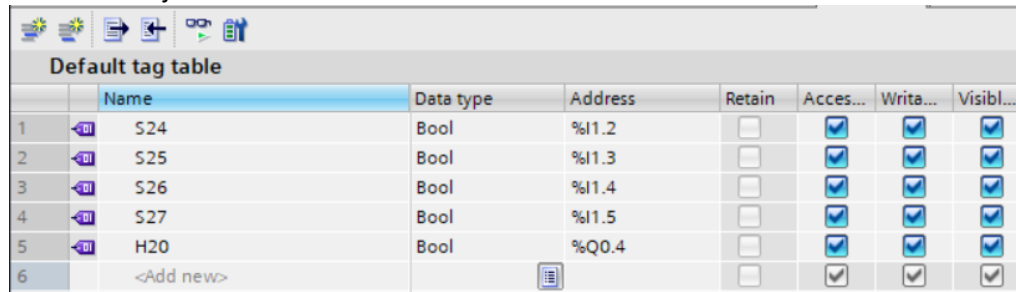


Övningar 1 – Datatyper - 1200

TIA-Portal (Siemens)

1. Skapa ett nytt projekt.
2. Deklarera följande variabler i TAG-listan.

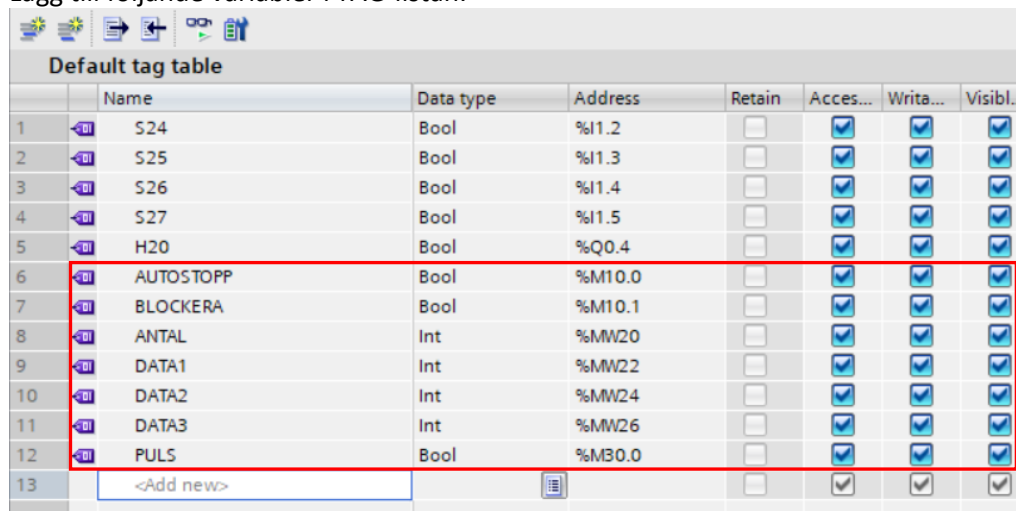


	Name	Data type	Address	Retain	Acces...	Writa...	Visibl...
1	S24	Bool	%I1.2	☐	☒	☒	☒
2	S25	Bool	%I1.3	☐	☒	☒	☒
3	S26	Bool	%I1.4	☐	☒	☒	☒
4	S27	Bool	%I1.5	☐	☒	☒	☒
5	H20	Bool	%Q0.4	☐	☒	☒	☒
6	<Add new>			☐	☒	☒	☒

BINÄR GRINDLOGIK

Samtliga uppgifter ska programmeras i **PROGRAMBLOCK_1**

3. Skapa ett nytt FC och tilldela namnet **PROGRAMBLOCK_1** Välj själv programspråk.
4. Programmera en SET- och RESET-funktion för **H20**.
Använd **S24** för SET och **S26** för RESET.
Spara, kompilera, ladda ned till CPU och testa.
5. Komplettera i föregående uppgift så att **H20** även kan slås ifrån med **S25**.
Spara, kompilera, ladda ned till CPU och testa.
6. Lägg till följande variabler i TAG-listan.



