Tutorial

Siemens PLC

Simatic S7-300 STEP7 WinCC Flexible



SIEMENS

De Haagse Hogeschool Academie voor TIS / Delft 9 mei 2013

J.E.J. op den Brouw

Inhoudsopgave

1.	INL	LEIDING	4
2.	DE	PC	6
3.	DE	PLC	8
	3.1	BESCHRIJVING PLC	
-	3.1.	1 Processormodule CPU315F-2 PN/DP	9
	3.1.	2 Voedingsmodule PS307	9
	3.1.	3 Digitale I/O module SM323	
	3.1.	4 Analoge I/O module SM334	
	3.1.	5 Simulatie I/O module SM 374	
2	3.2	Programmeertalen	
	3.3	GEBRUIK PLC	
4.	TU	TORIAL STEP7 EN LAD	14
4	4.1	PROJECT AANMAKEN	14
4	4.2	NETWERK CONFIGUREREN	
4	4.3	PLC CONFIGUREREN	
4	4.4	CONFIGURATIE DOWNLOADEN	
4	4.5	Symbolen definiëren	
4	4.6	PROGRAMMA INVOEREN	
2	4.7	LADEN PLC PROGRAMMA	
4	4.8	MONITOREN VAN HET PROGRAMMA	
_			
5.	TU	TORIAL S7-GRAPH	
5.	ТU 5.1	TORIAL S7-GRAPH Project aanmaken	34
5.	TU 5.1 5.2	TORIAL S7-GRAPH Project aanmaken Configuratie en Blocks uit PLC halen	34
5.	TU 5.1 5.2 5.3	TORIAL S7-GRAPH Project aanmaken Configuratie en Blocks uit PLC halen Symbolenlijst opstellen	34
5.	TU 5.1 5.2 5.3 5.4	TORIAL S7-GRAPH Project aanmaken Configuratie en Blocks uit PLC halen Symbolenlijst opstellen Een blok vooraf	
5.	TU 5.1 5.2 5.3 5.4 5.5	TORIAL S7-GRAPH Project aanmaken Configuratie en Blocks uit PLC halen Symbolenlijst opstellen Een blok vooraf Blokken, stappen en overgangen	34 34 35 37 37 38
5.	TU 5.1 5.2 5.3 5.4 5.5 5.6	TORIAL S7-GRAPH Project aanmaken Configuratie en Blocks uit PLC halen Symbolenlijst opstellen Een blok vooraf Blokken, stappen en overgangen Aanmaken Function Block	34 34 35 37 37 37 38 38 39
5.	TU 5.1 5.2 5.3 5.4 5.5 5.6 5.7	TORIAL S7-GRAPH Project aanmaken Configuratie en Blocks uit PLC halen Symbolenlijst opstellen Een blok vooraf Blokken, stappen en overgangen Aanmaken Function Block Starten S7-Graph editor	34 34 35 37 37 37 38 39 40
	TU 5.1 5.2 5.3 5.4 5.5 5.6 5.7 5.8	TORIAL S7-GRAPH PROJECT AANMAKEN CONFIGURATIE EN BLOCKS UIT PLC HALEN SYMBOLENLIJST OPSTELLEN EEN BLOK VOORAF BLOKKEN, STAPPEN EN OVERGANGEN AANMAKEN FUNCTION BLOCK STARTEN S7-GRAPH EDITOR INVOEREN EERSTE STAP	34 34 35 37 37 37 38 39 40 41
	TU 5.1 5.2 5.3 5.4 5.5 5.6 5.7 5.8 5.9	TORIAL S7-GRAPH PROJECT AANMAKEN CONFIGURATIE EN BLOCKS UIT PLC HALEN SYMBOLENLIJST OPSTELLEN EEN BLOK VOORAF BLOKKEN, STAPPEN EN OVERGANGEN BLOKKEN, STAPPEN EN OVERGANGEN AANMAKEN FUNCTION BLOCK STARTEN S7-GRAPH EDITOR INVOEREN EERSTE STAP INVOEREN VOLGENDE STAPPEN	34 34 35 37 37 37 38 39 40 41 42
	TU 5.1 5.2 5.3 5.4 5.5 5.6 5.7 5.8 5.9 5.10	TORIAL S7-GRAPH PROJECT AANMAKEN CONFIGURATIE EN BLOCKS UIT PLC HALEN SYMBOLENLIJST OPSTELLEN EEN BLOK VOORAF BLOKKEN, STAPPEN EN OVERGANGEN AANMAKEN FUNCTION BLOCK STARTEN S7-GRAPH EDITOR INVOEREN EERSTE STAP INVOEREN VOLGENDE STAPPEN INVOEREN OB1	34 34 35 37 37 38 39 40 41 42 44
	TU 5.1 5.2 5.3 5.4 5.5 5.6 5.7 5.8 5.9 5.10 5.11	TORIAL S7-GRAPH PROJECT AANMAKEN CONFIGURATIE EN BLOCKS UIT PLC HALEN SYMBOLENLIJST OPSTELLEN EEN BLOK VOORAF BLOKKEN, STAPPEN EN OVERGANGEN AANMAKEN FUNCTION BLOCK STARTEN S7-GRAPH EDITOR INVOEREN EERSTE STAP INVOEREN VOLGENDE STAPPEN INVOEREN OB1 LADEN PLC MET PROGRAMMA	34 34 35 37 37 37 38 39 40 40 41 41 42 44
	TU 5.1 5.2 5.3 5.4 5.5 5.6 5.7 5.8 5.9 5.10 5.11 5.12	TORIAL S7-GRAPH. PROJECT AANMAKEN CONFIGURATIE EN BLOCKS UIT PLC HALEN SYMBOLENLIJST OPSTELLEN EEN BLOK VOORAF BLOKKEN, STAPPEN EN OVERGANGEN AANMAKEN FUNCTION BLOCK STARTEN S7-GRAPH EDITOR INVOEREN EERSTE STAP INVOEREN VOLGENDE STAPPEN INVOEREN OB1 LADEN PLC MET PROGRAMMA VARIABELENTABEL	34 34 35 37 37 38 39 40 40 41 42 44 45 45
	TU 5.1 5.2 5.3 5.4 5.5 5.6 5.7 5.8 5.9 5.10 5.11 5.12 5.13	TORIAL S7-GRAPH PROJECT AANMAKEN CONFIGURATIE EN BLOCKS UIT PLC HALEN SYMBOLENLIJST OPSTELLEN EEN BLOK VOORAF BLOKKEN, STAPPEN EN OVERGANGEN AANMAKEN FUNCTION BLOCK STARTEN S7-GRAPH EDITOR INVOEREN EERSTE STAP INVOEREN VOLGENDE STAPPEN INVOEREN VOLGENDE STAPPEN INVOEREN OB1 LADEN PLC MET PROGRAMMA VARIABELENTABEL MONITOREN VAN HET PROGRAMMA EN VARIABELEN	34 34 35 37 37 38 39 40 40 41 42 44 42 44 45 45 47
	TU 5.1 5.2 5.3 5.4 5.5 5.6 5.7 5.8 5.9 5.10 5.11 5.12 5.13 5.14	TORIAL S7-GRAPH PROJECT AANMAKEN CONFIGURATIE EN BLOCKS UIT PLC HALEN SYMBOLENLIJST OPSTELLEN EEN BLOK VOORAF BLOKKEN, STAPPEN EN OVERGANGEN AANMAKEN FUNCTION BLOCK STARTEN S7-GRAPH EDITOR INVOEREN EERSTE STAP INVOEREN VOLGENDE STAPPEN INVOEREN VOLGENDE STAPPEN INVOEREN OB1 LADEN PLC MET PROGRAMMA VARIABELENTABEL MONITOREN VAN HET PROGRAMMA EN VARIABELEN EXTRA OPDRACHT	34 34 35 37 37 38 39 40 40 41 41 42 44 45 45 45 47 47
5.	TU 5.1 5.2 5.3 5.4 5.5 5.6 5.7 5.8 5.9 5.10 5.11 5.12 5.13 5.14 TU	TORIAL S7-GRAPH PROJECT AANMAKEN CONFIGURATIE EN BLOCKS UIT PLC HALEN SYMBOLENLIJST OPSTELLEN EEN BLOK VOORAF BLOKKEN, STAPPEN EN OVERGANGEN AANMAKEN FUNCTION BLOCK STARTEN S7-GRAPH EDITOR INVOEREN EERSTE STAP INVOEREN VOLGENDE STAPPEN INVOEREN VOLGENDE STAPPEN INVOEREN OB 1 LADEN PLC MET PROGRAMMA VARIABELENTABEL MONITOREN VAN HET PROGRAMMA EN VARIABELEN EXTRA OPDRACHT TORIAL SCL	34 34 35 37 37 38 39 40 40 41 42 44 45 45 45 45 47 47 49
6.	TU 5.1 5.2 5.3 5.4 5.5 5.6 5.7 5.8 5.7 5.8 5.9 5.10 5.11 5.12 5.13 5.14 TU 6.1	TORIAL S7-GRAPH. PROJECT AANMAKEN. CONFIGURATIE EN BLOCKS UIT PLC HALEN. SYMBOLENLIJST OPSTELLEN. EEN BLOK VOORAF . BLOKKEN, STAPPEN EN OVERGANGEN . AANMAKEN FUNCTION BLOCK. STARTEN S7-GRAPH EDITOR . INVOEREN EERSTE STAP. INVOEREN VOLGENDE STAPPEN . INVOEREN OB1 . LADEN PLC MET PROGRAMMA. VARIABELENTABEL MONITOREN VAN HET PROGRAMMA EN VARIABELEN . EXTRA OPDRACHT . TORIAL SCL ANALOGE INGANG EN TEMPERATUUR METEN .	34 34 35 37 37 37 38 39 40 40 41 41 42 44 45 45 45 45 47 47 47 49
6.	TU 5.1 5.2 5.3 5.4 5.5 5.6 5.7 5.8 5.9 5.10 5.11 5.12 5.13 5.14 TU 6.1 5.2	TORIAL S7-GRAPH PROJECT AANMAKEN CONFIGURATIE EN BLOCKS UIT PLC HALEN SYMBOLENLIJST OPSTELLEN EEN BLOK VOORAF BLOKKEN, STAPPEN EN OVERGANGEN AANMAKEN FUNCTION BLOCK STARTEN S7-GRAPH EDITOR INVOEREN EERSTE STAP INVOEREN VOLGENDE STAPPEN INVOEREN OB1 LADEN PLC MET PROGRAMMA VARIABELENTABEL MONITOREN VAN HET PROGRAMMA EN VARIABELEN EXTRA OPDRACHT TORIAL SCL ANALOGE INGANG EN TEMPERATUUR METEN PROJECT AANMAKEN	34 34 35 37 37 37 38 39 40 40 41 41 42 44 45 45 45 47 47 47 47 47 49 50
6.	TU 5.1 5.2 5.3 5.4 5.5 5.6 5.7 5.8 5.9 5.10 5.11 5.12 5.13 5.14 TU 5.13 5.14 5.12 5.13 5.14	TORIAL S7-GRAPH PROJECT AANMAKEN CONFIGURATIE EN BLOCKS UIT PLC HALEN SYMBOLENLIJST OPSTELLEN EEN BLOK VOORAF BLOKKEN, STAPPEN EN OVERGANGEN AANMAKEN FUNCTION BLOCK STARTEN S7-GRAPH EDITOR INVOEREN EERSTE STAP INVOEREN VOLGENDE STAPPEN INVOEREN OB1 LADEN PLC MET PROGRAMMA VARIABELENTABEL MONITOREN VAN HET PROGRAMMA EN VARIABELEN EXTRA OPDRACHT TORIAL SCL ANALOGE INGANG EN TEMPERATUUR METEN PROJECT AANMAKEN CONFIGURATIE PLC	34 34 35 37 37 38 39 40 40 41 42 44 45 45 47 47 47 47 47 47 50 50 50
6.	TU 5.1 5.2 5.3 5.4 5.5 5.6 5.7 5.8 5.7 5.8 5.7 5.8 5.10 5.11 5.12 5.13 5.14 TU 5.13 5.14 TU 5.3 5.4	TORIAL S7-GRAPH. PROJECT AANMAKEN CONFIGURATIE EN BLOCKS UIT PLC HALEN SYMBOLENLIJST OPSTELLEN EEN BLOK VOORAF BLOKKEN, STAPPEN EN OVERGANGEN AANMAKEN FUNCTION BLOCK STARTEN S7-GRAPH EDITOR INVOEREN EERSTE STAP INVOEREN VOLGENDE STAPPEN INVOEREN VOLGENDE STAPPEN INVOEREN OB1 LADEN PLC MET PROGRAMMA VARIABELENTABEL MONITOREN VAN HET PROGRAMMA EN VARIABELEN EXTRA OPDRACHT TORIAL SCL ANALOGE INGANG EN TEMPERATUUR METEN PROJECT AANMAKEN CONFIGURATIE PLC SYMBOLENLIJST	34 34 35 37 37 37 38 39 40 40 41 41 42 44 45 45 47 47 47 47 47 47 50 50 50 53

