Beschreibung: SIE_Logo_Layer_Petrol_RGB_A4_56mm

**SCE 교육 커리큘럼**

Siemens Automation Cooperates with Education | 05/2017

TIA Portal Module 032-600  
글로벌 데이터 블록

교육 커리큘럼에 따른 적합한 SCE 트레이너 패키지

**SIMATIC 컨트롤러**

* **SIMATIC ET 200SP Open Controller CPU 1515SP PC F 및 HMI RT SW**주문 번호: 6ES7677-2FA41-4AB1
* **SIMATIC ET 200SP Distributed Controller CPU 1512SP F-1 PN Safety**

주문 번호: 6ES7512-1SK00-4AB2

* **SIMATIC CPU 1516F PN/DP Safety**

주문 번호: 6ES7516-3FN00-4AB2

* **SIMATIC S7 CPU 1516-3 PN/DP**주문 번호: 6ES7516-3AN00-4AB3
* **SIMATIC CPU 1512C PN(소프트웨어 장착) 및 PM 1507**주문 번호: 6ES7512-1CK00-4AB1
* **SIMATIC CPU 1512C PN(소프트웨어 장착), PM 1507 및 CP 1542-5 (PROFIBUS)**주문 번호: 6ES7512-1CK00-4AB2
* **SIMATIC CPU 1512C PN(소프트웨어 장착)**주문 번호: 6ES7512-1CK00-4AB6
* **SIMATIC CPU 1512C PN(소프트웨어 장착) 및 CP 1542-5 (PROFIBUS)**주문 번호: 6ES7512-1CK00-4AB7

**교육용 SIMATIC STEP 7 소프트웨어**

* **SIMATIC STEP 7 Professional V14 SP1 - 단일 라이센스**

주문 번호: 6ES7822-1AA04-4YA5

* **SIMATIC STEP 7 Professional V14 SP1 - 강의실 라이센스 (최대 인원 6명)**

주문 번호: 6ES7822-1BA04-4YA5

* **SIMATIC STEP 7 Professional V14 SP1 - 업그레이드 라이센스 (최대 인원 6명)**

주문 번호: 6ES7822-1AA04-4YE5

* **SIMATIC STEP 7 Professional V14 SP1 - 학생 라이센스 (최대 인원 20명)**

주문 번호: 6ES7822-1AC04-4YA5

위 트레이너 패키지는 필요 시 후속 모델 패키지로 대체가 된다는 점에 유의하십시오. 현재 출시된 SCE 패키지에 대한 개요는 [siemens.com/sce/tp](http://www.siemens.com/sce/tp)에서 제공됩니다.

보충 교육

지멘스의 지역별 SCE 보충 교육에 대한 내용은 해당 지역의 SCE 고객 센터로 문의하시기 바랍니다.

[siemens.com/sce/contact](http://www.siemens.com/sce/contact)

SCE 관련 추가 정보

[siemens.com/sce](http://www.siemens.com/sce)

사용 관련 정보

통합 자동화 솔루션인 TIA(Totally Integrated Automation)를 위한 SCE 교육 커리큘럼은 공교육 시설 및 R&D 기관 교육 목적의 "SCE(Siemens Automation Cooperates with Education) 프로그램을 위해 마련된 것입니다. Siemens AG는 프로그램의 내용을 보증하지 않습니다.

본 문서는 지멘스 제품/시스템을 초기 교육하는 용도로만 사용되어야 합니다. 따라서 교육 범위 내에서의 사용 목적으로 전체 또는 일부를 복사하여 교육생들에게 제공할 수 있습니다. 본 문서는 공공 교육 및 고등 교육 시설 내에서의 교육을 위한 목적으로의 배포, 복사 및 내용의 공유가 가능합니다.

예외적인 경우에는 Siemens AG 담당자의 서면 동의가 필요합니다. Roland Scheuerer

[roland.scheuerer@siemens.com](mailto:roland.scheuerer@siemens.com).

해당 규정의 위반 시에는 그에 대한 책임이 부과될 수 있습니다. 특히 특허가 부여되었거나 실용신안 또는 의장등록이 된 경우, 번역을 포함한 제반 권리는 지멘스의 소유입니다.

산업체 고객을 위한 교육 과정의 사용은 명시적으로 금지됩니다. 지멘스는 교육 커리큘럼의 상업적 이용을 거부합니다.

드레스덴공대(TU Dresden), 특히 공학 박사 Leon Urbas 교수와 Michael Dziallas Engineering Corporation, 그리고 본 교육 커리큘럼을 준비하는 과정에서 도움을 주신 모든 관계자들께 감사의 말씀을 전합니다.

목차

TOC

SIMATIC S7-1500을 위한 글로벌 데이터 블록

# 1 목표

이 챕터에서는 TIA Portal 프로그래밍 툴을 통해 SIMATIC S7-1500에서 글로벌 데이터 블록을 사용하는 방법에 대해 배워보겠습니다.

이 모듈에는 글로벌 데이터 블록의 구조와 SIMATIC S7-1500에서 글로벌 데이터 블록을 생성하고 액세스하는 방법이 설명되어 있습니다. 또한, TIA Portal에서 글로벌 데이터 블록을 생성하고 프로그램에서 이를 읽기/쓰기 액세스하기 위한 단계들이 나와 있습니다.

제3장에 기술된 SIMATIC S7 제어 장치를 사용할 수 있습니다.

# 2 전제 조건

SIMATIC S7 CPU1516F-3 PN/DP에서 챕터 "아날로그 값"에서 학습한 내용을 토대로 합니다.   
이 챕터에서는 예를 들어 SCE\_EN\_032-500\_Analog\_Values\_R1508.zap13 같은 프로젝트를 이용할 수 있습니다.

# 3 필요한 하드웨어 및 소프트웨어

**1** 엔지니어링 스테이션: 하드웨어 및 운영 시스템이 필요합니다(자세한 정보는 TIA 포털의 설치 DVD Readme/Liesmich를 참조하세요).

**2** TIA 포털의 소프트웨어 SIMATIC STEP 7 Professional – V13부터

3 SIMATIC S7-1500/S7-1200/S7-300 제어 장치, 예: CPU 1516F-3 PN/DP – 펌웨어 버전 V1.6 이상, 메모리 카드와 16DI/16DO 및 2AI/1AO 포함  
참고: 디지털 입력과 아날로그 입력 및 출력은 컨트롤 패널에서 실행해야 합니다.

**4** 엔지니어링 스테이션과 제어 장치 간 이더넷 연결



**2** SIMATIC STEP 7   
Professional(TIA 포털), V13 이상



**1** 엔지니어링 스테이션

**4** 이더넷 연결



컨트롤 패널

****

**3** SIMATIC S7 제어 장치

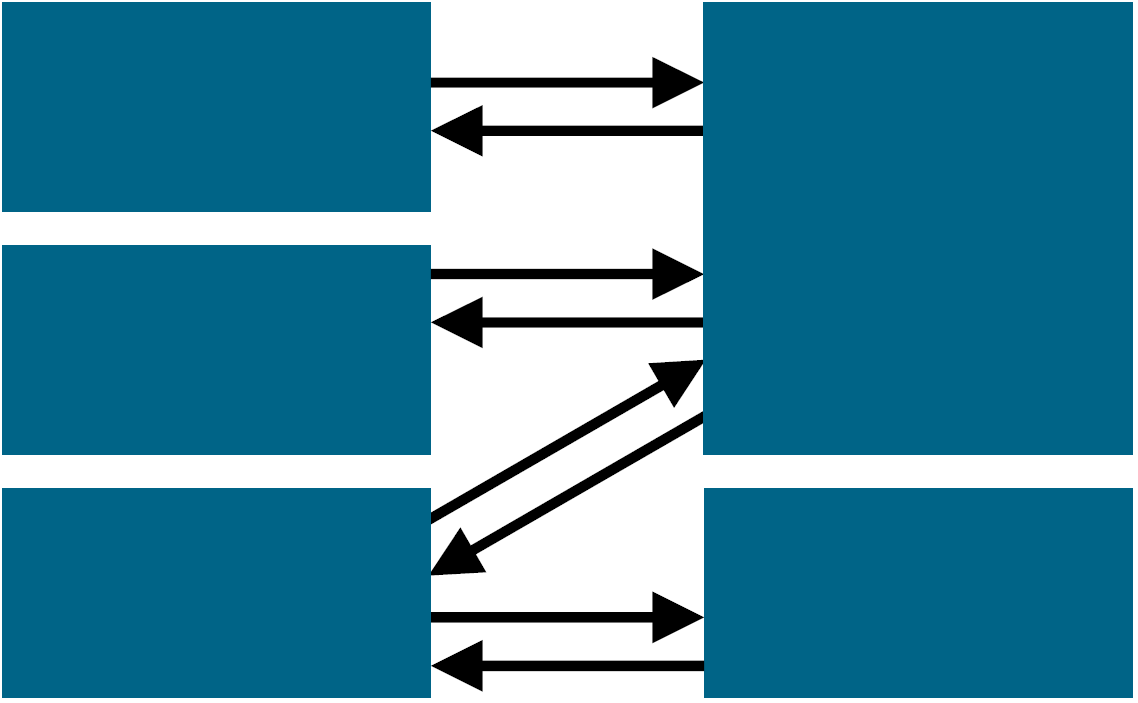
# 4 이론

## 4.1 데이터 블록

로직 블록과 대조적으로 데이터 블록에는 명령어가 포함되어 있지 않습니다. 그 보다는 사용자 데이터를 위한 메모리 역할을 합니다.

따라서 데이터 블록에는 사용자 프로그램에서 사용되는 가변 데이터를 저장합니다. 필요에 따라 글로벌 데이터 블록의 구조를 정의할 수 있습니다.

글로벌 데이터 블록에는 *기타 모든 블록들*에서 사용이 가능한 데이터가 저장되어 있습니다(그림 1 참조). 인스턴스 데이터 블록은 관련된 펑션 블록에서만 액세스해야합니다. 데이터 블록의 최대 크기는 CPU에 따라 다릅니다.



인스턴스 DB (DB\_Instance)

펑션 data block\_12에만 액세스

모든 블록에 액세스

글로벌 DB (DB\_Global)

Function\_  
block\_12

Function\_11

Function\_10

그림 1: 글로벌 DB와 인스턴스 DB 간의 차이

글로벌 데이터 블록의 애플리케이션 예는 다음과 같습니다.

- 스토리지 시스템에 대한 정보 저장 (예: "어떤 제품이 어디에 위치해 있는가?")

- 특정 제품에 대한 레시피 저장

대부분의 경우 데이터 블록의 데이터는 안전하게 저장됩니다. 따라서 정전이 발생한 경우나 CPU의 STOP/START 이후에도 데이터가 유지됩니다.

## 4.2 SIMATIC S7-1500의 데이터 타입

SIMATIC S7-1500은 다양한 숫자 형식을 표현할 수 있도록 다양한 유형의 데이터를 지원합니다.   
몇 가지 기본적인 데이터 타입의 목록이 아래 나와 있습니다.

|  |  |  |  |
| --- | --- | --- | --- |
| 데이터 타입 | 크기 (비트) | 범위 | 상수 입력의 예 |
| Bool | 1 | 0~1 | TRUE, FALSE, O, 1 |
| Byte | 8 | 16#00 ~ 16#FF | 16#12, 16#AB |
| Word | 16 | 16#0000 ~ 16#FFFF | 16#ABCD, 16#0001 |
| DWord | 32 | 16#00000000 ~ 16#FFFFFFFF | 16#02468ACE |
| Char | 8 | 16#00 ~ 16#FF | 'A', ‘r’, ‘@’ |
| Sint | 8 | -128~127 | 123,-123 |
| Int | 16 | -32,768 ~ 32,767 | 123, -123 |
| Dint | 32 | -2,147,483,648 ~ 2,147,483,647 | 123, -123 |
| USInt | 8 | 0 ~ 255 | 123 |
| Ulnt | 16 | 0 ~ 65,535 | 123 |
| UDInt | 32 | 0 ~ 4,294,967,295 | 123 |
| Real | 32 | +/-1.18 x 10 -38 ~ +/-3.40 x 10 38 | 123.456, -3.4, 1.2E+123.4E-3 |
| LReal | 64 | +/-2.23 x 10 -308 ~ +/-1.79 x 10 308 | 12345.123456789-1.2E+40 |
| Time | 32 | T#-24d\_20h\_31 m\_23s\_648ms ~ T#24d\_20h\_31 m\_23s\_647ms  다음과 같이 저장됨: -2,147,483,648 ms ~ +2,147,483,647 ms | T#5m\_30s  5#-2d  T#1d\_2h\_15m\_30x\_45ms |
| String | 가변적 | 0 ~ 254개 문자 (바이트 크기) | 'ABC' |
| ARRAY |  | 어레이는 동일한 유형의 데이터가 차례로 정렬이 되어 주소 영역에서 연속적으로 어드레스를 할당. 각 어레이 요소의 속성은 동일하며 어레이 태그에서 구성. |  |
| STRUCT |  | STRUCT는 서로 다른 데이터 타입으로 구성된 다수의 데이터를 포함한 데이터 구조를 표현. 데이터 타입이 STRUCT 또는 ARRAY인 컴포넌트는 중첩 구조 가능. |  |
| 기타 |  | 기타 데이터 타입은 온라인 도움말을 참조 |  |

## 4.3 최적화된 블록

S7-1500 컨트롤러는 최적화된 데이터 스토리지를 가지고 있습니다. 최적화된 블록에서는 모든 태그가 데이터 타입에 따라 자동으로 저장됩니다. 분류는 태그들 간의 데이터 격차를 최소화하고 컨트롤러에 의한 액세스를 최적화하는 방식으로 태그를 저장할 수 있도록 해줍니다.

* 시스템에 의한 파일 저장은 최적화된 방식에 따라 선언과는 별개로 이루어지기 때문에 태그는 언제든 최대한 빠른 속도로 액세스할 수 있습니다.
* 심볼 액세스가 주로 사용되기 때문에 올바르지 않은 직접 액세스로 인한 불일치가 발생할 위험이 없습니다.
* 예를 들어 프로세스 시각화 시스템에 의한 액세스가 심볼 방식으로 이루어지기 때문에 선언을 변경해도 액세스 오류가 일어나지 않습니다.
* 선택적으로 각각의 태그를 보존형으로 정의할 수 있습니다.
* 인스턴스 데이터 블록에서 설정이 필요하지 않거나 가능하지 않습니다. 모든 것이 할당된 FB에 설정이 됩니다 (예: 보존력).
* 데이터 블록에 메모리 공간이 확보되어 있기 때문에 실제 값의 손실 없이 변경을 수행할 수 있습니다 (재초기화 없이 다운로드).

## 4.4 재초기화(Reinitialization) 없이 다운로드

향후, CPU에서 이미 실행 중인 사용자 프로그램을 수정하기 위해 S7-1500 컨트롤러는 작동 중에 최적화된 펑션 또는 데이터 블록의 인터페이스를 확장할 수 있는 옵션을 제공합니다. 컨트롤러를 STOP 모드로 전환하지 않으면서도 이전에 다운로드된 태그의 실제 값에 영향을 미치지 않고 수정된 블록을 다운로드할 수 있습니다.