	Name	Data type	Address	Retain	Acces...	Writa...	Visibl...
1	S24	Bool	%I1.2	☐	☒	☒	☒
2	S25	Bool	%I1.3	☐	☒	☒	☒
3	S26	Bool	%I1.4	☐	☒	☒	☒
4	S27	Bool	%I1.5	☐	☒	☒	☒
5	H20	Bool	%Q0.4	☐	☒	☒	☒
6	AUTOSTOPP	Bool	%M10.0	☐	☒	☒	☒
7	BLOCKERA	Bool	%M10.1	☐	☒	☒	☒
8	ANTAL	Int	%MW20	☐	☒	☒	☒
9	DATA1	Int	%MW22	☐	☒	☒	☒
10	DATA2	Int	%MW24	☐	☒	☒	☒
11	DATA3	Int	%MW26	☐	☒	☒	☒
12	PULS	Bool	%M30.0	☐	☒	☒	☒
13	<Add new>			☐	☒	☒	☒

Observera datatypen.

7. Komplettera med följande funktion.
Om man inte trycker på **S25** eller **S26** inom 5,4 s för att slå ifrån **H20**, ska det ske automatiskt.
Då **H20** varit till i 5,4 sekunder ska **H20** automatiskt slås ifrån.
Använd variabeln **AUTOSTOPP**
Spara, kompilera, ladda ned till CPU och testa.

8. Komplettera med följande funktion.
Då **H20** varit till 3 gånger ska en räknare aktiveras och blockera (bryta) funktionen för SET av **H20**.
För att återställa denna blockeringsfunktion ska man trycka på **S27**.
Använd variabeln **BLOCKERA**
Använd **S27** för att återställa countern och dess funktion.
Spara, kompilera, ladda ned till CPU och testa.
9. Komplettera med följande funktion.
Återställningsfunktionen av countern som sker med **S27** ska endast vara möjlig då countern aktiverat sin utgång d.v.s. efter att **H20** varit till 3 gånger.
Spara, kompilera, ladda ned till CPU och testa.

INT (INTEGER) (16 bit) FUNCTION

10. Komplettera med följande funktion.
Då **H20** är från ska decimala värdet 6 skrivas till variabeln **DATA1** och när **H20** är till ska värdet 7 skrivas till **DATA1**.
Spara, kompilera, ladda ned till CPU och testa.
11. Komplettera med följande funktion.
Värdet i **DATA1** ska adderas med konstanten 3 och dess summa ska skrivas till variabeln **DATA2**.
Funktionen ska ske ovillkorligt.
Spara, kompilera, ladda ned till CPU och testa.
12. Komplettera med följande funktion.
Värdet i **DATA2** ska divideras med konstanten 3 och dess kvot ska skrivas till variabeln **DATA3**.
Funktionen ska ske ovillkorligt.
Spara, kompilera, ladda ned till CPU och testa.

Att fundera på!

Varför blir det samma svar i DATA3 oavsett om DATA2 är 9 eller 10?

13. Komplettera med följande funktion.
En adderare ska räkna in hur många gånger **H20** varit till. Värdet ska lagras i variabeln **ANTAL**.
Använd MATH-funktion ADD för uppgiften. Välj i ADD funktionen datatypen INT.
Spara, kompilera, ladda ned till CPU och testa.

Att fundera på!

Vad sker med värdet i ANTAL när S24 trycks in?

Förklara varför.

14. Komplettera med följande funktion.
Lägg till en P_TRIG precis för ingångsbenet EN på adderaren i föregående uppgift.
Använd variabeln **PULS** till denna.
Spara, kompilera, ladda ned till CPU och testa.

Att fundera på!

Vad sker med värdet i ANTAL när S24 trycks in?

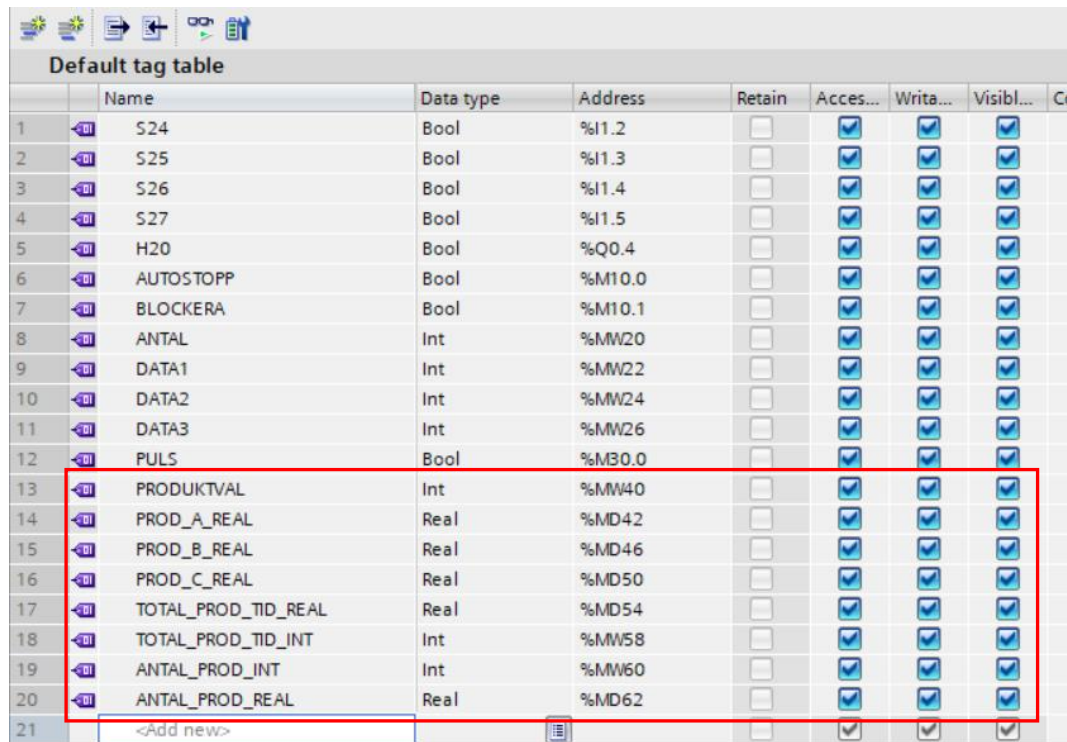
Förklara varför.