6.6	INVOEREN SCL-CODE	54
6.7	COMPILEREN SCL-CODE	
6.8	TWEEDE SCL-BRONBESTAND	57
6.9	OB1 programma	
6.10	LADEN PLC PROGRAMMA	
6.11	MONITOREN TEMPERATUUR	
7. T	UTORIAL WINCC FLEXIBLE	61
7.1	AANMAKEN NIEUW PROJECT	61
7.2	CONFIGURATIE UPLOADEN	
7.3	HMI STATION PLAATSEN	64
7.4	MPI-NETWERK	65
7.5	HMI-STATION CONFIGUREREN	
7.6	SYMBOLENLIJST	
7.7	STARTEN WINCC FLEXIBLE	
7.8	DEVICE SETTINGS	
7.9	VERDINIC MET DI C MAKEN	
7.10	V ERBINDING MET FLC MAKEN	74 74
7.11	SCHEDM ODDOUWEN	
7.12	STARTEN RUNTIME	
7.14	AFSLUITEN RUNTIME	82
е т		92
o. I	UTORIAL DATA BLOCK	
8.1	AANMAKEN DATA BLOCK	
8.1 8.2	AANMAKEN DATA BLOCK Een stuk je SCL-code	
8.1 8.2 8.3	Aanmaken Data Block Een stuk je SCL-code Symbolen, OB1 en downloaden Monitoppic	
8.1 8.2 8.3 8.4 8 5	Aanmaken Data Block Een stuk je SCL-code Symbolen, OB1 en downloaden Monitoring Instance Data Block	
8.1 8.2 8.3 8.4 8.5	Aanmaken Data Block Een stuk je SCL-code Symbolen, OB1 en downloaden Monitoring Instance Data Block	
8.1 8.2 8.3 8.4 8.5 9. T	Aanmaken Data Block Een stuk je SCL-code Symbolen, OB1 en downloaden Monitoring Instance Data Block IPS, TRICKS & TROUBLESHOOT	
8.1 8.2 8.3 8.4 8.5 9. T 9.1	AANMAKEN DATA BLOCK EEN STUK JE SCL-CODE SYMBOLEN, OB1 EN DOWNLOADEN MONITORING INSTANCE DATA BLOCK IPS, TRICKS & TROUBLESHOOT PROJECTEN ARCHIVEREN	
8.1 8.2 8.3 8.4 8.5 9. T 9.1 9.2	AANMAKEN DATA BLOCK EEN STUK JE SCL-CODE SYMBOLEN, OB1 EN DOWNLOADEN MONITORING INSTANCE DATA BLOCK INSTANCE DATA BLOCK IPS, TRICKS & TROUBLESHOOT PROJECTEN ARCHIVEREN PROJECTEN INLEZEN	
8.1 8.2 8.3 8.4 8.5 9. T 9.1 9.2 9.3	AANMAKEN DATA BLOCK EEN STUK JE SCL-CODE SYMBOLEN, OB1 EN DOWNLOADEN MONITORING INSTANCE DATA BLOCK IPS, TRICKS & TROUBLESHOOT PROJECTEN ARCHIVEREN PROJECTEN INLEZEN PG/PC INTERFACE.	
8.1 8.2 8.3 8.4 8.5 9. T 9.1 9.2 9.3 9	AANMAKEN DATA BLOCK EEN STUK JE SCL-CODE	
8.1 8.2 8.3 8.4 8.5 9. T 9.1 9.2 9.3 9	AANMAKEN DATA BLOCK EEN STUK JE SCL-CODE	83 85 86 87 88 88 89 89 91 93 93 93 95 96
8.1 8.2 8.3 8.4 8.5 9. T 9.1 9.2 9.3 9 9.4 9.4	AANMAKEN DATA BLOCK	83 85 86 87 88 88 89 89 91 93 93 93 93 93 93 93
8.1 8.2 8.3 8.4 8.5 9. T 9.1 9.2 9.3 9 9.4 9.5 9.6	AANMAKEN DATA BLOCK EEN STUK JE SCL-CODE SYMBOLEN, OB1 EN DOWNLOADEN MONITORING	83 85 86 87 88 89 89 91 93 93 93 93 95 96 97 98
8.1 8.2 8.3 8.4 8.5 9. T 9.1 9.2 9.3 <i>9</i> 9.4 9.5 9.6	AANMAKEN DATA BLOCK	83 85 86 87 88 88 89 91 93 93 93 93 93 93 93 93 93
8.1 8.2 8.3 8.4 8.5 9. T 9.1 9.2 9.3 9 9.4 9.5 9.6 10.	AANMAKEN DATA BLOCK	83 85 86 87 88 88 89 91 91 93 93 93 93 93 93 93 95 96 97 98 98 99
8.1 8.2 8.3 8.4 8.5 9. T 9.1 9.2 9.3 9.3 9 9.4 9.5 9.6 10. BIII	AANMAKEN DATA BLOCK EEN STUK JE SCL-CODE SYMBOLEN, OB1 EN DOWNLOADEN MONITORING INSTANCE DATA BLOCK IPS, TRICKS & TROUBLESHOOT PROJECTEN ARCHIVEREN PROJECTEN INLEZEN PG/PC INTERFACE 3.1 PC/PG interface via de USB adapter 3.2 PC/PG interface omzetten PLC DIAGNOSTIEK ONLINE BLOKKEN IN DE PLC VERWIJDEREN CLOCK MEMORY BYTE BIJLAGEN AGE 1 – LAD-PROGRAMMA VERKEERSLICHTSYSTEEM	83 85 86 87 88 88 89 91 91 93 93 93 93 93 93 93 95 96 97 98 99 99
8.1 8.2 8.3 8.4 8.5 9. T 9.1 9.2 9.3 9 9.4 9.5 9.6 10. BIJI	AANMAKEN DATA BLOCK	83 85 86 87 88 89 91 91 93 93 93 93 93 95 95 96 97 98 98 99 99 102
8.1 8.2 8.3 8.4 8.5 9. T 9.1 9.2 9.3 9 9.4 9.5 9.6 10. BIJI BIJI	AANMAKEN DATA BLOCK EEN STUK JE SCL-CODE SYMBOLEN, OB1 EN DOWNLOADEN MONITORING INSTANCE DATA BLOCK IPS, TRICKS & TROUBLESHOOT PROJECTEN ARCHIVEREN PROJECTEN INLEZEN PG/PC INTERFACE 3.1 PC/PG interface via de USB adapter 3.2 PC/PG interface omzetten PLC DIAGNOSTIEK ONLINE BLOKKEN IN DE PLC VERWIJDEREN CLOCK MEMORY BYTE BIJLAGEN AGE 1 – LAD-PROGRAMMA VERKEERSLICHTSYSTEEM AGE 2 – GRAPH PROGRAMMA AGE 3 – PLC BLOCKS, INPUT- EN OUTPUTADRESSEN, MEMORY	83 85 86 87 88 89 91 91 93 93 93 93 93 93 93 93 93 95 96 97 98 99 99 102 104
8.1 8.2 8.3 8.4 8.5 9. T 9.1 9.2 9.3 9 9.4 9.5 9.6 10. BIJI BIJI BIJI	AANMAKEN DATA BLOCK	83 85 86 87 88 89 91 91 93 93 93 93 93 93 93 95 96 97 98 99 99 102 102 104
8.1 8.2 8.3 8.4 8.5 9. T 9.1 9.2 9.3 9 9.4 9.5 9.6 10. BIJI BIJI BIJI BIJI	AANMAKEN DATA BLOCK	83 85 86 87 88 89 91 93 93 93 93 93 93 93 93 93 93 93 93 93

