

**Q** SCADA 화면에서 엑셀 시트를 띄울 수 있나요?

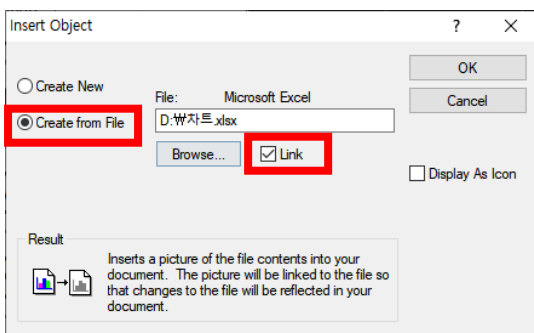
**A** SCADA 기능 중 “개체 삽입” 기능을 이용하면 가능합니다.

1. SCADA 화면에서 띄울 엑셀 파일을 생성합니다.

	A	B	C	D	E	F
1	시간	A1	A2	A3	A4	A5
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						
27						
28						
29						
30						
31						
32	최대값	0	0	0	0	0
33	최소값	0	0	0	0	0
34	평균값	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!

양식을 만들고 이번 FAQ 에서는 시간 및 현재 값을 취득하여 최대, 최소, 평균 값을 구해보도록 하겠습니다.

2. CimonD – 편집 – 개체 삽입에 들어가 엑셀 양식을 등록합니다.





### 3. 데이터베이스에 태그를 등록합니다.

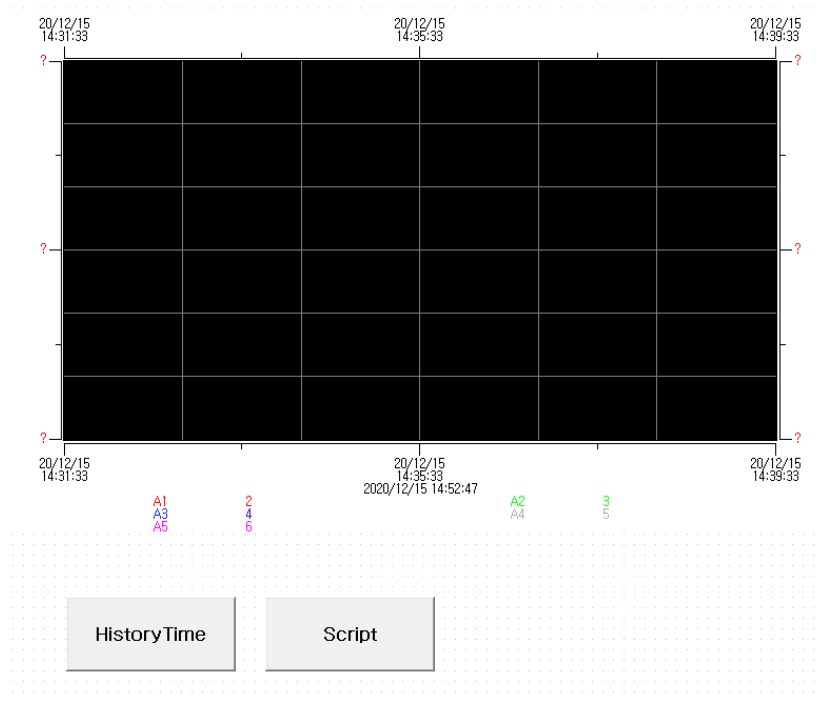
이름	종류	디바이스	어드레스	초기값	부가기능	주석
CELL_NO	아날로그			0		
EXCEL	디지털			0		
A1	아날로그			65535	수집	
A2	아날로그			0	수집	
A3	아날로그			55535	수집	
A4	아날로그			10000	수집	
A5	아날로그			20000	수집	
시작_년	문자열					
시작_월	문자열					
시작_일	문자열					
시작_시	문자열					
시작_분	문자열					
시작_초	문자열					
종료_년	문자열					
종료_월	문자열					
종료_일	문자열					
종료_시	문자열					
종료_분	문자열					
종료_초	문자열					

CELL\_NO 태그는 엑셀의 셀 위치를 확인하기 위한 태그이며, EXCEL 태그는 엑셀 개체 삽입에 등록된 양식의 변화를 SCADA 화면에서 보기 위한 동작 태그입니다.