REAL (32 bit) FUNCTION

I följande uppgifter ska datatypen REAL bl.a. användas. Datatypen REAL är ett s.k. "flyttal" vilket utgörs av en mantissa och en exponent. Detta gör det möjligt att hantera ett decimalkomma som "flyter" och man kan hantera decimaler. Dessutom kan det hantera både positiva och negativa tal i intervallet $\pm 1.175\,495\text{e-}38$ till $\pm 3.402823\text{e}+38$

Innehållet i ett flyttal kan dock inte skickas ut på en fysisk analog utgång för att styra en process. Därför måste man konvertera det till en datatyp som kan hantera det, vanligen datatypen WORD.

15. Lägg till följande variabler i TAG-listan.



	Name	Data type	Address	Retain	Acces...	Writa...	Visibl...	C...
1	S24	Bool	%I1.2	☐	☒	☒	☒	
2	S25	Bool	%I1.3	☐	☒	☒	☒	
3	S26	Bool	%I1.4	☐	☒	☒	☒	
4	S27	Bool	%I1.5	☐	☒	☒	☒	
5	H20	Bool	%Q0.4	☐	☒	☒	☒	
6	AUTOSTOPP	Bool	%M10.0	☐	☒	☒	☒	
7	BLOCKERA	Bool	%M10.1	☐	☒	☒	☒	
8	ANTAL	Int	%MW20	☐	☒	☒	☒	
9	DATA1	Int	%MW22	☐	☒	☒	☒	
10	DATA2	Int	%MW24	☐	☒	☒	☒	
11	DATA3	Int	%MW26	☐	☒	☒	☒	
12	PULS	Bool	%M30.0	☐	☒	☒	☒	
13	PRODUKTVAL	Int	%MW40	☐	☒	☒	☒	
14	PROD_A_REAL	Real	%MD42	☐	☒	☒	☒	
15	PROD_B_REAL	Real	%MD46	☐	☒	☒	☒	
16	PROD_C_REAL	Real	%MD50	☐	☒	☒	☒	
17	TOTAL_PROD_TID_REAL	Real	%MD54	☐	☒	☒	☒	
18	TOTAL_PROD_TID_INT	Int	%MW58	☐	☒	☒	☒	
19	ANTAL_PROD_INT	Int	%MW60	☐	☒	☒	☒	
20	ANTAL_PROD_REAL	Real	%MD62	☐	☒	☒	☒	
21	<Add new>			☐	☒	☒	☒	

Observera datatypen.

16. Komplettera med följande funktion.
Variabeln **ANTAL_PROD_INT** som är ett integer (INT) ska konverteras till ett flyttal (REAL).

ANTAL_PROD_REAL

För att konvertera använd operations CONVERT.

Konverteringen ska ske ovillkorligt.

Vid test får man mata in olika värden i **ANTAL_PROD_INT** genom en Watch table.

Spara, kompilera, ladda ned till CPU och testa.

17. Komplettera med följande funktion.

En maskin ska kunna producera 3 olika produkter (A, B och C). De olika produkterna har olika produktionstid, vilka är A=16,1 s, B=22,5 s och C=38,4 s

För att ställa om maskinen till ny produkttillverkning, tar det 25 min.