Voor suggesties en/of opmerkingen over deze handleiding kan je je wenden tot J. op den Brouw, kamer 411, of je kunt E-mail versturen naar: J.E.J.opdenBrouw@hhs.nl.

1. Inleiding

Het zal de meeste mensen niet opvallen, maar veel van onze producten worden gemaakt in fabrieken. We denken hierbij aan bijvoorbeeld mobiele telefoons, microprocessoren maar ook colaflessen (en het vullen ervan).

Het vervaardingen van deze producten wordt, omdat er massaproductie nodig is, gedaan door middel van een productielijn, in de volksmond lopende band genoemd. Vroeger waren hier grote groepen arbeiders voor nodig, tegenwoordig worden veel van deze klussen gedaan door machines.

Deze machines moeten bestuurd worden. Het besturen wordt gedaan door Programmable Logic Controllers, afgekort tot PLC¹. Een PLC is te beschouwen als een computer met gespecialiseerde I/O. De computer, bestaande uit een microprocessor en geheugen, is te programmeren. De I/O wordt gebruikt om de machines aan te sturen en informatie betreffende de voorgang van het proces te vergaren.

De Academie voor Technology, Innovation and Society / Delft beschikt over een schaalmodel van een productiesysteem van Festo en PLC's van Siemens. Deze beide componenten worden verderop toegelicht. Zie figuur 1-1.



Figuur 1-1

Noot: de PC wordt voor meedere doeleinden gebruikt.

¹ In de Verenigde Staten wordt veelal de term Programmable Controller gebruikt, afgekort tot PC. Dit geeft echter verwarrring omdat een ander veel voorkomend apparaat, de Personal Computer, óók wordt afgekort tot PC.

Een korte introductie van de blokken:

- PLC (Programmable Logic Controller), de aanstuurder van (een deel van) het productiesysteem.,
- Productiesysteem, de te besturen productieomgeving, bv. van de firma Festo,
- PC, de computer waarop de software voor de PLC wordt ontwikkeld, tevens monitoring systeem.
- HMI-station/Operator Panel, een terminal waarop de engineer/operator het productieproces kan volgen.

Tussen de diverse blokken is communicatie mogelijk:

- PLC↔Productiesysteem, meetsignalen en stuursignalen (gezien vanuit de PLC).
- PLC↔PC, programmeerkabel, communcatie d.m.v. MPI of Ethernet. Deze kabel kan voor meer doeleinden gebruikt worden, maar in eerste instantie voor programmeren.
- PLC↔HMI-station/OP, communicatie d.m.v. MPI, Profibus en Profinet.

Deze handleiding begeleidt de cursist aan de hand van diverse tutorials door de veelzijdigheid van de STEP7-programmeeromgeving. De eerste tutorial laat een ladderprogramma zien voor een eenvoudig verkeerslichtsysteem. In de tweede tutorial wordt het programma opnieuw ingevoerd, maar nu in S7-Graph, tevens is hier een uitbreiding wat betreft functionaliteit van het programma. De derde tutorial betreft iets geheel anders; met behulp van een NTC-weerstand en een weerstand wordt een thermometer ontwikkeld. Het bijbehorende programma is in SCL geschreven. De vierde tutorial gaat over WinCC Flexible, het SCADA-pakket. De thermometer uit de vierde tutorial dient als basis. De laatste tutorial gaat over het gebruik van Data Blocks. Als laatste volgen enkele Tips, Tricks & Troubleshoot.

Hoe moet dit document gelezen worden:

In de eerste twee tutorials wordt de lezer "aan de hand meegenomen". Van bijna alles wat de lezer moet invullen of aanklikken is een *screenshot* weergegeven. Vanwege de layout zijn sommige screenshots na de daarwerkelijke tekst geplaatst, bijvoorbeeld op de volgende bladzijde. De lezer kan het best een alinea per keer lezen om een beeld te krijgen van de te verrichten handelingen.

Noot betreffende de plaats waar de projectbestanden moeten worden opgeslagen:

Maak geen gebruik van een USB-stick tijdens het werken met de Simatic software. Er worden zeer veel bestanden aangemaakt en er is veel verkeer tussen harddisk en de programmatuur. Het gebruik van de H:-schijf kan wel (is niet echt onderzocht). Gebruik de D:-schijf en archiveer naar de H:-schijf of USB-stick. Archiveren wordt besproken in hoofdstuk 9.