A1 ~ A5 는 데이터를 취득할 아날로그 태그입니다. 트렌드에서 사용하기 위해 반드시 데이터수집을 설정해야 합니다. 시작\_년부터 종료\_초 태그까지는

“TREND History Mode 에서 임의의 시간을 설정하여 데이터 보는 방법 FAQ”를 참고하시면 됩니다.

### 4. TREND 를 등록하고 Script 실행 버튼, Real, History 버튼 Object 를 등록합니다.





5. Script 를 작성합니다. TREND 를 사용하지 않고 데이터만 보려면 아래의 Script 만 사용하시면 됩니다.

```

1 Sub Script1()
2
3 Set ExcelApp = CreateObject("Excel.Application")
4
5 fFormName$ = "D:\₩차트.xlsx" '양식파일 경로 및 이름
6
7 fTodayName$ = "D:\₩차트.xlsx" '새파일 경로 및 이름
8
9 If (FileExists(fTodayName$) = True) Then
10
11 Set DayRpt = ExcelApp.Workbooks.Open(fTodayName$) '파일을 Open 한다.
12
13 Else
14
15 FileCopy fFormName$ , fTodayName$ '파일 복사하며 생성
16
17 Set DayRpt = ExcelApp.Workbooks.Open(fTodayName$)
18
19 SetTagVal "CELL_NO",1 'Cell 위치 초기화
20
21 End If
22
23 Set Sheet1 = DayRpt.Worksheets(1)
24
25 i = GetTagVal("CELL_NO") 'Cell 태그를 i 변수로 지정
26
27 For i = 2 To 31 'Cell을 2번 Cell부터 31번 Cell까지 반복 실행
28
29 Set Cell = Sheet1.Range("A"+CSTR(i))
30 Cell.Value = Timestr(44)
31 Set Cell = Sheet1.Range("B"+CSTR(i))
32 Cell.Value = GetTagVal("A1")
33 Set Cell = Sheet1.Range("C"+CSTR(i))
34 Cell.Value = GetTagVal("A2")
35 Set Cell = Sheet1.Range("D"+CSTR(i))
36 Cell.Value = GetTagVal("A3")
37 Set Cell = Sheet1.Range("E"+CSTR(i))
38 Cell.Value = GetTagVal("A4")
39 Set Cell = Sheet1.Range("F"+CSTR(i))
40 Cell.Value = GetTagVal("A5")
41
42 Sleep 1000
43
44 Next i
45
46 SetTagVal "CELL_NO", i 'Cell 값을 Cell 태그에 삽입
47
48 If (GetTagVal("CELL_NO") = 32) Then 'Cell이 32가 되면 조건문 실행
49
50 SetTagVal "EXCEL", 1 'EXCEL 비트를 1로하여 수정된 액셀 개체를 적용
51
52 Sleep 1000
53
54 SetTagVal "EXCEL", 0 '다음 동작을 위해 EXCEL 비트를 0으로 변경
55
56 End If
57
58 SetTagVal "CELL_NO",1 '셀을 초기화
59
60 Sheet1.Calculate
61 DayRpt.Save
62 ExcelApp.Quit
63 Set ExcelApp = Empty
64
65 End Sub

```

TREND 와 같이 사용하려면 위의 Script 중간에 새로운 Script 를 추가 작성합니다.