Med variabeln **PRODUKTVAL** ska man kunna välja vilken produkt man vill köra.

Värde på PRODUKTVAL	Produkttyp
1	A
2	B
3	C

För att identifiera aktuellt värde (1,2 eller 3) använd en COMPARATOR OPERATIONS av typen CMP= . Överst lägger man in den variabel man ska jämföras med och underst värdet man vill

jämföra mot. I detta fall, överst PRODUKTVAL och underst siffran 1. Man får göra 3 st sådana, en för varje (1,2 och 3).

För att ändra värdet i **PRODUKTVAL** använd watchtable.

Skapa ett program som beräknar total produktionstid för vald produkt utifrån antalet produkter man vill köra. De variabler som deklarerats bör räcka för uppgiften men om du behöver fler skapar du dessa självständigt.

Vid test ändra antalet produkter och även produkttyp.

Spara, kompilera, ladda ned till CPU och testa.

18. Komplettera i **PROGRAMBLOCK_1** följande funktion.

Värdet i **TOTAL_PROD_TID_REAL** ska konverteras till ett integer (INT) i **TOTAL_PROD_TID_INT**.

I denna variabel ska man kunna avläsa total produktionstid i sekunder.

OBS! Eftersom **TOTAL_PROD_TID_INT** är en 16-bitars heltalvariabel kan den maximalt hantera värdet +32767 sekunder. Om antalet är för stort kommer total tid att bli felaktig.

Spara, kompilera, ladda ned till CPU och testa.

Kontrollfrågor

1. Vad innebär kompilering?
2. Hur definieras en digital variabel som endast kan anta tillstånden 0 och 1?
3. Vad innebär prioritet när det gäller SET resp. REST-funktioner?
4. Hur definierar man konstanten 5 för en counter enligt IEC?
5. Hur definierar man tiden 1min och 15,3 s till en timer enligt IEC?
6. Hur fungerar den digitala utgången för en tillslagsfördröjd timer (TON) enligt IEC?
7. Hur fungerar den digitala utgången för en counter (CTU) enligt IEC?
8. Vad krävs till en counter för att den ska kunna starta om räkningen?
9. Hur definieras ett enkelt dataord om 16 bitar?
Vilka datatyper finns som är 16-bitar?
10. En typ av dataord (enkelord) kan hantera negativa och positiva tal, vilket?
11. Inom vilket talområde kan detta dataord hantera tal?
12. Hur ofta sker en funktion som t.ex. en ADD av enkelord som finns i ett program?
13. Hur kan man begränsa att en funktion t.ex. ADD endast sker en gång?
14. Beskriv i korta drag datatypen REAL.
15. Hur skriver man konstanten 125,75 i flyttal (REAL)?
16. Hur skrivs decimala talet 24 i BCD-format?

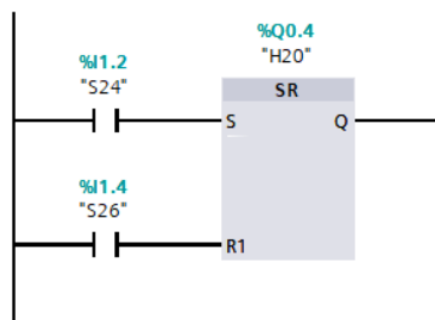
Svar kontrollfrågor

1. Vad innebär kompilering?
Alla delar i ett projekt översätts till nedladdningsbar kod.
2. Hur definieras en digital variabel som endast kan anta tillstånden 0 och 1?
BOOL
3. Vad innebär prioritet när det gäller SET resp. REST-funktioner?
Den instruktion som kommer sist i programmet för en given adress (variabel) kommer att ha prioritet om villkoren för både SET och RESET sker samtidigt.
4. Hur definierar man konstanten 5 för en counter enligt IEC?
5
5. Hur definierar man tiden 1min och 15,3 s till en timer enligt IEC?
T#1m15s300ms
6. Hur fungerar den digitala utgången för en tillslagsfördröjd timer (TON) enligt IEC?
Ingångsvillkoret (RLO) för timern måste vara uppfyllt hela tiden då timern arbetar. När ingångsvillkoret (RLO) till timern inte är uppfyllt går timerns utgång från.