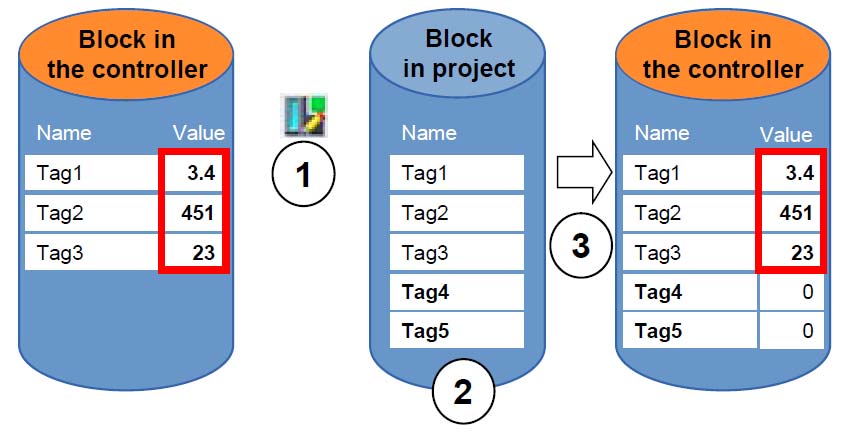


그림 2: 재초기화 없이 다운로드

컨트롤러가 RUN 모드에 있는 동안 아래와 같은 단계들을 수행할 수 있습니다.

1. "Download without reinitialization"을 활성화

2. 새로 정의된 태그를 기존 블록에 삽입

3. 컨트롤러에 확장된 블록을 다운로드

새로 정의된 태그는 초기화됩니다. 기존 태그는 현재 값을 유지합니다.

전제 조건: 사전에 메모리 공간이 확보된 블록을 CPU에 다운로드해야 합니다.

# 5 과제

이 챕터에서는 중앙에서 "MOTOR\_SPEEDCONTROL“ [FC10] 및 "MOTOR\_ SPEEDMONITORING“ [FC11] 펑션에 파라미터를 제공하는 데이터 블록을 포함시키도록 챕터 "SCE\_EN\_032-500 Analog Values“에서 생성된 프로그램을 확장해볼 것입니다.

# 6 계획 수립

"MOTOR\_SPEEDCONTROL" [FC10] 및 "MOTOR\_SPEEDMONITORING" [FC11] 펑션을 위한 데이터 관리 및 설정값 설정은 글로벌 데이터 블록 "SPEED\_MOTOR" [DB2]를 이용해 수행할 것입니다.

이 기능은 "032-500\_Analog\_Values" 프로젝트에 추가될 것입니다. 사전에 미리 이 프로젝트의 압축을 풀어야 합니다.

그런 다음, "Main" [OB1] 오거나이제이션 블록에서 두 펑션 "MOTOR\_SPEEDCONTROL" [FC10] 및 "MOTOR\_SPEEDMONITORING" [FC11]를 글로벌 데이터 블록 "SPEED\_MOTOR" [DB2]에서 나온 태그들에 연결해야 합니다.

## 6.1 모터의 속도 제어 및 속도 모니터링을 위한 글로벌 데이터 블록

속도 설정값과 실제 속도 값은 "SPEED\_MOTOR" [DB2] 데이터 블록의 첫 태그로서 Real 데이터 타입 (32비트 부동 소수점 수)으로 생성이 됩니다. 속도 설정값은 초기값 + 14 U/min으로 설정할 것입니다.

그런 다음, 양(+)의 속도 제한 값을 모니터링할 수 있도록 구조 (Struct) 'Positive\_Speed'를 생성합니다.

이 구조에는 Real 데이터 타입 (32비트 부동 소수점 수)의 두 태그 'Threshold\_Error' (초기값 + 15 rpm) 및 'Threshold\_Warning' (초기값 + 10 rpm)와 Bool 데이터 타입 (이진수)의 두 태그 'Error' 및 'Warning'가 포함되어 있습니다.

구조 (Struct) 'Positive\_Speed'의 복사본을 추가하고, 음(-)의 속도 한계를 모니터링할 수 있도록 'Negative\_Speed'으로 이름이 변경합니다.

초기값 - 16 rpm을 'Threshold\_Error' 태그에, 초기값 - 14 rpm를 'Threshold\_Warning' 태그에 할당합니다.

## 6.2 기술 다이어그램

아래에 과제를 위한 기술 다이어그램이 나와 있습니다.

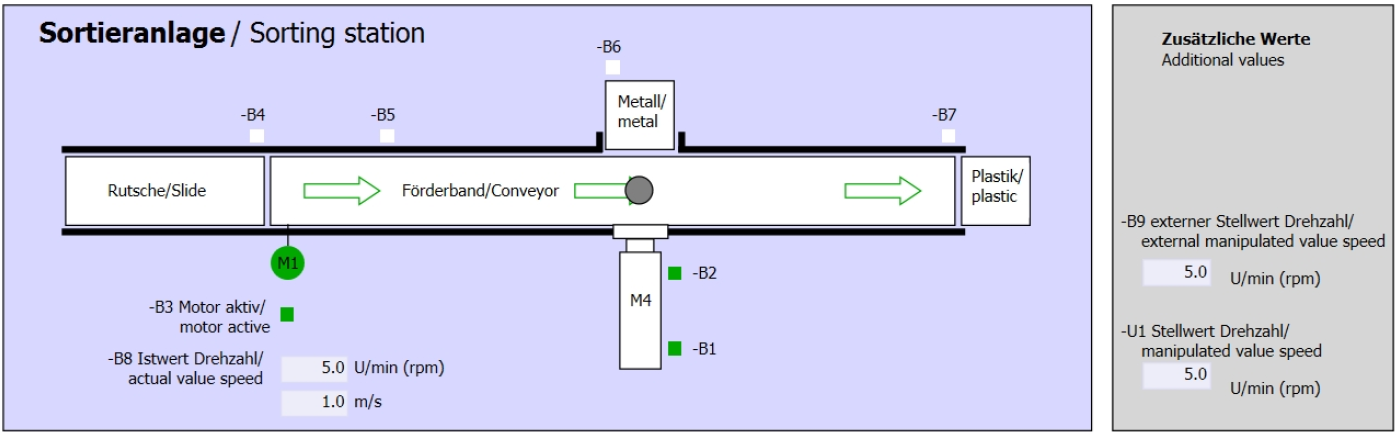


그림 3: 기술 다이어그램

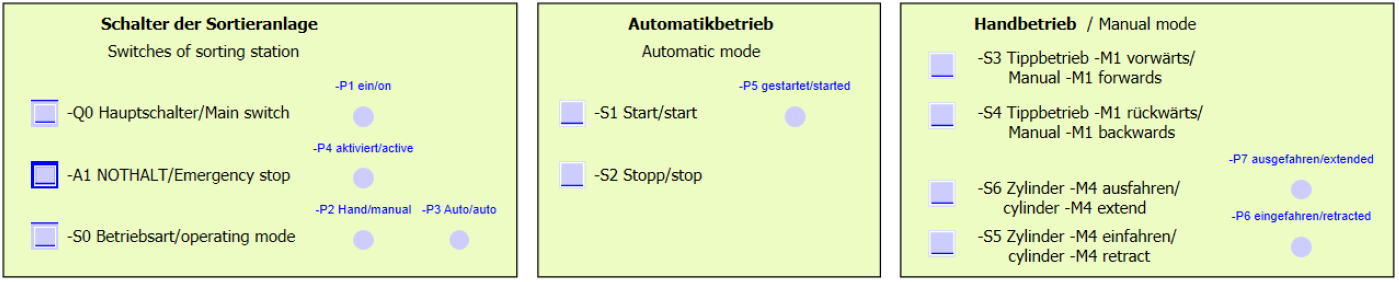


그림 4: 제어 패널

## 6.3 참조 목록

이 과제를 위한 글로벌 오퍼랜드로서 아래와 같은 신호들이 필요합니다.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DI | 유형 | 식별자 | 기능 | NC/NO |
| I 0.0 | BOOL | -A1 | 반환 신호 비상 정지 OK | NC |
| I 0.1 | BOOL | -K0 | 메인 스위치 "온" | NO |
| I 0.2 | BOOL | -S0 | 모드 선택 수동 (0) / 자동 (1) | 수동 = 0  자동 = 1 |
| I 0.3 | BOOL | -S1 | 푸시버튼 자동 시작 | NO |
| I 0.4 | BOOL | -S2 | 푸시버튼 자동 정지 | NC |
| I 0.5 | BOOL | -B1 | 센서 실린더 -M4 복귀 | NO |
| I 1.0 | BOOL | -B4 | 슬라이드의 센서 부분 | NO |
| I 1.3 | BOOL | -B7 | 컨베이어 끝 센서 부분 | NO |
| IW64 | BOOL | -B8 | 센서 실제 속도 값 +/- 10V는 +/- 50 rpm에 해당 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DO | 유형 | 식별자 | 기능 |  |
| Q 0.2 | BOOL | -Q3 | 컨베이어 모터 -M1 가변 속도 |  |
| QW 64 | BOOL | -U1 | 2방향 모터의 조작 속도 값 +/- 10V는 +/- 50 rpm에 해당 |  |

*참조 목록 범례*

DI Digital Input DO Digital Output

AI Analog Input AO Analog Output

I Input Q Output

NC Normally Closed

NO Normally Open

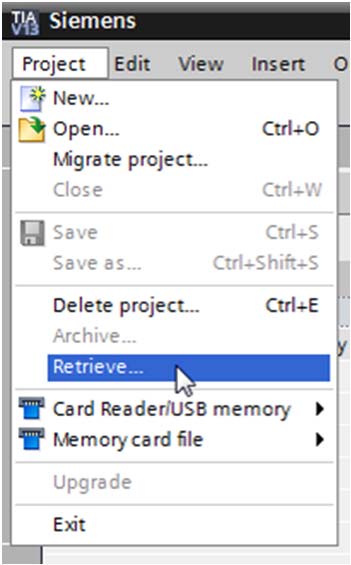
# 7 단계별 따라 해보기

아래에는 계획을 수립하는 방법에 대한 지침이 나와 있습니다. 모든 내용을 이미 충분히 숙지했다면 숫자가 표시된 단계로 넘어가도 좋습니다. 그렇지 않다면 아래에 나와 있는 지침의 단계를 따라가면 됩니다.

## 7.1 기존 프로젝트 압축풀기

→ 챕터 "SCE\_EN\_032-500 Analog Values"에서 생성된 "SCE\_EN\_032-500\_Analog\_Values\_R1508.zap13" 프로젝트를 확장할 수 있으려면 먼저 이 프로젝트의 압축을 풀어야 합니다. 아카이브된 기존 프로젝트의 압축을 풀려면 “Project”의 “Retrieve”로 가서 해당되는 아카이브를 선택해야 합니다. “Open”을 클릭해 선택합니다.

(→ Project → Retrieve → .zap 아카이브 선택 → Open)

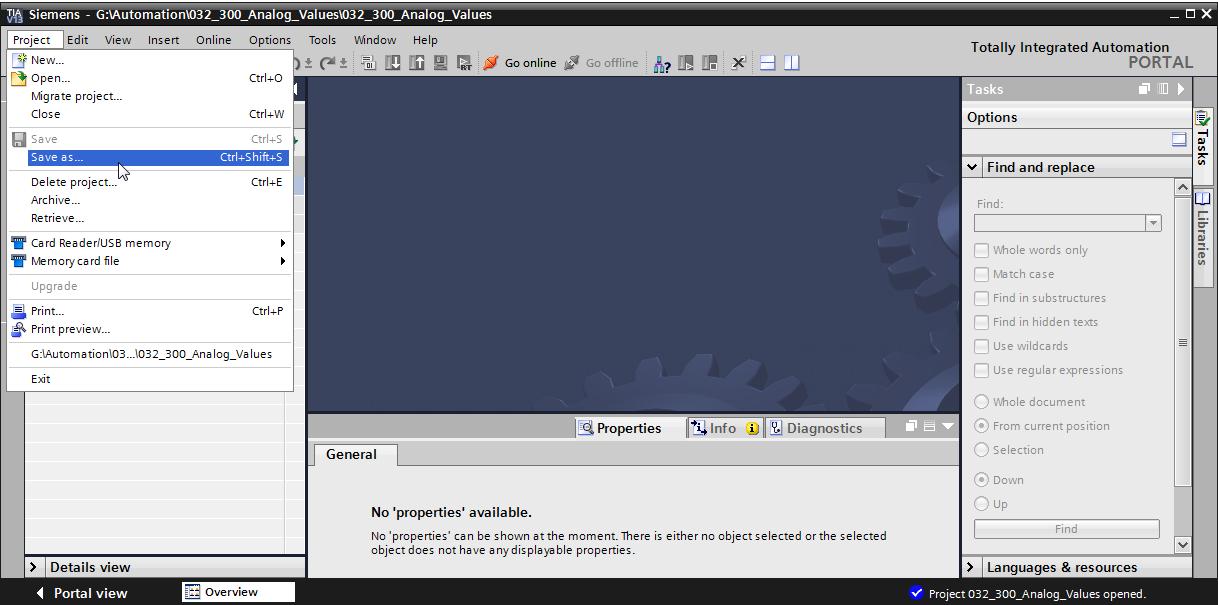


→ 그 다음으로 이 프로젝트가 저장될 대상 디렉토리를 선택합니다. "OK"를 눌러 선택합니다.

(→ Target directory → OK)

→ 열린 프로젝트를 032-600\_Global\_Data\_Blocks라는 이름으로 저장을 합니다.

