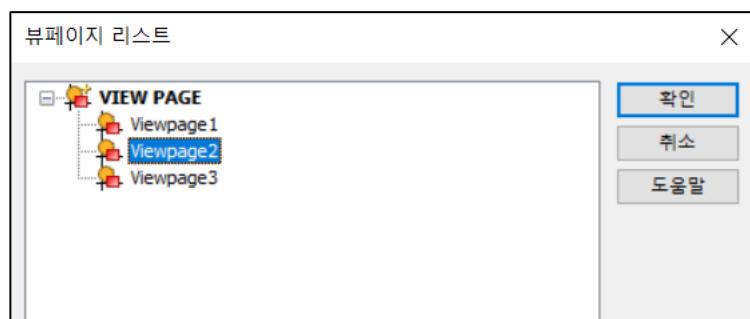
메시지박스의 버튼이 예/아니오, 확인/취소로 설정된 경우 버튼 값을 태그에 저장합니다. 사용자의 선택사항을 시스템 운영에 반영할 때 활용합니다.

5.9.8.3 뷰 페이지 열기

오브젝트 클릭했을 때 뷰 페이지를 열거나, 닫거나, 전환할 수 있도록 설정합니다. 아래와 같이 뷰페이지 선택 옵션은 3가지입니다.



페이지명 아이콘을 누르면 프로젝트에 등록된 뷰 페이지 선택리스트 창이 나타납니다.



가. 페이지 열기

새로운 뷰 페이지 창을 생성하고 뷰 페이지를 불러옵니다.

나. 페이지 닫기

뷰 페이지를 닫습니다. 페이지를 닫아도 백그라운드로 실행되고 있습니다.

다. 페이지 전환

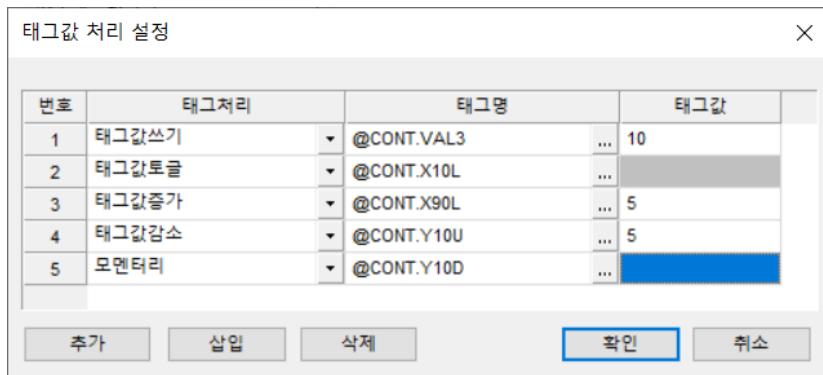
현재 페이지에 새로운 페이지를 팝업형식으로 오픈하는 방식이 아닌 현재위치에 새로운 페이지를 오픈 하는 방식을 의미합니다. 이전 페이지는 닫습니다.

5.9.8.4 태그값 처리

오브젝트 클릭했을 때 태그 값을 내보내도록 설정합니다. 내보내기는 5가지 종류가 있습니다.

태그값 처리 체크하고 **TAG값 처리** 버튼 누르면 설정 창이 나타납니다. 설정창에서 태그처리 콤보박스 누르면 5가지 처리방식이 있습니다. 하나를 선택하고 태그명을 선택 입력합니다. 그리고 내보낼 태그값을 입력합니다. 모멘터리와 토글방식은 특별히 태그 값 입력 할 필요 없습니다.

다음과 같이 태그처리 설정을 5개의 태그처리를 설정하였습니다. 이것은 5개의 태그에 대해 1번부터 순차적으로 태그 값 처리를 합니다.



태그값 쓰기 : 입력된 태그 값을 태그명에 씁니다.

태그값 토글 : 태그 값이 0인면 1로, 1이면 0으로 마우스 클릭할 때 변경합니다.

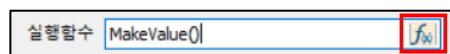
태그값 증가 : 지정한 태그에 설정한 태그 값을 계속 더한 값을 저장합니다.

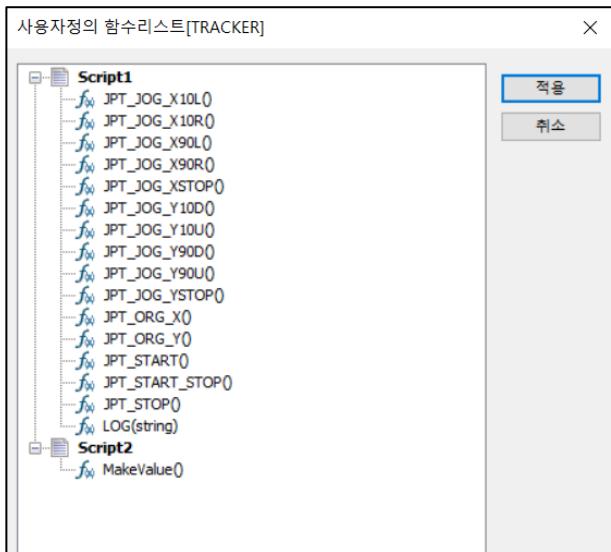
태그값 감소 : 지정한 태그에 설정한 태그 값을 계속 뺀 값을 저장합니다.

모멘터리 : 마우스를 눌렀을 때 1을 내보내고 놓았을 때 0을 내보냅니다.

5.9.8.5 스크립트 실행

사용자가 작성한 스크립트 함수를 마우스 클릭할 때 실행합니다. 실행함수 입력창 우측의 버튼을 클릭을 하면 프로젝트에 등록된 스크립트를 지정하는 함수 리스트 설정창이 열립니다.





함수 리스트에서 원하는 함수를 더블 클릭하면 아래 그림과 같이 페이지명에 선택한 스크립트에 입력됩니다.



입력된 스크립트 함수에 파라메터가 있는 경우는 호출함수에 직접 입력해주어야 합니다. 그렇지 않으면 올바르게 실행되지 않습니다.

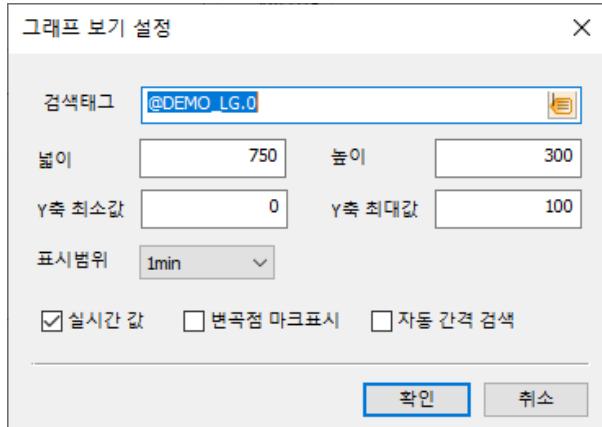
예를 들어서 함수의 원형이 **void Test1(int nValue);** 이렇게 작성된 사용자 함수가 있다면 파라메터 nValue에 해당 값을 써 주어야 합니다. 함수를 선택하면 아래와 같이 파라메터 부분에 파라메터 데이터형이 표시됩니다. 이 부분을 써 주어야 합니다.

→ 예) Test1(**int**) → Test1(**3**)

5.9.8.6 그래프 보기

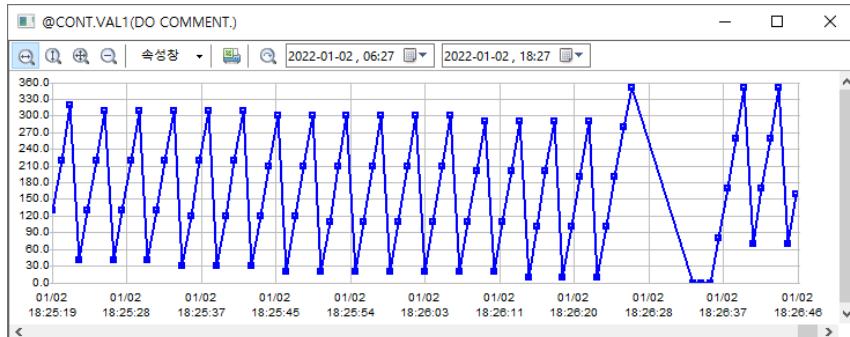
태그값의 실시간 변화값을 그래프로 표시할 때 사용합니다. 그래프 오브젝트와 동일하게 속성창을 클릭해서 여러 조건으로 그래프를 사용할 수 있습니다. 뷰페이지에 다양한 아날로그, 디지털 태그의 변화곡선을 표시할 때 사용하면 편리한 기능입니다.

그래프 보기 체크를 하고 [그래프보기] 버튼을 누르면 다음과 같이 설정 창이 나타납니다.



항목	설명
검색태그	그래프 보기창에 표시할 태그를 설정합니다.
넓이	그래프창의 넓이
높이	그래프창의 높이
Y축 최소값	Y축 범위 최소값
Y축 최대값	Y축 범위 최대값
검색기간	그래프 보기창이 열릴 때 디폴트 검색기간입니다. 현재시간으로부터 몇시간 이전 검색을 하라는 의미입니다.
실시간 값	태그값을 실시간 그래프로 표시합니다.
변곡점 마크표시	그래프의 값이 변하는 지점에 표식을 합니다.
자동 간격검색	검색시간을 단축하기위해 압축검색을 합니다.

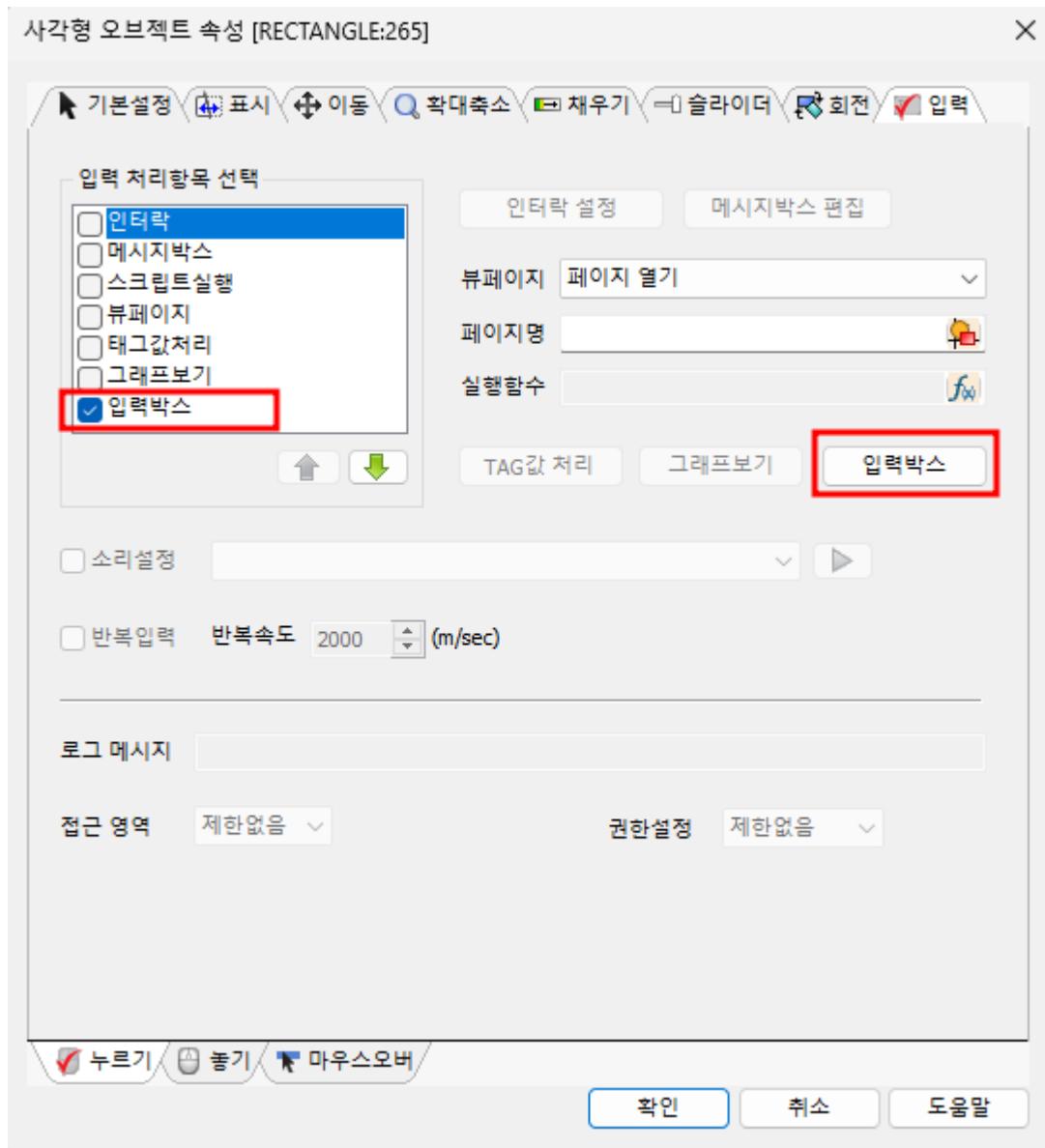
[런타임에 그래프 보기 창이 열린 이미지]



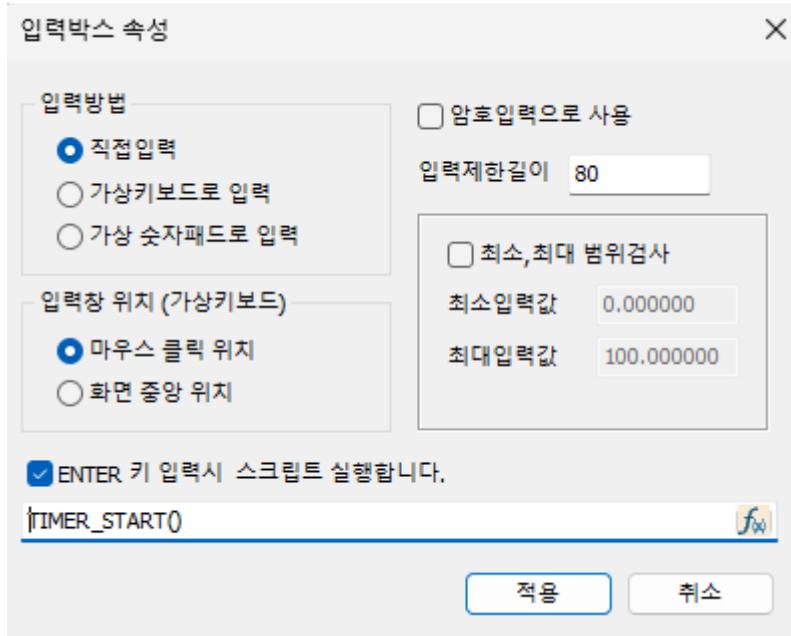
5.9.8.7 입력박스

입력박스는 사용자가 수치 값 또는 문자열 데이터를 입력해야 할 경우 사용합니다. 터치 스크린을 사용하는 경우 키보드 없이 입력이 가능하도록 가상키보드 (숫자전용, 숫자+문자)를 제공합니다.

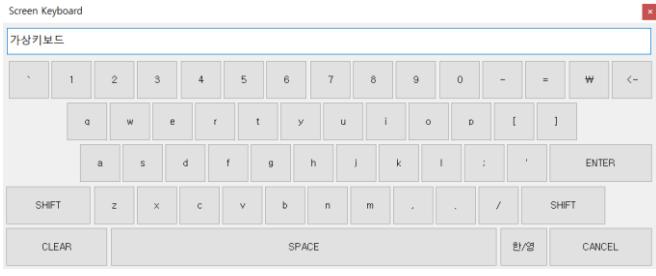
입력박스 항목체크를 하고 [입력박스] 버튼을 누르면 다음과 같이 설정 창이 나타납니다



"입력박스" 버튼을 누르면 아래와 같이 입력박스 속성창이 나타납니다.

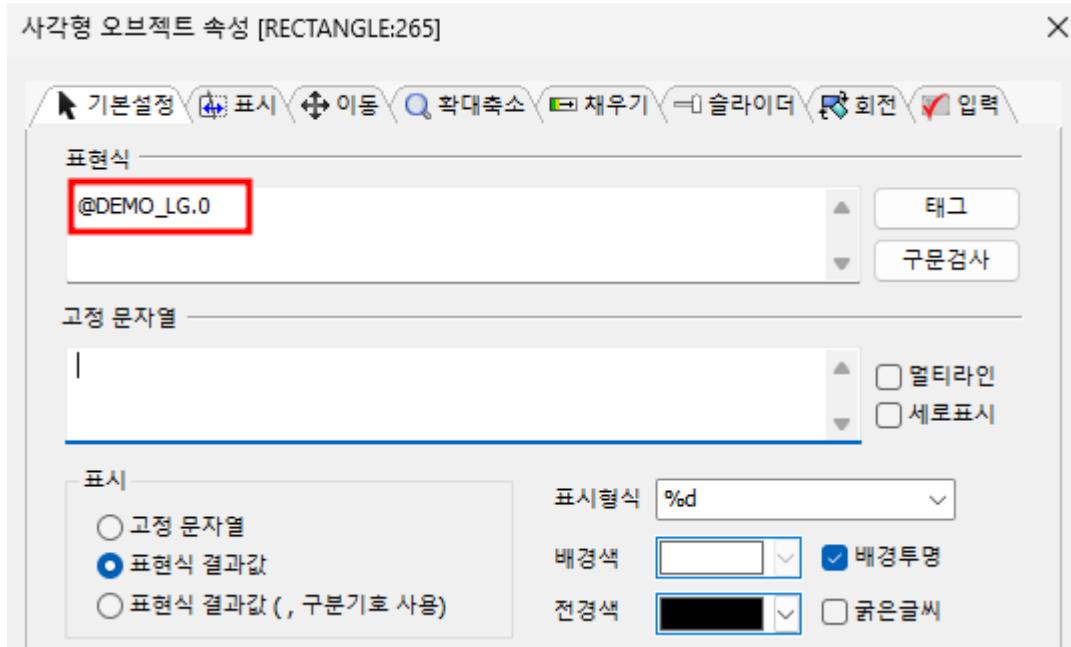


항목	설명
암호입력	암호입력이 필요한 경우 설정합니다. 입력되는 문자에 대해서 화면에는 "*"로 표시됩니다.
최소, 최대 범위검사	입력값에 대한 최소, 최대값 범위 초과여부를 검사합니다. 입력값이 지정한 값을 벗어날 경우 경고창으로 알려줍니다.
입력제한길이	입력 가능한 문자열의 길이를 제한합니다. 초기 설정값은 80입니다.
ENTER키 입력시 스크립트 실행	입력이 끝나고 ENTER 키가 입력되었을 때 실행스크립트 함수를 입력합니다. 브라우저 버튼을 누르면 아래와 같이 정의되어 있는 사용자 스크립트 함수목록이 나타납니다.
	<p>사용자정의 함수리스트[SARCO]</p> <p>COMMON</p> <ul style="list-style-type: none"> f_w ChkPassword0 f_w ChkPassword20 f_w ChkPassword30 f_w InitPwd0 f_w initReceptor0 f_w main0 f_w OpenPagePrice0 f_w OpenPagePrice20 <p>보고서</p> <ul style="list-style-type: none"> f_w GetLastDay(int) f_w GetTagGroup(int) f_w InitReportAddCombo() f_w InitReportPage() f_w ReportAutoManFlagReset() f_w ReportDaily(string) f_w ReportDaily2(string) f_w ReportDailyAuto() f_w ReportMonthly(string) f_w ReportMonthly2(string) f_w ReportMonthlyAuto() f_w ReportPriceAuto() f_w ReportTotalDaily(string) f_w ReportTotalMonthly(string) f_w ReportTotalYearly(string) f_w ReportYearly(string) f_w ReportYearlyAuto() f_w RepOut() f_w SetPriceTag(int) <p>서브페이지</p> <p>프로젝트내에서 사용자가 정의한 스크립트 함수리스트입니다.</p>

입력방법 선택	<p>실물 키보드 없이 입력을 할경우 선택을 할 수 있도록 합니다. 기본은 직접입력이지만 가상 키보드, 가상 숫자패드로 입력이 있습니다.</p> <p>[가상 숫자패드]</p> 
입력창 위치	<p>[가상 키보드]</p>  <p>가상키보드로 설정할 경우 가상키보드의 화면표시위치를 지정합니다.</p>

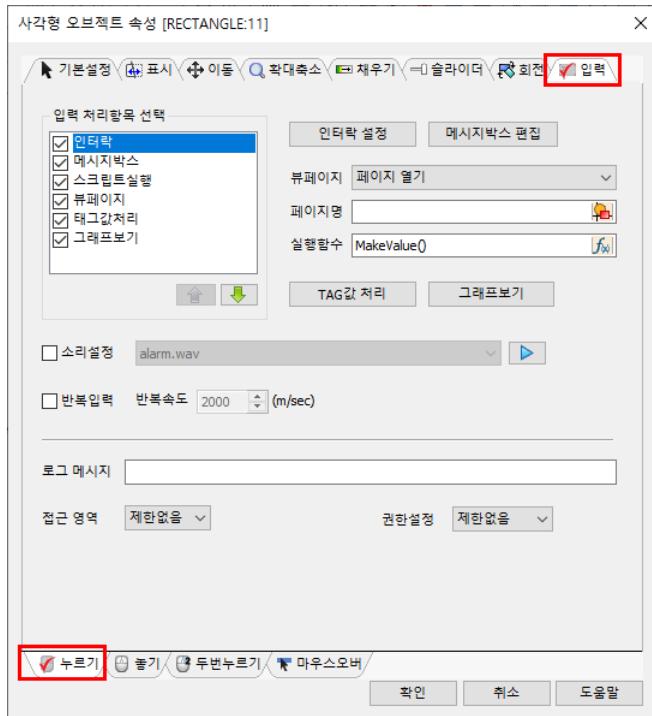
입력값은 오브젝트 기본설정의 표현식에 연결된 태그에 값을 씁니다.

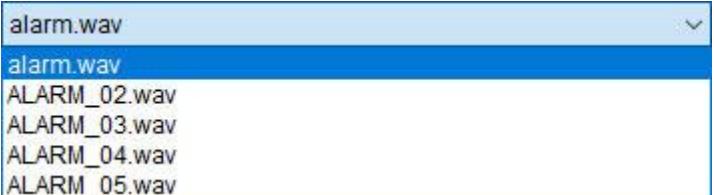
[주의] 표현식에는 @붙여서 태그이름만 넣어주어야 합니다.



5.9.8.8 누르기

마우스 왼쪽 버튼이 눌려지면 입력 처리되는 누르기 속성 설정창입니다. 입력 속성에서 누르기, 놓기, 두번 누르기는 마우스 버튼 클릭 이벤트만 다르고 동작방식은 동일합니다. 그래서 별도로 놓기 두번누르기 설명은 생략하겠습니다. 그리고 앞에서 입력 처리항목 6가지에 대한설명은 하였으니, 입력 – 누르기 속성창을 띄우고 그 다음 부분 설정내용을 설명하겠습니다.



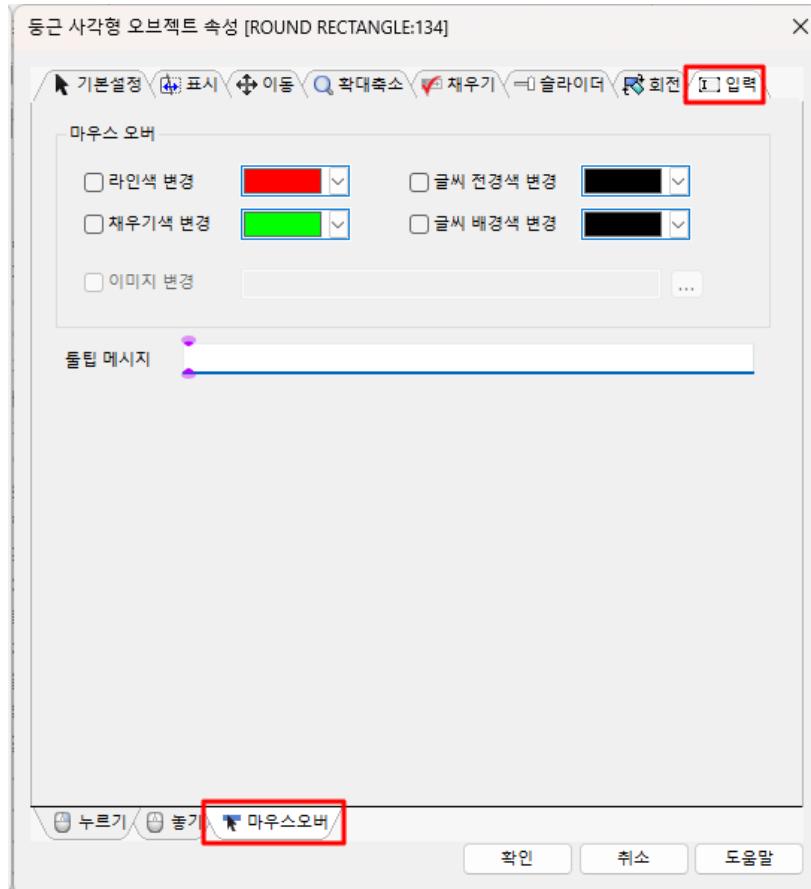
항목	설명
소리설정	<p>누르기 할 때 누르기 음을 설정합니다. 환경설정에서 설정된 음원이 리스트에 표시되고 선택합니다. 음원파일은 WAV 형식입니다.</p> 
반복입력	<p>마우스가 눌림 상태에 있는 동안 반복적으로 입력처리 여부를 설정합니다. 반복속도 : 반복 입력에 대한 INTERVAL을 설정합니다. 기본은 2초. 단위는(ms) 예시) 입력동작설정 → 태그값 쓰기, 태그이름 → "Demo.출력값1", 태그값 → 10 반복 → 사용, 반복속도 → 2000 마우스의 왼쪽버튼이 Down되는 시점부터 UP이 될 때까지 2초마다 "Demo.출력값1"태그에 10을 씁니다. 마우스의 왼쪽버튼이 Up이 되면 반복입력은 해제됩니다. ※ 이 기능은 입력 속성의 누르기에만 적용됩니다.</p>
로그메시지	로그파일에 저장할 입력관련 로그메시지를 설정합니다. 로그메시지는 %배포폴더\Log%RunnerLog 폴더에 저장됩니다.
접근영역	설정된 객체 영역은 사용자 설정에서 설정한 접근 영역과 동일하며, 사용자에 따라 입력을 제한하는 기능입니다.
권한설정	관리자의 경우 관리자만 입력을 설정할 수 있으며 사용자의 경우 객체영역에 해당되는 사용자만이 입력설정이 가능합니다.

5.9.8.9 마우스 오버

마우스오버는 마우스 커서가 오브젝트 위에 있을 때 글씨색, 배경색, 라인색, 이미지등을 바꾸어서 보여지도록 하는 기능입니다.

항목	설명
라인색 변경	도형의 라인색을 지정한 색상으로 변경합니다.
채우기색 변경	해당 도형의 채우기색을 지정한 색상으로 변경합니다.
글씨 전경색 변경	해당 도형내의 글씨색을 지정한 색상으로 변경합니다.

글씨 배경색 변경	해당 도형내의 글씨배경색을 지정한 색상으로 변경합니다.
이미지 변경	해당 이미지를 지정한 이미지로 변경합니다. ※ 이미지, 비트맵 오브젝트에만 해당
반복입력	사용안함

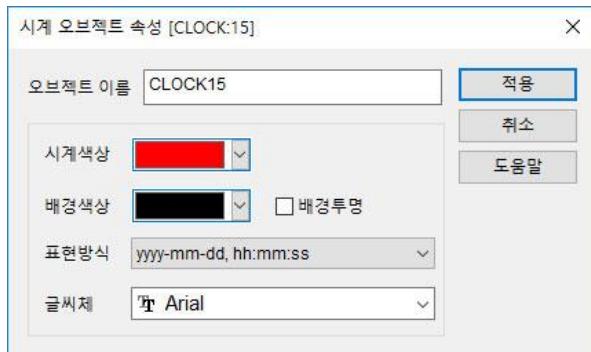


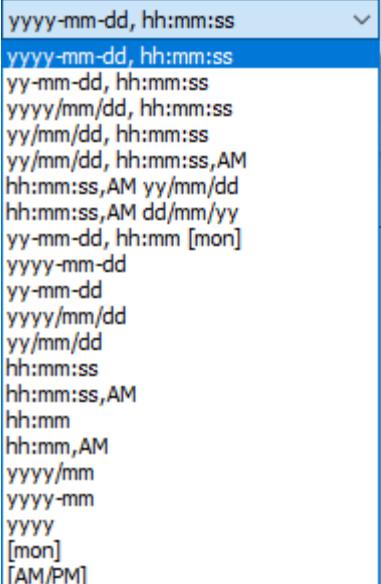
5.10 시계 오브젝트

현재 시스템 시간을 설정된 형식으로 표시하는 오브젝트입니다.

2013-08-14, 15:41:58

5.10.2 속성설정



항목	설명
오브젝트 이름	오브젝트의 이름을 설정합니다.
시계색상	전경 텍스트 색상을 설정합니다.
배경색상	텍스트 배경색상을 설정합니다. 배경투명여부설정이 안되어 있는 경우 설정할 수 있습니다.
배경투명	배경 투명여부를 설정합니다. 배경투명여부를 사용할 경우 배경색상은 비활성화가 됩니다.
표현방식	시간 표시 형식을 설정합니다. 콤보박스를 클릭하면 아래와 같은 이미지가 표시됩니다. 

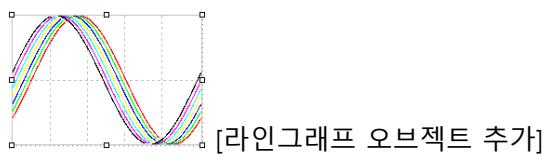
글씨체	글씨체를 선택합니다. 글씨체 크기는 오브젝트 영역 크기에 맞추어 자동조정됩니다.

표현방식은 총 21가지 방식이 있습니다.

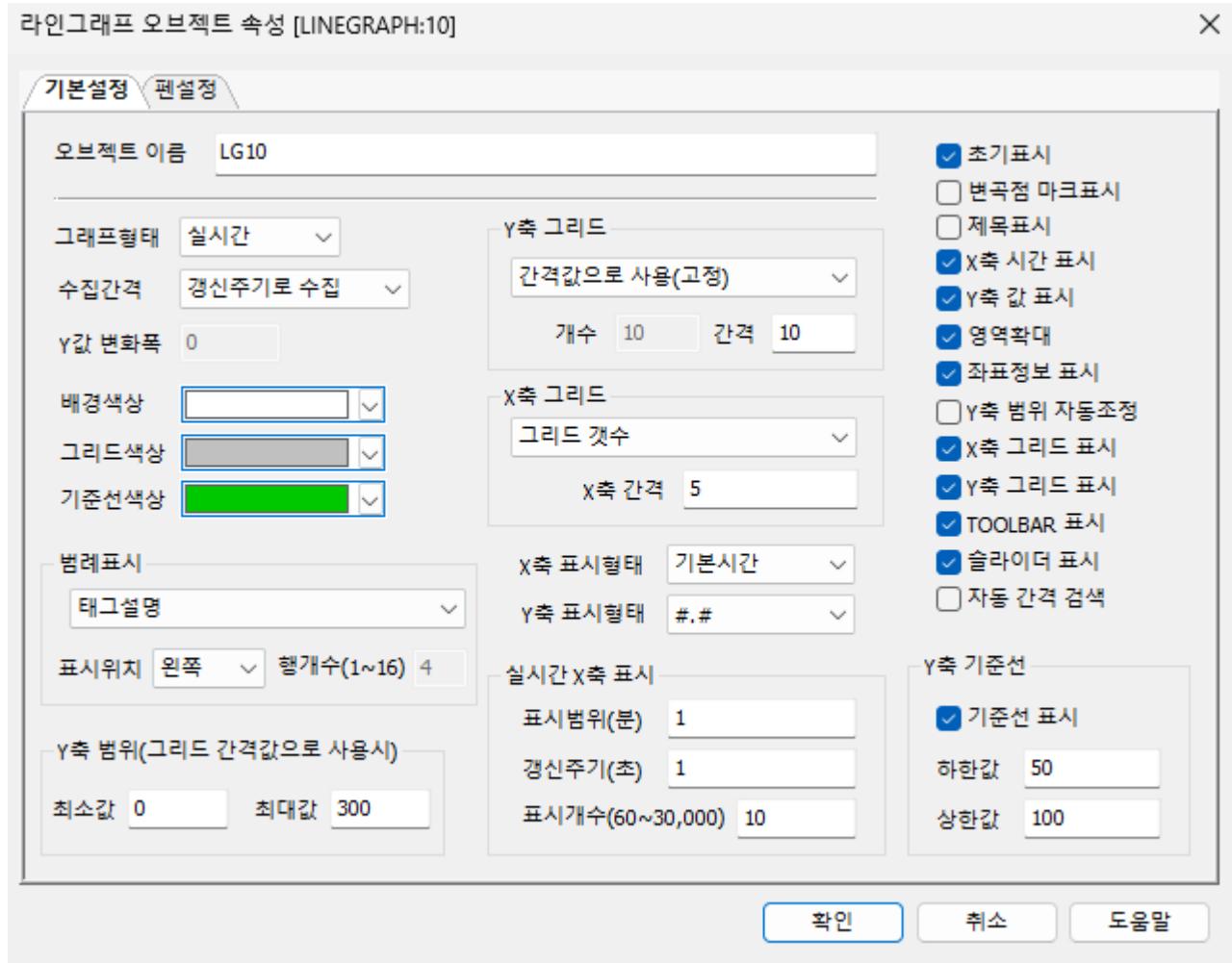
표현방식	예시
yyyy-mm-dd, hh:mm:ss	2008-04-11, 15:05:37
yy-mm-dd, hh:mm:ss	08-04-11, 15:05:37
yyyy/mm/dd, hh:mm:ss	2008/04/11, 15:05:37
yy/mm/dd, hh:mm:ss	08/04/11, 15:05:37
yy/mm/dd, hh:mm:ss, AM	08/04/11, 03:05:37, PM
hh:mm:ss, AM yy/mm/dd	03:05:37, 오후 08/04/11
hh:mm:ss, AM dd/mm/yy	03:05:37, 오후 11/04/08
yy-mm-dd, hh:mm [mon]	08-04-11, 15:05 [FRI]
yyyy-mm-dd	2008-04-11
yy-mm-dd	08-04-11
yyyy/mm/dd	2008/04/11
yy/mm/dd	08/04/11
hh:mm:ss	15:05:37
hh:mm:ss, AM	03:05:37, PM
hh:mm	15:05
hh:mm, AM	03:05, PM
yyyy/mm	2008/10
yyyy-mm	2008-10
yyyy	2008
[mon]	화
[AM/PM]	오후

5.11 라인그래프 오브젝트

라인그래프는 시간에 따라 저장된 태그값의 트랜드를 시각적으로 쉽게 파악하는데 사용합니다. 리얼타임, 히스토리 그래프로 구분되며 다양한 선택사항을 지원하기 때문에 편리합니다. 그래프 데이터를 CSV 파일포맷 출력 지원하며 MS-EXCEL파일의 XLS포맷으로 그래프 이미지와, 데이터를 파일에 저장 가능합니다.

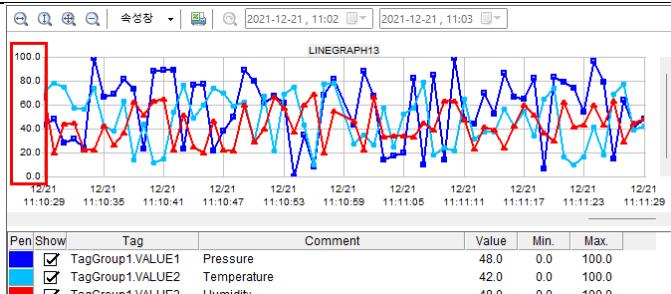


5.11.1. 기본설정 속성



항목	설명
오브젝트 이름	오브젝트의 이름을 설정합니다. ※ 라인그래프 오브젝트는 스크립트에서 제어를 할 수 있기 때문에 뷰페이지 내에서 유일한 이름이어야 합니다.
그래프 형태	실시간 : 태그 현재값 표시그래프 히스토리 : 태그 과거데이터 표시그래프
수집간격	실시간 그래프에서 데이터 표시방식을 선택합니다. 갱신주기로 수집할 경우에는 표시범위 및 갱신주기 설정값에 제한이 있습니다. (표시

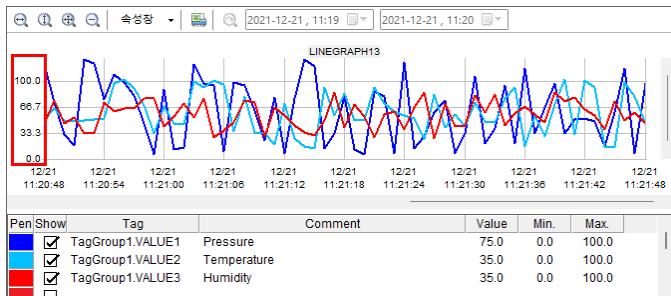
	<p>범위 * 60초)</p> <p>갱신주기로 : [갱신주기]의 설정값으로 그래프 데이터를 갱신합니다.</p> <p>화면비율로 : 그래프 오브젝트 표시영역 비율에 따라 갱신합니다.</p> <p>변화폭에 따라서 : 이전값과 현재값 차이가 설정된 Y폭값 이상일 때.</p>																								
Y값 변화폭	실시간 그래프 오브젝트에 대하여 수집간격을 변화폭에 따라 표시합니다. 태그값의 변화폭을 지정해서																								
색상	<p>배경색상 : 그래프 표시영역의 배경색</p> <p>그리드 : 격자 라인의 색</p> <p>기준선 : Y축 기준값 표시라인 색</p>																								
범례표시	<ul style="list-style-type: none"> ■ 태그명으로 표시 태그속성에 등록된 태그이름으로 표시합니다. ■ 태그설명으로 표시 태그속성에 등록된 설명 텍스트를 표시합니다. (직관적임) ■ 태그명, 태그설명, 현재값으로 표시 <p>※ 실시간 그래프 오브젝트에서만 사용 가능합니다.</p> <table border="1"> <thead> <tr> <th>Pen>Show</th> <th>Tag</th> <th>Comment</th> <th>Value</th> <th>Min.</th> <th>Max.</th> </tr> </thead> <tbody> <tr> <td><input checked="" type="checkbox"/></td> <td>TagGroup1.VALUE1</td> <td>Pressure</td> <td>20.0</td> <td>0.0</td> <td>100.0</td> </tr> <tr> <td><input checked="" type="checkbox"/></td> <td>TagGroup1.VALUE2</td> <td>Temperature</td> <td>19.0</td> <td>0.0</td> <td>100.0</td> </tr> <tr> <td><input checked="" type="checkbox"/></td> <td>TagGroup1.VALUE3</td> <td>Humidity</td> <td>21.0</td> <td>0.0</td> <td>100.0</td> </tr> </tbody> </table>	Pen>Show	Tag	Comment	Value	Min.	Max.	<input checked="" type="checkbox"/>	TagGroup1.VALUE1	Pressure	20.0	0.0	100.0	<input checked="" type="checkbox"/>	TagGroup1.VALUE2	Temperature	19.0	0.0	100.0	<input checked="" type="checkbox"/>	TagGroup1.VALUE3	Humidity	21.0	0.0	100.0
Pen>Show	Tag	Comment	Value	Min.	Max.																				
<input checked="" type="checkbox"/>	TagGroup1.VALUE1	Pressure	20.0	0.0	100.0																				
<input checked="" type="checkbox"/>	TagGroup1.VALUE2	Temperature	19.0	0.0	100.0																				
<input checked="" type="checkbox"/>	TagGroup1.VALUE3	Humidity	21.0	0.0	100.0																				
Y축 그리드	<ul style="list-style-type: none"> ■ 그리드 개수로 사용 등록된 태그의 최소,최대값의 범위가 다른 경우 Y축 표시값은 일련번호로 보이고 펜(태그)의 값은 각각의 최소, 최대값으로 표현됩니다. 이 옵션은 태그값의 트랜스 참고용으로 표시할 때 사용합니다. <table border="1"> <thead> <tr> <th>Pen>Show</th> <th>Tag</th> <th>Comment</th> <th>Value</th> <th>Min.</th> <th>Max.</th> </tr> </thead> <tbody> <tr> <td><input checked="" type="checkbox"/></td> <td>TagGroup1.VALUE1</td> <td>Pressure</td> <td>20.0</td> <td>0.0</td> <td>100.0</td> </tr> <tr> <td><input checked="" type="checkbox"/></td> <td>TagGroup1.VALUE2</td> <td>Temperature</td> <td>19.0</td> <td>0.0</td> <td>100.0</td> </tr> <tr> <td><input checked="" type="checkbox"/></td> <td>TagGroup1.VALUE3</td> <td>Humidity</td> <td>21.0</td> <td>0.0</td> <td>100.0</td> </tr> </tbody> </table> <ul style="list-style-type: none"> ■ 고정 간격값으로 사용 설정된 Y축 그리드 표시간격 값으로 표시합니다. 	Pen>Show	Tag	Comment	Value	Min.	Max.	<input checked="" type="checkbox"/>	TagGroup1.VALUE1	Pressure	20.0	0.0	100.0	<input checked="" type="checkbox"/>	TagGroup1.VALUE2	Temperature	19.0	0.0	100.0	<input checked="" type="checkbox"/>	TagGroup1.VALUE3	Humidity	21.0	0.0	100.0
Pen>Show	Tag	Comment	Value	Min.	Max.																				
<input checked="" type="checkbox"/>	TagGroup1.VALUE1	Pressure	20.0	0.0	100.0																				
<input checked="" type="checkbox"/>	TagGroup1.VALUE2	Temperature	19.0	0.0	100.0																				
<input checked="" type="checkbox"/>	TagGroup1.VALUE3	Humidity	21.0	0.0	100.0																				



■ 비율 간격값으로 사용

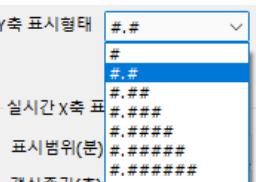
(최대값-최소값) / Y축 그리드간격 결과값의 비율 간격으로 표시합니다.

예) 최대값 100, Y축 그리드간격 값 5 일경우 아래와 같습니다.



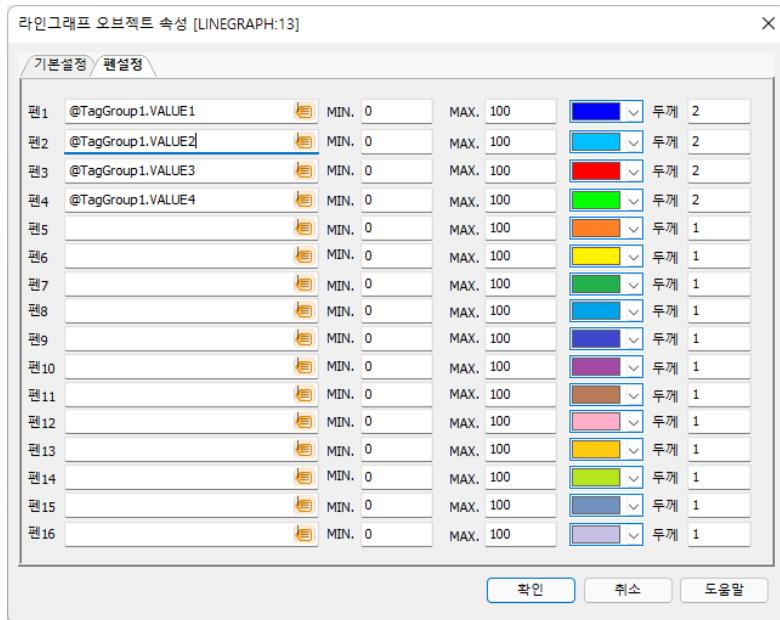
■ 개수 : 그리드 개수로 표시할 경우 라인개수를 입력합니다. (기본 5)

■ 간격 : 그리드 간격으로 표시할 때 간격값입니다.

X축 그리드	<ul style="list-style-type: none"> ■ 그리드 개수 : X축의 값의 표시개수 지정 (MAX 10) ■ 시간 간격값 : 시작시간부터 종료시간까지 시간간격으로 표시선택.
X축 표시형태	<p>기본시간 : 현재 시스템 시간을 년월일,시분초 형태로 표시합니다.</p> <p>경과시간 : 프로젝트 실행 시작된 뒤로 경과된 시간으로 표시합니다.</p>
Y축 표시형태	<p>Y축 데이터 표시값 형태를 선택합니다.</p> 
Y축 범위(최소,최대)	[Y축 그리드] 설정항목의 그리드 간격값(고정, 비율)으로 표시할 경우 설정합니다.
실시간 X축 표시	<ul style="list-style-type: none"> ■ 표시범위 몇분동안의 데이터를 화면 스크롤없이 표시할 것인지 설정합니다. ■ 갱신주기 화면 업데이트 주기로 초단위입니다. 숫자가 적을수록 빠르게 업데이트 합니다.
Y축 기준선	<p>Y축 기준선(참고라인)을 지정된 색으로 표시합니다.</p> <p>Y축 그리드 표시가 그리드 개수가 아닌 경우 사용합니다.</p>
초기표시	프로젝트 실행할 때 오브젝트 표시여부
변곡점 마크표시	태그값의 변화지점에 마크표시

제목표시	그래프 오브젝트 이름 상단에 표시
X축 시간표시	X축 눈금 시간값 표시.
Y축 값 표시	Y축 눈금 값 표시.
영역확대	실시간, 히스토리 그래프 영역을 마우스로 상,하,좌,우 확대합니다.
좌표정보 표시	마우스 이동시 툴팁으로 포인트 좌표에 대한 시간, 측정값 표시
Y축 범위 자동조정	그래프 데이터 값에 따라 Y축 범위 자동으로 변경합니다.
X축 그리드 표시	X축 눈금에 따라 라인표시.
Y축 그리드 표시	Y축 눈금에 따라 라인표시.
TOOLBAR 표시	그래프 오브젝트 도구바 표시.
슬라이더 표시	좌우, 상하 스크롤 슬라이더 표시합니다.
자동간격 검색	히스토리 데이터 검색할 때 빠른 검색을 하도록 화면 그래프 영역의 크기에 따라 요약 검색을 합니다.

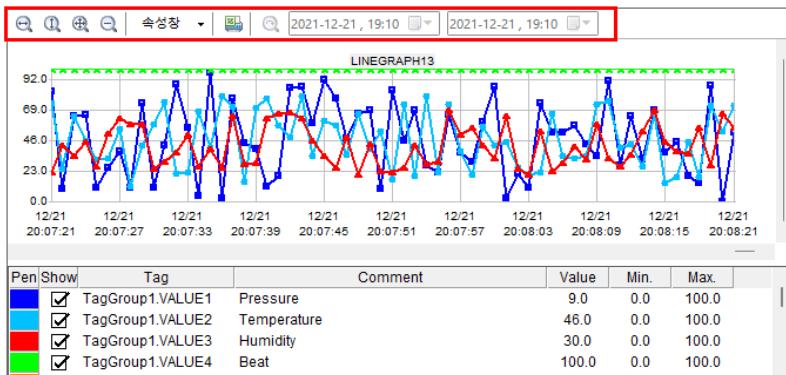
5.12.2 펜설정 속성



항목	설명
펜1~16	<p>펜은 태그명을 최대 16개까지 설정합니다.</p> <p>각각의 펜에 대한 최소, 최대값, 라인색, 라인두께를 설정합니다.</p> <p>최소, 최대값은 기본속성에서 [Y축 그리드 개수]로 표시할 경우에 의미 있게 사용되며, 그렇지 않을 경우 무시됩니다.</p>

5.12.3 프로젝트 실행시 도구바

그래프 오브젝트 속성에서 TOOLBAR 표시선택이 되어있는 경우 아래와 같이 오브젝트 상단에 나타납니다.

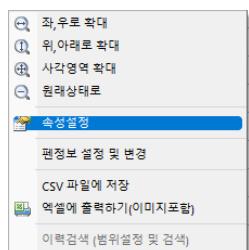


항목	설명
확대, 축소버튼	좌, 우, 상, 하 영역을 마우스로 확대합니다. 휠 부분을 조작하면 편리하게 확대, 축소 기능을 이용할 수 있습니다.
속성창	도구바에서 속성창 클릭하면 아래와 같이 3가지 선택메뉴가 나타납니다. <ul style="list-style-type: none"> ■ 속성설정 스튜디오에서 설정된 속성을 프로젝트 실행 중에도 변경이 가능합니다. ■ 펜정보 설정 및 변경 펜을 변경하고 색, 두께를 변경합니다. ■ CSV파일에 저장 보여지는 그래프를 CSV 파일포맷으로 저장합니다.

5.12.4 프로젝트 실행시 팝업메뉴

그래프 오브젝트 영역에서 마우스 우클릭 하면 팝업메뉴가 나타납니다.

단, 속성에서 TOOLBAR 표시선택이 되어있지 않은 경우 해당합니다.



도구바에 있는 메뉴와 동일합니다. 다만, 이력검색(범위설정 및 검색) 메뉴 하나가 더 있습니다. 이 메뉴는 [일반 오브젝트]-[입력] 속성에 [그래프 표시] 기능을 사용할 때 활성화됩니다. 보다 자세한 사항은 일반 오브젝트 입력 부분 설명을 참고하시기 바랍니다.

5.12 그리드 오브젝트

그리드는 엑셀의 시트처럼 사용자 입력이 가능한 시트를 말합니다.

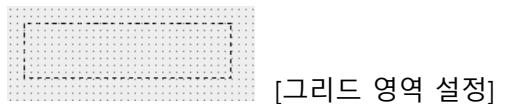
그리드 오브젝트를 이용하여 데이터베이스에 저장된 데이터를 조회할 수 있으며, 사용자가 데이터를 설정할 수 있는 다양한 양식품을 만들 경우 사용합니다.

주로 공정 레시피 작성을 해야하는 경우 활용할 수 있습니다.

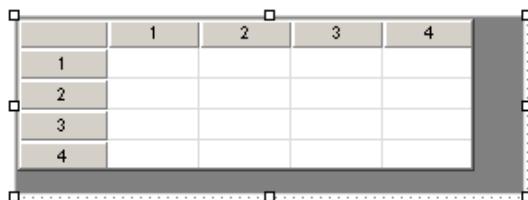
5.13.1 오브젝트 추가

그리드 오브젝트를 추가하기 위해서는 도구 툴바에서  버튼을 클릭하거나 [메뉴] -[오브젝트]-[그리드]를 선택하면 마우스 커서가 +로 변경됩니다.
키보드의 ESC 버튼을 누르면 그리기가 취소됩니다.

마우스가 드래그 되는 동안은 아래 그림과 같이 실선 사각형으로 그려집니다.

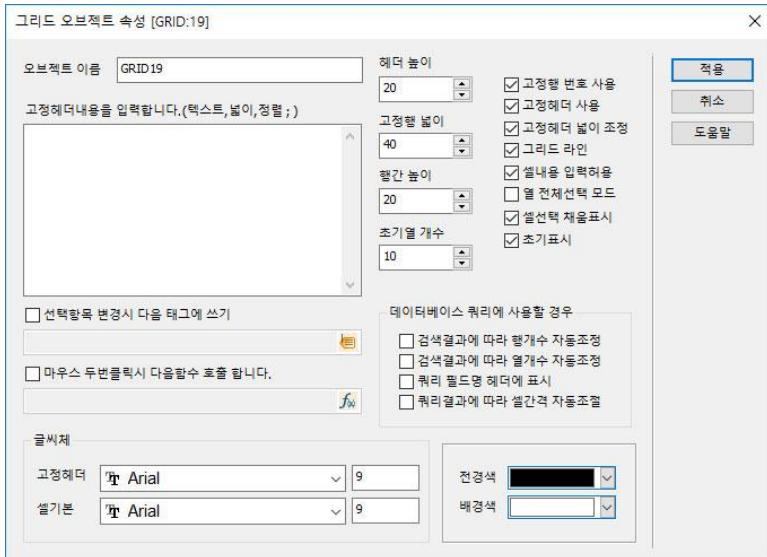


원하는 끝 위치에서 클릭하고 있던 마우스 왼쪽 버튼을 놓으면 아래 그림과 그리드 오브젝트가 추가되면서 표시됩니다.

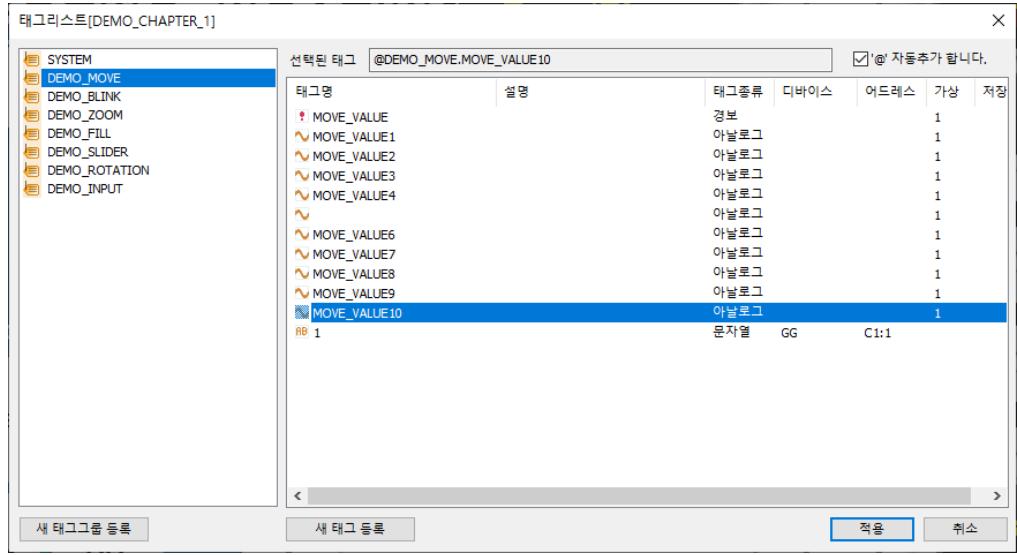


5.13.2 속성설정

그리드 속성 설정창은 다음과 같습니다.



항목	설명
오브젝트 이름	오브젝트의 이름을 설정합니다. ※ 그리드 오브젝트는 스크립트에서 사용자가 제어를 할 수 있는 오브젝트이기 때문에 다른 어떤 오브젝트의 이름과 동일해서는 안됩니다. 그리드 오브젝트 속성에서의 설정된 사항은 프로젝트 실행하여 확인할 수 있습니다.
고정헤더내용 입력	고정헤더 사용시 고정헤더의 내용을 입력 합니다. 헤더의 내용은 <u>텍스트(제목)</u> , <u>넓이</u> , <u>정렬</u> ; 순으로 입력합니다. 정렬은 0 : 왼쪽, 1 : 중앙, 2 : 오른쪽 정렬을 의미합니다.
헤더 높이	그리드의 헤더 높이를 설정합니다. (픽셀단위)
고정행 넓이	그리드의 고정행 넓이를 설정합니다. (픽셀단위)
행간 높이	그리드 행간의 높이를 설정합니다. (픽셀단위)
초기열 개수	그리드의 초기열 개수를 설정합니다.
고정행 번호사용	고정행 번호 사용 여부를 설정합니다. (1,2,3,...)
고정헤더 사용	고정헤더 사용여부(컬럼명)
고정헤더 넓이 조정	프로젝트 실행시 고정헤더의 넓이를 사용자에 의한 조정 여부를 설정합니다.

그리드 라인	셀보더 표시																																																																																				
셀 내용 입력허용	키보드 입력 여부																																																																																				
열전체 선택모드	열 전체를 선택																																																																																				
셀선택 채움표시	셀 보더만 표시할것인지 선택채움을 표시할 것인지 여부																																																																																				
초기표시	프로젝트 실행할 때 오브젝트 표시여부																																																																																				
	프로젝트 실행할 때 그리드의 셀 선택 항목이 변경되면 해당 셀의 값을 연결된 태그에 씁니다. 태그는 문자열 태그인 경우 올바르게 처리됩니다																																																																																				
선택항목 변경시 다음태그에 쓰기	<p>체크 박스에 체크 후 [태그]  버튼을 클릭하면 아래와 같은 태그 리스트가 표시됩니다.</p>  <table border="1"> <thead> <tr> <th>태그명</th> <th>설명</th> <th>태그종류</th> <th>디바이스</th> <th>어드레스</th> <th>가상</th> <th>저장</th> </tr> </thead> <tbody> <tr> <td>MOVE_VALUE</td> <td>경보</td> <td>아날로그</td> <td></td> <td>1</td> <td></td> <td></td> </tr> <tr> <td>MOVE_VALUE1</td> <td>아날로그</td> <td>아날로그</td> <td></td> <td>1</td> <td></td> <td></td> </tr> <tr> <td>MOVE_VALUE2</td> <td>아날로그</td> <td>아날로그</td> <td></td> <td>1</td> <td></td> <td></td> </tr> <tr> <td>MOVE_VALUE3</td> <td>아날로그</td> <td>아날로그</td> <td></td> <td>1</td> <td></td> <td></td> </tr> <tr> <td>MOVE_VALUE4</td> <td>아날로그</td> <td>아날로그</td> <td></td> <td>1</td> <td></td> <td></td> </tr> <tr> <td>MOVE_VALUE5</td> <td>아날로그</td> <td>아날로그</td> <td></td> <td>1</td> <td></td> <td></td> </tr> <tr> <td>MOVE_VALUE6</td> <td>아날로그</td> <td>아날로그</td> <td></td> <td>1</td> <td></td> <td></td> </tr> <tr> <td>MOVE_VALUE7</td> <td>아날로그</td> <td>아날로그</td> <td></td> <td>1</td> <td></td> <td></td> </tr> <tr> <td>MOVE_VALUE8</td> <td>아날로그</td> <td>아날로그</td> <td></td> <td>1</td> <td></td> <td></td> </tr> <tr> <td>MOVE_VALUE9</td> <td>아날로그</td> <td>아날로그</td> <td></td> <td>1</td> <td></td> <td></td> </tr> <tr> <td>MOVE_VALUE10</td> <td>아날로그</td> <td>아날로그</td> <td>GG</td> <td>C1:1</td> <td></td> <td></td> </tr> </tbody> </table> <p>왼쪽부분에는 프로젝트에 등록된 태그그룹 리스트가 표시되고, 항목이 선택되면 우측에 해당 태그그룹에 등록되어있는 태그리스트가 표시됩니다. 우측 리스트에서 마우스 더블클릭을 하거나, 하단부의 [적용]버튼을 눌러서 태그를 선택합니다. '@' 자동 추가합니다. 항목은 태그명을 입력할 때 '@'문자를 자동으로 삽입여부를 설정합니다.</p>	태그명	설명	태그종류	디바이스	어드레스	가상	저장	MOVE_VALUE	경보	아날로그		1			MOVE_VALUE1	아날로그	아날로그		1			MOVE_VALUE2	아날로그	아날로그		1			MOVE_VALUE3	아날로그	아날로그		1			MOVE_VALUE4	아날로그	아날로그		1			MOVE_VALUE5	아날로그	아날로그		1			MOVE_VALUE6	아날로그	아날로그		1			MOVE_VALUE7	아날로그	아날로그		1			MOVE_VALUE8	아날로그	아날로그		1			MOVE_VALUE9	아날로그	아날로그		1			MOVE_VALUE10	아날로그	아날로그	GG	C1:1		
태그명	설명	태그종류	디바이스	어드레스	가상	저장																																																																															
MOVE_VALUE	경보	아날로그		1																																																																																	
MOVE_VALUE1	아날로그	아날로그		1																																																																																	
MOVE_VALUE2	아날로그	아날로그		1																																																																																	
MOVE_VALUE3	아날로그	아날로그		1																																																																																	
MOVE_VALUE4	아날로그	아날로그		1																																																																																	
MOVE_VALUE5	아날로그	아날로그		1																																																																																	
MOVE_VALUE6	아날로그	아날로그		1																																																																																	
MOVE_VALUE7	아날로그	아날로그		1																																																																																	
MOVE_VALUE8	아날로그	아날로그		1																																																																																	
MOVE_VALUE9	아날로그	아날로그		1																																																																																	
MOVE_VALUE10	아날로그	아날로그	GG	C1:1																																																																																	
마우스 두번 클릭 시 함수호출	마우스로 셀을 더블 클릭했을 때 사용자가 작성한 함수를 호출해서 실행합니다. 함수이름 우측의  버튼을 클릭을 하면 프로젝트에 등록된 스크립트 함수목록이 나타나고 함수를 선택합니다.																																																																																				

	<p>프로젝트내에서 사용자가 정의한 스크립트 할수리스트입니다.</p>
데이터베이스 쿼리에 사용할 경우	<p>그리드 오브젝트를 SQL 데이터베이스 쿼리결과 표시창으로 사용할 경우 해당합니다.</p> <ul style="list-style-type: none"> ■ 검색결과에 따라 행개수 자동조정 <ul style="list-style-type: none"> - 검색결과의 필드개수만큼 행개수를 자동 조절합니다. ■ 검색결과에 따라 열개수 자동조정 <ul style="list-style-type: none"> - 검색결과의 필드개수만큼 열개수를 자동조절합니다. ■ 쿼리 필드명 헤더에 표시 <ul style="list-style-type: none"> - 검색결과의 쿼리 필드명을 그리드 고정열에 표시합니다 ■ 쿼리결과에 따라 셀간격 자동조절 <ul style="list-style-type: none"> - 셀간격을 자동으로 조절합니다.
글씨체	<ul style="list-style-type: none"> ■ 고정헤더 : 고정헤더의 폰트와 글자크기를 설정합니다. ■ 셀기본 : 셀기본의 폰트와 글자크기를 설정합니다.
전경색	글씨체의 폰트색깔을 설정합니다.
배경색	그리드의 배경색을 설정합니다.

5.13 경보 오브젝트

경보태그로 설정된 태그의 현재 값이 LO, LOLO, HI, HIHI 아날로그 기준치를 넘어가거나, ON/OFF 접점의 경보조건에 해당이 될 경우 표시하기 위한 윈도우 오브젝트입니다.

경보 오브젝트는 현재경보와 과거 경보이력을 검색해서 표시할 수 있습니다.

현재경보 오브젝트는 뷰페이지 어디에 있어도 동일한 내용이 표시됩니다.

또한 도구바를 지원하는데, 이력검색이나, 현재<->과거 경보이력 모드로 전환이 가능합니다.

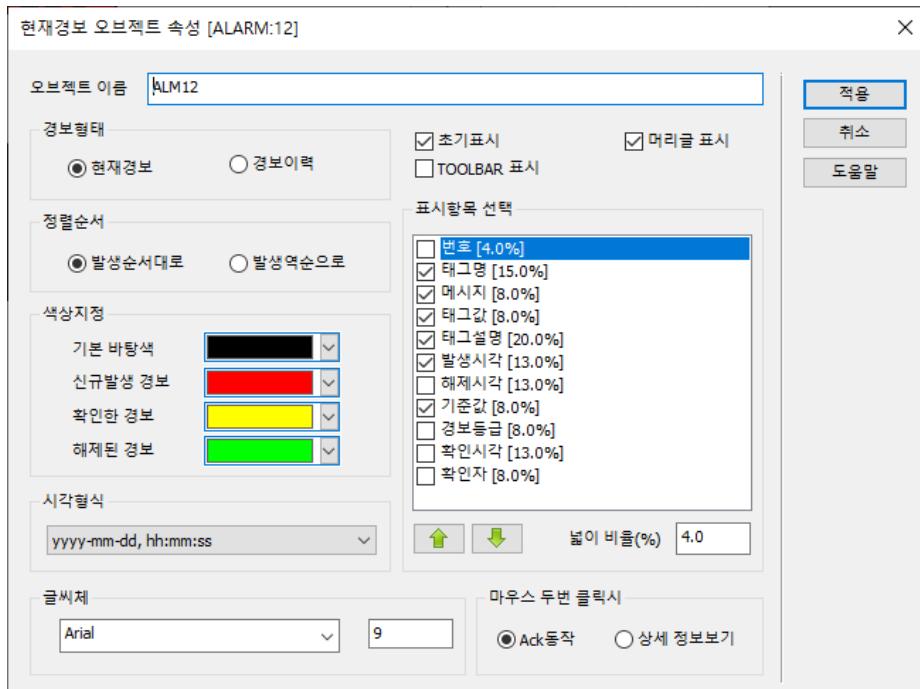
5.14.1 오브젝트 추가

현재경보 오브젝트를 추가하기 위해서는 뷰 도구툴바에서  버튼을 클릭하거나 [메뉴]-[오브젝트]-[현재경보]를 선택하면 마우스 커서가 +로 변경됩니다.
키보드의 ESC 버튼을 누르면 그리기가 취소됩니다.

태그명	메시지	태그값	태그설명	발생시각	기준값	
system,run1	OFF	0	:system run status ..	-12-17, 17:2	0	
system,run1	OFF	0	:system run status ..	-12-17, 17:2	0	
system,run1	OFF	0	:system run status ..	-12-17, 17:2	0	

5.14.2 속성설정

경보 오브젝트 속성설정 창은 다음과 같습니다.