2. De PC

De PC is een gewone IBM compatible computer en heeft in dit project meerdere functies: het is een ontwikkelstation om de PLC te configureren en programmeren, het is een ontwikkelstation voor de SCADA software en het is een monitoring station voor SCADA. Dit is in figuur 2-1 schematisch weergegeven.



Figuur 2-1

Aan de PC is een PC Adapter USB van Siemens gekoppeld. Hiervoor zijn drivers geinstalleerd. De PC is met de PLC verbonden via MPI, een Siemens-eigen bus en protocol. Deze interface kan omgeschakeld worden naar Profibus, een gestandaardiseerd bussysteem dat gebruikt wordt in de automatiseringstechniek. Beide maken gebruik van het RS-485 protocol. De PC is ook uitgerust met een Ethernetkaart. Communicatie tussen de PLC en PC, bv als HMI-station, kan ook via de Ethernetkaart. Op de PC is de volgende software geinstalleerd.

Windows XP SP2

Eén van de Operating Systems van Microsoft, aangevuld met de laatste Service Packs.

Simatic STEP 7 5.4 Professional 2006

Dit is de software van Siemens waarmee het PLC-programma ontwikkeld wordt en waarmee dat programma in de PLC geladen wordt. Dit laden gebeurt via de MPI-bus. De professional versie heeft naast de vertrouwde programmeertalen LAD, FDB en STL ook Graph en SCL.

WinCC Flexible 2005 Engineering Station

Deze software van Siemens wordt gebruikt voor procesvisualisatie (HMI = Human Machine Interface) en toezichthouderscontrole (SCADA = Supervisory Control And Data Aquisition). WinCC Flexible ES is het deel van het totale pakket waarmee een *runtime* ontworpen wordt. Hier kan je dus knoppen, I/O-velden, etc. aanmaken en acties koppelen aan die knoppen en I/O velden. Is een runtime ontworpen, dan wordt het gecompileerd en kan het draaien in de WinCC Flexible Runtime (RT).

WinCC Flexible 2005 Runtime

WinCC Flexible Runtime is de component die de gecompileerde code uitvoert en daadwerkelijk interactie met de PLC vertoont. Het zal dus gegevens uit de PLC opvragen en bijvoorbeeld in een trenddatabase loggen. Zo kan de gebruiker informatie over langere tijd te vergaren en opslaan om bijvoorbeeld historische trends te analiseren, zodat het productieproces verbeterd kan worden. De PC is te gebruiken als HMI-station.

3. De PLC

Het aansturen van fabrieksmachines wordt gedaan met een Programmable Logic Controller, afgekort tot PLC. We maken gebruik van één uit de familie van de S7-300 serie van Siemens.

3.1 Beschrijving PLC

Het exemplaar dat tijdens dit blok wordt gebruikt is de CPU315F-2 PN/DP, gecombineerd met voeding, digitale en analoge I/O-module en een simulatiemodule. Een foto van de opstelling is te zien in figuur 3-1.





3.1.1 Processormodule CPU315F-2 PN/DP

Dit is het hart van het systeem. Op de foto is deze module in het middel gesitueerd. Het bevat een CPU-eenheid en een FLASH-opslagkaart. De CPU kan worden geprogrammeerd door middel van STEP7 software. De module heeft naast een MPI/Profibus-aansluiting ook een Ethernet aansluiting voor Industrial Ethernet of Profinet.



De volgende onderdelen zijn (figuur 3-2):

- 1. Bus Fail; statusleds betreffende de bussen.
- 2. Diverse statusleds, oa. System Fail, Run en Stop mode.
- 3. Het MMC-slot. Hierin wordt de MMCkaart geplaatst. Zonder deze kaart werkt de PLC niet.
- 4. Operation Switch. Zet de PLC in RUN of STOP.
- 5. Leds van de Ethernet Link.
- 6. De Ethernet/Profinet-aansluiting.
- 7. Voedingsspanning aansluiting (+24V)
- 8. De MPI/Profibus-aansluiting

Figuur 3-2

3.1.2 Voedingsmodule PS307

Het geheel wordt gevoed met een aparte voeding, links op de foto. De voeding kan 5A leveren en is krachtig genoeg om de CPU en de I/O-modules aan te sturen.



Figuur 3-3

De volgende onderdelen zijn (figuur 3-3):

- 1. "24 Volt aanwezig"-lamp.
- 2. Aansluitingen voor 24V DC.
- 3. Trekontlasting.
- 4. Spanningaansluiting 230V AC en aarde.
- 5. 24 V DC aan-uit schakelaar.

3.1.3 Digitale I/O module SM323

Deze module, rechts naast de CPU-module, heeft 16 digitale ingangen en uitgangen. De module werkt op 24V gelijkspanning. Voor zowel de ingangen als de uitgangen geldt: een logische '0' komt overeen met 0 V, een logische '1' komt overeen met 24 V. De uitgangen kunnen maximaal 0,5 A leveren.



De onderdelen zijn (figuur 3-4):

- 1. Kanaalnummer.
- 2. Statusled (groen indien logische '1').
- 3. Backplane

Figuur 3-4

Links op figuur 3-4 is de voorkant van de module afgebeeld. Links zijn de ingangen, rechts de uitgangen gesitueerd.

3.1.4 Analoge I/O module SM334

Deze module staat rechts naast de digitale I/O-module op de foto. Het heeft 4 analoge ingangen en 2 analoge uitgangen. De ingangen kunnen d.m.v spanning of stroom worden aangestuurd. De uitgangen kunnen een spanning of een stroom leveren. Het spanningsbereik ligt tussen 0 en 10 V, het stroombereik tussen 0 en 20 mA. Aangezien de PLC verder een digitaal systeem is, moeten de analoge waarden worden geconverteerd. Er wordt gebruik gemaakt van 8 bits resolutie voor zowel ingangen als uitgangen.

De digitale waarden bestaan allemaal uit 16 bit, ook al worden maar 8 bits effectief gebruikt. De analoge spanningen zijn allemaal positief, het tekenbit is dan ook 0. Daarna volgen de 8 bits. De overige bits zijn allemaal 0. Zie ook tabel 3-1.

Bitnummer	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Waarde	0	v	v	v	v	V	v	v	v	0	0	0	0	0	0	0

Tabel 3-1

Voor zowel spanning als stroom geldt een "rated range", het bereik waarin normaal gewerkt wordt. Mocht een spanning of stroom daarboven komen, dan wordt over de "overshoot range" gesproken. Bij een nog hogere spanning of stroom wordt gesproken over "overflow". Zie ook tabel 3-2.

Decimaal	Hexadecimaal	Spanning	Stroom	Opmerking
32767	7FFF	11.852 V*	23,70 mA*	Overflow*
32512	7F00			Overnow ·
32511	7EFF	11,759 V	23,52 mA	Overshoot
27649	6C01			Overshoot
27648	6C00	10,000 V	20,00 mA	
20736	5100	7,500 V	15,00 mA	Deted repos
128	80	46,296 mV	92,59 uA	Kated lange
0	0	0V	0 mA]

* zie tekst

Tabel 3-2

Indien een uitgang in de overflow range wordt gestuurd, levert de uitgang een spanning van 0V of een stroom van 0 mA. Indien een ingang in de overflow range wordt gestuurd levert dit een waarde van 7FFF op.



De onderdelen zijn (figuur 3-5):

- 1. Interne voeding.
- 2. Analoog-digitaal converter.
- 3. Ingangen: spanningsmeting.
- 4. Uitgangen: spanningaansluiting
- 5. Digitaal-analoog converter.
- 6. Backplane
- 7. Equipotential bonding.
- 8. Functionele aarding.

Figuur 3-5

3.1.5 Simulatie I/O module SM 374

Geheel rechts op de foto is een I/O-module te zien, waarmee digitale ingangen en uitgangen gesimuleerd kunnen worden. Nu is simuleren niet helemaal het juiste woord, het zijn wel gewoon in- en uitgangen. Alleen zijn de ingangen te bedienen door een gebruiker die hiermee de werking van een machine kan simuleren. De module is te gebruiken als:

- 16x ingang via schakelaars,
- 16x uitgang via leds,
- 8x ingang en 8x uitgang gecombineerd.

In deze laatste modus wordt de module gebruikt tijdens de tutorials.