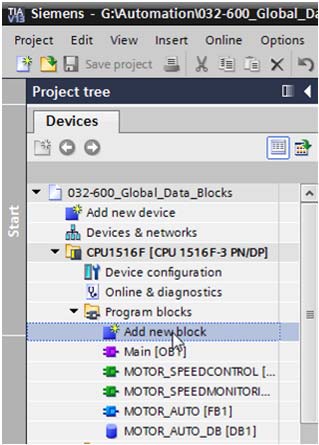
(→ Project → Save as… → 032-600\_Global\_Data\_Blocks → Save)



## 7.2 글로벌 데이터 블록 "SPEED\_MOTOR" 생성

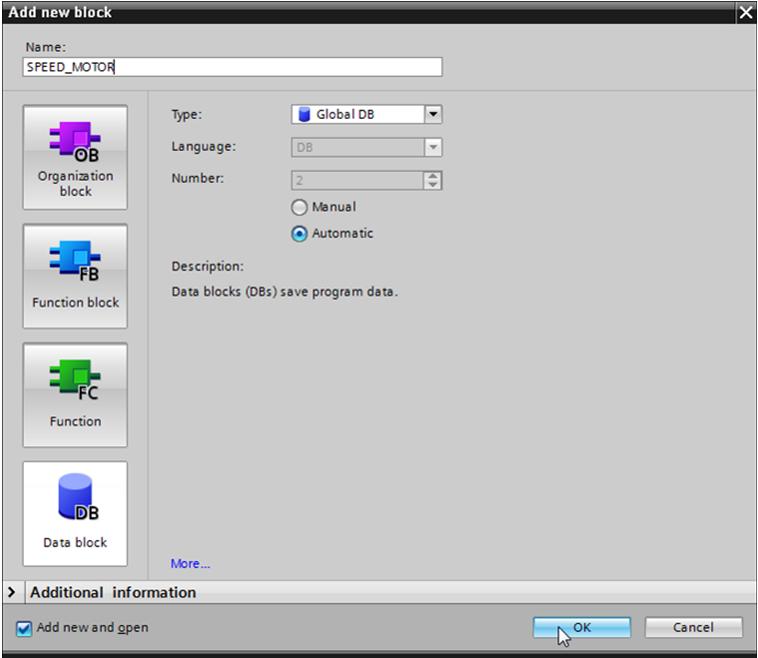
→ CPU 1516F-3 PN/DP의 '프로그램 블록' 폴더를 선택한 다음, "Add new block"를 클릭해 새로운 글로벌 데이터 블록을 생성합니다.

(→ CPU\_1516F [CPU 1516F-3 PN/DP] → Add new block)



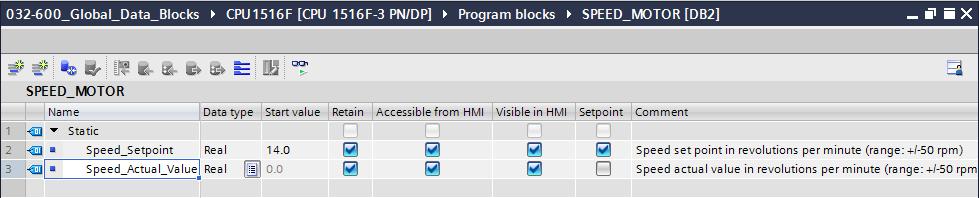
→ 다음 대화상자에서 E:\#--인터스피어코리아\0617_지멘스\images\img3192.jpg를 선택하고 새 블록의 이름을 "SPEED\_MOTOR"으로 변경합니다. 타입은 "Global DB"로 선택합니다. 숫자 '2'가 자동으로 지정됩니다. "Add new and open" 체크박스를 선택합니다. 'OK'를 클릭합니다.

(→ E:\#--인터스피어코리아\0617_지멘스\images\img3192.jpg → Name: SPEED\_MOTOR → Type: Global DB → E:\#--인터스피어코리아\0617_지멘스\images\img3211.jpg Add new and open → OK)



→ "SPEED\_MOTOR" 데이터 블록이 자동으로 표시됩니다. 아래와 같이 태그 'Speed\_Setpoint' 및 'Speed\_Actual\_Value'를 코멘트를 포함하여 생성합니다. 데이터 타입으로 'Real'을 선택합니다. 또한 'Speed\_Setpoint'의 초기값을 14.0 rpm으로 설정합니다.

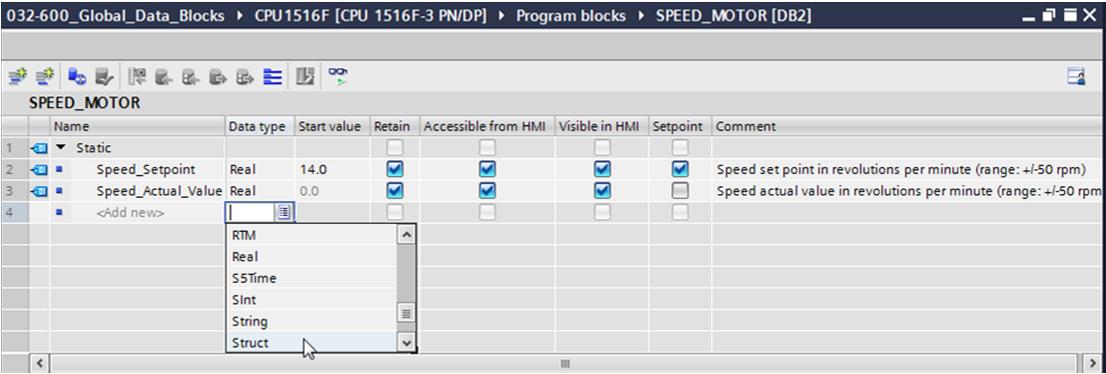
(→ Speed\_Setpoint → Real → 14.0 → Speed\_Actual\_Value → Real)



*참고: 올바른 데이터 타입을 사용해야 합니다.*

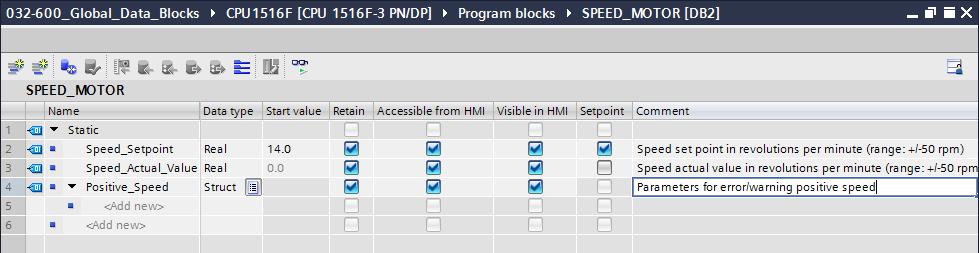
→ 다음으로 이후에 반복하여 사용이 가능하도록 태그 구조 'Struct'를 생성합니다.

(→ Struct)

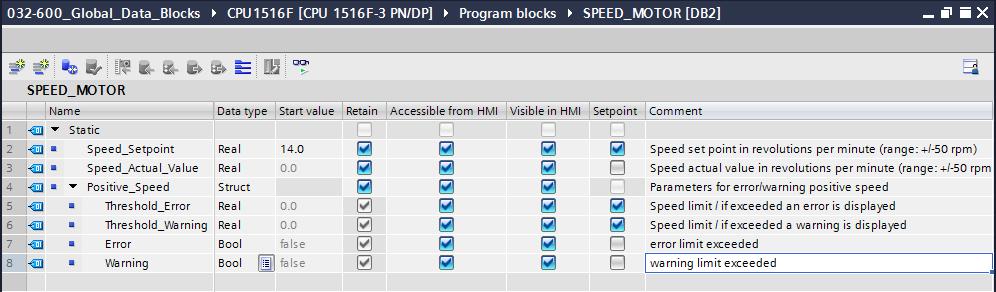


→ 구조를 'Positive\_Speed'라고 명명하고 코멘트를 입력합니다.

(→ Positive\_Speed)



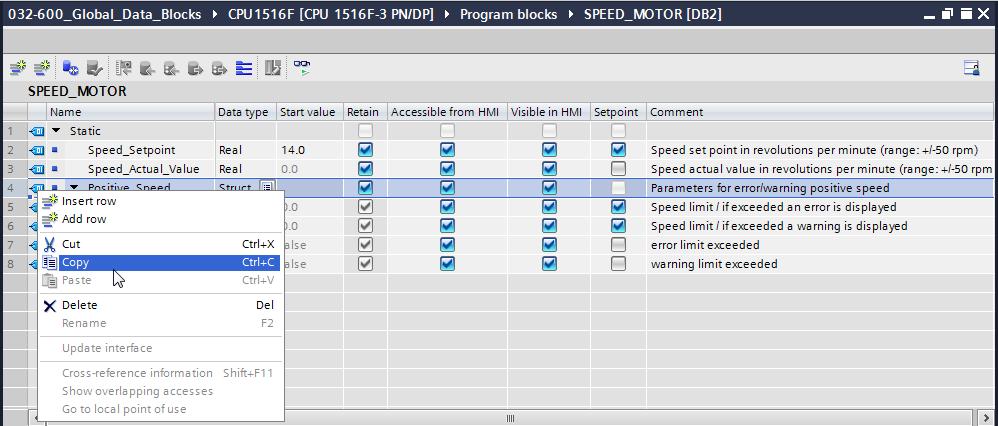
→ 아래와 같이 'Positive\_Speed'에 속도 모니터링을 위한 태그를 생성합니다.



*참고: 올바른 데이터 타입을 사용해야 합니다.*

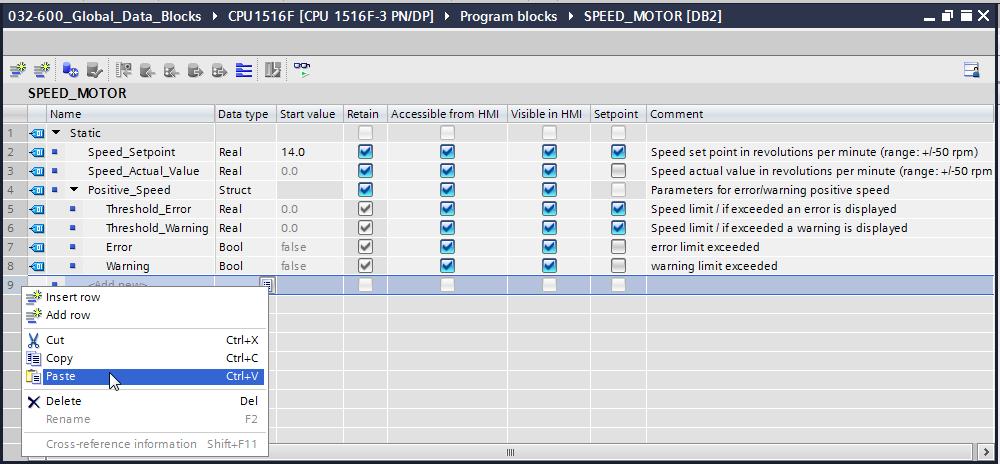
→ 그런 다음 'Positive\_Speed' 구조를 선택하고 복사를 합니다.

(→ Copy)



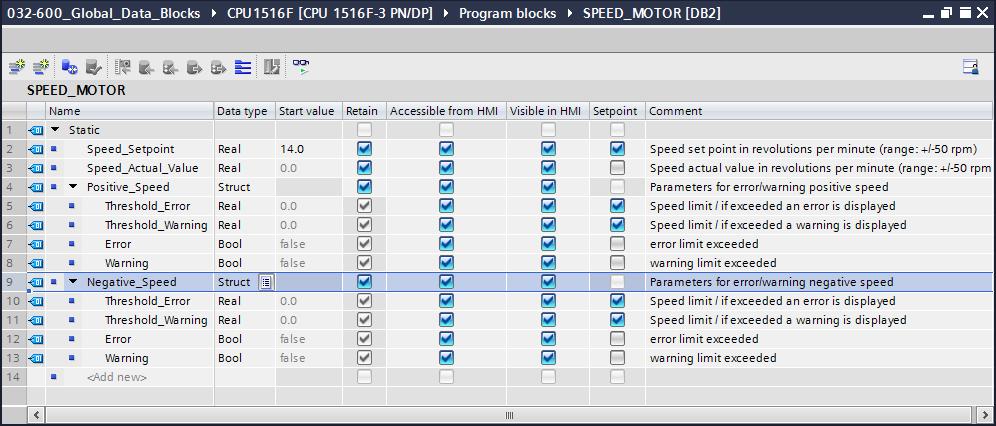
→ 복사된 구조체를 'Positive\_Speed' 구조체 아래에 다시 붙여 넣습니다.

(→ Paste)



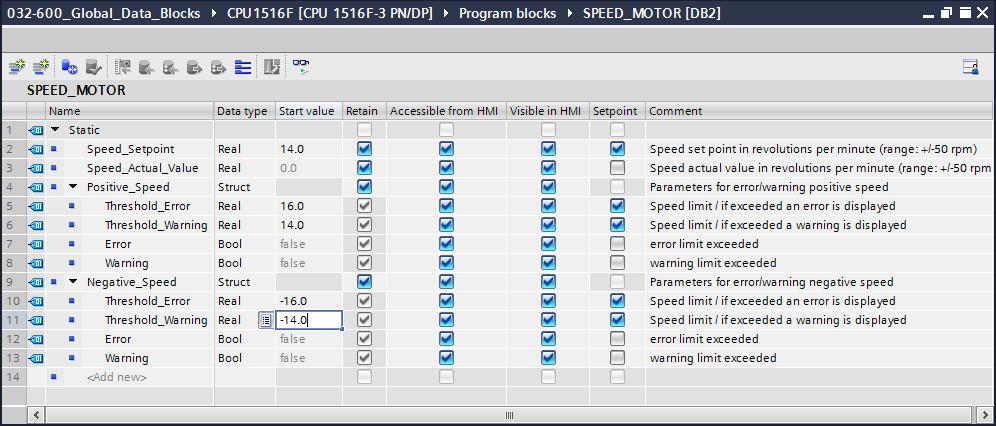
→ 새 구조체의 이름을 'Negative\_Speed'로 변경하고 코멘트를 입력합니다.

(→ Negative\_Speed)



E:\#--인터스피어코리아\0617_지멘스\images\img3063.jpg를 클릭하는 것을 잊지 않도록 합니다. 완성된 글로벌 데이터 블록 "SPEED\_MOTOR" [DB2]이 아래에 표시됩니다. "Retain(유지)" E:\#--인터스피어코리아\0617_지멘스\images\img3211.jpg이 선택되었는지, 그리고 해당 초기값이 모든 태그에 입력되었는지 확인합니다. 이에 따라, 정전이 발생한 경우나 CPU의 STOP/START 이후에도 데이터가 유지됩니다. Accessible from HMI' E:\#--인터스피어코리아\0617_지멘스\images\img3211.jpg 및 'Visible in HMI' E:\#--인터스피어코리아\0617_지멘스\images\img3211.jpg 체크박스를 모두 클릭해야만 향후 프로젝트에 시각화 시스템(HMI)이 추가될 경우, 모든 태그에 액세스할 수 있습니다. 데이터 블록의 기본 값들에 대해서만 '설정값' 체크박스 E:\#--인터스피어코리아\0617_지멘스\images\img3211.jpg를 클릭합니다.