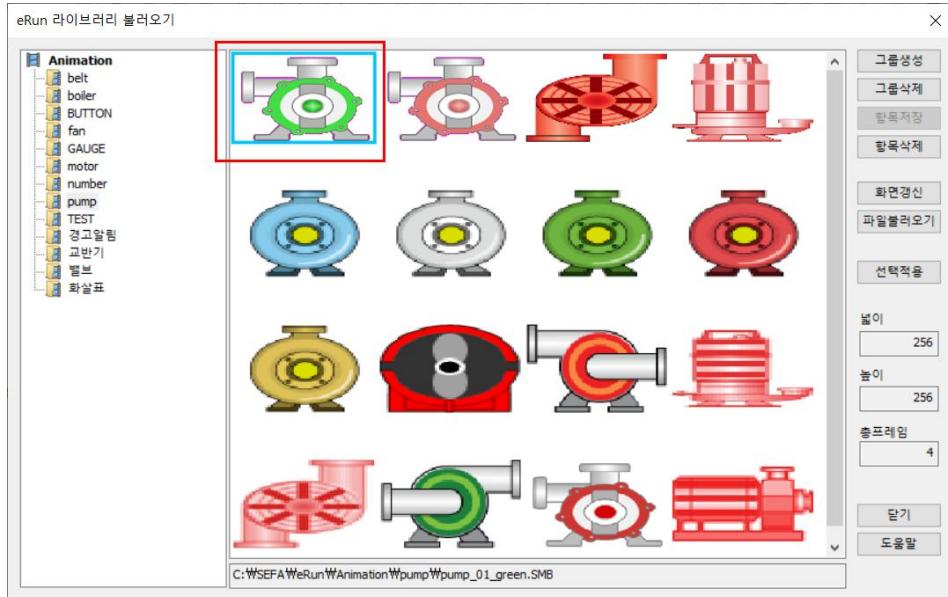


항목	설명
오브젝트 이름	오브젝트의 이름을 설정합니다. ※ 경보 오브젝트는 스크립트에서 사용자가 제어를 할 수 있는 오브젝트이기 때문에 다른 어떤 오브젝트의 이름과 동일해서는 안됩니다.
경보형태	■ 현재경보 : 리스트에 현재경보를 요약해서 표시합니다. ■ 경보이력 : 리스트에 경보이력을 표시합니다.
정렬순서	■ 발생 순서대로 : 경보발생 순서대로 리스트에 표시합니다. ■ 발생 역순으로 : 경보발생 역순으로 리스트에 표시합니다.
색상지정	■ 기본바탕색 : 기본 바탕색상을 설정합니다. ■ 신규발생 경보 : 신규로 발생되는 경보에 대한 색상을 설정합니다. ■ 확인된 경보 : 확인된 경보에 대하여 색상을 설정합니다. ■ 해제된 경보 : 해제된 경보에 대한 색상을 설정합니다.
시간형식	콤보박스에서 시간 포맷을 선택하며, 표현방식은 총 8가지 방식이 있습니다.

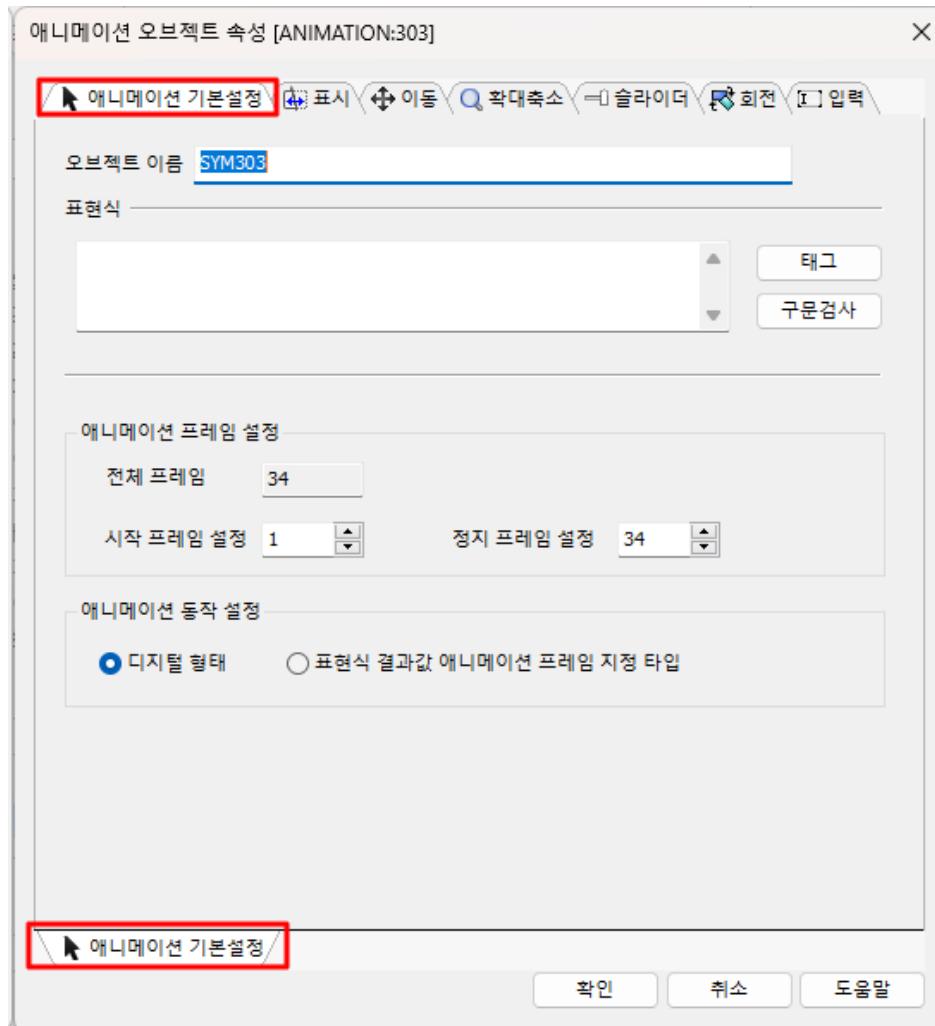
	<p>다음과 같은 형식으로 표시됩니다.</p> <p>yyyy-mm-dd, hh:mm:ss -> 2008-04-11, 15:05:37 yy-mm-dd, hh:mm:ss -> 08-04-11, 15:05:37 yyyy/mm/dd, hh:mm:ss -> 2008/04/11, 15:05:37 yy/mm/dd, hh:mm:ss -> 08/04/11, 15:05:37 yy/mm/dd, hh:mm:ss, -> AM 08/04/11, 3:05:37, PM hh:mm:ss, AM yy/mm/dd -> 3:05:37, PM 08/04/11 hh:mm:ss, AM dd/mm/yy -> 3:05:37, PM 11/04/08 yy-mm-dd, hh:mm [mon] -> 08-04-11, 15:05 [FRI]</p>
초기표시	프로젝트 실행할 때 현재경보를 보이게 할것인지 선택합니다
머리글 표시	머리글 표시 여부를 설정합니다.
TOOLBAR 표시	도구바 표시
표시항목 표시	<p>프로젝트 실행시 보여줄 항목을 선택합니다.</p>  선택항목 한칸 위로 이동합니다.  선택항목 한칸 아래로 이동합니다. <p>넓이비율 (%) : 선택항목의 표시 넓이 비율을 지정합니다.</p>
글씨체	글씨체와 크기 지정
마우스 두번클릭 시	<ul style="list-style-type: none"> ■ ACK 동작 : 경보를 확인합니다. ■ 상세 정보 보기 : 상세 정보 보기 동작으로 설정합니다

5.14 애니메이션 오브젝트

설비의 모양을 동적으로 움직임을 설정해서 사실적이고 직관적인 감시화면을 제공합니다. 애니메이션 오브젝트는 스튜디오에서 편집된 애니메이션 파일을 선택합니다. 태그와 연동해서 움직임에 대한 제어를 합니다.



이미지 아이콘을 마우스 두번 클릭하면 뷰 페이지 삽입됩니다. 속성창은 일반 오브젝트와 동일한 속성창을 이용합니다. 단, 기본설정 탭 이름이 애니메이션 기본설정 탭으로 바뀝니다. 서브탭은 별도로 없습니다.



가. 표현식

애니메이션 움직임과 연결되는 태그 값. 자세한 사항은 표현식 설정부분 참고하세요.

나. 애니메이션 프레임 설정

전체 프레임을 모두 표시하거나, 일부 구간 프레임만 표시하게 할 수 있습니다.

전체 프레임: 전체 프레임의 수 (수정불가)

시작 프레임 설정: 시작 프레임 번호 (1부터 시작)

정지 프레임 설정: 정지 프레임 번호.

다. 애니메이션 동작 설정

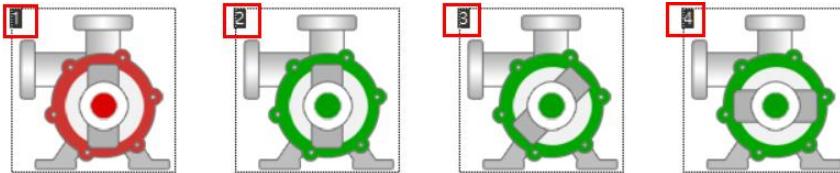
움직임을 어떻게 가져갈 것인지 선택합니다.

디지털 형태:

표현식 결과값 프레임 지정 형태:

표현식 값을 프레임 번호로 사용합니다. 설비 동작상태를 여러 개의 프레임 이미지로 그려 넣고 이 옵션을 사용

하면 설비의 움직임을 맞게 표시합니다. 적색 박스부분이 프레임 번호입니다.



5.15 날짜콘트롤 오브젝트

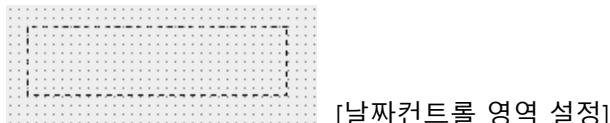
날짜와 시간을 선택할 때 사용합니다. 태그값의 이력검사를 해야 할 경우 시작일자와 종료일자를 선택하도록 합니다.

5.16.1 오브젝트 추가하기

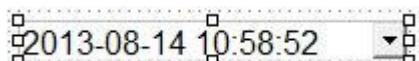
날짜컨트롤 오브젝트를 추가하기 위해서는 도구툴바에서 버튼을 클릭하거나 [메뉴]-[오브젝트]-[날짜컨트롤]을 선택하면 마우스 커서가 +로 변경됩니다.

키보드의 ESC 버튼을 누르면 그리기가 취소됩니다.

마우스가 드래그 되는 동안은 아래 그림과 같이 실선 사각형으로 그려집니다.

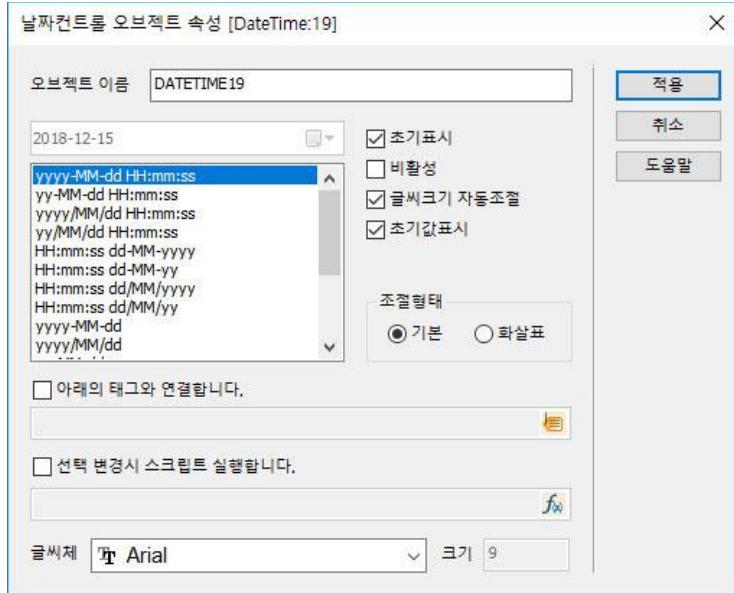


원하는 끝 위치에서 클릭하고 있던 마우스 왼쪽 버튼을 놓으면 아래 그림과 날짜컨트롤 오브젝트가 표시됩니다.



5.16.2 속성설정

날짜콘트롤 속성 설정창은 다음과 같습니다.



항목	설명
오브젝트 이름	<p>오브젝트의 이름을 설정합니다. 예) START_DATE or 시작일자</p> <p>※ 날짜콘트롤 오브젝트는 스크립트에서 직접 호출되기 때문에 뷰페이지에서 유일한 문자열 이름을 입력합니다.</p>
날짜표기방식 선택	<p>날짜 표기 방식을 선택합니다. (각 표기 방식 선택시 상단에 날짜 보여주는 방식 미리보기로 보여줌 - ex: YYYY-MM-dd HH:mm:ss -> 2013-01-08 12:18:53)</p>
초기표시	프로젝트 실행할 때 날짜콘트롤의 표시여부를 설정합니다
비활성	프로젝트 실행할 때 비활성 여부를 설정합니다.
글씨크기 자동조절	글씨크기의 자동 조절 여부를 설정합니다.
초기값 표시	연결태그 또는 문자열 항목에 설정된 초기값을 화면에 표시여부를 설정합니다.
조절형태	<p>날짜, 시간 선택 또는변경 방식을 말합니다.</p> <p>기본형 : 오브젝트의 우측 끝에 ▾ 버튼 누르면 월단위의 날짜 선택창이 아래와 같이 나타납니다.</p>

	화살표 : 위 아래 화살표 버튼을 클릭해서 날짜 변경합니다. 
태그연결	날짜 변경값이 반영될 태그명을 입력합니다. 공란이면 스크립트 함수를 통해서 변경된 값을 읽어옵니다. 태그명 앞에 '@'문자를 반드시 붙여야 합니다.
선택변경시 스크립트 실행	날짜 변경이 되면 실행할 스크립트 함수를 입력합니다.
글씨체, 크기	사용자 지정폰트 사용을 선택시 해당 글씨체와 크기로 보여줍니다.

5.16 콤보박스 오브젝트

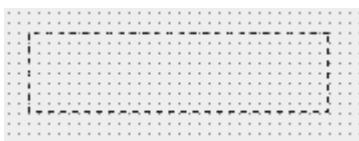
콤보박스 오브젝트는 드롭다운 목록 또는 리스트 상자와 편집 가능한 1줄 텍스트 상자의 결합으로, 사용자는 직접 값을 입력하거나 목록에서 값을 선택할 수 있도록 하는 오브젝트입니다.

5.16.1 오브젝트 추가하기

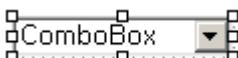
콤보박스 오브젝트를 추가하기 위해서는 도구툴바에서 버튼을 클릭하거나 [메뉴]-[오브젝트]-[콤보박스]를 선택하면 마우스 커서가 +로 변경됩니다.

키보드의 ESC 버튼을 누르면 그리기가 취소됩니다.

마우스가 드래그 되는 동안은 아래 그림과 같이 실선 사각형으로 그려집니다.

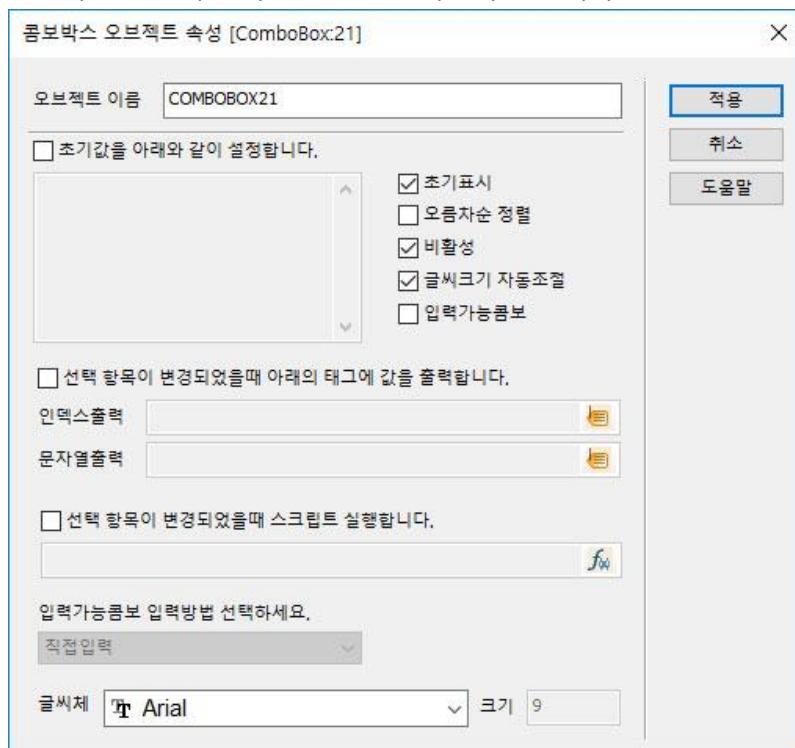


원하는 끝 위치에서 클릭하고 있던 마우스 왼쪽 버튼을 놓으면 아래 그림과 같이 콤보박스 오브젝트가 표시됩니다.



5.16.2 속성설정

콤보박스 오브젝트 속성 설정창은 다음과 같습니다.



항목	설명
오브젝트 이름	오브젝트의 이름을 설정합니다. ※ 콤보박스 오브젝트는 스크립트에서 직접 호출되기 때문에 뷰페이지에서 유일한 문자열 이름을 입력합니다.
초기표시	프로젝트 실행할 때 설정된 초기값의 표시 여부를 설정합니다.
오름차순 정렬	초기값 설정된 항목들을 오름차순으로 정렬을 해서 표시합니다.
비활성	프로젝트 실행할 때 비활성 여부를 설정합니다.
글씨크기	글씨크기의 자동 조절 여부를 설정합니다.
입력가능 콤보	콤보박스에 직접 입력 여부를 설정합니다
선택변경 태그연 결	사용자가 콤보박스 오브젝트의 항목선택을 할 때 태그에 선택항목의 인덱스값, 문자열 값을 태그에 저장합니다. 태그명 앞에 '@'문자를 반드시 붙여야합니다.
항목선택 변경시 스크립트 실행	항목선택을 하면 실행할 스크립트 함수를 입력합니다.
입력가능 콤보 입 력방법 선택	직접 입력 : 키보드직접 입력합니다. 가상키보드로 입력 : 가상 키보드로 입력합니다. 가상 숫자패드로 입력 : 가상 숫자패드로 입력합니다.
글씨체, 크기	사용자 지정폰트 사용을 선택시 해당 글씨체와 크기로 보여줍니다.

5.17 리스트박스 오브젝트

리스트박스란?

행의 갯수가 1개인 다중열로 구성된 리스트입니다.

리스트박스는 DB 쿼리결과용 또는 고정으로 표시된 데이터를 선택하여 처리하는 경우 사용합니다.

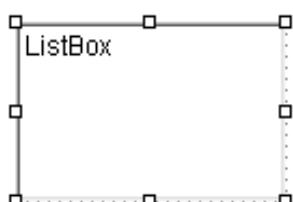
또는 사용자 운영 로그텍스트를 표시하는 용도로 활용합니다.

5.17.1 오브젝트 추가하기

리스트박스 오브젝트를 추가하기 위해서는 도구툴바에서 버튼을 클릭하거나

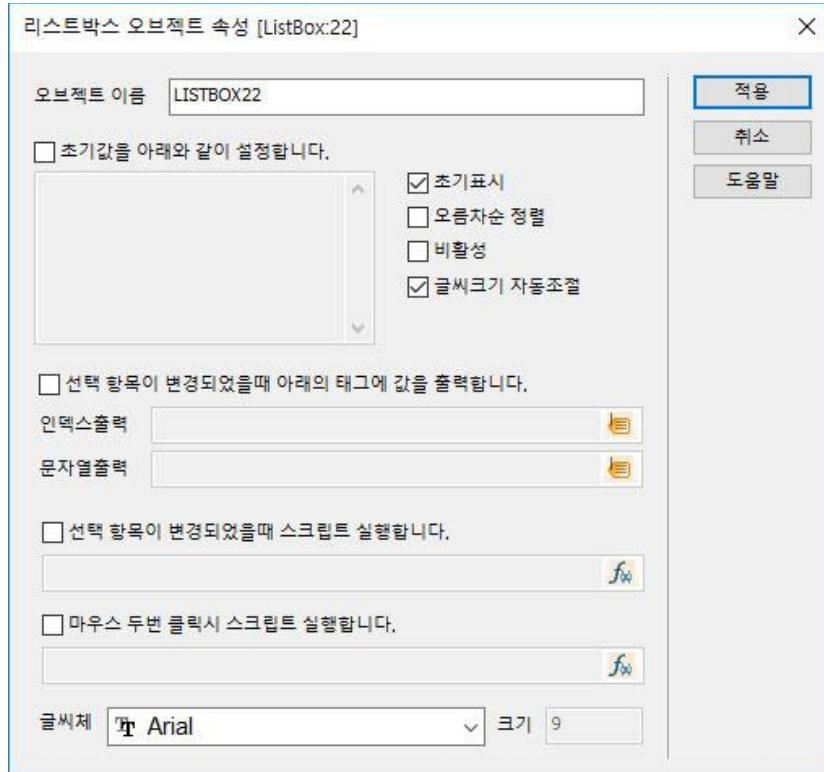
[메뉴]–[오브젝트]–[리스트박스]를 선택하면 마우스 커서가 +로 변경됩니다.

키보드의 ESC 버튼을 누르면 그리기가 취소됩니다.



5.17.2 속성설정

리스트박스 오브젝트 속성 설정창은 다음과 같습니다.



항목	설명
오브젝트 이름	오브젝트의 이름을 설정합니다. ※ 리스트박스 오브젝트는 스크립트에서 직접 호출되기 때문에 뷰페이지에서 유일한 문자열 이름을 입력합니다.
초기표시	프로젝트 실행할 때 설정된 초기값의 표시 여부를 설정합니다.
오름차순 정렬	초기값 설정된 항목들을 오름차순으로 정렬을 해서 표시합니다.
비활성	프로젝트 실행할 때 비활성 여부를 설정합니다.
글씨크기 자동조절	글씨크기의 자동 조절 여부를 설정합니다.
태그연결	리스트박스 항목 선택이 변경될 때 인덱스 값, 문자열값을 각각 태그에 저장합니다. 태그명 앞에 '@'문자를 반드시 붙여야 합니다.
항목선택 변경시 스크립트 실행	항목선택을 하면 실행할 스크립트 함수를 입력합니다.
마우스 두번클릭시 스크립트 실행	선택항목을 마우스두번 클릭할 때 실행할 스크립트 함수를 입력합니다.
글씨체, 크기	사용자 지정폰트 사용을 선택시 해당 글씨체와 크기로 보여줍니다.

5.18.3 스크립트 함수

함수	설명
_ListAddValue()	문자열을 추가합니다.
_ListDelValue()	목록에서 지정된 인덱스 번호에 해당하는 항목을 삭제합니다.
_ListFindFile()	지정 폴더에 있는 파일 및 폴더명을 모두 표시합니다.
_ListGetCount()	리스트박스 오브젝트의 전체 항목 개수를 돌려줍니다.
_ListGetSel()	선택된 항목의 순번을 돌려줍니다. (0부터 시작합니다.)
_ListGetValue()	인덱스 번호에 해당하는 항목을 문자열 형태로 돌려줍니다.
_ListReset()	내용을 모두 삭제합니다.
_ListSetSel()	인덱스 번호에 해당하는 항목을 선택합니다

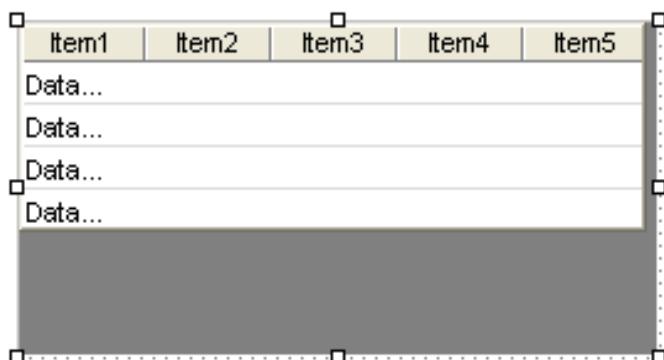
5.18 리스트콘트롤 오브젝트

리스트컨트롤 오브젝트는 확장리스트라고 하며 그리드 오브젝트와 유사한 속성을 가지고 있습니다. 여러가지 형식의 셀을 지정하고 입력이 가능한 그리드 오브젝트와 다르게 리스트콘트롤은 데이터 표시전용 셀리스트 오브젝트입니다.

주로 SQL 데이터베이스 쿼리 검색결과 표시용 오브젝트로 사용합니다.

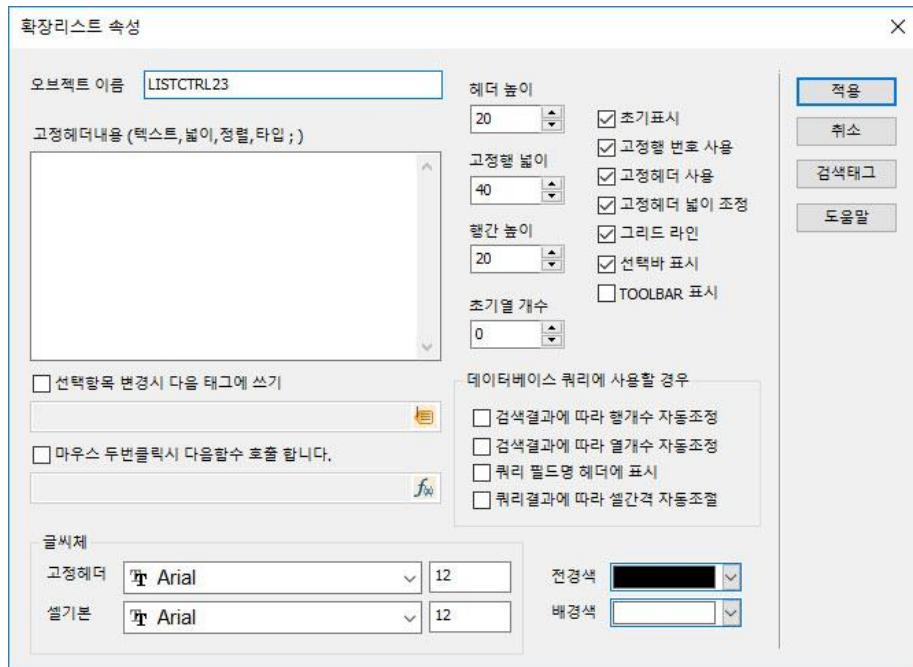
5.18.1 오브젝트 추가하기

도구바에서 버튼을 클릭하거나 [메뉴]-[오브젝트]-[리스트컨트롤]을 선택하면 마우스 커서가 +로 변경됩니다. 키보드의 ESC 버튼을 누르면 그리기가 취소됩니다.



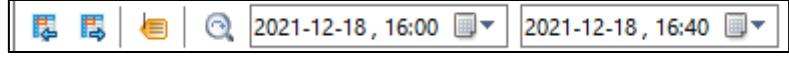
5.18.2 속성설정

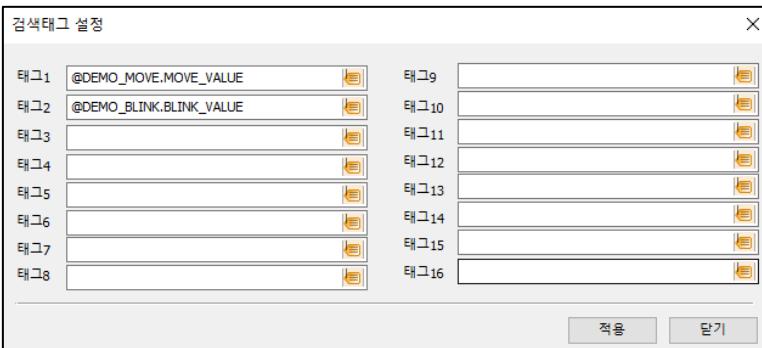
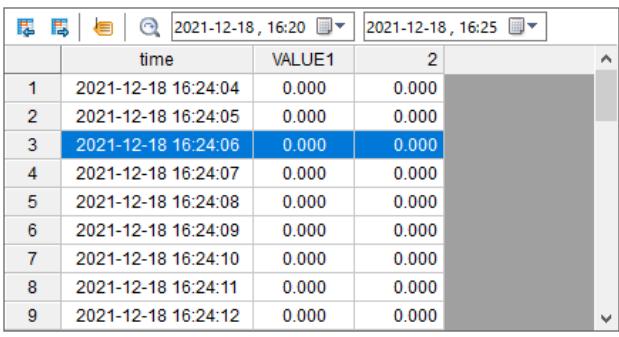
리스트콘트롤 설정 속성창은 다음과 같습니다.



항목	설명
오브젝트 이름	<p>오브젝트의 이름을 설정합니다. ※ 리스트콘트롤 오브젝트는 스크립트에서 사용자가 제어를 할 수 있는 오브젝트이기 때문에 다른 어떤 어떤 오브젝트의 이름과 동일해서는 안됩니다.</p>
고정 헤더 내용	<p>고정 헤더의 부분의 내용을 입력합니다. 텍스트, 넓이, 정렬, 타입; 순으로 입력합니다.</p> <p>※ 텍스트 : 문자열 ※ 넓이 : 픽셀단위의 넓이 ※ 정렬 : 정렬방식 (0 : 왼쪽, 1 : 중앙, 2 : 오른쪽) ※ 타입 : 해당 컬럼이 문자열이면 0을, 숫자형이면 1을 지정합니다. 숫자형이면 천단위 구분기호(,)를 표시합니다.</p> <p>(EX) Name,300,2,0; → 텍스트, 폭(픽셀), 오른쪽, 문자형 필드</p>

헤더 높이	그리드의 헤더 높이를 설정합니다. (픽셀단위)
고정행 넓이	그리드의 고정행 넓이를 설정합니다. (픽셀단위)
행간 높이	그리드 행간의 높이를 설정합니다. (픽셀단위)
초기열 개수	그리드의 초기열 개수를 설정합니다.
고정행 번호 사용	고정행 번호 사용 여부를 설정합니다. (1,2,3,...)
고정헤더 사용	고정헤더 사용여부(컬럼명)
고정헤더 넓이 조정	<p>프로젝트 실행시 고정헤더의 넓이를 사용자에 의한 조정 여부를 설정합니다.</p>
그리드 라인	셀보더 표시
선택바 표시	선택열의 선택바를 표시합니다.
열전체 선택모드	열 전체를 선택
셀선택 채움표시	셀 보더만 표시할것인지 선택채움을 표시할 것인지 여부
초기표시	프로젝트 실행할 때 오브젝트 표시여부

선택항목 태그연 결	항목선택시 선택 열번호를 태그에 저장합니다.
마우스 두번 클릭 시 함수호출	마우스로 셀을 더블 클릭했을 때 사용자가 작성한 함수를 호출해서 실행합니다. 함수이름 우측의  버튼을 클릭해서 이미 작성된 스크립트 함수를 선택합니다.
데이터베이스 쿼 리에 사용할 경우	확장리스트 오브젝트를 SQL 데이터베이스 쿼리결과 표시창으로 사용할 경우 해당합니다. <ul style="list-style-type: none"> ■ 검색결과에 따라 행개수 자동조정 <ul style="list-style-type: none"> - 검색결과의 필드개수만큼 행개수를 자동 조절합니다. ■ 검색결과에 따라 열개수 자동조정 <ul style="list-style-type: none"> - 검색결과의 필드개수만큼 열개수를 자동조절합니다. ■ 쿼리 필드명 헤더에 표시 <ul style="list-style-type: none"> - 검색결과의 쿼리 필드명을 확장리스트 고정열에 표시합니다 ■ 쿼리결과에 따라 셀간격 자동조절 <ul style="list-style-type: none"> - 셀간격을 자동으로 조절합니다.
글씨체	<ul style="list-style-type: none"> ■ 고정헤더 : 고정헤더의 폰트와 글자크기를 설정합니다. ■ 셀기본 : 셀기본의 폰트와 글자크기를 설정합니다.
전경색	글씨체의 폰트색깔을 설정합니다.
배경색	리스트콘트롤의 배경색을 설정합니다.
TOOLBAR 표시	CSV 파일저장, CSV읽어오기, 태그값 이력검색용으로 사용할 때 도구바를 표시하도록 합니다. 

<p>검색태그</p> <p>태그값 이력검색용으로 확장리스트를 사용할 수 있습니다. 최대 16개의 태그를 지정할수 있습니다.</p> 																																								
<p>프로젝트 실행할 때 태그이력 검색결과입니다..</p>  <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th></th> <th>time</th> <th>VALUE1</th> <th>VALUE2</th> </tr> </thead> <tbody> <tr><td>1</td><td>2021-12-18 16:24:04</td><td>0.000</td><td>0.000</td></tr> <tr><td>2</td><td>2021-12-18 16:24:05</td><td>0.000</td><td>0.000</td></tr> <tr><td>3</td><td>2021-12-18 16:24:06</td><td>0.000</td><td>0.000</td></tr> <tr><td>4</td><td>2021-12-18 16:24:07</td><td>0.000</td><td>0.000</td></tr> <tr><td>5</td><td>2021-12-18 16:24:08</td><td>0.000</td><td>0.000</td></tr> <tr><td>6</td><td>2021-12-18 16:24:09</td><td>0.000</td><td>0.000</td></tr> <tr><td>7</td><td>2021-12-18 16:24:10</td><td>0.000</td><td>0.000</td></tr> <tr><td>8</td><td>2021-12-18 16:24:11</td><td>0.000</td><td>0.000</td></tr> <tr><td>9</td><td>2021-12-18 16:24:12</td><td>0.000</td><td>0.000</td></tr> </tbody> </table>		time	VALUE1	VALUE2	1	2021-12-18 16:24:04	0.000	0.000	2	2021-12-18 16:24:05	0.000	0.000	3	2021-12-18 16:24:06	0.000	0.000	4	2021-12-18 16:24:07	0.000	0.000	5	2021-12-18 16:24:08	0.000	0.000	6	2021-12-18 16:24:09	0.000	0.000	7	2021-12-18 16:24:10	0.000	0.000	8	2021-12-18 16:24:11	0.000	0.000	9	2021-12-18 16:24:12	0.000	0.000
	time	VALUE1	VALUE2																																					
1	2021-12-18 16:24:04	0.000	0.000																																					
2	2021-12-18 16:24:05	0.000	0.000																																					
3	2021-12-18 16:24:06	0.000	0.000																																					
4	2021-12-18 16:24:07	0.000	0.000																																					
5	2021-12-18 16:24:08	0.000	0.000																																					
6	2021-12-18 16:24:09	0.000	0.000																																					
7	2021-12-18 16:24:10	0.000	0.000																																					
8	2021-12-18 16:24:11	0.000	0.000																																					
9	2021-12-18 16:24:12	0.000	0.000																																					

5.19 푸시버튼 오브젝트

5.19.1 오브젝트 추가하기

버튼 오브젝트를 추가하기 위해서는 도구툴바에서  버튼을 클릭하거나

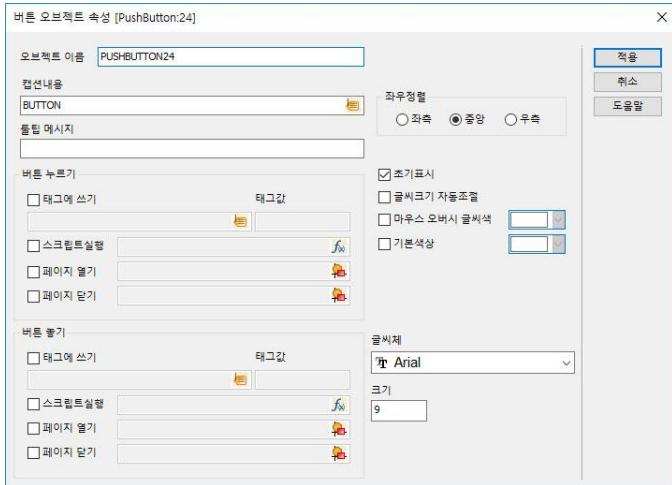
[메뉴]-[오브젝트]-[버튼]을 선택하면 마우스 커서가 로 변경됩니다.

키보드의 ESC 버튼을 누르면 그리기가 취소됩니다.



5.19.2 속성설정

푸시버튼 설정 속성창은 다음과 같습니다.



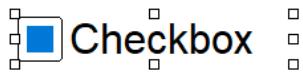
항목	설명
오브젝트 이름	<p>오브젝트의 이름을 설정합니다. ※ 푸시버튼 오브젝트는 스크립트에서 사용자가 제어를 할 수 있는 오브젝트이기 때문에 다른 어떤 오브젝트의 이름과 동일해서는 안 됩니다.</p>
캡션내용	버튼에 표시될 내용을 설정합니다.
툴팁메시지	현재 입력에 대해서 eRun에서 툴팁메시지에 표시할 메시지를 설정합니다.
좌우정렬	캡션내용의 정렬을 설정합니다.
버튼 누르기	<p>버튼을 눌렀을 때 태그 및 스크립트, 페이지 열기, 닫기를 설정합니다.</p> <ul style="list-style-type: none"> ■ 태그에 쓰기 : 태그에 미리 입력된 태그값을 씁니다. ■ 스크립트 실행 : 입력된 스크립트 함수를 실행합니다. ■ 페이지 열기 : 뷰페이지를 엽니다. 입력창 우측 버튼누르면, 아래와 같이 뷰페이지 선택창이 나타납니다. <div style="border: 1px solid #ccc; padding: 5px; margin-top: 5px;"> </div> <ul style="list-style-type: none"> ■ 페이지 닫기 : 뷰페이지를 닫습니다.
버튼 놓기	<p>버튼을 놓았을 때 태그 및 스크립트, 페이지 열기, 닫기를 설정합니다.</p> <ul style="list-style-type: none"> ■ 태그에 쓰기 : 태그에 미리 입력된 태그값을 씁니다. ■ 스크립트 실행 : 입력된 스크립트 함수를 실행합니다. ■ 페이지 열기 : 뷰페이지를 엽니다. ■ 페이지 닫기 : 뷰페이지를 닫습니다.
초기표시	프로젝트 실행할 때 버튼을 보이게 할 것인지 선택합니다.

글씨크기 자동조절	해당 버튼의 사이즈에 맞게 글씨를 자동 크기 조절합니다.
마우스 오버시글씨색	마우스가 버튼위에 위치한 경우 글씨색을 설정합니다.
기본색	글씨의 기본색상을 설정합니다.
글씨체	버튼내의 글씨에 대하여 글씨체와 크기를 지정합니다.

5.20 체크박스 오브젝트

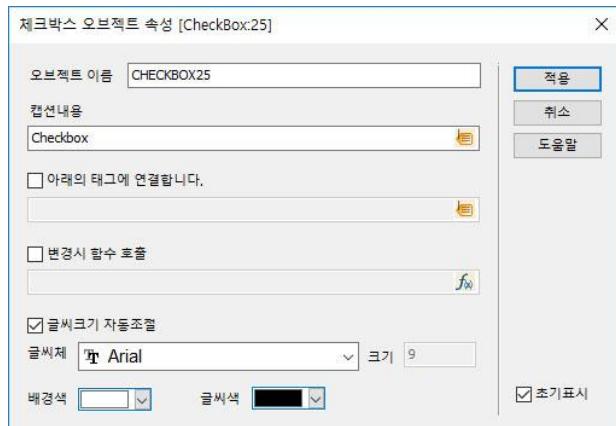
5.20.1 오브젝트 추가하기

체크박스 오브젝트를 추가하기 위해서는 도구툴바에서 버튼을 클릭하거나 [메뉴]-[오브젝트]-[체크박스]를 선택하면 마우스 커서가 +로 변경됩니다. 키보드의 ESC 버튼을 누르면 그리기가 취소됩니다.



5.20.2 속성설정

체크박스 오브젝트 속성 설정창입니다.



항목	설명
오브젝트 이름	오브젝트의 이름을 설정합니다. ※ 체크박스 오브젝트는 스크립트에서 사용자가 제어할 수 있는 오브젝트이기 때문에 다른 어떤 오브젝트의 이름과 동일해서는 안됩니다.
캡션내용	체크박스 타이틀에 들어가는 내용을 설정합니다. 타이틀 텍스트를 태그명으로 넣어주면 태그값으로 표시됩니다. @태그명으로 입력합니다.

태그연결	체크상태 연결할 태그를 설정합니다.
변경시 함수호출	체크박스 클릭시 스크립트 함수 호출합니다.
초기표시	프로젝트 실행할 때 체크박스를 보이게 할 것인지 선택합니다.
글씨크기 자동조절	해당 버튼의 사이즈에 맞게 글씨를 자동 크기 조절합니다.
배경색, 글씨색	글씨의 기본색상을 설정합니다.
글씨체	글씨체와 크기를 지정합니다.

5.21 라디오버튼 오브젝트

여러 개의 항목에서 하나만 선택하게 할 때 사용하는 오브젝트입니다.

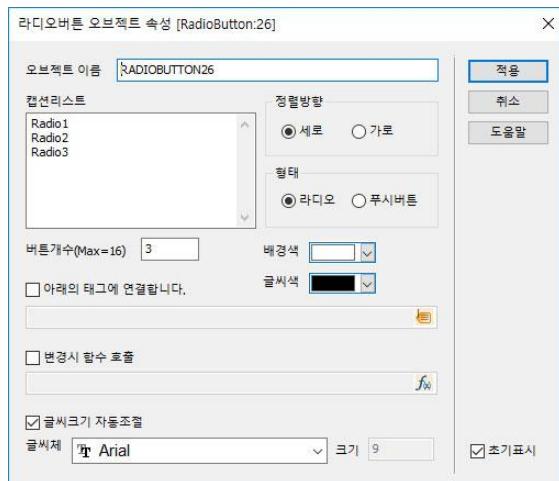
5.21.1 오브젝트 추가하기

라디오버튼 오브젝트를 추가하기 위해서는 도구툴바에서 버튼을 클릭하거나 [메뉴]-[오브젝트]-[라디오버튼]를 선택하면 마우스 커서가 +로 변경됩니다. 키보드의 ESC 버튼을 누르면 그리기가 취소됩니다.



5.21.2 속성설정

라디오버튼 속성 설정창은 다음과 같습니다.



항목	설명
----	----

오브젝트 이름	오브젝트의 이름을 설정합니다. ※ 라디오버튼 오브젝트는 스크립트에서 사용자가 제어를 할 수 있는 오브젝트이기 때문에 다른 어떤 오브젝트의 이름과 동일해서는 안됩니다.
캡션리스트	라디오버튼의 항목에 들어가는 내용을 설정합니다.
버튼개수	라디오버튼 개수를 설정합니다. 최대 16개까지 가능하며 기본은 3개입니다.
정렬방향	라디오버튼의 가로/세로 방향을 설정합니다.
형태	라디오버튼의 형태를 설정합니다.
태그연결	라디오버튼 선택항목이 변경될 때 항목번호를 태그에 저장합니다.
변경시 함수호출	라디오버튼 선택 변경이 되었을 시 해당 함수를 호출합니다.
초기표시	프로젝트 실행할 때 라디오버튼 오브젝트를 보이게 할 것인지 선택합니다.
글씨크기 자동조절	해당 버튼의 사이즈에 맞게 글씨를 자동 크기 조절합니다.
배경색, 글씨색	글씨의 기본색상을 설정합니다.
글씨체	글씨체와 크기를 지정합니다.

5.22 차트 오브젝트

실시간으로 태그값 상태에 따라 시각적으로 다양한 형태로 보여지게 하기 위해서 eRun SCADA에서는 차트 오브젝트를 제공합니다. 주로 과거 이력데이터보다 실시간 상태 표시용으로 사용합니다.

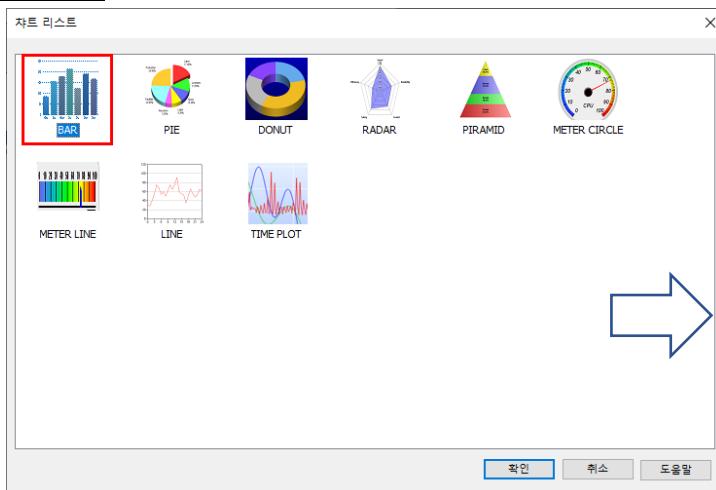
BAR 그래프, PIE 차트, DONUT, RADAR, PIRAMID, METER CIRCLE, METER-LINE, LINE, TIME PLOT 등 현재 버전에서 9개 종류가 있습니다.

5.22.1 오브젝트 추가하기

도구바에서 버튼을 선택하거나, [메뉴]-[오브젝트]-[차트]를 선택하고 추가합니다.

키보드의 ESC 버튼을 누르면 그리기가 취소됩니다.

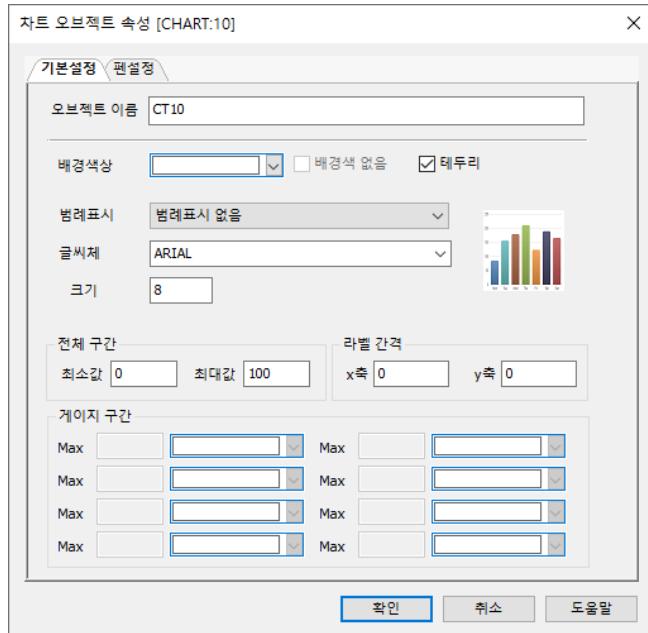
제공되는 차트를 선택할 수 있는 차트리스트 창이 나타납니다.



5.22.2 속성설정

차트 오브젝트 속성 설정은 [기본설정], [펜설정] 2개의 창으로 다음과 같습니다.

■ 기본설정 창

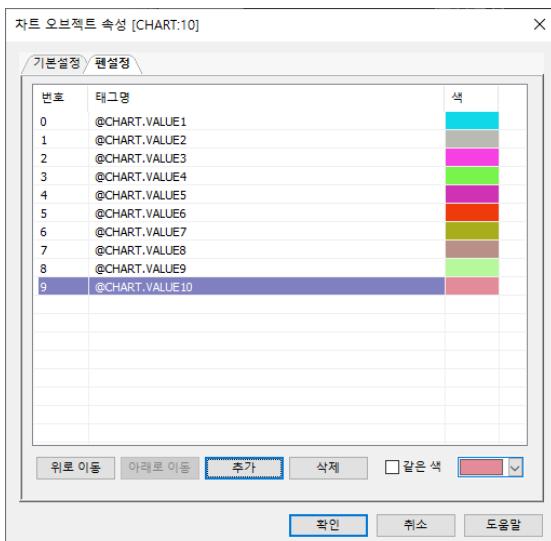


항목	설명
오브젝트 이름	오브젝트의 이름을 설정합니다. ※ 차트 오브젝트는 스크립트 함수를 통해서 제어하지 않습니다. 그러나 오브젝트 이름은 해당 뷰페이지에서는 유일 해야 합니다.
배경색상	차트 배경색을 지정합니다.
배경색 없음	배경색 없이 투명하게 표시할 경우 체크합니다.
테두리	오브젝트 사각 경계선을 표시합니다.

범례표시	항목별 제목을 다음과 같은 종류로 표시합니다. ■ 태그명 ■ 태그설명 ■ 태그명, 태그설명, 현재값 ■ 순서 : 일련번호로 표시합니다.
글씨체, 크기	범례표시 부분의 텍스트 글씨체입니다.
전체구간 최소,최대값	
라벨간격	
게이지구간	

■ 펜설정 창

펜 설정창은 차트를 구성하는 항목에 대한 태그설정을 합니다. 최대 16개까지 선택할 수 있습니다.



항목	설명
위, 아래로이동	태그표시 순서를 위, 아래로 조정해서 정렬합니다.
추가	태그리스트 선택창을 열어서 태그를 최대 16개까지 선택합니다. 마우스 드래그 드롭을 해서 한 번에 선택할 수 있습니다.

삭제	선택한 태그표시 삭제합니다.
같은색	<p>태그표시를 모두 동일한 색으로 표시합니다. LINE차트는 BAR CHART와 같은 속성을 갖지만 연속된 라인형태로 표시됩니다. 그래서 무조건 같은색 라인으로 표시되고, 자동으로 비활성화 됩니다.</p>

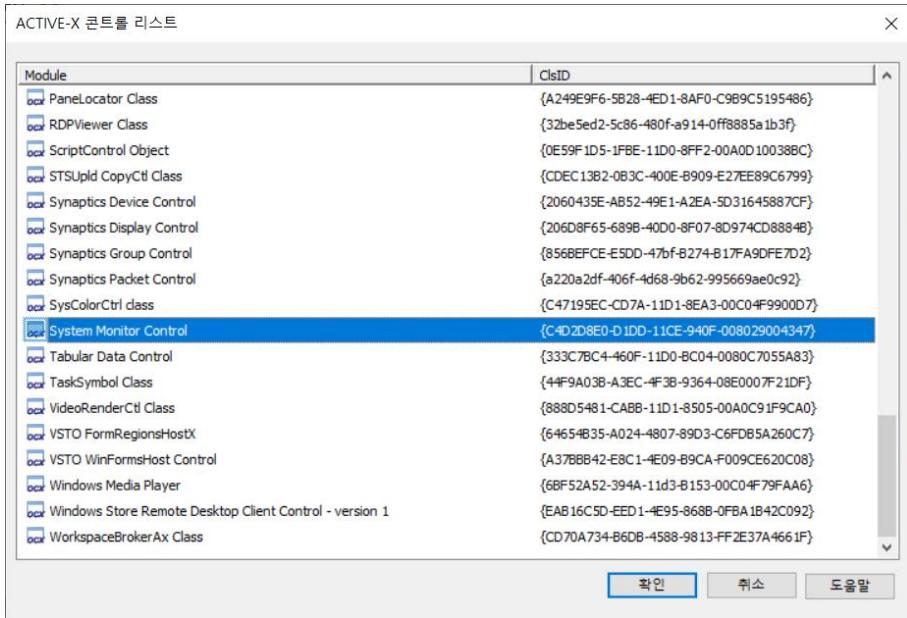
5.23 ACTIVE-X 오브젝트

액티브X(ActiveX)는 마이크로소프트사가 개발한 재사용 가능한 객체 지향적인 소프트웨어 구성 요소 개발에 사용되는 기술입니다. 액티브X 컴포넌트 오브젝트는 특정 프로세스를 단독으로 실행하는 일종의 외부 프로그램입니다. 다만 호출해주는 프로세스가 있어야 실행이 가능합니다.

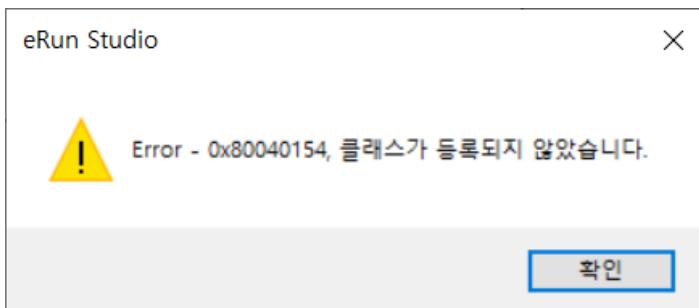
※ ACTIVE-X 모듈은 대체적으로 제작사의 설명이 부족한 경우가 많습니다. 확실히 검증된 모듈을 사용하는 것이 좋습니다. 직접 ACTIVE-X 모듈을 만들어서 eRun과 융합해서 사용하면 아주 좋습니다.

5.23.1 오브젝트 추가

액티브X 모듈은 라이선스가 있어야만 사용이 가능한 것들도 있습니다. 시스템마다 설치된 모듈이 매우 다양합니다. 액티브X 삽입을 하면 시스템에 설치되어 있는 모델 리스트 창이 나타나고, 모듈을 선택해서 오브젝트로 삽입합니다.



오류가 있는 모듈이거나 라이선스가 있어야만 사용이 가능한 모듈이면 다음과 같이 메시지창이 뜨고 삽입에 실패합니다.

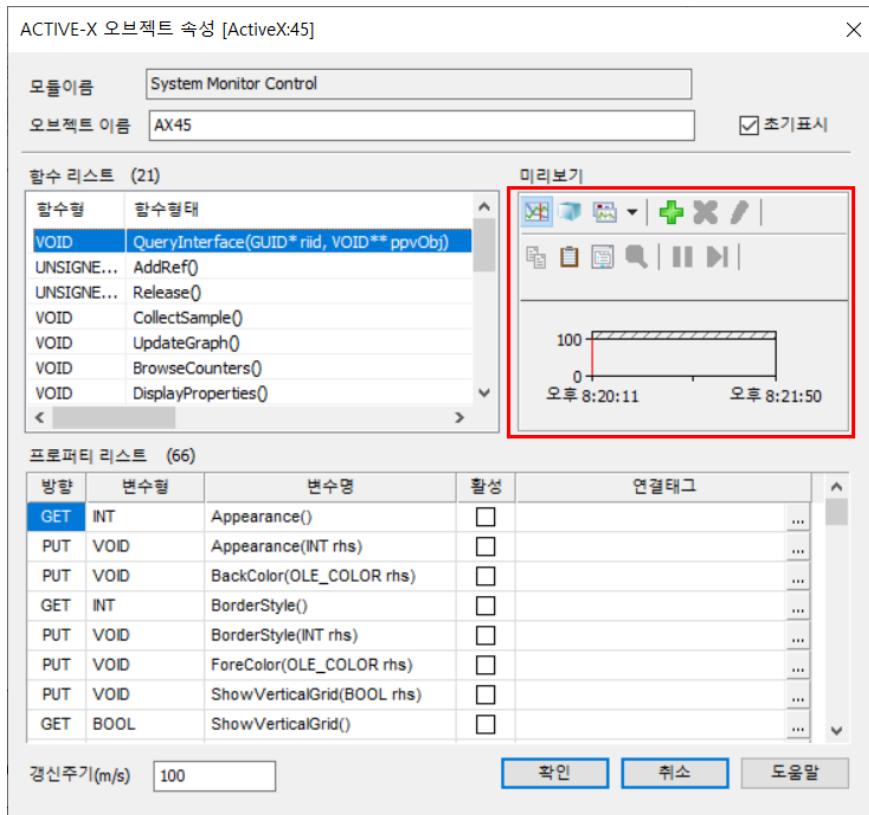


정상적으로 사용할 수 있는 모듈이면 다음과 같이 액티브X 오브젝트 아이콘 이미지가 나타나고 정상적정상 삽입되었다는 것을 알 수 있습니다.



5.23.2 속성설정

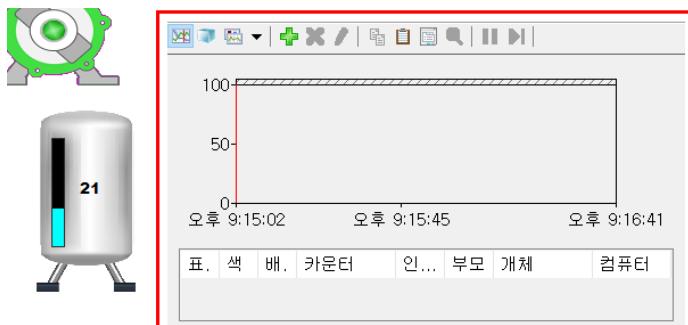
액티브X 속성창을 들어가면 다음과 같습니다. 액티브X 모듈에 따라 당연히 함수 리스트, 프로퍼티, UI가 모두 다릅니다.



항목	설명
모듈이름	액티브X 모듈 이름이 표시됩니다. (편집불가)
오브젝트 이름	뷰 페이지에서 유일한 오브젝트의 이름을 설정합니다.
초기표시	프로젝트 실행할 때 보이게 합니다.
함수리스트	<p>모듈이 제공하는 함수리스트입니다. 스크립트에서 액티브X 내부함수 호출할 때 사용합니다. 이 함수는 스크립트에서 _ComMethod() 함수 이용해서 호출합니다. 함수 파라메터도 적당하게 넣어주어야 합니다.</p> <p>(예시)</p> <pre>ret = _ComMethod("Active89", "AboutBox"); 오브젝트 이름 "Active89"에 정의되어 있는 AboutBox() 함수를 호출합니다.</pre>
프로퍼티	<p>액티브X 모듈에서 제공하는 내부변수를 말합니다. 스크립트에서 이 값을 변경하때 _ComSetValue() 함수를 사용하고, 읽을 때는 _ComGetValue()를 사용합니다. 또는 활성 체크를 하고 태그를 설정해주면 자동으로 모듈의 내부변수 값이 태그에 반영됩니다.</p>

	<p>(예시)</p> <pre>_ComSetValue("Active89", "Col", 100) value = _ComGetValue("Active89", "Col");</pre> <p>(주의)</p> <p>액티스X에서 제공하는 함수이름, 프로퍼티의 변수명은 대소문자를 구분하므로 맞추어 주어야 합니다.</p>
갱신주기	연결된 태그를 액티브X 모듈 내부변수 값에 반영하는 주기입니다. 기본 100ms
미리보기	작업중인 액티브X 모듈의 실행 화면 미리보기 창입니다.

다음은 실행화면에 나타나는 프로젝트 실행 화면입니다.



5.24 비전 오브젝트

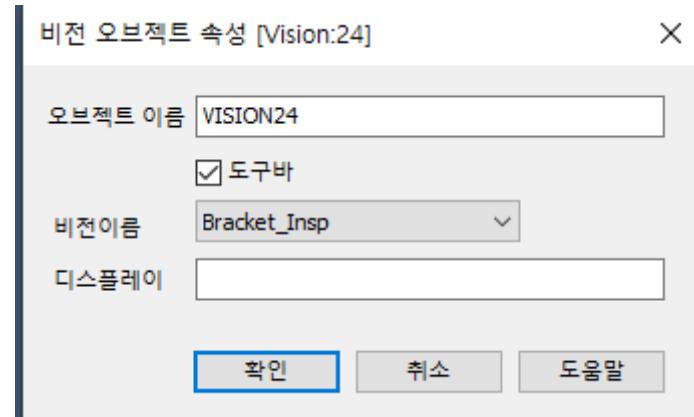
앞서 [4.4.4 VISION 설정]에서 설정을 마친 상태에서 실제 구현을 위한 세부적인 사항을 설명합니다.

5.24.1 오브젝트 추가하기

툴바에서 버튼을 선택하고, [메뉴]-[오브젝트]-[머신비전]을 선택하고 추가합니다.
뷰페이지에 이미지가 표시되는 곳을 선택하여 그려넣습니다.
키보드의 ESC 버튼을 누르면 그리기가 취소됩니다.

5.24.2 속성설정

편집하고자 하는 오브젝트를 더블클릭하거나, [마우스오른쪽버튼]-[속성설정]을 눌러 아래와 같이 속성을 편집 할 수 있습니다.



항목	설명						
오브젝트 이름	뷰 페이지에서 유일한 오브젝트의 이름을 설정합니다.						
도구바	프로젝트 실행할 때 영상이 표현되는 곳에 영상의 확대, 축소등이 가능한 툴바가 생성됩니다.						
비전이름	<p>환경설정에서 설정한 비전의 이름의 목록시 표시되며, 필요한 이름을 선택합니다. 환경설정 [Barcket_Insp]</p> <p> Setting</p> <p>기본 로고설정 HSMS Vision </p> <p>비전실행 리스트</p> <table border="1"> <thead> <tr> <th>비전이름</th> <th>프로그램 파일</th> <th>활성화</th> </tr> </thead> <tbody> <tr> <td>Bracket_Insp</td> <td>Insp.ivs</td> <td>활성</td> </tr> </tbody> </table>	비전이름	프로그램 파일	활성화	Bracket_Insp	Insp.ivs	활성
비전이름	프로그램 파일	활성화					
Bracket_Insp	Insp.ivs	활성					
디스플레이	<p>설록에서 작성한 여려개의 디스플레이중 출력하고자 하는 디스플레이의 이름을 선택합니다.</p> <p> Sherlock - C:\SEFA\Project\Barcket_Insp\Vision\Insp.ivs</p> <p>Program Edit View Run Image window Options Window Help</p> <p>imgA: Image dir: C:\SEFA\Project\Barcket_Insp\Vision\Images\image2</p>						

제 6 장 스크립트

6.1 소개 및 특징

eRun SCADA에서는 강력한 스크립트 기능을 제공합니다. 입력에 대한 빠른 반응속도와 처리를 할 수 있으며 프로젝트 개발에서 꼭 필요한 사용자 요구사항에 유연하게 대응하기 위하여 스크립트 언어를 제공합니다. 스크립트 문법은 간결한 C언어의 문법 형식을 따르며 반드시 필요한 최소한의 문법과 키워드를 사용하기 때문에 약간의 코딩학습만으로도 강력한 스크립트 코드를 작성할 수 있습니다. 사용자의 다양한 요구를 현장에서 바로바로 구현이 가능하기 때문에 확장성이 아주 좋습니다. 자체적으로 개발된 스크립트 엔진은 10여년 이상의 현장적용을 통해 안정적이고 창의적이고 강력한 SCADA 시스템 구축도구가 될 것입니다.

- 기능별 독립적인 함수의 형태로 종류별 프로그램 가능.
- 작고 명료하며 제작이 간단함.
- 정수형 데이터, 실수형 데이터, 문자열 데이터, 배열 타입지원.
- 260여개의 내장함수 지원.
- 여러가지 형태의 논리연산 가능.
- 사칙연산 및 데이터형 변환 가능.
- 반복처리, 조건제어문.
- 함수에서 다른 사용자함수를 호출가능.
- 태그데이터 조작이 용이.
- 사용자 UI 오브젝트 제어.
- 상용 데이터 베이스 SQL 질의처리.
- 애니메이션 아이콘에 대한 제어처리.
- 경보사운드 파일 재생.
- 외부 실행파일 호출 가능.
- 레지스트리 데이터 제어 가능.
- 그래프, 그리드, 리스트, 콤보, 경보내용 등 윈도우 오브젝트 제어 가능.

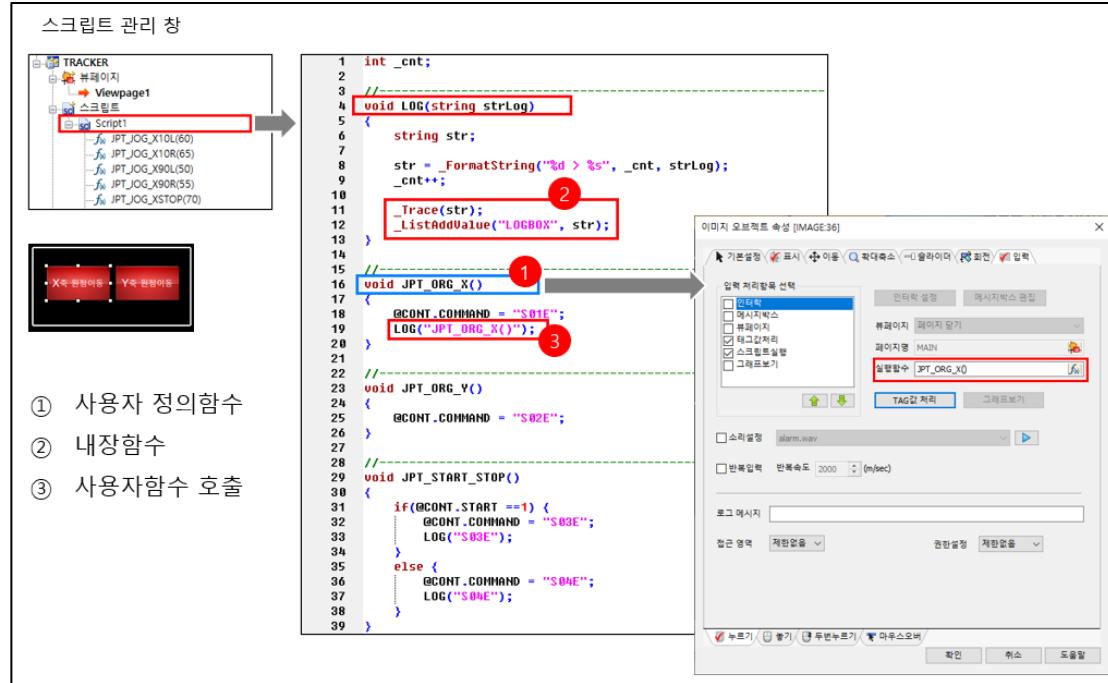
6.2 스크립트 프로세스 및 활용

1. 스크립트 파일 생성
2. 스크립트 함수 코딩
3. 스크립트 컴파일 및 오류검증
4. 스크립트 함수 오브젝트 연결

■ 스크립트 함수 적용 예시

스크립트 관리창에서 파일 하나를 생성하고 스크립트 파일 편집창에서 코딩을 합니다.

코딩된 함수를 오브젝트에서 호출하도록 설정합니다.

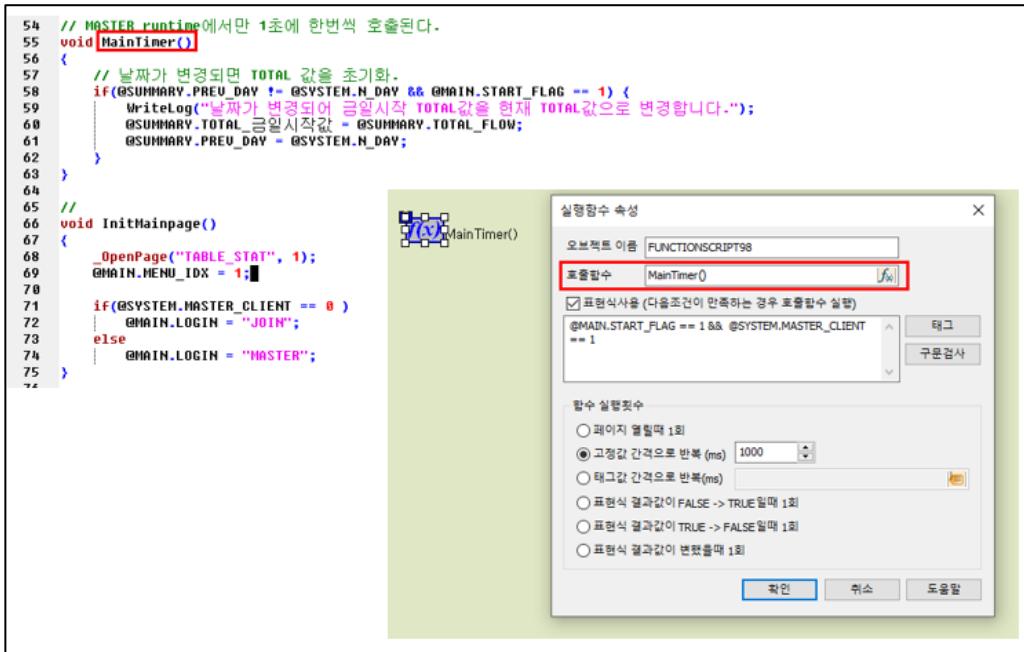


- ① 사용자 정의함수
- ② 내장함수
- ③ 사용자함수 호출

■ 실행함수 오브젝트에 의한 함수호출

실행함수 속성의 표현식 부분의 결과가 TRUE이면 사용자 정의함수 MainTimer()를

1초 주기로 실행합니다.