De onderdelen in figuur 3-6 zijn:

- 1. Schakelaars.
- 2. Modus-selector.
- 3. Kanaalnummer.
- 4. Leds

Tijdens de tutorials wordt de 8x input / 8x output modus gebruikt. De selector moet daarvoor in het midden staan. Het bovenste gedeelte werkt als 8x output; de schakelaars hebben geen functie. Het onderste gedeelte werkt als 8x input; de leds geven de stand van de schakelaars weer.

Figuur 3-6

3.2 Programmeertalen

De PLC is ontstaan in de energie- en automatiseringstechniek en niet bij de "echte" computerhoek. De programmeurs hadden geen opleiding tot computerprogrammeurs gevolgd. Hierdoor ontbreekt een echte programmeertaal. In de loop der jaren zijn diverse methoden van programmeren ontworpen. Noot: hieronder volgt slechts een korte beschijving. Voor meer informatie, zie het boek en sheets.

De internationale norm IEC 61131-3² beschrijft een aantal programmeertalen. Hieronder een overzicht. In de opsomming wordt eerste de IEC naam gegeven en tussen de haakjes de IEC afkorting en de afkorting die Siemens gebruikt.

Ladder Diagram (LD, LAD)

Deze taal bestaat uit een set van symbolische instructies die op een grafische wijze worden gepresenteerd. Er zijn vijf catagorieen van instructies: relais³, timer/counter, rekenkundig, data manipulatie en programmabesturing. Deze symbolen kunnen zo worden opgesteld dat de gewenste werking in het geheugen wordt ingevoerd. Ladderdiagrammen worden gebruikt voor het besturen van (kontakt-)uitgangen gebaseerd op ingangscondities.

Function Block Diagram (FBD, FBD)

Dit is een grafische taal waarbij de logische functies als blokken worden weergegeven. Naast de bekende AND en OR zijn er ook blokken voor teller, timers enz. Het is een alternatief voor ladderdiagrammen.

Instruction List (IL, STL)

Dit is het best te omschrijven als een assemblertaal voor PLC's. Diverse instructies zoals L (load) en T (output) zijn voor handen. Hierin is het mogelijk om zeer nauwkeurig de werking van een

² Zie http://www.plcopen.org

³ Vroeger waren de uitgangen van relais voorzien. Tegenwoording worden de uitgangen ook met transistoren of thyristoren uitgevoerd. Men heeft echter de oude terminologie aangehouden.

programma te beschrijven omdat de instructies één op één worden vertaald naar machinetaal. STL staat voor Statement List. STL is <u>niet</u> conform de IEC-norm.

Structured Text (ST, SCL)

Deze taal lijkt erg op Pascal. Er kan geprogrammeerd worden als een echte programmeertaal, compleet met IF's, functieaanroepen, verschillende datatypes etc. SCL staat voor Structured Control Language. SCL is conform de IEC norm, en heeft enkele uitbreidingen.

Sequential Function Chart (SFC, Graph)

Ook dit is weer een grafische taal. Het is te vergelijken met het beschrijven van een toestandsmachine uit de digitale techniek; de condities van de overgangen worden beschreven d.m.v. LD's of FDB's. Aftakkingen (Branch) en parallelisme (Simultaneous Branch) zijn mogelijk. Siemens heeft een variant, S7 Graph genaamd, die voldoet aan de IEC 61131-3 norm. Daarvoor moet wel een optie worden geactiveerd.

Siemens levert een softwareapplicatie, genaamd STEP7 Professional, waarin al deze programmeertalen zijn opgenomen.

3.3 Gebruik PLC

De PLC is een robuust apparaat. Dat neemt niet weg dat het gebruik zorgvuldig dient te geschieden. De PLC kan tijdens de tutorials en werkzaamheden gewoon onder spanning staan. Wel is het raadzaam om de PLC in STOP te zetten als er wijzigingen in het geheugen worden gedaan. Hierbij valt te denken aan configuratie-aanpassing, nieuwe blokken verzenden.

Hieronder staan de operating modes van de PLC afgebeeld. Zie figuur 3-7.



Figuur 3-7

1.	Na aanzettten komt de CPU in STOP.	2.	De CPU gaat naar STARTUP als de
			schakelaar in RUN wordt gezet.
3.	De CPU komt in STOP als de schakelaar in	4.	Hold mode (wordt niet besproken).
	STOP wordt gezet of de PLC een fout		
	detecteert.		
5.	Hold mode (wordt niet besproken).	6.	Hold mode (wordt niet besproken).
7.	De CPU komt in RUN als alles goed is	8.	De CPU komt in STOP als de
	gegaan.		schakelaar in STOP wordt gezet of
			de PLC een fout detecteert.
9.	Hold mode (wordt niet besproken).	10.	Hold mode (wordt niet besproken).

4. Tutorial STEP7 en LAD

In deze tutorial wordt uitgelegd hoe een project moet worden aangemaakt. Daarin wordt een eenvoudig verkeerslichtsysteem ingevoerd. Eerst wordt een nieuw project aangemaakt, dan worden de netwerk- en hardwareconfiguratie van de PLC opgesteld. Daarna wordt de symbolenlijst aangemaakt zodat het programmeren wat eenvoudiger wordt. Vervolgens wordt het programma ingevoerd en als laatste wordt het programma online gevolgd (*monitoren*).

STEP7 is een groot programma met zeer veel mogelijkheden. In deze tutorial wordt slechts een deel van de mogelijkheden getoond.

4.1 Project aanmaken

Klik op het pictogram Simatic Manager. Zie figuur 4-1. De manager wordt gestart.



Figuur 4-1

Direct hierna kan de Project Wizard verschijnen, maar dat hoeft niet. Deze wordt niet gebruikt, dus klik op **Cancel**. Zie figuur 4-2.



Figuur 4-2

Nu verschijnt de Simatic Manager die verder leeg is. Zie figuur 4-3.



Figuur 4-3

Klik nu op **File** \rightarrow **New** om een nieuw project te starten. Er wordt een venster geopend zoals in figuur 4-4. Vul de projectnaam tutorial in, vul eventueel de map in, en zorg dat de rest overeenkomt met de figuur. Klik op **OK**.

Noot: gebruik geen USB-stick tijdens deze tutorial, later kan het natuurlijk wel dienen als opslag.

New Project		×
User projects Libra	ries 🛛 Multiprojects	
Name	Storage path	
L Add to current mul	tiproject	
Na <u>m</u> e: Interial		Type:
Storage location		E Library
D:\PROJECTS\STEF	97	Browse
OK		Cancel Help

Figuur 4-4

Er verschijnt een vrijwel leeg projectscherm (figuur 4-5).

🔄 tutorial D:\PROJECTS\STE	P7\tutorial

Figuur 4-5

4.2 Netwerk configureren⁴

Eerst moet nu het netwerk waarin de PLC zich bevindt worden ingevoerd. Dubbelklik op de naam MPI(1), of klik op het pictogram **Configure Network** (figuur 4-6).





Het programma Netpro wordt gestart met daarin een oranje-rode lijn die de MPI-bus voorstelt. Aan de rechterkant van Netpro bevinden zich de onderdelen die aan het netwerk kunnen worden toegevoegd. Aangezien de PLC ook een Ethernet-aansluiting heeft, moet er een Industrial Ethernet worden toegevoegd. Dit is de groene lijn in figuur 4-7. Klik rechts op Subnets→Industrial Ethernet en sleep dit onder de MPI-bus.

Daarnaast moet ook een S7-300 PLC worden toegevoegd. Voor Netpro is alleen van belang dat het type wordt opgegeven; de exacte configuratie van de PLC wordt m.b.v een ander programma ingevoerd. Voeg een SIMATIC 300 toe zoals is weergegeven in figuur 4-7.

Noot: het programma plaatst de onderdelen links op het scherm, deze kunnen niet zichtbaar zijn. Met de scroll-bars kan dan worden gezocht naar de geplaatste onderdelen.

⁴ Hoewel dit een eenvoudige setup is, waarin deze stap kan worden overgeslagen, wordt toch het netwerk geconfigureerd. Bij grote omgevingen is deze stap wel nodig.



Figuur 4-7

4.3 PLC configureren

De PLC moet nu geconfigureerd worden. Hierbij wordt opgegeven waar de PLC uit bestaat en hoe de adressering van de ingangen en uitgangen in elkaar zit.

Dubbelklik op het onderdeel SIMATIC 300(1) uit figuur 4-7⁵. Er volgt nu een dialoog waarin wordt aangegeven dat het netwerk wordt opgeslagen. Klik op **OK**. Zie ook figuur 4-8.