추가 작성하는 Script 는 마찬가지로,

“TREND History Mode 에서 임의의 시간을 설정하여 데이터 보는 방법 FAQ”를 참고하시면 됩니다.

```

1 Sub Script1()
2
3 Dim fVal as Long 'fVal을 Long 데이터(-9,223,372,036,854,775,808~9,223,372,036,854,775,807(8 byte)로 선언
4
5 Set ExcelApp = CreateObject("Excel.Application")
6
7 fFormName$ = "D:\#차트.xlsx" '양식파일 경로 및 이름
8 fTodayName$ = "D:\#차트.xlsx" '새파일 경로 및 이름
9
10 SetTagVal "시작-년", TimeStr(50); SetTagVal "시작-월", TimeStr(51); SetTagVal "시작-일", TimeStr(52)
11 SetTagVal "시작-시", TimeStr(53); SetTagVal "시작-분", TimeStr(54); SetTagVal "시작-초", TimeStr(55)
12
13 sTime$ = Format$(GetTagVal("시작-년"), "0000") 'sTime$를 YYYY 형태로 변환
14 sTime$ = sTime$ & "/" & Format$(GetTagVal("시작-월"), "00") 'sTime$를 YYYY/MM 형태로 변환
15 sTime$ = sTime$ & "/" & Format$(GetTagVal("시작-일"), "00") 'sTime$를 YYYY/MM/DD 형태로 변환
16 sTime$ = sTime$ & ":" & Format$(GetTagVal("시작-시"), "00") 'sTime$를 YYYY/MM/DD HH 형태로 변환
17 sTime$ = sTime$ & ":" & Format$(GetTagVal("시작-분"), "00") 'sTime$를 YYYY/MM/DD HH:MM 형태로 변환
18 sTime$ = sTime$ & ":" & Format$(GetTagVal("시작-초"), "00") 'sTime$를 YYYY/MM/DD HH:MM:SS 형태로 변환
19
20
21 dTime$ = Format$(GetTagVal("시작-년"), "0000") 'dTime$를 YYYY 형태로 변환
22 dTime$ = dTime$ & "-" & Format$(GetTagVal("시작-월"), "00") 'dTime$를 YYYY-MM 형태로 변환
23 dTime$ = dTime$ & "-" & Format$(GetTagVal("시작-일"), "00") 'dTime$를 YYYY-MM-DD 형태로 변환
24 dTime2$ = Format$(GetTagVal("시작-시"), "00") 'dTime2$를 HH 형태로 변환
25 dTime2$ = dTime2$ & ":" & Format$(GetTagVal("시작-분"), "00") 'dTime2$를 HH:MM 형태로 변환
26 dTime2$ = dTime2$ & ":" & Format$(GetTagVal("시작-초"), "00") 'dTime2$를 HH:MM:SS 형태로 변환
27
28
29 If (FileExists(fTodayName$) = True) Then
30 Set DayRpt = ExcelApp.Workbooks.Open(fTodayName$) '파일을 Open 한다.
31 Else
32 FileCopy fFormName$, fTodayName$ '파일이 없으면 양식으로 부터 복사하여 새로운 파일을 생성
33 Set DayRpt = ExcelApp.Workbooks.Open(fTodayName$)
34 SetTagVal "CELL_NO", 1 'Cell 위치 초기화
35 End If
36
37 Set Sheet1 = DayRpt.Worksheets(1)
38
39 i = GetTagVal("CELL_NO") 'Cell 태그를 i 변수로 지정
40
41 For i = 2 To 31 'Cell을 2번 Cell부터 31번 Cell까지 반복 실행
42
43 Set Cell = Sheet1.Range("A"+CSTR(i))
44 Cell.Value = TimeStr(44)
45 Set Cell = Sheet1.Range("B"+CSTR(i))
46 Cell.Value = GetTagVal("A1")
47 Set Cell = Sheet1.Range("C"+CSTR(i))
48 Cell.Value = GetTagVal("A2")
49 Set Cell = Sheet1.Range("D"+CSTR(i))
50 Cell.Value = GetTagVal("A3")
51 Set Cell = Sheet1.Range("E"+CSTR(i))
52 Cell.Value = GetTagVal("A4")
53 Set Cell = Sheet1.Range("F"+CSTR(i))
54 Cell.Value = GetTagVal("A5")
55
56 Sleep 1000
57
58 Next i
59
60
61
62
63
64
65
66 SetTagVal "종료-년", TimeStr(50); SetTagVal "종료-월", TimeStr(51); SetTagVal "종료-일", TimeStr(52)
67 SetTagVal "종료-시", TimeStr(53); SetTagVal "종료-분", TimeStr(54); SetTagVal "종료-초", TimeStr(55)
68
69 eTime$ = Format$(GetTagVal("종료-년"), "0000") 'eTime$를 YYYY 형태로 변환
70 eTime$ = eTime$ & "-" & Format$(GetTagVal("종료-월"), "00") 'eTime$를 YYYY-MM 형태로 변환
71 eTime$ = eTime$ & "-" & Format$(GetTagVal("종료-일"), "00") 'eTime$를 YYYY-MM-DD 형태로 변환
72 eTime2$ = Format$(GetTagVal("종료-시"), "00") 'eTime2$를 HH 형태로 변환
73 eTime2$ = eTime2$ & ":" & Format$(GetTagVal("종료-분"), "00") 'eTime2$를 HH:MM 형태로 변환
74 eTime2$ = eTime2$ & ":" & Format$(GetTagVal("종료-초"), "00") 'eTime2$를 HH:MM:SS 형태로 변환
75
76 fVal = Datediff("s", dTime$, eTime$) 'eTime$ - dTime$의 값(YYYY-MM-DD)을 초 값으로 반환
77 sVal = Datediff("s", dTime2$, eTime2$) 'eTime2$ - dTime2$의 값(HH:MM:SS)을 초 값으로 반환
78
79 SetTrendMode "TREND", 1 'Historical Mode로 트렌드를 전환
80 TrendSetTime "TREND", 3, fVal + sVal 'Historical 트렌드의 최대감시시간을 초단위 fVal + sVal 값으로 삽입
81 TrendSetTime "TREND", 0, sTime$ 'Historical 트렌드의 시작시간을 (임의의 시간)으로 하고 sTime$(YYYY/MM/DD HH:MM:SS)형태로 삽입
82
83 SetTagVal "CELL_NO", i
84
85 If (GetTagVal("CELL_NO") = 32) Then 'Cell이 32가 되면 조건문 실행
86 SetTagVal "EXCEL", 1 'EXCEL 비트를 1로하여 수정된 액셀 개체를 적용
87
88 Sleep 1000
89
90 SetTagVal "EXCEL", 0 '다음 동작을 위해 EXCEL 비트를 0으로 변경
91
92 End If
93
94 SetTagVal "CELL_NO", 1 '셀을 초기화
95
96 Sheet1.Calculate
97 DayRpt.Save
98 ExcelApp.Quit
99 Set ExcelApp = Empty
100
101 End Sub