1. 스크립트창에 함수 코딩

```
void MainTimer() { }
```

2. 뷰페이지 실행함수 오브젝트 추가.

3. 실행함수 오브젝트 속성 – 호출함수 입력.

■ 이미지 오브젝트 클릭시 함수호출

뷰페이지에 그려진 이미지를 선택하고 속성창에 들어가서 순서대로 체크해 줍니다.

프로젝트 실행할 때 뷰페이지의 이미지를 클릭하면 사용자가 정의한 함수 JPT_ORG_X()를 실행합니다.



6.3 스크립트 생성 및 편집

프로젝트에 새로운 스크립트 파일을 생성하고 사용자 함수를 생성 및 편집합니다.

한 개의 스크립트 파일에는 무제한의 사용자 스크립트 함수를 만들 수 있습니다.

그러나 가능한 한 동일한 목적으로 사용되는 함수들을 모아서 스크립트 파일을 구성하시는 것이 편리합니다.

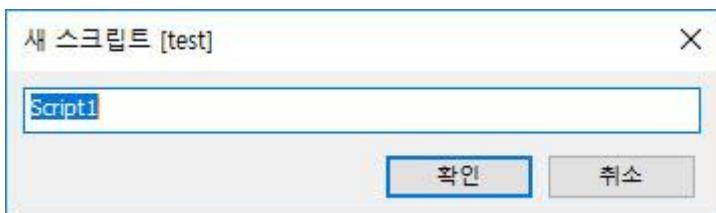


프로젝트 관리자창에서 스크립트 매니저 항목에서 마우스 우클릭을 하면 스크립트 팝업메뉴가 나타납니다.

[새 스크립트] 클릭하면 아래 그림과 같은ダイ얼로그가 표시됩니다.

입력창에 파일이름을 입력하고 확인버튼을 누르면 스크립트 편집파일 이름이 추가됩니다.

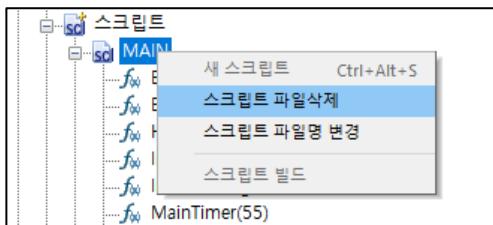
※ 스크립트 파일이름에 특수문자를 사용할 수 없습니다.



6.4 스크립트 파일삭제

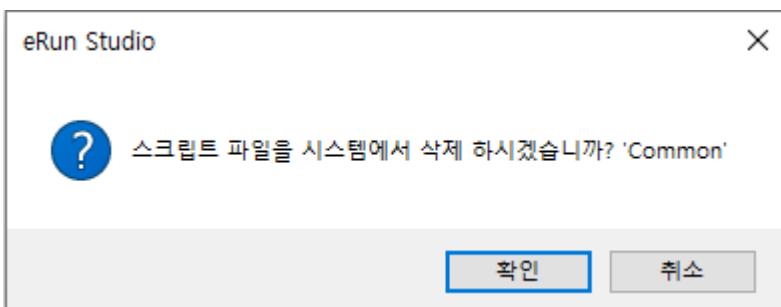
프로젝트 관리에서 현재 선택한 스크립트 파일을 삭제합니다.

프로젝트 관리에서 해당 스크립트에서 우측 마우스 클릭하여 팝업 메뉴를 띄웁니다.



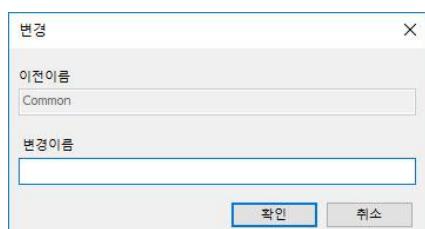
스크립트 삭제 메뉴를 클릭하면 아래와 같은 메시지 창이 표시됩니다.

“예”를 선택하면 스크립트 파일 ‘Common’ 삭제하고 파일 안에 있는 사용자정의 스크립트 함수 모두가 삭제됩니다. “아니오”를 선택하면 스크립트파일 삭제 작업을 취소합니다.



6.5 스크립트 파일명 변경

스크립트 파일의 이름을 변경하고자 할 때 이름변경 메뉴를 클릭하면 아래 그림과 같은 이름변경ダイ얼로그가 표시됩니다. 변경이름 입력 창에 변경하고자 하는 이름을 입력하고 확인버튼을 클릭하면 됩니다 ※ 스크립트 파일이름에 특수문자를 사용할 수 없습니다.



6.6 스크립트 빌드

프로젝트의 모든 스크립트를 컴파일 해서 바이너리코드를 생성합니다. 그리고 스크립트 문법에 구문오류가 있는지 검사하고, 사용된 태그나 오브젝트가 유효한지 검증을 합니다.



제 6 장 스크립트

스크립트 빌드는 도구바에서 버튼 클릭 또는 스크립트에서 마우스 오른쪽 클릭하여 '스크립트 빌드' 메뉴 선택을 하면 됩니다.

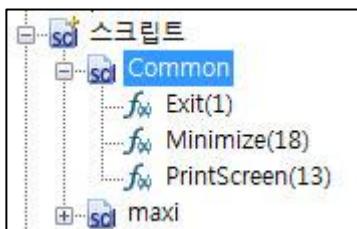
스크립트 컴파일이 성공적으로 끝났을 경우 표시되는 메시지입니다.

```
Compiling...전광판.scl  
Compiling...UnderPage.scl  
Compiling...SUMMARY.scl  
Compiling...REPORT.scl  
Compiling...DATABASE.scl  
Compiling...REPORT_AUTO.scl  
  
Linking...  
링크오류가 0개 있습니다..  
스크립트 코드가 정상적으로 생성되었습니다. !!!  
  
Verifying...  
스크립트 코드의 유효성 검사가 정상적으로 완료되었습니다. !!!  
  
[ скрипт 빌드 / 문자열 찾기 / DB메시지 / DB처리결과 / ]
```

아래의 그림은 구문에 오류가 있는 경우 나타나며 해당 오류메시지를 마우스 선택 후 두번 클릭하면 해당 오류가 있는 위치로 이동하게 됩니다.

```
Compiling...전광판.scl  
Compiling...UnderPage.scl  
Compiling...SUMMARY.scl  
Compiling...REPORT.scl  
Compiling...DATABASE.scl  
Compiling...REPORT_AUTO.scl  
  
Linking...  
#HDRMS:스크립트.%MAIN.scl, 정의되지 않은 할수입니다.(_DeleteGoooo), line=224,col=21  
링크오류가 1개 있습니다...
```

성공적으로 스크립트 컴파일이 되고 나면 아래 그림과 같이 프로젝트 관리의 스크립트 편집 항목에 각각의 스크립트 파일에 정의 되어있는 사용자 함수 리스트가 추가됩니다.



사용자가 정의한 스크립트 함수리스트와 위치해 있는 라인번호가 보여집니다.

해당 함수의 위치로 바로 가려면 마우스를 선택하고 왼쪽버튼을 더블 클릭하면 이동합니다.

6.7 스크립트 편집메뉴 구성

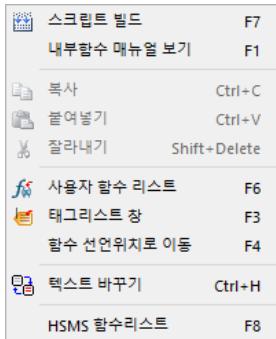
스크립트 편집화면에 들어가면 아래와 같은 도구가 보여집니다.

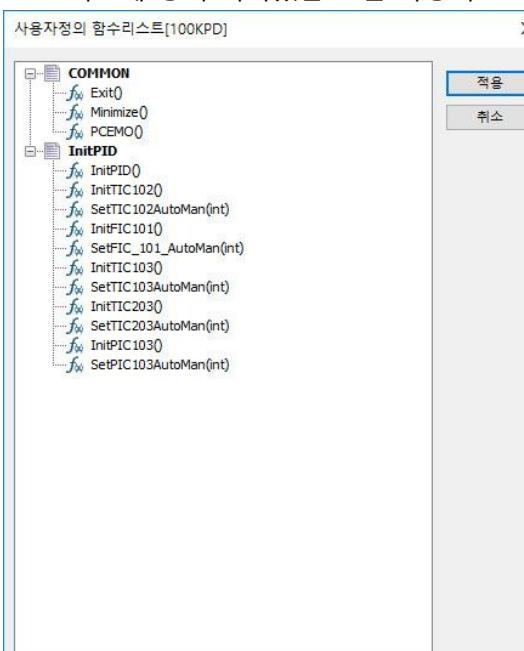
컴파일, 사용자 정의 함수, 문자열 찾기, 문자열 변경, 태그리스트 보기 등을 할 수 있습니다.



[스크립트 편집 도구바]

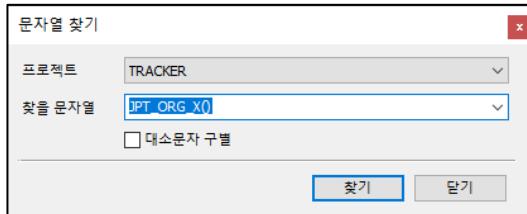
스크립트 창에서 마우스 우클릭하면 다음과 같이 팝업메뉴가 표시됩니다.



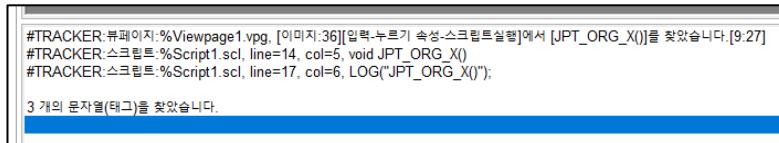
항목	설명
스크립트 빌드	현재 열려 있는 스크립트를 포함한 프로젝트에 있는 모든 스크립트를 스크립트 빌드하고 구문오류를 찾아냅니다.
내부함수 매뉴얼	eRun에서 제공하는 내장함수리스트 매뉴얼입니다.
사용자 함수리스트	현재 프로젝트에 정의 되어있는 모든 사용자 스크립트 함수 리스트를 보여줍니다.  <p>프로젝트내에서 사용자가 정의한 스크립트 함수리스트입니다.</p>
텍스트 바꾸기	특정한 문자열을 변경하고자 할 경우 사용합니다. 선택영역에 한해서 문자열 변경을 할 수도 있고 대소문자 구별, 일치하는 문자열만 변경할 수 있습니다.

	<p>태그리스트 창을 불러서 n개의 태그를 선택해서 스크립트에 삽입합니다. 태그이름 연산을 사용할 경우는 태그명 앞에 @문자를 반드시 붙여야 합니다.</p>
태그 리스트보기	
함수 선언위치로 이동	<p>선택한 함수가 정의되어 있는 위치로 이동합니다. 사용자가 작성한 함수에 대해서 적용이 됩니다.</p>
HSMS 함수리스트	<p>SECs/GEM 통신 프로젝트 개발시 사용합니다. HSMS 라이브러리에서 제공하는 함수를 스크립트창에 삽입할 때 사용합니다.</p> <p>왼쪽의 함수를 선택하면 우측창에 설명이 나타납니다. 함수선택을 하면 _HSMSMethod("Start", 삽입되고 코드를 완성하면 됩니다.</p>
문자열 찾기	<p>CTRL + F 키 누르면 다음과 같이 문자열 찾기 창이 나타납니다. 프로젝트 : 문자열을 찾으려고 하는 프로젝트를 선택합니다.</p>

찾을 문자열 : 함수이름 또는 태그명
대소문자 구별 : 대문자, 소문자 구분으로 찾을 경우 체크합니다.



찾기 버튼을 누르면 아래와 같이 해당 태그명(함수)가 입력된 곳이 나타납니다.
마우스 두번클릭하면 해당 오브젝트로 이동합니다.



6.8 함수의 구성

eRun SCADA 스크립트는 기능별 특정한 작업을 수행하도록 설계된 독립적인 프로그램을 처리하는 함수단위로 구성합니다. 함수를 사용하는 목적은 프로그램에서 여러 번 사용되는 부분들을 독립된 함수로 작성하고 여러 부분에서 함수를 호출하여 사용함으로써 불필요한 중복을 없애고, 프로젝트 개발을 용이하게 하고 프로그램을 읽기 쉽게 하는데 있습니다.

6.8.1 함수형 정의

함수의 구성형태는 아래와 같습니다.

```
함수형 함수명(매개변수)
{
    ...
}
```

함수형은 함수가 호출되고 값을 돌려주는 데이터의 형태를 의미하며, 여기에 가능한 함수형은 void, char, short, int, float, double, string 입니다.

함수명은 뷰페이지 오브젝트에서 호출하거나 스크립트 내에서 호출할 때 사용합니다. 당연히 프로젝트에서 함수명은 유일해야 합니다. 함수명의 구성은 알파벳(대소문자 혼용 가능) 문자열로 시작하며 32자 이내로 지정합니다.

매개변수는 함수를 호출하는 쪽에서 넘겨주는 데이터이며 0개 이상의 데이터를 지정할 수 있습니다 함수를 호



제 6 장 스크립트

출하는 쪽과 호출 받는 쪽(처리하는 함수)의 형태는 같아야 하고 지정방식은 약간 다릅니다. 예를 들어서 호출하는 쪽의 함수는 함수형은 빠지고 함수명과 매개변수만 지정합니다. 그리고 끝 부분에 ;(세미콜론) 문자를 붙여야 합니다.

호출 받는 함수의 정의 형태는 아래와 같이 여러 형태로 사용 가능합니다.

// 돌려주는 데이터는 없음, 받는 데이터 없음.

```
void main1()
```

```
{  
}
```

// 돌려주는 데이터 정수형, 받는 데이터 정수형 데이터변수.

```
int main2(int num)
```

```
{  
    return 100;  
}
```

// 돌려주는 데이터 실수형, 받는 데이터 실수형, 문자열형, 정수형 데이터변수.

```
double main3(double a, string name, int b)
```

```
{  
    return (a * b);  
}
```

// 돌려주는 데이터 문자열형, 받는 데이터 정수형 데이터변수.

```
string main(int tel)
```

```
{  
    return "Technology";  
}
```

호출하는 부분에서 사용하는 함수사용은 아래와 같습니다.

함수명(매개변수);

// main1() 함수를 넘겨주는 데이터 없이 호출합니다.

```
main1();
```

// main2() 함수를 호출하며 정수형 데이터상수 100을 넘겨줍니다.

```
main2(100);
```

// main3()함수를 호출하며 실수형, 문자열형, 정수형 데이터상수를 각각 넘겨줍니다.

```
main3(-99.5, "대한민국", 20000);
```

6.8.2 변수의 정의

연산처리를 위하여 여러가지 형태의 변수를 선언하고 사용하는데, 변수를 선언 할 경우는 함수의 정의 부분에서 제일 위쪽에 선언하여야 하고, 데이터형은 정수형, 실수형, 문자열형 3종류의 형태가 있으며 각각의 데이터형을 따르는 배열형태의 정의도 가능합니다. 배열형이라고 하면 동일한 데이터형을 동일한 변수이름으로 n개를 가지고 있는 것을 말합니다.

정수형 : char(1바이트), short(2바이트), int(4바이트) 3종류가 지원됩니다.

실수형 : float(4바이트), double(8바이트) 2종류가 지원됩니다.

문자열 : string(4096바이트)를 사용합니다.

변수사용 형식은 아래와 같습니다.

```
// 변수 a, b, c 가 모두 int형 정수형 변수로 선언합니다.
```

```
int a, b, c;
```

```
// 변수 d를 실수형 데이터 변수로 선언합니다.
```

```
double d;
```

```
// 변수 name을 문자열형 데이터 변수로 선언합니다.
```

```
string name;
```

```
// 배열 변수를 선언합니다. 79 바이트의 문자열 버퍼를 확보합니다.
```

```
char name[80];
```

14-5-3. 변수의 크기

변수의 종류별 표현 가능한 범위는 아래와 같습니다.

char	: -128 ~ +127	1 바이트
short	: -32768 ~ +32767	2 바이트
int	: -2147483648 ~ +2147483647	4 바이트
float	: 3.4e-38 ~ 3.4e+38	4 바이트
double	: 1.7e-308 ~ 1.7e+308	8 바이트
string	: 모든 문자열 변수형	4096 바이트
text	: 모든 문자열 변수형	256 바이트

void : 변수로는 사용할 수 없고, 함수형으로 사용합니다.

6.8.2 연산자 종류

산술계산을 위한 기본연산자를 제공합니다.

■ 치환연산자 (=)

대입 연산자 라고도 하며, 이 연산자는 값을 치환하는 역할을 하는 연산자입니다. 예를 들어서, 다음 문장은 value라는 변수에 0이라는 값을 치환합니다.

```
value = 0;
```

다시 말해서 왼쪽에는 “변수이름”이 오게 되며 오른쪽에는 “값”이 오게 됩니다.

치환 연산자의 수행은 오른쪽에서 왼쪽으로 이루어집니다. 예들 들어서

```
value = value + 1
```

와 같은 문장은 오른쪽에 있는 value라는 변수의 값에 1을 더해서 결과값을 왼쪽의 value 변수에 대입하라는 의미가 되며 결과적으로 value의 값을 1만큼 증가시키는 결과가 됩니다.

■ 덧셈연산자 (+)

이 연산자는 좌,우 양쪽의 값을 더하는 역할을 합니다. 예를 들어서

```
value = 300 + 100;
```

와 같은 문장은 오른쪽에 있는 정수 상수값 300과 100을 합산해서 왼쪽의 변수 value에 대입합니다.

좌,우에 사용할 수 있는 값은 정수, 실수, 문자열등 모두 사용 가능합니다.

문자열을 사용하는 경우에는 좌측의 문자열에 우측의 문자열을 붙여서 문자열형(string) 변수에 대입합니다. 예를 들어서

```
str = "People" + "Technology";
```

위와 같이 하게 되면 “People” 문자열에 “Technology” 문자열을 더해서 “People Technology” 문자열을 변수 str에 대입합니다. 이때 str은 문자열형(string) 변수형태로 선언 하여야 합니다.

■ 뺄셈연산자(-)

이 연산자는 왼쪽의 값에서 오른쪽의 값을 빼는 역할을 합니다. 따라서

```
X = Y - Z;
```

의 문장은 변수 Y에서 변수 Z값을 빼어 그 결과값을 X에 치환합니다. 뺄셈연산자에서는 문자열 연산을 할 수 없습니다.

■ 곱셈연산자(*)

곱셈연산자는 *로 나타냅니다. 따라서,

```
cm = 2.54 * in;
```

와 같은 문장은 변수 in에 실수상수 2.54를 곱하고 결과값을 변수 cm에 대입합니다.

eRun Script에서는 제곱연산자가 없음으로 제곱연산을 하기 위해서는 아래와 같이 하시면 됩니다.

```
value = max * max;
```

■ 나눗셈연산자(/)

나눗셈 연산자는 / (SLASH)로 나타냅니다. 기호 왼쪽의 값을 기호 오른쪽의 값으로나 누어 줍니다.

```
div = 100.5 / num;
```

■ 연산자 우선순위

여러 수식을 가지고 연산을 하기 위해서는 연산자의 처리순서를 정할 필요가 있습니다.

이에 따라서 eRun Script에서는 아래와 같은 순서로 산술연산 및 논리연산을 합니다.

() 부호 * / + -의 순서로 연산이 됩니다. 예를 들어서,

```
result = -(2+5) * 6 + (4+3 * (2+3));
```

result의 값은 -23이 됩니다.

■ 나머지 연산자 (%)

이 연산자(modulus operator)는 정수의 연산에만 사용됩니다. 기호 왼쪽에 있는 정수변수 또는 상수값을 기호 오른쪽의 정수값으로 나눗셈 한 후 그 나머지를 대입합니다.

예를 들어서

```
value = 13 % 5;
```

위의 문장이 실행되면 13을 5로 나눗셈 한 후 남은 숫자 3을 value에 대입합니다.

증가(++), 감소(--) 연산자

증가/ 감소 연산자는 피연산자를 1씩 증가 혹은 1씩 감소 시킵니다.

가령 $i++$; 과 같은 형태일 경우, i 는 피연산자, $++$ 은 연산자라고 합니다.

이해를 위해 아래의 표를 참고하시기 바랍니다.

연산자	설명
$10 * i++$	먼저 i 와 10을 곱한 후, i 의 값을 증가 시킵니다. (다음과 동일) $10 * i;$ $i = i + 1;$
$10 * i--$	먼저 i 와 10을 곱한 후, i 의 값을 감소 시킵니다. (다음과 동일) $10 * i;$ $i = i - 1;$
$10 * ++i$	먼저 i 의 값을 1증가 시킨 후, 10을 곱합니다. (다음과 동일) $i = i + 1;$ $10 * i;$
$10 * --i$	먼저 i 의 값을 1감소 시킨 후, 10을 곱합니다. (다음과 동일) $i = i - 1;$ $10 * i;$

■ 논리연산자(AND, OR, NOT)

이 연산자는 조건문 등에서 조건질의를 하는데 있어서 여러 조건을 동시에 질의하기 위하여 사용되며 왼쪽 오른쪽의 연산항에 대하여 논리연산을 수행합니다.

$\&\&(AND)$, $||(OR)$, $!(NOT)$ 같이 3가지 형태의 논리적 연산을 지원합니다. 예들 들어서

```
if(X>0 && X<100) {
    // to do....
}
```

위의 경우 X 가 0보다 크고(AND) 100보다 작을 경우 블록 문에 있는 구문을 처리하게 됩니다.

이것은 두가지 조건의 결과가 모두 참(TRUE)인 경우만 블록 구문을 처리합니다.

다음의 경우 X 의 값이 0보다 작거나(OR) 100보다 큰 경우 블록문의 구문을 처리합니다.

이것은 두가지 조건 중 어떤 한 개의 조건이라도 만족하면 블록 구문을 처리합니다.

```
if(X<0 || X>100) {
    // to do....
```

```
}
```

다음의 경우에는 X값이 50이 아니면(NOT) 블록 구문을 처리합니다.

```
f(X != 50) {
    // to do....
}
```

이러한 논리연산자는 괄호연산자를 사용하여 여러 겹으로 동시에 사용이 가능합니다.

```
if( (X>0 && X<100) || (Y>0 && Y<100) || (Z != 50) ) {
    // to do....
}
```

위의 경우에는 괄호 안에 있는 논리연산 조건을 먼저 수행하고 왼쪽부터 차례로 연산을 합니다.

■ 비트별 논리연산자(&, |, ^)

정수형 데이터를 가지고 비트별 논리연산을 할 수 있으며, 10진수 16진수 모두 사용이 가능합니다. 이것은 정수데이터에서 특정 비트의 값을 취할 때 유용하게 사용할 수 있습니다.

```
value =0X0000;
value2 = value & 0xFFFF;
value3 = value | 65535;
value4 = 0xFF00 ^ 0x00FF;
```

위와 같은 경우 변수 value의 값과 16진 상수값 FFFF를 비트 연산합니다. 결과를 변수 value2에 대입하며 값은 0이 됩니다. 그리고 변수 value의 값과 65535(0xFFFF)10진수 상수값과 OR(|)연산을 수행하고 결과를 변수 value3에 대입합니다. 결과는 65535(0xFFFF)가 됩니다.

마지막의 연산에서는 왼쪽의 16진 상수값 0xFF00과 오른쪽의 16진 상수값 0X00FF값을 서로 ^(XOR) 한 후 결과를 변수 value4에 대입합니다. 결과는 0xFFFF(65535)가 됩니다.

■ 시프트 연산자(<<, >>)

정수형 데이터에 대하여 왼쪽 또는 오른쪽으로 지정된 비트수 만큼 이동시킨 결과를 되돌려 줍니다.
// 1에 대한 정수값을 왼쪽으로 1비트만큼 이동시킨 결과를 value에 저장합니다.

value = 0X0001 << 1; => value에는 0x0002 가 저장된다.

// 0x8000 에 대한 정수값을 오른쪽으로 1비트만큼 이동시킨 결과를 value에 저장합니다.

value = 0x8000 >> 1 => value에는 0x4000이 저장된다.

6.8.3 예약어 및 기본문법

■ 흐름제어문(if문, else문)

어떠한 조건에 따라서 스크립트 실행의 방향을 바꾸기 위한 것으로 여러 관계연산자와 논리연산자 그리고 산술연산자의 조합으로 조건절을 이끌어 내며 그에 따른 수행상태가 변하게 됩니다.
여기에 주로 사용되는 예약어로는 if문과 else문으로 둘을 조합해서 사용할 수도 안 할 수도 있습니다.

if문의 구조.

```
if( 조건수식 ) {
    문장
}
```

일반적으로 if의 조건수식에는 비교수식이 들어갑니다.(예를 들면, $x > y$ 또는 $c == 6$ 등). 만약 수식의 값이 참이면 (즉, x 가 y 보다 크다, c 의 값이 6이다) 수식의 값이 1이 되고 그 밑의 문장이 수행됩니다. 참이 아니면, 그 밑의 문장은 무시되고 그 다음으로 수행이 옮겨집니다. 앞에서 모든 수식은 값을 가진다고 하였습니다. 이 때에도 if에 연결된 수식이 참이면 1의 값을 갖고, 거짓이면 0을 갖습니다.

조건수식이 참을 수행하는 문장은 하나 이상일 수도 있습니다. 그 때는 물론, 다음과 같이 괄호{}로 묶어주고 복합 명령문이라고 합니다.

```
if(score > big)           // 단순 명령문
    result =1;

if(value > 100) {          // 복합명령문
    num++;
    result =100;
}
```

if와 else의 구조문.

```
if( 조건수식 )
    수행할 문장.
else
    수행할 문장.
```

if 만 쓰일 때는 수식이 거짓일 경우 아무것도 하지 않았습니다. 하지만, if-else문의 경우는 수식이 거짓이면 else 밑의 문장을 수행합니다. 다시 말해서 수식이 참이냐, 거짓이냐에 따라서 두 가지 문장중의 하나를 선택할 수 있다는 것입니다.

■ 다중선택의 if와 else문의 구조문.

```

if( 조건수식 )
    수행할 문장 A.
else if( 조건수식 )
    수행할 문장 B.
Else
    수행할 문장 C.

```

첫줄에서 if문의 수식이 참일 경우 수행할 문장 A를 수행하고, 그렇지 않고 else if문의 수식이 참이면 수행할 문장 B를 수행하며, 첫번째 두번째 수식 모두 거짓일 경우 마지막 문장 C가 수행됩니다.

■ 관계연산자와 수식.

관계연산자는 주로 두개의 수식값을 비교하기 위해서 사용합니다. 이 비교수식에서 사용되는 **변수의 데이터형은 정수형, 실수형, 문자열 비교등이 사용되며, 가능하면 두개의 데이터형을 동일하게 사용하는 것이 안정성 있게 사용할 수 있습니다.** 예를 들어서 정수형의 데이터와 실수형의 데이터를 비교하게 되면 계산시의 오차로 인하여 ==, !=, >=, <=등의 연산을 알아보기 어렵게 되는 경우가 발생 할 수 있습니다. 특히 주의하여 주시기 바랍니다.

관계연산자(비교연산자)로 사용되는 기호로는 아래와 같습니다.

A < B	: A가 B보다 작으면
A <= B	: A가 B보다 작거나 같으면
A == B	: A와 B가 같으면
A >= B	: A가 B보다 크거나 같으면
A > B	: A가 B보다 크면
A != B	: A가 B와 같지 않다면.

관계수식에서 비교연산의 결과가 참인 경우만 해당 if 또는 else if의 수행문장을 수행하게 되는데, 참이라는 의미는 어떤 연산 결과값이 0이면 거짓이고 그렇지 않고 0이 아닌 경우 참이 되는 것입니다.

위의 연산자에 A와 B에는 정수값, 실수값, 문자열값 등이 올 수 있습니다. 예를 들어서,

```

if( a == 100 )          // a값이 100이면
// 수행...
else if( name == "SBP" ) // name의 값이 "SBP" 이면
// 수행...
else if(val >= 123.5) // val의 값이 123.5보다 크거나 같으면
// 수행...

```

그러나 문자열을 비교 할 경우에는 ==(같음), !=(다름) 관계연산자만 사용 가능합니다.

■ 논리연산자와 관계연산자

논리연산자는 위에서 설명하였듯이 &&(AND), ||(OR), !(NOT) 연산자를 말하며, 관계연산자와 혼용해서 사용할 수 있습니다.
복합적인 조건에 대하여 제어를 하고자 할 경우 사용합니다.

■ 반복문(while)

반복문 while의 사용문법은 아래와 같습니다.

```
while( 조건수식 )
```

```
    수행문장.
```

위와 같이 반복 예약어인 while문은 조건수식이 참인 경우 계속적으로 루프(LOOP)를 돌면서 수행문장을 실행합니다. 조건수식이 거짓일 경우에만 루프에서 빠져나오게 되므로 주의하지 않으면 무한루프를 돌게 되면서 실행이 중단될 수도 있습니다. 사용에 있어서 각별히 주의를 하시기 바랍니다.

while문도 여러 개의 반복 문으로 중첩이 될 수 있으며 수행해야 할 부분이 많은 경우 괄호{}를 사용하여 수행 블록을 코딩할 수 있습니다.

while문은 조건수식이 참인 경우는 정상적으로 괄호 안에 있는 문장이 수행되지만, 반대로 거짓일 경우는 한번도 수행을 하지 않을 수 있습니다.

조건수식에는 if-else구문에서와 마찬가지로 관계연산자를 사용할 수 있으며 또한 논리연산자도 사용가능 합니다.

```
int a;
string str;
a =0;
while( a < 100) {
    str =_NumToStr(a);
    _MsgBox(str, "결과", 0, 0);
    a++;
}
```

위의 구문은 정수형 변수 a를 선언하고, 문자열변수 str을 선언합니다. 그리고 변수 a에 초기값을 0으로 대입하고 루프로 들어가서 a가 100보다 작으면 괄호{}안의 문장을 수행합니다.

수행하는 첫번째 문장은 정수값 a의 현재값을 문자열로 변환해서 변수 str에 대입합니다. 그리고 결과를 화면에 메시지박스를 띄워서 보여집니다. 그리고 변수 a의 값을 1만큼 증가 시킵니다.

다시 while문으로 들어가서 조건수식을 검사하는데, 다시 a의 값이 100보다 작은지를 판단하면서 이 조건이 거짓(a가 100보다 큰 경우)이 되면 루프를 빠져나오게 됩니다.

```
int a;
string str;
a =0;
```

```
while( 1 ) {  
    if( a >= 100 ) break;  
    str =_NumToStr(a);  
    _MsgBox(str, "결과", 0, 0);  
    a++;  
}
```

위의 문장은 문장에 if 조건문과 루프를 빠져 나오게 하는 break문을 추가해서 수정하였습니다.
결과는 이전 코드와 동일한 수행결과가 나옵니다.

■ 루프의 탈출

무한루프(while)을 돌다가 적당한 조건이 맞았을 경우 루프를 빠져 나오기 위해서 break를 사용합니다. 이 break 예약어는 반드시 괄호{} 블록이 묶여있고 루프구조에서만 사용할 수 있습니다.

■ 누적치환 연산자 (+=, -=, *=, /=)

기본적으로 치환 연산자는 = 으로서 이것은 오른쪽의 값을 왼쪽의 변수에 대입하기 위해서 사용하는 연산자입니다. 이와 유사하게 변수의 값이 일정한 차이 또는 일정한 비율로 변화할 때 사용할 수 있는 간단한 연산자를 제공합니다.

아래에 여러가지 사용 예가 있습니다.

value +=20 은 value = value + 20 과 같은 의미입니다.

value -=20 은 value = value - 20 과 같은 의미입니다.

value *=20 은 value = value * 20 과 같은 의미입니다.

value /=3.5 는 value = value / 3.5 와 같은 의미입니다.

이상의 누적치환 연산자는 증가, 감소연산자와 유사합니다만, 증가, 감소 연산자는 변수의 값을 1만큼씩만 변화 시킨다는 점이 다릅니다.

제 7 장 태그관리

7.1 소개 및 특징

설비공정의 감시와 기계의 각종 센서들의 움직임을 실시간으로 수집해서 관제시스템에 반영합니다. 이것이 SCADA 시스템의 주된 임무 중 하나입니다. 이렇게 실시간으로 변하는 상태를 데이터화 해서 표현하는 일종의 변수를 태그라고 합니다.

태그는 여러 가지 다양한 형태의 데이터를 다루기 위한 기본요소가 되며, 디바이스로부터의 통신 인터페이스를 통하여 기계 상태 값을 실시간으로 감시할 수 있는 실태그와 프로젝트내에서 보다 직관적이고 다양한 동작상태를 처리할 수 있는 가상태그를 지원합니다.

가상태그는 I/O디바이스(실제 물리적으로 연결된 각종 장치)의 상태값을 나타내는 실태그와는 별도로 사용자가 임의로 만든 태그(데이터)입니다. 보통 스크립트에서 사용하는 전역 변수라고 생각할 수 있습니다.

번호	이름	태그종류	디바이스	어드레스	가상	설명
1	Status_00	아날로그			<input checked="" type="checkbox"/>	프로그램상태정보_00
2	Status_01	아날로그			<input checked="" type="checkbox"/>	

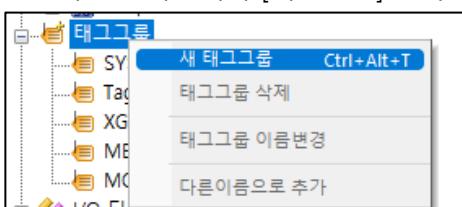
디지털 태그, 아날로그 태그, 문자열 태그, 경보태그 4가지 종류의 태그를 지원하며 경보태그를 제외하고 모두 실태그, 가상태그 형태를 지원합니다.

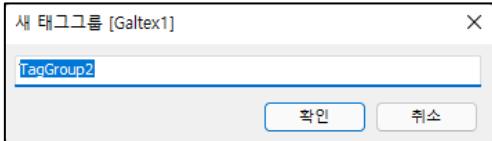
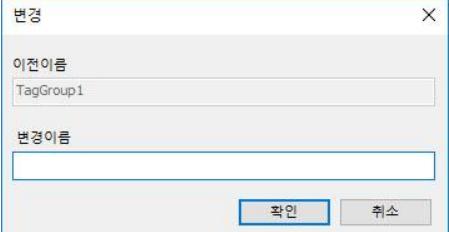
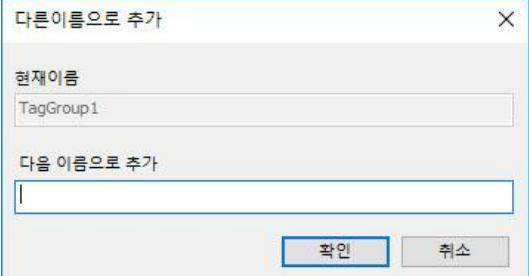
- 스튜디오에서의 태그 생성, 삭제, 복사 편집이 편리합니다.
- 디지털, 아날로그, 문자열, 경보 태그
- 실태그, 가상태그
- 아날로그 실태그에 계산식 반영가능
- 마지막 상태값 저장기능
- MS EXCEL을 이용한 손쉬운 대용량 태그편집기능 제공
- PLC값에 따라 문자열태그 텍스트 자동변경
- 태그값에 따라 스크립트 함수호출

7.2 태그그룹 생성하기

개별태그는 태그그룹이 있고 그 태그그룹내에 태그이름으로 관리됩니다. 그래서 태그를 하나 만들기 위해서는 태그그룹을 우선 생성한 후에 생성 할 수 있습니다.

프로젝트 관리창에서 [태그그룹] 항목을 마우스 우클릭으로 선택하면 태그관리 팝업메뉴가 나타납니다.



항목	설명
새 태그그룹	태그그룹 하나를 생성하고 프로젝트에 추가합니다. 
태그그룹 삭제	해당 태그그룹을 프로젝트에서 삭제합니다. 태그그룹에 있는 모든 태그도 함께 삭제됩니다. 
태그그룹 이름변경	태그그룹의 이름을 변경합니다.  프로젝트에서 태그명 사용시 [태그그룹.태그명] 형태로 사용됩니다. 그러므로 태그그룹 이름이 변경되면 [태그명 일괄변경] 기능을 사용해서 변경된 태그명을 프로젝트에 적용해서 업데이트 해주어야 합니다.
다른이름으로추가	현재 선택한 태그그룹을 다른이름으로 저장합니다.  동일한 사양의 I/O디바이스가 2대 이상일 경우 태그속성은 그대로 두고 태그그룹 이름만 변경합니다.

다음과 같이 태그그룹 리스트창이 추가되고 태그등록 및 편집이 가능합니다. 태그그룹은 각 파일단위로 구성되고 파일명이 태그그룹명이 됩니다.

Galtex1:TagGroup1 Galtex1:TagGroup2						
번호	이름	태그종류	디바이스	어드레스	가상	설명

태그를 생성하는 방법은 태그그룹 리스트창에서 마우스 우클릭하면 나타나는 태그편집 팝업메뉴에서 [새 태그] 메뉴를 선택하거나 태그그룹 관리도구바에서 [새 태그] 아이콘을 누르면 됩니다.

태그그룹 관리도구바	마우스 오른쪽 버튼 클릭시 팝업메뉴
	

이렇게 태그를 추가하면 아래 리스트 예시와 같이 표시되고 각각의 태그를 편집합니다.

번호	이름	태그종류	디바이스	어드레스	가상	설명
1	NAME0	아날로그			<input checked="" type="checkbox"/>	
2	NAME1	아날로그			<input checked="" type="checkbox"/>	
3	NAME2	아날로그			<input checked="" type="checkbox"/>	
4	NAME3	아날로그			<input checked="" type="checkbox"/>	
5	1	아날로그	USER	C2:1	<input type="checkbox"/>	
6	2	아날로그	USER	C2:2	<input type="checkbox"/>	
7	DATA1	문자열	USER	C2:3	<input type="checkbox"/>	
8	DATA1_CNT	아날로그	USER	C2:9	<input type="checkbox"/>	
9	DATA2	문자열			<input checked="" type="checkbox"/>	
10	VALUE1	아날로그			<input checked="" type="checkbox"/>	Pressure
11	VALUE2	아날로그			<input checked="" type="checkbox"/>	Temperature
12	VALUE3	아날로그			<input checked="" type="checkbox"/>	Humidity
13	VALUE4	아날로그			<input checked="" type="checkbox"/>	Beat
14	STRING	문자열			<input checked="" type="checkbox"/>	문자열 태그입니다
15	STRING1	문자열			<input checked="" type="checkbox"/>	문자열 태그입니다
16	ALM	경보			<input checked="" type="checkbox"/>	경보태그 시험
17	ALM1	아날로그			<input checked="" type="checkbox"/>	
18	ALM2	아날로그			<input checked="" type="checkbox"/>	경보태그 시험
19	ALM3	아날로그			<input checked="" type="checkbox"/>	경보태그 시험
20	TIME1	아날로그			<input checked="" type="checkbox"/>	DO COMMENT.
21	TIME2	아날로그			<input checked="" type="checkbox"/>	DO COMMENT.
22	TRG1	아날로그			<input checked="" type="checkbox"/>	트리거

항목	설명
이름	태그이름을 32문자 이내(영문,숫자,한글)로 입력합니다. 영문, 한글, 숫자사용 가능하나 특수문자는 사용불가 입니다. (₩ / : * ? " < > ! @ # \$ % ^ & * ())

태그종류	디지털 : On / Off 상태 값. 아날로그 : 연속적인 변화량에 대한 디지털 표시값. 문자열 : 텍스트 위주의 표시값. 경보 : 기준치와 상이한 상황에 대한 이벤트 처리.
디바이스	프로젝트에 등록된 I/O 디바이스 이름은 입력합니다. 디바이스 입력란에 콤보버튼을 누르면 현재 프로젝트에 등록된 I/O 디바이스 리스트가 표시됩니다. [실태그 디바이스 설정방법] 참고하세요.
어드레스	설정한 I/O 디바이스의 통신 어드레스. 설정규칙은 [실태그 디바이스 설정방법] 참고하세요.
가상태그	프로젝트의 전역변수로 사용할 경우 가상태그로 설정합니다.
설명	태그에 대한 직관적인 설명 텍스트를 입력합니다. 이 부분은 그래프의 펜에 대한 범례표시, 경보상태 창에서 경보설명 부분으로 사용되기에 잘 활용하면 편리합니다.

■ 태그 라이선스 정보

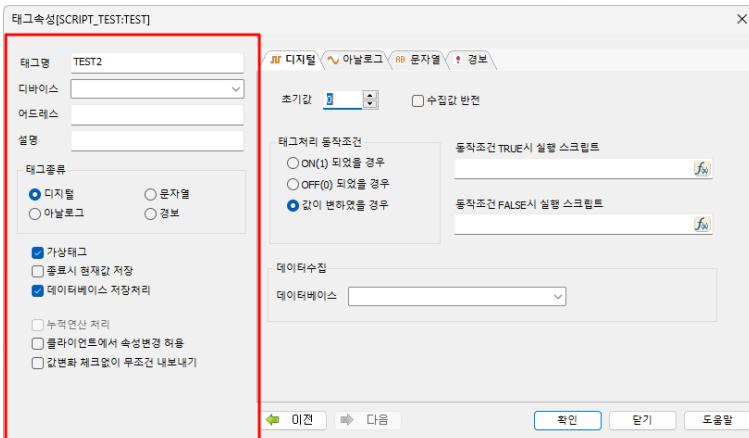
태그리스트 창 하단부에는 다음과 같이 태그용량 정보표시가 됩니다.

개발용태그: 65535	실행용태그: 65535	등록태그: 29	사용가능: 65506	태그수: 17	선택: 1
--------------	--------------	----------	-------------	---------	-------

- 개발용태그 : 라이센스에 등록되어 있는 개발자 태그 수를 표시합니다.
- 실행용태그 : 라이선스에 등록되어 있는 실행용 태그 수를 표시합니다.
- 등록태그 : 작업중인 프로젝트에 등록된 태그 수를 표시합니다.
- 사용가능 : 프로젝트에 추가 등록할 수 있는 태그 수를 표시합니다.
- 태그수 : 활성화된 태그리스트 창의 태그 수를 표시합니다.
- 선택 : 선택된 태그 수를 표시합니다.

7.3 태그종류

태그편집 팝업메뉴에서 [태그속성]을 선택하면 태그의 종류에 따라서 속성을 설정합니다.



위 그림의 적색 박스부분은 태그종류와 관계없이 공통으로 입력하는 부분입니다.

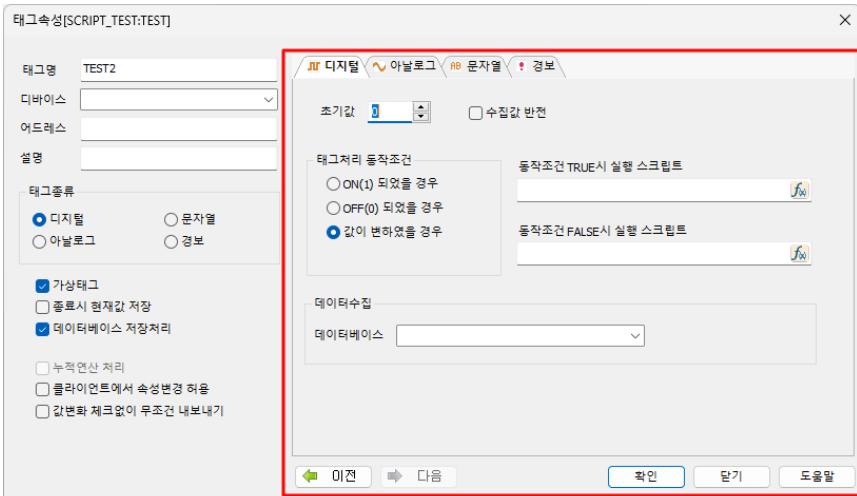
태그종류에 따라서 오른쪽 편에 속성 설정창이 변경됩니다.

항목	설명
종료시 현재값 저장	프로젝트 실행한 후 종료를 할 때의 태그값을 저장합니다. 프로젝트 재실행 할 때 저장된 값이 다시 나타납니다. 상태유지 값 저장용 태그로 사용합니다.
데이터베이스 저장처리	eRunDB에 저장.
누적연산처리	아날로그 태그에 사용하며, 수신된 값을 합산해서 적산용 태그로 사용합니다.
클라이언트에서 속성변경 허용	프로젝트 실행중에 스크립트 함수 _SetTagInfo()를 이용해서 태그속성을 변경할 수 있습니다. 주로 경보기준값 변경시 사용합니다.
값변화 체크없이 무조건 내보내기	태그값의 변화폭과 관계없이 무조건 I/O서버로 현재 태그값을 송출합니다. 기본적으로 I/O서버 통신트래픽을 줄여서 전체 실행성능을 높이기 위해서 태그값의 변화가 없는 경우 디바이스 송출을 하지 않도록 합니다. 그러나 값변화와 무관하게 무조건 내보내야 하는 경우도 있습니다.

7.3.1 디지털태그

디지털 태그는 비연속적인 상태 즉, 1과 0의 상태만을 나타내기 위해서 사용합니다.

센서의 ON상태, OFF상태를 저장하기 위해서 사용하면 1바이트의 저장공간을 갖습니다.

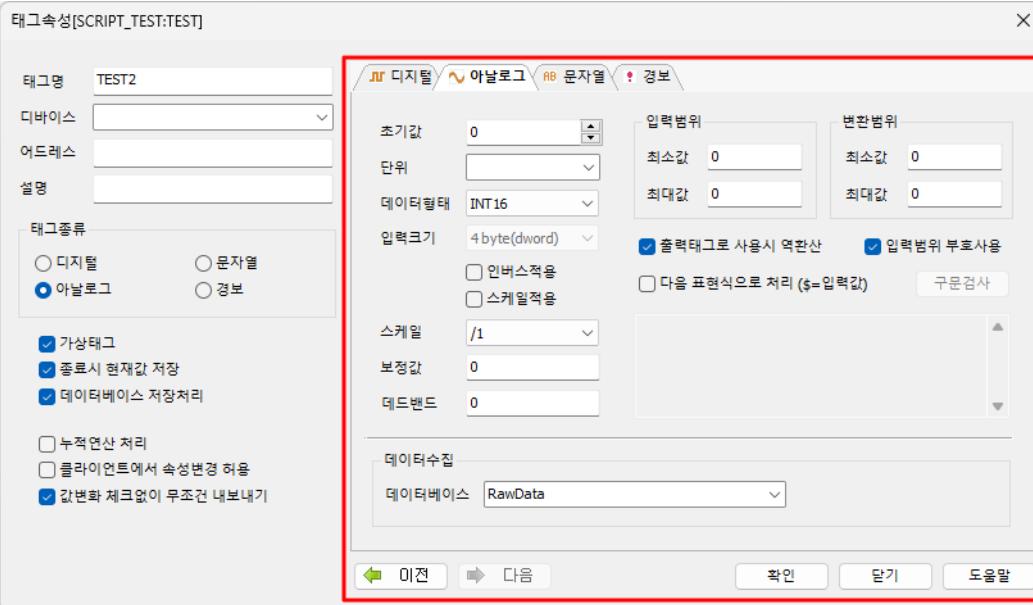


항목	설명
초기값	태그의 초기값 (초기:0)
수집값 반전	수신된 값을 반전시킵니다. (0<->1)
태그처리 동작조건	<p>디지털 태그는 ON/OFF 이벤트 수신용으로 사용할 경우 특정 비트가 살았을 때 스크립트 함수를 호출합니다. 스크립트 함수는 디지털 태그속성창 하단부에 입력합니다.</p> <ul style="list-style-type: none"> ■ ON(1) 되었을 때 함수실행 ■ OFF(0) 되었을 때 함수실행 ■ 값이 변했을 때 함수실행
데이터베이스 수집	태그 데이터를 DB로 저장하고자 하는 데이터베이스를 선택합니다. 이것은 이미 프로젝트에서 등록되어 있는 데이터베이스를 선택합니다. '데이터베이스 저장처리' 항목이 설정되어야 활성화됩니다.
동작조건	<p>태그처리 동작조건 설명을 참고하세요. 태그처리 동작조건에서 선택한 값이 들어온 경우 TRUE가 되고, 그렇지 않을 경우 FALSE가 됩니다. 함수선택이 가능한 입력형 콤보박스이며 우측끝의 함수선택 버튼을 누르면 프로젝트에 작성된 함수리스트 창이 나오며 선택합니다.</p>
이전, 다음버튼	이전태그 또는 다음태그로 선택을 변경합니다.

7.3.2 아날로그 태그

아날로그 태그는 연속적인 데이터의 변화량을 디지털 값으로 나타내기 위해서 사용합니다.

센서 또는 측정기기의 지시치, 변화폭, 변환값 등을 수치로 표현하는데 정수형, 실수형, 지수형의 크기를 갖으며 최소 2바이트 크기부터 최대 8바이트 크기의 저장공간을 사용합니다.



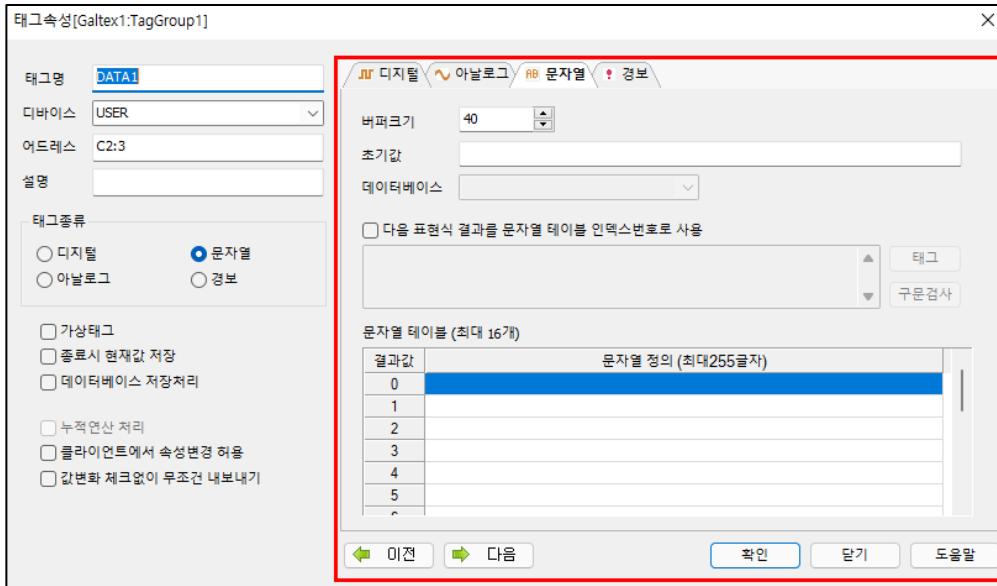
항목	설명
초기값	태그의 초기값 (초기:0)
단위	태그 값을 감시화면에 표시할 때 태그값의 단위표기 기본 단위목록이 있고, 사용자가 추가 입력 가능 합니다.
데이터 형태	아날로그 태그의 저장크기를 선택합니다. INT16 : 정수형 (-32,767 ~ 32,768) UINT16 : 부호 없는 정수형 (0 ~ 65,535) INT32 : 긴 정수형 (-2 ³¹ ~ 2 ³¹ -1) UNIT32 : 부호가 없는 긴 정수형 (0 ~ 2 ³² -1) INT64 : 64비트 긴 정수형 (-2 ⁶³ ~ 2 ⁶³ -1) UNIT64 : 부호가 없는 64비트 긴 정수형 (0 ~ 2 ⁶⁴ -1) REAL : 8바이트 Double형 (-10 ³⁰⁸ ~ 10 ³⁰⁸) FLOAT : 4바이트 IEEE 754 표준 부동 소수점형 (-10 ³⁸ ~ 10 ³⁸)
입력크기	데이터 형태를 REAL 또는 FLOAT으로 설정한 경우 PLC로부터 읽어야 하는 데이터의 바이트 길이를 선택합니다. 이것은 PLC의 통신설정에 따라야 하기 때문에 상황에 맞게 설정을 해주어야 합니다.
인버스 적용	입력 값의 상위, 하위 자리 수를 서로교환해서 계산합니다.
스케일 적용	입력 값을 나누거나 곱하거나 해서 계산하도록 합니다. 스케일 적용옵션을 선택하고 스케일 항목을 선택합니다. 보통 PLC통신할 때 실수형 데이터를 읽거나 쓸 때 사용합니다.

보정값	수신된 데이터에 더하기 빼기 값을 넣어줍니다. 보정값 스케일등이 끝나고 마지막에 계산됩니다.
데드밴드	직전값과의 변화폭을 입력합니다. 입력된 변화폭 만큼의 차이가 없을 경우 태그 값 저장 할 필요 없습니다.
입력범위	<p>통신으로 수신된 아날로그값을 변환하는 방법을 지정합니다.</p> <p>■ 입력범위 vs 변환범위 비율로 계산 수신된 최소, 최대값을 ENG(변환)값의 최소, 최대 범위로 환산.</p> <p>■ 표현식으로 계산 수신된 수치값에 비례식 또는 산술식을 적용해서 계산합니다. 예) \$ / 100.0 >> 4; -> 현재값을 100으로 나누고 오른쪽으로 4bit 이동한 값 ※ \$ -> 통신으로 수신된 소스값.</p>
출력태그로 사용시 역환산	태그값을 통신으로 내보낼 때 역계산을 합니다. (default :TRUE) 즉, 변환값 -> 입력값으로 반대로 계산합니다.
입력범위 부호사용	입력값에 대해서 부호를 사용해야 할 때 선택합니다. (default:TRUE)
데이터베이스 수집	태그 데이터를 DB로 저장하고자 하는 데이터베이스를 선택합니다. 이것은 이미 프로젝트에서 등록되어 있는 데이터베이스를 선택합니다. '데이터베이스 저장처리' 항목이 설정되어야 활성화 됩니다.
이전, 다음버튼	이전태그 또는 다음태그로 선택을 변경합니다.

7.3.3 문자열 태그

문자열 태그는 텍스트 위주의 메시지 표시 값을 저장하고 나타내기 위해서 사용합니다.

최대 256개의 문자 크기의 저장공간을 사용합니다.

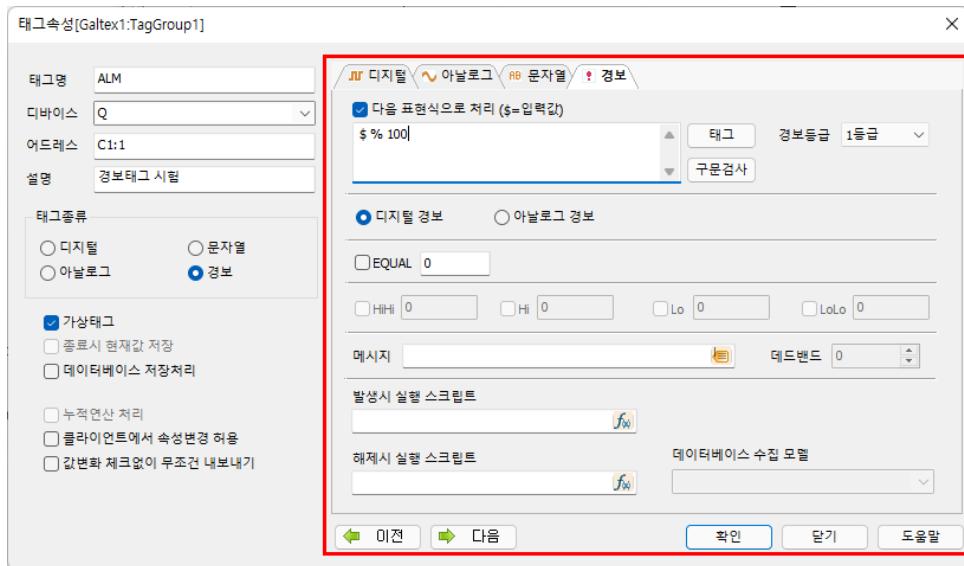


항목	설명
버퍼크기	문자열 태그의 저장크기 (max:256)
초기값	초기 시작 표시문자열
데이터베이스 수집	태그 데이터를 DB로 저장하고자 하는 데이터베이스를 선택합니다. 이것은 이미 프로젝트에서 등록되어 있는 데이터베이스를 선택합니다. '데이터베이스 저장처리' 항목이 설정되어야 활성화 됩니다.
표현식	문자열 테이블에서 사용할 인덱스 번호를 계산하도록 합니다. 태그명 또는 산술 표현식을 입력해서 계산합니다.
문자열 테이블 (최대 16개)	최대 16개의 문자열 테이블을 정의하고 태그값에 따라 문자열을 표시하도록 하기위해서 사용합니다. 위에 있는 표현식의 결과값으로 선택됩니다.
이전, 다음버튼	이전태그 또는 다음태그로 선택을 변경합니다.

7.3.4 경보태그

경보태그는 기준치와 상이한 값이 들어올 때 처리되는 태그입니다.

2종류 형식의 경보태그 종류가 있는데 ON 또는 OFF 경보와, 기준치보다 내려갔을 때, 반대로 기준치보다 올라갔을 때 저장처리 됩니다.



항목	설명
표현식	태그명 또는 산술 표현식을 입력해서 계산하고 결과값을 기준치와 비교계산합니다. ※ \$ -> 통신으로 수신된 소스값.
경보등급	본 태그의 경보등급을 분류합니다.
경보종류	■ 디지털 경보 : ON / OFF 상태경보. ■ 아날로그 경보 : 아날로그값의 하한, 상한경보
EQUAL	디지털 경보인 경우 기준치 입력합니다. 0으로 해두면 표현식의 결과가 0일 경우 경보발생 처리합니다.
상하한값	아날로그 경보인 경우 기준치 입력합니다. HiHi : 2차 상한 기준치 Hi : 1차 상한 기준치 Lo : 1차 하한 기준치 LoLo : 2차 하한 기준치
메시지	경보발생처리 할 때 저장할 메시지를 입력합니다. 태그명을 입력하면 태그값이 메시지로 저장됩니다. 태그명은 @XXX.XXX 와 같이 @문자를 붙이면 태그로 인식합니다.
데드밴드	직전 값과의 변화폭을 입력해서 입력된 변화폭만큼의 차이가 날 경우 경보발생 처리합니다.
발생시 실행 스크립트	기준치에 의해 경보가 발생하면 호출할 스크립트 함수. 주) 호출되는 함수에서 처리되는 내용은 태그값 변경등의 간단한 내용의 실행함수를 사용권유 합니다.
해제시 실행 스크립트	해당 태그의 경보해제시 호출할 스크립트 함수.
데이터베이스 수집	데이터베이스 관리에 등록된 eRunDB 이름을 입력합니다. 콤보버튼 눌러서 선택할 수 있습니다.
이전, 다음버튼	이전태그 또는 다음태그로 선택을 변경합니다.

7.3.5 시스템 태그

프로젝트 생성할 때 시스템 태그그룹이 자동으로 추가됩니다. 편집 및 추가, 삭제는 불가능합니다.

시스템 태그는 LOCAL PC시간, 서버연결, 현재 발생한 경보 수, 프로젝트 운영과 관련된 상태 태그로 구성됩니다.

번호	이름	태그종류	디바이스	어드레스	가상	설명
1	N_YEAR	아날로그			<input checked="" type="checkbox"/>	System Year
2	N_MONTH	아날로그			<input checked="" type="checkbox"/>	System Month
3	N_DAY	아날로그			<input checked="" type="checkbox"/>	System Day
4	N_HOUR	아날로그			<input checked="" type="checkbox"/>	System Hour
5	N_MIN	아날로그			<input checked="" type="checkbox"/>	System Minute
6	N_SEC	아날로그			<input checked="" type="checkbox"/>	System second
7	MASTER_CLIENT	아날로그			<input checked="" type="checkbox"/>	0=slave, 1=master
8	IOSVR_STAT	아날로그			<input checked="" type="checkbox"/>	0=off-line, 1=on-line
9	DBSVR_STAT	아날로그			<input checked="" type="checkbox"/>	SFDB process Status.
10	TOTAL_ALM	아날로그			<input checked="" type="checkbox"/>	Total alarm Count.
11	LOGIN_USER	아날로그			<input checked="" type="checkbox"/>	Number of users on online.
12	LICENSE	아날로그			<input checked="" type="checkbox"/>	0=no License

항목	설명
N_YEAR, N_MONTH, N_DAY N_HOUR, N_MIN, N_SEC	시스템 년도, 월, 일, 시, 분, 초
MASTER_CLIENT	VIEWER의 구분을 합니다. 1 : MASTER(LOCAL), 0 : CLIENT(JOIN) 스크립트에서 본 태그값으로 인터락을 처리할 때 사용합니다.
IOSVR_STAT	I/O서버 연결상태입니다. 1 : ON-LINE 0 : OFF-LINE
DBSVR_STAT	현재 버전에서는 사용안함.
TOTAL_ALM	현재 경보개수
LOGIN_USER	LOCAL 연결포함 및 CLIENT 연결수.
LICENSE	제품 라이선스가 있는지 표시합니다. 1 : OK 0 : NO LICENSE

7.4 실태그 어드레스 설정방법

I/O 디바이스에 따라 적당히 어드레스를 설정하면 기계 센서의 현재값들을 태그에 저장됩니다.

어드레스 설정에는 오프셋 번호만 사용하는 경우와 오프셋, 비트번호를 사용하는 경우 이렇게 2가지 설정방법이

있습니다.

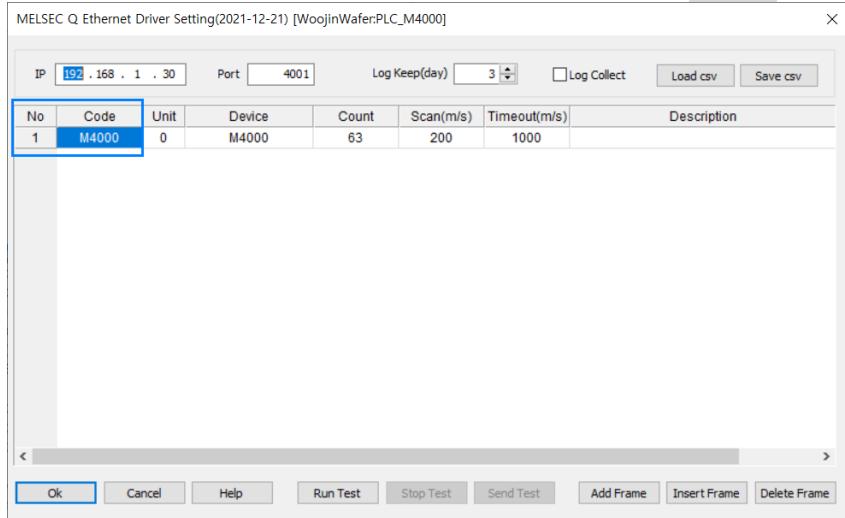
어드레스 설정 형식은 다음과 같습니다.

프레임코드 : 오프셋[비트번호]

■ 프레임코드

디바이스마다 설정된 통신프레임의 해당코드명.

보통 직관적인 이름을 등록하고 사용합니다. (아래의 예 : **M4000**)



[MELSEC PLC 디바이스]

■ 오프셋 번호만 사용하는 경우

프레임코드 : 오프셋

프레임코드 : 0 -> 해당 프레임 통신상태 (1:OK, 0:FAIL)

프레임코드 : 1 -> 해당 프레임 첫번째 데이터

TEST_MANUAL:Alarm							TEST_MANUAL:M1000							TEST_MANUAL:SYSTEM						
번호	이름	태그종류	디바이스	어드레스	가상	설명	번호	이름	태그종류	디바이스	어드레스	가상	설명	번호	이름	태그종류	디바이스	어드레스	가상	설명
1	M10000_COM	디지털	PLC	M1000:0	<input type="checkbox"/>	1:정상연결, 0:연결끊김	2	M10000	디지털	PLC	M1000:1,0	<input type="checkbox"/>		3	M10001	디지털	PLC	M1000:1,1	<input type="checkbox"/>	
4	M10002	디지털	PLC	M1000:1,2	<input type="checkbox"/>		5	M10003	디지털	PLC	M1000:1,3	<input type="checkbox"/>								

■ 오프셋, 비트번호 사용하는 경우

프레임코드 : 오프셋,비트번호

프레임코드 : 1,0 → 첫번째 오프셋값의, 0번 비트값

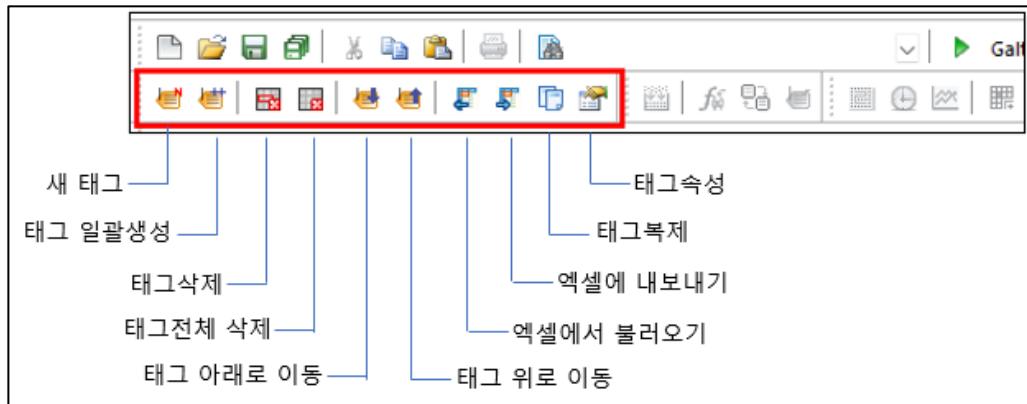
프레임코드 : 1,15 → 첫번째 오프셋값의, 15번 비트값

번호	이름	태그종류	디바이스	어드레스	가상
1	X1000	디지털	PLC	X1000:1.0	<input type="checkbox"/>
2	X1001	디지털	PLC	X1000:1.1	<input type="checkbox"/>
3	X1002	디지털	PLC	X1000:1.2	<input type="checkbox"/>
4	X1003	디지털	PLC	X1000:1.3	<input type="checkbox"/>
5	X1004	디지털	PLC	X1000:1.4	<input type="checkbox"/>
6	X1005	디지털	PLC	X1000:1.5	<input type="checkbox"/>
7	X1006	디지털	PLC	X1000:1.6	<input type="checkbox"/>
8	X1007	디지털	PLC	X1000:1.7	<input type="checkbox"/>
9	X1008	디지털	PLC	X1000:1.8	<input type="checkbox"/>
10	X1009	디지털	PLC	X1000:1.9	<input type="checkbox"/>
11	X100A	디지털	PLC	X1000:1.10	<input type="checkbox"/>
12	X100B	디지털	PLC	X1000:1.11	<input type="checkbox"/>
13	X100C	디지털	PLC	X1000:1.12	<input type="checkbox"/>
14	X100D	디지털	PLC	X1000:1.13	<input type="checkbox"/>
15	X100E	디지털	PLC	X1000:1.14	<input type="checkbox"/>
16	X100F	디지털	PLC	X1000:1.15	<input type="checkbox"/>

7.5 태그편집 도구바

태그관리 도구바 메뉴는 다음과 같습니다.