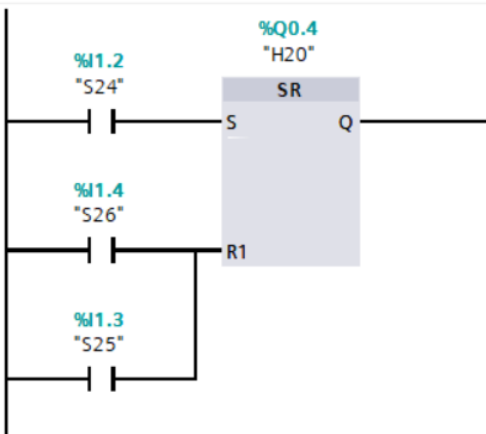
7. Hur fungerar den digitala utgången för en counter (CTU) enligt IEC?
När countern registrerat n antal signaler på ingången går counterns utgång till. Utgången är till tills det att counterns återställning aktiveras. Då rensas även det aktuella räknevärdet.
8. Vad krävs till en counter för att den ska kunna starta om räkningen?
Att den återställs.
9. Hur definieras ett enkelt dataord om 16 bitar?
WORD
Vilka datatyper finns som är 16-bitar?
WORD (unsign=ingen teckenbit) och INT (integer, sign=teckenbit MSB)
10. En typ av dataord (enkelord) kan hantera negativa och positiva tal, vilket?
INT (integer)
11. Inom vilket talområde kan detta dataord hantera tal?
WORD 0-65535 (65536) och INT -32768 -> +32767
12. Hur ofta sker en funktion som t.ex. en ADD av enkelord som finns i ett program?
Varje programcykelvarv och så länge ingångsvillkoret (RLO) till funktionen är uppfyllt.
13. Hur kan man begränsa att en funktion t.ex. ADD endast sker en gång?
Genom att avbryta ingångsvillkoret till ett programcykelvarv med en trig-funktion.
14. Beskriv i korta drag datatypen REAL.
Använder sig av en mantissa och en exponent vilket gör att decimalkommats position kan bestämmas. Dessutom har det en teckenbit som anger om talet är negativt eller positivt.
15. Hur skriver man konstanten 125,75 i flyttal (REAL)?
125.75 eller 1.25 e+2
16. Hur skrivs decimala talet 24 i BCD-format?
Varje siffra skrivs med 4 bitar i dekader. För decimala talet 24 = 0010 0100

Lösningar

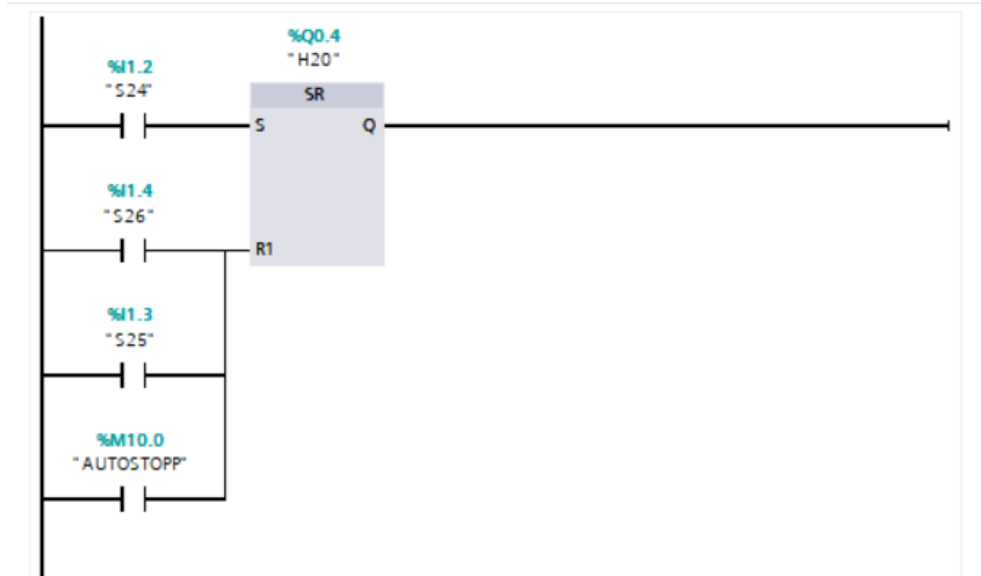
4.



5.

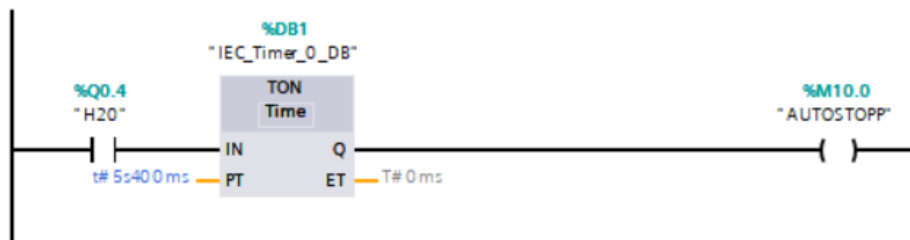


7.