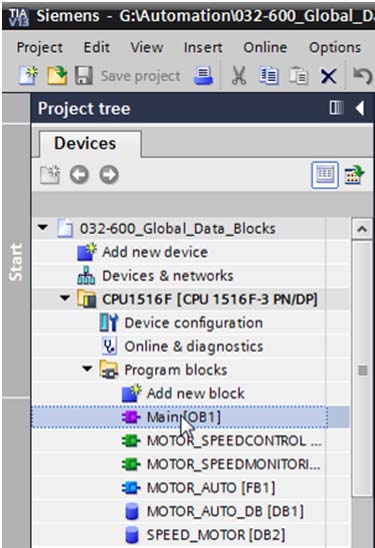
(→ E:\#--인터스피어코리아\0617_지멘스\images\img3211.jpgE:\#--인터스피어코리아\0617_지멘스\images\img3211.jpgE:\#--인터스피어코리아\0617_지멘스\images\img3211.jpgE:\#--인터스피어코리아\0617_지멘스\images\img3211.jpg )



*참고: 설정값 사용은 모듈의 단계별 따라해 보기에 보다 자세하게 설명되어 있습니다.*

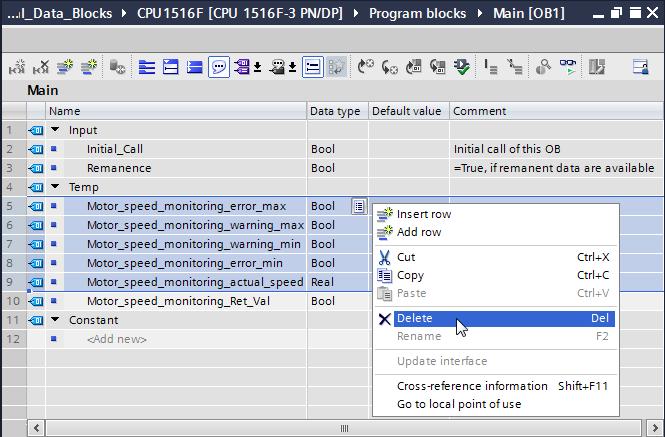
## 7.3 오거나이제이션 블록의 데이터 블록 데이터에 액세스

→ "Main [OB1]" 오거나이제이션 블록을 더블클릭해서 엽니다.



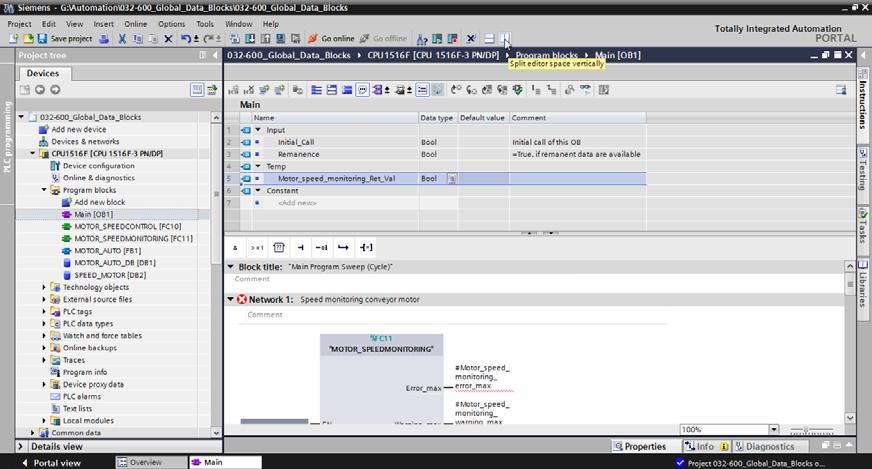
→ Bool 형식의 'Motor\_Speed\_Control\_Ret\_Val' 태그를 제외한, 나머지 더 이상 필요가 없는 Main [OB1]의 임시 태그들을 삭제합니다.

(→ Delete)



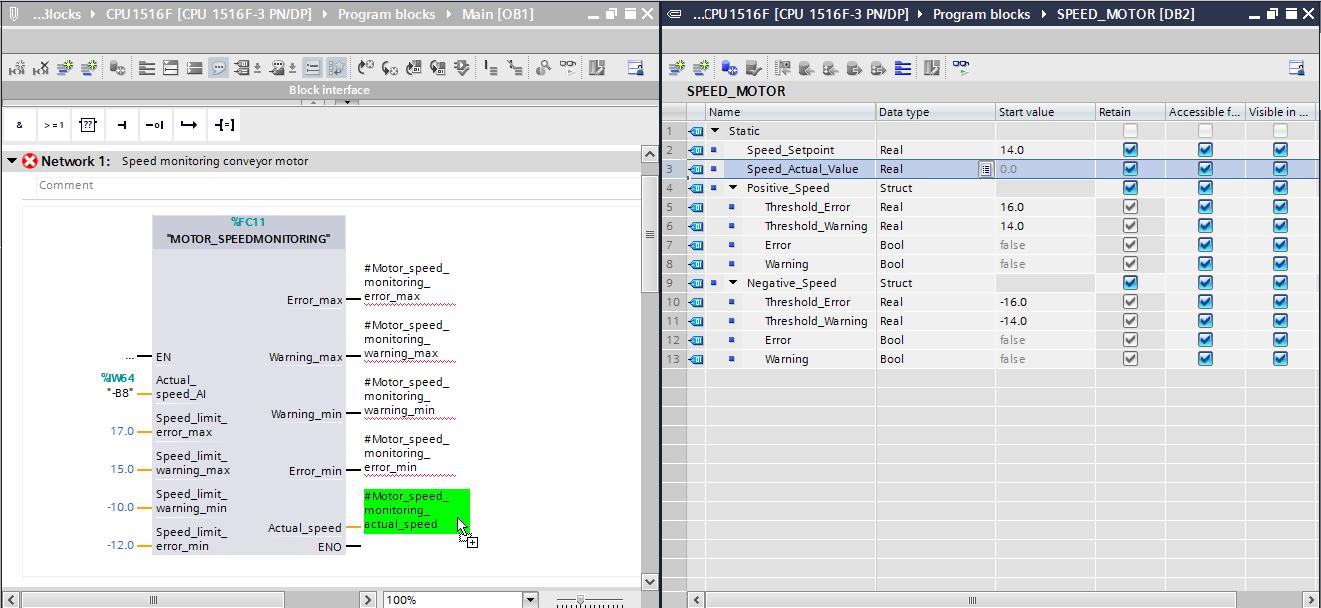
→ 편집기 영역을 수직 분할하기 위해 E:\#--인터스피어코리아\0617_지멘스\images\img3222.jpg 아이콘을 클릭해서 "SPEED\_MOTOR" [DB2] 데이터 블록 및 "Main" [OB1] 오거나이제이션 블록을 나란히 표시합니다.

(→ E:\#--인터스피어코리아\0617_지멘스\images\img3222.jpg)

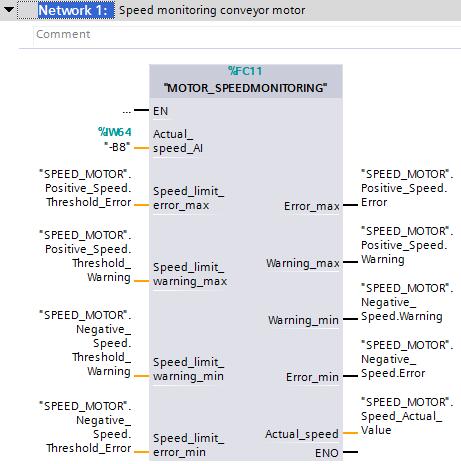


→ 끌어다 놓기 기능을 통해 "SPEED\_MOTOR" [DB2] 데이터 블록에서 상호 연결에 필요한 태그들을 "Main" [OB1] 에서 호출된 펑션 및 펑션 블록으로 이동시킵니다. 먼저, 'Speed\_Actual\_Value' 태그를 "MOTOR\_SPEEDMONITORING" [FC11] 블록의 'Actual\_speed' 출력으로 이동시킵니다.

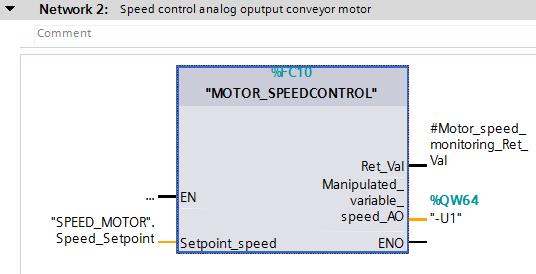
(→ Speed\_Actual\_Value)



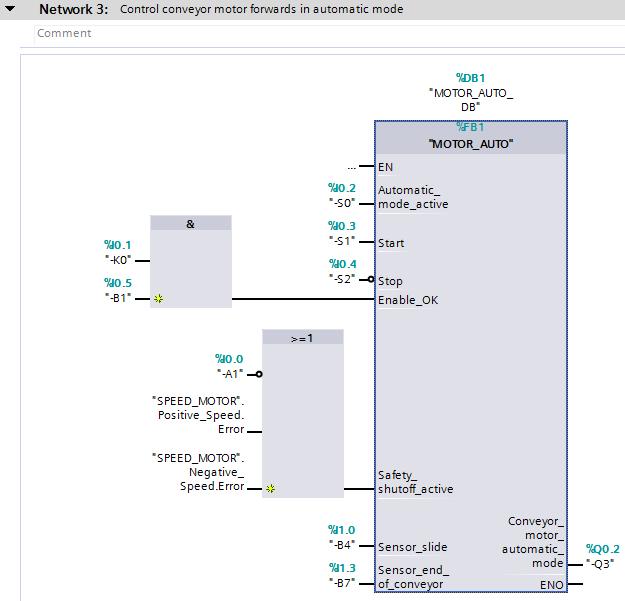
→ 그리고 아래와 같이 네트워크 1의 다른 접점들을 "SPEED\_MOTOR" [DB2] 데이터 블록에서 나온 태그들에 연결합니다.



→ 아래 그림과 같이 네트워크 2의 접점들을 "SPEED\_MOTOR" [DB2] 데이터 블록에서 나온 태그들에 연결합니다.



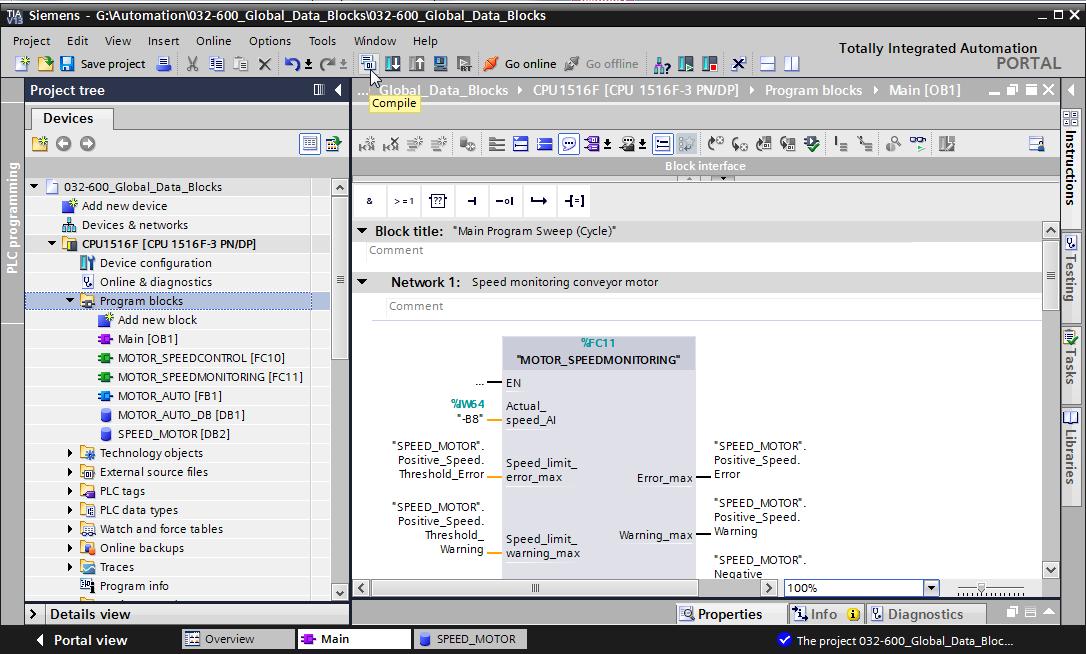
→ 아래 그림과 같이 네트워크 3의 접점들을 "SPEED\_MOTOR" [DB2] 데이터 블록에서 나온 태그들에 연결합니다.



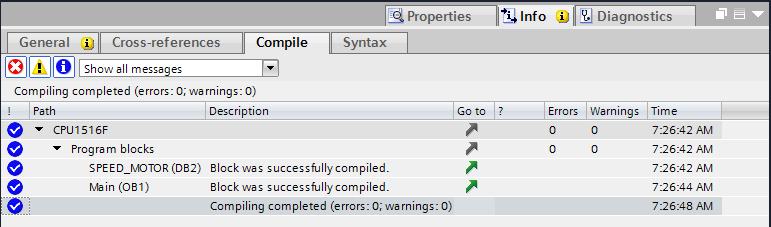
## 7.4 프로그램 저장 및 컴파일

→ 프로젝트를 저장하려면 메뉴에서 E:\#--인터스피어코리아\0617_지멘스\images\img3063.jpg 버튼을 선택합니다. 모든 블록을 컴파일하려면 "Program blocks" 폴더를 클릭하고 메뉴에서 컴파일을 위한 E:\#--인터스피어코리아\0617_지멘스\images\새홀리기\img2056.jpg 아이콘을 선택합니다.

(→ E:\#--인터스피어코리아\0617_지멘스\images\img3063.jpg → Program blocks → E:\#--인터스피어코리아\0617_지멘스\images\새홀리기\img2056.jpg)

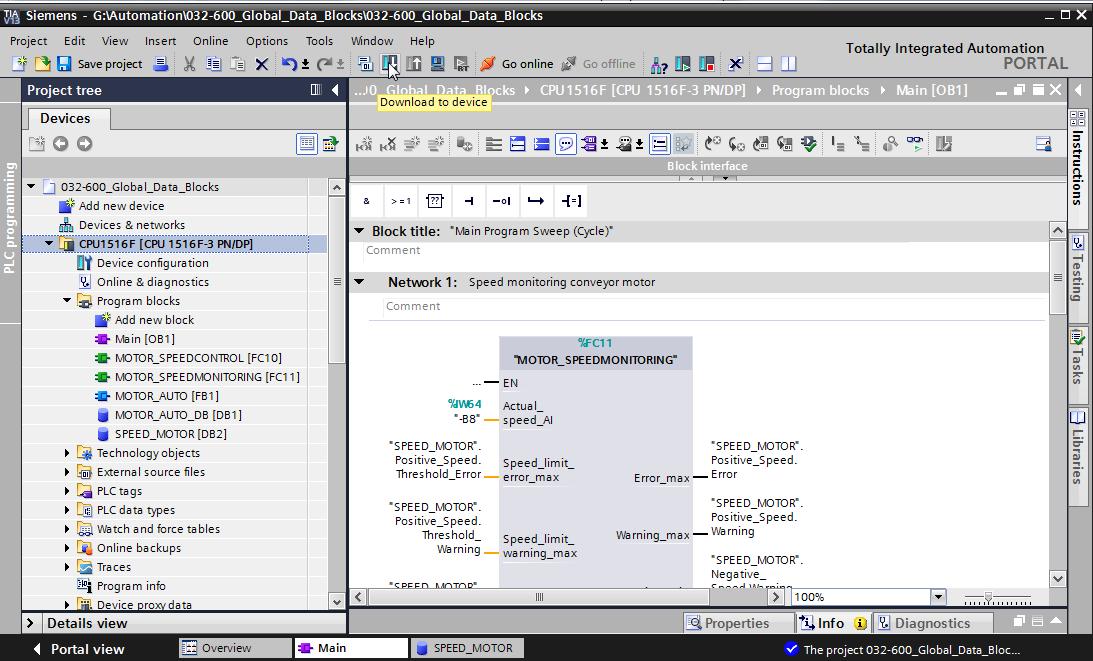


→ "Info" 아래의 "Compile" 영역에 블록이 성공적으로 컴파일이 되었는지 나타납니다.