태그 그룹리스트 창이 활성화 상태일 때 선택 가능합니다.



새 태그

새로운 태그 1개를 추가등록 합니다. 마지막 태그의 속성에 연속성을

태그 일괄생성

한번에 동일한 태그종류, 디바이스, 어드레스를 여러 개 추가할 때 사용합니다. 한 번에 추가하고자 하는 태그개수를 최소 1개 이상 입력합니다. 상세설명은 뒷장의 [태그 일괄생성]부분 참고하세요.

태그삭제

현재 선택된 태그 1개 이상의 태그를 삭제합니다.

태그전체 삭제

태그 전체를 삭제합니다.(태그 전체 삭제에는 삭제할 태그 수량이 같이 표시됩니다.)

아래로 이동

선택된 태그를 아래에 있는 태그와 순서를 바꿉니다.

위로 이동

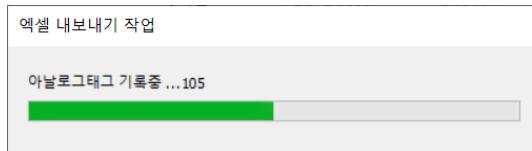
선택된 태그를 위에 있는 태그와 순서를 바꿉니다.

불러오기

MS-EXCEL파일을 현재 태그그룹에 불러와서 덮어쓰기 합니다. 주의할 점은 현재 태그그룹에 등록 되어있는 태그가 있다면 모두 삭제되니 주의하시기 바랍니다.

내보내기

현재 태그그룹 내용을 MS-EXCEL 파일형식으로 저장합니다. 대량의 태그관리를 편리하게 합니다.



다음과 같이 엑셀파일 형식으로 저장합니다.

1	태그이름	태그종류	디바이스	어드레스	가상	마지막값저장	DB처리	누적연산	속성변경허용	무조건내보내기	설명
2	D6000	1	PLC_D6000	D6000.1	0	0	0	0	0	0	
3	D6001	1	PLC_D6000	D6000.2	0	0	0	0	0	0	
4	D6002	1	PLC_D6000	D6000.3	0	0	0	0	0	0	
5	D6003	1	PLC_D6000	D6000.4	0	0	0	0	0	0	
6	D6004	1	PLC_D6000	D6000.5	0	0	0	0	0	0	
7	D6005	1	PLC_D6000	D6000.6	0	0	0	0	0	0	
8	D6006	1	PLC_D6000	D6000.7	0	0	0	0	0	0	
9	D6007	1	PLC_D6000	D6000.8	0	0	0	0	0	0	
10	D6008	1	PLC_D6000	D6000.9	0	0	0	0	0	0	
11	D6009	1	PLC_D6000	D6000.10	0	0	0	0	0	0	
12	D6010	1	PLC_D6000	D6000.11	0	0	0	0	0	0	

태그복제

태그 속성까지 복사합니다.

태그를 복사 (Ctrl + C) 및 붙여넣기 (Ctrl + V)시 선택한 태그에 대해서 태그복제가 설정되어 있으면 속성까지 복사하고 태그복제가 설정되어 있지 않으면 선택한 문자열만 복사합니다.

태그속성

태그 속성창을 띄웁니다.태그속성에 대한 설명은 7.3 태그종류에 상세히 기술되어 있습니다.

7.6 태그 일괄생성 하기

한번에 동일한 태그종류, 디바이스, 어드레스를 여러 개 추가할 때 사용합니다.

태그 일괄 생성

태그수: 10 태그수는 1이상 1024 이하로 입력합니다.

태그이름:

태그종류: 디지털 아날로그 문자열 경보

데이터형태: INT16

디바이스:

프레임명: 사용
 고정 증가

오프셋: 사용
 고정 증가

비트번호: 사용
 고정 증가

가상태그
 종료시 현재값 저장
 누적연산 처리
 클라이언트에서 속성변경 허용
 값변화 체크없이 무조건 내보내기
 데이터수집
 LOADCELL

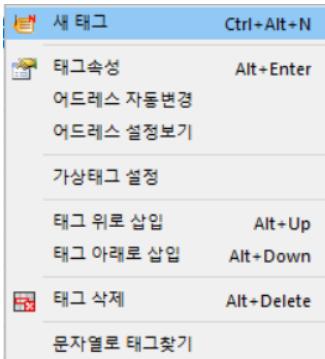
설명: DO COMMENT...

생성 취소 도움말

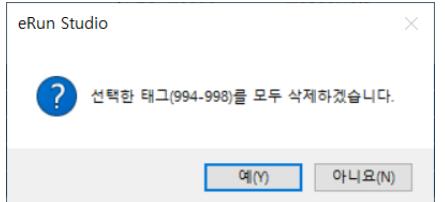
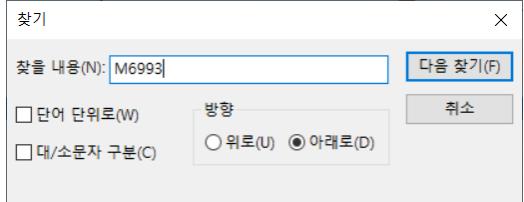
항목	설명
태그수	추가하려는 태그 수를 입력합니다. 디바이스와 어드레스 계산해서 개수를 맞추면 편리합니다.
태그이름	태그명 입력합니다. 태그명 뒤에 일련번호가 자동으로 추가됩니다.
태그종류	태그종류를 선택합니다.
디바이스	I/O 디바이스 이름을 선택합니다.
프레임명	실태그인 경우 프레임명을 입력합니다. 사용체크 해주면 오프셋, 비트번호 입력 부분이 활성화됩니다.
오프셋	프레임의 오프셋 시작번호를 입력합니다. 아날로그 태그인 경우는 [증가] 속성을 선택합니다.
데이터 형태	아날로그 태그인 경우 데이터형을 선택합니다.

7.7 태그편집 팝업메뉴

태그 편집창에서 태그를 선택하고 마우스 우클릭하면 팝업창이 나타납니다.



항목	설명
새 태그	태그 하나를 추가합니다.
태그속성	태그 속성창을 띄웁니다. 태그 속성에 대한 설명은 7.3 태그종류에 상세히 기술되어 있습니다
어드레스 자동변경	<p>다수개의 태그를 선택한 상태에서 제일 첫태그의 어드레스 오프셋, 비트번호를 시작으로 자동증가 시킵니다.</p> <p>다음과 같이 변경됩니다.</p>
어드레스 설정보기	선택 태그의 디바이스 설정화면 표시합니다.
가상태그 설정	가상태그 설정 또는 해제합니다.
태그 위로 삽입	현재 선택된 태그의 속성을 복사해서, 선택태그 바로 위로 한개의 태그를 삽입추가 합니다. 태그의 선택 위치는 변경되지 않습니다.
태그 아래로 삽입	현재 선택된 태그의 속성을 복사해서, 선택태그 바로 아래로 한개의 태그를 삽입추가 합니다. 태그의 선택 위치는 변경되지 않습니다.

태그삭제	1개 이상의 태그를 선택하고 삭제합니다. 
문자열로 태그 찾기	현재 태그그룹 창에서 특정태그를 찾을 때 사용합니다.  태그위치로 이동합니다.

7.8 태그명 일괄변경

프로젝트 편집중에 태그명을 변경하거나 불필요한 태그를 삭제하는 경우가 있습니다.

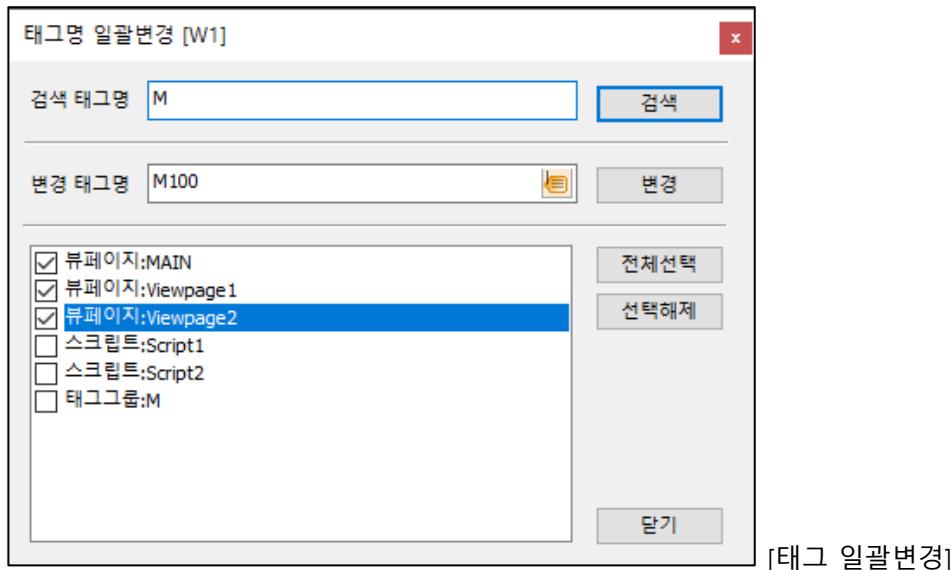
오브젝트, 스크립트, 프로젝트 속성에 사용된 태그가 이름이 변경되거나 삭제될 경우 오브젝트, 스크립트에 사용된 태그도 같이 변경해 주어야 프로젝트 빌드할 때 오류가 없습니다.

태그이름 일괄변경은 두가지 방식을 제공합니다.

- ① 프로젝트내의 모든 태그명을 한꺼번에 변경
- ② 뷰 페이지에서 선택한 오브젝트만 태그명을 변경

이와 같이 태그명을 한꺼번에 변경해야 할 경우 사용하는 기능 [태그명 일괄변경] 입니다.

스튜디오 상단 주메뉴에 [편집]-[태그명 변경하기]를 선택하거나, 단축키 CTRL+R을 눌러 기능을 사용합니다.



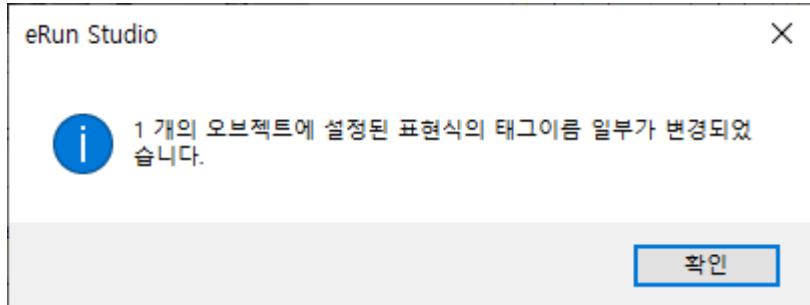
[태그 일괄변경]

항목	설명
검색 태그명	변경하고자 하는 태그명을 입력합니다. 검색 태그명 규칙은 입력한 태그명 일부 또는 전체를 변경 태그명으로 바꿉니다.
검색버튼	검색 태그명을 포함하고 있는 리스트를 모두 보여줍니다.
변경태그명	검색 태그명을 변경하려고 하는 이름을 입력합니다.
변경버튼	검색 태그명을 변경 태그명으로 모두 바꿉니다.
리스트 선택창	체크된 리스트에서만 태그이름을 변경합니다.

다음 창은 뷰 페이지에서 선택한 오브젝트만 태그명을 변경 할 경우의 창입니다.



화면제목창에는 “선택한 오브젝트 태그명변경”과 선택한 오브젝트 개수 [프로젝트 이름] 이렇게 표시됩니다. 검색 태그명을 바꾸고자 하는 태그명의 일부 또는 전체이름을 입력하고 변경태그명 역시 입력합니다. 변경을 눌러서 선택한 오브젝트의 태그명 모두를 바꿉니다,



7.9 태그리스트 엑셀파일 구성

아래의 표는 현재 편집중인 태그리스트를 [MS-EXCEL로 내보내기] 하게 되면 지정된 엑셀파일에 기록되는 셀 내용을 설명합니다.

대량의 태그를 편집할 경우 엑셀에서 아래정보를 참고하여 정확하게 작성한 후 파일을 태그관리에서 불러오기하면 쉽게 대량의 태그정보를 편집하실 수 있습니다.

셀번호	시트구분				
	기본속성	디지털태그	아날로그태그	문자열태그	경보태그
A	태그이름	태그이름	태그이름	태그이름	태그이름
B	태그종류 0 = 디지털 1 = 아날로그 2 = 문자열 3 = 경보	초기값	초기값	초기문자열	표현식사용 0 = OFF 1 = ON
C	디바이스	수집값반전 0 = OFF 일 때 1 = ON 일 때 2 = 값이 변하였을 경우	데이터형태 0 = INT16 1 = UINT16 2 = INT32 3 = UINT32 4 = INT64 5 = UINT64 6 = REAL 7 = FLOAT	길이	표현식 문자열
D	어드레스	처리조건 0 = OFF 1 = ON 2 = 값이 변하였을 경우	입력크기 0 = 1 바이트 1 = 2 바이트 2 = 4 바이트 3 = 8 바이트	데이터베이스 이름	경보형태 0 = 디지털 1 = 아날로그
E	가상태그 0 = OFF 1 = ON	데이터베이스 이름	보정값	표현식사용 0 = OFF 1 = ON	EQUAL 사용 0 = OFF 1 = ON

F	마지막값 저장 0 = OFF 1 = ON	동작조건 TRUE 시 스크립트 실행	인버스 0 = OFF 1 = ON	표현식 문자열	디지털경보인 경우 기준값
G	DB 처리 0 = OFF 1 = ON	동작조건 FALSE 시 스크립트 실행	스케일사용 0 = OFF 1 = ON	문자열 테이블 인덱스번호	HiHi 사용 0 = OFF 1 = ON
H	누적연산 0 = OFF 1 = ON		스케일 0 = /1 1 = /10 2 = /100 3 = /1000 4 = /10000 5 = /100000 6 = /1000000 7 = /0.1 8 = /0.01 9 = /0.001 10 = /0.0001 11 = /0.00001 12 = /0.000001	문자열 0	HiHi 기준값
I	속성변경허용 0 = OFF 1 = ON		데드밴드값	문자열 1	Hi 사용 0 = OFF 1 = ON
J	무조건 내보내기 0 = OFF 1 = ON		입력최소값	문자열 2	Hi 기준값
K	태그설명		입력최대값	문자열 3	Lo 사용 0 = OFF 1 = ON
L			변환최소값	문자열 4	Lo 기준값
M			변환최대값	문자열 5	LoLo 사용 0 = OFF 1 = ON
N			역환산 0 = OFF 1 = ON	문자열 6	LoLo 기준값
0			부호사용 0 = OFF 1 = ON	문자열 7	경보등급 0 = 1 등급 1 = 2 등급 2 = 3 등급 3 = 4 등급 4 = 5 등급



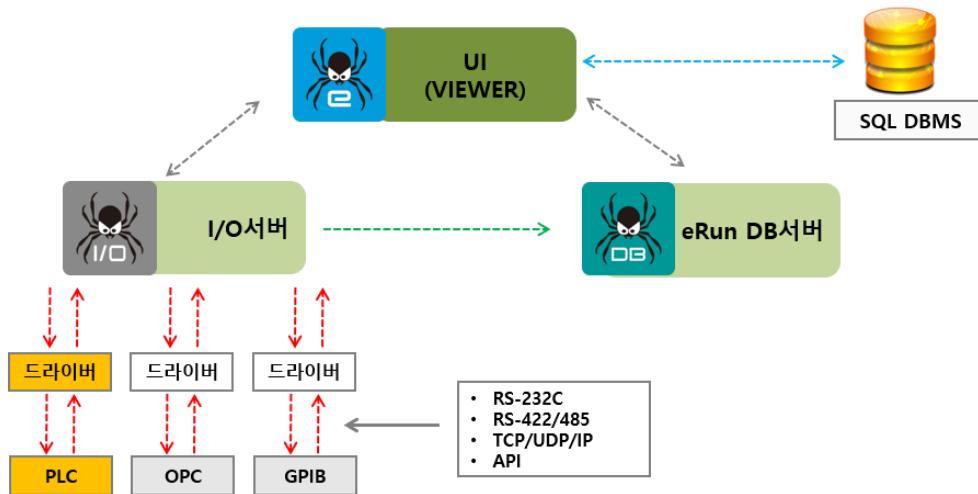
제 7 장 태그관리

P			표현식사용 0 = OFF 1 = ON	문자열 8	데드밴드
Q			표현식 문자열	문자열 9	경보문자열
R			데이터베이스 이름	문자열 10	경보발생시 스크립트 함수호출
S				문자열 11	해제시 스크립트 함수호출
T				문자열 12	데이터베이스 이름
U				문자열 13	
V				문자열 14	
W				문자열 15	

제 8 장 I/O 디바이스

산업현장에서는 다양한 장치들을 컴퓨터와 연결하여 자동으로 제어를 한다든지 아니면 현장에서 생산되는 수많은 데이터들을 컴퓨터에 저장하여 생산효율을 극대화하려고 노력하고 있습니다. 또한 자동 디바이스 장치 제조사에서는 이를 위하여 컴퓨터 통신을 위한 통신 터널(Tunnel)을 제공하게 되는데 이를 통하여 컴퓨터와의 인터페이스를 가능하게 합니다.

기본적으로 인터페이스가 가능하도록 하기 위해서는 표준 인터페이스 장치 및 소프트웨어 개념의 통신규약인 프로토콜이 제공되는데 이것은 장치 제조사별 상이한 방식으로 제공되고 있습니다. 이렇게 다양하고 상이한 장치 인터페이스를 통일된 방식의 사용자 인터페이스로 제공하고자 I/O 디바이스 드라이버가 필요하게 됩니다.



이번 장에서는 당사에서 현재 준비 되어있는 장치 통신드라이버를 eRun에 등록하고 프로젝트에 연동하는 방법을 소개합니다. 통신드라이버에 대한 내부설정 방법은 별도로 제공되는 **15장 디바이스 드라이버 설정 방법**을 참조하시기 바랍니다.

아래는 프로젝트 관리자에 I/O디바이스에 등록된 드라이버 예시입니다.

“디바이스 이름(디바이스 드라이버 파일) ; 디바이스 설명”



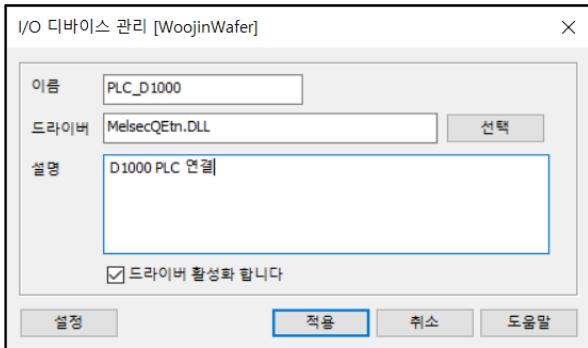
8.1 디바이스 생성하기

프로젝트 관리창에서 [I/O디바이스] 항목을 마우스 우클릭으로 선택하면 I/O디바이스 팝업메뉴가 나타납니다. 메뉴에서 [새 디바이스]를 선택하면 새로운 I/O디바이스 드라이버를 생성하고 프로젝트에 추가합니다.

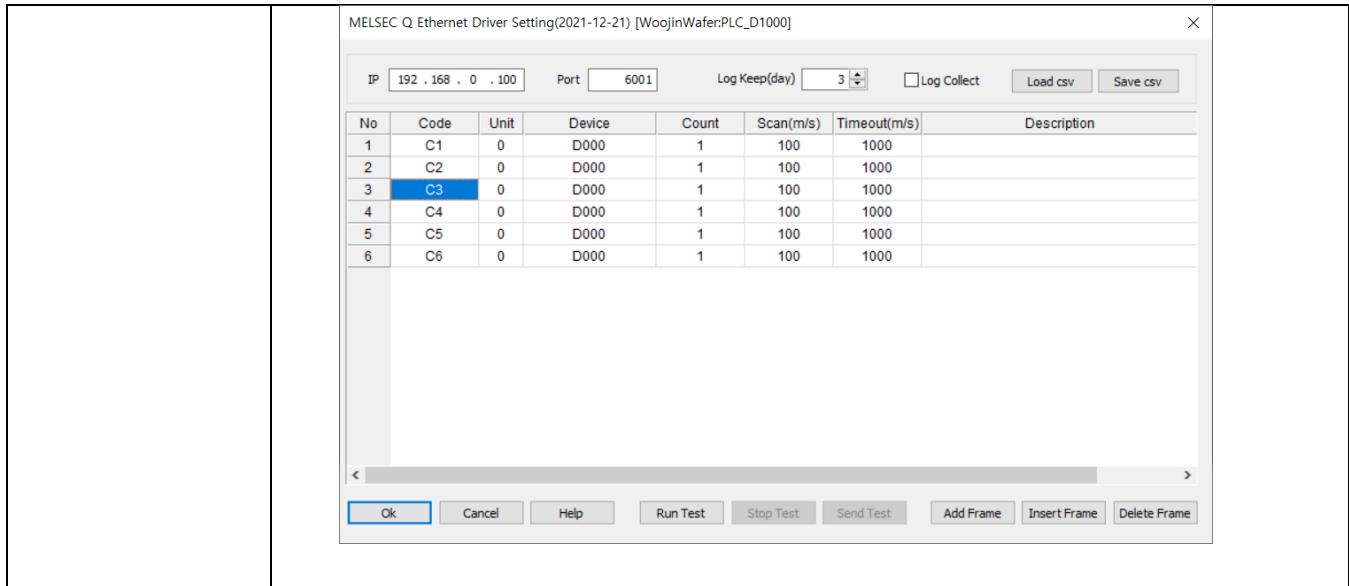


항목	설명
디바이스 활성화	프로젝트 실행할 때 디바이스 통신 활성화를 선택합니다. 비활성화가 되면 프로젝트 관리자에 디바이스 리스트 아이콘이 회색으로 바뀝니다.
새 디바이스	프로젝트에 디바이스를 생성추가합니다.
디바이스 편집	디바이스 설정 편집합니다.
이름변경	디바이스 이름을 변경합니다. 디바이스 이름은 태그에서 사용합니다.
디바이스 삭제	프로젝트에서 디바이스를 삭제합니다.
디바이스 통신시험	디바이스와 연결을 시험하고 설정된 통신 프레임이 유효한지 확인합니다.
다른이름으로 추가	동일한 드라이버를 다수개 사용할 경우 동일한 설정으로 디바이스 이름만 변경하여 추가합니다.
불러오기	다른 프로젝트에서 디바이스 설정정보를 가져올 때 사용합니다.
내보내기	현재 프로젝트에 등록된 디바이스 설정정보를 다른 프로젝트로 복사할 경우 사용합니다.

새 디바이스 추가 설정파일이 나타납니다.

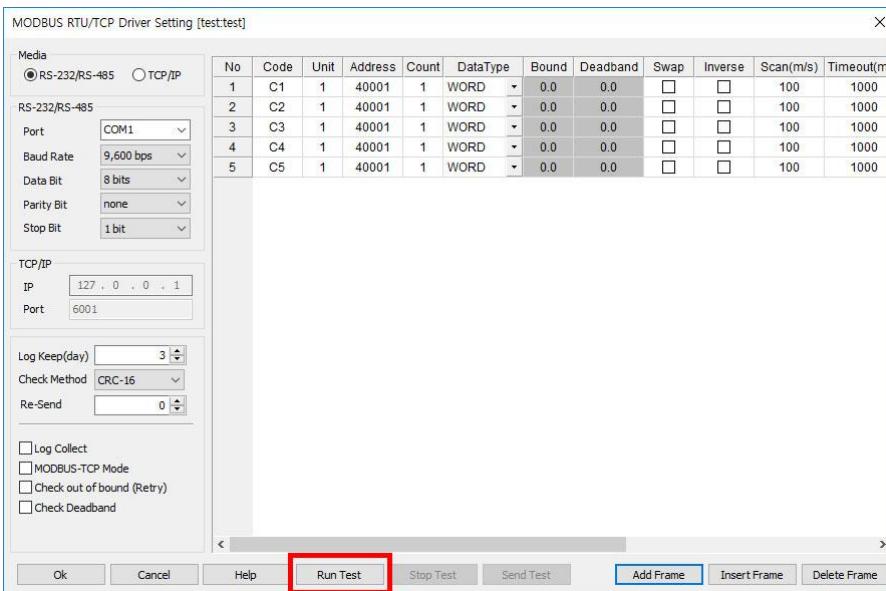


항목	설명																																																															
이름	디바이스 이름을 입력합니다. 스튜디오에서 태그속성 설정할 때 연결 디바이스명으로 사용됩니다.																																																															
드라이버	디바이스 드라이버 DLL파일을 선택합니다. 선택버튼을 누르면 아래와 같이 드라이버 리스트가 나타납니다. <div style="border: 1px solid #ccc; padding: 5px;"> <p>드라이버 리스트</p> <table border="1"> <thead> <tr> <th>드라이버 파일</th> <th>버전</th> <th>설명</th> </tr> </thead> <tbody> <tr><td>M240G.iot</td><td>3.0.1.139</td><td>스파이어스사코社 유량컴퓨터(M240G) RS-232C Serial Interface Template</td></tr> <tr><td>M700K.DLL</td><td>3.0.1.21</td><td>Spirax Sarco社 M700K Flow Indicator Serial Interface Driver</td></tr> <tr><td>MASTERKCNET.DLL</td><td>3.0.1.25</td><td>LS산전社 MASTERK CNET 통신드라이버</td></tr> <tr><td>MASTERK_FENET.DLL</td><td>3.0.1.21</td><td>LS산전社 MASTERK FENET(TCP) 통신드라이버</td></tr> <tr><td>MAVLink.DLL</td><td>2.6.0.98</td><td>MAVLink V1/V2 Serial Interface Driver</td></tr> <tr><td>MBSVR.DLL</td><td>3.0.1.38</td><td>MODBUS RTU/TCP Server Driver</td></tr> <tr><td>MELSECQ.DLL</td><td>3.0.1.6</td><td>MITSUBISHI社 MELSEC PLC Q Series Driver</td></tr> <tr><td>MelsecQEtln.DLL</td><td>3.0.1.19</td><td>MITSUBISHI MelsecQ, MC Protocol Ethernet Interface Driver</td></tr> <tr><td>MODBUS.DLL</td><td>3.2.1.1</td><td>MODBUS Driver</td></tr> <tr><td>MODBUS2.DLL</td><td>3.0.1.428</td><td>MODBUS2 Driver</td></tr> <tr><td>MODBUS3.DLL</td><td>3.0.1.70</td><td>MODBUS3 RTU/TCP, Active & Passive Driver</td></tr> <tr><td>NiceBusT4.DLL</td><td>3.0.1.22</td><td>Nice Module BUST4 Interface Driver</td></tr> <tr><td>NiceHttp.DLL</td><td>3.0.1.20</td><td>Nice Module Http Interface Driver</td></tr> <tr><td>NI_DAQmx_USB6003.DLL</td><td>3.0.1.23</td><td>National Instrument社 DAQmx USB-6003 통신드라이버</td></tr> <tr><td>NUDAM6000.DLL</td><td>3.0.1.20</td><td>ADLINK社 NUDAM-6000 시리즈 RS485통신드라이버</td></tr> <tr><td>OMRONFINS.DLL</td><td>3.0.1.37</td><td>OMRON PLC CS/CJ Fins Ethernet UDP/TCP 통신드라이버</td></tr> <tr><td>OMRONSERIAL.DLL</td><td>3.0.1.19</td><td>OMRON 시리얼 통신드라이버</td></tr> <tr><td>PCIDASK.DLL</td><td>3.0.1.62</td><td>ADLink社 PCI NuDAQ PCI Card Driver</td></tr> <tr><td>PCI_7442.DLL</td><td>3.0.1.27</td><td>ADLink社 PCI NuDAQ PCI-7442 통신드라이버</td></tr> <tr><td>SIEMENS_PLC_TCPIP.DLL</td><td>3.0.1.20</td><td>SIEMENS社 PLC_TCPIP 통신드라이버</td></tr> </tbody> </table> <p>설치된 드라이버 개수 : 49</p> <p style="text-align: right;">적용 취소</p> </div>	드라이버 파일	버전	설명	M240G.iot	3.0.1.139	스파이어스사코社 유량컴퓨터(M240G) RS-232C Serial Interface Template	M700K.DLL	3.0.1.21	Spirax Sarco社 M700K Flow Indicator Serial Interface Driver	MASTERKCNET.DLL	3.0.1.25	LS산전社 MASTERK CNET 통신드라이버	MASTERK_FENET.DLL	3.0.1.21	LS산전社 MASTERK FENET(TCP) 통신드라이버	MAVLink.DLL	2.6.0.98	MAVLink V1/V2 Serial Interface Driver	MBSVR.DLL	3.0.1.38	MODBUS RTU/TCP Server Driver	MELSECQ.DLL	3.0.1.6	MITSUBISHI社 MELSEC PLC Q Series Driver	MelsecQEtln.DLL	3.0.1.19	MITSUBISHI MelsecQ, MC Protocol Ethernet Interface Driver	MODBUS.DLL	3.2.1.1	MODBUS Driver	MODBUS2.DLL	3.0.1.428	MODBUS2 Driver	MODBUS3.DLL	3.0.1.70	MODBUS3 RTU/TCP, Active & Passive Driver	NiceBusT4.DLL	3.0.1.22	Nice Module BUST4 Interface Driver	NiceHttp.DLL	3.0.1.20	Nice Module Http Interface Driver	NI_DAQmx_USB6003.DLL	3.0.1.23	National Instrument社 DAQmx USB-6003 통신드라이버	NUDAM6000.DLL	3.0.1.20	ADLINK社 NUDAM-6000 시리즈 RS485통신드라이버	OMRONFINS.DLL	3.0.1.37	OMRON PLC CS/CJ Fins Ethernet UDP/TCP 통신드라이버	OMRONSERIAL.DLL	3.0.1.19	OMRON 시리얼 통신드라이버	PCIDASK.DLL	3.0.1.62	ADLink社 PCI NuDAQ PCI Card Driver	PCI_7442.DLL	3.0.1.27	ADLink社 PCI NuDAQ PCI-7442 통신드라이버	SIEMENS_PLC_TCPIP.DLL	3.0.1.20	SIEMENS社 PLC_TCPIP 통신드라이버
드라이버 파일	버전	설명																																																														
M240G.iot	3.0.1.139	스파이어스사코社 유량컴퓨터(M240G) RS-232C Serial Interface Template																																																														
M700K.DLL	3.0.1.21	Spirax Sarco社 M700K Flow Indicator Serial Interface Driver																																																														
MASTERKCNET.DLL	3.0.1.25	LS산전社 MASTERK CNET 통신드라이버																																																														
MASTERK_FENET.DLL	3.0.1.21	LS산전社 MASTERK FENET(TCP) 통신드라이버																																																														
MAVLink.DLL	2.6.0.98	MAVLink V1/V2 Serial Interface Driver																																																														
MBSVR.DLL	3.0.1.38	MODBUS RTU/TCP Server Driver																																																														
MELSECQ.DLL	3.0.1.6	MITSUBISHI社 MELSEC PLC Q Series Driver																																																														
MelsecQEtln.DLL	3.0.1.19	MITSUBISHI MelsecQ, MC Protocol Ethernet Interface Driver																																																														
MODBUS.DLL	3.2.1.1	MODBUS Driver																																																														
MODBUS2.DLL	3.0.1.428	MODBUS2 Driver																																																														
MODBUS3.DLL	3.0.1.70	MODBUS3 RTU/TCP, Active & Passive Driver																																																														
NiceBusT4.DLL	3.0.1.22	Nice Module BUST4 Interface Driver																																																														
NiceHttp.DLL	3.0.1.20	Nice Module Http Interface Driver																																																														
NI_DAQmx_USB6003.DLL	3.0.1.23	National Instrument社 DAQmx USB-6003 통신드라이버																																																														
NUDAM6000.DLL	3.0.1.20	ADLINK社 NUDAM-6000 시리즈 RS485통신드라이버																																																														
OMRONFINS.DLL	3.0.1.37	OMRON PLC CS/CJ Fins Ethernet UDP/TCP 통신드라이버																																																														
OMRONSERIAL.DLL	3.0.1.19	OMRON 시리얼 통신드라이버																																																														
PCIDASK.DLL	3.0.1.62	ADLink社 PCI NuDAQ PCI Card Driver																																																														
PCI_7442.DLL	3.0.1.27	ADLink社 PCI NuDAQ PCI-7442 통신드라이버																																																														
SIEMENS_PLC_TCPIP.DLL	3.0.1.20	SIEMENS社 PLC_TCPIP 통신드라이버																																																														
설명	디바이스 참고용 설명을 달아두면 편리합니다.																																																															
드라이버 활성화	프로젝트 실행할 때 I/O서버에서 디바이스 통신을 하도록 활성화 합니다.																																																															
설정	통신내용 및 프레임을 등록 편집합니다. 설정창 내용은 디바이스 드라이버의 종류에 조금씩 다릅니다. 자세한 내용은 디바이스 드라이버 설명서를 참고하세요. 예시)																																																															



8.2 디바이스 통신시험

스튜디오에서 디바이스를 여러 개 등록한 후 통신포트 및 설정이 정상적인지 시험하고, 디바이스와 연결을 시험하고 설정된 통신 프레임이 유효한지 확인합니다. 디바이스 설정창으로 들어가서 [RunTest]를 통해서 통신연결 확인도 가능하지만, 이것은 한번에 하나의 디바이스 통신시험만 가능합니다. 여러 개 디바이스 통신시험이 필요할 경우 [디바이스 통신시험] 기능을 사용하면 가능합니다. 통신시험 창은 설정창과 동일하지만, 설정내용 및 프레임등록 편집은 불가합니다.



통신시험 창에서 하단에 [Run Test] 버튼을 클릭하여 통신 시험을 합니다.



제 8 장 I/O 디바이스

The screenshot shows the 'I/O Device Monitoring' window. At the top, it displays device information: 'Device Name: test', 'Driver File: MODBUS.dll', 'Version: 2.6.0.26', 'Setting: COM1-9600-8-N-1', and 'Status: OK'. Below this is a large text area showing repeated [OUT] data frames. At the bottom, there are three tabs: 'Ascii View' (selected), 'Hex View', and 'List View', followed by 'Pause' and 'Clear' buttons.

디바이스 통신시험 Monitoring 페이지에서 통신이 제대로 되면 [OUT], [IN] 데이터가 나오고 통신이 제대로 안 되면 [OUT] 데이터만 나옵니다.

송수신 데이터를 확인하는 방식은 Ascii, Hex, List View 3가지로 볼 수 있습니다.

[ASCII VIEW]

아스키코드 문자형태로 표시합니다. 제어코드 또는 아스키코드가 아니면 <>괄호로 표시되고 HEX값으로 나타납니다.

[OUT] <01><03><00><00><00><01><84><0A>
[OUT] <01><03><00><00><00><01><84><0A>
[OUT] <01><03><00><00><00><01><84><0A>
[OUT] <01><03><00><00><00><01><84><0A>

[Ascii View]

[HEX VIEW]

16진 HEX CODE값으로 표시됩니다.

[OUT] 01 03 00 00 00 01 84 0A
[OUT] 01 03 00 00 00 01 84 0A
[OUT] 01 03 00 00 00 01 84 0A
[OUT] 01 03 00 00 00 01 84 0A

[Hex View]

[LIST VIEW]

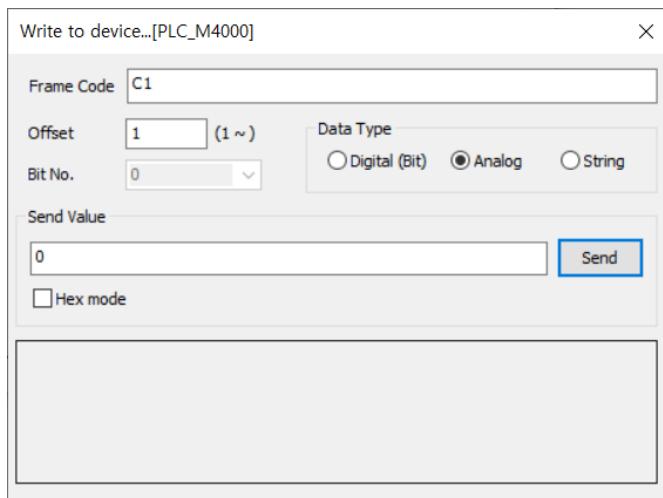
통신 프레임별 리스트 형태로 표시하며, 각 프레임의 송신횟수, 수신횟수, 수신오류율을 표시합니다.

FRAME	RECEIVE	SCOUNT	RCOUNT	ERROR
C1(0)		10	0	90.0%
C2(1)		11	0	90.9%
C3(2)		10	0	90.0%
C4(3)		10	0	90.0%

[List View]

[디바이스 쓰기시험]

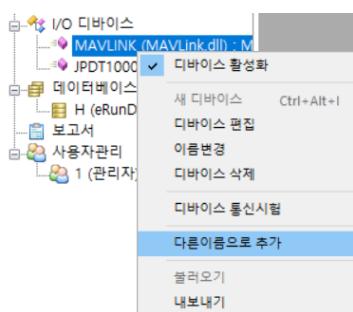
[Run Test]실행으로 디바이스와 통신중에 특정 값을 디바이스에 쓰고자 할 경우 [Send Test] 버튼을 누르면 아래와 같이 [Write to device] 창이 나타납니다. 적당한 값을 넣어주고 [Send]버튼을 누르면 디바이스로 즉시 출력됩니다.

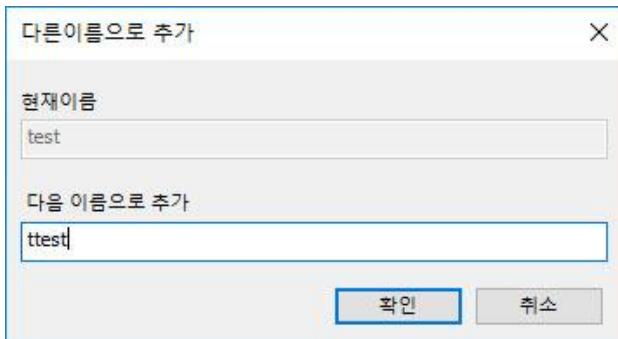


항목	설명
Frame Code	시험중인 디바이스에 등록된 프레임코드명을 입력합니다.
Offset	프레임 코드에서 변위값을 지정합니다.(1부터 시작)
Bit No.	OFFSET에 해당하는 데이터의 비트번호(0부터 시작)
Data Type	디지털(0또는1), 아날로그, 문자열 데이터형으로 내보냅니다.
Send Value	출력값
Hex mode	아날로그 데이터일 때 Send Value값을 Hex값으로 인식해서 내보냅니다.

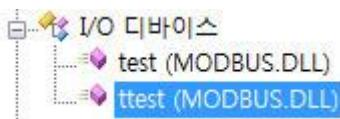
8.3 다른 이름으로 추가하기

설정된 프레임 리스트를 똑같이 사용하는 디바이스가 여러 개 등록되어야 할 경우 다른 이름으로 추가하면 됩니다.





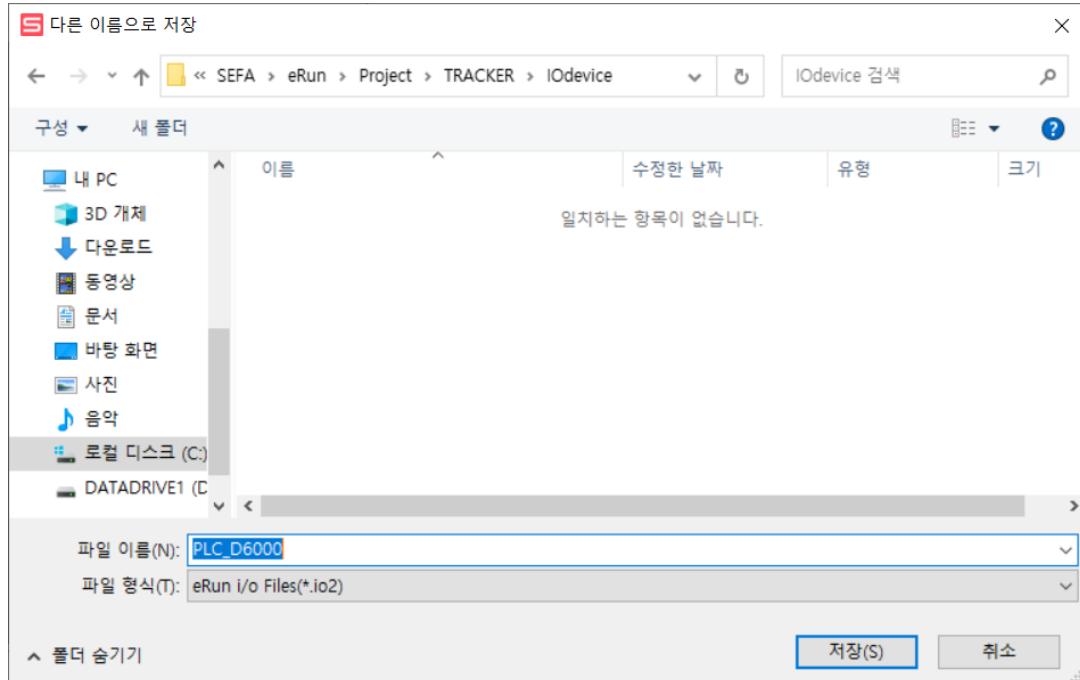
test 디바이스가 ttest 이름의 디바이스로 추가되었습니다.



8.4 디바이스 내보내기(Export) / 불러오기(Import)

현재 프로젝트에 등록된 디바이스를 다른 프로젝트로 복사할 경우 [내보내기]를 통해서 파일로 저장하고 다른 프로젝트에서 [불러오기] 하면 됩니다.

[내보내기]



8.5 디바이스 관련 스크립트 함수

프로젝트 실행할 때 스크립트 함수를 통해서 디바이스에 직접 내보내야 하는 데이터를 처리하거나 디바이스 송수신 횟수등을 가져 올수 있습니다. 대량의 데이터를 한 번에 내보내서 통신과 관련하여 전체적인 성능 저하를 줄이기 위한 방법으로 사용합니다. 단, 해당 스크립트는 디바이스 드라이버에 따라서 지원하지 않는 경우도 있으니 드라이버 매뉴얼을 참고하시기 바랍니다.

```
_DeviceWriteString("디바이스명", "프레임명", "문자열");
```

문자열(string)을 디바이스로 직접 내보낸다. 한 번에 내보낼 수 있는 최대길이는 4000바이트.

```
_DeviceWriteBytes("device name", "frame code", offset, bytes_data, bytesSize);
```

바이트 데이터를 디바이스로 직접 내보낸다. 최대 바이트길이 4000바이트 이내.

8.6 아스키코드(ASCII) 테이블

DEC	HEX	CHAR									
0	0x00	NUL	32	0x20	SP	64	0x40	@	96	0x60	'
1	0x01	SOH	33	0x21	!	65	0x41	A	97	0x61	A
2	0x02	STX	34	0x22	"	66	0x42	B	98	0x62	B
3	0x03	ETX	35	0x23	#	67	0x43	C	99	0x63	C
4	0x04	EOT	36	0x24	\$	68	0x44	D	100	0x64	D
5	0x05	ENQ	37	0x25	%	69	0x45	E	101	0x65	E
6	0x06	ACK	38	0x26	&	70	0x46	F	102	0x66	F
7	0x07	BEL	39	0x27	`	71	0x47	G	103	0x67	G
8	0x08	BS	40	0x28	(72	0x48	H	104	0x68	H
9	0x09	HT	41	0x29)	73	0x49	I	105	0x69	I
10	0x0A	LF	42	0x2A	*	74	0x4A	J	106	0x6A	J
11	0x0B	VT	43	0x2B	+	75	0x4B	K	107	0x6B	K
12	0x0C	FF	44	0x2C	,	76	0x4C	L	108	0x6C	L
13	0x0D	CR	45	0x2D	-	77	0x4D	M	109	0x6D	M
14	0x0E	SO	46	0x2E	.	78	0x4E	N	110	0x6E	N
15	0x0F	SI	47	0x2F	/	79	0x4F	O	111	0x6F	O
16	0x10	DEL	48	0x30	0	80	0x50	P	112	0x70	P
17	0x11	DC1	49	0x31	1	81	0x51	Q	113	0x71	Q
18	0x12	DC2	50	0x32	2	82	0x52	R	114	0x72	R
19	0x13	DC3	51	0x33	3	83	0x53	S	115	0x73	S
20	0x14	DC4	52	0x34	4	84	0x54	T	116	0x74	T
21	0x15	NAK	53	0x35	5	85	0x55	U	117	0x75	U
22	0x16	SYN	54	0x36	6	86	0x56	V	118	0x76	V
23	0x17	ETB	55	0x37	7	87	0x57	W	119	0x77	W
24	0x18	CAN	56	0x38	8	88	0x58	X	120	0x78	X
25	0x19	EM	57	0x39	9	89	0x59	Y	121	0x79	Y
26	0x1A	SUB	58	0x3A	:	90	0x5A	Z	122	0x7A	z
27	0x1B	ESC	59	0x3B	;	91	0x5B	[123	0x7B	{
28	0x1C	FS	60	0x3C	<	92	0x5C	₩	124	0x7C	
29	0x1D	GS	61	0x3D	=	93	0x5D]	125	0x7D	}
30	0x1E	RS	62	0x3E	>	94	0x5E	^	126	0x7E	~
31	0x1F	US	63	0x3F	?	95	0x5F	_	127	0x7F	DEL

제 9 장 데이터베이스

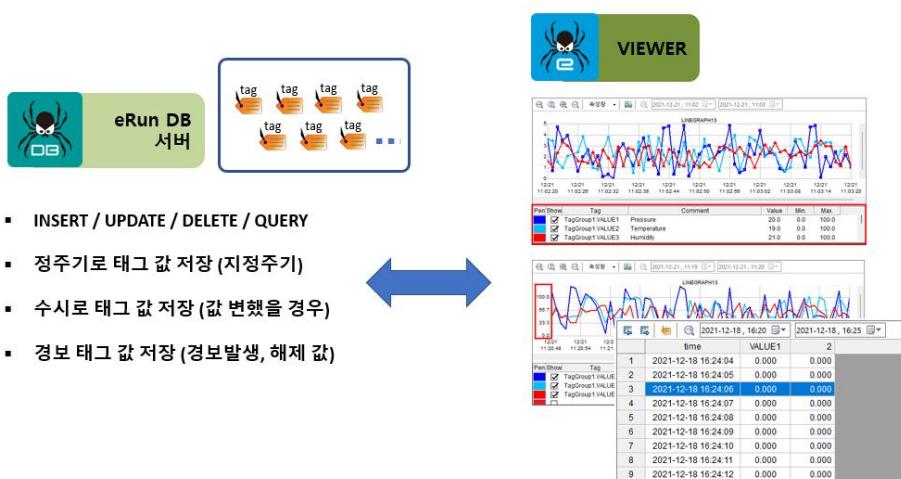
9.1 소개 및 특징

eRun SCADA에서는 수천, 수만개의 데이터 포인트를 감시하고 제어를 효율적으로 하고 그 변화이력을 적절하게 제공하기 위해서 Realtime Database 서버를 운영합니다. 데이터베이스 시스템은 실시간으로 변하는 수많은 데이터를 빠르게 파일시스템에 저장하고, 원하는 기간동안의 데이터 검색서비스를 제공합니다. eRun SCADA에서는 eRunDB 데이터베이스 서버 서비스를 기본적으로 제공하며, 감시이력에 대한 조회를 빠르게 처리합니다.

아울러 근거리, 원거리 관계없이 서로 다른 데이터들을 유기적으로 연결시켜 주고 동시에 접근 및 가공이 가능하도록 하는 시스템, 그리고 요즘 현재 사용되는 대부분의 데이터베이스는 관계형 데이터베이스 모델로 RDBMS라고 합니다. eRun SCADA에서는 스크립트 함수를 통해서 MS-SQL, ORACLE, MySQL, MariaDB 등과 같은 상용 관계형 데이터베이스를 사용할 수 있습니다.

이번 장에서는 여러 DBMS에 대한 설정방법과 필드정보를 eRun에 등록하고 프로젝트에 연동하는 방법을 소개합니다.

- 스크립트를 통한 복잡한 데이터에 대한 저장 및 가공이 용이합니다.
- 대량의 태그에 대한 쿼리속도가 빠르고 일정합니다.
- 상용 RDBMS와의 연결성을 제공하며, MS-ACCESS, MS-SQL, ORACLE, MySQL, MariaDB, ODBC에 대한 연결을 100% 지원합니다.
- 5가지 형태의 데이터베이스 프로세스 형태를 제공하고 있어서 활용성이 좋습니다.
- 스크립트를 통한 SQL 질의문을 사용하여 손쉽게 데이터 조작 프로그램이 가능합니다.
- 로컬 & 리모트 데이터베이스 처리가 가능합니다.
- 정주기, 비주기로 실시간 처리 데이터를 저장합니다.
- 경보이력에 대한 데이터 관리가 용이합니다.



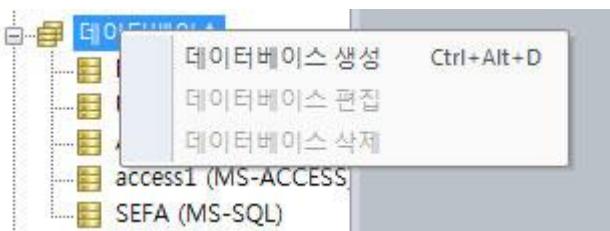
9.2 데이터베이스 생성하기

프로젝트내에 데이터베이스를 생성한다는 것은 데이터베이스 처리모델을 생성한다는 의미입니다.

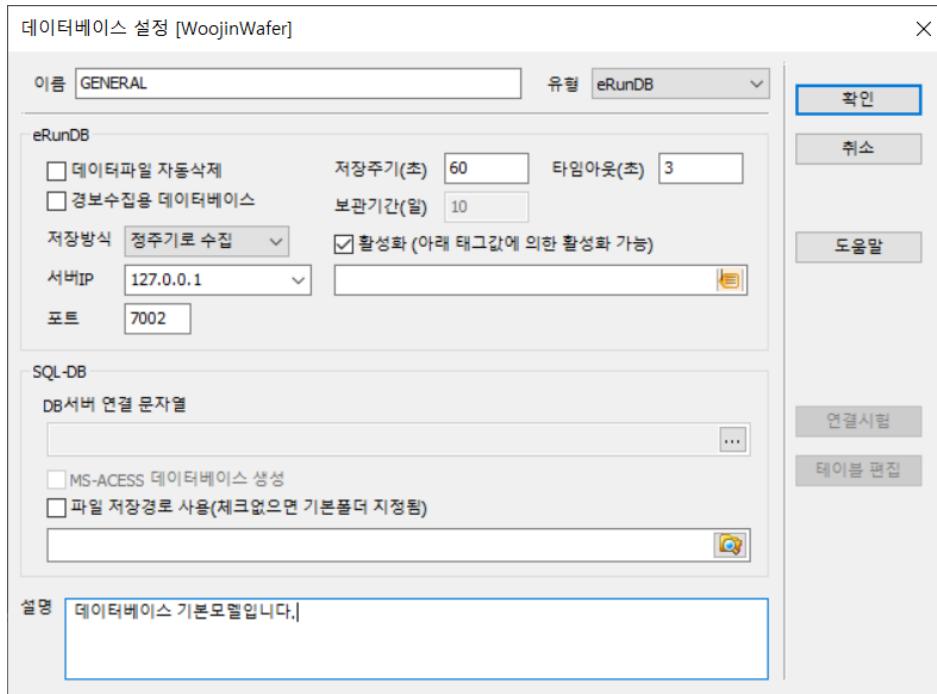
데이터베이스 처리모델이라는 것은 모델유형, 저장주기 등을 설정한 기준을 말합니다.

데이터 저장은 처리모델에 따라 저장되며, 여러 개의 데이터베이스 모델을 생성할 수 있습니다.

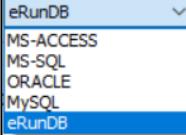
프로젝트 관리에서 데이터베이스를 선택하고 마우스 우측버튼을 클릭하면 아래 그림과 같이 팝업메뉴가 나타납니다. 이것은 새로운 데이터베이스를 현재의 프로젝트에 생성 추가할 경우 나타나게 되는데 메뉴에서 '데이터베이스 생성' 메뉴만 활성화되어 있습니다.

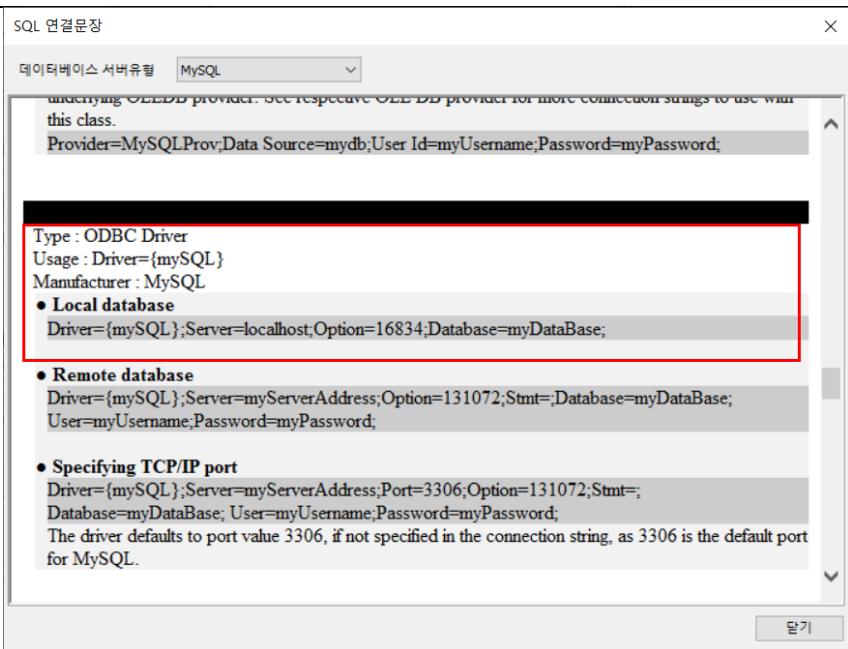
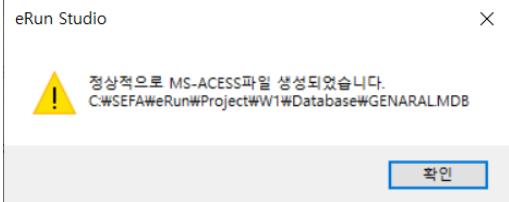
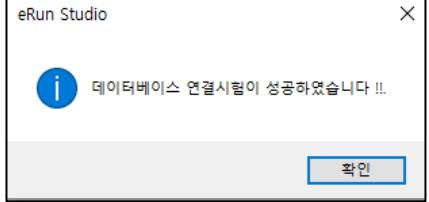


데이터베이스 생성 메뉴를 클릭하면 아래와 같은 설정 화면이 나타납니다.



항목	설명
이름	데이터베이스 모델이름을 입력합니다. 태그 데이터베이스 저장설정할때 사용되는 이름입니다. 프로젝트에서 유일하게 입력해주어야 합니다.
유형	5가지의 데이터베이스 유형이 있습니다.

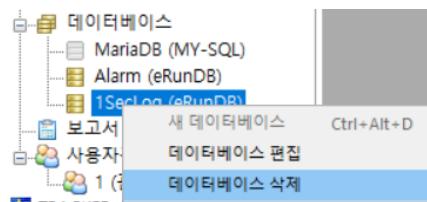
	 <p>MS-ACCESS : Microsoft MS-access 로컬 데이터베이스. MS-SQL : Microsoft MS-SQL Server 로컬/원격 데이터베이스 ORACLE : ORACLE 로컬/원격 데이터베이스 MySQL : ORACLE 로컬/원격 데이터베이스 eRunDB : eRun SCADA RTDB 데이터베이스</p>
데이터파일 자동삭제	데이터베이스 유형이 eRunDB일 경우 사용되는 항목이고 보관기간이 지난 데이터 파일은 삭제됩니다.
경보수집용 데이터베이스	경보태그에 대해서 알람이 발생하면 저장하는 데이터베이스입니다.
저장방식	<ul style="list-style-type: none"> ■ 정주기로 수집 태그값을 일정한 저장주기 간격으로 파일에 저장합니다. ■ 수시로 수집 태그값이 변경될 때마다 파일에 저장합니다.
서버IP, 포트번호	본 데이터베이스 모델이 연결해야 하는 DB서버 IP, 포트번호입니다. 이 정보는 프로젝트 속성에서 설정한 데이터베이스 IP, 포트를 입력해 줍니다.
저장주기	정주기 저장방식의 모델일 때 사용하는 저장간격 값입니다. 최소 1초간격.
타임아웃	본 데이터베이스 모델이 DB서버에 접속하는 타임제한시간입니다. 디폴트 3초
보관기간	데이터파일의 파일저장 기간입니다. 예로 90으로 설정되었다면 90일 이전 데이터 파일은 모두 삭제합니다.
활성화	유형이 eRunDB일 때 사용하며, 프로젝트 실행할 때 데이터베이스 프로세스 실행 여부 선택합니다. 활성화 태그 내용이 없고 활성화 체크가 되어있으면 무조건 프로세스 실행을 합니다. 반대로 활성화 태그가 등록이 되어있으면 태그값에 따라서 가동/비가동이 됩니다.
SQL-DB 서버 연결문자열	<p>SQL DB에 접속하기 위한 DB로 설정할 경우 SQL Server에 접속하기 위한 접속정보 문자열을 입력합니다. SQL을 기본으로 사용하는 DB에 OLE-DB로 접속을 시도하도록 합니다. 이것은 DB 제조사에 따라서 다르므로 ... 버튼을 클릭하여 각 제품별, 모델별로의 DB 연결 문자를 확인하여 입력하시면 됩니다.</p> <p>입력하실 문자열은 각 데이터베이스의 서버유형과 각 제품의 유형에 맞추어 보시면 빨간색 문자가 있습니다. 그것을 기초로 만들어 넣으면 됩니다</p>

	 <p>예시) ODBC 드라이버 사용할 경우 Driver={MySQL ODBC 5.1 Driver};Server=localhost;Database=TestDB; User=Woo;Password=Woo12;</p>
MS-ACCESS 데이터베이스 생성	DB 유형이 MS-ACCESS 인 경우 사용되며, 저장경로에 파일이 없을 경우 생성하도록 합니다. MDB 파일명은 데이터베이스 설정의 이름이 됩니다. 정상적으로 생성되면 다음과 같은 메시지 창이 나타납니다. 
파일저장 경로사용	데이터파일, 경보파일 등이 저장될 위치를 지정합니다. 가급적이면 데이터베이스 마다 별도의 폴더를 지정하도록 합니다.
연결시험	데이터베이스 유형이 eRunDB가 아닌 SQL서버를 이용할 때 연결테스트를 합니다. SQL서버 연결이 정상적이면 아래와 같이 메시지가 나타납니다.  연결에 실패한 경우 아래와 같이 메시지가 나타납니다.

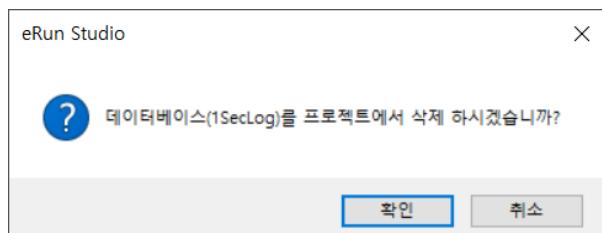
	eRun Studio Error : 80004005 Source : Microsoft OLE DB Provider for ODBC Drivers description : [MySQL][ODBC 5.1 Driver]Can't connect to MySQL server on 'localhost' (10061) 확인	
테이블 편집	DB유형이 MS-ACCESS, MS-SQL, ORACLE, MySQL인 경우에 생성된 데이터베이스에 새 테이블을 생성할 때 사용합니다. 자세한 설명은 별도 [테이블 편집] 부분을 참고하세요	
설명	본 데이터베이스 모델에 대한 설명을 입력합니다.	

9.3 데이터베이스 삭제

데이터베이스 삭제는 모델을 삭제하는 것입니다. 데이터베이스 항목에서 삭제하고자 하는 데이터베이스를 선택하고 우측마우스를 클릭하면 아래 그림과 같이 팝업메뉴가 나타납니다.



팝업메뉴에서 [데이터베이스 삭제]를 클릭하면 해당 데이터베이스 '1SecLog' 삭제 확인메시지 박스가 나타납니다. '확인'을 클릭하여 프로젝트에 등록 되어있는 데이터베이스를 삭제합니다.



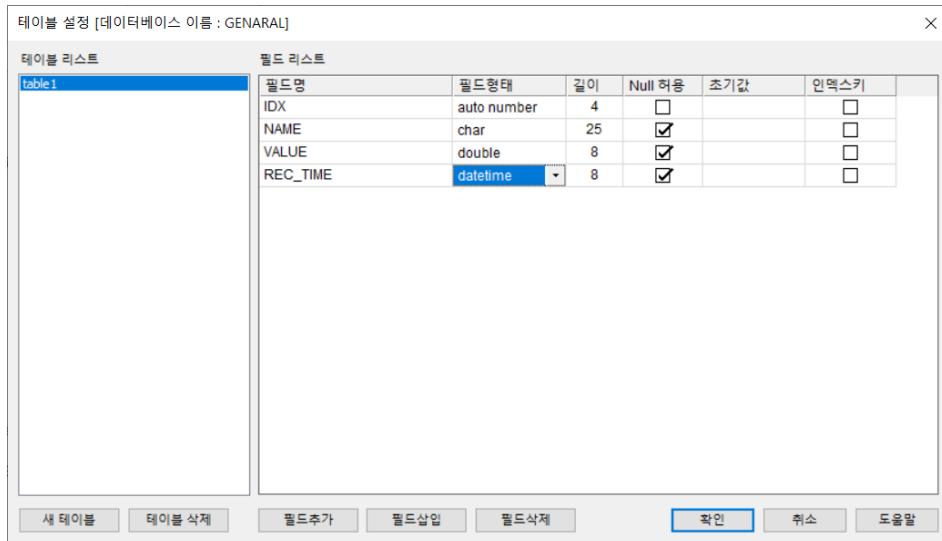
9.4 SQL 테이블 생성하기

데이터베이스 유형이 MS-ACCESS, MS-SQL, ORACLE, MySQL 일 때 스튜디오에서 생성된 데이터베이스에 테이블을 생성하거나, 편집할 때 사용합니다.

eRun SCADA는 SQL 데이터베이스를 생성하거나, 테이블 생성, 편집을 스크립트를 이용해서도 진행이 가능합니다. 이부분은 뒷장의 SQL 브라우저 활용 부분을 참고하세요.

(주의)테이블 편집은 스튜디오에서 생성된 테이블만 편집이 가능합니다.

테이블편집 버튼을 누르면 아래와 같이 창이 나타납니다.



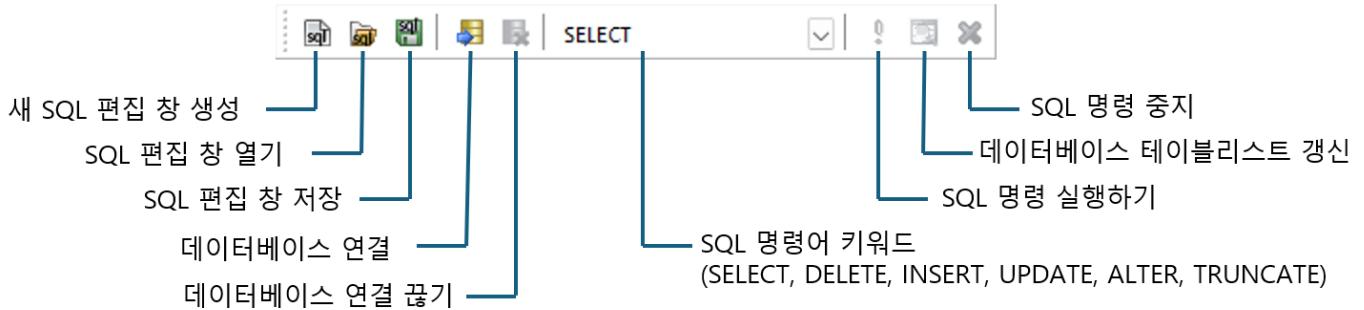
항목	설명
테이블 리스트	<p>본 데이터베이스에 등록된 테이블 리스트 표시창입니다. 필드추가를 하려면 먼저 테이블 리스트에서 편집하려는 테이블을 선택합니다.</p>
필드 리스트	<p>테이블 필드항목을 추가하거나 삭제합니다.</p> <ul style="list-style-type: none"> ■ 필드명 필드이름을 알파벳으로 시작되는 이름으로 입력합니다. DB엔진에서 사용 불가능한 필드명이 있으니 주의바랍니다. (예)time, date, ... ■ 필드형태 필드 데이터형을 선택합니다. ■ 길이 필드데이터 길이입니다. 보통 필드형태에 의해 정해집니다. ■ NULL 허용 새 레코드 추가시 NULL(빈) 허용할지 선택합니다. ■ 초기값 프로젝트 실행중에 새 레코드 추가할 때 본 필드의 초기 값을 저장하도록 합니다. ■ 인덱스키 검색을 빠르게 하기위한 키를 선택합니다.
새 테이블	<p>새로운 테이블을 데이터베이스에 추가생성합니다.</p>

테이블 삭제	선택한 테이블을 데이터베이스에서 삭제합니다. 테이블에 정의된 필드정보 및 저장된 데이터 모두 삭제됩니다.
필드 추가	본 테이블 마지막라인에 필드추가 합니다. 
필드 삽입	본 테이블의 커서가 있는 위치에 필드를 삽입합니다.
필드 삭제	커서가 있는 필드를 본 테이블에서 삭제합니다.