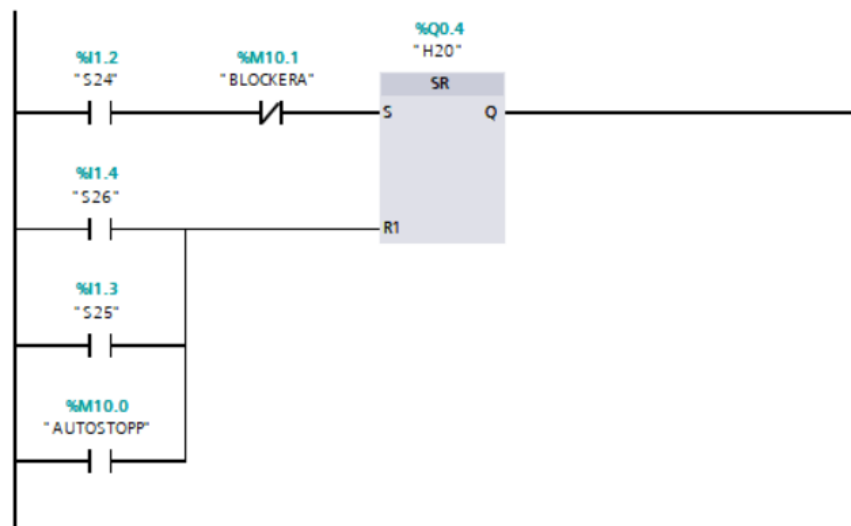
Network 2:

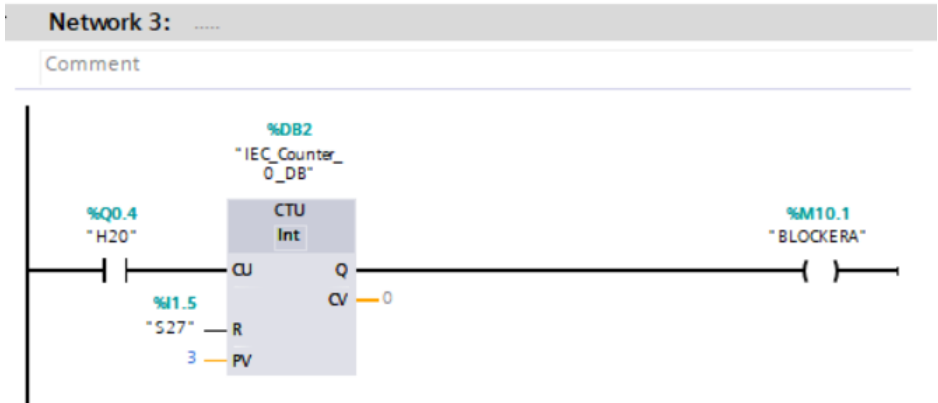
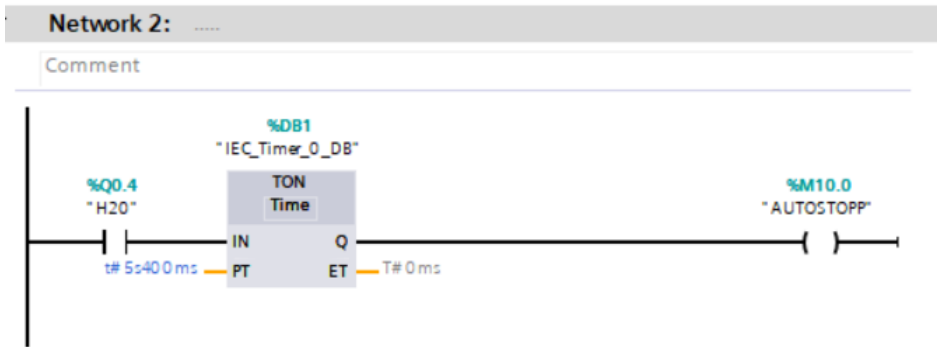
Comment