```

```

1 Sub Script1()
2
3   Set ExcelApp = CreateObject("Excel.Application")
4
5   fFormName$ = "D:\₩차트.xlsx" '양식파일 경로 및 이름
6
7   fTodayName$ = "D:\₩차트.xlsx" '새파일 경로 및 이름
8
9   If (FileExists(fTodayName$) = True) Then
10
11     Set DayRpt = ExcelApp.Workbooks.Open(fTodayName$) '파일을 Open 한다.
12
13     Else
14
15   FileCopy fFormName$ , fTodayName$ '파일 복사하며 생성
16
17   Set DayRpt = ExcelApp.Workbooks.Open(fTodayName$)
18
19   SetTagVal "CELL_NO",1 'Cell 위치 초기화
20
21   End If
22
23   Set Sheet1 = DayRpt.Worksheets(1)
24
25   i = GetTagVal("CELL_NO") 'Cell 태그를 i 변수로 지정
26
27   For i = 2 To 31 'Cell을 2번 Cell부터 31번 Cell까지 반복 실행
28
29     Set Cell = Sheet1.Range("A"+CSTR(i))
30     Cell.Value = Timestr(44)
31     Set Cell = Sheet1.Range("B"+CSTR(i))
32     Cell.Value = GetTagVal("A1")
33     Set Cell = Sheet1.Range("C"+CSTR(i))
34     Cell.Value = GetTagVal("A2")
35     Set Cell = Sheet1.Range("D"+CSTR(i))
36     Cell.Value = GetTagVal("A3")
37     Set Cell = Sheet1.Range("E"+CSTR(i))
38     Cell.Value = GetTagVal("A4")
39     Set Cell = Sheet1.Range("F"+CSTR(i))
40     Cell.Value = GetTagVal("A5")
41
42     Sleep 1000
43
44   Next i
45
46   SetTagVal "CELL_NO", i 'Cell 값을 Cell 태그에 삽입
47
48   If (GetTagVal("CELL_NO") = 32) Then 'Cell이 32가 되면 조건문 실행
49
50     SetTagVal "EXCEL", 1 'EXCEL 비트를 1로하여 수정된 엑셀 개체를 적용
51
52     Sleep 1000
53
54     SetTagVal "EXCEL", 0 '다음 동작을 위해 EXCEL 비트를 0으로 변경
55
56   End If
57
58   SetTagVal "CELL_NO",1 '셀을 초기화
59
60   Sheet1.Calculate
61   DayRpt.Save
62   ExcelApp.Quit
63   Set ExcelApp = Empty
64
65 End Sub