9.5 SQL 브라우저 활용하기

SQL (**S**tructured **Q**uery **L**anguage, 구조화 질의어)는 관계형 데이터베이스 관리 시스템(RDBMS)의 데이터를 관리하기 위해 설계된 특수 목적의 프로그래밍 언어입니다. 관계형 데이터베이스 관리 시스템에서 자료의 검색과 관리, 데이터베이스 스키마 생성과 수정, 데이터베이스 객체 접근 조정 관리를 위해 고안되었으며, 많은 수의 데이터베이스 관련 프로그램들이 SQL을 표준으로 채택하고 있습니다.

eRun SCADA에서는 SQL 데이터베이스를 내장된 스크립트를 통하여 편리하게 데이터베이스 프로그램이 가능하기 때문에 강력한 SCADA TOOL이라고 할 수 있습니다. 스튜디오에 내장된 SQL 브라우저를 통하여 데이터베이스에 접근하거나 간단한 SQL 명령어 시험 등을 할 수 있습니다.



새 SQL 편집창 생성

새로운 태그 1개를 추가등록 합니다. 마지막 태그의 속성에 연속성을

SQL 편집창 열기

한번에 동일한 태그종류, 디바이스, 어드레스를 여러 개 추가할 때 사용합니다. 한번에 추가하고자 하는 태그개수를 최소 1개 이상 입력합니다. 상세설명은 뒷장의 [태그 일괄생성]부분 참고하세요.

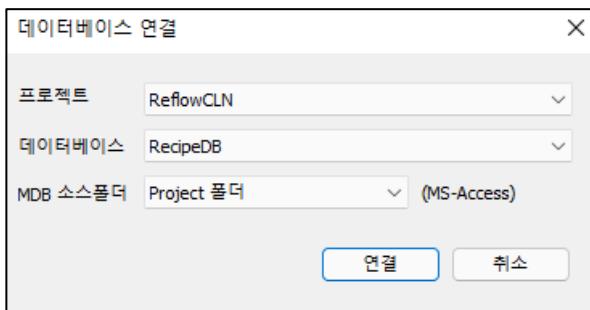
SQL 편집창 저장

현재 선택된 태그 1개 이상의 태그를 삭제합니다.

데이터베이스 연결

프로젝트에 등록된 SQL모델 유형의 데이터베이스 연결을 합니다.

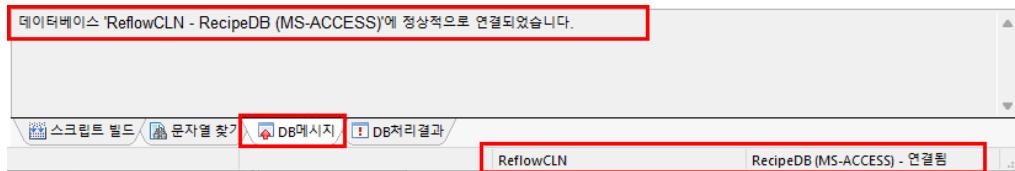
프로젝트를 선택하면 등록된 데이터베이스 리스트가 표시됩니다. 연결버튼을 통해서 접속을 합니다.



항목	설명
프로젝트	프로젝트를 먼저 선택합니다.
데이터베이스	선택한 프로젝트에 SQL 데이터베이스 모델이 있으면 표시됩니다.
MDB소스폴더	<p>데이터베이스 유형이 MS-ACCESS인 경우 MDB파일이 생성됩니다. MDB파일은 최초에 Project 폴더에 생성되고, 프로젝트 배포를 할 때 Publish 폴더에 MDB파일이 없으면 복사가 됩니다. 프로젝트가 실행되면 데이터베이스는 Publish 폴더에 있는 MDB파일을 사용하게 됩니다. 그렇기 때문에 스튜디오에서 SQL브라우저를 사용할 경우 MDB파일 폴더를 선택하고 진행합니다.</p> <ul style="list-style-type: none"> ■ Project 폴더 : 최초 MDB 파일 생성할 때 저장되는 폴더 ■ Publish 폴더 : 프로젝트 실행 때 저장되는 폴더 <p>데이터베이스 이름앞에 “Project” 또는 “Publish”로 구분해서 표시합니다.</p>

--	--

연결이 되면 아래와 같이 DB메시지 창과 연결상태 텍스트가 표시됩니다.



데이터베이스 연결 끊기

현재 연결 되어있는 데이터베이스 접속을 종료합니다.

SELECT SQL 명령어

SQL 명령어 문법 사용예를 표시합니다. 현재 버전에서 지원하는 명령어는 다음과 같습니다.

SELECT, INSERT, DELETE, UPDATE, ALTER, TRUNCATE 입니다.

[SELECT 선택]

```
//SELECT field1,field2, ...]  
//      FROM table1  
//      WHERE [condition]  
//      GROUP BY field  
//      HAVING [condition]  
//      ORDER BY [asc | desc]
```

[INSERT 선택]

```
//INSERT INTO table_name(field_name1, field_name2) VALUES('value_name', 123.45)
```

[DELETE 선택]

```
//DELETE FROM table_name WHERE [condition]
```

[UPDATE 선택]

```
//UPDATE table_name SET field_name=data WHERE [condition]
```

[ALTER 선택]

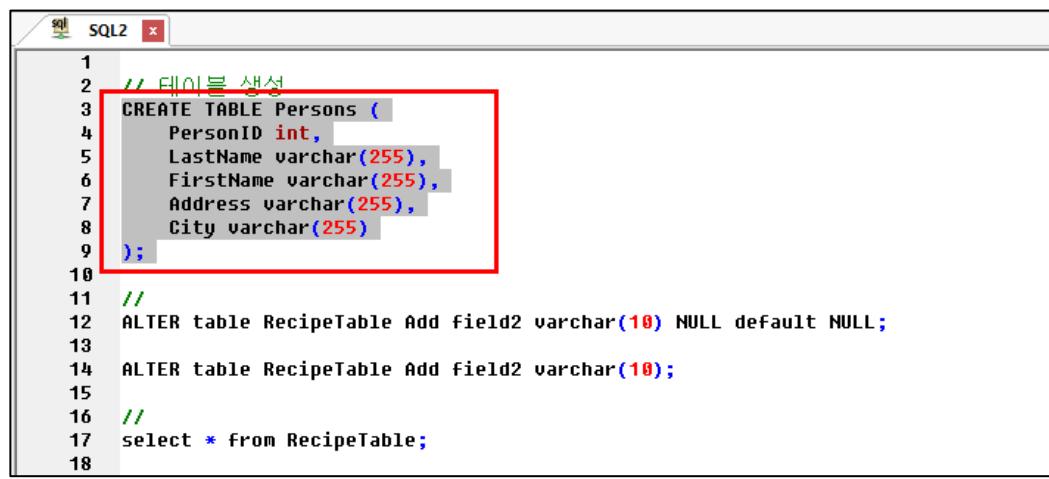
```
//ALTER TABLE table_name ADD field_name VARCHAR(10)
```

[TRUNCATE 선택]

```
//TRUNCATE TABLE table_name
```

SQL명령 실행하기

SQL 문법에따라 명령을 작성하고 실행할 영역을 마우스로 선택한 후 명령실행 버튼을 누르거나, [SHIFT+ENTER]를 클릭하면 SQL구문이 실행됩니다.



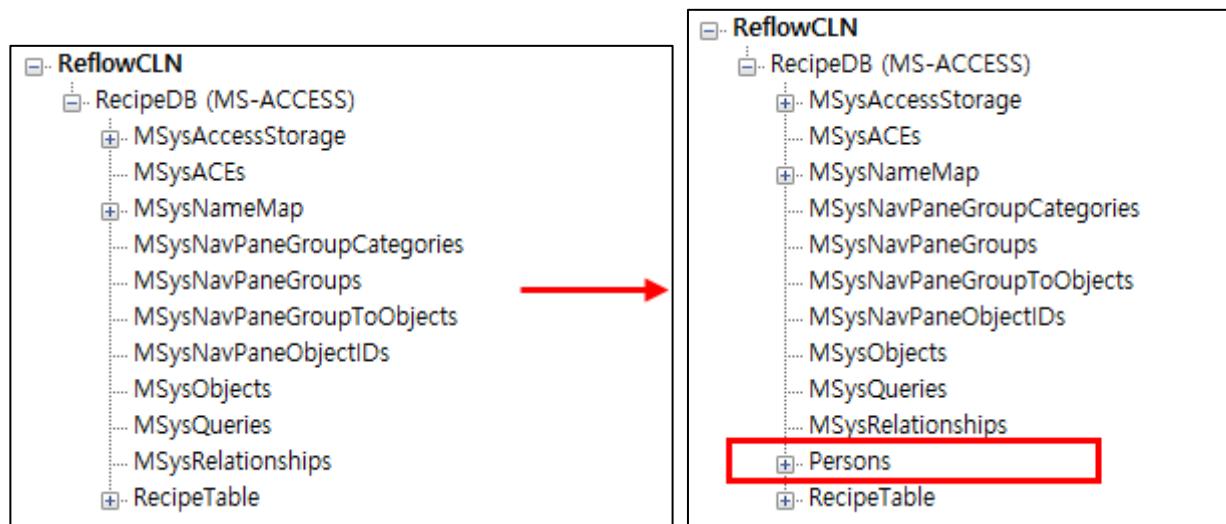
```

1
2 // 테이블 생성
3 CREATE TABLE Persons (
4     PersonID int,
5     LastName varchar(255),
6     FirstName varchar(255),
7     Address varchar(255),
8     City varchar(255)
9 );
10
11 //
12 ALTER table RecipeTable Add field2 varchar(10) NULL default NULL;
13
14 ALTER table RecipeTable Add field2 varchar(10);
15
16 //
17 select * from RecipeTable;
18

```

[선택영역 실행]

위의 SQL구문은 테이블('Persons')을 생성합니다.



데이터베이스 테이블리스트

데이터베이스에 존재하는 테이블 리스트를 모두 보여줍니다.

SQL 명령실행 중지

SQL명령을 중단시킵니다.

제 10 장 보고서

eRun SCADA의 보고서 관리기능을 통해서 프로젝트 실행 중에 저장된 태그데이터를 가공하여 일정한 보고서 양식에 맞게 사용자가 원할 때 프린터에 출력하거나 자동으로 매일 매일 정해진 시간에 출력하거나 파일에 저장할 수 있습니다.

- 보고서의 양식은 eRun의 SF-FORM과 엑셀(MS-EXCEL)로 나누어집니다.
- SF-FORM은 그리기 편집도구를 이용하여 보고서 양식을 생성하고 편집합니다.
- 스크립트를 작성해서 보고서에 반영할 수 있습니다.
- 사용자가 수동으로 특정 일보/주보/월보/연보의 보고서를 출력할 수 있습니다.
- 자동으로 사용자가 지정한 시간에 출력할 수 있습니다.
- 현재 태그 값을 출력할 수 있습니다.
- 보고서 결과는 파일에 저장하거나 프린터로 출력할 수 있습니다.

10.1 보고서 모델 생성하기

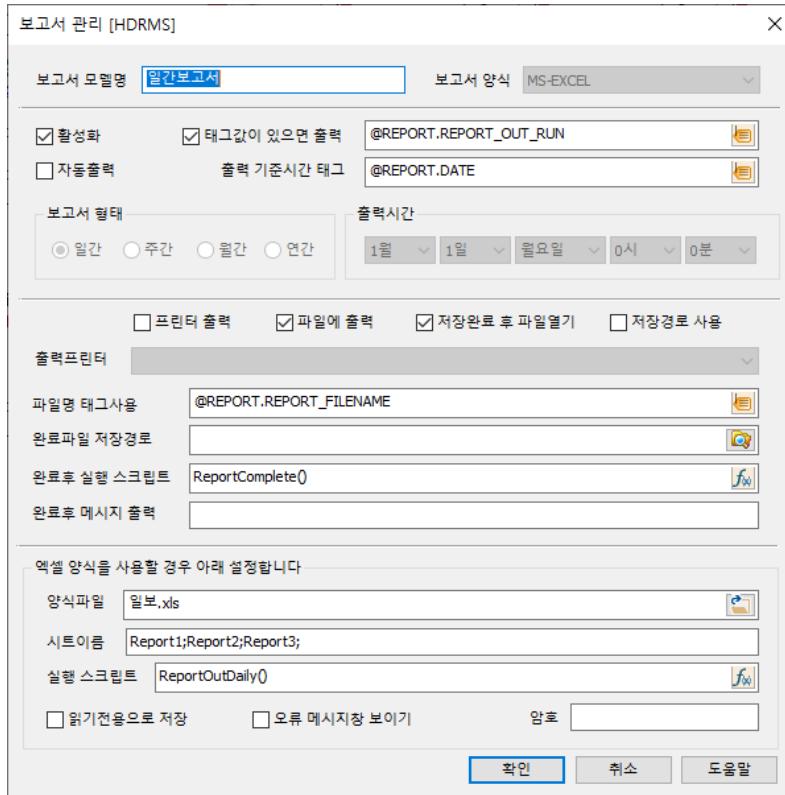
보고서 양식에 따라 보고서 템플릿을 선택하고 출력방식이나 출력시간 등을 상황에 맞게 설정해서 여러 개의 보고서를 등록하고 사용할 수 있습니다. 엑셀 양식을 사용하는 보고서 모델생성은 아래와 같은 순서로 진행합니다.

- ① 보고서 양식 엑셀파일 생성 (c:/sefa/eRun/Project/프로젝트명/ReportTemplate/template1.xls)
- ② 보고서 생성용 스크립트 함수 작성
- ③ 보고서 모델 생성
- ④ 엑셀 양식파일 선택
- ⑤ 실행 스크립트 선택 (사용자가 작성한 보고서 처리함수)

프로젝트 관리에서 보고서 항목을 선택하면 보고서관리 팝업메뉴가 아래와 같이 나타나고 [새 보고서]만 활성화되어있습니다.



[새 보고서] 클릭하면 아래와 같이 생성 창이 나타납니다.



항목	설명
보고서 모델명	보고서 이름을 입력합니다. 보고서의 모델명을 스크립트 내부함수 등에서 호출되므로 다른 오브젝트의 이름과 중복되지 않도록 합니다.
보고서 양식	보고서 양식으로 SF-FORM, MS-EXCEL 두 가지 모델이 있으며, SF-FORM 양식을 지정하면 보고서 품을 스튜디오를 이용하여 품을 직접 그리고, 출력 값을 지정할 수 있으며, MS-EXCEL 양식을 지정하면 엑셀에서 작성한 품을 이용해서 원하는 시트, 셀에 보고서 내용을 출력합니다.
활성화	보고서 활성화 여부를 설정합니다. 기본적으로 보고서 출력을 원할 때는 활성화가 되어 있어야 합니다.
태그 값이 있으면 출력	보고서 출력동작을 사용자 필요에 따라서 해야 할 경우 지정합니다. 지정된 태그 값이 0이 아니면 보고서 출력이 됩니다. 보고서 출력동작이 끝난후에 태그 값이 "0"으로 리셋 됩니다. 태그는 디지털 태그 또는 아날로그 태그로 지정합니다. 다수의 보고서가 있는 경우 이 항목에 대해서는 서로 다른 태그를 설정해야 합니다. 태그이름은 '@' 붙여서 입력해 줍니다.

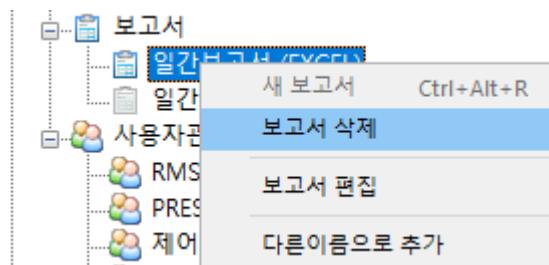
출력 기준시간 태그	수동으로 출력을 할 때, 원하는 날짜의 데이터를 출력하고자 할 경우 미리 태그에 날짜를 넣어주고 선택하도록 합니다. 설정이 되어 있지 않으면 당일 날짜 데이터를 기준으로 출력합니다. 날짜의 형태는 "YYYY-MM-DD hh:mm:ss"의 문자열 이어야 합니다.
자동출력	자동 출력 여부를 설정합니다. 자동출력을 정하면 보고서 형태와 출력시간이 활성화 되며, 설정된 사항 (파일, 프린터)로 출력됩니다.
보고서 출력형태	일간/주간/월간/연간 형태를 선택하며, 그 선택에 따라서 출력시간을 선택할 수 있습니다. <ul style="list-style-type: none"> ■ 일간 : 매일 출력 시간에 맞추어서 자동 출력/저장합니다. ■ 주간 : 매주 출력날짜 및 시간에 맞추어 자동 출력/저장합니다. ■ 월간 : 매월 출력날짜 및 시간에 맞추어 자동 출력/저장합니다. ■ 연간 : 매년 출력 월일 및 시간에 맞추어 자동 출력/저장합니다.
출력시간	출력/저장시간을 지정합니다. 보고서 형태에 따라 월, 일, 요일, 시간을 지정합니다.
프린터 출력	보고서 결과를 출력하고자 하는 프린터를 선택합니다.
파일에 출력	생성된 보고서 파일의 저장경로를 지정해야 합니다. 저장경로가 올바르지 않거나 존재하지 않으면 경고 메세지가 표시됩니다.
저장완료 후 파일 열기	보고서가 완성된 후 저장하고 자동으로 파일을 엽니다. 수동으로 출력을 하는 경우에만 설정을 권장합니다.
출력 프린터	SF-FORM에서 출력할 프린터를 지정합니다.
저장경로	보고서 파일을 저장할 경로를 지정합니다. 저장경로가 올바르지 않거나 존재하지 않으면 경고 메세지가 표시됩니다.
파일명 태그 사용	보고서 양식이 MS-EXCEL일 때 활성화되며, 파일명을 태그로 사용하여 저장합니다.
완료후 실행 스크립트	보고서 출력 (프린터, 파일저장)이 완료된 후 어떤 후 어떤 특정한 메시지를 띄우거나 로그를 기록하거나 등의 작업을 할 경우 호출되는 사용자 스크립트 함수를 지정합니다. 수동으로 출력할 경우에만 설정을 권장합니다.
완료후 메시지 출력	보고서 생성(프린터, 파일저장)이 완료된 후 지정한 메시지를 출력합니다. 수동으로 출력할 경우에만 설정을 권장합니다.
양식파일	엑셀양식 파일을 선택합니다. 양식파일 우측의 버튼을 클릭해서 양식파일을 선택합니다.
시트이름	MS-EXCEL에 여러 개의 시트로 구성된 경우 저장할 시트이름을 입력합니다. 한 개 이상의 시트명을 지정할 수 있으며, 시트이름 구분자로 세미콜론(;)을 사용하며, 지정하지 않으면 EXCEL 기본 시트명으로 출력됩니다.
실행 스크립트	MS-EXCEL 양식의 보고서는 출력하고자 하는 데이터를 스크립트에서 작성하여야 합니다. 프로젝트에서 출력 실행 스크립트를 작성해두고 그 함수를 지정해서 EXCEL 보고서 양식 출력으로 활용합니다.
읽기 전용으로 저장	엑셀파일을 저장시 읽기 전용으로 저장 여부를 설정합니다. 보고서파일을 사용자

	가 수정할 수 없도록 하기 위해서 사용합니다.
오류 메시지창 보기	MS-EXCEL을 이용한 출력시 오류가 발생하는 경우 이를 화면에 보여주도록 하기 위해 필요합니다. 정상 운전시에는 미설정을 권장합니다.
암호	저장파일에 암호를 설정합니다. 보고서 파일의 보안을 위해서 암호를 사용합니다.

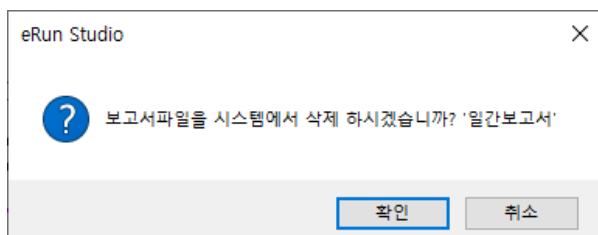
10.2 보고서 삭제

이미 만들어진 보고서 위치에서 마우스 우측버튼을 클릭하면 아래 그림같이 팝업메뉴가 나타납니다. 이것은 기존의 보고서를 삭제 할 수 있도록 합니다.

보고서관리 항목에서 삭제하고자 하는 보고서 모델을 선택하고 우측마우스를 클릭하면 아래 그림과 같이 팝업 메뉴가 나타납니다.



팝업메뉴에서 [보고서 삭제]를 클릭하면 해당 보고서 모델 '일간보고서' 삭제 확인메시지 박스가 나타납니다. '확인'을 클릭하여 프로젝트에 등록 되어있는 보고서 모델을 삭제합니다.



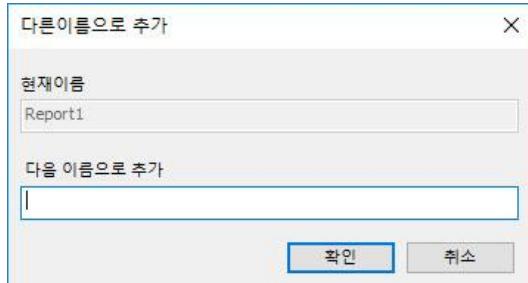
10.3 보고서 모델 다른이름으로 추가

동일한 양식으로 이름만 변경할 경우 현재의 보고서를 다른 이름으로 추가합니다.



[다른이름으로 추가] 메뉴를 클릭하면 아래와 같이 나타납니다.

보고서의 이름은 스크립트 내부함수 등에서 호출되므로 다른 오브젝트의 이름과 중복되지 않도록 합니다.



10.4 MS-EXCEL 보고서 처리용 스크립트 함수

보고서 유형을 MS-EXCEL 파일을 사용할 때 사용되는 내장함수입니다.

`_RepGetValue("보고서모델명", 시트번호, "셀명", "태그명", "시간", 값종류, "검색조건")`

태그값을 데이터베이스에서 검색하고 결과를 MS-EXCEL 파일의 지정시트, 셀에 출력합니다.

데이터베이스는 eRunDB 유형이고 SQL 데이터베이스는 해당되지 않습니다.

파라미터에 대한 설명을 아래와 같습니다.

보고서모델명 (문자열)

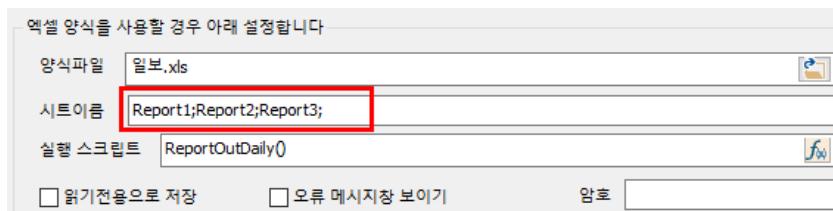
프로젝트에 등록되어 있는 보고서 모델이름을 문자열 형태로 지정합니다.

시트번호 (정수)

엑셀 시트번호를 0번부터 지정합니다. 보고서 모델에 시트가 등록되어 있어야 합니다.

아래 그림에서 시트이름이 3개이다. `Report1;Report2;Report3;`

보고서 파일이 생성되고 파일을 열어보면 엑셀시트명이 순서대로 3개가 만들어집니다.



셀명 (문자열)

해당 시트의 특정 셀번호를 입력합니다. 예를 들어서 "C10" 같은 형식으로 셀의 번호를 입력합니다. 동일한 시트에 동일한 셀을 두번 사용하면 두번째 셀은 검은색으로 무시되니 주의하시기 바랍니다.

태그명 (문자열)

검색하고자 하는 태그 이름을 입력합니다. (예 - "TagGroup1.POWER" - 태그구분자 기호 '@'는 붙이지 않습니다.)

시간 (문자열)

필요한 순간의 데이터를 검색하기 위해서 입력합니다. 시간의 형태는 아래와 같습니다.

■ 일간보고서 출력에 필요한 형태

형식은 : DAY + 'D' + ',' + TIME + 'H'

기준시간일자를 정해서 특정시간 데이터를 가져옵니다. 몇일, 몇시 데이터를 가져오는지에 대한 의미입니다. 기준시간이 설정 되어있지 않는 경우는 오늘 날짜를 기준시간으로 사용합니다.

DAY : 기준시간보다 몇일 전(-) 또는 몇일 후(+)의 정수를 입력합니다. 0을 입력하면 기준시간을 사용합니다.

TIME : 몇시 데이터를 의미하며 1시 (00:00:00~00:59:59) 데이터부터 24시(23:00:00~23:59:59) 데이터를 사용합니다.

(사용예)

기준시간일자 : 2009-1-9 09:00:00 설정되어있을 경우.

0D,1H : 2009-1-9 00:00:00 ~ 2009-1-8 00:59:59

0D,24H : 2009-1-9 23:00:00 ~ 2009-1-8 23:59:59

-1D,1H : 2009-1-8 00:00:00 ~ 2009-1-8 00:59:59

-1D,24H : 2009-1-8 23:00:00 ~ 2009-1-8 23:59:59

1D,1H : 2009-1-10 00:00:00 ~ 2009-1-10 00:59:59

1D,24H : 2009-1-10 23:00:00 ~ 2009-1-10 23:59:59

■ 주간보고서 출력에 필요한 형태

형식은 : WEEK + 'W' + ',' + DATE

기준시간일자를 정해서 특정요일의 종일 데이터를 가져옵니다. 몇째 주 몇 요일 데이터를 가져오는지에 대한 의미입니다. 기준시간이 설정되어있지 않는 경우는 오늘 날짜를 기준시간으로 사용합니다.

WEEK : 기준일자시간보다 몇주 전(-) 또는 몇주 후(+)의 정수를 입력한다. 0을 입력하면 기준시간을 사용합니다.

DATE : 주간 요일을 정해진 문자열로 입력합니다. (SUN, MON, TUE, WED, THU, FRI, SAT)

(사용예)

기준시간일자 : 2013-4-29 09:00:00 설정되어있을 경우.

0W,SUN : 기준시간일자가 속해있는 주의 앞선 일요일의 종일 데이터
(2013-4-29 00:00:00 ~ 2013-4-29 23:59:59)

-1W,SUN	:	일주일	전	일요일	종일	데이터
		(2013-4-21 00:00:00 ~ 2013-4-21 23:59:59)				
1W,SUN	:	일주일	후	일요일	종일	데이터
		(2013-5-5 00:00:00 ~ 2013-5-5 23:59:59)				

■ 월간보고서 출력에 필요한 형태

형식은 : MONTH + 'M' + ',' + DAY + 'D'

기준시간일자를 정해서 특정 월의 특정일의 종일 데이터를 가져옵니다. 몇월 몇일 데이터를 가져오는지에 대한 의미입니다. 기준시간이 설정되어있지 않는 경우는 오늘 날짜를 기준시간으로 사용합니다.

MONTH : 기준일자시간보다 몇개월 전(-) 또는 몇개월 후(+)의 정수를 입력합니다. 0을 입력하면 기준월을 사용합니다.

DAY : 지정일자를 정수로 입력합니다. (1 ~31)

(사용예)

기준시간일자 : 2009-1-9 09:00:00 설정되어 있을 경우.

0M,1D : 2009-1-1 00:00:00 ~ 2009-1-1 23:59:59 (당월 1일 데이터)

0M,31D : 2009-1-31 00:00:00 ~ 2009-1-31 23:59:59 (당월 31일 데이터)

-1M,1D : 2008-12-1 00:00:00 ~ 2008-12-1 23:59:59 (전월 1일 데이터)

-1M,31D : 2008-12-31 00:00:00 ~ 2008-12-31 23:59:59 (전월 31일 데이터)

1M,1D : 2009-2-1 00:00:00 ~ 2009-2-1 23:59:59 (익월 1일 데이터)

1M,31H : 2009-2-28 0:00:00 ~ 2009-2-28 23:59:59 (익월 31일 데이터)

■ 연간보고서 출력에 필요한 형태

형식은 : YEAR + 'Y' + ',' + MONTH + 'M'

기준시간일자를 정해서 특정 년도의 특정월의 1개월 데이터를 가져옵니다. 몇년 몇월 데이터를 가져오는 것을 의미합니다. 기준시간이 설정되어있지 않는 경우는 오늘 날짜를 기준시간으로 사용합니다.

YEAR : 기준일자시간보다 몇년 전(-) 또는 몇년 후(+)의 정수를 입력합니다. 0을 입력하면 기준년도를 사용합니다.

MONTH : 지정월을 정수로 입력합니다. (1 ~12)

(사용예)

기준시간일자 : 2009-1-9 09:00:00 설정 되어있을 경우.

0Y,1M : 2009-1-1 00:00:00 ~ 2009-1-31 23:59:59 (당해 1월 데이터)

0Y,12M : 2009-12-1 00:00:00 ~ 2009-12-31 23:59:59 (당해 12월 데이터)

-1Y,1M : 2008-1-1 00:00:00 ~ 2008-1-31 23:59:59 (전년 1월 데이터)

-1Y,12M : 2008-12-1 00:00:00 ~ 2008-12-31 23:59:59 (전년 12월 데이터)

1Y,1M : 2010-1-1 00:00:00 ~ 2010-1-31 23:59:59 (익년 1월 데이터)

1Y,12M : 2010-12-1 0:00:00 ~ 2010-12-31 23:59:59 (익년 12월 데이터)

값종류 (정수)

검색 종류를 (정시값, 평균값, 최소값, 최대값 등) 입력합니다. 값종류에 대한 인덱스 번호와 매크로는 아래와 같습니다. 일반적으로 내부스크립트 함수에서는 인덱스 번호를 사용합니다.

[보고서 내 SFDB 검색명령어 & 인덱스 번호]

매크로	번호	설명
raw_value	0	기본 저장값
stat_tvalue	1	통계데이터 모두 . 시 간격
stat_dvalue	2	통계데이터 모두 . 일 간격
stat_mvalue	3	통계데이터 모두 . 월 간격
flow_value	10	순간값 (정시값)
avg_value	11	평균값
differ_value	12	적산차
min_value	13	최소값
max_value	14	최대값
sum_value	15	합계값
start_value	16	시작값 (해당시간 첫번째 값)
last_value	17	마지막 값 (해당시간 마지막 값)
record_cnt	18	레코드 개수 (주어진 시간내에 기록된 데이터 개수)
run_time1	30	운전시간 1
run_time2	31	운전시간 2
run_count	32	운전횟수
stop_time1	33	정지시간 1
stop_time2	34	정지시간 2
stop_count	35	정지횟수
cat_daily	101	통계요청이 시간별 데이터 요청
cat_monthly	102	통계요청이 일별 데이터 요청
cat_yearly	103	통계요청이 우러별 데이터 요청
alarm_data	201	경보이력

검색조건

검색 값에 대한 조건식을 넣어줍니다. 예를 들어서 검색 값이 특정 값이거나 특정 범위에 있는 값일 경우 결과값으로 사용합니다.

"\$ > 100" -> 검색값이 100보다 큰 경우만 결과값으로 돌려줍니다. (\$는 검색된 값)

(사용예)

```
void RepGetValue
{
    _RepGetValue("ReportDaily", 0, "C10", "TagGroup1.POWER", "-1D,1H", 10, "$>100");
}
```

파라메터	값
보고서모델명	ReportDaily
시트번호	0
셀번호	C10
태그 이름	TagGroup1.POWER
검색시간	-1D,1H
검색종류	10 (순간 값)
검색조건	100보다 큰 순간 값

_RepSetValue("보고서모델명", 시트번호, "셀명", 출력값)

출력값(태그값, 상수값)을 MS-EXCEL 파일에서 지정시트, 지정셀에 출력값을 표시합니다.
데이터베이스와는 관계없이 현재 태그값, 계산값 등을 내보냅니다.

파라메터에 대한 설명을 아래와 같습니다.

보고서모델명 (문자열)

프로젝트에 등록 되어있는 보고서 모델이름을 문자열 형태로 지정합니다.

시트번호(정수)

엑셀 시트번호를 0번부터 지정하며, 보고서 모델에 시트가 등록 되어있어야 합니다.

셀명(문자열)

해당시트의 특정 셀번호를 입력합니다. 예를 들어서 "C10" - C열 10행

동일한 시트에 동일한 셀을 두번 사용하면 두번째 셀은 검색이 무시됩니다.

출력값 (정수, 실수, 문자열)

출력하려고 하는 값을 입력합니다. 입력값은 일반숫자, 문자열, 변수등을 지정할 수 있습니다.

(사용예)

```
void RepSetValue
{
    _RepSetValue("일일보고서", 0, "C10", 123);
```

```
_RepSetValue("일일보고서", 0, "C11", "eRun");
_RepSetValue("일일보고서", 0, "C12", @TagGroup1.POWER);
}
```

파라메터	값
보고서모델명	일일보고서
시트번호	0
셀번호	C10, C11, C12
출력값	123, "eRun", @TagGroup1.POWER

※ eRun 엑셀보고서 출력스크립트 예

```
// 일일보고서 출력 스크립트.

void DailyReport() <-
{
    // 보고서모델명, Sheet번호, Cell번호, 검색태그, 시간, 값종류
    _RepGetValue("Report3", 0, "C1", "TagGroup1.NAME0", "-1D1H", 10, "");
    _RepGetValue("Report3", 0, "C1", "TagGroup1.NAME1", "-1D2H", 10, "");
    _RepGetValue("Report3", 0, "C3", "TagGroup1.NAME2", "-1D3H", 10, "");

    // 보고서모델명, Sheet번호, Cell번호, 출력값.
    _RepSetValue("Report3", 0, "A1", "eRun");
    _RepSetValue("Report3", 0, "A2", "SEFA Technology");
    _RepSetValue("Report3", 0, "A3", "Korea.");
    _RepSetValue("Report3", 0, "A4", "Republic of Korea");
    _RepSetValue("Report3", 0, "B1", @TagGroup1.ANA0);
    _RepSetValue("Report3", 0, "B2", @TagGroup1.ANA1);
    _RepSetValue("Report3", 0, "B3", @TagGroup1.ANA2);
    _RepSetValue("Report3", 1, "B4", @TagGroup1.ANA3);
    _RepSetValue("Report3", 1, "B5", @TagGroup2.ANA0);
    _RepSetValue("Report3", 1, "B6", @TagGroup2.ANA1);
    _RepSetValue("Report3", 1, "B7", @TagGroup2.ANA2);
    _RepSetValue("Report3", 1, "B8", 12345.6789);
    _RepSetValue("Report3", 2, "B9", 12345.6789);
    _RepSetValue("Report3", 4, "A5", 4312345.6789);
}
```

보고서 모델 등록시 실행스크립트 설정부분이 위에서 작성한 함수가 입력되어 있습니다.



제 10 장 보고서

엑셀 양식을 사용할 경우 아래 설정합니다

양식파일	일보.xls	
시트이름	Report1;Report2;Report3;	
실행 스크립트	DailyReport0	
<input type="checkbox"/> 읽기전용으로 저장	<input type="checkbox"/> 오류 메시지창 보기	암호 <input type="text"/>

제 11 장 사용자 관리

사용자 아이디/암호를 관리를 통하여 시스템 운영에 있어서 입력 및 제어를 제한함으로써 이로 인한 보안 및 시스템 운영의 오류를 최소화할 수 있습니다. 또한 사용자별 접근제한이나 운영로그기록으로 시스템 유지관리를 용이하게 합니다. 사용자 정보는 반드시 최소한 1개 이상 등록을 해야만 합니다.

eRun SCADA에서 제공하는 사용자관리는 간단하면서도 사용상의 특징은 다음과 같습니다.

- ① 등록 및 삭제가 간단합니다.
- ② 등록 인원에 제한이 없습니다.
- ③ 특정 영역에 대해서 접근 설정을 할 수 있습니다.
- ④ 사용자별 시작 뷰 페이지를 지정할 수 있습니다.
- ⑤ 다중사용 환경지원으로 사용자별 로그관리

11.1 사용자 정보 추가하기

프로젝트 관리에서 사용자관리를 선택하고 마우스 우측버튼을 클릭하면 팝업메뉴가 나타납니다.



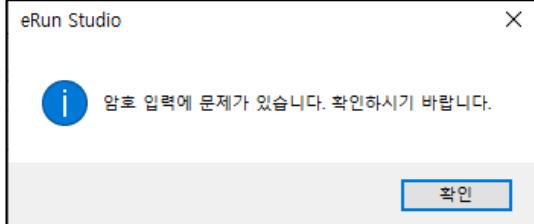
[새 사용자]를 클릭하면 아래와 같이 새 사용자 정보 설정창이 나타납니다.

The dialog box contains the following fields:

사용자 이름 (User Name)	발전운영팀
암호설정 (Password Setting)	•
암호확인 (Password Confirmation)	•
시작페이지 (Start Page)	TopMenu
사용자등급 (User Level)	관리자
접근영역 (Access Area)	제한없음 (Unrestricted)

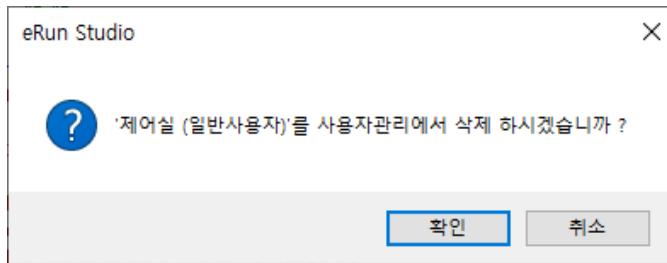
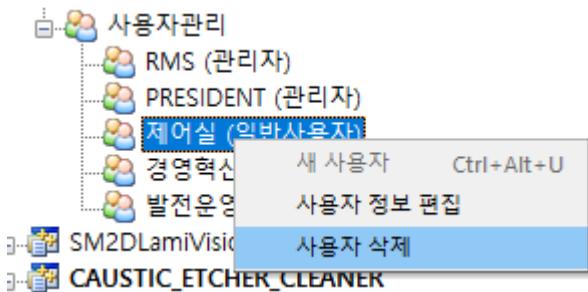
설명 (Description):
발전운영팀 접속정보

Buttons at the bottom: 적용 (Apply), 취소 (Cancel), 도움말 (Help)

항목	설명
사용자 이름	사용자 이름을 입력합니다.
암호설정	암호 입력합니다.
암호확인	암호를 한번 더 입력합니다. 암호설정 값과 암호확인 값이 다르면 그림과 같은 메시지가 출력됩니다. 
시작페이지	사용자 이름에 따라 시작페이지를 다르게 부여합니다. 프로젝트에 등록된 뷰페이지 목록에서 선택합니다. 다중사용자 환경에서 사용자에 따라 감시화면을 다르게 하면 유용합니다.
사용자등급	사용자 등급을 설정합니다.(관리자/일반사용자) 사용자 등급에 따라 뷰페이지 오브젝트 인터랙션을 설정합니다.
접근영역	입력에 대한 접근영역을 설정합니다.(A - Z) 접근영역은 오브젝트에 입력속성에서 설정한 접근영역으로 오브젝트 입력 속성에서 설정된 접근영역보다 낮은 경우에는 사용자 등급이 관리자라도 입력을 할 수 없습니다. 접근영역은 A가 가장 높고 Z로 갈수록 낮아집니다. 접근영역은 자신보다 낮은 접근영역에 대해서 접근을 할 수 있습니다. 예를 들어서 접근영역이 B인 경우에는 B를 포함 Z까지 접근을 할 수 있습니다.
설명	사용자 설명을 입력합니다.

11.2 사용자 삭제

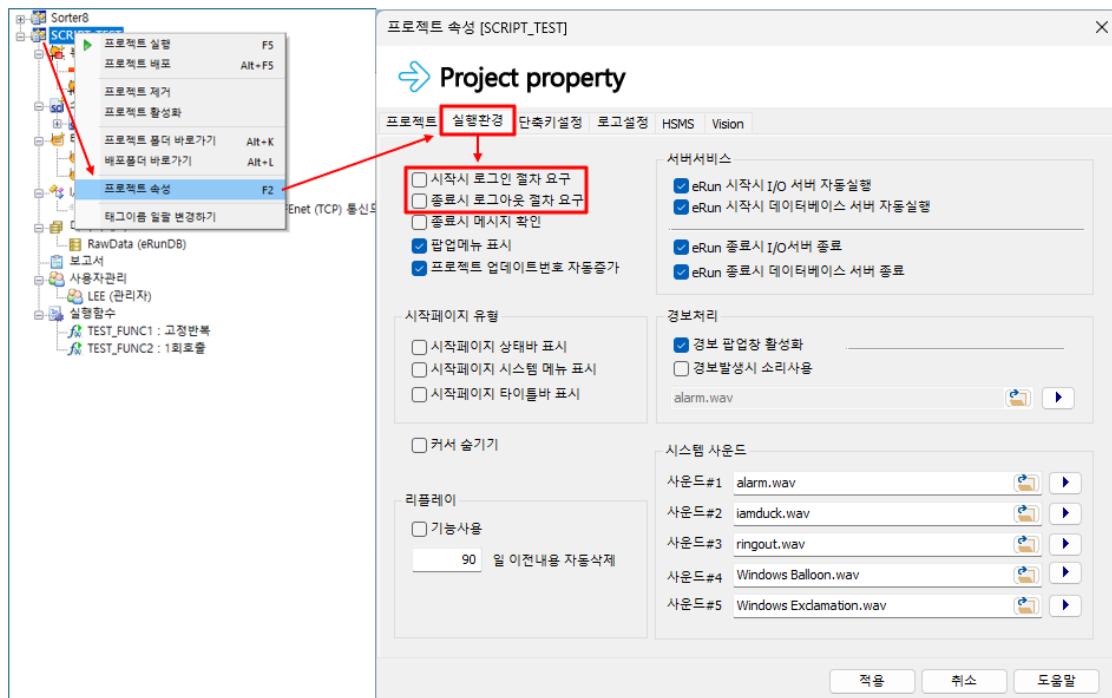
사용자정보를 프로젝트에서 삭제합니다. 프로젝트에서는 등록된 사용자가 최소 하나 이상 있어야 정상적으로 운영할 수 있습니다. 사용자관리 항목에서 마우스 우클릭 하고 팝업메뉴에서 사용자 삭제를 선택합니다.



11.3 프로젝트에서 사용자 정보 활용

프로젝트 실행할 때 사용자 로그인 창을 띄우고 아이디/암호가 일치해야 정상적으로 프로젝트 실행이 가능합니다. 마찬가지로 프로젝트 종료할 때도 사용자 암호가 일치해야 정상적인 종료가 됩니다.

사용자 로그인 창을 띄우려면 프로젝트 환경설정에서 다음과 같이 로그인 옵션사항을 선택해주어야 합니다.



프로젝트 실행될 때 다음과 같이 로그인 창이 나타납니다.



제 11 장 사용자 관리

Login

사용자명 암호 아이디저장

[로그인]

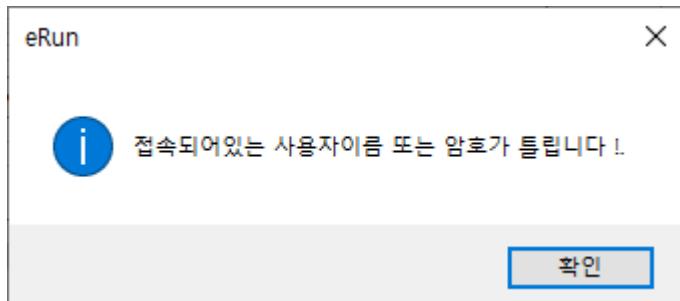
프로젝트 종료할 때 다음과 같이 사용자 로그아웃 창이 나타납니다. 이 때 사용자명은 시작할 때 선택된 사용자 명이 표시가 됩니다. 이 사용자 암호를 입력해야만 정상적으로 종료가 됩니다.

LogOut

사용자명 1 암호 아이디저장

[로그아웃]

사용자명 '1'에 대한 암호가 틀린 경우 아래와 같이 메시지 창이 뜨고 정상적인 종료가 안됩니다.



제 12 장 실행함수

eRun SCADA는 다중 프로세스를 지원하는 솔루션입니다. 다시 말해서 순차적으로 진행하는 것이 아니고 동시에 여러가지 동작에 대한 제어를 할 경우 제공되는 기능입니다.

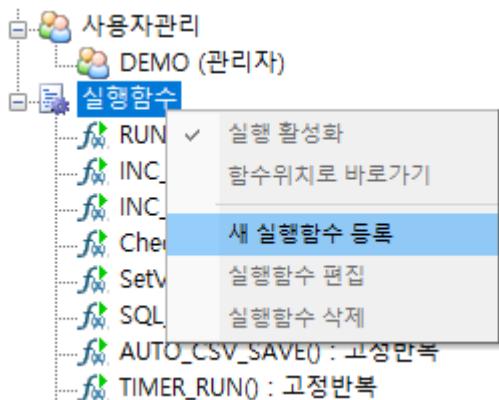
운영 목적으로 작성된 스크립트 함수를 주기적으로 또는 조건에 따라 반복적으로 실행합니다.

예를 들어서, 아래와 같은 경우 사용하면 활용도가 좋습니다.

- ① 주기적으로 난수값을 발생시켜 태그에 써야하는 경우
- ② 태그데이터를 3rd Party DBMS에 주기적으로 저장을 해야 하는 경우
- ③ 설정된 조건 (레시피)에 따른 다양한 데이터를 자동으로 처리해야 하는 경우 등
- ④ 디바이스 통신 트리거 신호를 받고 실행해야 하는 프로세스 제어

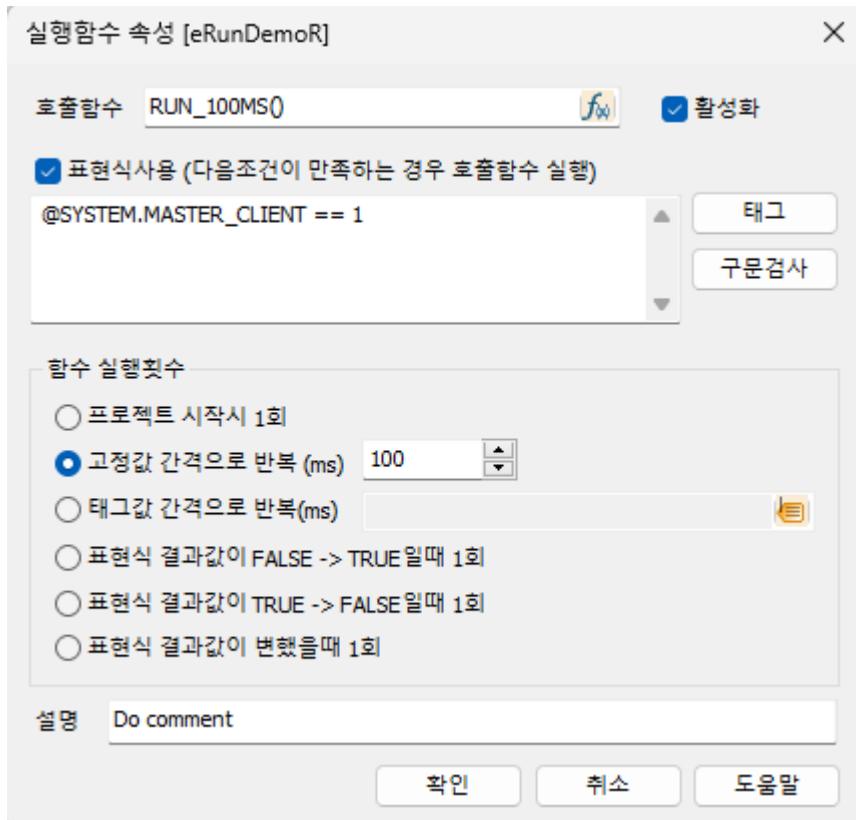
12.1 실행함수 추가하기

프로젝트 관리에서 실행함수 메뉴를 선택하고 마우스 우측버튼을 클릭하면 팝업메뉴가 나타납니다.



- ① **실행 활성화** : 런타임에 함수 실행여부를 설정합니다.
- ② **함수위치로 바로가기** : 실행함수가 정의된 위치로 이동합니다.
- ③ **새 실행함수 등록** : 새로운 실행조건과 함수를 등록합니다.
- ④ **실행함수 편집** : 이미 등록된 실행함수 실행조건 창을 열어서 편집합니다.
- ⑤ **실행함수 삭제** : 실행함수 항목을 리스트에서 삭제합니다.

12.2 실행함수 편집



12.3 속성창 설명

항목	설명
활성화	함수실행을 하려면 활성화 속성이 체크되어 있어야 합니다. 등록은 되었지만 실행을 보류해야하거나 할 경우 선택을 해제합니다.
호출함수	<p>사용자가 작성한 스크립트 함수 이름을 입력합니다. 브라우저 버튼을 누르면 아래와 같이 이미 작성된 스크립트 함수 목록이 나타납니다.</p> <div style="border: 1px solid #ccc; padding: 5px; width: fit-content;"> <p>사용자정의 함수리스트[SARCO]</p> <ul style="list-style-type: none"> COMMON <ul style="list-style-type: none"> f_w ChkPassword0 f_w ChkPassword20 f_w ChkPassword30 f_w InitPwd0 f_w initReceptor0 f_w main0 f_w OpenPagePrice0 f_w OpenPagePrice20 보고서 <ul style="list-style-type: none"> f_w GetLastDay(int) f_w GetTagGroup(int) f_w InitReportAddCombo() f_w InitReportPage() f_w ReportAutoManFlagReset() f_w ReportDaily(string) </div>
표현식 사용	<p>표현식의 결과값이 만족하는 경우 해당 스크립트 함수를 실행합니다. 표현식에는 태그명이나, 논술연산, 사칙연산도 적용이 가능합니다. (ex) 다음과 같은 표현식이 가능합니다.</p>

	<p>① @TAG.PRESS ② @TAG.PRESS && @TAG.TEMP >100.0 ③ (@TAG.SUM / 100) > 10</p>
함수실행 횟수	<p>■ 프로젝트 시작시 1회 프로젝트 실행할 때 초기화가 필요한 부분을 한 번만 실행합니다. 예를 들어서 그리드의 초기화, 데이터베이스 초기화 등 초기화가 필요한 곳에 사용을 합니다.</p> <p>■ 고정값 간격으로 반복(ms) 주기적으로 반복하면서 스크립트 함수를 호출합니다. 단위는 ms이고 기본값은 100ms입니다.</p> <p>■ 태그값 간격으로 반복(ms) 설정된 태그의 값이 반복실행시간으로 설정됩니다.</p> <p>■ 표현식 결과값이 FALSE -> TRUE일 때 1회 표현식 사용이 설정이 되어야 하며 표현식 결과의 값이 FALSE에서 TRUE로 변경시 한 번만 호출이 됩니다.</p> <p>■ 표현식 결과값이 TRUE -> FALSE일 때 1회 표현식 사용이 설정이 되어야 하며 표현식 결과의 값이 TRUE에서 FALSE로 변경시 한 번만 호출이 됩니다.</p> <p>■ 표현식 결과값이 변했을 때 1회 표현식 결과값이 변경되었을 때 한 번 호출이 됩니다.</p>

제 13 장 심볼관리

eRun SCADA에서는 설비제어나 모니터링 감시를 그래픽화 해서 직관적인 운영을 가능하게 하도록 여러가지의 그래픽 오브젝트를 제공합니다. 이것은 설비 동작의 정상적인(NORMAL) 상태 또는 비정상적인(FAIL) 상태를 이미지로 표현하거나, 움직임 상태를 동적인 이미지로 연출하여 보다 효율적인 운전감시 기능을 제공합니다.

스튜디오에서는 다양하게 디자인된 정적인 심볼이미지(**라이브러리**로 표현)와 동적인 심볼이미지(**애니메이션**으로 표현)를 제공합니다. 그리고 사용자가 직접 이미지를 가지고 라이브러리 또는 애니메이션을 제작해서 사용할 수 있습니다. 스튜디오에서는 뷰페이지 편집에서 적당한 이미지 또는 애니메이션을 그려 넣고 태그를 연결해서 감시화면을 보다 화려하고 동적으로 제작할 수 있습니다.

- 간단하게 애니메이션 파일 제작
- 태그 값에 의한 프레임 제어 가능
- 동적인 이미지 제작 가능
- 사용자에 의한 프레임 편집 가능
- 하나의 태그를 이용하여 다양한 애니메이션 연출 가능
- 기존의 애니메이션 수정이 용이
- 프레임 별 유지시간 다르게 설정 가능
- 모든 이미지 포맷파일 적용 가능

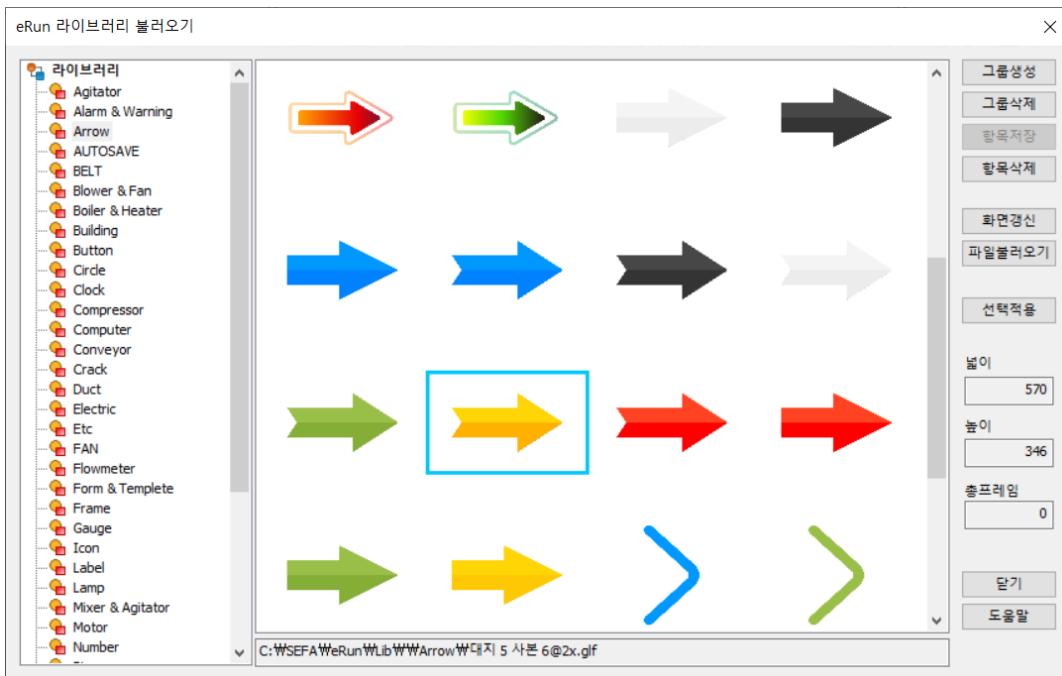
13.1 정적 라이브러리

라이브러리 파일은 분야별 필요한 이미지들을 모아 놓은 정적 심볼이미지 파일입니다. 뷰 페이지에서는 이미지 오브젝트로 사용하며 속성을 들어가보면 최대 6개의 다른 상태를 표현하는 이미지로 설정할 수 있습니다.

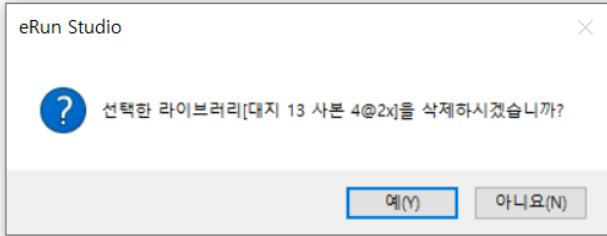
스튜디오 상단의 주 메뉴 도구바에 보면 아래와 같이 정적심볼 라이브러리 창을 불러오는 아이콘이 있습니다. 뷰 페이지에 마우스 초점이 있는 경우에만 아이콘이 활성화된 모양입니다. 또는 뷰페이지에서 마우스 우클릭 하면 나타나는 팝업메뉴에서 [라이브러리 불러오기]를 선택할 수 있습니다.



다음과 같이 라이브러리 선택창이 나타납니다. 사용자가 직접 외부 이미지를 불러와서 라이브러리에 추가해서 프로젝트에 사용할 수도 있습니다.



라이브러리 트리목록에서 항목을 선택하면 그 항목에 해당하는 심볼이 보이고 마우스로 더블 클릭하면 뷰 페이지로 선택한 심볼 이미지가 표시됩니다.

항목	설명
그룹생성	트리목록에 라이브러리 그룹이름을 생성합니다. 유사한 이미지를 모아서 등록해 두면 됩니다.
그룹삭제	트리목록에서 라이브러리 그룹을 제거합니다.
항목저장	선택한 이미지를 라이브러리 리스트에서 삭제합니다.
항목삭제	 
화면갱신	
파일불러오기	라이브러리에 추가할 이미지를 불러옵니다.
선택적용	선택한 심볼 이미지를 뷰 페이지에 삽입하고 표시합니다.

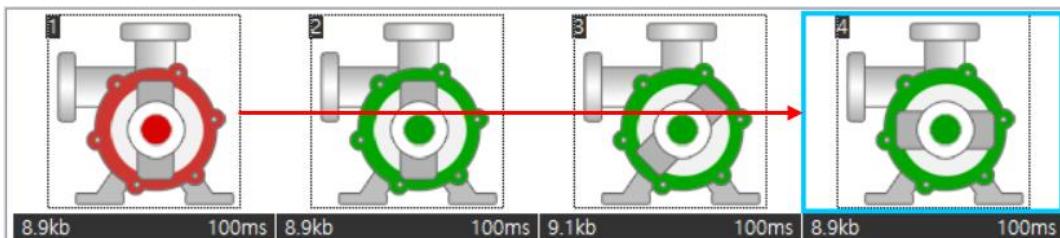
뷰 페이지에 삽입된 라이브러리 속성에 들어가면 아래와 같이 2개의 라이브러리를 삽입해서 표현식 태그 값이 0일 때 이미지, 태그값이 1일 때 보여지는 이미지로 구분해서 설정해 놓습니다. 이렇게 최대 6개의 서로 다른 상태 이미지를 설정해서 6개의 감시상태를 표현할 수 있습니다. 오브젝트 속성에 대한 자세한 설명은 이미지 오브젝트 속성 설정부분을 참고하세요.



13.2 동적 라이브러리(애니메이션)

애니메이션 파일은 분야별 필요한 이미지들을 모아서 움직이는 동적 심볼 이미지 파일입니다. 뷰 페이지에서는 애니메이션 오브젝트를 이용해서 감시 설비의 동적인 움직임을 조절해서 보다 사실적인 모니터링 서비스를 제공합니다. 정적 심볼라이브러리는 6개의 움직임 상태가 있지만, 애니메이션은 다수의 프레임으로 동작상태를 구성하는데 그 수가 제한이 없습니다. 그래서 다양한 상태를 세밀하게 표현이 가능합니다.

아래 애니메이션 파일은 4개의 상태 이미지로 프레임을 구성하였습니다. 각 프레임번호를 태그 값과 연동하여 그 값에 해당하는 프레임 이미지를 표시합니다. 또는 태그 값이 ON(1)일 때 계속 반복적으로 움직이고 OFF(0)일 때 정지하게 설정할 수 있습니다.

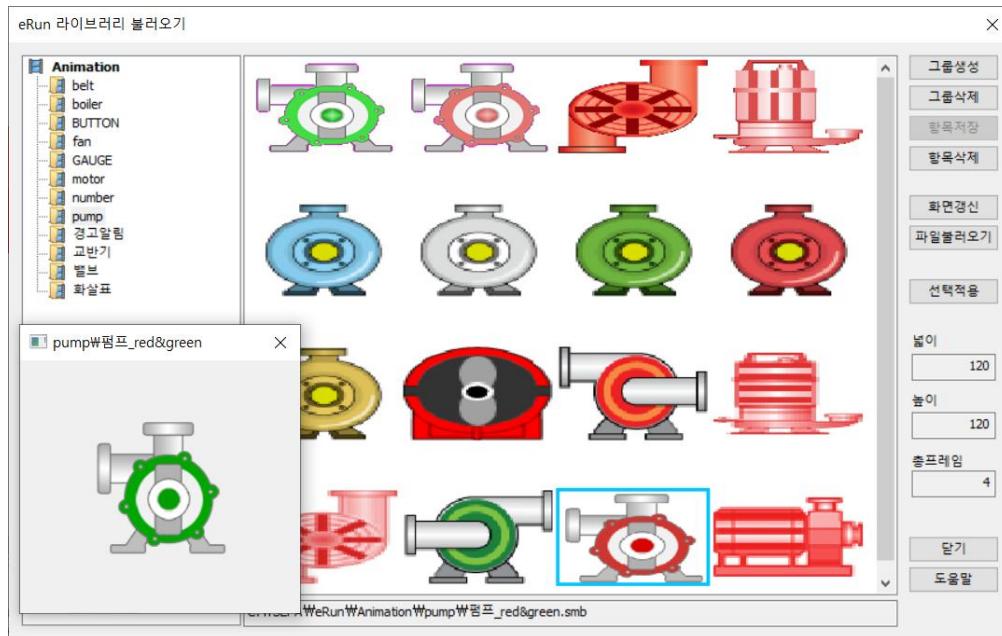


스튜디오 뷰페이지 원도우 도구바에서 애니메이션 선택창을 불러오는 아이콘이 있습니다. 뷰 페이지에 마우스

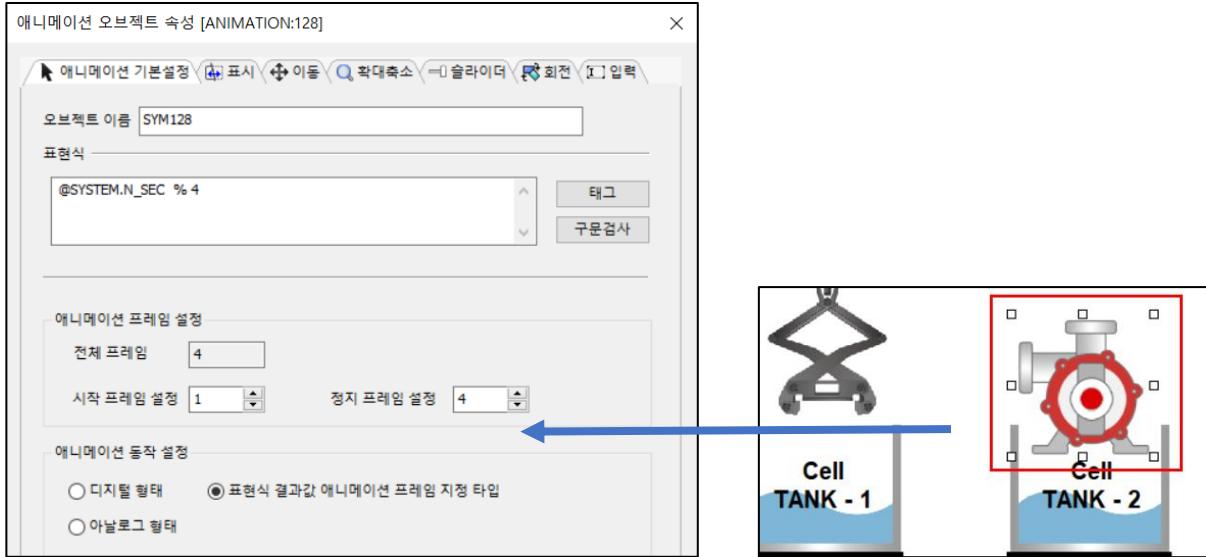
초점이 있는 경우에만 아이콘이 활성화된 모양입니다.



다음과 같이 애니메이션 파일 선택창이 나타납니다. 사용자가 [애니메이션 생성] 기능을 사용해서 직접 애니메이션 파일을 제작해서 애니메이션 라이브러리에 추가해서 프로젝트에 사용할 수도 있습니다.

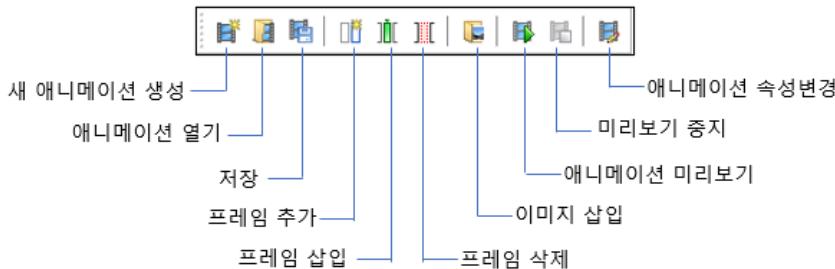


애니메이션 창의 기능도 정적 라이브러리 창의 기능과 유사합니다. 심볼을 선택하면 미리보기 창이 나타나서 동적인 이미지를 확인할 수 있습니다. 아래 그림은 우측의 애니메이션 오브젝트의 속성창입니다. 자세한 설명은 애니메이션 속성설정 설명부분을 참고하세요.



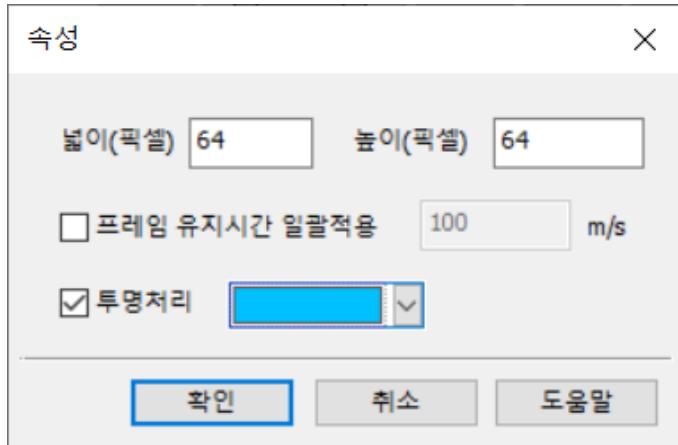
13.3 애니메이션 파일 생성하기

eRun SCADA에서는 감시화면에서 동적으로 움직이는 설비상태를 표현하려는 애니메이션을 직접 제작할 수 있습니다. 스튜디오의 애니메이션 편집 도구바는 아래와 같습니다.