Edit Oper	n Object (2263:19)	×
٩	In order to execute this action, changes will be saved in 'tutorial'	
<u>OK</u>	Cancel	

Figuur 4-8

⁵ Dit kan ook door eerst Netpro af te sluiten en via de Simatic Manager de hardware van de PLC te configureren.

Het hardware-configuratieprogramma HWConfig wordt gestart. Als eerste moet een rack worden toegevoegd. Dit is te kiezen in het rechterscherm via SIMATIC 300-RACK-300-Rail. Het resultaat is te zien in figuur 4-9.

HW Config - [SIMATIC 300(1) (Configuration) tutorial]								
Undow Help								<u>-9×</u>
	N?				_			
					-			
						Eind:		<u>м</u> † М і
						Profile: Sta	andard	•
						E - 📅 PROFIE	BUS DP	
		_					BUS-PA INET IO	
⇒(0) UR		T					IC 300	
2						📔 🕀 🧰 C7	, 2300	
3							PU-300	
· 4 5		-				📙 🕀 🧰 FM	1-300 Noway	
6							-300	
		-				🗎 🕀 🧰 M7	7-EXTENSION	
9						B B BA		
		-					Rail	
		-				E ⊕ SMATI	1-300 TIC 400	
							TC HMI Station	
					Ţ	I ⊕ - 🛄 SIMATI ⊕ - 🖳 SIMATI	TC PC Based Control 300/400 TC PC Station	
				Þ				
		,			_			
Slot Module Order number	Firmware	MPI address	I address	Q addre C				
					-			
3								
5								
6								
8					┛	Available in var	?U-UAAU rious lengths	<u>⁺≺</u>
9					•			
I Press F1 to get Help.								Chg //

Figuur 4-9

Nu moeten de voeding, de CPU-module, de digitale I/O, analoge I/O en de simulatiemodule worden toegevoegd.

Selecteer via SIMATIC $300 \rightarrow PS - 300 \rightarrow PS$ 307 5A de voeding PS 307 5A en plaats deze in regel 1 (*slot 1*). Selecteer daarna de CPU 315F-2 PN/DP eenheid en sleep die in slot 2. Alleen in deze slot mag de CPU voorkomen. Het ordernummer is 6ES7 315-2FH13-0AB0, het versienummer is 2.5.

Let op het juiste Siemens ordernummer en versienummer!

Voordat de CPU in slot 2 zichtbaar is, moeten eerst enige gegevens worden ingevuld voor het IPadres en het MPI-adres. Hiervoor worden nieuwe invoerdialogen gestart. Zie figuur 4-10.

Vul het IP-adres in zoals is aangegeven in figuur 4-10. Selecteer ook Ethernet(1) zodat de CPU aan dit netwerk verbonden wordt. Klik daarna op **OK**.

Properties - Ethernet interface PN-IO (R0/52	2.2)	×
General Parameters		1
	If a subnet is selected, the next available addresses are suggested.	
IP address: 192.168.0.1 Su <u>b</u> net mask: 255.255.256.0	Gateway © Do not use router © Use router Address: 192.168.0.1	
<u>S</u> ubnet:		
not networked Ethemet(1)	<u>N</u> ew	
	Properties	
	Delete	
OK	Cancel Help	

Figuur 4-10

In figuur 4-11 is te zien dat de CPU-module geplaatst is, evenals de Profinet-verbinding.

HW Config - [SIMATIC 300(1)	(Configuration) tutorial) w Options Window Help								
		B № ?							
									ㅋㅋ
							<u>F</u> ind:		M‡ M‡
Ethe	rnet(1): PROFINET-IO-System (100)					Profile:	Standard	
								- CPU 315F-2 PN/DP	
<u>= (0) U</u>	R							ES7 315-2FH13-UABU	
$\frac{1}{2}$	PS 307 5A		<u> </u>					V2.5	
xī.	MPI/DP						. E	W V2.6	
	PN-10 Port 1						t t	CPU 316-2 DP	
3							+		
4							. E	- 🔲 CPU 317F-2	
5							. E	- CPU 317F-2 PN/DP	
							. <u>+</u>		
8			-						
		<u>_</u>							
							. E	- 🔁 CPU M7	
							🛛 🕀 💼) FM-300	
						-	📗 🗄 😐	Gateway	
						•		I IM-300 NARATENCION	
								PS-300	
(0) UR									
Slot Module	Order number	Firmware	MPI address	I address	Q addre	C		📱 PS 307 10A	
1 S 307 5A	6ES7 307-1EA00-0AA0							PS 307 2A	
2 CPU 315F-2 PN/DP	6ES7 315-2FH13-0AB0	V2.5	2					📳 PS 307 5A	
XT NFL/DF			2	2047*		_		MAUN-300 SM-300	
X2 FN-10				2046"		_	E SI	MATIC 400	
3				2040"			🗍 🗄 – 🛄 SI	MATIC HMI Station	_
4							\$7-300 ar	d M7-300 CPUs	۰
5									
<u>6</u>						-			
I Press F1 to get Help									Cha /

Figuur 4-11

Nu moet ook nog de MPI-interface gekoppeld worden aan de MPI-bus. Dubbelklik in slot 2 onder onderdeel X1 op MPI/DP (zie figuur 4-10). Nu volgt een dialoogvenster als figuur 4-12. Zorg ervoor dat de type op MPI staat. Klik daarna op **Properties**.

Properties - MPI/DP -	(R0/S2.1) ×
General Addresses	Operating Mode Configuration Clock
Short Description:	MPI/DP
<u>N</u> ame:	MPI/DP
_ Interface	
<u>I</u> ype:	PI 🔽
Address: 2	
Networked: No	Properties
<u>C</u> omment:	
	A
OK	CancelHelp

Figuur 4-12

Er volgt een nieuw venster zoals figuur 4-13. Kies 2 als adres en selecteer MPI(1) als netwerk.

Properties - MPI interface MPI/DP (R0/S2.1)		×
General Parameters		
Address: 2 ▼ Highest address: 31 Transmission rate: 187.5 Kbps	If a subnet is selected, the next available add	ress is proposed.
<u>S</u> ubnet:		
not networked MPI(1) 187.5	Kbos	<u>N</u> ew
		P <u>r</u> operties
		Delete
OK	Car	ncel Help

Figuur 4-13

Klik daarna op OK. Nu wordt het dialoog uit figuur 4-12 weer zichtbaar. Klik hier ook weer OK.

Als laatste moeten de drie modules worden toegevoegd. Selecteer uit het rechterpaneel via SIMATIC 300-SM-300 deze modules.

Selecteer de juiste modules! Let goed op de simulatiemodule!

Het ordernummer voor de digitale I/O-module is 6ES7 323-1BL00-0AA0 (slot 4). Het ordernummer voor de analoge I/O-module is 6ES7 334-0CE0<u>1</u>-0AA0 (slot 5). Het ordernummer voor de simulatiemodule is 6ES7 323-1BH01-0AA0 (slot 6).

🔣 HW Config - [SIMATIC 300(1) (C	Configuration) tutorial]							×
In <u>Station</u> Edit Insert PLC View	Options <u>W</u> indow <u>H</u> elp						_ 8	×
D 🚅 🐎 🖩 🖏 🎒 🖻 🖻	. 🔬 🏫 🕞 🖴 😵	N?						
								×I
Ethoma	with DDOEINET IO Sustem (1	00)					lituo: lituo:	4
Etheme	s(1). FROFINE 140-System (1	00)					Profile: Standard	F
								▲
🚍 (0) UR								
1	PS 307 5A	-	T				E	
2	CPU 315F-2 PN/DP		1				E Gateway	
X1	MPI/DP						⊞ (d.c.n.c,	
	PN-10							
X2 P1	Port 1						🗄 💼 PS-300	
3								
4	DI16/D016x24V/0.5A						🚊 🚊 SM-300	
5	AI4/AU2x8/8Bit						🗎 📋 Al-300	
	D18/D08xDC24V70,5A							
			-1				SM 334 AI4/A02	
			<u> </u>				SM 334 AI4/AU2x12Bit	
							5M 334 AI4/AU2x8/8Bit	
							SM 334 AI4/AU2X0/00IL	
						-	1 SM 335 AI4/A04x14/12Bit	
							- DI-300	
							🔚 SM 323 DI16/D016x24V/0.5A	
Slot Module I	Order number	Firmware	MPI address	I address	Q addre	C	SM 323 DI8/D08x24V/0.5A	
1 PS 307 5A 6	ES7 307-1EA00-0AA0						SM 323 DI8/D08xDC24V/0,5A	
2 CPU 315F-2 PN/DP 6	ES7 315-2FH13-0AB0	¥2.5	2				SM 323 DI8/D08xDC24V/0,5A	
X1 MFV/DF			2	2047*			SM 323 DI8/D08xDC24V/0.5A	
X2 FN-10				2046*			□ SM 327 DI8/DX8	
X21 Rot 1				2045*				
3								1
4 DI16/D016x24V/0.5A 6	ES7 323-1BL00-0AA0			01	01	_ ∟	SIMATIC S7-300, M7-300 and C7 modules (central	٠<
5 AI4/AU2x8/8Bit 6	ES7 334-UCEUT-UAAU			272279	2/22/5	<u>+</u>	lack)	
	ES7 323-18H01-0AA0	1		8	8	_		
Press F1 to get Help								

Zie het één en ander in figuur 4-14.