```

위의 Script 는 SCADA 에서 엑셀에 데이터를 넣고 저장하는 Script 입니다.

48 번 줄은 셀이 32 셀까지 도달하면 EXCEL 비트를 살려 개체 삽입에 넣은 엑셀 파일을 동작 시키는 조건 문입니다.

추가 Script 는 TREND History Mode 에서 임의의 시간을 설정하여 데이터를 확인하기 위하여 시간 형식과 TREND 최대 감시 시간을 표현하기 위한 Script 입니다.

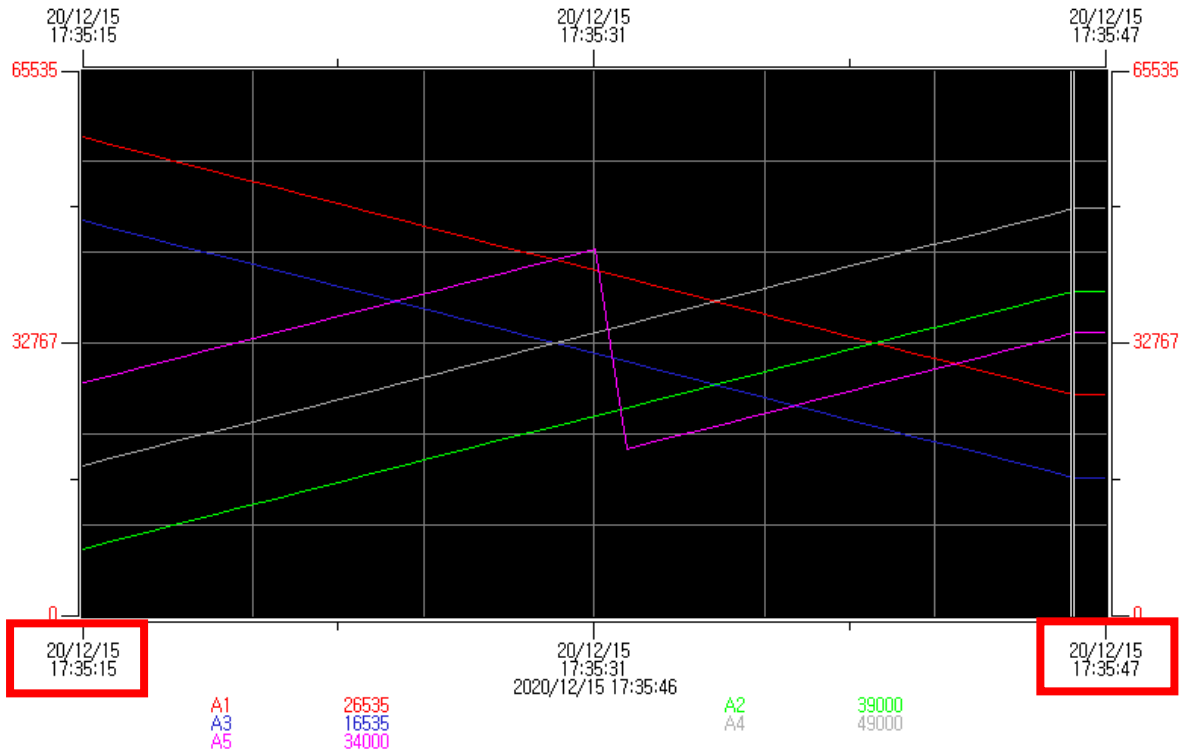
CimonX 를 실행하고 Script 를 실행합니다.