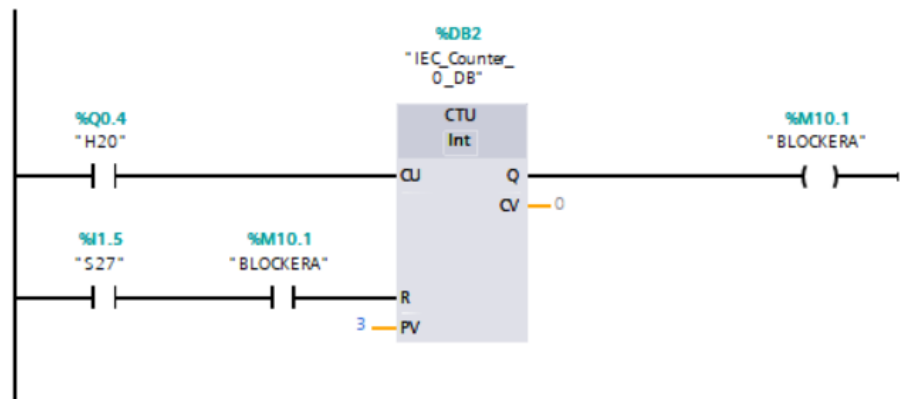
8.

Comment

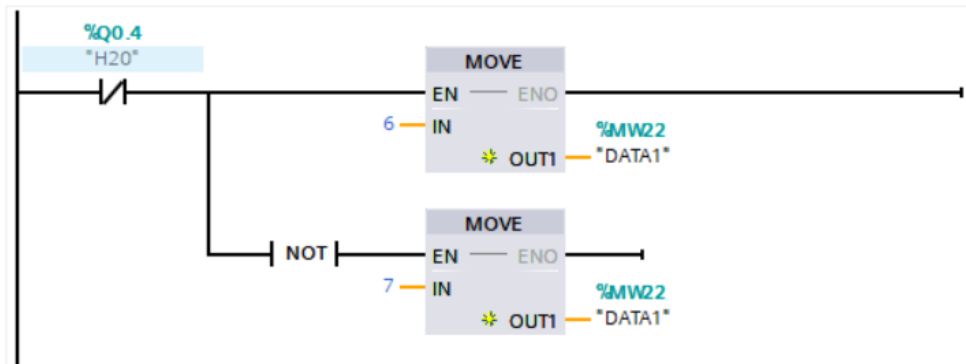




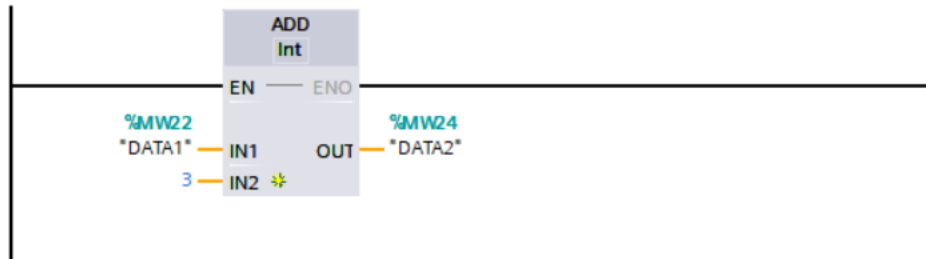
9.



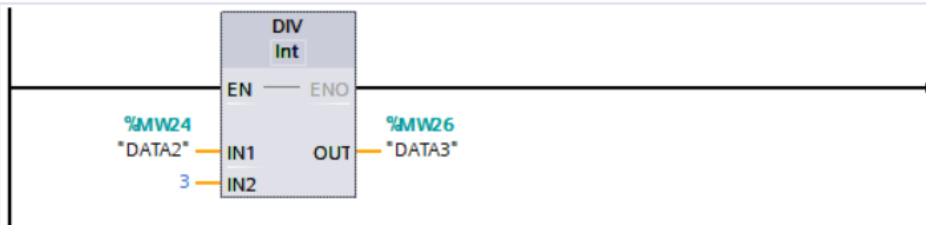
10.



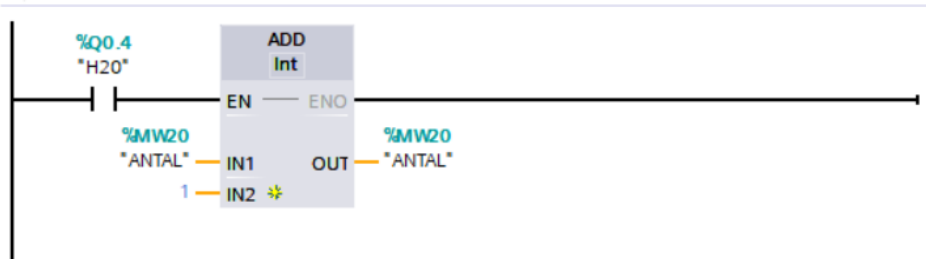
11.