새 애니메이션

애니메이션 오브젝트는 파일단위로 구성되어 있으며 여러 개의 프레임을 추가해서 각각의 프레임에 상태 이미지를 삽입해서 제작합니다. 아래 그림과 같이 새 애니메이션의 가로, 세로크기를 설정해서 생성합니다.



항목	설명
넓이	픽셀단위의 이미지 넓이
높이	픽셀단위의 이미지 높이
프레임 유지시간 일괄적용	모든 프레임 표시유지 시간을 동일하게 적용합니다. (Interval)
투명처리	체크 박스 체크시에는 프레임 이미지에 대하여 투명 처리합니다.

불러오기

애니메이션 파일을 불러옵니다. (기존의 애니메이션을 편집할 경우 활용합니다.)

저장

애니메이션 파일을 저장합니다. 애니메이션 저장 버튼을 클릭하면 아래와 같이 저장할 위치를 설정하는 창이 나오는데 여기서 애니메이션 폴더를 지정(**C:\WeRun\Animation**)하여 애니메이션 폴더내에 저장하시면 됩니다.

프레임 추가

애니메이션의 이미지를 넣을 프레임을 추가합니다.

프레임 삽입

프레임 중간에 새 프레임을 삽입합니다.



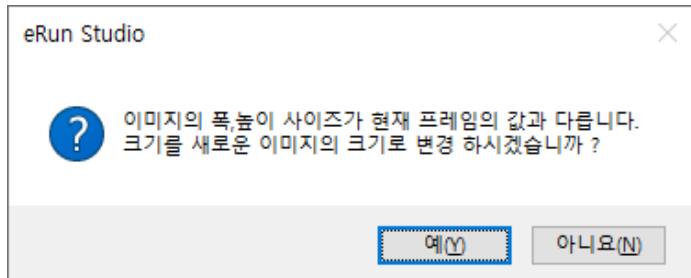
프레임 삭제

선택한 프레임을 삭제합니다.



이미지 삽입

현재 프레임에 표시할 이미지 파일을 불러옵니다. 이때 이미지 넓이와 폭 사이즈가 현재 프레임에 설정된 사이즈와 다른 경우 사이즈 변경을 묻는 메시지가 뜹니다.



'예'를 선택하면 모든 프레임의 사이즈를 새로 선택한 이미지의 넓이, 폭사이즈로 변경합니다.
프레임에 이미지를 복사(CTRL+C)해서 붙여넣기(CTRL+V)도 동일하게 이미지 삽입이 됩니다.



애니메이션 미리보기

애니메이션을 플레이하여 미리보기로 확인합니다.



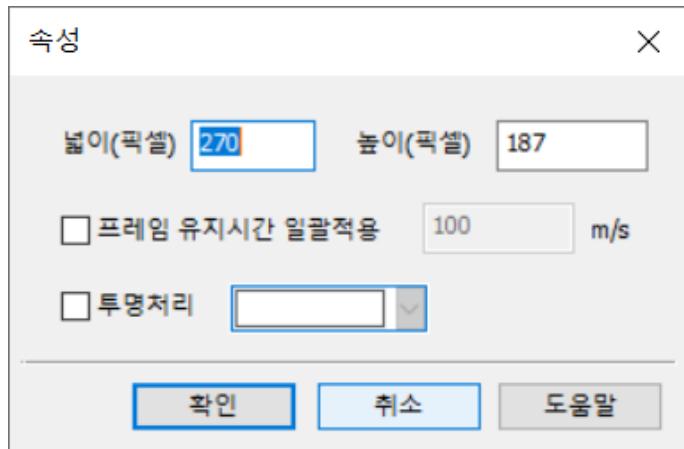
미리보기 중지

미리보기를 중지합니다.



애니메이션 속성변경

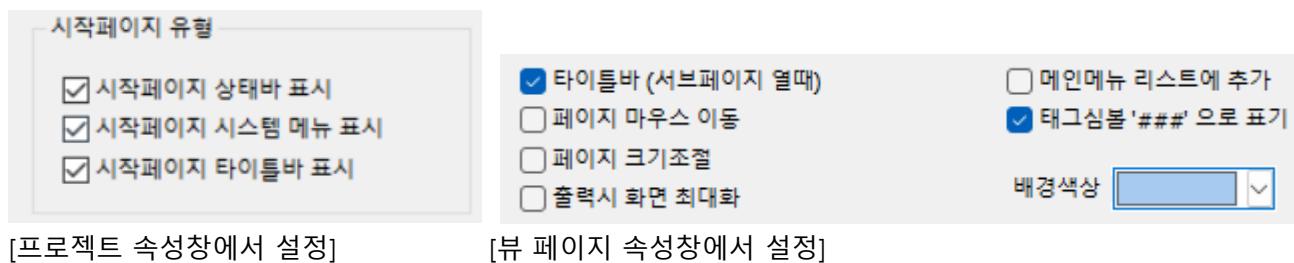
애니메이션의 속성을 변경합니다.



제 14 장 프로젝트 실행

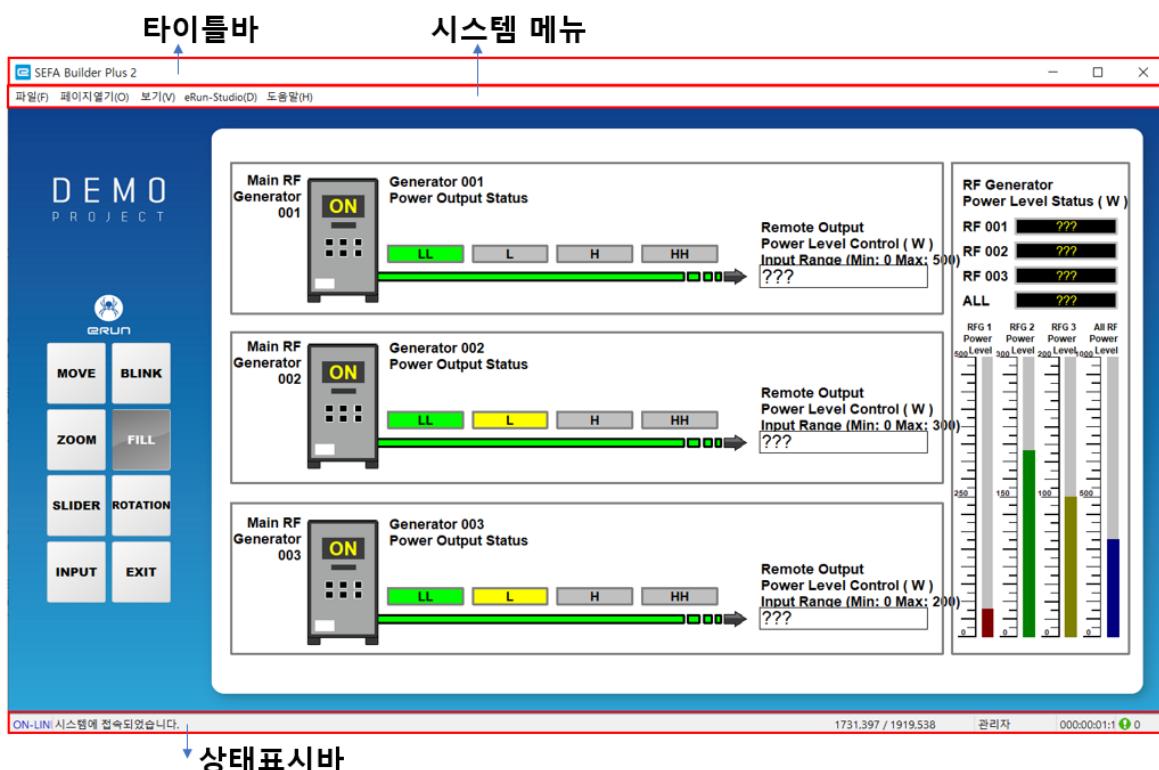
Studio에서 편집된 프로젝트를 실행하는 단계입니다. 프로젝트 실행은 개발단계에서 실행하는 방법과 최종 사용자가 실행하는 3가지 방법이 있습니다. 이부분은 앞서 [3.2 런타임 실행하기] 편에서 설명드렸으니 참고하시기 바랍니다.

첫 화면으로 나타나는 뷰페이지 창의 형태는 4가지 형태가 있는데, [프로젝트 속성 - 실행환경]에서 선택할 수 있습니다. 프로그램을 실행할 때 첫 화면 형태만 아래와 같이 프로젝트 속성에서 하며, 서브 페이지를 열 때 나타나는 형태는 각 뷰페이지 속성에서 선택을 할 수 있습니다.



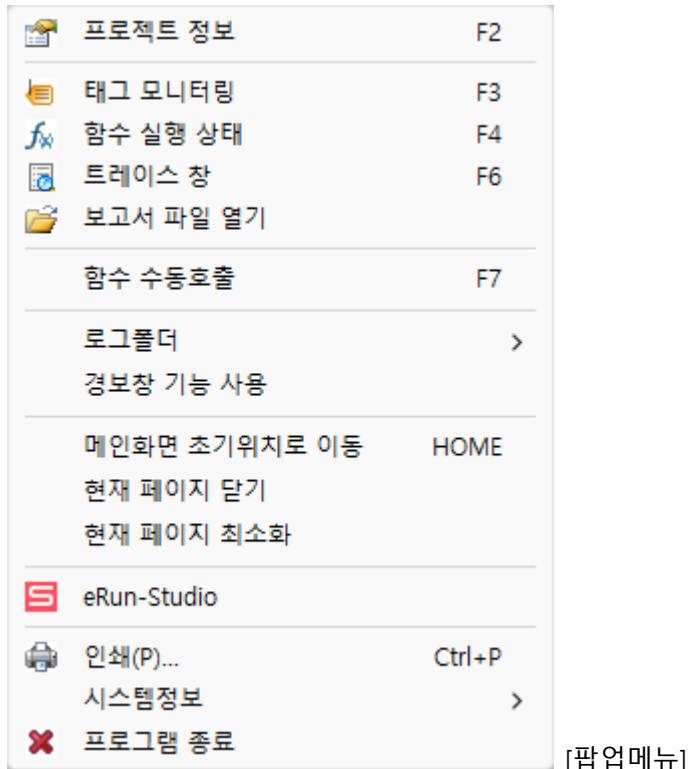
[프로젝트 속성창에서 설정] [뷰 페이지 속성창에서 설정]

다음 그림은 프로젝트 실행 첫 화면입니다.



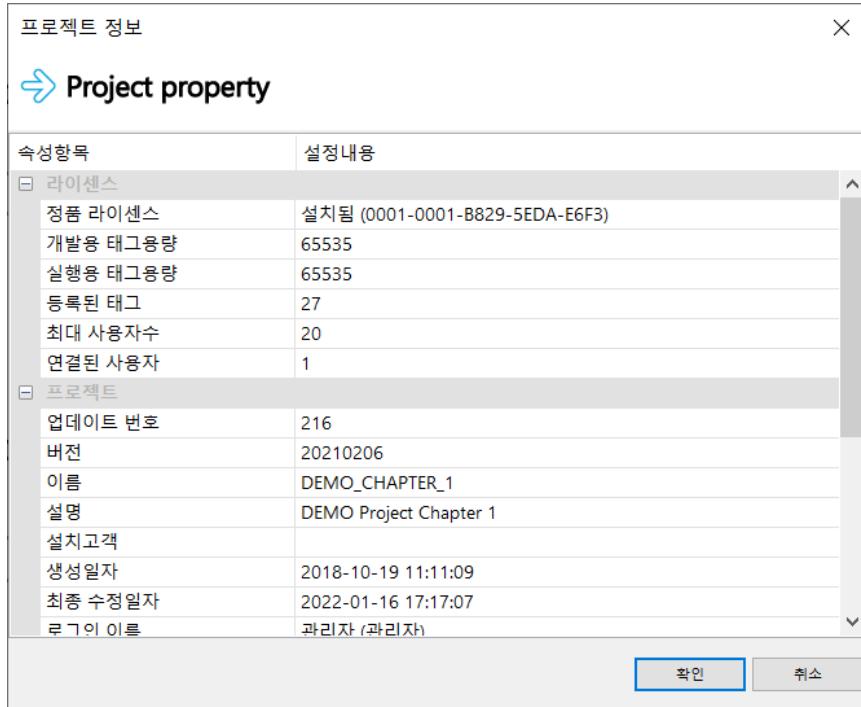
14.1 런타임 메뉴

뷰 페이지에서 마우스 오른쪽 버튼 클릭하면 팝업메뉴가 나타납니다. 팝업메뉴는 프로젝트 편집할 때 주로 사용하며 프로젝트 완성이 된 이후에는 사용을 하지 않습니다. 팝업메뉴에는 여러가지 기능이 있는데 다음에서 설명드리겠습니다.



■ 프로젝트 정보

라이선스 내용, 프로젝트 속성, I/O서버, DB서버, 사용자 정보 창입니다.



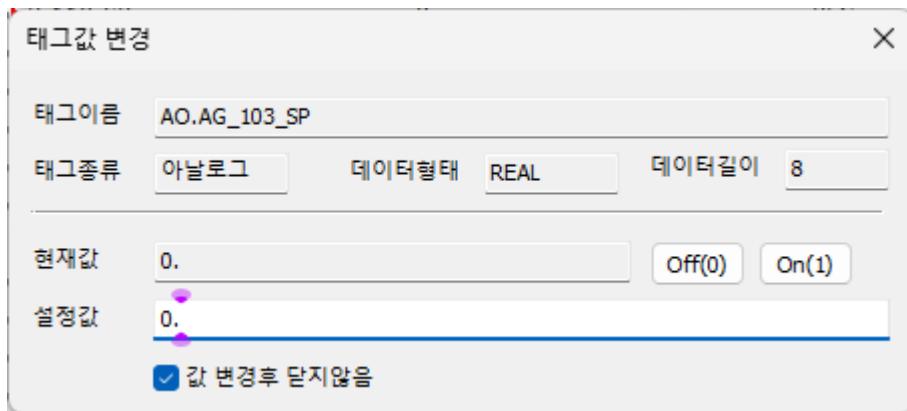
■ 태그 모니터링

태그 그룹별 리스트에 태그값 리스트창을 표시합니다. 태그값을 모니터링 하면서 특정 태그의 값을 변경할 수 있습니다. 실태그인 경우 디바이스로 값을 내보낼 수 있습니다. 태그를 마우스로 두번클릭하면 변경 창이 나타납니다.

태그상태 모니터링 [그룹 : MUX1] - Total Tags : 697				
	태그명	태그값	디바이스	어드레스
▶ SYSTEM (12)	▶▶ 통신상태	0 (0x0000)	MUX11	40101:0
▶ MAIN (19)	▶▶ COMM1	0 (0x0000)	MUX11	40101:0
▶ VIRTUAL (13)	▶▶ FLOW1	0.	MUX11	40101:3
▶ SUMMARY (78)	▶▶ VOL1	0.	MUX11	40101:7
▶ MUX1 (85)	▶▶ TEMP1	0.	MUX11	40101:9
▶ MUX2 (61)	▶▶ PRESS1	0.	MUX11	40101:15
▶ MUX4 (25)	▶▶ TOTAL1	0.	MUX11	40101:25
▶ MUX3 (68)	▶▶ TOTALVOL1	0.	MUX11	40101:29
▶ MUX5 (91)	▶▶ COMM2	0 (0x0000)	MUX11	40201:0
▶ MUX6 (38)	▶▶ FLOW2	0.	MUX11	40201:3
▶ MUX7 (25)	▶▶ VOL2	0.	MUX11	40201:7
▶ MUX8 (35)	▶▶ TEMP2	0.	MUX11	40201:9
▶ MUX9 (40)	▶▶ PRESS2	0.	MUX11	40201:15
▶ MUX10 (54)	▶▶ TOTAL2	0.	MUX11	40201:25
▶ DATA (13)	▶▶ TOTALVOL2	0.	MUX11	40201:29
▶ DCS (17)	▶▶ COMM3	0.	MUX11	40301:0
▶ REPORT (20)	▶▶ FLOW3	0.	MUX11	40301:3
▶ TREND (3)				

태그값 변경창은 다음과 같습니다. 태그이름과 종류, 데이터형, 데이터길이, 태그 현재값이 표시됩니다. 설정값

입력창에 변경값을 입력하고 Enter를 누르면 태그값이 변경되고 디바이스로 출력됩니다.



■ 함수 실행 상태

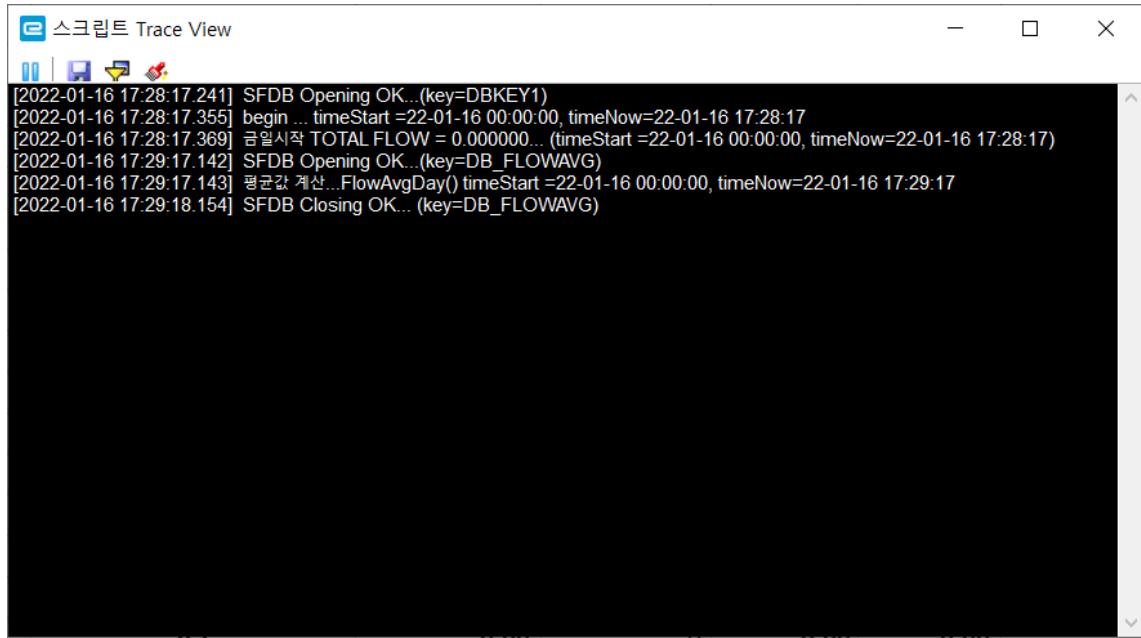
스크립트 실행함수의 실행정보 창을 표시합니다. 스크립트 실행함수는 기존에 뷰페이지마다 실행함수 오브젝트를 삽입하던 것을 프로젝트 구성 관리자의 실행함수 영역에 함수를 등록하는 방식으로 변경되었습니다. 기능이 변경됨에 따라 실행함수 모니터링도 모든함수가 한꺼번에 표시되는 방식으로 변경되었습니다.

함수이름	실행상태	동작설정	반복속도(m/s)	표현식 결과값	설명
f_w InitGui()	finished	한번만 실행	--	1.000	Do comment
f_w NEW_AUTO_START()	x	0 -> 1	--	0.000	Do comment
f_w TEST_Trigger()	x	고정반복 실행	1000	0.000	Do comment
f_w HEART_BIT()	run	고정반복 실행	1000		Do comment
f_w ALARM_STATUS_CHECK()	run	고정반복 실행	500		Do comment
f_w CALC_WORK_TIME()	x	0 <-> 1	--	0.000	Do comment
f_w ADEX_STEP_PROCESS2()	x	0 -> 1	--	0.000	Do comment
f_w HEART_BIT_ONLY()	run	고정반복 실행	500		Do comment
f_w AUTO_PROC_START()	x	0 -> 1	--	0.000	Do comment
f_w REFRESH_PROD_DATA_TABLE()	x	0 <-> 1	--	4.000	Do comment
f_w ADMIN_GET_STAN_VALUE()	x	0 -> 1	--	0.000	Do comment
f_w PASSWORD_CHANGE()	x	0 -> 1	--	0.000	Do comment
f_w MANUAL_MEASURE()	x	0 -> 1	--	0.000	Do comment
f_w ADEX_STEP_PROCESS_MAN()	x	0 -> 1	--	0.000	Do comment
f_w ADMIN_GET_STAN_VALUE_STOP()	x	0 -> 1	--	1.000	Do comment
f_w ALARM_EXPORT_CSV()	x	0 -> 1	--	0.000	Do comment
f_w SORT_COND_GET_STAN_VALUE()	x	고정반복 실행	10	0.000	Do comment
f_w RANK_HI_LO_CALC_APPLY()	x	0 -> 1	--	0.000	Do comment
f_w SORT_COND_GET_STAN_VALUE_STOP()	x	0 -> 1	--	1.000	Do comment
f_w LOGIN()	x	0 -> 1	--	0.000	Do comment

■ 트레이스 창

스크립트 실행상태를 모니터링 하거나 특정 프로세스의 진행방향등을 점검할 때 사용합니다. 주로 스크립트 함

수에서 `_Trace()`, `_TraceEx()` 와 같은 함수를 사용하면 트레이스 창에 그 결과가 표시됩니다.

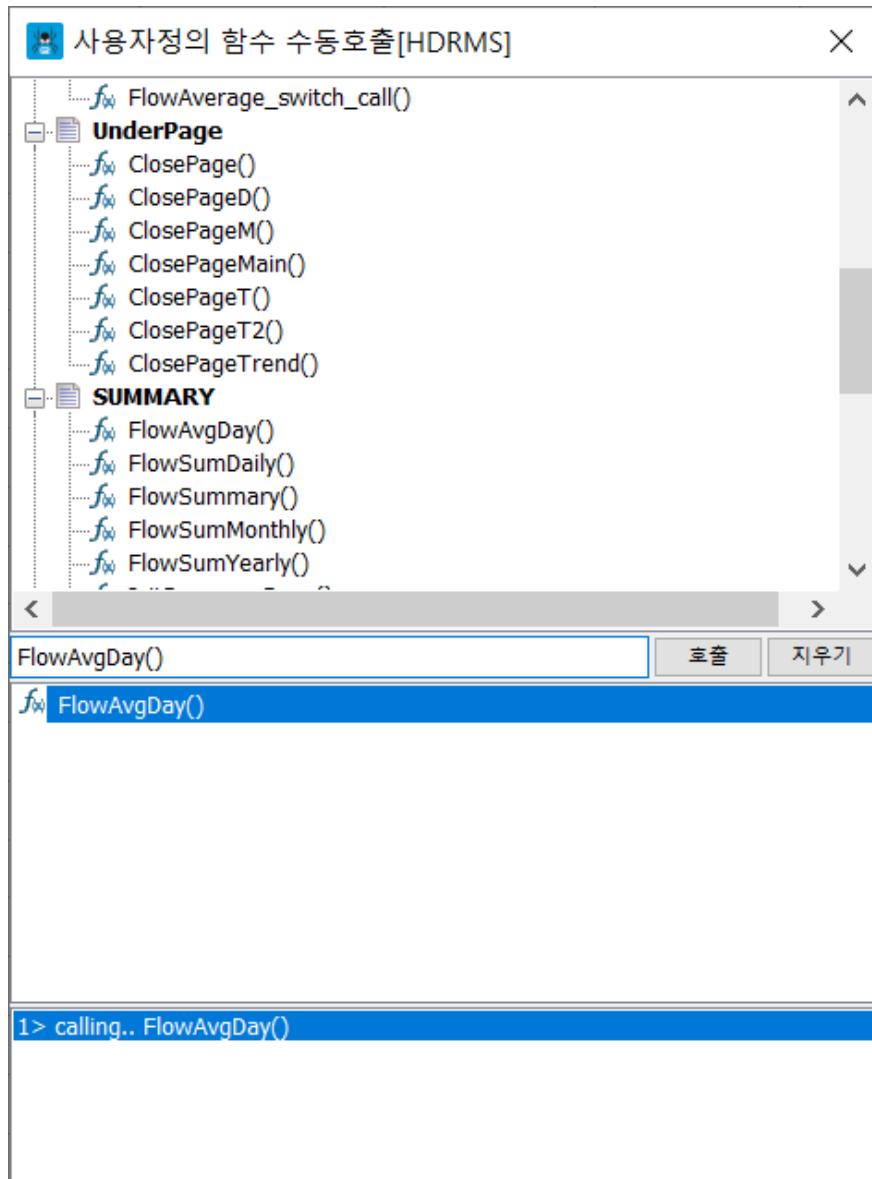


■ 보고서 파일열기

SF-FORM으로 작성된 보고서파일을 미리보기 합니다.

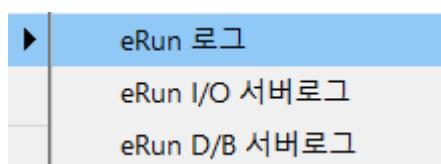
■ 함수 수동호출

사용자가 코딩한 스크립트 함수 목록을 표시합니다. 프로그램 시험을 하기위해서 특정 함수를 호출합니다. 사용자 함수 수동호출 기능은 프로젝트 편집할 때 임시로 정의된 함수를 실행하거나 할 경우 편리하게 사용합니다.



■ 로그폴더

프로젝트 실행이 되면 텍스트 포맷의 로그파일을 각 폴더에 생성이 됩니다. 선택 메뉴의 해당 로그 폴더창을 표시합니다. 폴더는 프로젝트 Publish 폴더에 Log창이 기록되고 그 아래 3개의 폴더가 생성됩니다. 프로젝트 실행에 버그 등이 있을 경우 로그 파일을 참고합니다.



■ 경보창 기능사용

경보태그에 의한 경보발생이 되었을 때 경보메시지 창이 팝업이 되는데, [경보창 기능사용] 체크가 되어 있지 않으면 경보메시지 창이 나타나지 않습니다.

■ 메인화면 초기위치로 이동

현재 뷰페이지를 eRun을 실행했을 때 표시된 위치로 복귀합니다.

■ 현재 페이지 닫기

현재 뷰페이지를 닫습니다. 메인 뷰페이지에서 페이지 닫기는 프로그램 종료와 동일합니다.

■ 현재 페이지 최소화

현재 페이지를 최소화합니다.

■ eRun-Studio

스튜디오를 실행합니다.

■ 인쇄

현재 뷰페이지 화면을 프린터로 인쇄합니다.

■ 시스템 정보

라이선스 및 웹사이트 바로가기 기능이 있습니다.

■ 프로그램 종료

eRun 프로젝트 실행을 종료합니다.

제 15 장 다중 사용자접속

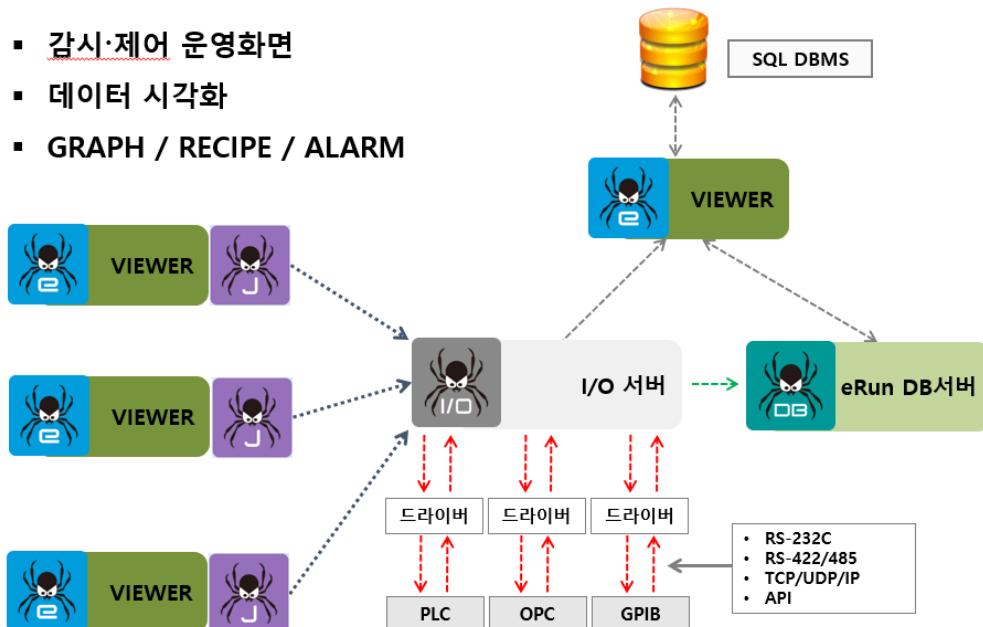
eRun은 다수의 사용자가 하나의 프로젝트를 여러 PC에서 모니터링이 가능한 환경을 제공합니다. JOIN은 이때 사용하기 위한 다중 사용자용 접속 프로그램입니다.

15.1 기능 및 특징

- 원격 I/O서버로부터 프로젝트를 다운로드합니다.
- 여러 원격 프로젝트 실행이 편리합니다.
- 원격 프로젝트 업데이트 이후에 재접속을 통해서 신속하게 동기화 이루어짐.
- 서버 리스트를 통한 다수 프로젝트 관리

MULTI USER

- 감시·제어 운영화면
- 데이터 시작화
- GRAPH / RECIPE / ALARM

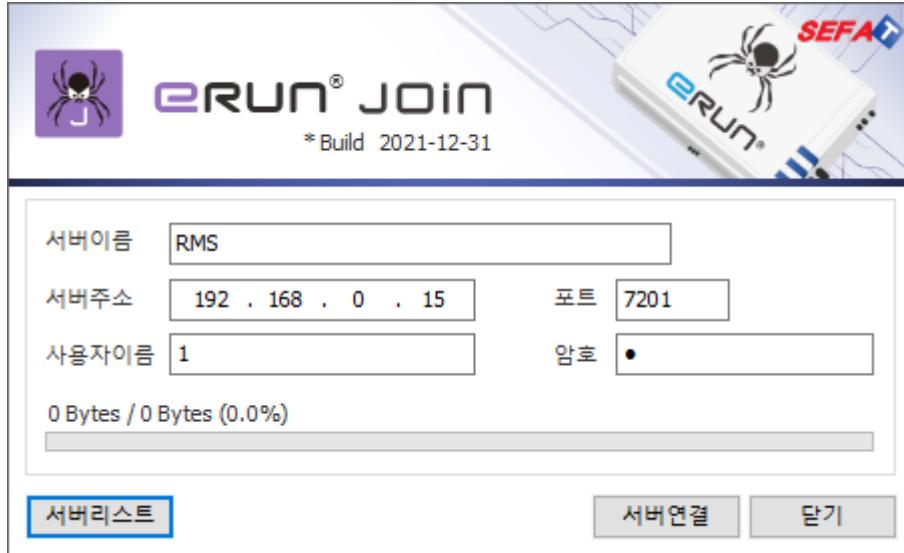


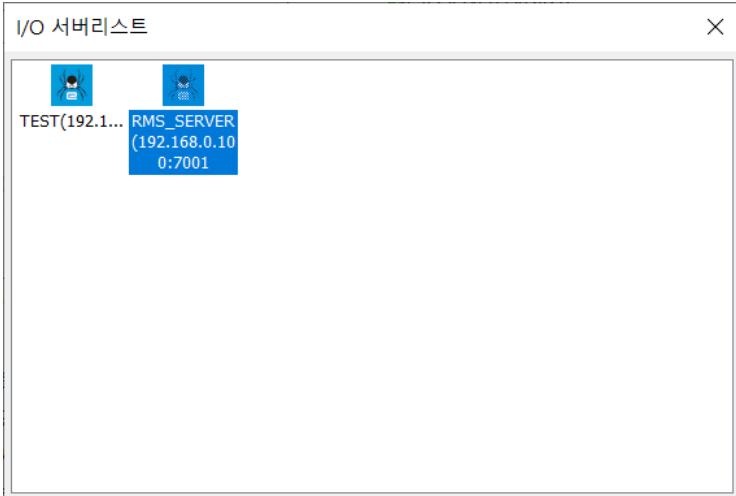
15.2 JOIN 실행

다중 사용자용 접속프로그램 JOIN의 실행은 "C:\WEFA\WeRun 버전"에서 eRunJoin 버튼을 클릭하면 실행됩니다. 이전에 접속했던 원격서버 정보가 있으면 접속할 것인지 메시지 창을 띄워서 물어봅니다. 아니면 직접 서버 접속 IP, 포트, 사용자정보를 입력해서 접속을 할 수 있습니다. 또는 서버접속 정보를 별도의 리스트 파일 (JoinServer.txt)에 기록해두고 편리하게 원격 프로젝트에 접속할 수 있습니다.



[바탕화면 eRunJoin 실행아이콘]



항목	설명
서버이름	원격 서버의 이름을 입력합니다. 프로젝트를 구분하기 위한 이름으로 사용자 임의로 입력하면 됩니다.
서버주소	원격서버의 IP주소를 입력합니다.
포트	서버주소에 대한 프로젝트 I/O 서버 포트번호를 입력합니다.
사용자 이름	원격 프로젝트에 등록된 사용자 이름을 입력합니다.
암호	원격 프로젝트에 등록된 사용자 이름에 대한 암호를 입력합니다.
서버리스트	스튜디오 폴더에 있는 JoinServer.txt 파일에 기록되어있는 원격 프로젝트 서버리스트 항이 나타납니다. 원격서버를 더블클릭해서 접속합니다. 

서버연결	접속정보가 올바르게 입력되었으면 연결을 시도합니다. 연결이 된 후 원격 프로젝트와 현재 PC에 있는 프로젝트와 다르다면 프로젝트 다운로드를 통해서 업데이트 한 후 프로젝트가 실행됩니다.

제 16 장 디바이스 드라이버

본 장에서는 eRun SCADA에서 제공하는 디바이스 I/O 통신 드라이버의 종류와 설정 및 사용방법에 대하여 설명하겠습니다. 산업용 통신 드라이버에는 프로토콜이라는 통신규약에 따라서 디바이스 상호간 통신이 이루어지며 종류도 굉장히 다양합니다. 본 매뉴얼에서는 주로 많이 사용하는 드라이버 중심으로 설명하도록 하겠습니다.

16.1 시리얼 통신이란?

시리얼은 또 다른 말로는 직렬(전송)이라고도 하며, 한 번에 한 가지 사건만이 일어나는 것을 의미합니다. 데이터 비트를 1개의 비트단위로 외부로 송수신하는 방식으로써 구현하기가 쉽고, 멀리갈 수가 있고, 기존의 통신선로(전화선등)를 쉽게 활용할 수가 있어 비용의 절감이 크다는 장점이 있습니다.

직렬통신의 대표적인 것으로 모뎀, LAN, RS232 및 X.25등이 있습니다. 하지만 크게 직렬통신을 구분하면 비동기식 방식과 동기식 방식 2가지로 나누어집니다. 많은 사람들이 비동기식 통신방식을 RS-232C로 알고 있는데 실질적으로 RS-232C라는 것은 비동기식 통신콘트롤러에서 나오는 디지털신호를 외부와 인터페이스 시키는 전기적인 신호 방식의 하나일 뿐입니다. 그러나 일반적으로 RS-232C를 비동기식 통신방식으로 인식하고 있는 것도 큰 무리는 없습니다.

비동기식 통신방식을 지원하는 대표적인 콘트롤러는 NS사의 16C450과 16C550이며 그 외 호환되는 콘트롤러가 다수의 회사에서 생산되지만 성능상의 차이는 없고 호환은 되지 않지만 비동기 통신의 기능을 갖는 콘트롤러는 수십가지의 종류가 있습니다.

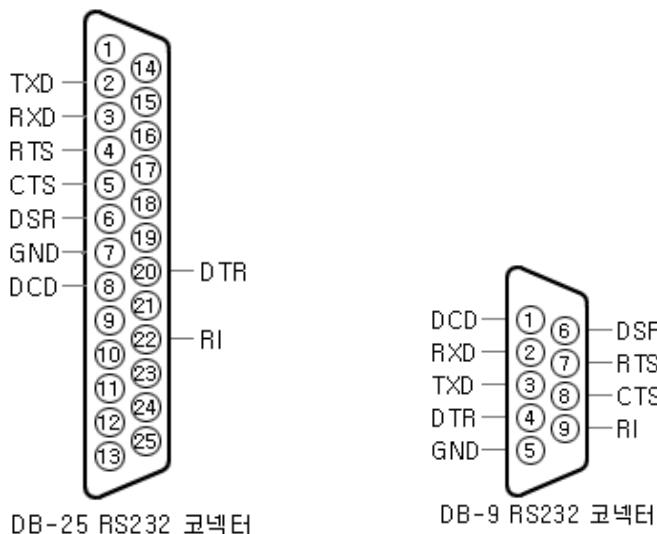
비동기식 통신 콘트롤러를 일반적으로 UART(Universal Asynchronous Receiver/ Transmitter)라 부르는데, UART에서 나오는 신호는 보통 TTL신호레벨을 갖기 때문에 노이즈에 약하고 통신거리에 제약이 있습니다. 이러한 TTL신호를 입력받아 노이즈에 강하고 멀리 갈 수 있게 해주는 인터페이스 IC를 LINE DRIVER/RECEIVER라 부르며 이 중 대표적인 것이 RS-232C, RS-422 및 RS-485가 있습니다.

- RS232, RS422, RS485 등의 Specification 살펴보면 아래와 같습니다.

Specification	RS232C	RS422	RS485
동작 모드	Single-Ended	Differential	Differential
최대 Driver/Receiver 수	1 Driver 1 Receiver	1 Driver 10 Receivers	32 Drivers 32 Receivers
최대 통달거리	약 15 m	약 1.2 km	약 1.2 km

최고 통신속도	20 Kb/s	10 Mb/s	10 Mb/s
신호선 수	9	5	2/4
지원 전송방식	Full Duplex	Full Duplex	Half Duplex
최대 출력전압	$\pm 25V$	-0.25V to +6V	-7V to +12V
최대 입력전압	$\pm 15V$	-7V to +7V	-7V to +12V
장단점	-저속 통신 -통달거리가 짧다. -가장 일반적 방식	-통신속도가 빠름 -통달거리가 길다. -Multi-Drop 지원	-통신속도가 빠름 -통달거리가 길다. -Multi-Drop 지원
적용 예	-시리얼 마우스 -모뎀 -PC COM Port	-원거리 전광판 -PLC 등	-CCTV -PLC 등

■ RS-232C 커넥터 사양

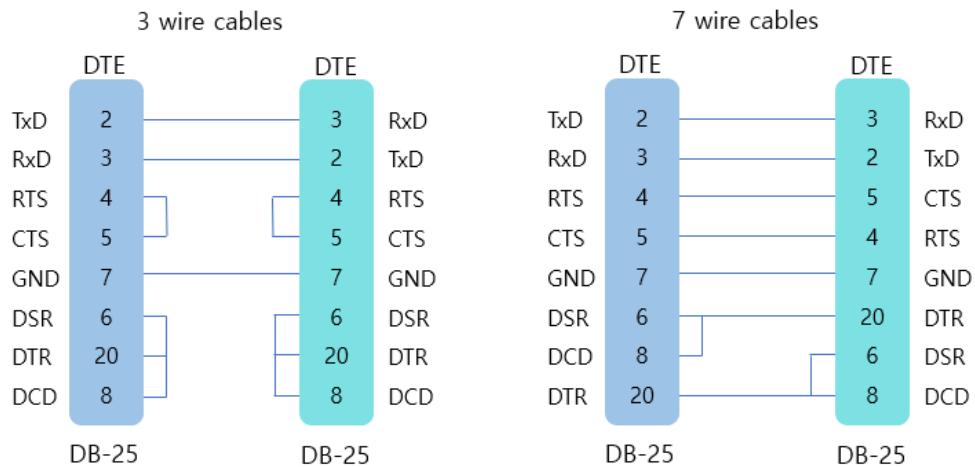


■ 신호선에 대한 설명

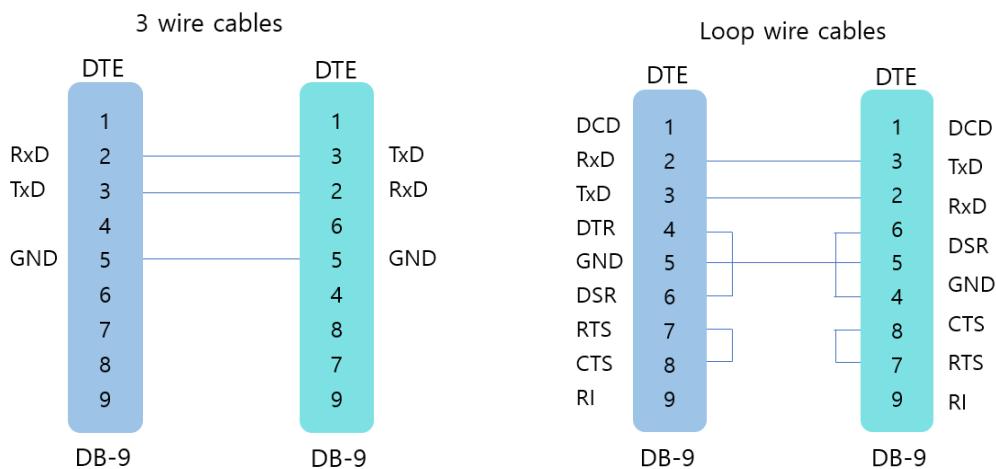
PIN	이름	설명
TxD	Transmit Data	외부장치로 보낼 때 직렬통신 데이터가 나오는 신호선이다
RxD	Receive Data	외부 장치에서 들어오는 직렬통신 데이터를 입력 받는 신호선이다
RTS	Ready To Send	컴퓨터와 같은 DTE장치가 모뎀 또는 프린터와 같은 DCE장치에게 데이터를 받을 준비가 됐음을 나타내는 신호선이다.
CTS	Clear To Send	모뎀 또는 프린터와 같은 DCE장치가 컴퓨터와 같은 DTE장치에게 데이터를

		받을 준비가 됐음을 나타내는 신호선이다.
DTR	Data Terminal Ready	컴퓨터 또는 터미널이 모뎀에게 자신이 송수신 가능한 상태임을 알리는 신호선이며 일반적으로 컴퓨터 등이 전원 인가 후 통신 포트를 초기화한 후 이 신호를 출력시킨다.
DSR	Data Set Ready	모뎀이 컴퓨터 또는 터미널에게 자신이 송수신 가능한 상태임을 알려주는 신호선이며 일반적으로 모뎀에 전원 인가 후 모뎀이 자신의 상태를 파악한 후 이상이 없을 때 이 신호를 출력시킨다.
DCD	Data Carrier Detect	모뎀이 상대편 모뎀과 전화선 등을 통해서 접속이 완료되었을 때 상대편 모뎀이 캐리어신호를 보내오며 이 신호를 검출하였음을 컴퓨터 또는 터미널에 알려주는 신호선이다.
RI	Ring Indicator	상대편 모뎀이 통신을 하기위해서 먼저 전화를 걸어오면 전화 벨이 울리게 된다. 이때 이 신호를 모뎀이 인식하여 컴퓨터 또는 터미널에 알려주는 신호선이며 일반적으로 컴퓨터가 이 신호를 받게 되면 전화벨 신호에 응답하는 프로그램을 인터럽터 등을 통해서 호출하게 된다.

■ RS-232C 결선도 (25-DSUB)

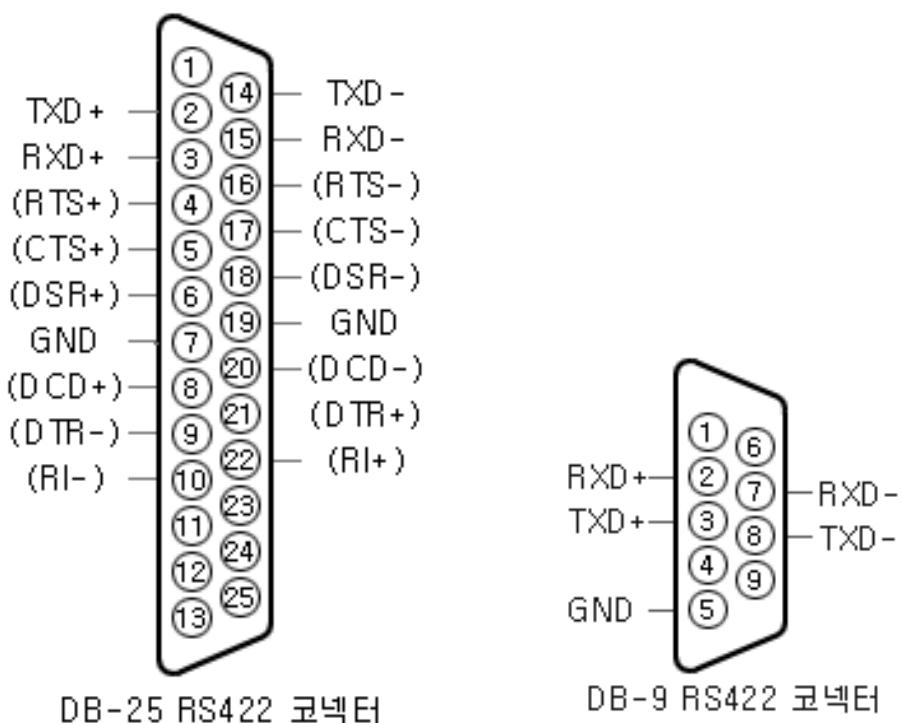


■ RS-232C 결선도 (9-DSUB)

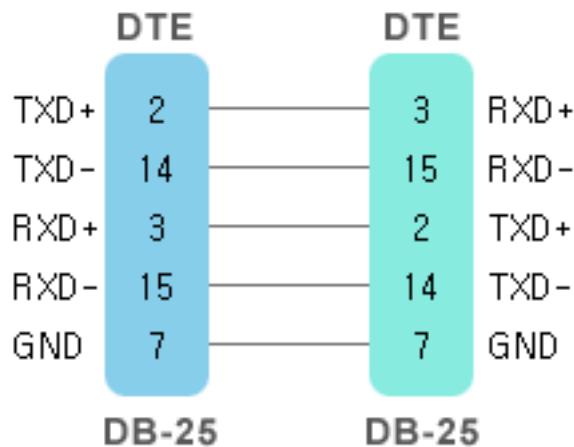


■ RS-422 커넥터 사양

신호선에 할당된 핀번호는 제품 제조사에 따라 틀릴 수 있습니다. 일반적으로 사용되는 신호선은 TXD+, TXD-, RXD+ 및 RXD- 이고 나머지 신호선은 거의 사용되지 않습니다.



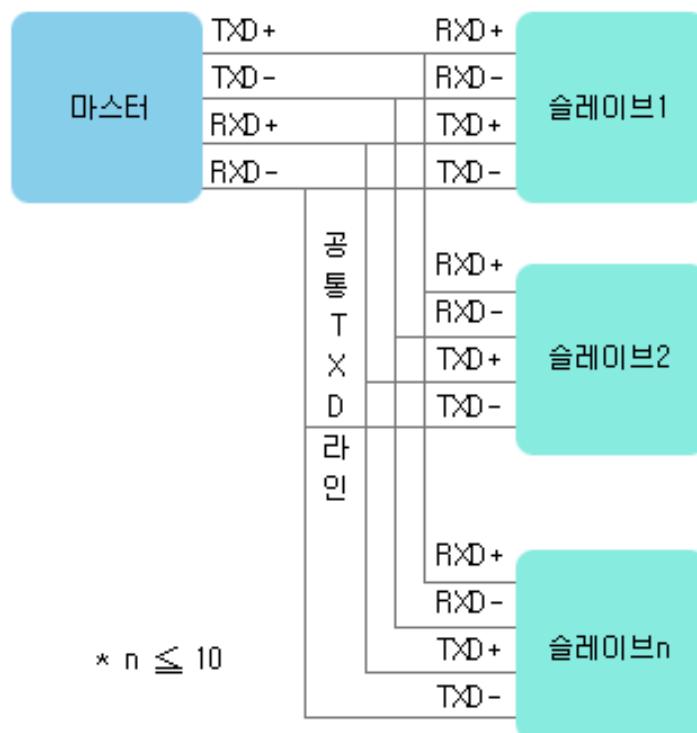
■ RS-422 결선도 (POINT to POINT 모드)



■ RS-422 결선도 (MULTI DROP모드)

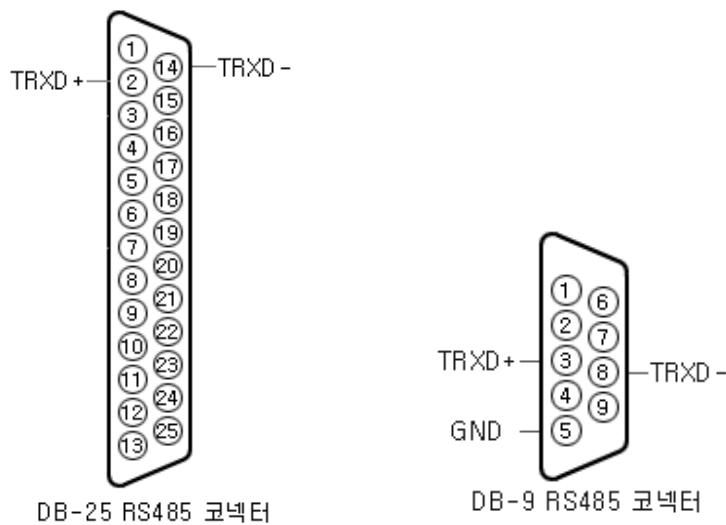
GND는 연결하지 않아도 됨.

극을 맞추어 Plus(+) 신호선은 Plus(+)신호선과, Minus(-)신호선은 Minus(-)신호선과 연결함

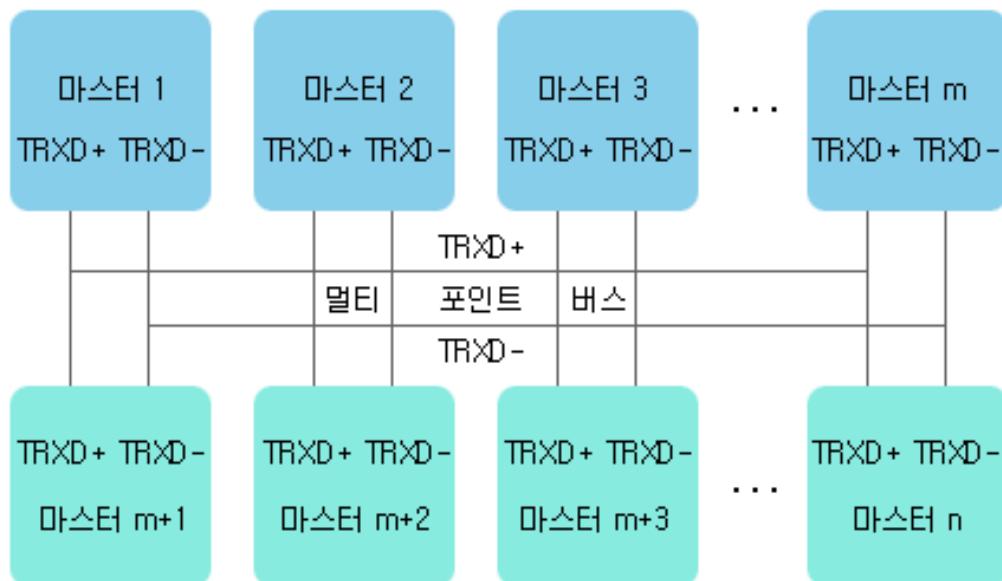


■ RS-485 컨넥터 사양

RS-485 신호선도 RS-422과 마찬가지로 신호선에 할당된 핀번호는 제품 제조사에 따라 틀릴 수 있습니다. 일반적으로 사용되는 신호선은 TXD+, TXD-, RXD+ 및 RXD- 이고 나머지 신호선은 거의 사용되지 않습니다.



■ RS-485 결선도



16.2 아스키코드 테이블

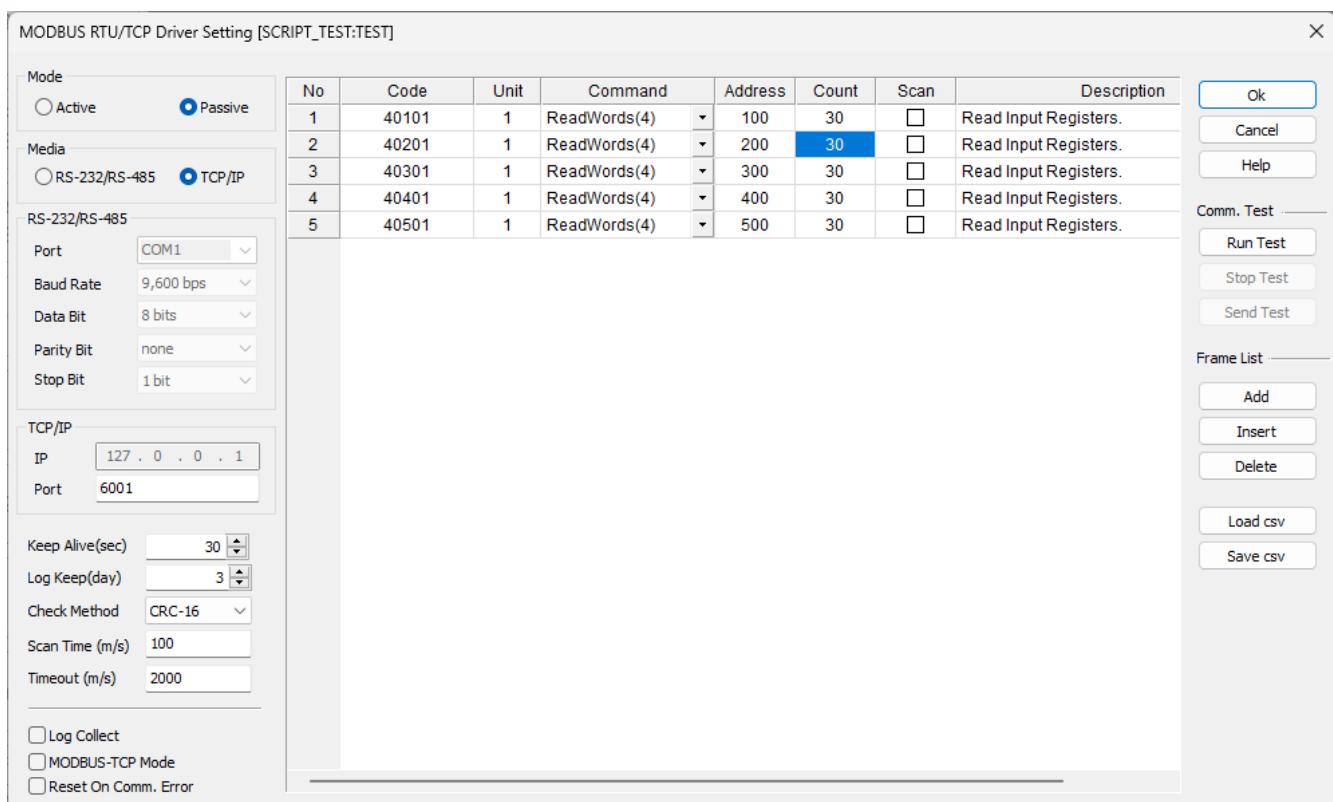
DEC	HEX	CHAR									
0	0x00	NUL	32	0x20	SP	64	0x40	@	96	0x60	'
1	0x01	SOH	33	0x21	!	65	0x41	A	97	0x61	A
2	0x02	STX	34	0x22	"	66	0x42	B	98	0x62	B
3	0x03	ETX	35	0x23	#	67	0x43	C	99	0x63	C
4	0x04	EOT	36	0x24	\$	68	0x44	D	100	0x64	D
5	0x05	ENQ	37	0x25	%	69	0x45	E	101	0x65	E
6	0x06	ACK	38	0x26	&	70	0x46	F	102	0x66	F
7	0x07	BEL	39	0x27		71	0x47	G	103	0x67	G
8	0x08	BS	40	0x28	(72	0x48	H	104	0x68	H
9	0x09	HT	41	0x29)	73	0x49	I	105	0x69	I
10	0x0A	LF	42	0x2A	*	74	0x4A	J	106	0x6A	J
11	0x0B	VT	43	0x2B	+	75	0x4B	K	107	0x6B	K
12	0x0C	FF	44	0x2C	,	76	0x4C	L	108	0x6C	L
13	0x0D	CR	45	0x2D	-	77	0x4D	M	109	0x6D	M
14	0x0E	SO	46	0x2E	.	78	0x4E	N	110	0x6E	N
15	0x0F	SI	47	0x2F	/	79	0x4F	O	111	0x6F	O
16	0x10	DEL	48	0x30	0	80	0x50	P	112	0x70	P
17	0x11	DC1	49	0x31	1	81	0x51	Q	113	0x71	Q
18	0x12	DC2	50	0x32	2	82	0x52	R	114	0x72	R
19	0x13	DC3	51	0x33	3	83	0x53	S	115	0x73	S
20	0x14	DC4	52	0x34	4	84	0x54	T	116	0x74	T
21	0x15	NAK	53	0x35	5	85	0x55	U	117	0x75	U
22	0x16	SYN	54	0x36	6	86	0x56	V	118	0x76	V
23	0x17	ETB	55	0x37	7	87	0x57	W	119	0x77	W
24	0x18	CAN	56	0x38	8	88	0x58	X	120	0x78	X
25	0x19	EM	57	0x39	9	89	0x59	Y	121	0x79	Y
26	0x1A	SUB	58	0x3A	:	90	0x5A	Z	122	0x7A	z
27	0x1B	ESC	59	0x3B	;	91	0x5B	[123	0x7B	{
28	0x1C	FS	60	0x3C	<	92	0x5C	\	124	0x7C	
29	0x1D	GS	61	0x3D	=	93	0x5D]	125	0x7D	}
30	0x1E	RS	62	0x3E	>	94	0x5E	^	126	0x7E	~
31	0x1F	US	63	0x3F	?	95	0x5F	-	127	0x7F	DEL

16.3 MODBUS RTU/TCP

모드버스(modbus)는 1979년 모디콘(Modicon)이라는 회사에서 만든 시리얼 통신 프로토콜입니다. 프로토콜이 단순하지만, 장비 제어와 모니터링에 필요한 기능들을 수행할 수 있기에 사실상의 표준 프로토콜로 현재까지 산업용 전자 장치들을 서로 연결하는 목적으로 널리 사용됩니다. 네트워크 통신 환경의 발전에 따라서 시리얼 통신을 넘어서 TCP/IP 통신용 버전까지 확장되어 왔습니다.

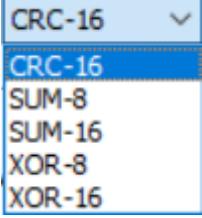
eRun SCADA에서는 통합형 MODBUS 드라이버를 제공합니다.

- ACTIVE, PASSIVE 모드 지원
- RS-232/RS-485/TCP 통신 가능
- RTU/TCP 모드 지원
- 큰 사이즈 데이터를 한 번에 전송 가능 (스크립트)
- 로그기능



■ 통신포트 및 선택사항 설명

항목	설명
MODE	두가지의 연결모드를 지원합니다. 디바이스로 연결을 할 것인지 아니면 디바이스가 eRun으로 연결할 것인지 결정합니다.

	<p>■ ACTIVE : PC에서 상대 디바이스로 연결을 시도합니다. (클라이언트 모드)</p> <p>■ PASSIVE 상대 디바이스가 PC로 연결을 시도합니다. (서버모드)</p>
미디어	통신장치에 따라 RS-232C / RS-485 / TCP 방식이 있습니다.
통신포트	RS-232C, RS-485 방식일 경우 입력이 가능하며, RTU 방식입니다. 포트이름/통신속도/데이터비트/파리티비트/스톱비트 설정합니다.
TCP/IP	미디어가 TCP/IP인 경우 입력이 가능하며 MODBUS-TCP방식으로 통신합니다. ACTIVE 모드인 경우 TCP IP입력이 활성화되며 상대 디바이스 연결주소를 입력합니다.
KEEP ALIVE	
LOG KEEP	LOG COLLECT 옵션이 선택이 되어있어야 하며 통신로그 기록 파일의 보관기간을 입력합니다. 기간이 지난 파일은 삭제합니다. 통신에 장애가 발생할 때 원인 파악 분석을 위하여 남겨놓는 데이터 파일입니다. 일반적으로 3일정도로 설정합니다.
CHECK METHOD	디바이스 제조사마다 MODBUS 통신 프로토콜 체크코드 계산방식을 변형해서 사용하기도 합니다. 그래서 여기서는 5가지 방식을 지원합니다.  <p>표준 방식에서는 CRC-16 계산방식을 주로 사용합니다.</p>
Scan Time(m/s)	우측 통신프레임에 설정된 다수의 프레임간 통신간격을 밀리초 단위로 입력합니다. 0으로 설정하면 프레임간 Interval이 전혀 없이 고속의 데이터 송수신이 가능합니다. 그러나 다수의 드라이버를 사용하는 경우 밸런스를 위해서 최소값은 0이 아닌 1로 설정해주시기 바랍니다.
Timeout (m/s)	프레임 송신을 하고 응답을 기다리는 시간입니다. 이 값을 너무 짧게 하면 정상적으로 송수신이 안될 수 있습니다. 드라이버 디폴트 값을 그대로 사용합니다.
LOG COLLECT	통신 로그데이터를 파일에 기록할 것인지 선택합니다. 로그파일은 해당 프로젝트의 IOserverLog 폴더에 다음과 같은 파일명 규칙으로 텍스트파일 형태로 기록된다. <u>"/Publish/프로젝트/Log/IOserverLog/DRV.MODBUS_2021_1_31.txt"</u>
MODBUS-TCP Mode	모드버스 TCP 방식인 경우 선택합니다. MODBUS-RTU와 MODBUS-TCP의 프로토콜은 유사하지만 동일하지 않으므로 구분할 필요가 있습니다.

	TCP 방식은 BODY부분은 RTU와 동일하지만, CHECK-METHOD가 없고, 6바이트로 구성된 헤더데이터가 앞부분에 들어갑니다.
Reset On Comm. Error	디바이스로부터 통신요구에 대한 응답이 없을 경우 해당 프레임의 결과데이터를 모두 0으로 초기화 할 것인지 선택합니다. 통신상태를 나타내는 프레임은 해당되지 않습니다.
Add (Frame)	통신 명령을 추가합니다. (이것을 프레임 추가한다고 말합니다.)
Insert (Frame)	선택된 프레임을 뒤로 미루고 사이에 프레임을 삽입합니다.
Delete (Frame)	선택한 프레임을 리스트에서 삭제합니다.
Load CSV	다른 MODBUS 통신프레임 설정파일을 불러옵니다.
Save CSV	통신 프레임 설정내용을 CSV 포맷으로 저장합니다.
Run Test	설정된 내용으로 디바이스와 통신시험을 합니다. READ / WRITE 모두 가능합니다.
Stop Test	통신시험을 그만합니다.
Send Test	통신시험중에 디바이스로 특정 데이터를 출력하는 시험을 합니다.

■ MODBUS 통신프레임 설명

항목	설명
Code	프레임의 이름입니다. 실태그 추가할 때 어드레스로 사용됩니다. 어드레스의 선 두번지를 이름으로 보통 사용합니다.
Unit	MODBUS 통신은 254대의 디바이스를 하나의 네트워크에 연결이 가능하기 때문에 각 디바이스의 ID가 필요합니다. 그 ID번호를 입력합니다.
Command	모드버스 명령 레지스터를 선택합니다. 6 가지의 명령이 있습니다. 자세한 설명은 뒤에 나오는 명령어 부분을 참고하시기 바랍니다. ReadBits(1) ReadBits(2) ReadWords(3) ReadWords(4) WriteBit(5) WriteWord(6)
Address	시작 어드레스를 입력합니다. 모드버스 프로토콜을 사용하는 디바이스는 어ドレス 리스트를 기본적으로 제공합니다.
Count	시작 어드레스부터 읽어들일 워드수를 입력합니다.
Scan	호출여부를 선택합니다.
Description	프레임 내용을 설명합니다.

■ Command 설명

명령	설명
ReadBits(1)	<p>Read Coils (FC01) 디지털 출력포트 상태를 읽어옵니다. 포트 개수에 따라 바이트 단위로 추가되며 1 바이트에 8개의 포트를 비트 단위로 나타냅니다. 시작 주소에 해당되는 포트부터 LSB에서 MSB 방향으로 표시되고 비트 값 0은 OFF를 1은 ON을 의미하며 요청되지 않았거나 제품에 없는 포트에 해당되는 비트의 값은 0으로 채워집니다.</p> <p>이 명령은 Read Count가 비트의 개수를 의미합니다. 예를 들어서 Read Count의 값이 16이면 16개의 비트값을 요구하며 결과는 3개의 바이트 데이터값이 수신됩니다. 3개의 바이트 데이터는 3개의 WORD 단위로 변환되어 태그 어드레스 설정시 아래와 같이 설정합니다. 태그형태는 디지털 태그로 설정하고, 첫번째 데이터 어드레스는 CODE:1,0 ~ CODE:1,7 (8개 비트 데이터) 두번째 데이터 어드레스는 CODE:1,0 ~ CODE:2,7 (8개 비트 데이터) 세번째 데이터 어드레스는 CODE:1,0 ~ CODE:3,7 (8개 비트 데이터)</p>
ReadBits(2)	<p>Read Discrete Inputs (FC02) 디지털 입력포트 상태를 나타냅니다. 포트 개수에 따라 바이트 단위로 추가되며 1 바이트에 8개의 포트를 비트 단위로 나타냅니다. 시작 주소에 해당되는 포트부터 LSB에서 MSB 방향으로 표시되고 비트 값 0은 OFF를 1은 ON을 의미하며 요청되지 않았거나 제품에 없는 포트에 해당되는 비트의 값은 0으로 채워집니다.</p> <p>* ReadBit(1)과 사용방법은 동일합니다.</p>
ReadWords(3)	<p>Read Holding Registers (FC03) 디지털/아날로그 포트의 상태를 나타냅니다. 디지털 포트는 레지스터 하나에 16 개의 포트를 비트 단위로 나타냅니다. 시작 주소에 해당되는 포트부터 LSB에서 MSB 방향으로 표시되고 비트 값 0은 OFF를 1은 ON을 의미하며 제품에 없는 포트에 해당되는 비트는 0으로 채워집니다</p>
ReadWords(4)	<p>Read Input Register (FC04) 디지털/아날로그 입력포트의 상태를 나타냅니다. 디지털 포트는 레지스터 하나에 16개의 포트를 비트 단위로 나타냅니다. 시작 주소에 해당되는 포트부터 LSB에서 MSB 방향으로 표시되고 비트 값 0은 OFF를 1은 ON을 의미하며 제품에 없는 포트에 해당되는 비트는 0으로 채워집니다.</p>
WriteBit(5)	<p>Write Single Coil (FC 05) 하나의 디지털 출력포트 ON/OFF 제어에 사용됩니다.</p>
WriteWord(6)	<p>Write Single Register (FC 06) 출력포트 제어에 이용할 값이며 레지스터 하나에 16개의 출력포트를 비트 단위로 나타냅니다. 레지스터 주소에 해당되는 포트부터 LSB에서 MSB 방향으로 적용되고 비트 값 0</p>



제 16 장 디바이스 드라이버

	은 OFF를 1은 ON을 의미하며 출력포트 개수를 넘거나 제품에 없는 포트에 해당되는 비트의 값은 무시됩니다.
--	---



제 16 장 디바이스 드라이버

■ MODBUS RTU 패킷구조

MODBUS-TCP 구조와는 다르게 CRC-16 체크코드가 추가되고 8바이트의 길이 구조입니다.