Figuur 4-14

De configuratie is nu klaar en kan in de PLC worden geladen. Kies nu via het menu Station \rightarrow Save and Compile.

Eerst wordt nog gevraagd hoe de CPU-berichten moeten worden opgesteld. Daar wordt geen gebruik van gemaakt. Kies de eerste mogelijkheid en klik dan op **OK**. Dit dialoog verschijnt éénmalig. Zie figuur 4-15.

tutorial - Message Number Assignment Selection	_ 🗆 🗵
Settings for the current project / library If you create a new project or library, you have to select one of the following options:	
 Assign CPU-oriented unique message numbers (as of WinCC V6, ProTool V6 or STEP 7 V5.2 It is no longer possible to convert to project-oriented or STEP 7 V5.1) Assign project-oriented unique message numbers (Previous method) 	
	Options >>
	Help

Figuur 4-15

Er volgt een venster waarin de voortgang wordt weergegeven. Zie figuur 4-16.

Compile	×
Station:	
SIMATIC 300(1)	
Module:	
[0/2/0] CPU 315F-2 PN/DP	
	Cancel

Figuur 4-16

4.4 Configuratie downloaden

Zet de PLC nu eerst in STOP door de operating mode schakelaar van RUN naar STOP te zetten. Nu kan de configuratie naar de PLC worden gestuurd via het menu **PLC**→**Download** of via het pictogram in figuur 4-17.



Figuur 4-17

Selecteer in het volgende venster welk *target module* moet worden geladen. Er is er maar één, dus dat is eenvoudig. Klik op **OK**. Zie figuur 4-18.

Daarna volgt een dialoog waarin het MPI-adres van de target module kan worden geselecteerd. Adres 2 is al geselecteed. Klik op **OK** (figuur 4-19).

Figuur 4-20 laat zien wat er gebeurt als de CPU-module die geconfigureerd is, anders is dan die geprogrammeerd is. De *online* en *offline*-versie verschillen. In deze figuur zijn onder andere de station- en modulenaam anders. Zorg er in ieder geval voor dat het ordernummer en

versienummer van de CPU correct is. Anders kan de CPU niet functioneren. Deze figuur hoeft niet te verschijnen.

Select Target Module	×
Module CPU 315F-2 PN/DP	Racks Slot 0 2
<u>S</u> elect All	
OK	Cancel Help



Select Node Ad	ldress				×
Over which stati PN/DP?	on address is the progr	amming dev	vice connect	ed to the module C	PU 315F-2
<u>R</u> ack:	0 -				
<u>S</u> lot:	2 -				
Target Station:		red by mear	ns of gateway		
Enter connecti	on to target station:				
MPI address	Module type Station CPU 315F READ	TE CF	odule name PU315 R	Plant designation Brouw Fab	
Accessible Node	88				
	I	⊻iew	v		
ОК				Cancel	Help



Name	Utfline	Unline	Status
Order number	6ES7 315-2FH13-0AB0	6ES7 315-2FH13-0AB0	~
irmware version	V2.5	V2.5.1	✓
Station name	SIMATIC 300(1)	READ_TEMP	Δ
dodule name	CPU 315F-2 PN/DP	CPU315_READ_TEMP	$\overline{\mathbf{A}}$
Plant designation		Brouw Fab	$\overline{\mathbf{A}}$
ocation ID		BrouwNet Control Room	Ā

Figuur 4-20

Het laden wordt nu gestart. Dit duurt enkele seconden. Zie figuur 4-21. Als het laden niet lukt, kan het zijn dat de *PG/PC Interface* verkeerd is ingesteld. Raadpleeg dan de docent.

Download	×
Station: SIMATIC 300(1) Module: [0/2/0] CPU 315F-2 PN/DP	
	Cancel

Figuur 4-21

Sluit de hardwareconfiguratie af via menu **Station**→**Exit**. Er wordt weer terug gekeerd naar het netwerkprogramma Netpro. Hierin is duidelijk te zien dat de PLC verbonden is aan het MPI- en Ethernet-netwerk. Zie figuur 4-22.



Figuur 4-22

De hardwareconfiguratie is nu klaar. Netpro kan worden afgesloten via het menu **Network**→**Exit**.

4.5 Symbolen definiëren

Het onthouden van de adressen als ingangen, uitgangen en timers, is een lastige en vermoeiende bezigheid. Gelukkig kan een symbolenlijst worden opgesteld. Er kan dan gerefereerd worden met de symbolische naam in plaats van het eigenlijke adres. Dubbelklik in de projectscherm op Symbols (figuur 4-23). De symbolen-editor wordt dan gestart. Voer de symbolen in zoals is weergegeven in figuur 4-24 (het veld Data Type wordt door de editor zelf ingevuld). Sluit de editor af, de definities worden automatisch opgeslagen. Soms verschijnt een dialoog zoals in figuur 4-25. Kies dan **Yes**.

🔤 tutorial D:\PROJECTS\STE	P7\tutorial		<u> </u>
E- tutorial SIMATIC 300(1) E- I CPU 315F-2 PN/DP I CPU 315F-2 PN/DP II - I S7 Program(1)	Sources	Blocks Blocks	

Figuur 4-23

📥 S7	🖥 S7 Program(1) (Symbols) tutorial/SIMATIC 300(1)/CPU 315F-2 PN/DP						
	Status	Symbol	Add	dress 🛆	Data type	Comment	
1		drukknop1	1	8.0	BOOL	Drukknop 1	
2		drukknop2	1	8.1	BOOL	Drukknop 2	
3		reset	1	8.7	BOOL	Reset systeem	
4		voet_gezien	М	8.0	BOOL	Voetganger gezien	
5		timer_geel_klaar	М	8.1	BOOL	Timer voor geel klaar met tijdmeten	
6		timer_groen_klaar	М	8.2	BOOL	Timer voor groen klaar met tijdmeten	
7		voet_gezien_flank	М	8.3	BOOL	Flank detector voet_gezien	
8		voet_lampje	Q	8.0	BOOL	Voetganger gezien feedback lamp	
9		auto_rood	Q	8.2	BOOL	Verkeerslicht auto rood	
10		auto_geel	Q	8.3	BOOL	Verkeerslicht auto geel	
11		auto_groen	Q	8.4	BOOL	Verkeerslicht auto groen	
12		voet_rood	Q	8.6	BOOL	Verkeerslicht voetganger rood	
13		voet_groen	Q	8.7	BOOL	Verkeerslicht voetganger groen	
14		timer_geel	T	0	TIMER	Timer voor auto geel	
15		timer_groen	T	1	TIMER	Timer voor auto groen	
16							

Figuur 4-24



Figuur 4-25

4.6 Programma invoeren

Zoals bekend moet een PLC-programma altijd Organization Block 1 (OB1) bevatten, ook al roept deze niets anders aan dan Functions (FC) en Function Blocks (FB).

Eerst wordt FC1 geprogrammeerd met het eigenlijke verkeerslichtprogramma, vervolgens wordt OB1 geprogrammeerd met daarin een aanroep van FC1. Deze volgorde is van belang.

Vanuit de Simatic Manager moet FC1 aangemaakt worden. Selecteer in het linkerschem Blocks, en klik in het rechterscherm (onder System Data en OB1) op de **rechter** muisknop. Er wordt een contextgevoelig menu geopend en selecteer daarmee **Insert New Object**→**Function**. Zie figuur 4-26.