## 7.5 프로그램 다운로드

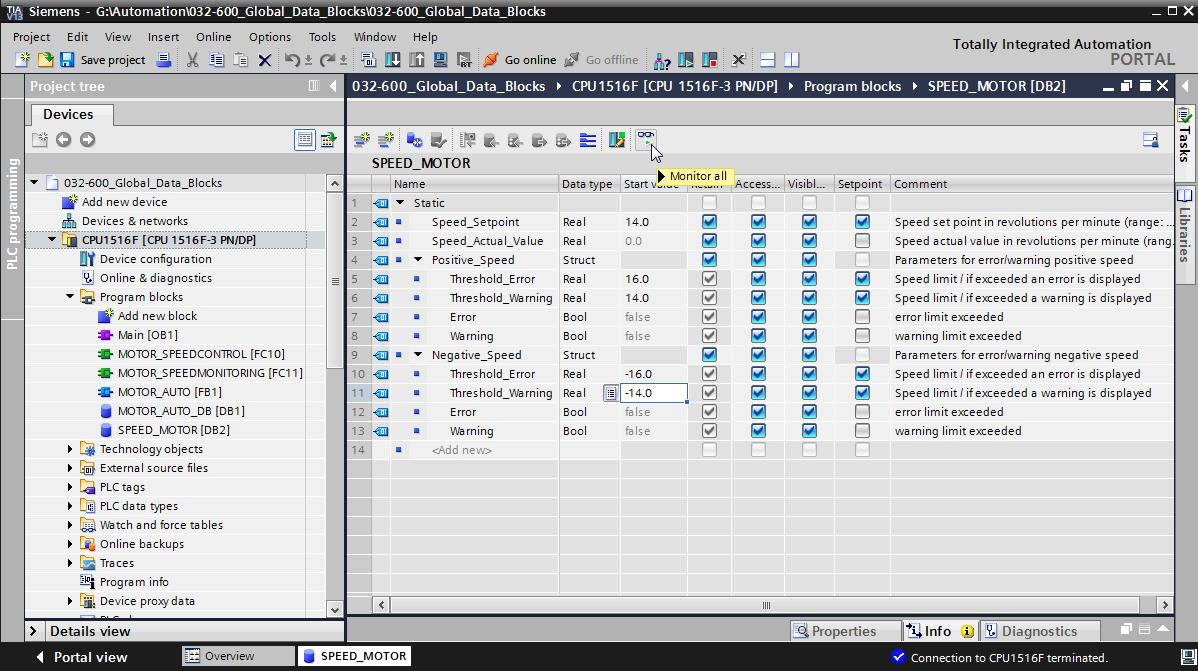
→ 컴파일이 성공적으로 완료되고 나면 앞서 설명한 하드웨어 구성을 포함해 생성된 프로그램과 함께 전체 컨트롤러를 다운로드할 수 있습니다. (→ E:\#--인터스피어코리아\0617_지멘스\images\img2061.jpg)



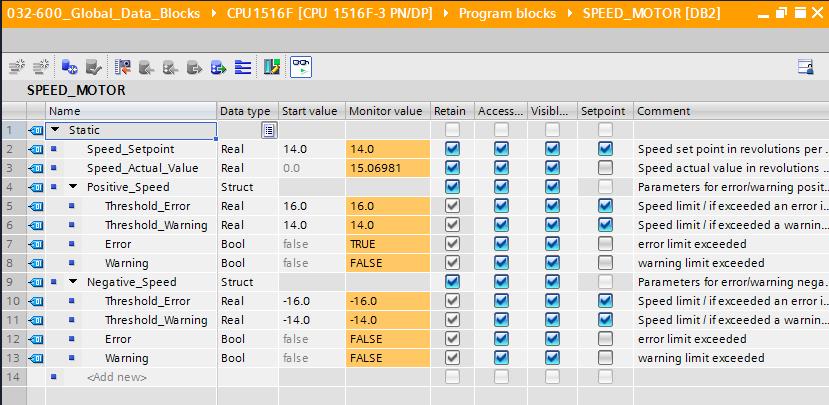
## 7.6 데이터 블록 값 모니터링/수정

→ 다운로드된 데이터 블록의 태그를 모니터링하려면 원하는 블록을 열어야 합니다. E:\#--인터스피어코리아\0617_지멘스\images\img2065.jpg 아이콘을 클릭해서 모니터링을 활성화/비활성화할 수 있습니다.

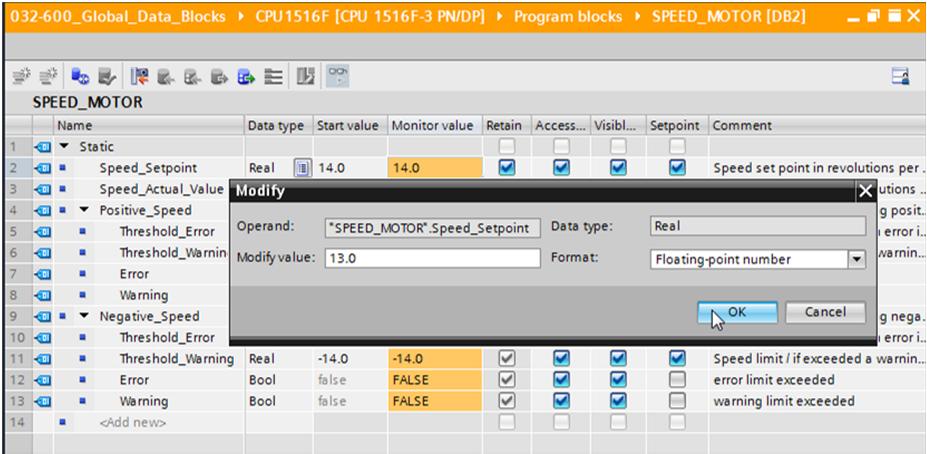
(→ SPEED\_MOTOR [DB2] → E:\#--인터스피어코리아\0617_지멘스\images\img2065.jpg)



→ "Monitor value" 열에서 해당 태그의 CPU 현재 값을 모니터링할 수 있습니다.



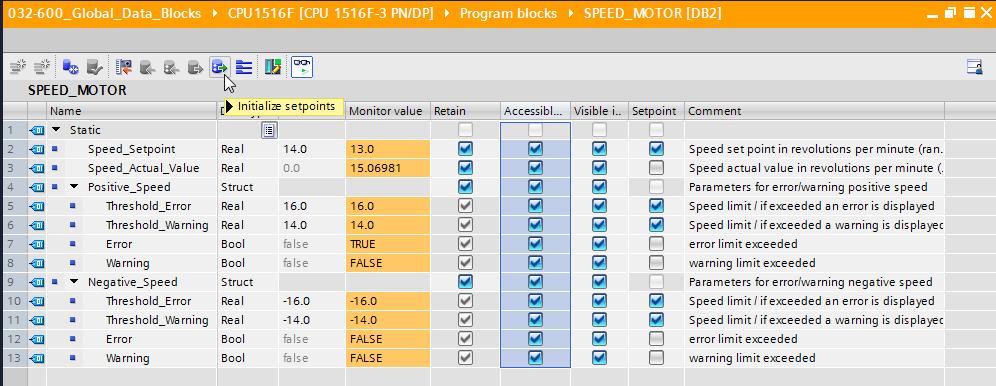
→ 값들 중 하나를 마우스 오른쪽 버튼으로 클릭하면 이들 값을 수정할 수 있는 "Modify" 대화상자가 열립니다 (→ Modify → Modify value: 13.0 → OK)

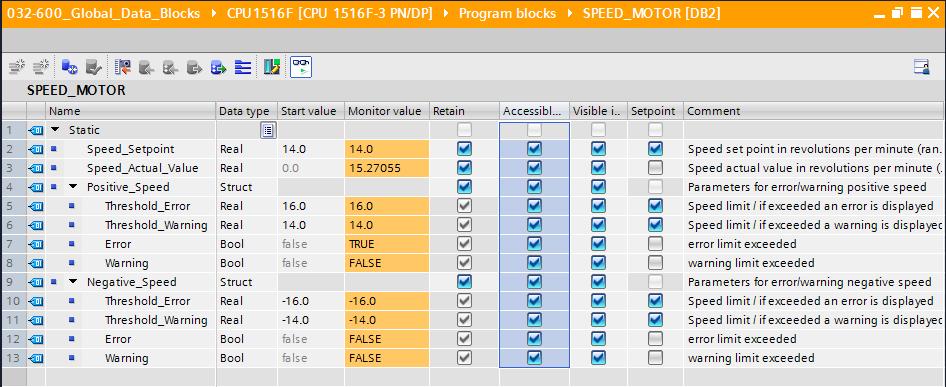


## 7.7 설정값 초기화/초기값 재설정

→ E:\#--인터스피어코리아\0617_지멘스\images\img3248.jpg 아이콘을 클릭해서 설정값을 초기화할 수 있습니다. 그러면, E:\#--인터스피어코리아\0617_지멘스\images\img3211.jpg "Setpoint" 체크박스가 선택된 태그의 초기값이 현재 값으로 적용됩니다.

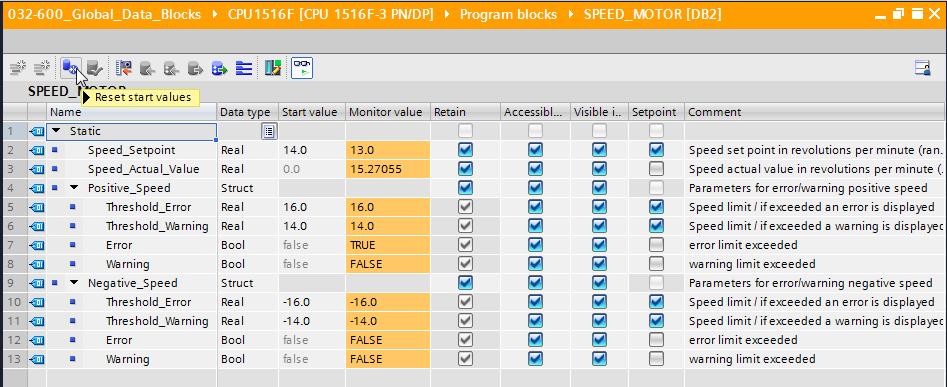
(→ E:\#--인터스피어코리아\0617_지멘스\images\img3248.jpg)

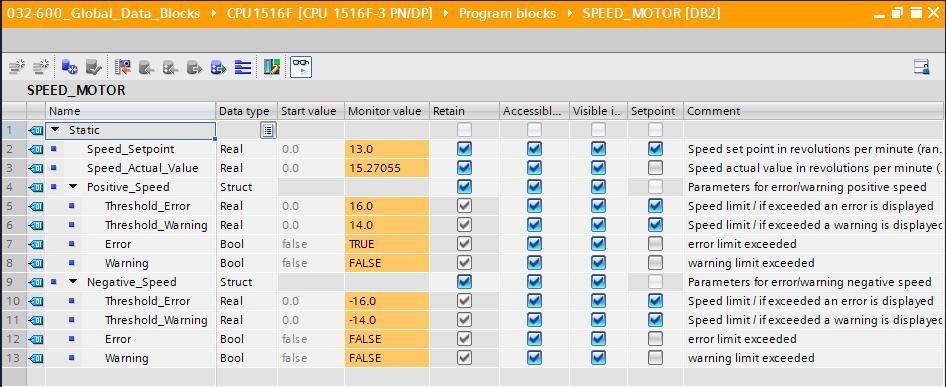




→ E:\#--인터스피어코리아\0617_지멘스\images\img3255.jpg 아이콘을 클릭하면 모든 초기값이 리셋됩니다.

(→ E:\#--인터스피어코리아\0617_지멘스\images\img3255.jpg)

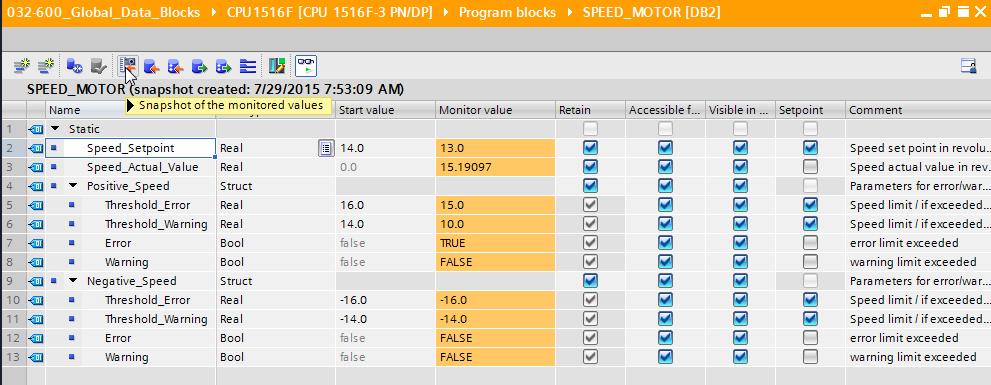


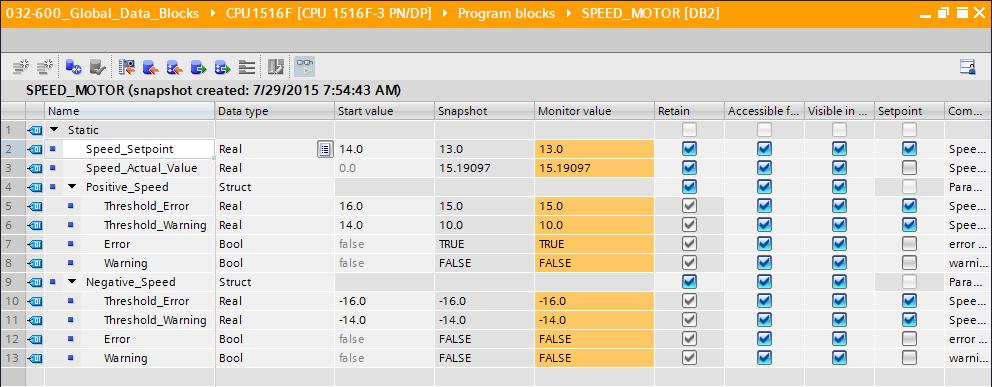


## 7.8 데이터 블록의 스냅샷

→ E:\#--인터스피어코리아\0617_지멘스\images\img3262.jpg 아이콘을 클릭하면 모니터링된 값들에 대한 스냅샷을 가져와서 이들 값을 초기값으로 적용하거나 나중에 CPU로 다시 전송할 수 있습니다.