1) DEVICE 요구포맷

1	2	3	4	5	6	7	8
Slave Address	Function Code	Address Hi	Address Lo	Read Hi	Read Lo	CRC Hi	CRC Lo
0x01	0x04	0x00	0x00	0x00	0x01	0x31	0xCA

2) DEVICE 응답포맷

1	2	3	4	5	6	7
Slave Address	Function Code	Read Byte	Data Hi	Data Lo	CRC Hi	CRC Lo
0x01	0x04	0x02	0x00	0x0A	0x39	0x37

■ MODBUS TCP 패킷구조

RTU와 다르게 6바이트의 헤더패킷이 추가되고, CRC 체크코드 2바이트가 생략됩니다.

1) DEVICE 요구포맷

1	2	3	4	5	6	7	8	9	10	11	12
Transaction ID Hi	Transaction ID Lo	Protocol ID Hi	Protocol ID Lo	Unit Identifier	Length	Slave Address	Function Code	Address Hi	Address Lo	Read Hi	Read Lo
0x00	0x01	0x00	0x00	0x00	0x06	0x01	0x04	0x00	0x00	0x00	0x01

2) DEVICE 응답포맷

1	2	3	4	5	6	7	8	9	10	11
Transaction ID Hi	Transaction ID Lo	Protocol ID Hi	Protocol ID Lo	Unit Identifier	Length	Slave Address	Function Code	Read Byte	Read Hi	Read Lo
0x00	0x01	0x00	0x00	0x00	0x05	0x01	0x04	0x02	0xaa	0xbb

■ MODBUS 드라이버 사용예

1) 1번 프레임 추가

No	Code	Unit	Command	Address	Count
1	40101	1	ReadWords(4)	100	30

위와 같이 추가한 프레임에 대한 설명입니다.

항목	설명	비고
Code	문자열형식으로 태그에서 참조하는 어드레스	보통 Address와 동일한 이름을 주면 인식이 편합니다.
Unit	상대 디바이스의 국번	디바이스의 설정에 따름.
Command	통신명령 또는 방향	읽기/쓰기 명령
Address	디바이스 어드레스	모드버스를 지원하는 디바이스 매뉴얼 참조
Count	읽을 워드 개수	1 = 1*2바이트-> 2바이트 5 = 5*2바이트-> 10바이트

2) 통신 결과 (로그파일 참조)

5개의 WORD값을 요구했고, 수신된 데이터는 아래와 같이 5개 워드(10바이트)가 됩니다.

[40100] Tx :(1) 01, 04, 00, 64, 00, 05, 71, D6,

[40100] Rx :(1) 01 04 0A 00 0A 00 0B 00 0C 00 0D 00 0E

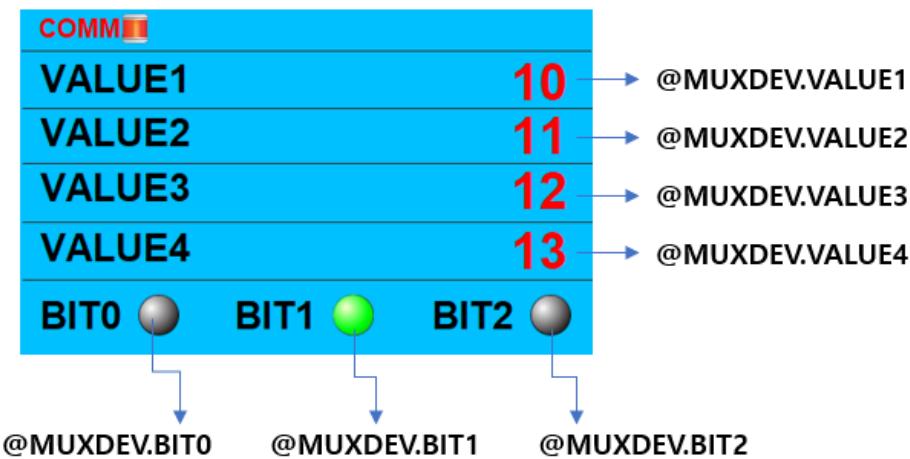
3) 태그생성 및 디바이스 어드레스 설정

HDRMS:MUX4 HDRMS:DATA HDRMS:MUXDEV						
번호	이름	태그종류	디바이스	어드레스	가상	설명
1	COMM	아날로그	MUX20	40100:0	<input type="checkbox"/>	통신상태
2	VALUE1	아날로그	MUX20	40100:1 ①	<input type="checkbox"/>	1번 WORD
3	VALUE2	아날로그	MUX20	40100:2 ②	<input type="checkbox"/>	2번 WORD
4	VALUE3	아날로그	MUX20	40100:3 ③	<input type="checkbox"/>	3번 WORD
5	VALUE4	아날로그	MUX20	40100:4 ④	<input type="checkbox"/>	4번 WORD
6	BIT0	아날로그	MUX20	40100:5,0	<input type="checkbox"/>	5번 첫번째 On/Off
7	BIT1	아날로그	MUX20	40100:5,1 ⑤	<input type="checkbox"/>	5번 두번째 On/Off
8	BIT2	아날로그	MUX20	40100:5,2	<input type="checkbox"/>	5번 세번째 On/Off

4) 뷰페이지 문자열 오브젝트 추가



5) 실행화면



6) 태그 모니터링 화면

태그상태 모니터링 [그룹 : MUXDEV] - Total Tags : 705			
태그명	태그값	디바이...	어드...
COMM	1 (0x0001)	MUX20	40100:0 통신상태
VALUE1	10 (0x000A)	MUX20	40100:1 1번 WORD
VALUE2	11 (0x000B)	MUX20	40100:2 2번 WORD
VALUE3	12 (0x000C)	MUX20	40100:3 3번 WORD
VALUE4	13 (0x000D)	MUX20	40100:4 4번 WORD
BIT0	0 (0x0000)	MUX20	40100... 5번 첫번째 On/Off
BIT1	1 (0x0001)	MUX20	40100... 5번 두번째 On/Off
BIT2	0 (0x0000)	MUX20	40100... 5번 세번째 On/Off

■ 스크립트 함수호출로 다수개의 데이터 송출

연속적으로 보내야 하는 자료를 한 번에 보낼 때 사용합니다. 다량의 데이터를 태그를 통해서 보내면 전체적인 통신에 영향을 주게 됩니다. 그래서 여러 번에 걸쳐서 보내내는 것을 스크립트 함수를 통하여 한 번에 내보낼 수 있습니다. 모드버스 드라이버는 이 함수를 지원합니다.

_DeviceWriteString("디바이스명", "프레임명", string);

문자열(string)을 태그통하지 않고 디바이스로 직접 내보낸다. 최대 글자수는 4000바이트.

_DeviceWriteBytes("device name", "frame code", offset, bytes_data[], bytesSize);

바이트 데이터를 디바이스로 직접 내보낸다. 4000바이트 이내.

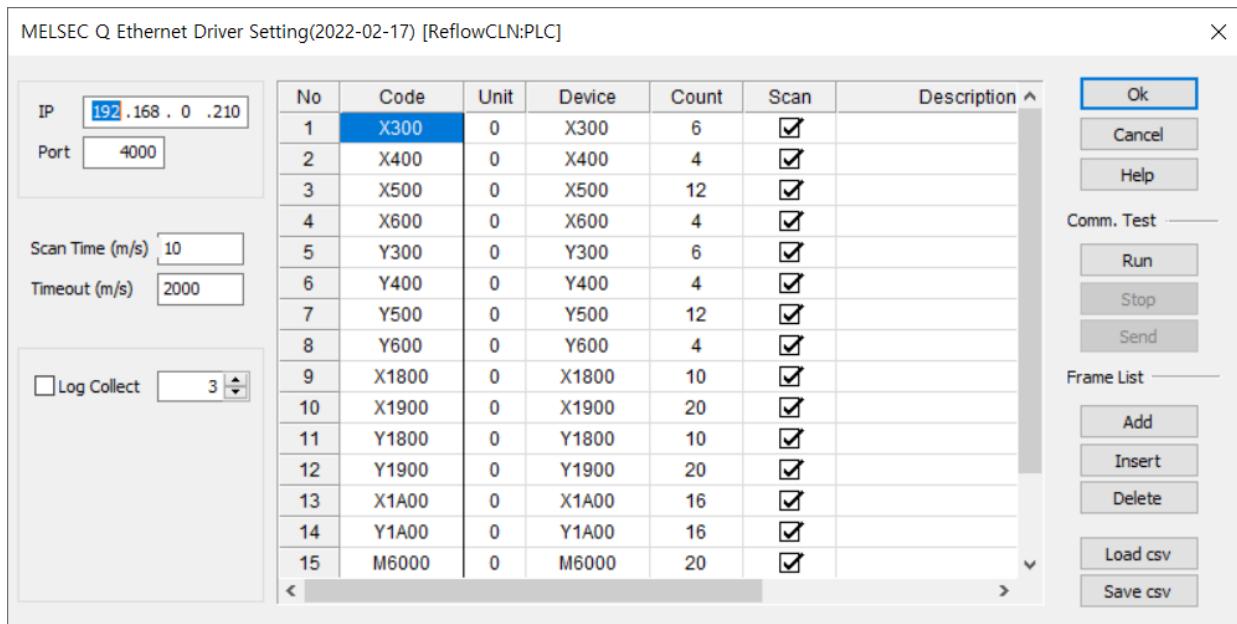
_DeviceWriteWords("device name", "frame code", offset, word_data[], bytesSize);

워드 데이터를 디바이스로 직접 내보낸다. 4000바이트 이내.

16.4 MELSEC-Q Ethernet

MITSUBISHI PLC의 Q 대응 MELSEC Ethernet 모듈 인터페이스 통신 드라이버입니다

- ACTIVE 모드동작
- MC-PROTOCOL TCP 통신전용
- 큰 사이즈 데이터를 한 번에 전송 가능 (스크립트)
- 로그기능



■ 통신포트 및 선택사항 설정

항목	설명
TCP/IP	PLC 네트워크에 설정된 IP주소를 입력합니다.
PORT	PLC 네트워크에 설정된 TCP 포트번호를 입력합니다.
Scan time	프레임간 시간간격을 설정합니다.
Timeout	PC에서 호출하고 응답에 대한 초과시간을 설정합니다.
LOG COLLECT	통신 로그데이터를 파일에 기록할 것인지 선택합니다. 로그파일은 해당 프로젝트의 IOserverLog 폴더에 다음과 같은 파일명 규칙으로 텍스트파일 형태로 기록된다. <u>"/Publish/프로젝트/Log/IOserverLog/DRV.MS001_2021_1_31.txt"</u>
Add (Frame)	통신 명령을 추가합니다. (이것을 프레임 추가한다고 말합니다.)
Insert (Frame)	선택된 프레임을 뒤로 미루고 사이에 프레임을 삽입합니다.
Delete (Frame)	선택한 프레임을 리스트에서 삭제합니다.

Run Test	설정된 내용으로 디바이스와 통신시험을 합니다. READ / WRITE 모두 가능합니다.
Stop Test	통신시험을 그만합니다.
Send Test	통신시험중에 디바이스로 특정 데이터를 출력하는 시험을 합니다.
Load CSV	디바이스 설정정보가 저장된 CSV 파일을 불러옵니다.
Save CSV	현재 설정된 디바이스 정보를 CSV로 저장합니다.

■ 통신프레임 설명

항목	설명
Code	프레임의 이름입니다. 실태그 추가할 때 어드레스로 사용됩니다. 어드레스의 선두 번지를 이름으로 보통 사용합니다.
Unit	PLC 국번을 설정해줍니다. PLC국번은 MELSEC Config에서 설정된 국번을 입력해 주어야 합니다.
Device	MELSEC PLC에서 지원하는 디바이스명 어드레스를 붙여서 설정해줍니다. 사용하는 레지스터는 X,Y,B,W,D,M,L,SB,SW,ZR 등입니다.
Count	시작 어드레스부터 읽어들일 워드수를 입력합니다.
Scan	호출여부를 선택합니다.
Description	프레임 내용을 설명합니다.

■ [RunTest] 통신상태

[로그파일 내용]

[17:49:49.683] TO : 500000FF03FF000018000A04010000ZR0000000032

[17:49:49.692] FROM:

[17:49:49.732] TO : 500000FF03FF000018000A04010000D*0090000064

[17:49:49.741] FROM:

■ 스크립트 함수호출로 다수개의 데이터 송출

연속적으로 보내야 하는 자료를 한 번에 보낼 때 사용합니다. 다양한 데이터를 태그를 통해서 보내면 전체적인 통신에 영향을 주게 됩니다. 그래서 여러 번에 걸쳐서 내보내는 것을 스크립트 함수를 통하여 한 번에 내보낼 수 있습니다. MELSECQEtn 드라이버는 이 함수를 지원합니다.

```
_DeviceWriteString("디바이스명", "프레임명", string);
```

문자열(string)을 태그를 통하지 않고 디바이스로 직접 내보낸다. 최대 글자수는 4000바이트.

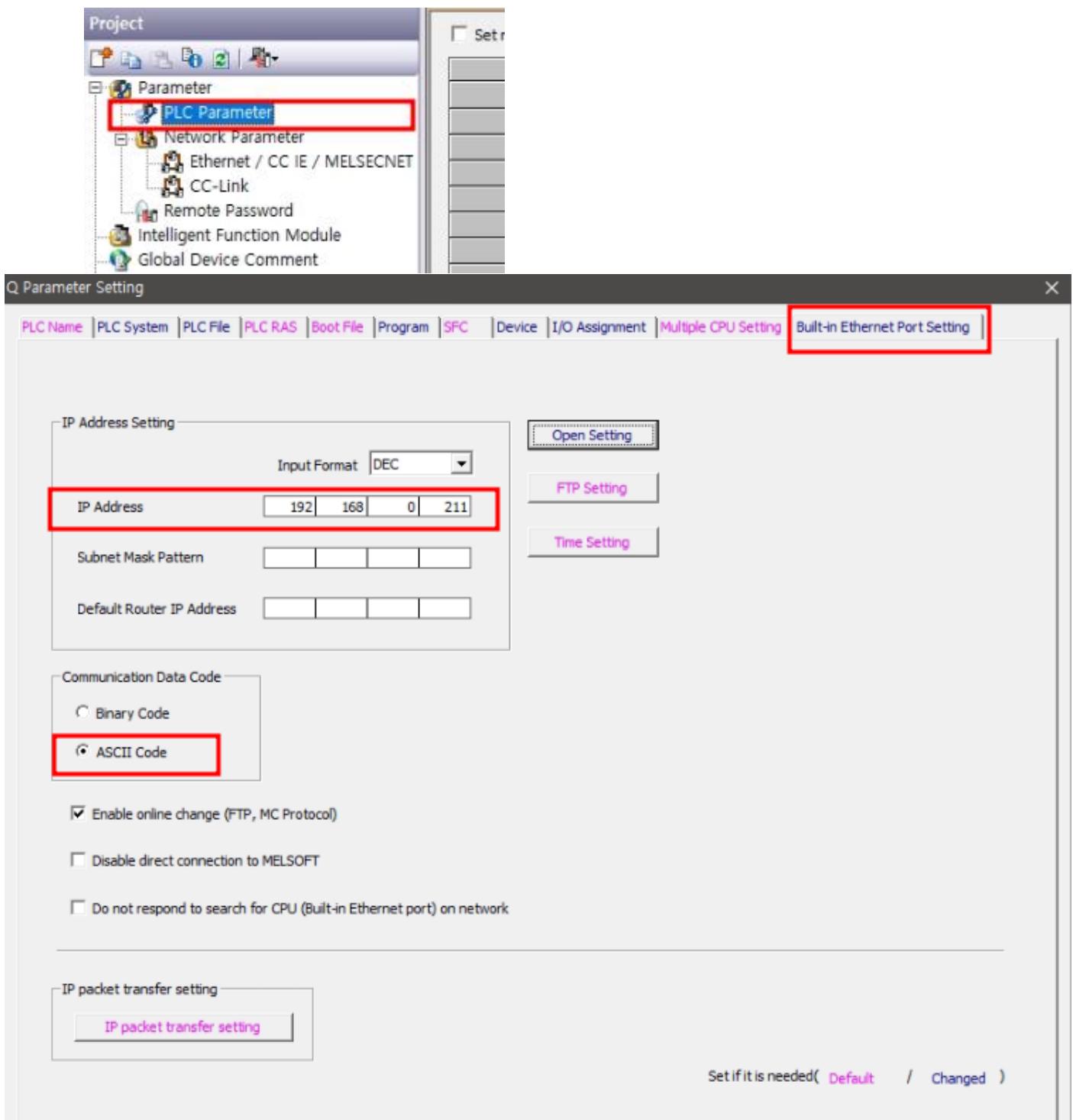
```
_DeviceWriteBytes("device name", "frame code", offset, bytes_data[], bytesSize);
```

바이트 데이터를 디바이스로 직접 내보낸다. 4000바이트 이내.

```
DeviceWriteBytes("device name", "frame code", offset, word data[], bytesSize);
```

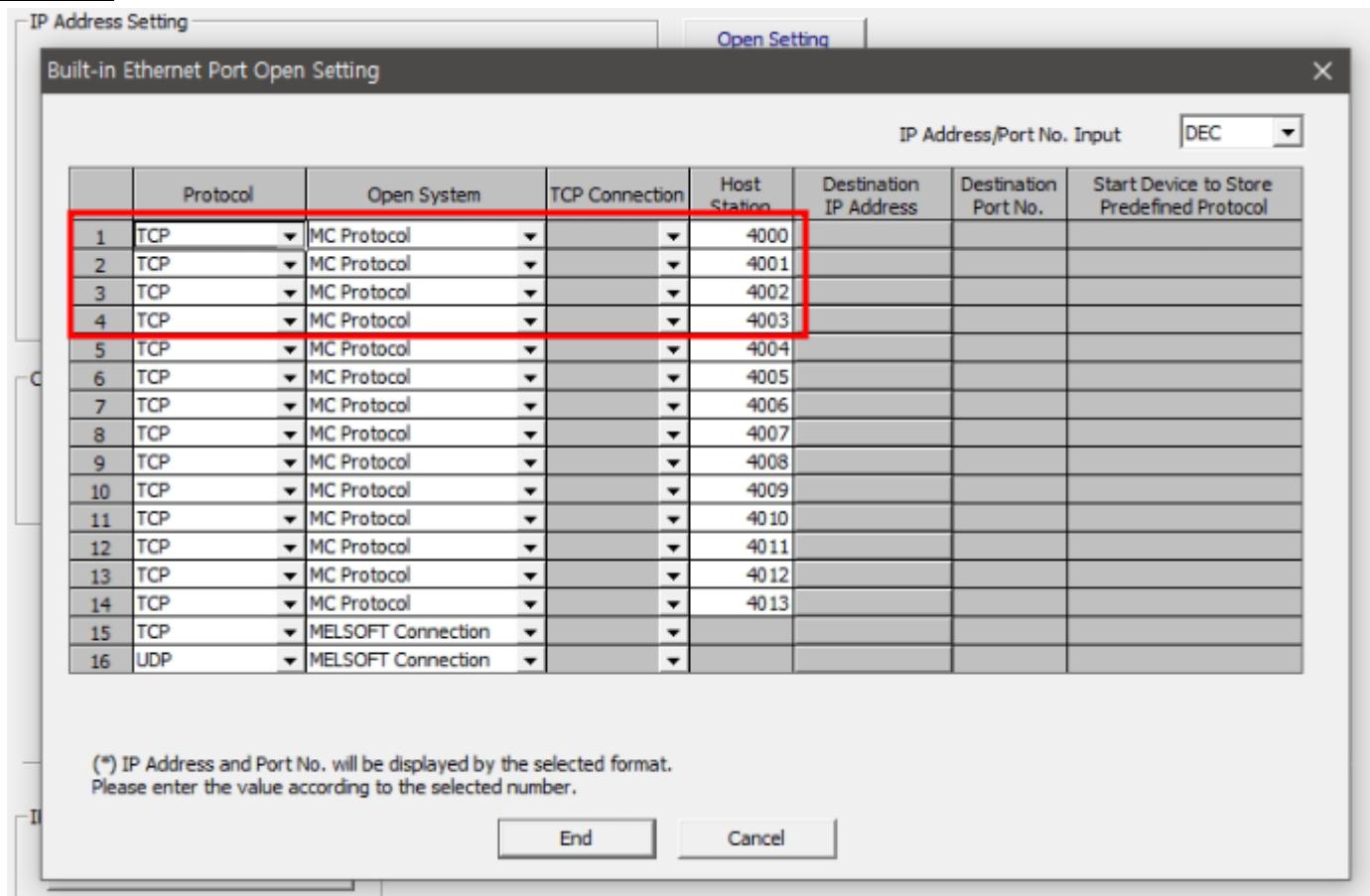
워드 데이터를 디바이스로 직접 내보낸다. 4000바이트 이내.

- MELSEC-Q PLC 네트워크 설정 (Built-in Ethernet Setting)
MELSEC-Q 시리즈 Ethernet 연결 방법은 2가지의 경우가 있습니다.
우선 CPU 포트(Built-in)의 Ethernet 연결설정은 아래 그림과 같이 “P



위의 그림과 같이 적색 박스부분을 설정해 줍니다.

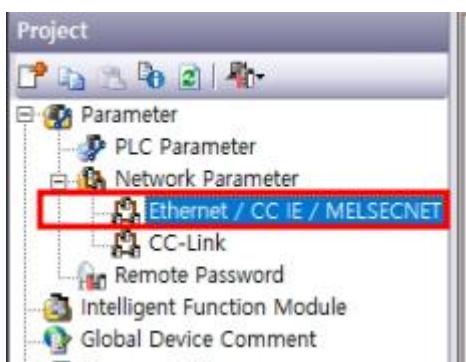
우측에 있는 "Open Setting" 버튼을 클릭해서 프로토콜을 설정합니다.



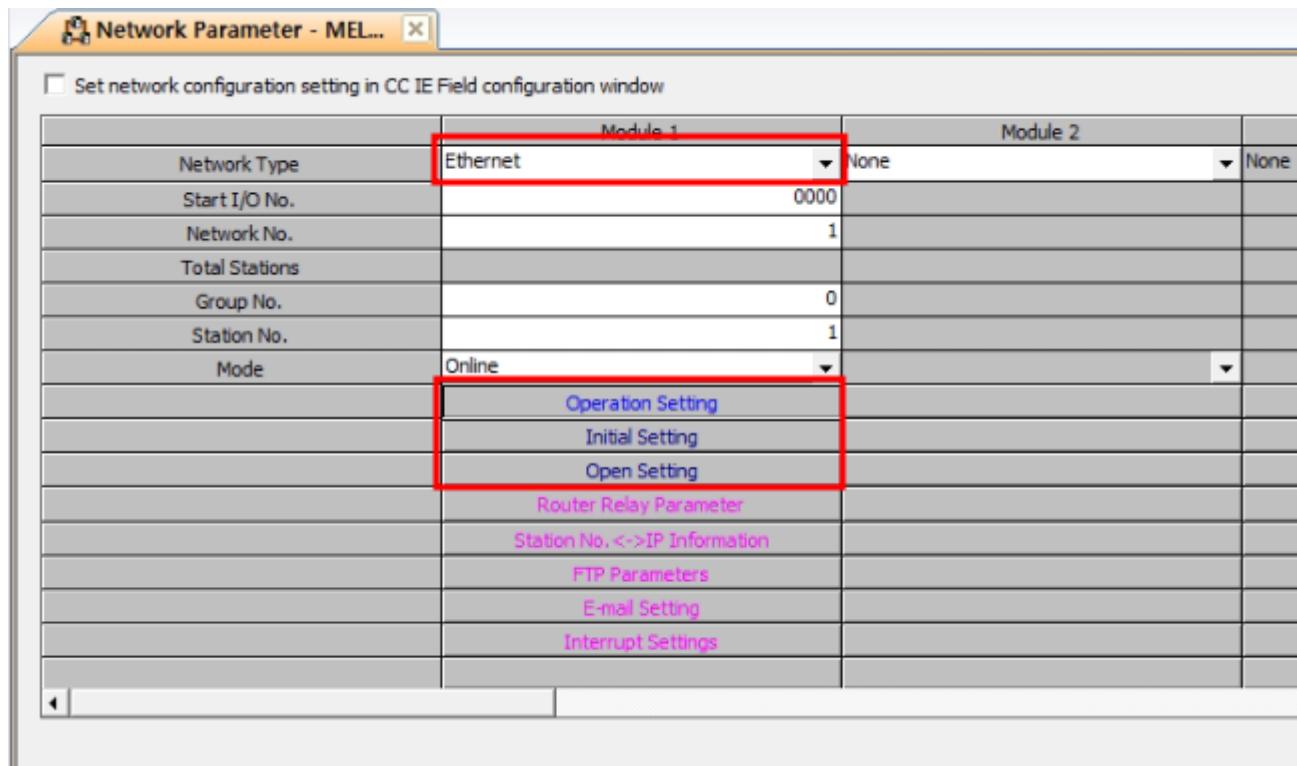
적색 박스부분과 같이 TCP – MC Protocol – 4000 설정해줍니다.

최대 14개의 서로다른 포트를 설정해서 원활한 통신을 하도록 적당히 분배합니다.

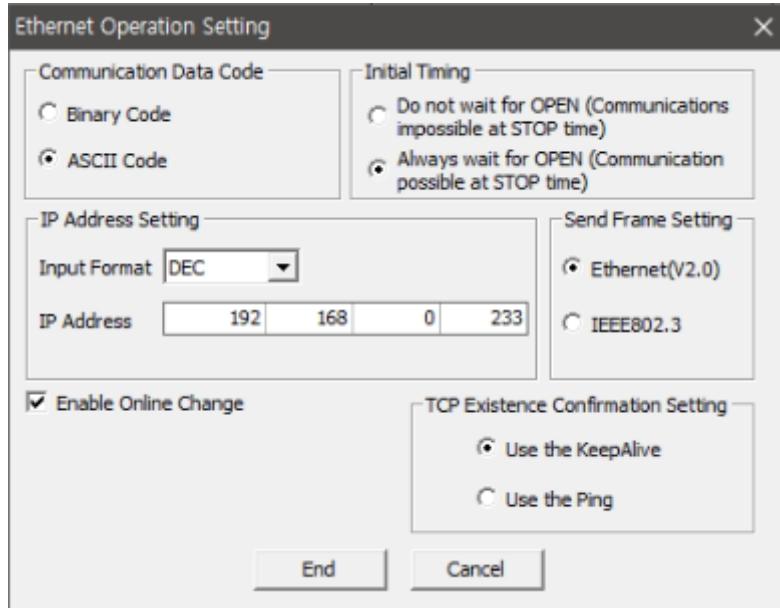
- Network Parameter (확장 Ethernet card 설정)
별도의 Ethernet Unit에 연결할 경우는 아래의 순서로 설정해 줍니다.



그림과 같이 Network Parameter 의 Ethernet / CCIE / MELSECNET 메뉴를 클릭합니다.



1) Operation Setting 버튼을 선택하면 아래와 같은 설정화면이 나오고 동일하게 설정해줍니다.



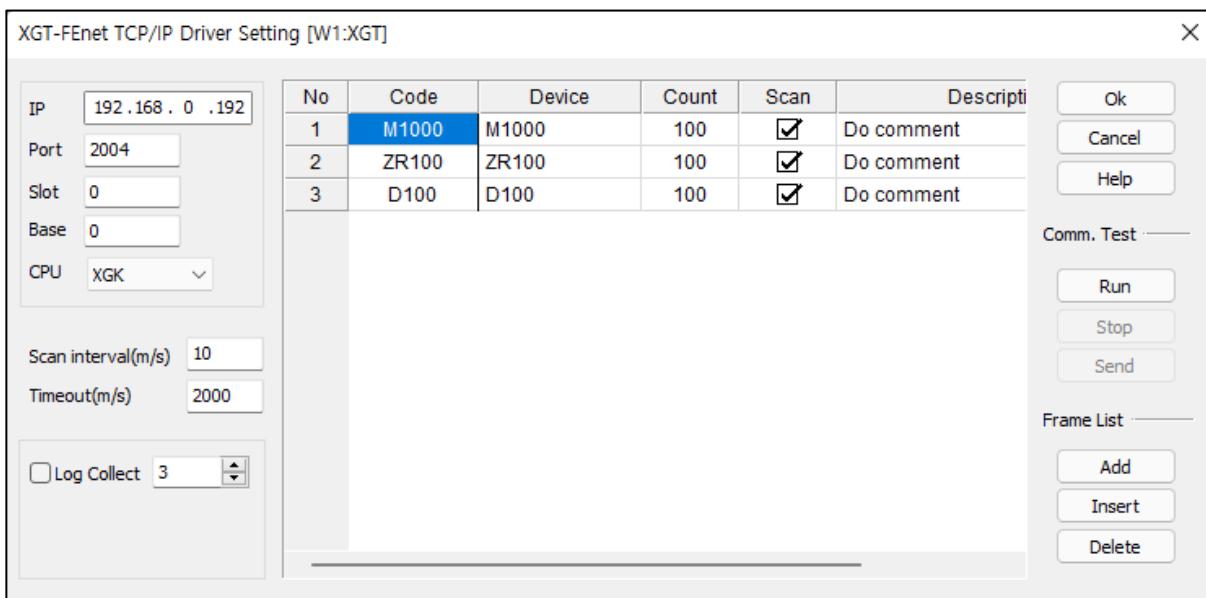
2) Open Setting 버튼을 선택해서 아래와 같이 창이 나타나면 적색부분과 같이 설정합니다.

	Protocol	Open System	Fixed Buffer	Fixed Buffer Communication	Pairing Open	Existence Confirmation	Host Station Port No.	Destination IP Address	Destination Port No.	IP Address/Port No. Input Format	DEC
1	UDP	▼	▼ Send	▼ Procedure Exist	▼ Disable	▼ No Confirm	1025	Broadcast Together	65535		
2	TCP	▼ Unpassive	▼ Send	▼ Procedure Exist	▼ Disable	▼ Confirm	1026				
3	TCP	▼ Fullpassive	▼ Send	▼ Procedure Exist	▼ Disable	▼ Confirm	1027	192.168.0.1	1027		
4	TCP	▼ Unpassive	▼ Send	▼ Procedure Exist	▼ Disable	▼ No Confirm	5000				
5	▼	▼	▼	▼	▼	▼					
6	▼	▼	▼	▼	▼	▼					
7	▼	▼	▼	▼	▼	▼					
8	▼	▼	▼	▼	▼	▼					
9	▼	▼	▼	▼	▼	▼					
10	▼	▼	▼	▼	▼	▼					
11	▼	▼	▼	▼	▼	▼					
12	▼	▼	▼	▼	▼	▼					
13	▼	▼	▼	▼	▼	▼					
14	▼	▼	▼	▼	▼	▼					
15	▼	▼	▼	▼	▼	▼					
16	▼	▼	▼	▼	▼	▼					

16.5 XGTFEnet

LS XGT PLC Ethernet 모듈 인터페이스 통신 드라이버입니다

- ACTIVE 모드동작
- LS XGK, XGI TCP 통신전용
- 큰 사이즈 데이터를 한 번에 전송 가능 (스크립트)
- 로그기능



■ 통신포트 및 선택사항 설명

항목	설명
TCP/IP	PLC 네트워크에 설정된 IP주소를 입력합니다.
PORT	PLC 네트워크에 설정된 TCP 포트번호를 입력합니다. (기본 2004)
Slot	ETHERNET I/F 모듈의 슬롯 번호. (Default 0)
Base	ETHERNET I/F 모듈의 베이스 번호 (Default 0)
CPU	PLC CPU에 따라서 지원되는 디바이스와 어드레스 체계가 조금 다릅니다. XGK(0), XGI(1)
Scan Interval (m/s)	프레임간 통신간격을 입력합니다. (Default : 10ms) 이값이 크면 데이터 갱신이 늦어집니다.
Timeout (m/s)	프레임에 대한 호출응답을 기다리는 시간입니다. Scan Interval 값보다는 커야 합니다.
LOG COLLECT	LOG COLLECT 옵션이 선택이 되어있어야 하며 통신로그 기록 파일의 보관기간을 입력합니다. 기간이 지난 파일은 삭제합니다. 통신에 장애가 발생할 때 원인파악

	분석을 위하여 남겨놓는 데이터 파일입니다. 일반적으로 3일정도로 설정합니다. 통신 로그데이터를 파일에 기록할 것인지 선택합니다. 로그파일은 해당 프로젝트의 IOserverLog 폴더에 다음과 같은 파일명 규칙으로 텍스트파일 형태로 기록됩니다. <u>"Publish/프로젝트/Log/IOserverLog/DRV.MS001_2021_1_31.txt"</u>
Add Frame	통신 명령을 추가합니다. (이것을 프레임 추가한다고 말합니다.)
Insert Frame	선택된 프레임을 뒤로 미루고 사이에 프레임을 삽입합니다.
Delete Frame	선택한 프레임을 리스트에서 삭제합니다.
Run Test	설정된 내용으로 디바이스와 통신시험을 합니다. READ / WRITE 모두 가능합니다.
Stop Test	통신시험을 그만합니다.
Send Test	통신시험중에 디바이스로 특정 데이터를 출력하는 시험을 합니다.

■ 통신프레임 설명

항목	설명
Code	프레임의 이름입니다. 실태그 추가할 때 어드레스로 사용됩니다. 어드레스의 선두 번지를 이름으로 보통 사용합니다.
Device	XGT PLC 에서 지원하는 디바이스명 어드레스를 붙여서 설정해줍니다. 예를 들어서, M0, M100, D0, D100, P0, P100, ZR0, ZR100, ... 연결하는 PLC 모델이 XGK 계열인, XGI 계열인 경우에 사용가능한 레지스터 디바이스가 조금 다릅니다. XGK 모델 지원 디바이스 : P, M, L, K, F, T, C, D, S, ZR 등 XGI 모델 지원 디바이스 : M, D, 등입니다.
Count	시작 어드레스부터 읽어들일 워드수를 입력합니다.
Scan	호출여부를 선택합니다.
Description	프레임 내용을 설명합니다.

■ [RunTest] 통신상태



제 16 장 디바이스 드라이버

Device Name	Driver File	Version	Setting	Status
XGTFENET	XGTFENET.dll	3.0.1.58	192.168.0.177 : 2004	OK
<IN> 55 00 14 00 00 00 00 00 01 00 04 00 00 00 00 00				
<OUT> 4C 53 49 53 2D 58 47 54 00 00 00 A0 33 B2 00 10 00 00 F0 54 00 14 00 00 00 01 00 04 00 25 4D 42 30 04 00				
<IN> 55 00 14 00 00 00 00 00 01 00 04 00 00 00 00 00				
<OUT> 4C 53 49 53 2D 58 47 54 00 00 00 A0 33 B3 00 10 00 00 F1 54 00 14 00 00 00 01 00 04 00 25 4D 42 30 04 00				
<IN> 55 00 14 00 00 00 00 00 01 00 04 00 00 00 00 00				
<OUT> 4C 53 49 53 2D 58 47 54 00 00 00 A0 33 B4 00 10 00 00 F2 54 00 14 00 00 00 01 00 04 00 25 4D 42 30 04 00				
<IN> 55 00 14 00 00 00 00 00 01 00 04 00 00 00 00 00				
<OUT> 4C 53 49 53 2D 58 47 54 00 00 00 A0 33 B5 00 10 00 00 F3 54 00 14 00 00 00 01 00 04 00 25 4D 42 30 04 00				
<IN> 55 00 14 00 00 00 00 00 01 00 04 00 00 00 00 00				
<OUT> 4C 53 49 53 2D 58 47 54 00 00 00 A0 33 B6 00 10 00 00 F4 54 00 14 00 00 00 01 00 04 00 25 4D 42 30 04 00				
<IN> 55 00 14 00 00 00 00 00 01 00 04 00 00 00 00 00				
<OUT> 4C 53 49 53 2D 58 47 54 00 00 00 A0 33 B7 00 10 00 00 F5 54 00 14 00 00 00 01 00 04 00 25 4D 42 30 04 00				
<IN> 55 00 14 00 00 00 00 00 01 00 04 00 00 00 00 00				
<OUT> 4C 53 49 53 2D 58 47 54 00 00 00 A0 33 B8 00 10 00 00 F6 54 00 14 00 00 00 01 00 04 00 25 4D 42 30 04 00				
<IN> 55 00 14 00 00 00 00 00 01 00 04 00 00 00 00 00				
<OUT> 4C 53 49 53 2D 58 47 54 00 00 00 A0 33 B9 00 10 00 00 F7 54 00 14 00 00 00 01 00 04 00 25 4D 42 30 04 00				
<IN> 55 00 14 00 00 00 00 00 00 01 00 04 00 00 00 00 00				
<OUT> 4C 53 49 53 2D 58 47 54 00 00 00 A0 33 BA 00 10 00 00 F8 54 00 14 00 00 00 01 00 04 00 25 4D 42 30 04 00				
<IN> 55 00 14 00 00 00 00 00 00 01 00 04 00 00 00 00 00				
<OUT> 4C 53 49 53 2D 58 47 54 00 00 00 A0 33 BB 00 10 00 00 F9 54 00 14 00 00 00 01 00 04 00 25 4D 42 30 04 00				
<IN> 55 00 14 00 00 00 00 00 00 01 00 04 00 00 00 00 00				
<OUT> 4C 53 49 53 2D 58 47 54 00 00 00 A0 33 BC 00 10 00 00 FA 54 00 14 00 00 00 01 00 04 00 25 4D 42 30 04 00				
<IN> 55 00 14 00 00 00 00 00 00 01 00 04 00 00 00 00 00				
<OUT> 4C 53 49 53 2D 58 47 54 00 00 00 A0 33 BD 00 10 00 00 FB 54 00 14 00 00 00 01 00 04 00 25 4D 42 30 04 00				
<IN> 55 00 14 00 00 00 00 00 00 01 00 04 00 00 00 00 00				
<OUT> 4C 53 49 53 2D 58 47 54 00 00 00 A0 33 BE 00 10 00 00 FC 54 00 14 00 00 00 01 00 04 00 25 4D 42 30 04 00				
<IN> 55 00 14 00 00 00 00 00 00 00 01 00 04 00 00 00 00				
<OUT> 4C 53 49 53 2D 58 47 54 00 00 00 A0 33 BF 00 10 00 00 FD 54 00 14 00 00 00 01 00 04 00 25 4D 42 30 04 00				
<IN> 55 00 14 00 00 00 00 00 00 00 00 01 00 04 00 00 00 00				

■ 로그파일 내용

[READ 내용]

[17:49:49.683] TO : 500000FF03FF000018000A04010000ZR0000000032

[17:49:49.692] FROM:

[17:49:49.732] TO : 500000FF03FF000018000A04010000D*0090000064

[17:49:49.741] FROM:

[WRITE 이후 수신된 데이터 내용]

[WORD 값 100 을 M0 에 출력]

[13:43:33.401] TO : LSIS-XGT 00 00 00 00 a0 33 78 00 12 00 00 b8 58 00 02 00 00 00 01 00 04 00 25 4d 57
30 02 00 64 00

[13:43:33.403] FROM : 59 00 02 00 00 00 00 00 01 00

[M0 Read]

[13:43:33.909] TO : LSIS-XGT 00 00 00 00 a0 33 79 00 10 00 00 b7 54 00 14 00 00 00 01 00 04 00 25 4d 42
30 02 00

[13:43:33.912] FROM : 55 00 14 00 00 00 00 01 00 02 00 **64 00**

■ 스크립트 함수호출로 다수개의 데이터 송출

연속적으로 보내야 하는 자료를 한 번에 보낼 때 사용합니다. 다량의 데이터를 태그를 통해서 보내면 전체적인 통신에 영향을 주게 됩니다. 그래서 여러 번에 걸쳐서 보내내는 것을 스크립트 함수를 통하여 한 번에 보내낼 수 있습니다. XGT-FEnet 드라이버는 이 함수를 지원합니다.

_DeviceWriteString("디바이스명", "프레임명", string);

문자열(string)을 태그를 통하지 않고 디바이스로 직접 보내내며, 최대 글자수는 1400바이트입니다.

No	Code	Device	Count	Scan	Description
1	M1000	M1000	100	<input checked="" type="checkbox"/>	Do comment
2	ZR100	ZR100	100	<input checked="" type="checkbox"/>	Do comment
3	D100	D100	100	<input checked="" type="checkbox"/>	Do comment

```
void DeviceWriteString()
{
    string str;

    str = "eRun";
    _DeviceWriteString("XGT", "D100", str);
}
```

위 함수 실행한 후 문자열 태그로 수신된 상태를 보여줍니다.

▲▶ A13	0 (0x0000)	XGT	ZR100:13
▲▶ A14	0 (0x0000)	XGT	ZR100:14
▲▶ A15	0 (0x0000)	XGT	ZR100:15
■■▶ STRING_DATA	eRun	XGT	D100:1

_DeviceWriteBytes("device name", "frame code", offset, word_data[], bytesSize);

워드 데이터를 디바이스로 직접 내보낸다. 1400바이트 이내. (최대 700개의 WORD)

No	Code	Device	Count	Scan	Description
1	M1000	M1000	100	<input checked="" type="checkbox"/>	Do comment
2	ZR100	ZR100	100	<input checked="" type="checkbox"/>	Do comment
3	D100	D100	100	<input checked="" type="checkbox"/>	Do comment

다음은 예제 스크립트로 5개의 WORD데이터를 한 번에 쓰기.

```
void DeviceWriteBytes()
{
    short bData[5]; // 2바이트 데이터형의 배열 5개를 선언. 총 10바이트 할당.

    bData[0] = @TAG1.VAL1;
    bData[1] = @TAG1.VAL2;
    bData[2] = @TAG1.VAL3;
    bData[3] = 40000;
    bData[4] = 50000;

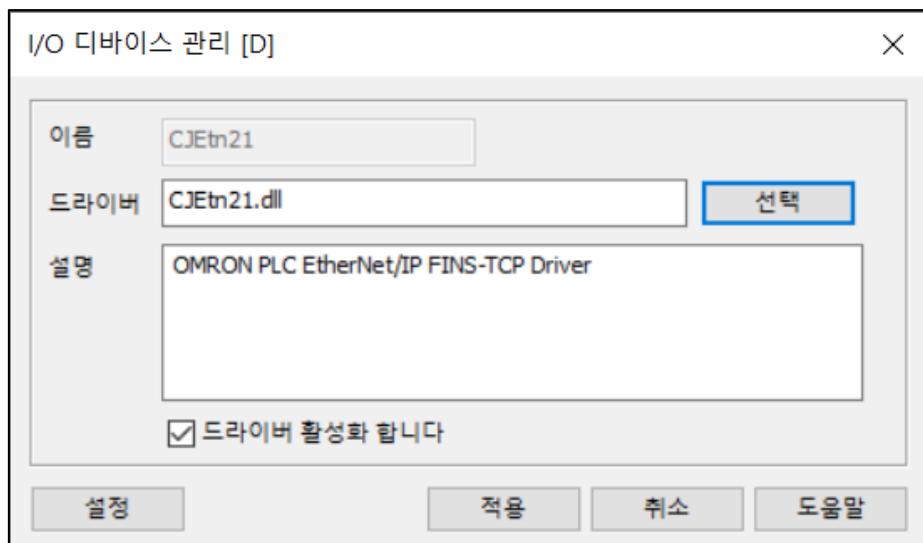
    // Device name : XGT
    // FRAME : ZR100
    // OFFSET : 0
    // 데이터 배열변수명 : bData
    // 바이트단위의 길이
    _DeviceWriteBytes("XGT", "ZR100", 0, bData, 10);
}
```

16.6 OMRON PLC – EtherNet/IP

OMRON PLC의 EtherNet/IP 프로토콜 지원 유니트 통신 인터페이스 드라이버입니다. OMRON에서 제공하는 FINS/TCP 프로토콜을 사용합니다.

- FINS/TCP 프로토콜 전용
- EIP21 유니트
- NJ CPU 유니트 내장 EtherNet/IP
- 큰 사이즈 데이터를 한 번에 전송 가능 (스크립트 이용)
- FINS/TCP는 동시접속 16개까지 접속가능합니다.
- 로그기능

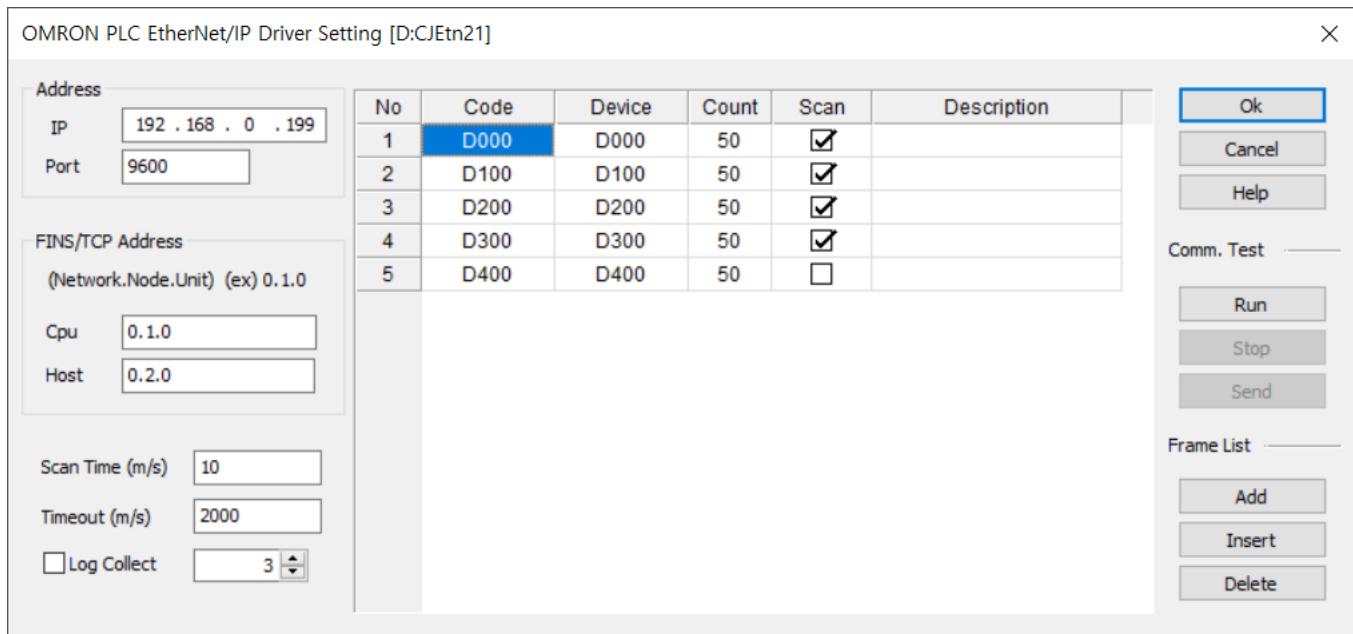
DLL 파일은 CJEtn21.DLL 입니다. 선택버튼을 눌러서 드라이버를 선택합니다.



항목	설명
이름	태그에서 사용되는 디바이스 이름을 입력합니다. 영문 또는 숫자로 구성합니다.
드라이버	선택버튼을 눌러서 드라이버 설정합니다. (CJEtn21.dll)
드라이버 활성화	<p>프로젝트 실행할 때 I/O서버에서 구동하기 위해서는 활성화를 합니다.</p> <p>프로젝트 관리자 I/O 디바이스 팝업메뉴의 "디바이스 활성화"에서 설정해도 동일합니다.</p>

설정	통신속성 및 프레임을 설정합니다.

■ 설정



■ 통신포트 및 선택사항 설명

항목	설명
Address	이 드라이버는 FINS/TCP 프로토콜 통신방식입니다. OMRON NETWORK 카드에 설정된 Ethernet IP주소를 입력합니다.
Port	FINS/TCP는 기본적으로 포트번호가 9600 입니다.
FINS/TCP Address	FINS 노드테이블 값을 서버측(CPU), 클라이언트측(Host) 구분해서 설정합니다. 이 드라이버는 자동으로 노드번호를 부여받기 때문에 수정할 필요없이 기본값으로 두면 됩니다. CPU : 0.1.0 HOST : 0.2.0
Scan Time(m/s)	우측 통신프레임에 설정된 다수의 프레임간 통신간격을 밀리초 단위로 입력합니다. 0 으로 설정하면 프레임간 Interval 이 전혀 없이 고속의 데이터 송수신이 가능합니다. 그러나 다수의 드라이버를 사용하는 경우 밸런스를 위해서 최소값은 0 이 아닌 1 로 설정해주시기 바랍니다.
Timeout (m/s)	프레임 송신을 하고 응답을 기다리는 시간입니다. 이 값을 너무 짧게 하면 정상적으로 송수신이 안될 수 있습니다. 드라이버 디폴트 값을 그대로 사용합니다.



제 16 장 디바이스 드라이버

Log Collect	통신로그 파일을 남기려면 체크합니다. 로그파일의 위치는 해당 프로젝트의 IOServerLog 폴더에 다음과 같은 파일명 규칙으로 텍스트파일 형태로 기록된다. <u>"/Publish/프로젝트/Log/IOServerLog/DRV.Device_2021_1_31.txt"</u>
Add (Frame)	통신 명령을 추가합니다. (이것을 프레임 추가한다고 말합니다.)
Insert (Frame)	선택된 프레임을 뒤로 미루고 사이에 프레임을 삽입합니다.
Delete (Frame)	선택한 프레임을 리스트에서 삭제합니다.
Run Test	설정된 내용으로 디바이스와 통신시험을 합니다. READ / WRITE 모두 가능합니다.
Stop Test	통신시험을 그만합니다.
Send Test	통신시험 중에 디바이스로 특정 데이터를 출력하는 시험을 합니다.

■ 통신프레임 설명

항목	설명
Code	프레임의 이름입니다. 실태그 추가할 때 어드레스로 사용됩니다. 어드레스의 선두번지를 이름으로 보통 사용합니다.
Device	OMRON PLC에서 지원하는 디바이스명 어드레스를 붙여서 설정해줍니다. 이 드라이버는 D 레지스터를 사용합니다.
Count	시작 어드레스부터 읽어들일 워드수를 입력합니다.
Scan	호출여부를 선택합니다.
Description	프레임 내용을 설명합니다.

■ [RunTest] 통신상태 보기

[로그파일 내용]

■ 스크립트 함수호출로 다수개의 데이터 송출

연속적으로 보내야 하는 자료를 한 번에 보낼 때 사용합니다. 다량의 데이터를 태그를 통해서 보내면 전체적인 통신에 영향을 주게 됩니다. 그래서 여러 번에 걸쳐서 내보내는 것을 스크립트 함수를 통하여 한 번에 내보낼 수 있습니다. OMRON PLC FINS/TCP 드라이버는 이 함수를 지원합니다.

[문자열]

```
_DeviceWriteString("디바이스명", "프레임명", string);
```

문자열(string)을 태그를 통하지 않고 디바이스로 직접 내보낸다. 최대 글자수는 4000바이트.

[바이트 데이터]

```
_DeviceWriteBytes("device name", "frame code", offset, bytes_data[], bytesSize);
```

바이트 데이터를 디바이스로 직접 내보낸다. 4000바이트 이내.

[워드데이터]

```
_DeviceWriteBytes("device name", "frame code", offset, word data[], bytesSize);
```

워드 데이터를 디바이스로 직접 내보낸다. 4000바이트 이내.

[태그그룹 전송]

_DeviceWriteGroup("device name", "frame code", "Tag Group")

레시피 데이터를 한 번에 전송할 때 사용한다.

이 드라이버에서 다음 함수는 프레임의 호출여부만 올라옵니다. (0 또는 1)

```
bScan = _DeviceGetScanInfo("device", "C1");
```

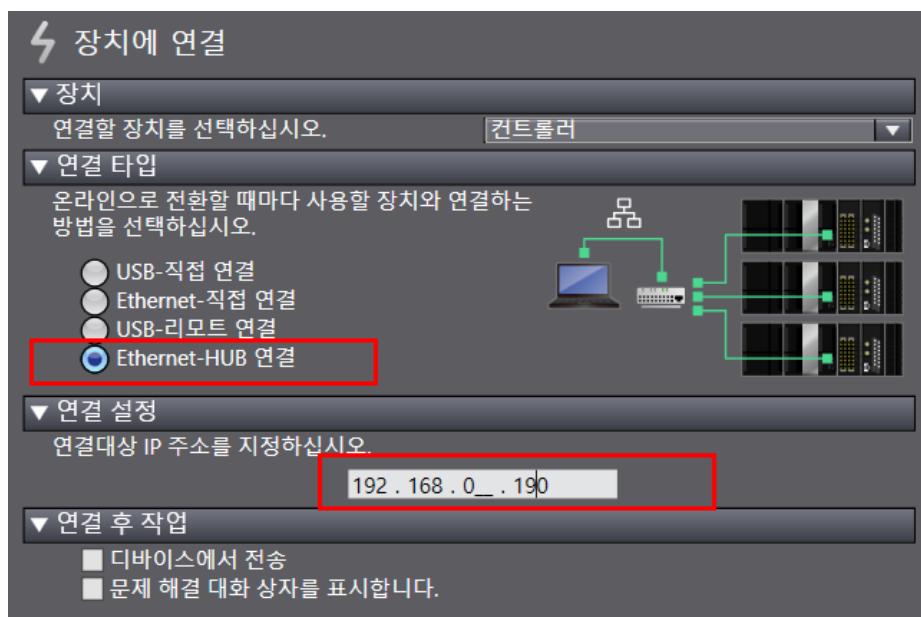
이 드라이버에서 다음 함수는 프레임의 호출여부만 설정합니다. (0=호출취소 또는 1=호출)

```
_DeviceSetScanInfo("device", "C1", 1 or 0);
```

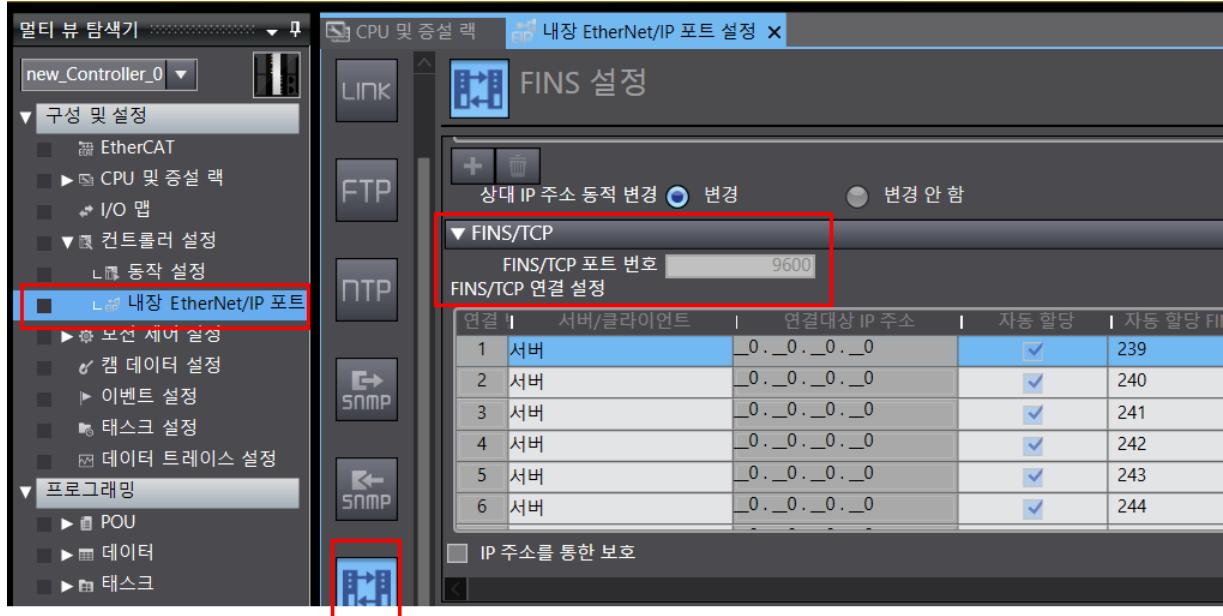
■ OMRON PLC설정 프로그램

OMRON Sysmac Studio TOOL에서 FINS/TCP 네트워크 설정을 해야 합니다.

1) 장치연결



2) 내장 EtherNet/IP 포트설정에서 FINS/TCP 설정하기



3) CPU 유니트 CJ1W-EIP21 설정에서 FINS/TCP 설정하기

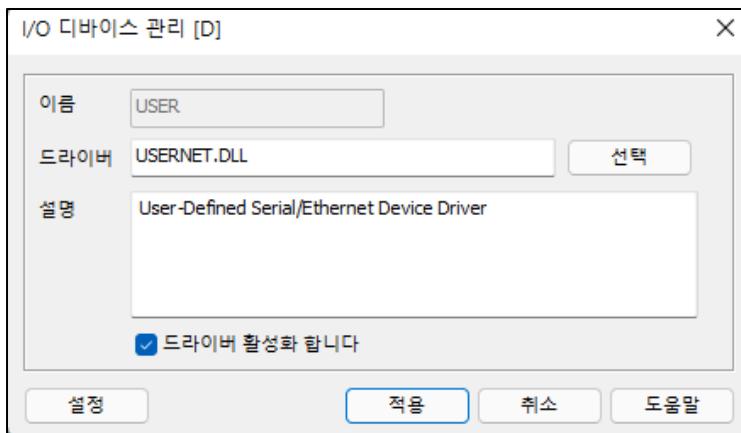


16.7 USERNET

아스키 데이터 통신 프로토콜을 지원하는 디바이스에 맞게 프로토콜을 편집할 수 있도록 하여 다양한 디바이스에 대한 통신이 가능하도록 하는 사용자정의 통신 드라이버입니다.

- 다양한 디바이스 통신 대응 가능
- RS-232C / RS-485 / RS-422 / TCPIP
- 아스크, 바이너리 모두 가능
- 큰 사이즈 데이터를 한 번에 전송 가능 (스크립트 이용)
- 로그기능

DLL 파일은 USERNET.DLL 입니다. 선택버튼을 눌러서 드라이버를 선택합니다.



User-Defined Device Driver Setting(2025-04-15) [Sorter8:ADEX]

No	Code	To Device	From Device	Scan
1	CFG	LNK %s<0d><0a>	NOP	<input type="checkbox"/>
2	U	STR U7<0d><0a>	NOP	<input type="checkbox"/>
3	FR	STR FR<0d><0a>	NOP	<input type="checkbox"/>
4	C1	STR FD<0d><0a>	NOP	<input type="checkbox"/>
5	C2	STR LNK %d<0d><0a>	NOP	<input type="checkbox"/>
6	C3	STR L3L LNK %d<0d><0a>	NOP	<input type="checkbox"/>
7	C4	STR L2L LNK %d<0d><0a>	NOP	<input type="checkbox"/>
8	C5	STR L1L LNK %d<0d><0a>	NOP	<input type="checkbox"/>
9	C6	STR L3H LNK %d<0d><0a>	NOP	<input type="checkbox"/>
10	C7	STR L2H LNK %d<0d><0a>	NOP	<input type="checkbox"/>
11	C8	STR L1H LNK %d<0d><0a>	NOP	<input type="checkbox"/>
12	C9	STR D3L LNK %s<0d><0a>	NOP	<input type="checkbox"/>
13	C10	STR D2L LNK %s<0d><0a>	NOP	<input type="checkbox"/>
14	C11	STR D1L LNK %s<0d><0a>	NOP	<input type="checkbox"/>
15	C12	STR D3H LNK %s<0d><0a>	NOP	<input type="checkbox"/>
16	C13	STR D2H LNK %s<0d><0a>	NOP	<input type="checkbox"/>
17	C14	STR D1H LNK %s<0d><0a>	NOP	<input type="checkbox"/>
18	C15	STR W1<0d><0a>	NOP	<input type="checkbox"/>
19	C16	STR <1B>L<0d><0a>	NOP	<input type="checkbox"/>
20	C17	STR <1B>R<0d><0a>	NOP	<input type="checkbox"/>
21	C18	STR E<0d><0a>	NOP	<input type="checkbox"/>
22	C20	STR E<1B>D<0d><0a>	CNT DSS 1 10 DSS 12 10 DSS 23 10 D ...	<input type="checkbox"/>

Media: RS-232/RS-485 TCP/IP

RS-232/RS-485

Port: COM3

Baud Rate: 9,600 bps

Data Bit: 8 bits

Parity Bit: none

Stop Bit: 1 bit

TCP/IP

IP: 127.0.0.1

Port: 6001

Scan Time (m/s): 50

Timeout (m/s): 5000

Reset On Comm. Error

Unable Clear Buffer On NOP

Handshake: Command

Log Collect

TX Delimiter: |

Log Text: Ascii

Comm. Test

Frame List

Ok Cancel Help

Run Stop Send Test

Add Insert Delete

항목	설명
MEDIA	이 드라이버는 RS-232C/RS-422/485, TCP/IP 연결방식을 지원합니다.
RS-232/422/485	디바이스의 통신사양에 맞춥니다. PC에서 사용할 수 있는 COM포트를 선택합니다. Ex) COM2-9600-8-N-1
TCP Address / Port	TCP 연결디바이스 IP주소, 포트번호를 입력합니다.
Log Collect	통신로그 파일을 남기려면 체크합니다. 로그파일의 위치는 해당 프로젝트의 IOserverLog 폴더에 다음과 같은 파일명 규칙으로 텍스트파일 형태로 기록된다. "/Publish/프로젝트/Log/IOserverLog/DRV.Device_2021_1_31.txt"
Reset On Comm. Error	연결이 끊어졌을 경우 해당 프레임의 결과데이터를 0 으로 리셋하려면 설정해둡니다.
Unable Clear Buffer On NOP	수신 매크로에 NOP 이 사용된경우 수신버퍼 내용삭제를 할것인지 여부를 설정합니다.
Handshake	어떤 방식으로 통신을 수행할지 선택합니다. Command : 명령어를 보내고 그에 대한 응답을 받는 방식입니다. Stream : 장비가 데이터를 일정 주기 또는 실시간으로 지속적으로 전송하는 방식입니다.
Log Keep (일)	LOG COLLECT 옵션이 선택이 되어있어야 하며 통신로그 기록 파일의 보관기간을 입력합니다. 기간이 지난 파일은 삭제합니다. 통신에 장애가 발생할 때 원인파악 분석을 위하여 남겨놓는 데이터 파일입니다. 일반적으로 3 일정도로 설정합니다.
Tx Delimiter	UserNet 프레임 구성은 통신하고자 하는 디바이스의 프로토콜 포맷에 맞추어야 하는데, 프레임을 구성하는 포맷에서 사용하는 구분자를 여기서 지정합니다. 가급적 아스키코드 데이터 통신에서 사용하지 않는 문자 1 바이트를 설정해 줍니다. TX 프레임(디바이스로 내보내는), RX 프레임(디바이스에서 들어오는) 정의 시 Tx Delimiter 를 사용하여야 합니다.
Log Text	통신 로그 데이터를 아스키문자 또는 16 진수로 기록합니다.
Scan Time(m/s)	우측 통신프레임에 설정된 다수의 프레임간 통신간격을 밀리초 단위로 입력합니다. 0 으로 설정하면 프레임간 Interval 이 전혀 없이 고속의 데이터 송수신이 가능합니다. 그러나 다수의 드라이버를 사용하는 경우 밸런스를 위해서 최소값은 0 이 아닌 1 로 설정해주시기 바랍니다.
Timeout (m/s)	프레임 송신을 하고 응답을 기다리는 시간입니다. 이 값을 너무 짧게 하면 정상적으로 송수신이 안될 수 있습니다. 드라이버 디폴트 값을 그대로 사용합니다.
Add (Frame)	통신 명령을 추가합니다. (이것을 프레임 추가한다고 말합니다.)
Insert (Frame)	선택된 프레임을 뒤로 미루고 사이에 프레임을 삽입합니다.
Delete (Frame)	선택한 프레임을 리스트에서 삭제합니다.

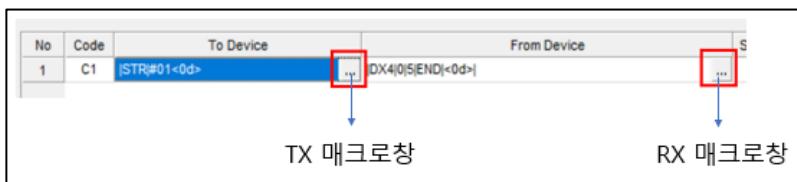
Run Test	설정된 내용으로 디바이스와 통신시험을 합니다. READ / WRITE 모두 가능합니다.
Stop Test	통신시험을 그만합니다.
Send Test	통신시험중에 디바이스로 특정 데이터를 출력하는 시험을 합니다.