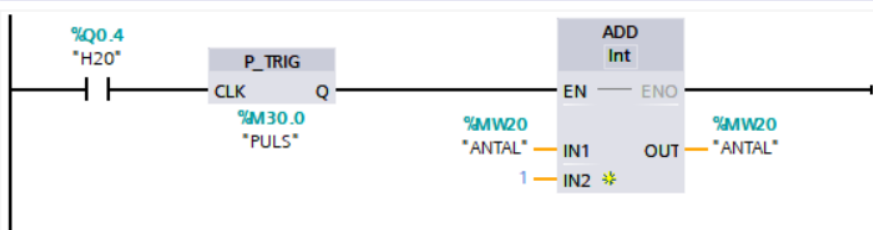
12.



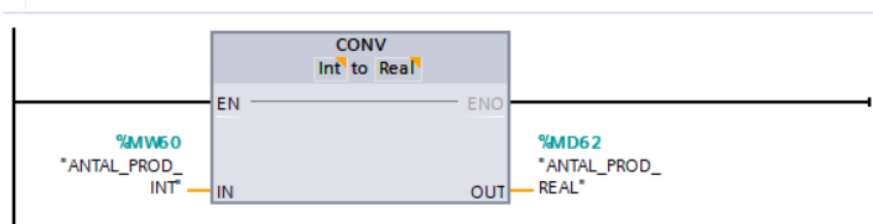
13.



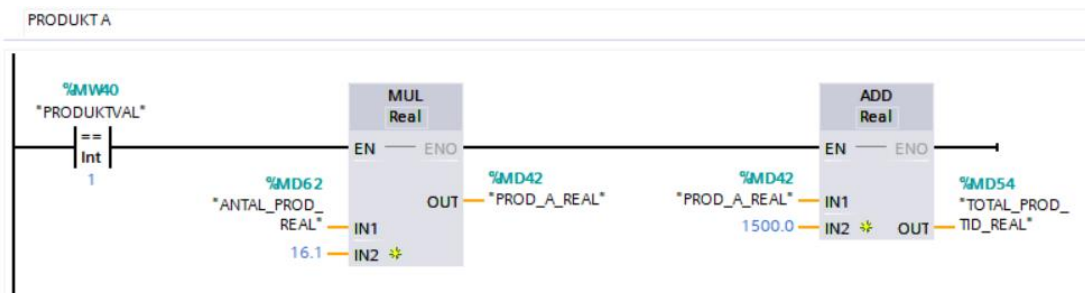
14.



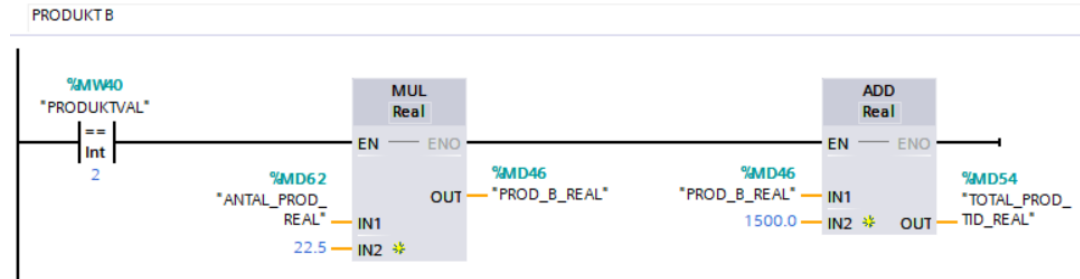
16.



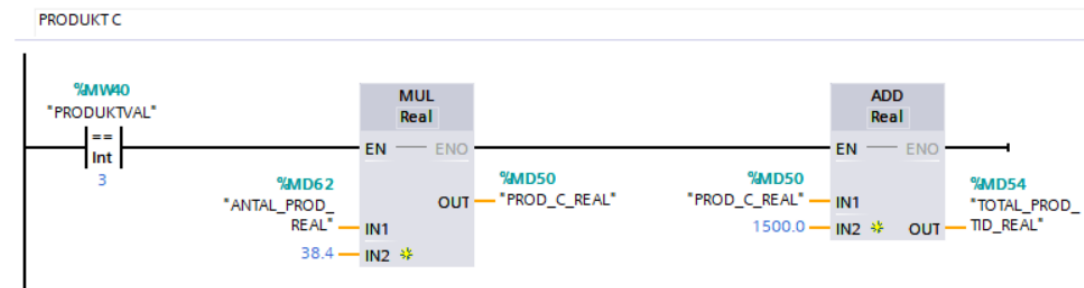
17. Produkt A



Produkt B



Produkt C



18.