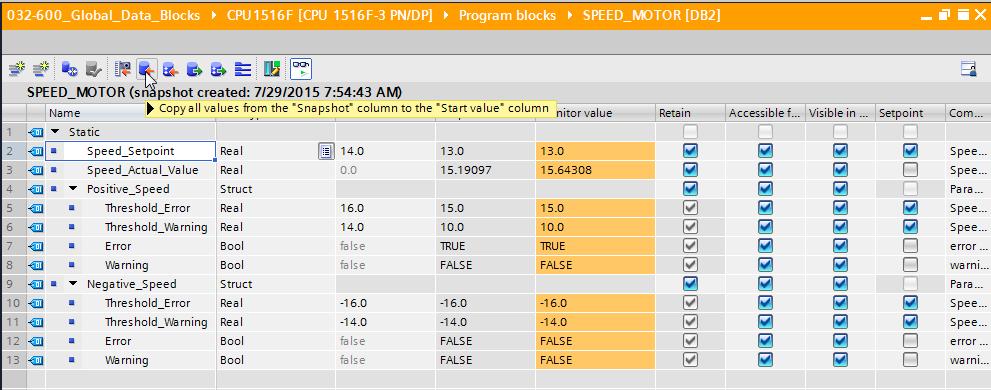
(→ E:\#--인터스피어코리아\0617_지멘스\images\img3262.jpg).

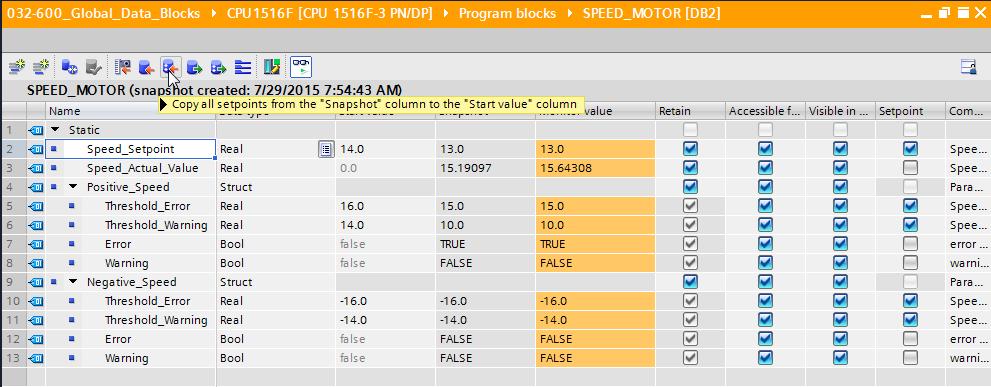


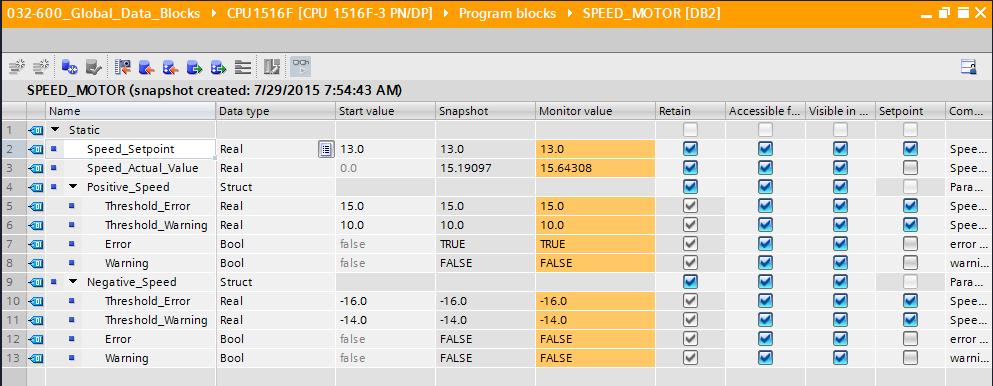


→ 아니면 E:\#--인터스피어코리아\0617_지멘스\images\img3269.jpg 아이콘을 클릭해 모든 값에 스냅샷의 값을 덮어쓰거나 E:\#--인터스피어코리아\0617_지멘스\images\img3272.jpg 아이콘을 클릭해 초기값에만 스냅샷의 값을 덮어쓸 수 있습니다. 대부분의 경우, 설정값으로 사용하는 태그에만 스냅샷을 적용합니다.

(→ E:\#--인터스피어코리아\0617_지멘스\images\img3272.jpg)

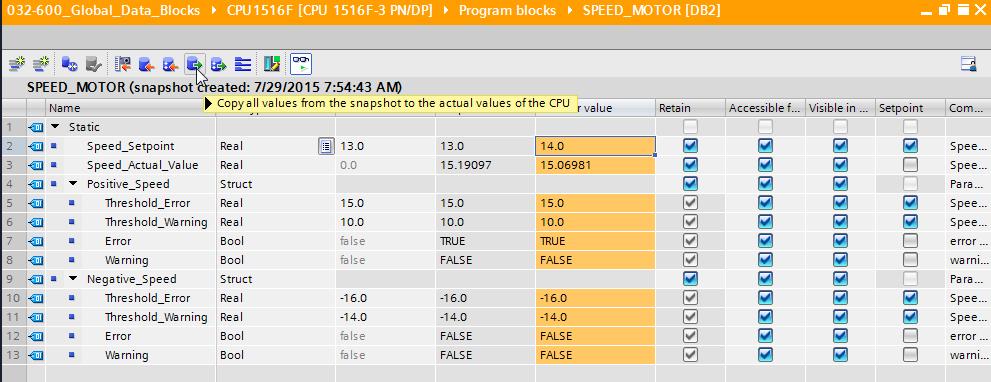


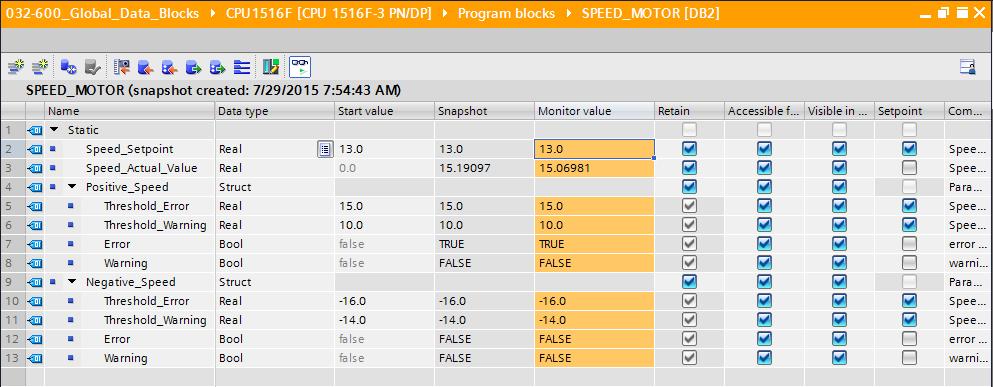




→ E:\#--인터스피어코리아\0617_지멘스\images\img3280.jpg를 클릭하면 스냅샷에 캡쳐된 데이터가 CPU로 전송됩니다.

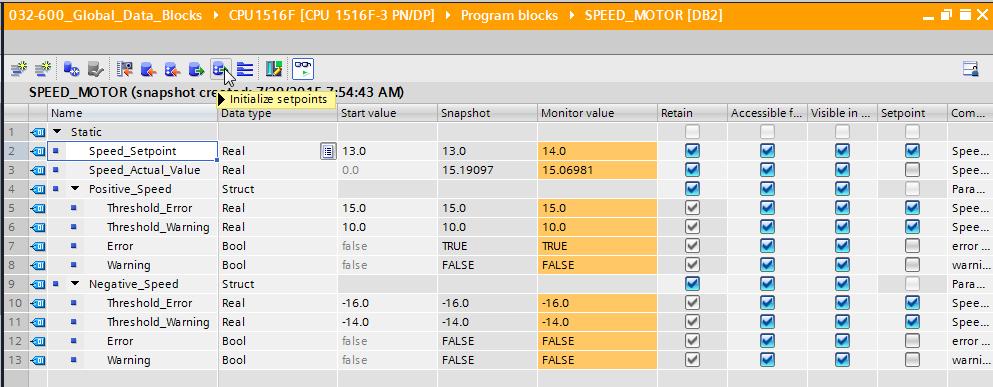
(→ E:\#--인터스피어코리아\0617_지멘스\images\img3280.jpg)

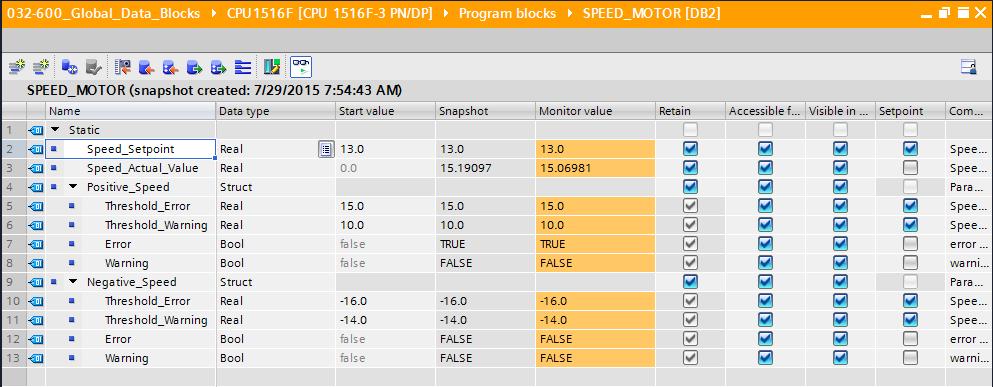




→ 모든 설정값을 초기값으로 덮어쓰기 하려면 E:\#--인터스피어코리아\0617_지멘스\images\img3287.jpg를 클릭해 이를 개시할 수 있습니다. "Setpoint" 체크박스를 선택하지 않은 CPU의 값들은 그대로 유지가 됩니다.

(→ E:\#--인터스피어코리아\0617_지멘스\images\img3287.jpg)

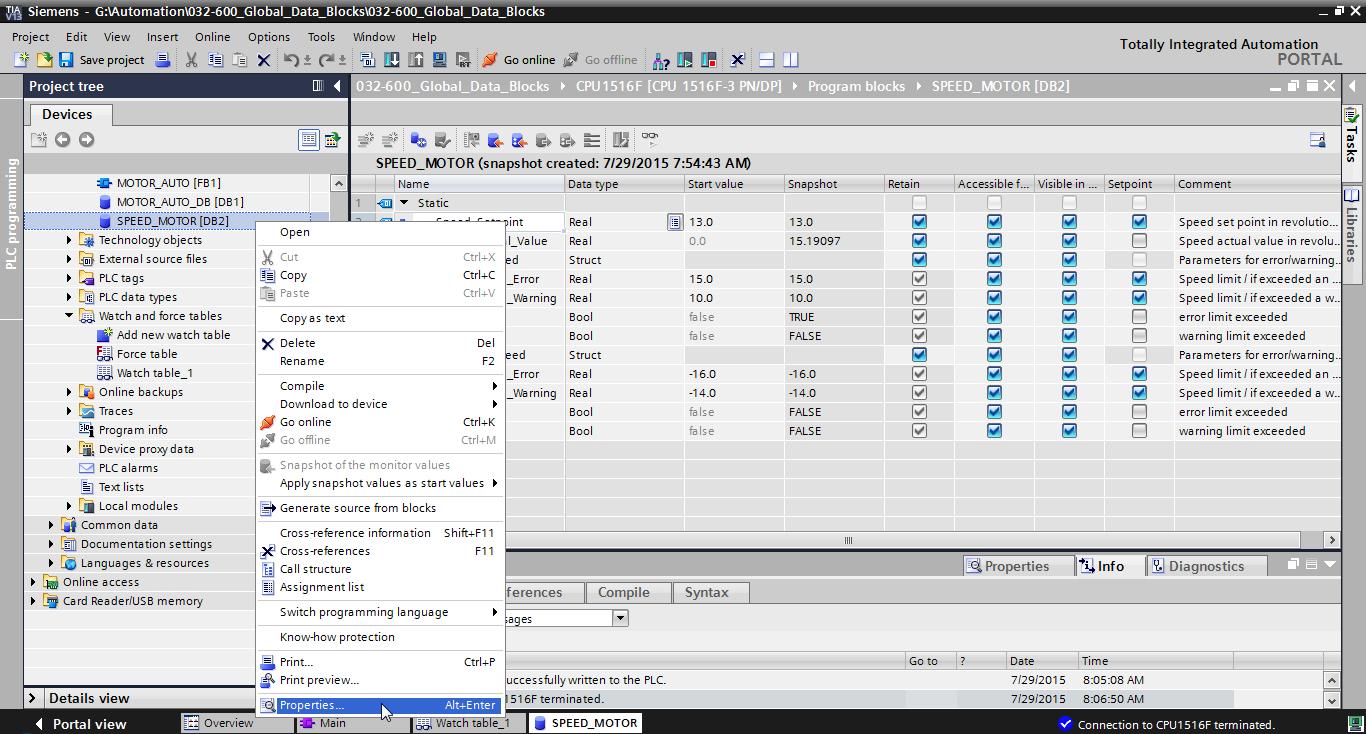




## 7.9 재초기화(Reninitialization) 없이 데이터 블록을 확장 및 다운로드

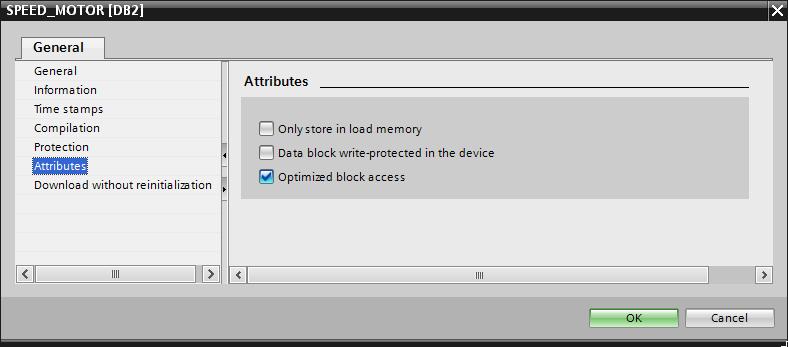
→ "SPEED\_MOTOR" [DB2] 데이터 블록에서 재초기화 없이 다운로드를 할 수 있으려면 E:\#--인터스피어코리아\0617_지멘스\images\img3294.jpg를 클릭한 다음, 데이터 블록의 속성을 열어야 합니다.