■ 통신프레임 설명

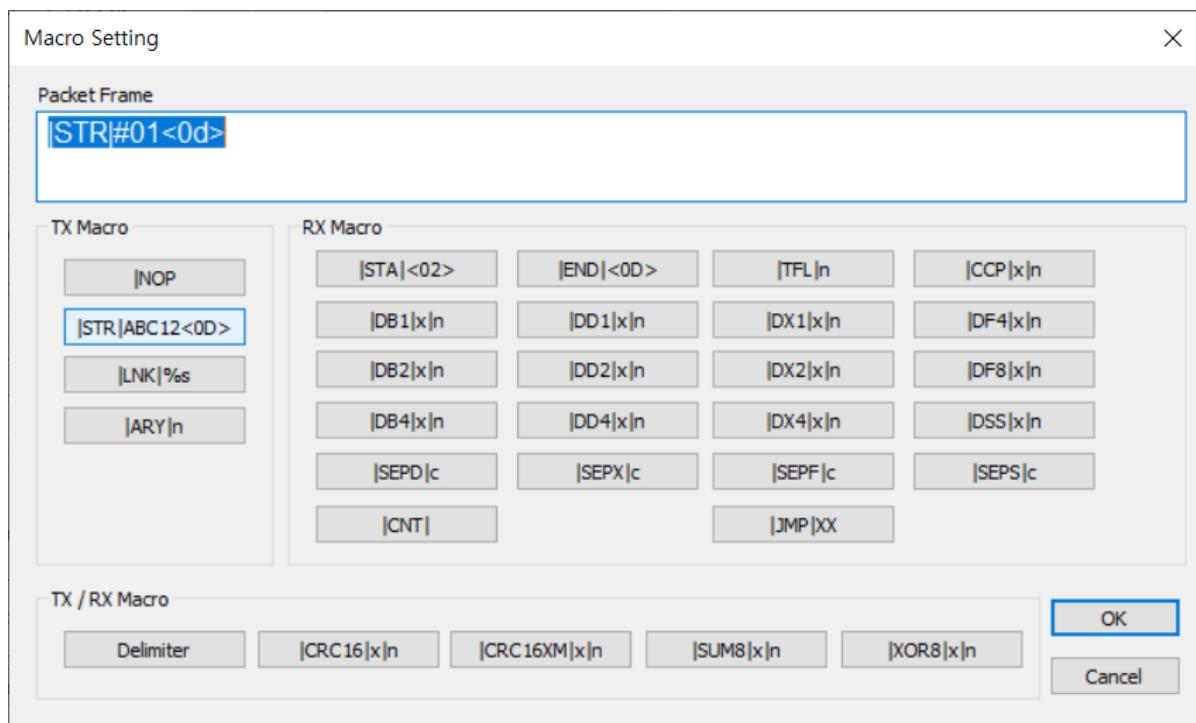
항목	설명
Code	<p>프레임의 이름입니다. 실태그 추가할 때 어드레스로 사용됩니다. 어드레스의 선두번지를 이름으로 보통 사용합니다.</p>
To Device (TX)	<p>디바이스와 통신을 하기 위해서는 특정 패킷프레임(Packet Frame)을 보내주어야 디바이스에서 해당하는 데이터를 되돌려주게 되는데, 이 때 사용할 프레임을 정의합니다. 프레임은 이미 정의되어 있는 매크로와 매크로에 대응하는 값들로 구성이 됩니다.</p> <p>DELIMITER() + MACRO(STR) + MACRO VALUE(erun scada) + DELIMITER()</p> <p style="color: red;"> STR erun scada </p> <p>아스키문자로 구성할 때는 위와 같이 해주고, 바이너리 코드는 괄호문자를 사용해서 <, >로 감싸주어야 합니다.</p> <p>(ex) CR(Carriage Return) => <0D> 또는 <0d>를 사용합니다. LF(Line Feed) <0A> or <0a> 사용합니다.</p> <p>TX 프레임에서 사용가능한 매크로는 NOP, LNK, STR, CRC16, SUM8, XOR8 해서 6 가지 입니다.</p> <div style="border: 1px solid black; padding: 10px; margin-top: 10px;"> <p>The diagram shows a packet frame structure with the following fields:</p> <ul style="list-style-type: none"> STR #01 LNK %02X SUM8 0 5 STR <0d> <p>Annotations with circled numbers:</p> <ol style="list-style-type: none"> ① 아스키 문자열로 그대로 송출된다. ② 상위로부터 데이터를 받아서 디바이스로 송출할 때 사용하는 매크로. ③ 상위데이터 변환포맷.(예)0xAB인 경우 ④ Check Code를 계산후 삽입(인덱스0부터 5바이트) ⑤ Carrige Return Code (0x0D) </div>

From Device (RX)	디바이스로 요구한 프레임에 맞게 디바이스에서 응답하는 포맷을 여기에 정의합니다. RX 프레임은 디바이스로부터 수신된 데이터를 알맞게 분리해서 순서대로 버퍼에 저장하도록 정의합니다. RX 매크로는 별도 설명부분을 참고하시기 바랍니다.
Scan	호출여부를 선택합니다.
Description	프레임 내용을 설명합니다.

■ 패킷 편집창



To Device는 TX 패킷을 입력하는 영역이고 직접 매크로 조합을 입력해도 되지만 "TX 매크로창"을 눌러서 매크로 편집창을 띄워서 편집을 하는 것이 편리합니다. From Device는 RX패킷을 입력하는 영역이고 TX 패킷영역과 마찬가지로 "RX 매크로창"을 띄워서 패킷조합을 할수 있습니다.



TX 패킷을 조합하는데 사용하는 TX Macro가 있고, RX패킷을 조합하는데 사용하는 RX Macro가 있습니다.

■ TX 매크로

매크로(MACRO)	매크로 설명	사용예시
NOP	디바이스로 별도의 패킷을 송출하지 않아도 디바이스에서 무조건 데이터를 보내는 경우 정의한다. 단, RX 프레임은 정의해 주어야 한다.	NOP
LNK	<p>RUN-TIME 실행중에 디바이스로 특정값을 전송해야 할 경우 사용하는 매크로.</p> <p>상위에서 넘겨받아서 송출하는 부분을 정의한다.</p> <p>LNK 가 정의되면 해당 프레임의 스캔타임은 내부에서 자동으로 0 으로 설정되도록 한다. 그러므로 RUN-TIME 실행중에 해당 프레임은 호출되지 않는다.</p> <p>(주의) LNK 매크로를 프레임 각 1 개만 사용할 것.</p> <p>한번에 송출가능한 최대 바이트 크기는 79 이다.</p>	LNK %c -> 1 BYTE 코드값. LNK %02X -> 16 진수값을 2 바이트로 변환. LNK %s -> 문자열 전송. LNK %d<0d> -> 정수값 + CR 값 (123+CR) LNK .2f -> 실수값 (123.5)
STR	문자열 그대로 송출하는 부분을 정의한다.	STR 12345 -> 문자열 "12345" 5 바이트 송출. STR 00WSS%%MW06 -> 송출 프레임 문자열. STR <01><04><AB><0D> -> HEX CODE 4 바이트.
CRC16	지정된 범위의 데이터를 CRC-16 계산으로 2 바이트를 생성하고, 결과값을 상위, 하위 위치 바꾸어서 넣어준다.	CRC16 0 8 -> 0 번째 부터 8 바이트를 CRC16 연산한다. 결과는 2 바이트.
SUM8	지정 범위 데이터를 모두 더해서 256 으로 나눈 나머지 값 1 바이트를 체크코드로 정의한다.	SUM8 1 8 -> 1 번째 부터 8 바이트를 모두 더해서 256 나머지값. 결과는 1 바이트.
XOR8	지정 범위 데이터를 모두 XOR 연산후 결과 1 바이트를 체크코드로 정의한다.	XOR8 0 8 -> 0 번째 부터 8 바이트를 모두 XOR 연산값. 결과는 1 바이트.

■ RX 매크로

매크로(MACRO)	매크로 설명	사용예시
STA	정상적인 수신처리를 위한 첫번째 수신문자를 정의. (SOH, STX, \$, #, etc)	STA <02> -> 0x02(STX)를 시작문자로 지정. STA <0d> -> 0x0D(CR)을 시작문자로 지정.
END	수신완성 처리를 위한 끝문자 1 바이트를 정의. (EOH, ETX, CR, LF, etc)	END \$ -> '\$' 끝 문자로 지정. END <0A> -> 0x0A(LF)를 끝 문자로 지정.
TFL	Total Frame Length. 디바이스로부터 수신되어야 하는 바이트 길이. End Character 와 같이 사용할 수 없다.	TFL 12 -> 정상적인 프레임의 길이 12 바이트.
CRC16	지정된 범위의 데이터를 CRC-16 계산으로 2 바이트를 생성하고, 결과값을 상위, 하위 위치 바꾸어서 넣어준다.	CRC16 0 8 -> 0 번째 부터 8 바이트를 CRC16 연산한다. 결과는 2 바이트.
SUM8	지정 범위 데이터를 모두 더해서 256 으로 나눈 나머지 값 1 바이트를 체크코드로 정의한다.	SUM8 1 8 -> 1 번째 부터 8 바이트를 모두 더해서 256 나머지값. 결과는 1 바이트.
XOR8	지정 범위 데이터를 모두 XOR 연산후 결과 1 바이트를 체크코드로 정의한다.	XOR8 0 8 -> 0 번째 부터 8 바이트를 모두 XOR 연산값. 결과는 1 바이트.
CCP	디바이스에서 받은 패킷에서 CRC16/SUM8/XOR8 계산값이 위치한 정보를 설정한다.	CCP 8 2 -> 8 번째 2 바이트
DB1	소스데이터를 HEX CODE 로 인식해서 실제 데이터 추출하고, 1 BYTE 형 데이터로 버퍼에 넣어준다.	DB1 3 1 <01><02> -> Data 3 번째 HEX 코드 1BYTE 로 인식처리.
DB2	소스데이터를 HEX CODE 로 인식해서 실제데이터 추출하고 2BYTE(WORD)형 데이터로 버퍼에 넣어준다.	DB2 3 4 -> <01><02><03><04> -> 3 번째 부터 4 바이트를 WORD 로 변환.

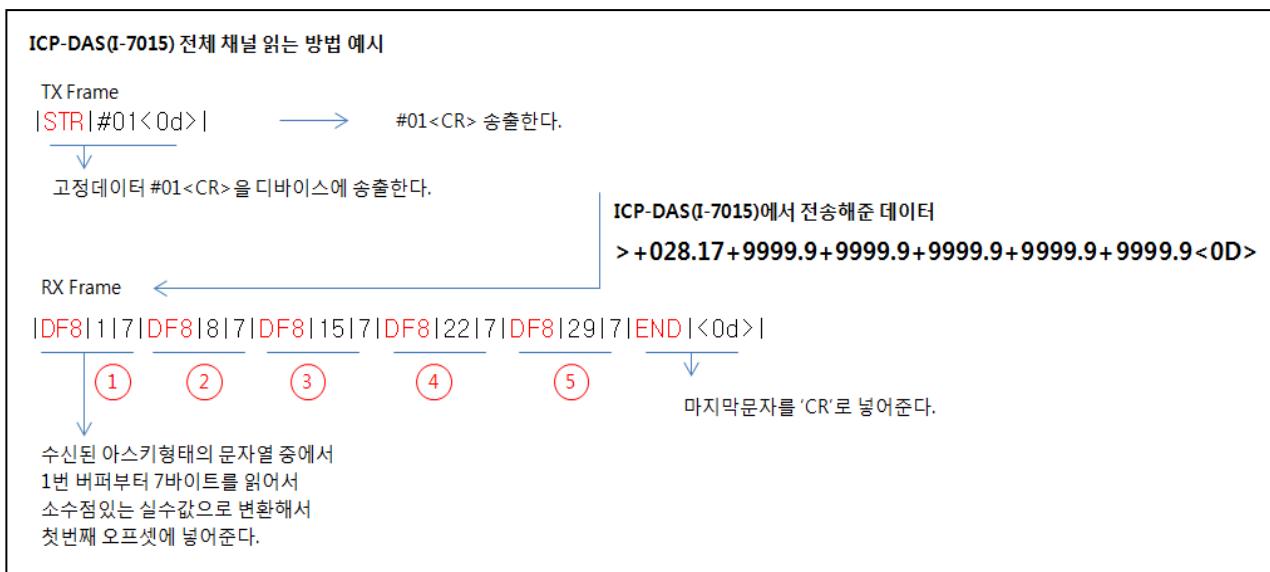


제 16 장 디바이스 드라이버

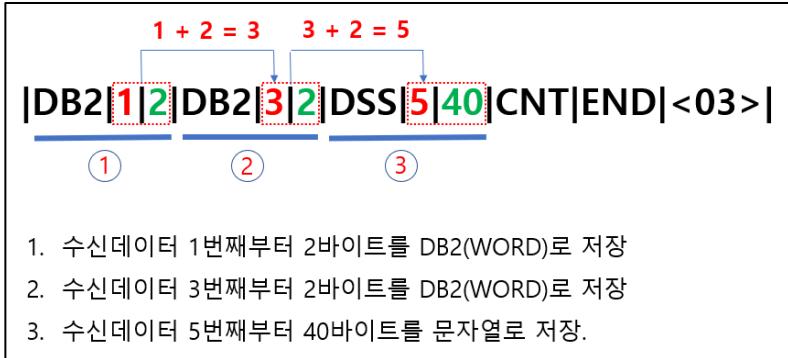
DB4	소스데이터를 HEX CODE 로 인식해서 실제데이터 추출하고 4BYTE(LONG)형 데이터로 버퍼에 넣어준다.	DB4 3 4 -> <01><02><03><04> ->3 번째 부터 4 바이트를 LONG 변환.
DD1	소스를 10 진수로 인식, 1 BYTE 형 데이터버퍼 저장.	DD1 3 2 ->12 ->3 번째 부터 2 바이트를 BYTE로 변환.
DD2	소스를 10 진수로 인식, 2 BYTE 형 데이터버퍼 저장.	DD2 3 4 -> 1234
DD4	소스를 10 진수로 인식, 4 BYTE 형 데이터버퍼 저장.	DD4 3 8 -> 12345678
DX1	소스를 16 진수로 인식, 1 BYTE 형 데이터버퍼 저장.	DX1 3 2 -> AB
DX2	소스를 16 진수로 인식, 2 BYTE 형 데이터버퍼 저장.	DX2 3 4 -> ABCD ->3 번째 부터 4 바이트를 WORD로 변환.
DX4	소스를 16 진수로 인식, 4 BYTE 형 데이터버퍼 저장.	DX4 3 8 -> ABCD1234 ->3 번째 부터 8 바이트를 LONG로 변환.
DF4	소스를 아스키 소수점 있는 숫자로 인식. (4 바이트형)	DF4 3 4 -> 123.4 ->3 번째 부터 4 바이트를 DOUBLE로 변환.
DF8	소스를 아스키 소수점 있는 숫자로 인식. (8 바이트형)	DF8 3 4 -> 123.4 ->3 번째 부터 4 바이트를 DOUBLE로 변환.
DSS	수신된 데이터소스를 아스키 문자열로 인식. (주의)DSS 매크로는 단독으로 사용할 것 수신 가능한 길이는 79 바이트.	DSS 3 4 ->3 번째 부터 4 바이트를 문자열로 변환없음.
SEPD	Seperate as Decimal 지정된 구분 문자까지의 데이터소스를 10 진수로 인식. 4 바이트 크기.	SEPF <0D> -> 구분문자 CR(0x0D) 까지의 문자열을 LONG 형으로 변환.
SEPX	Seperate as Hexa 지정된 구분 문자까지의 데이터소스를 16 진수로 인식. 4 바이트 크기.	SEPF <0D> -> 구분문자 CR(0x0D) 까지의 문자열을 16 진수로 LONG 형으로 변환
SEPF	Seperate as Float 지정된 구분 문자까지의 데이터소스를 소수점숫자로 인식. 8 바이트 크기	SEPF <0D> -> 구분문자 CR(0x0D) 까지의 문자열을 DOUBLE 형의 변환.
SEPS	SEPERATE AS STRING	SEPS <0A>

	지정된 구분 문자까지의 데이터소스를 문자열로 인식 (주의)SEPS 매크로는 단독으로 사용할 것 수신 가능한 길이는 79 바이트.	-> 구분문자 LF(0x0A) 까지의 문자열을 그대로 저장.
CNT	COUNTER 수신 패킷의 수신완료 때마다 1 씩 증가 4 바이트 크기.	CNT
JMP	송신프레임에서만 사용되며, 해당 프레임이 송출되고 디바이스로 부터 수신되면 다음 송출할 프레임으로 넘어가도록 한다 강제 JUMP 하려고 하는 프레임번호가 등록된 프레임개수를 넘어가면 안 된다. 만일 그러면 JUMP 는 무시되고 넘어간다. .	JMP 2 디바이스 수신완료 후 2 번 프레임 실행으로 넘어가게 한다. 수신 프레임에는 사용할 수 없다.

■ TX 매크로 정의와 RX 매크로 정의 예시



■ RX 패킷의 버퍼 인덱스정의 예시



■ RX 패킷 배열매치

STA <02> DD4 1 5 DX2 6 4 DF4 10 6 DS 16 8 CRC16 24 2 END <03>																											
INDEX	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
DATA	<02>	1	2	3	4	5	A	B	C	D	5	4	3	.	3	1	M	o	d	e	I	-	9	0	<F9>	<C5>	<03>
USERNET		STA	DD4 1 5				DX2 6 4				DF4 10 6				DS 16 8				CRC16		END						
STA <01> DB4 3 4 DB2 7 2 CRC16 9 2 TFL 11																											
INDEX	0	1	2	3	4	5	6	7	8	9	10																
DATA	<01>	<04>	<06>	<11>	<22>	<33>	<44>	<AB>	<CD>	<6A>	<21>																
USERNET		STA					DB4				DB2				CRC16		TFL										

■ 태그값을 디바이스로 출력정의 예시

송출 프레임 예시 -> |STR|#0001|LINK%#d<0d>

출력태그 설정 방법

번호	이름	태그종류	디바이스	어드레스	가상	스캔	공유
1	F1	아날로그	SEP	RX:1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
2	F2	아날로그	SEP	RX:2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
3	F3	아날로그	SEP	RX:3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
4	COUNT_RX	아날로그	SEP	RX:4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
5	SET_VALUE	아날로그	SEP	TX	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

No	Code	To Device	From Device
1	RX	NOP	... SEPD <2c> SEPD <2c> SEPD <2c> CNT END <0D> ...
2	TX	STR #0001 LNK %d<0d>	... NOP ...

- To Device한후 From Device 그리고 곧바로 Ack등을 보내주어야 하는경우 프레임설정
→ 바로 다음 프레임에 To Device에 |STR|<06>을 하고, From Device에는 |NOP|을 설정합니다.
- RX 패킷의 태그연결 예시

(EX1)

[CX_FR] |DB2|1|2|DB2|3|2|DSS|5|40|CNT|END|<03>|

DB2|1|2| -> CX_FR:1
 DB2|3|2| -> CX_FR:2
 DSS4|5|40| -> CX_FR:3
 CNT -> CX_FR:23 (offset = 3 + 40/2 = 23)

(EX2)

1) 디바이스로부터 수신된 패킷

"<02><01><80>NEFES@SEFA@@@<03>" 12바이트

2) RX 패킷 설정

[CX_FR] |DB2|1|2|DB2|3|1|DSS|4|12|CNT|END|<03>|

3) 태그연결 설정

DB2|1|2| -> CX_FR:1
 DB2|3|1| -> CX_FR:2
 DSS4|4|12| -> CX_FR:3
 CNT -> CX_FR:9 (offset = 3 + 12/2 = 9)

(EX3)

[CX_FR] |DB2|1|2|DB2|3|1|DSS|4|112|CNT|END|<03>|

DB2|1|2| -> CX_FR:1
 DB2|3|1| -> CX_FR:2
 DSS4|4|112| -> CX_FR:3
 CNT -> CX_FR:59 (offset = 3 + 112/2 = 59)

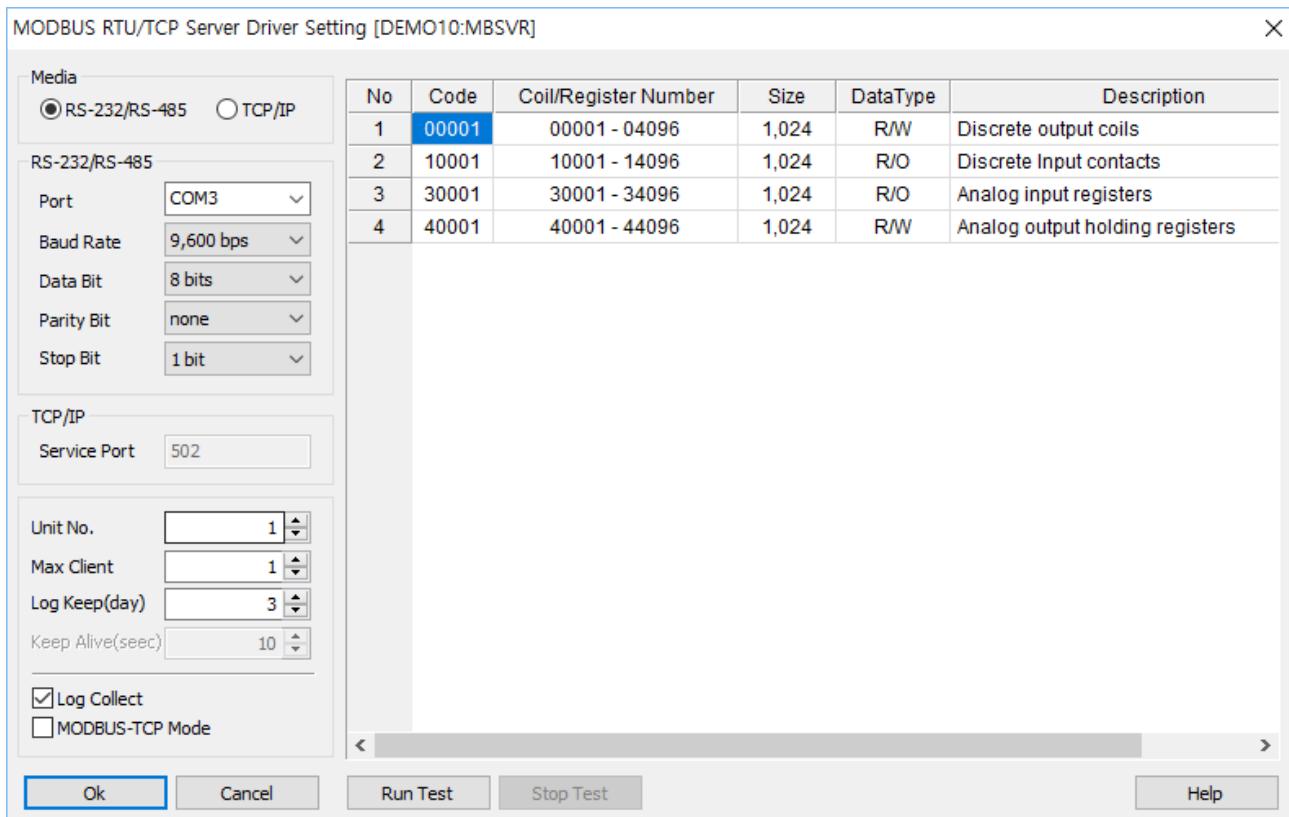
16.8 MODBUS SERVER

MODBUS RTU(Remote Terminal Unit), MODBUS-TCP 프로토콜에 의한 모드버스 슬레이브 서비스를 위한 서버 인터페이스 드라이버입니다. N대의 HMI간 데이터 공유가 필요할 경우 사용할 수 있습니다.

I/O Server에 등록되면, 해당포트를 서버서비스(Passive) 상태로 열고 클라이언트(다른 HMI)에서 MODBUS RTU/TCP 접속을 하용하고 데이터를 주고 받습니다. RS-232/RS-485 사용할 경우 1:1 통신을 하는 드라이버로 동작하며 TCP/IP 인 경우 1:N 통신 서비스 가능합니다.

- 모드버스 RTU / TCP 모드지원
- RS-232C / RS-485 / RS-422 / TCPIP
- N대의 모드버스 클라이언트 연결가능
- 큰 사이즈 데이터를 한 번에 전송 가능 (스크립트 이용)
- 로그기능

DLL 파일은 MBSVR.DLL 입니다. 선택버튼을 눌러서 드라이버를 선택합니다.



■ 설정내용

항목	설명
MEDIA	이 드라이버는 RS-232C/RS-422/485, TCP/IP 연결방식을 지원합니다.
RS-232/422/485	디바이스의 통신사양에 맞춥니다. PC에서 사용할 수 있는 COM포트를 선택합니다. Ex) COM2-9600-8-N-1
TCP Service Port	MODBUS TCP 서비스포트는 기본값이 502입니다.
Unit No.	MODBUS 프로토콜은 장치번호를 기본적으로 사용하므로 1~250 사이의 10 진수로 설정한다.
Max Client	미디어가 TCP/IP 인 경우 동시에 서비스 가능한 클라이언트 수를 설정한다. 최대 10 개 클라이언트 지원한다. RS-232/RS-485 인 경우는 의미가 없다.
Keep Alive	MODBUS TCP 통신인 경우 연결된 클라이언트로 부터 일정 시간동안 송수신 데이터가 없을 경우 원활한 서버서비스를 위해서 해당 클라이언트 접속을 끊는다. 이때 설정하는 초단위 시간이다. 이 값이 0 이면 이 기능을 사용하지 않는다
Log Keep	로그파일 기록기간을 입력합니다.
Log Collect	통신로그 파일을 기록합니다. <u>"/Publish/프로젝트/Log/IOServerLog/DRV.Device_2021_1_31.txt"</u>
	연결이 끊어졌을 경우 해당 프레임의 결과데이터를 0 으로 리셋하려면 설정해둡니다.
MODBUS-TCP Mode	MODSVR 드라이버는 MODBUS-RTU / MODBUS-TCP 서비스를 지원한다. MODBUS RTU 와 MODBUS TCP 의 프로토콜은 서로 상이하며 주로 미디어가 TCP/IP 인 경우에 사용한다. 미디어가 RS-232/RS-485 인 경우에도 MODBUS-TCP Mode 사용은 가능하지만 가급적 사용을 권하지 않는다.
Run Test	설정된 내용으로 디바이스와 통신시험을 합니다. READ / WRITE 모두 가능합니다.
Stop Test	통신시험을 그만합니다.

■ 통신프레임 설명

항목	설명
Code (프레임 인식코드)	해당 프레임을 대표하며 상위 실태그에서 해당 디바이스 값을 요구할 때 또는 디바이스에 값을 기록할 때 사용되는 디바이스 어드레스 코드이다
Coil / Register number	MODBUS 메모리 종류이며 4 가지로 사용된다. 1) Discrete output coils (R/W 용) Modbus Function code 0x01 에 해당한다. (1 - 4096) Modbus client에서 Server 측으로 전송하는 비트 데이터가 기록되는 프레임 버퍼이다.

	<p>이때, Client 측에서 사용하는 Function code 는 0x05.</p> <p>2) Discrete input status (R/O 용) Modbus Function code 0x02 에 해당한다. (10001 - 14096)</p> <p>3) Analog input register (R/O 용) Modbus Function code 0x04 에 해당한다. (30001 - 34096)</p> <p>4) Analog output holding register (R/W 용) Modbus Function code 0x03 에 해당한다. (40001 - 44096)</p> <p>Modbus client 에서 Server 측으로 전송하는 아날로그 데이터(16bit)가 기록되는 프레임 버퍼이다.</p> <p>이때, Client 측에서 사용하는 Function code 는 0x06.</p>
Size(버퍼크기)	<p>불필요한 메모리낭비를 위해서 필요에 따라 해당프레임의 크기를 지정한다.</p> <p>각 프레임의 크기는 최소 1에서 최대 4096 까지 설정가능 하다.</p> <p>00001, 10001 프레임은 최대 4096 바이트크기이며, 30001, 40001 프레임은 4096 워드크기(8192 바이트크기)이다.</p>
DataType (데이터형태)	<p>Client 측에서 바라볼 경우 사용되는 프레임 버퍼 액세스방향을 의미한다.</p> <p>R/O (Read Only) 읽기만 가능하다.</p> <p>R/W (Read Write) 읽기 그리고 쓰기가 가능한다.</p> <p>그러나 Server 측에서는 프레임 관계없이 데이터 읽기/쓰기가 모두 가능하다.</p>
Description(프레임설명)	해당프레임에 대한 설명이며 사용자가 편집할 수 없다.

1) Comms. View

- 클라이언트와의 통신메시지를 보여준다.

I/O Device Monitoring

Device Name	Driver File	Version	Setting	Service	Connected
MBSVR	MBSVR.dll	2.5.0.1	COM3-9600-8-N-1	OK	1/1

```

MODBUS-RTU Server started serial with..,COM3-9600-8-N-1
Command 3, Unit=1, Address=0, ReadLen=2(0x2)
-> Response Command 3, Unit=1, Address=0, ReadLen=2(0x2).
Command 4, Unit=1, Address=0, ReadLen=2(0x2)
-> Response Command 4, Unit=1, Address=0, ReadLen=2(0x2).
Command 6, Unit=1, Address=0, WriteData=255(0xFF)
-> Response Command 6, Unit=1, Address=0, WriteData=255(0xFF).
Command 6, Unit=1, Address=1, WriteData=255(0xFF)
-> Response Command 6, Unit=1, Address=1, WriteData=255(0xFF).
Command 6, Unit=1, Address=FFFF, WriteData=255(0xFF), illegal request data.
modbus-rtu : Check code error...
Command 4, Unit=1, Address=0, ReadLen=2(0x2)
-> Response Command 4, Unit=1, Address=0, ReadLen=2(0x2).
Command 3, Unit=1, Address=0, ReadLen=2(0x2)
-> Response Command 3, Unit=1, Address=0, ReadLen=2(0x2).

```

Comms.View Coil Outputs Digital Inputs Analog Inputs Holding Registers Pause Clear

2) Coil Outputs

- 클라이언트로부터 비트데이터를 받을 수 있다. Read / Write.

I/O Device Monitoring

Device Name	Driver File	Version	Setting	Service	Connected
MBSVR	MBSVR.dll	2.5.0.1	COM3-9600-8-N-1	OK	1/1

Address	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+10	+11	+12	+13	+14	+15
00001 - 00016	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00017 - 00032	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00033 - 00048	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00049 - 00064	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00065 - 00080	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00081 - 00096	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00097 - 00112	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00113 - 00128	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00129 - 00144	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00145 - 00160	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00161 - 00176	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00177 - 00192	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00193 - 00208	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00209 - 00224	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00225 - 00240	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00241 - 00256	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00257 - 00272	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00273 - 00288	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00289 - 00304	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00305 - 00320	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00321 - 00336	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Comms.View Coil Outputs Digital Inputs Analog Inputs Holding Registers Pause Clear

3) Digital Inputs

- 셀에 데이터를 설정하면 클라이언트에서 읽어간다. Read only.

I/O Device Monitoring																X	
Device Name		Driver File								Version		Setting				Service	Connected
MBSVR		MBSVR.dll								2.5.0.1		COM3-9600-8-N-1				OK	1/1
Address	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+10	+11	+12	+13	+14	+15	
10001 - 10016	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
10017 - 10032	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
10033 - 10048	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
10049 - 10064	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
10065 - 10080	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
10081 - 10096	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
10097 - 10112	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
10113 - 10128	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
10129 - 10144	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
10145 - 10160	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
10161 - 10176	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
10177 - 10192	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
10193 - 10208	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
10209 - 10224	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
10225 - 10240	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
10241 - 10256	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
10257 - 10272	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
10273 - 10288	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
10289 - 10304	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
10305 - 10320	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
10321 - 10336	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

Comms.View | Coil Outputs | **Digital Inputs** | Analog Inputs | Holding Registers | Pause | Clear

4) Analog Inputs.

- 셀에 WORD(2바이트) 데이터를 설정하면 클라이언트에서 읽어간다. Read Only.

I/O Device Monitoring

Device Name		Driver File				Version	Setting			Service	Connected
MBSVR		MBSVR.dll				2.5.0.1	COM3-9600-8-N-1			OK	1/1
Address	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	
30001 - 30010	0	0	0	0	0	0	0	0	0	0	
30011 - 30020	0	0	0	0	0	0	0	0	0	0	
30021 - 30030	0	0	0	0	0	0	0	0	0	0	
30031 - 30040	0	0	0	0	0	0	0	0	0	0	
30041 - 30050	0	0	0	0	0	0	0	0	0	0	
30051 - 30060	0	0	0	0	0	0	0	0	0	0	
30061 - 30070	0	0	0	0	0	0	0	0	0	0	
30071 - 30080	0	0	0	0	0	0	0	0	0	0	
30081 - 30090	0	0	0	0	0	0	0	0	0	0	
30091 - 30100	0	0	0	0	0	0	0	0	0	0	
30101 - 30110	0	0	0	0	0	0	0	0	0	0	
30111 - 30120	0	0	0	0	0	0	0	0	0	0	
30121 - 30130	0	0	0	0	0	0	0	0	0	0	
30131 - 30140	0	0	0	0	0	0	0	0	0	0	
30141 - 30150	0	0	0	0	0	0	0	0	0	0	
30151 - 30160	0	0	0	0	0	0	0	0	0	0	
30161 - 30170	0	0	0	0	0	0	0	0	0	0	
30171 - 30180	0	0	0	0	0	0	0	0	0	0	
30181 - 30190	0	0	0	0	0	0	0	0	0	0	
30191 - 30200	0	0	0	0	0	0	0	0	0	0	
30201 - 30210	0	0	0	0	0	0	0	0	0	0	

Comms.View | Coil Outputs | Digital Inputs | Analog Inputs | Holding Registers | Pause | Clear

5) Holding Registers.

- 클라이언트로부터 Word데이터를 받을 수 있다. Read / Write.

I/O Device Monitoring X

Device Name		Driver File		Version		Setting			Service		Connected
MBSVR	MBSVR.dll	2.5.0.1		COM3-9600-8-N-1			OK	1/1			
Address	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	
40001 - 40010	255	255	0	0	0	0	0	0	0	0	
40011 - 40020	0	0	0	0	0	0	0	0	0	0	
40021 - 40030	0	0	0	0	0	0	0	0	0	0	
40031 - 40040	0	0	0	0	0	0	0	0	0	0	
40041 - 40050	0	0	0	0	0	0	0	0	0	0	
40051 - 40060	0	0	0	0	0	0	0	0	0	0	
40061 - 40070	0	0	0	0	0	0	0	0	0	0	
40071 - 40080	0	0	0	0	0	0	0	0	0	0	
40081 - 40090	0	0	0	0	0	0	0	0	0	0	
40091 - 40100	0	0	0	0	0	0	0	0	0	0	
40101 - 40110	0	0	0	0	0	0	0	0	0	0	
40111 - 40120	0	0	0	0	0	0	0	0	0	0	
40121 - 40130	0	0	0	0	0	0	0	0	0	0	
40131 - 40140	0	0	0	0	0	0	0	0	0	0	
40141 - 40150	0	0	0	0	0	0	0	0	0	0	
40151 - 40160	0	0	0	0	0	0	0	0	0	0	
40161 - 40170	0	0	0	0	0	0	0	0	0	0	
40171 - 40180	0	0	0	0	0	0	0	0	0	0	
40181 - 40190	0	0	0	0	0	0	0	0	0	0	
40191 - 40200	0	0	0	0	0	0	0	0	0	0	
40201 - 40210	0	0	0	0	0	0	0	0	0	0	

Comms.View |
 Coil Outputs |
 Digital Inputs |
 Analog Inputs |
 Holding Registers
Pause
Clear

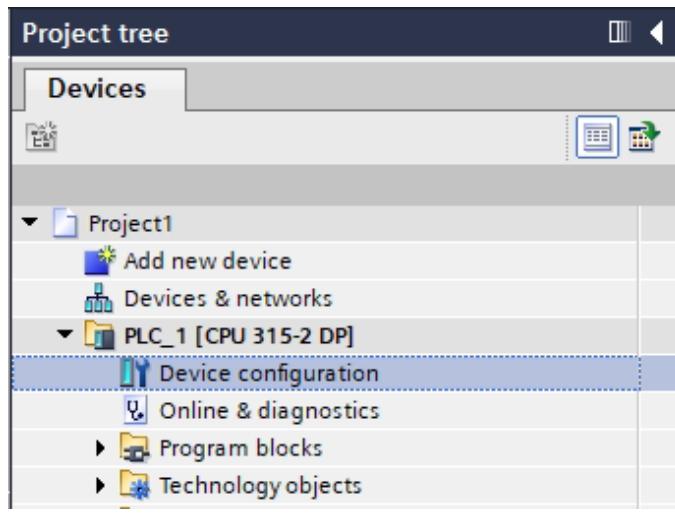


16.9 SIEMENS PLC S7-300

16.9.1 SIEMENS S7 300시리즈 이더넷 통신을 하기 위한 PLC 설정

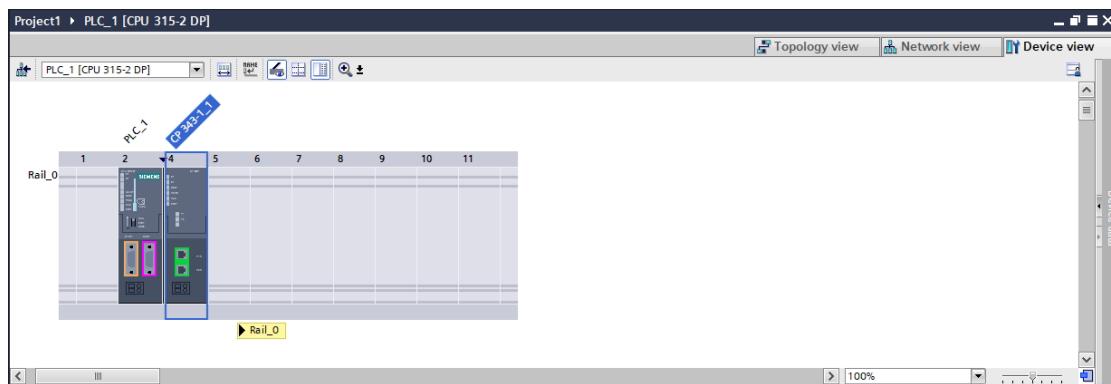
본 내용은 TIA Portal V16을 기준으로 작성되었으며, 지멘스 S7 PLC와 eRun간의 이더넷 통신을 하기위한 S7 PLC에서 설정해야할 사항들에 대한 내용입니다. TIA Portal의 프로젝트 생성 및 프로그램 업로드/다운로드는 해당 문서에 포함되지 않으니 TIA Portal 관련 매뉴얼을 참고하시기 바랍니다.

1. 프로젝트를 열면 아래 그림과 같이 Project Tree 가 화면 좌측에 표시가 됩니다.



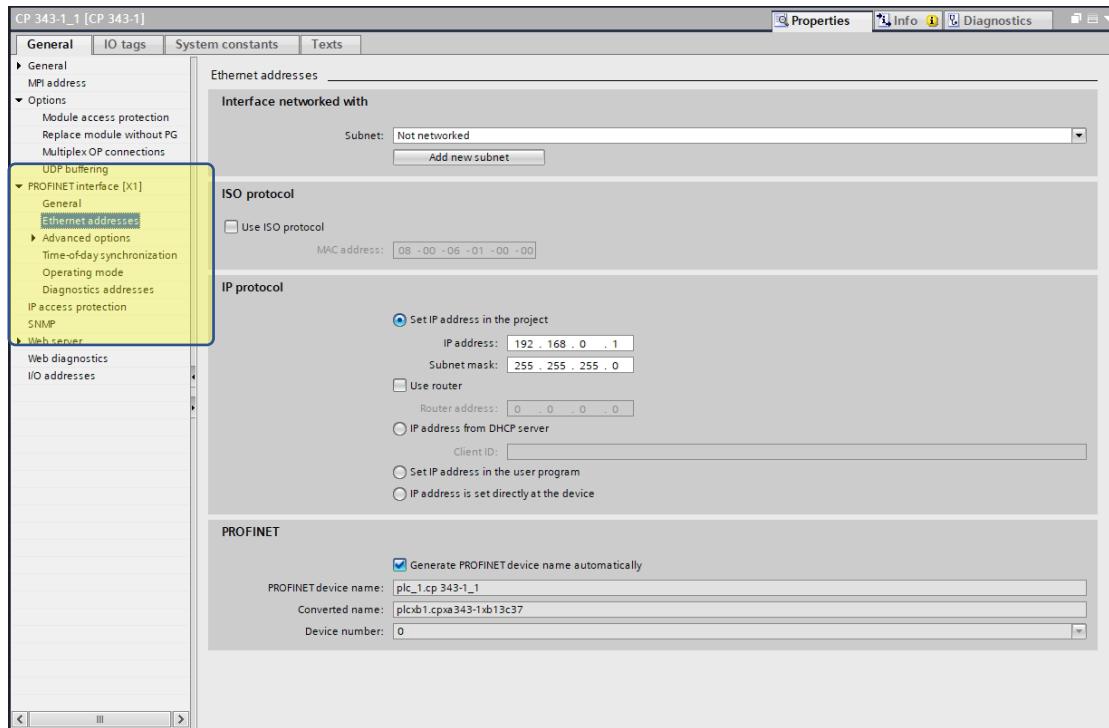
프로젝트 트리에서 Device configuration 항목을 더블클릭하면 화면 우측에 Device View 항목에 현재 설정된 PLC의 구성이 다음 그림과 같이 표시됩니다.

2. 이더넷 모듈을 선택하면 아래 그림과 같이 파란색으로 선택됩니다.



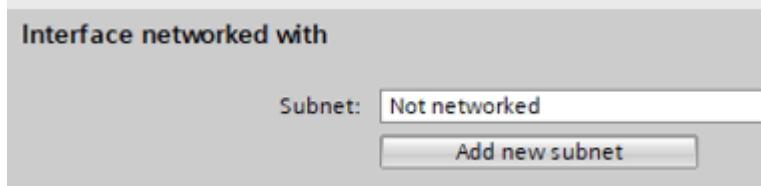
위의 그림처럼 이더넷 카드를 선택하면 다음 그림처럼 화면 아래에 이더넷 관련 항목을 설정할수 있는 속성항목들이 표시됩니다.

3. General 항목에서 PROFINET interface[X1]을 더블클릭하여 확장을 하면 아래 그림 좌측처럼 세항목이 표시됩니다. (노란영역)

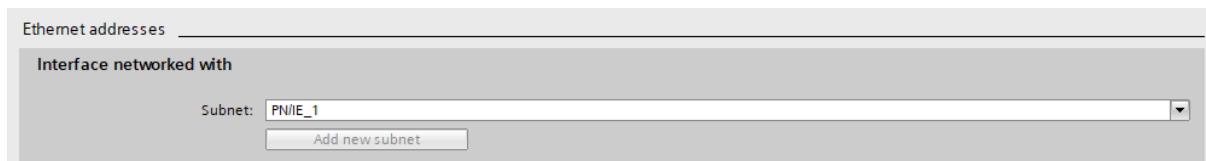


PROFINET interface[X1] 항목에서 Ethernet addresses 항목을 클릭하면 위의 그림처럼 우측에 이더넷 설정관련 항목이 우측에 표시됩니다.

4. Interface networked with 항목에서 아래 그림과 같이 subnet 항목이 Not networked 상태이면

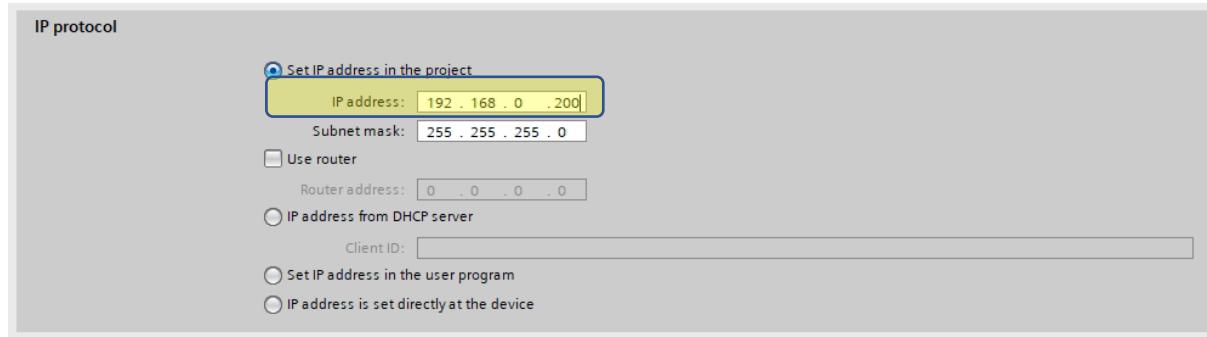


Add new subnet 을 클릭하여 새로운 subnet 을 생성합니다.



새로운 subnet이 생성이 되었으면 위의 그림과 같이 Subnet 항목에 새로운 항목이 추가된 것을 확인할 수 있습니다.

5. IP Address 를 설정합니다. (노란영역)



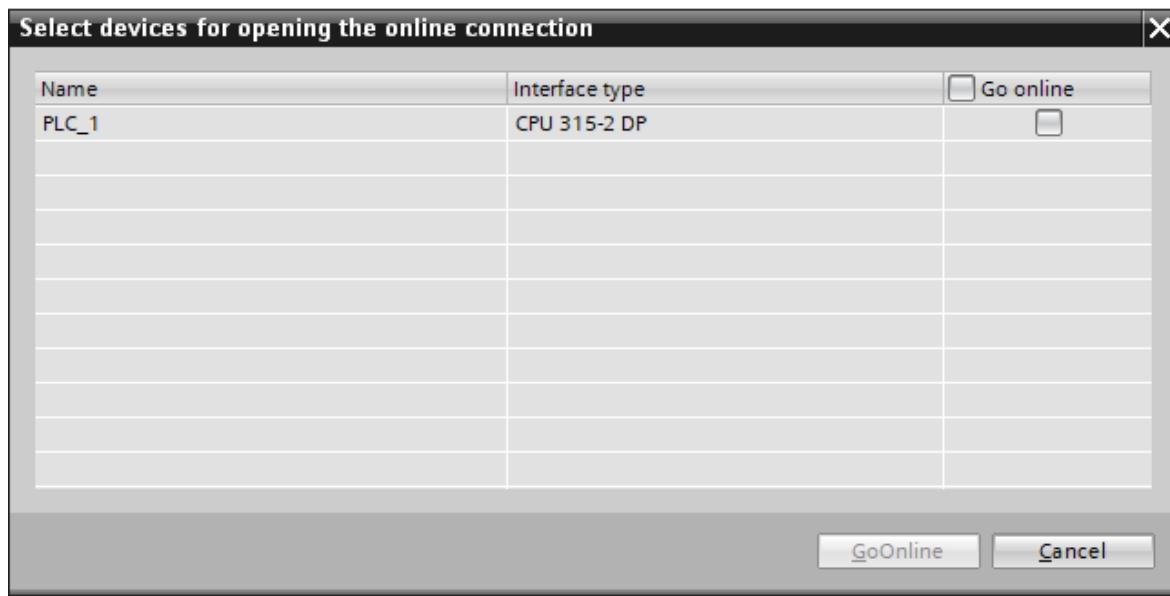
기본적으로 IP는 192.168.0.1로 설정되어 있습니다.

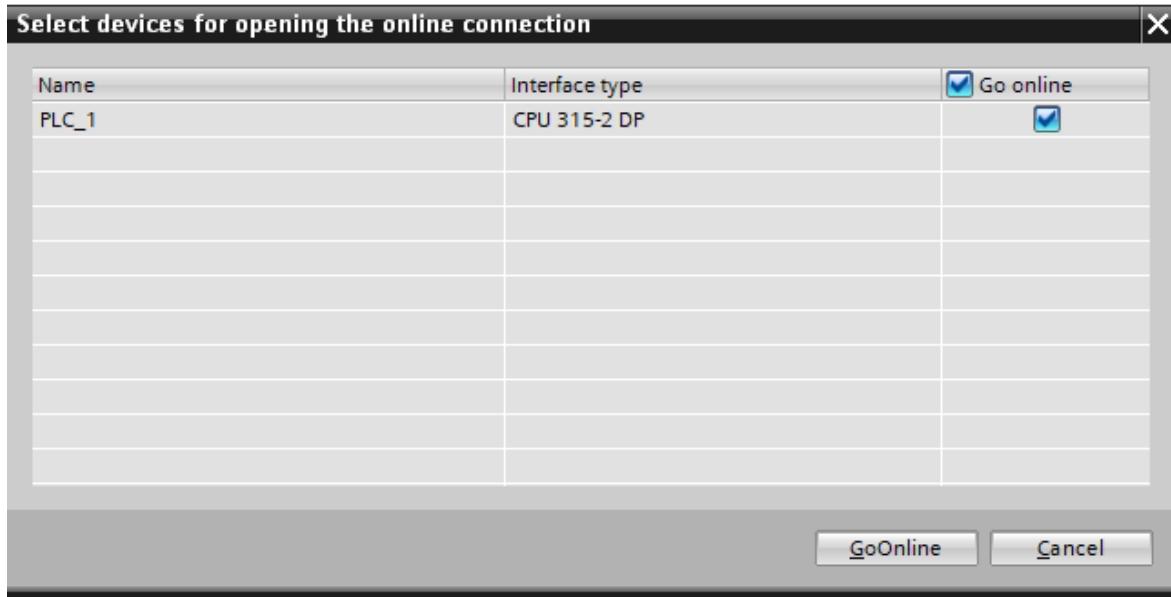
참고: 네트워크 IP 대역이 192.168.0.XXX인 경우에 기본으로 설정된 192.168.0.1을 사용하시지 마십시오. 192.168.0.1은 일반적으로 게이트웨이의 IP이기 때문에 IP 충돌이 발생될 수 있습니다.

6. IP를 설정후에 툴바에서 Go online 버튼을 클릭합니다. (노란영역)



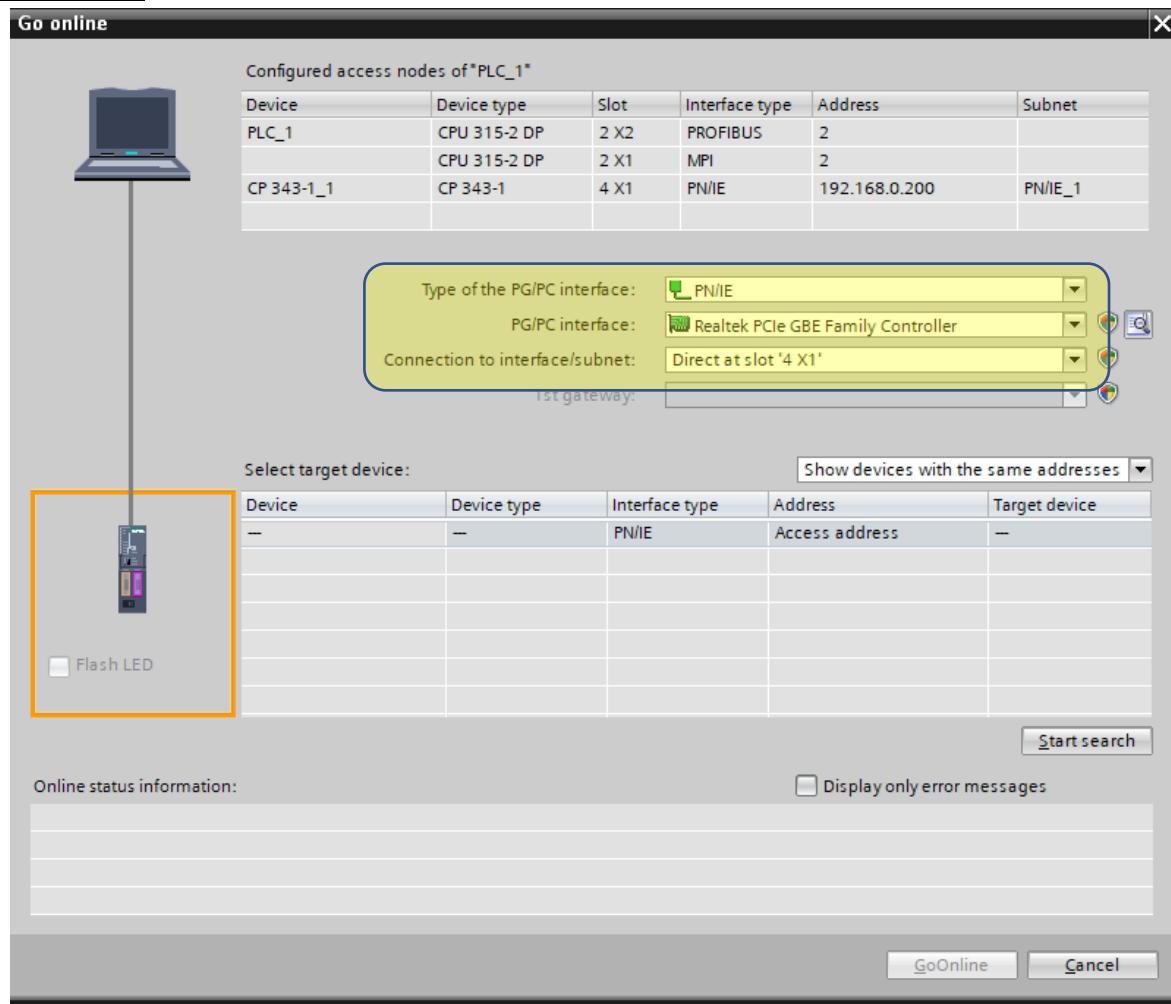
7. Go online 버튼을 누르면 아래와 같이 디바이스 선택창이 표시됩니다.





아래 그림과 같이 PLC를 선택 후 GoOnline 버튼을 클릭하면 다음과 같은 화면이 표시됩니다.

8. Go online 화면에서 우선적으로 노란색 영역의 설정을 확인합니다.

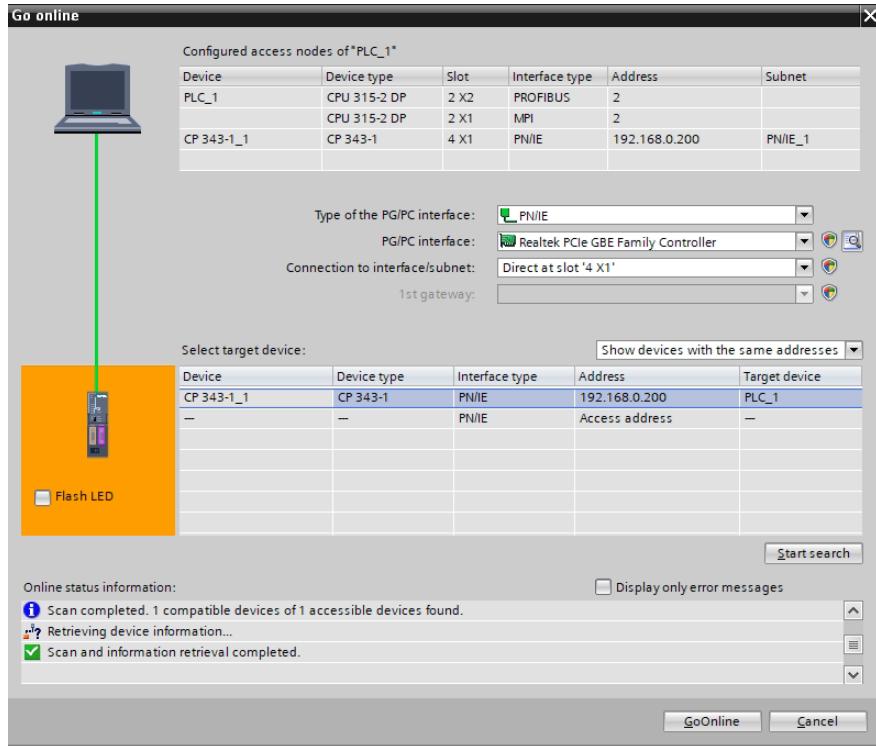


Type of the PG/PC interface 항목 PN/IE로 선택을 합니다.

PG/PC interface: 현재 PC의 이더넷 카드 모델을 선택합니다.

Connection to interface/subnet: Direct at slot '4X1'을 선택합니다.

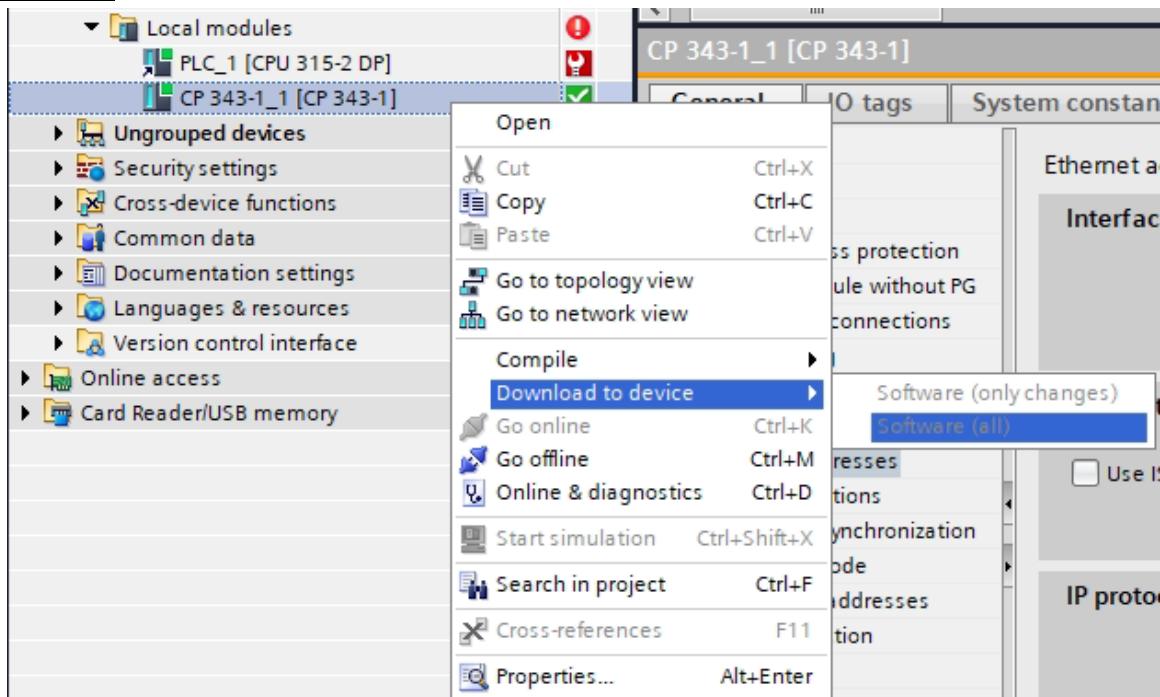
위의 사항을 확인후에 Start search 버튼을 누르면 현재 네트워크에 있는 PLC를 검색해서 다음 그림과 같이 검색 결과를 표시합니다.



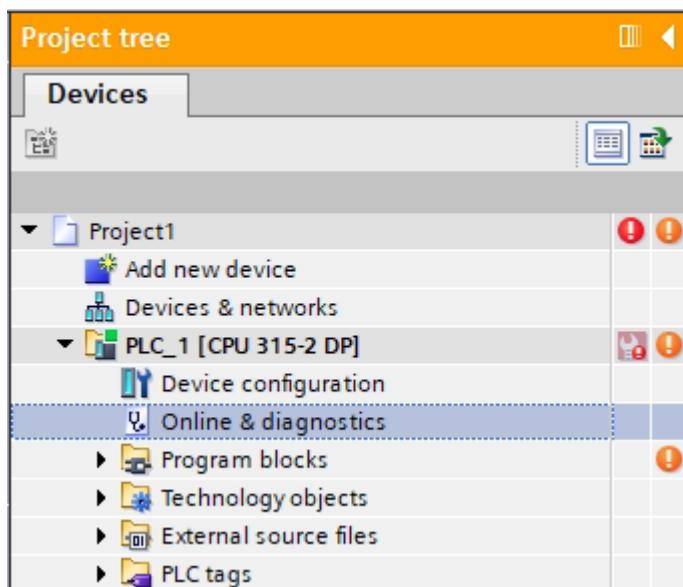
10. Select target device에서 PLC를 선택 후 GoOnline 을 선택합니다.

11. PLC에 접속이 되었으면 현재 설정된 이더넷 정보를 PLC에 다운로드합니다.

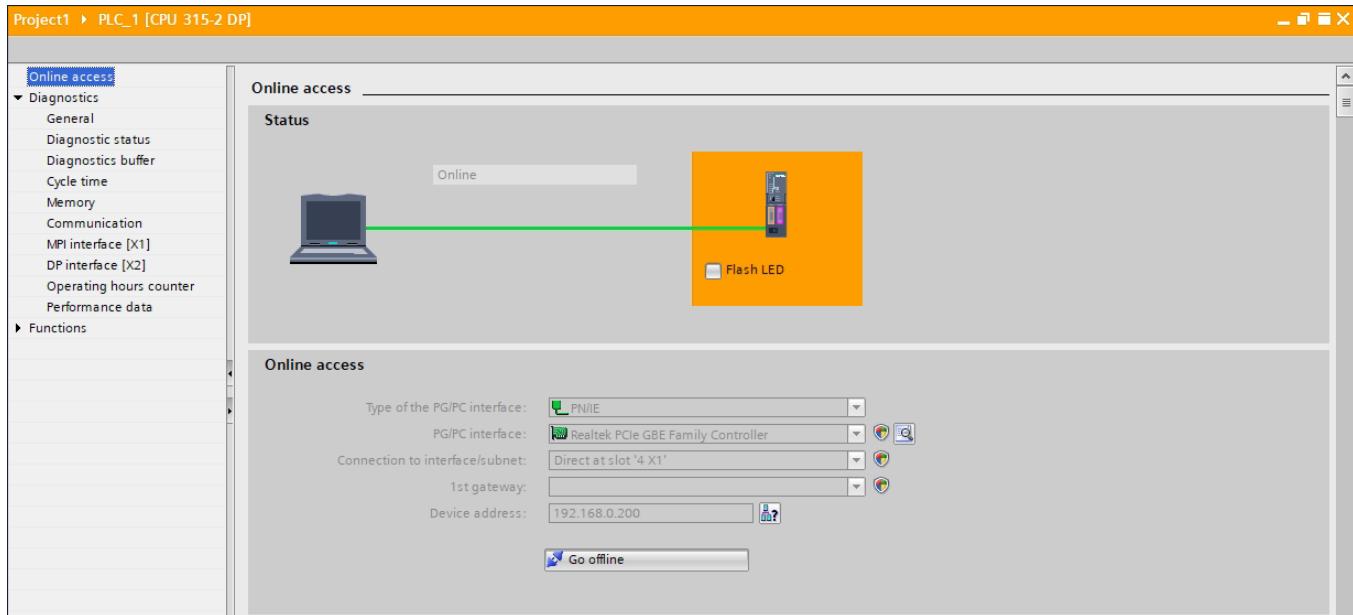
Project tree – Local moduels에서 이더넷 카드를 선택 후 마우스 오른쪽 버튼을 누르면 아래 그림과 같이 팝업메뉴가 표시됩니다. 팝업메뉴에서 Download to device -> software(all)을 선택하면 현재 이더넷 설정된 정보가 PLC에 다운로드가 됩니다.



12. 접속상태를 Project tree – online & diagnostics 항목을 클릭해서 확인합니다.



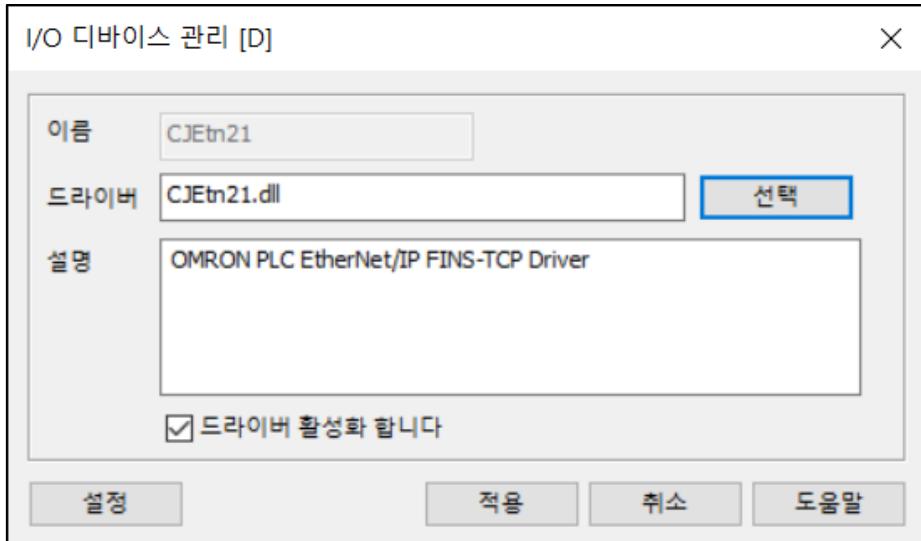
13. 아래 그림과 같이 Online access 항목에 PC 와 PLC 이더넷 카드간의 연결이 online 상태로 표기가 되면 PLC 의 네트워크 셋팅이 정상적으로 되었고 eRun 간 연결할 준비가 된 상태가 됩니다.



OMRON PLC의 EtherNet/IP 프로토콜 지원 유니트 통신 인터페이스 드라이버입니다. OMRON에서 제공하는 FINS/TCP 프로토콜을 사용합니다.

- FINS/TCP 프로토콜 전용
- EIP21 유닛
- NJ CPU 유닛 내장 EtherNet/IP
- 큰 사이즈 데이터를 한번에 전송 가능 (스크립트 이용)
- FINS/TCP는 동시접속 16개까지 접속가능합니다.
- 로그기능

DLL 파일은 CJEt21.DLL 입니다. 선택버튼을 눌러서 드라이버를 선택합니다.



항목	설명
이름	태그에서 사용되는 디바이스 이름을 입력합니다. 영문 또는 숫자로 구성합니다.
드라이버	선택버튼을 눌러서 드라이버 설정합니다. (CJEtn21.dll)
드라이버 활성화	<p>프로젝트 실행할 때 I/O서버에서 구동하기 위해서는 활성화를 합니다. 프로젝트 관리자 I/O 디바이스 팝업메뉴의 "디바이스 활성화"에서 설정해도 동일합니다.</p>
설정	통신속성 및 프레임을 설정합니다.