PAGE 1

시간	A1	A2	A3	A4	A5
2020년12월15일 17시35분15초	57535	8000	47535	18000	28000
2020년12월15일 17시35분17초	55535	10000	45535	20000	30000
2020년12월15일 17시35분18초	54535	11000	44535	21000	31000
2020년12월15일 17시35분19초	53535	12000	43535	22000	32000
2020년12월15일 17시35분20초	52535	13000	42535	23000	33000
2020년12월15일 17시35분21초	51535	14000	41535	24000	34000
2020년12월15일 17시35분22초	50535	15000	40535	25000	35000
2020년12월15일 17시35분23초	49535	16000	39535	26000	36000
2020년12월15일 17시35분24초	48535	17000	38535	27000	37000
2020년12월15일 17시35분25초	47535	18000	37535	28000	38000
2020년12월15일 17시35분26초	46535	19000	36535	29000	39000
2020년12월15일 17시35분27초	45535	20000	35535	30000	40000
2020년12월15일 17시35분28초	44535	21000	34535	31000	41000
2020년12월15일 17시35분29초	43535	22000	33535	32000	42000
2020년12월15일 17시35분30초	42535	23000	32535	33000	43000
2020년12월15일 17시35분31초	41535	24000	31535	34000	44000
2020년12월15일 17시35분32초	40535	25000	30535	35000	20000
2020년12월15일 17시35분33초	39535	26000	29535	36000	21000
2020년12월15일 17시35분34초	38535	27000	28535	37000	22000
2020년12월15일 17시35분35초	37535	28000	27535	38000	23000
2020년12월15일 17시35분36초	36535	29000	26535	39000	24000
2020년12월15일 17시35분37초	35535	30000	25535	40000	25000
2020년12월15일 17시35분38초	34535	31000	24535	41000	26000
2020년12월15일 17시35분39초	33535	32000	23535	42000	27000
2020년12월15일 17시35분40초	32535	33000	22535	43000	28000
2020년12월15일 17시35분41초	31535	34000	21535	44000	29000
2020년12월15일 17시35분42초	30535	35000	20535	45000	30000
2020년12월15일 17시35분43초	29535	36000	19535	46000	31000
2020년12월15일 17시35분44초	28535	37000	18535	47000	32000
2020년12월15일 17시35분45초	27535	38000	17535	48000	33000
<b>최대값</b>	<b>57535</b>	<b>38000</b>	<b>47535</b>	<b>48000</b>	<b>44000</b>
<b>최소값</b>	<b>27535</b>	<b>8000</b>	<b>17535</b>	<b>18000</b>	<b>20000</b>
<b>평균값</b>	<b>42068.33</b>	<b>23466.67</b>	<b>32068.33</b>	<b>33466.67</b>	<b>31800</b>

Script 를 실행하고 화면을 나눠서 올렸습니다. 위의 사진은 엑셀에 데이터를 넣고 저장한 파일을 SCADA 화면으로 불러온 화면입니다.





위의 트렌드는 엑셀 양식에서 수집된 데이터를 가져와 History TREND 에 뿌린 화면입니다.

위와 같이 개체 삽입 기능을 이용하면 엑셀 양식을 SCADA 화면에서 구현 가능합니다.