(→ E:\#--인터스피어코리아\0617_지멘스\images\img3294.jpg → SPEED\_MOTOR [DB2] → Properties)



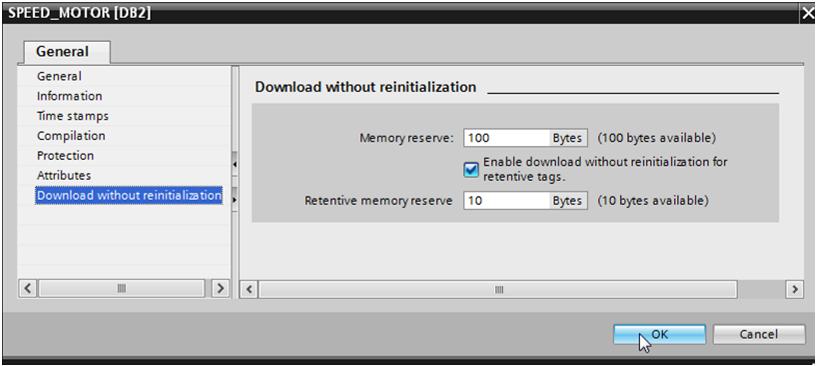
→ "General"의 "Attribtes" 아래로 가서 속성에서 "Optimized block access" 체크박스 E:\#--인터스피어코리아\0617_지멘스\images\img3211.jpg를 선택합니다.

(→ General → Attribtes → E:\#--인터스피어코리아\0617_지멘스\images\img3211.jpg Optimized block access)



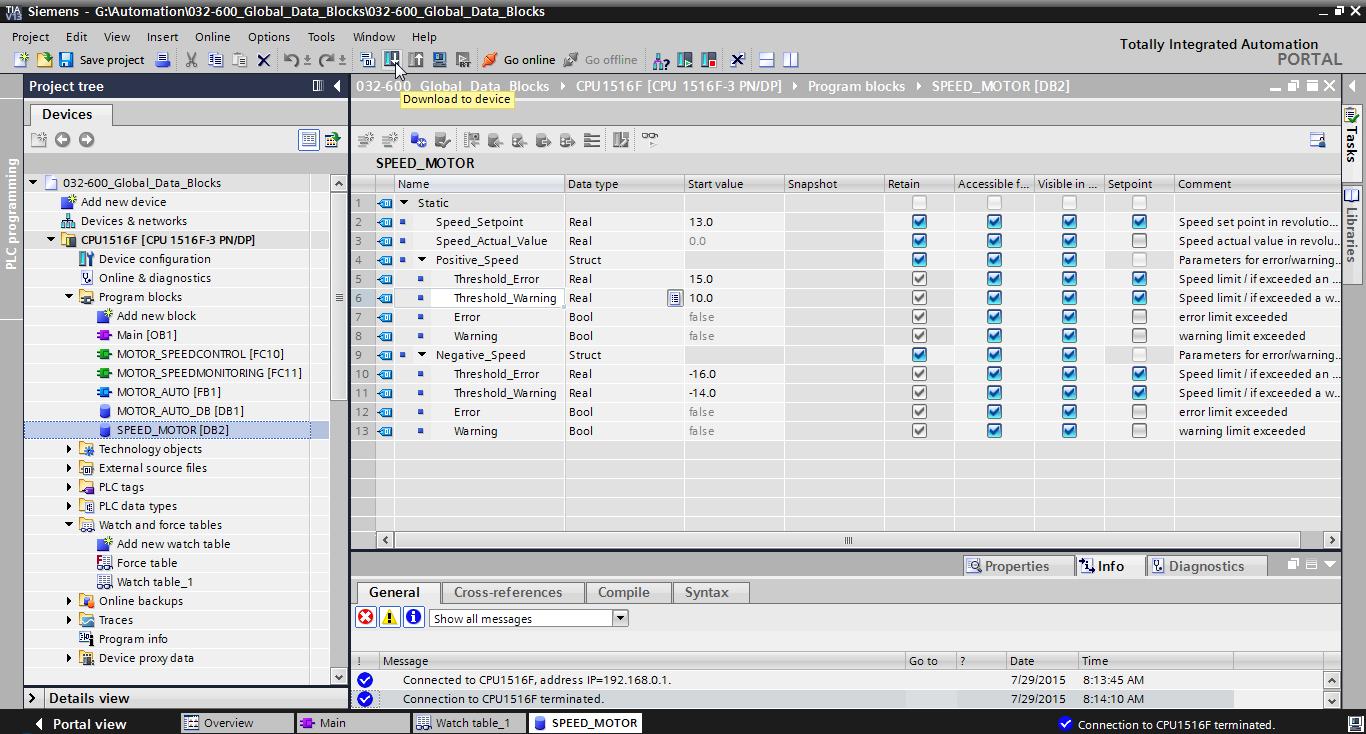
→ "Download without reninitialization"로 가서 데이터 블록에 'Retentive memory reserve'를 설정합니다.

(→ Download without reninitialization → Retentive memory reserve → 10Byte → OK)



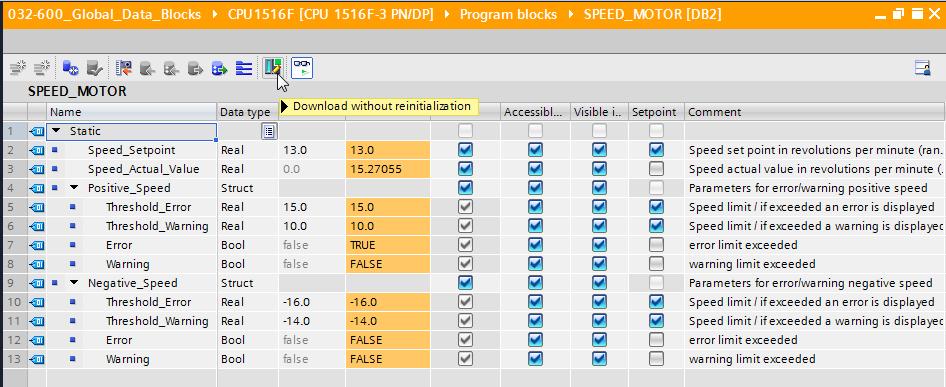
→ 컨트롤러에 "SPEED\_MOTOR" [DB] 데이터 블록을 다시 한 번 다운로드하고 E:\#--인터스피어코리아\0617_지멘스\images\img3302.jpg를 선택합니다.

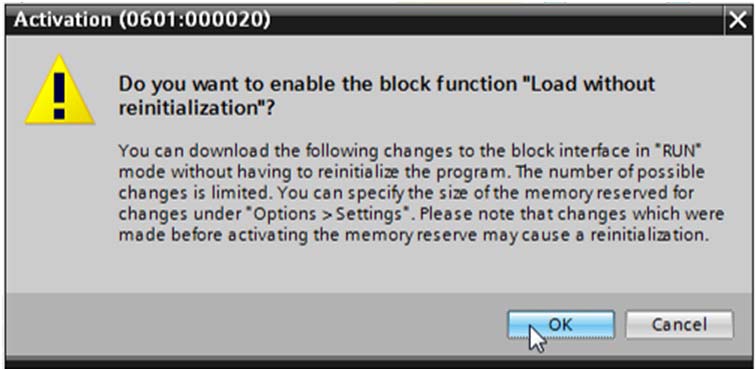
(→SPEED\_MOTOR [DB] → E:\#--인터스피어코리아\0617_지멘스\images\img2061.jpg → E:\#--인터스피어코리아\0617_지멘스\images\img3302.jpg)



→ 그런 다음, E:\#--인터스피어코리아\0617_지멘스\images\img3308.jpg 아이콘을 클릭하여 재초기화 없이 다운로드를 활성화하고 안전을 위한 확인창이 나타나면 "OK"를 클릭해 확인합니다.

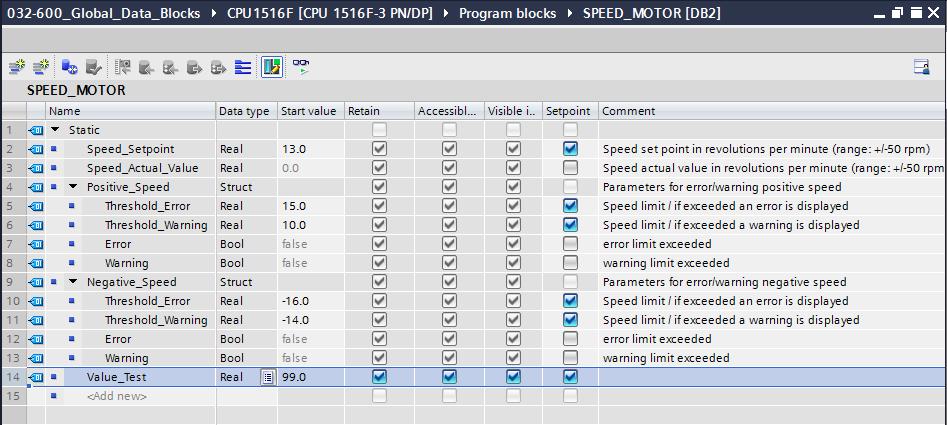
(→ E:\#--인터스피어코리아\0617_지멘스\images\img3308.jpg → OK)





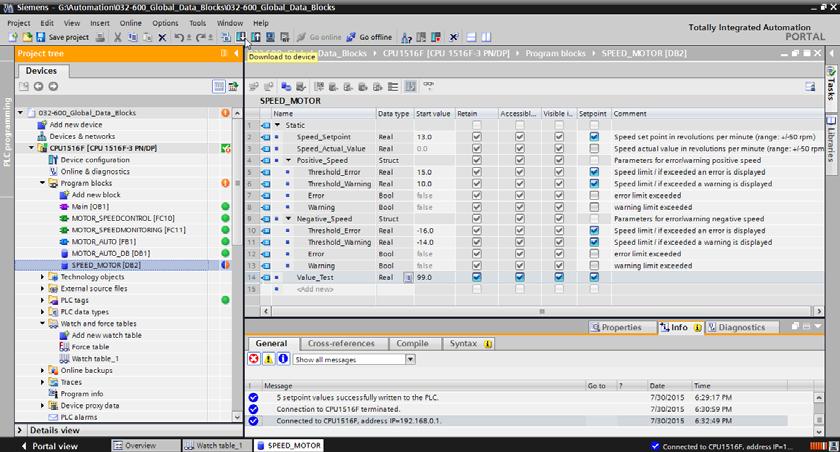
→ 이제 데이터 블록에 어떤 태그든 추가를 할 수 있습니다.

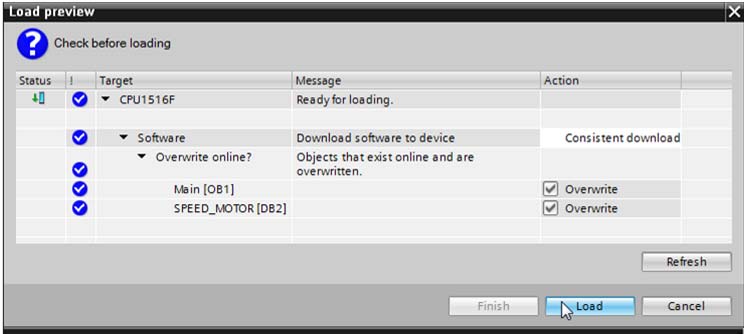
(→ Name: Value\_test → Data type: Real → Start value: 99)



→ 컨트롤러에 "SPEED\_MOTOR" [DB] 데이터 블록을 다시 한 번 다운로드합니다.

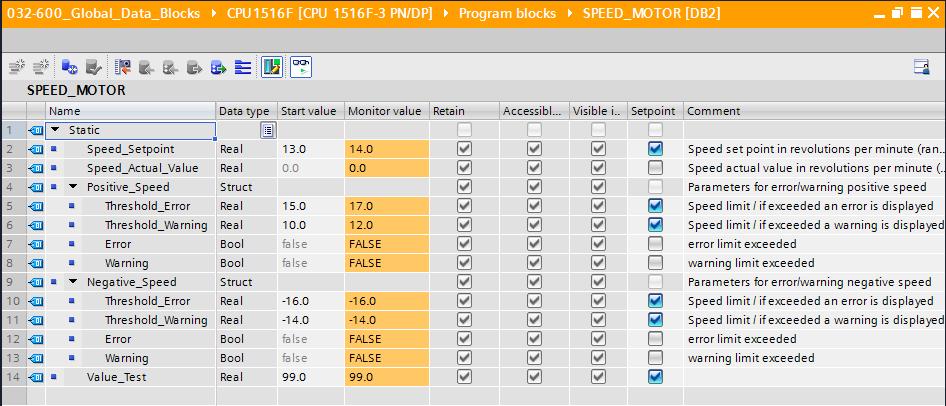
(→SPEED\_MOTOR [DB] → E:\#--인터스피어코리아\0617_지멘스\images\img2061.jpg → Download)





→ 다시 블록을 모니터링하기 위해 E:\#--인터스피어코리아\0617_지멘스\images\img2065.jpg를 클릭하면 모니터링된 값들이 초기값으로 덮어쓰기 되지 않았다는 것을 확인할 수 있습니다.

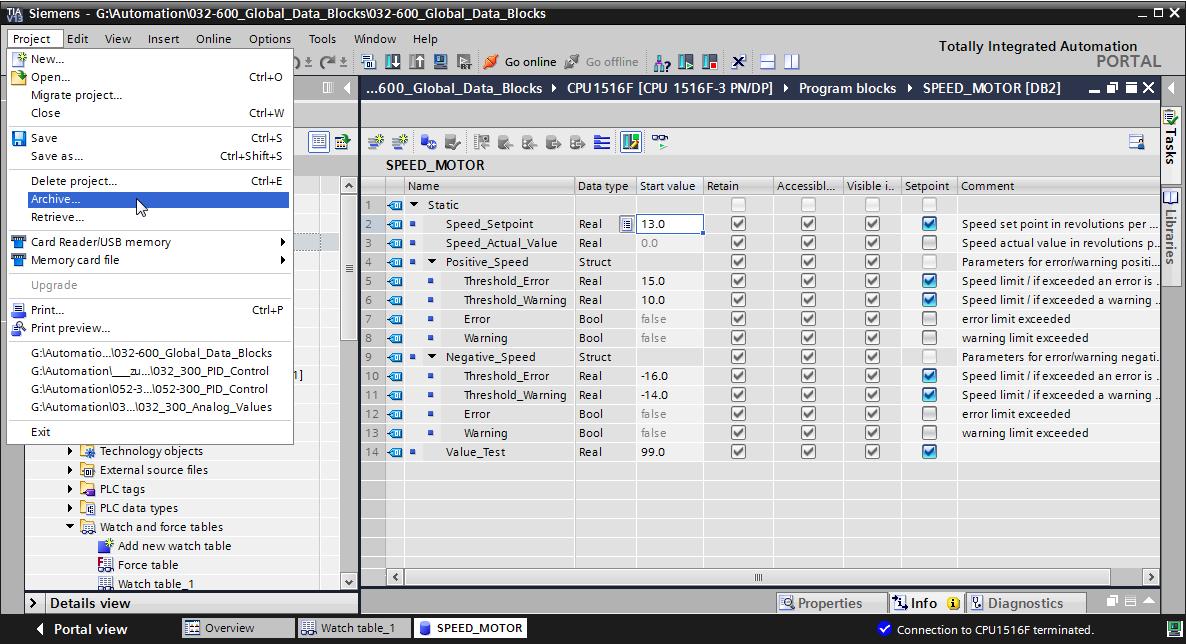
(→ E:\#--인터스피어코리아\0617_지멘스\images\img2065.jpg)



## 7.10 프로젝트 아카이브

→ 마지막 단계로 완성된 프로젝트를 아카이브하기 위해 "Project" 메뉴에서 "Archive..." 항목을 선택합니다. 프로젝트를 아카이브하고자 하는 폴더를 선택하고 "TIA Portal 프로젝트 아카이브" 파일 유형으로 이를 저장합니다.

(→ Project → Archive → TIA Portal project archive → 032-600\_Global\_Data\_Blocks…. → Save)



# 8 체크리스트

|  |  |  |
| --- | --- | --- |
| 번호 | 설명 | 완료 |
| 1 | 데이터 블록 SPEED\_MOTOR [DB2]가 성공적으로 다운로드 |  |
| 2 | Main [OB1]에서 프로그램 변경 수행 |  |
| 3 | 오류 메시지 없이 성공적으로 컴파일 |  |
| 4 | 오류 메시지 없이 성공적으로 다운로드 |  |
| 5 | 스테이션 전원 켜기 (-K0 = 1)  실린더 복귀 / 피드백 활성화 (-B1 = 1)  비상 정지 오프 (-A1 = 1)가 활성화되지 않음  자동 모드 (-S0 = 1)  푸시버튼 자동 정지가 구동되지 않음 (-S2 =1)  자동 시작 푸시버튼을 짧게 누르기 (-S2 = 1)  슬라이드의 센서 부분이 활성화되고 (-B4 = 1) 컨베이어 모터 M1 가변 속도 (-Q3 = 1) 스위치를 켠 상태로 유지  속도는 +/- 50 rpm 범위의 속도 설정값에 해당 |  |
| 6 | 컨베이어 끝의 센서 활성화 (-B7 = 1) → -Q3 = 0 (2초 후) |  |
| 7 | 자동 정지 푸시버튼을 짧게 누르기 (-S2 = 0) → -Q3 = 0 |  |
| 8 | 비상 정지 오프를 활성화 (-A1 = 0) → -Q3 = 0 |  |
| 9 | 수동 모드 (-S0 = 0) → -Q3 = 0 |  |
| 10 | 스테이션 전원 끄기 (-K0 = 0) → -Q3 = 0 |  |
| 11 | 실린더가 복귀되지 않음 (-B1 = 0) → -Q3 = 0 |  |
| 12 | 속도 > Motor\_speed\_monitoring\_error\_max → -Q3 = 0 |  |
| 13 | 속도 < Motor\_speed\_monitoring\_error\_min → -Q3 = 0 |  |
| 14 | 프로젝트가 성공적으로 아카이브 됨 |  |

# 9 연습

## 9.1 과제 – 연습

이 연습에서는 글로벌 데이터 블록 "MAGAZINE\_PLASTIC" [DB3]을 추가적으로 생성해 보겠습니다.

플라스틱 컴포넌트의 경우 카운터의 설정값 및 실제 값이 이 데이터 블록에서 설정 및 표시됩니다.

"MOTOR\_AUTO" [FB1] 펑션 블록에 설정값과 실제 값을 연결할 수 있는 입력과 출력이 추가됩니다.

## 9.2 기술 다이어그램

여기에는 과제를 위한 기술 다이어그램이 나와 있습니다.

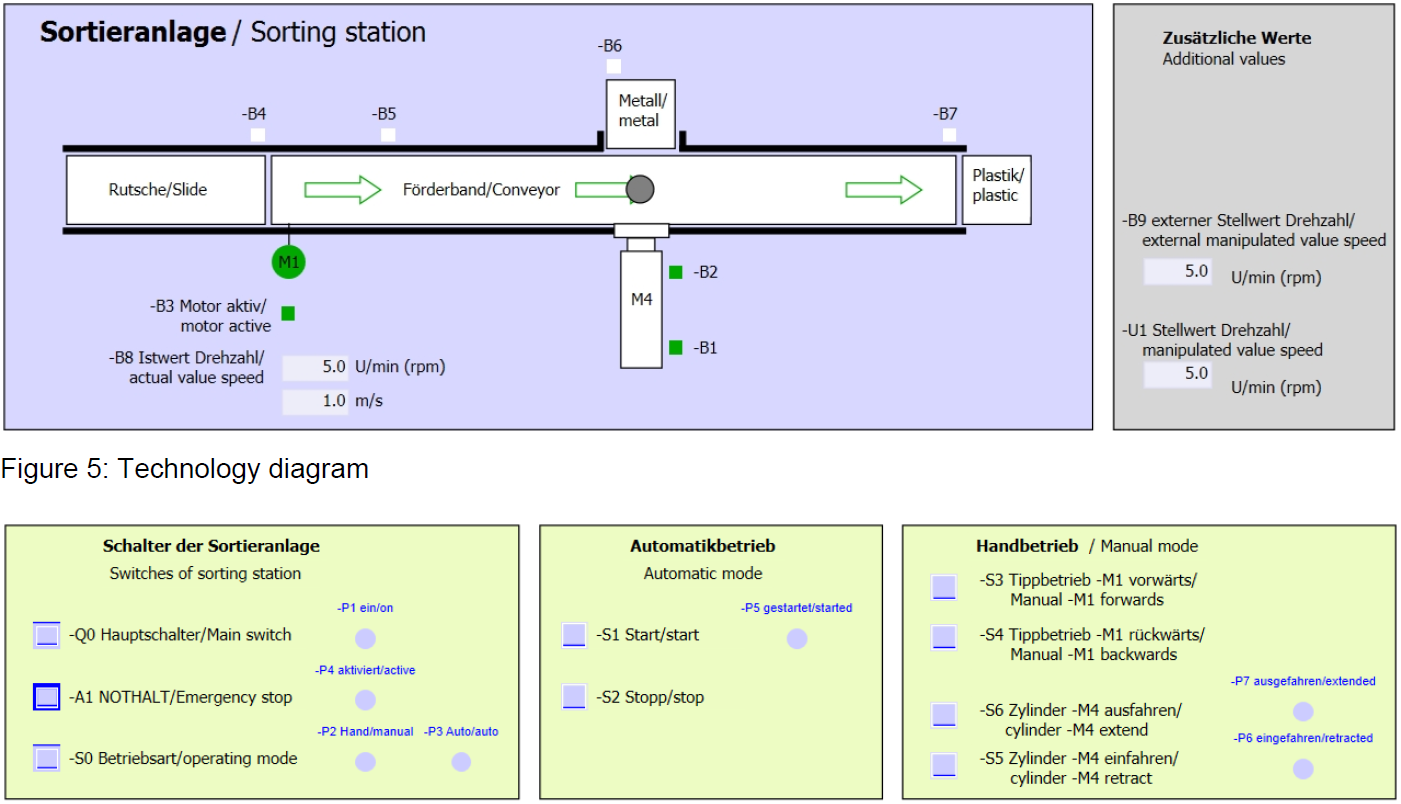


그림 5: 기술 다이어그램

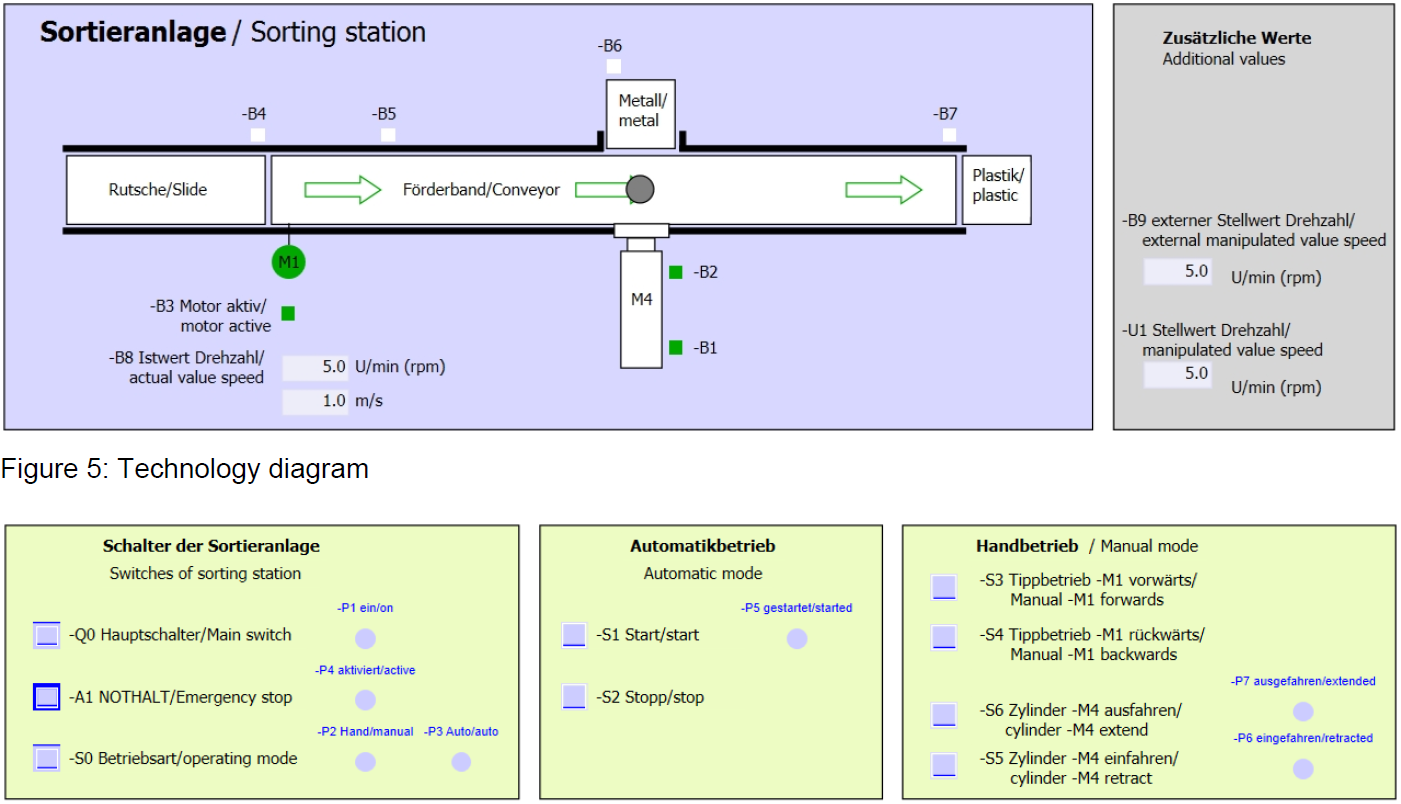


그림 6: 제어 패널

## 9.3 참조 목록

이 과제를 위한 글로벌 오퍼랜드로서 아래와 같은 신호들이 필요합니다.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DI | 유형 | 식별자 | 기능 | NC/NO |
| I 0.0 | BOOL | -A1 | 반환 신호 비상 정지 OK | NC |
| I 0.1 | BOOL | -K0 | 메인 스위치 "온" | NO |
| I 0.2 | BOOL | -S0 | 모드 선택 수동 (0) / 자동 (1) | 수동 = 0  자동 = 1 |
| I 0.3 | BOOL | -S1 | 푸시버튼 자동 시작 | NO |
| I 0.4 | BOOL | -S2 | 푸시버튼 자동 정지 | NC |
| I 0.5 | BOOL | -B1 | 센서 실린더 -M4 복귀 | NO |
| I 1.0 | BOOL | -B4 | 슬라이드의 센서 | NO |
| I 1.3 | BOOL | -B7 | 컨베이어 끝 센서 | NO |
| IW64 | BOOL | -B8 | 모터의 센서 실제 속도 값 +/- 10V는 +/- 50 rpm에 해당 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DO | 유형 | 식별자 | 기능 |  |
| Q 0.2 | BOOL | -Q3 | 컨베이어 모터 -M1 가변 속도 |  |
| QW 64 | BOOL | -U1 | 2방향 모터의 조작 속도 값 +/- 10V는 +/- 50 rpm에 해당 |  |

*참조 목록 범례*

DI Digital Input DO Digital Output

AI Analog Input AO Analog Output

I Input Q Output

NC Normally Closed

NO Normally Open

## 9.4 계획 수립

과제 수행에 대한 계획을 자체적으로 수립합니다.

## 9.5 체크리스트 – 연습

|  |  |  |
| --- | --- | --- |
| 번호 | 설명 | 완료 |
| 1 | 데이터 블록 MAGAZINE\_PLASTIC [DB3]이 성공적으로 다운로드 |  |
| 2 | MOTOR\_AUTO [FB1]에서 프로그램 변경이 수행 |  |
| 3 | Main [OB1]에서 프로그램 변경이 수행 |  |
| 4 | 오류 메시지 없이 성공적으로 컴파일 |  |
| 5 | 오류 메시지 없이 성공적으로 다운로드 |  |
| 6 | 스테이션 전원 켜기 (-K0 = 1)  실린더 복귀 / 피드백 활성화 (-B1 = 1)  비상 정지 오프 (-A1 = 1)가 활성화되지 않음  자동 모드 (-S0 = 1)  푸시버튼 자동 정지가 구동되지 않음 (-S2 =1)  자동 시작 푸시버튼을 짧게 누르기 (-S2 = 1)  슬라이드의 센서 부분이 활성화되고 (-B4 = 1) 컨베이어 모터 -M1 가변 속도 (-Q3 = 1) 스위치를 켠 상태로 유지  속도는 +/- 50 rpm 범위의 속도 설정값에 해당 |  |
| 7 | 컨베이어 끝의 센서 활성화 (-B7 = 1) → -Q3 = 0 (2초 후) |  |
| 8 | 자동 정지 푸시버튼을 짧게 누르기 (-S2 = 0) → -Q3 = 0 |  |
| 9 | 비상 정지 오프를 활성화 (-A1 = 0) → -Q3 = 0 |  |
| 10 | 수동 모드 (-S0 = 0) → -Q3 = 0 |  |
| 11 | 스테이션 전원 끄기 (-K0 = 0) → -Q3 = 0 |  |
| 12 | 실린더가 복귀되지 않음 (-B1 = 0) → -Q3 = 0 |  |
| 13 | 속도 > Motor\_speed\_monitoring\_error\_max → -Q3 = 0 |  |
| 14 | 속도 < Motor\_speed\_monitoring\_error\_min → -Q3 = 0 |  |
| 15 | 프로젝트가 성공적으로 아카이브 됨 |  |

# 10 추가 정보

초기 및 심화 교육에 방향을 제시하는 도구의 차원에서 TIA Portal 모듈에 대한 추가 정보를 활용할 수 있습니다. 시작하기, 동영상, 교재, 앱, 매뉴얼, 프로그래밍 지침, 체험용 소프트웨어/펌웨어 등을 아래 링크에서 찾아보실 수 있습니다.

[www.siemens.com/sce/s7-1500](http://www.siemens.com/sce/s7-1500)