🔄 tutorial D:\PROJECTS\STEF	7\tutorial	
tutorial D:\PROJECTS\STEF tutorial		
	Check Block Consistency Print Object Properties Alt+Ret Special Object Properties	

Figuur 4-26

Nu kunnen allerlei gegevens (eigenschappen) worden ingevoerd. Vul dit in zoals is weergegeven in figuur 4-27. Let erop dat als programmeertaal LAD wordt geselecteerd. Klik op **OK**.

In de projectscherm staat nu FC1. Dubbelklik hier op en de LAD-editor wordt geopend. Zie figuur 4-28.

Properties - Function		X
General - Part 1 General	- Part 2 Calls Attributes	
<u>N</u> ame:	FC1	
<u>S</u> ymbolic Name:	Verkeerslicht	
Symbol <u>C</u> omment:		
Created in Language:	LAD	
Project path:		
Storage location of project:	D:\PROJECTS\STEP7\tutorial	
	Code	Interface
Date created:	12/27/2007 10:34:55 AM	
Last modified:	12/27/2007 10:34:55 AM	12/27/2007 10:34:55 AM
C <u>o</u> mment:		×
ОК		Cancel Help











Een ladderdiagram bestaat uit één of meerdere netwerken. Per netwerk kan één logische schakeling (*rung*) geprogrammeerd worden met één of meerder uitgangen. Een netwerk kan worden toegevoegd door middel van het eerste, meest linkse pictogram uit figuur 4-29. Als ingangen kunnen PLC-ingangen en merkers dienen. Dit wordt aangegeven met kontakten. De tweede en derde pictogram zijn resp. *normally open* en *normally closed contacts*. Als uitgangen (*coils*) kunnen PLC-uitgangen of merkers gebruikt worden⁶. Deze worden toegevoegd met het vierde pictogram. Een logische OR wordt gemaakt met behulp van de laatste twee pictogrammen.

Het programmeren vergt enige routine. Om ingangen in te voegen moet b.v. eerst een rung van een netwerk worden geselecteerd worden zodat deze groen wordt. Dan worden de pictogrammen uit figuur 4-29 blauw en kunnen ze worden ingevoegd. Niet alle mogelijkheden kunnen via de pictogrammen worden ingevoerd. Aan de linkerkant staan alle mogelijkheden, zoals SR-elementen, timers, counters en rekenkundige operaties.

Tijdens het invoeren van adressen zoals M0.0 verschijnt de symbolenlijst. Dan kan eenvoudig het juiste symbool worden gekozen. Zie figuur 4-30.





Indien iets niet ingevoerd kan worden, zoals een BYTE-adres bij een contact, wordt dit rood afgebeeld en kan de laddereditor niet worden afgesloten. Ook als adressen niet zijn ingevuld, zoals in figuur 4-30 zichtbaar, kan de editor niet worden afgesloten.

Als voorbeeld is het eerste netwerk in figuur 4-31 afgebeeld. Het complete programma is opgenomen in bijlage 1. Voor correcte werking moet het hele programma ingevoerd worden.

Sluit de editor af via **File**→**Exit** als het hele ladderdiagram is ingevoerd.

⁶ Ook Q, DB en L kunnen als ingangen en uitgangen dienen.



Figuur 4-31

Nu FC1 ingevoerd is, moet OB1 nog worden geprogrammeerd. OB1 moet alleen FC1 aanroepen. Dit wordt gedaan in STL. OB1 is al aangemaakt; dat gebeurt bij het aanmaken van een nieuw project. Als OB1 niet bestaat, voeg het dan toe via **Insert New Object**→**Organization Block**. OB1 verschijnt nu in het overzicht in projectscherm. Zie figuur 4-32.

🔄 tutorial D:\PROJECTS\STEP7\tu	torial			
tutorial D:\PROJECTS\STEP7\tu	torial stem data Cut Copy Paste Delete Insert New Object PLC Rewire Compare Blocks Reference Data Check Block Consistency.	Ctil+X Ctil+C Ctil+V Del	Organization Block Function Block Function Data Block Data Type Variable Table	
	Check Block Consistency. Print Object Properties Special Object Properties	 Alt+Return		

Figuur 4-32

Als OB1 al bestaat, wijzig dan de eigenschappen door eerst OB1 te selecteren. Klik nu op de rechtermuisknop en selecteer **Object Properties**. Vul de eigenschappen in voor OB1 zoals is aangegeven in figuur 4-33 en klik op OK. Dubbelklik op OB1 om de STL-editor te openen⁷. Vul nu als enige regel de aanroep naar FC1 in. Zie figuur 4-34. Sla het ingevoerde programma op via **File** \rightarrow **Save** en sluit de editor af via het menu **File** \rightarrow **Exit**.

⁷ De oplettende lezer ziet dat de STL-editor en de ladder-editor dezelfde zijn. Dit is correct. Via het menu View kan worden geschakeld tussen STL, LAD en FBD.

Properties - Organization	Block	×
General - Part 1 General	- Part 2 Calls Attributes	
<u>N</u> ame:	OB1	
<u>S</u> ymbolic Name:	Main Cycle	
Symbol <u>C</u> omment:		
Created in Language:	STL 💌	
Project path:		
Storage location of project:	D:\PROJECTS\STEP7\tutorial	
	Code	Interface
Date created:	12/27/2007 11:30:23 AM	
Last modified:	12/27/2007 11:30:23 AM	12/27/2007 11:30:23 AM
C <u>o</u> mment:		×
ОК		Cancel Help

Figuur 4-33



Figuur 4-34

4.7 Laden PLC programma

Nu moeten de diverse onderdelen in de PLC geladen worden. Zet de PLC eerst in STOP. Klik met de rechtermuisknop op Blocks in de Simatic Manager en selecteer **PLC**-download (figuur 4-35). Het laden wordt nu begonnen. Als blokken, die geladen moeten worden, al in de PLC aanwezig zijn, zal een bevestiging worden gevraagd (figuur 4-36). Klik op de knop **All**, zodat niet voor elk blok een aparte vraag wordt gesteld.

🔄 tutorial D:\PROJECTS	\STEP7\tutorial				
tutorial SIMATIC 300(1) G SIMATIC 300(1) G	//DP (1) s	: OB1	⊕ FC1		
	Cut	Ctrl+X			
	Сору	Ctrl+C			
	Paste	Ctrl+V			
	Delete	Del			
	Insert New Object		•		
	PLC		Download	Ctrl+L	
	Rewire		Copy RAM to ROM		
	Compare Blocks		CPU Messages		
	Reference Data		 Display Force Values 		
	Check Block Consistency		Monitor/Modify Variables		
	Print		Hardware Diagnostics		
	Rename	F2	Module Information	Ctrl+D Ctrl+l	
	Object Properties	Alt+Return	Clear/Reset	Cuiti	
	Special Object Properties		Set Time of Day		

Figuur 4-35

Insert Organizatio	n Block									
The ot	iject 'OB1	'already e	xists. Do	you war	nt to over	write it?				
	1									
<u>n</u> ename										
Yes	All		No	1	None	1	Cancel	1	Help	

Figuur 4-36

Omdat alle blokken geselecteerd zijn zal ook de System Data worden geladen. Dit mag overgeslagen worden door op **No** te klikken. Als op **Yes** wordt geklikt, volgt er nog een aantal dialogen. Zie de figuren 4-37 tot en met 4-39. De figuren 4-40 en 4-41 verschijnen alleen als de PLC in RUN staat.

Download	(294:36)
1	Do you want to load the system data?
(<u>Y</u> es	No

Figuur 4-37



Figuur 4-38

Download	×
Station: SIMATIC 300(1) Module: [0/2/0] CPU 315F-2 PN/DP	
	Cancel

Figuur 4-39



Figuur 4-40



Figuur 4-41

4.8 Monitoren van het programma

Het programma is nu geladen; de werking kan worden getest. Start de PLC door operating mode schakelaar van STOP naar RUN te schakelen. De PLC gaat nu starten. Daarna kunnen de schakelaars op de simulatiemodule gebruikt worden om het programma te testen. Raadpleeg de docent.

Het programma kan ook gevolgd (*monitor*) worden via de ladder- of STL-editor. Dubbelklik hiervoor op FC1 om deze bouwsteen te openen in de editor.

Start het monitoren via het menu **Debug**→**Monitor** of klik op het brilletje zoals afgebeeld in figuur 4-42.



Figuur 4-42

De kleur van de titelbalk van de editor wordt nu lichtblauw ten teken dat het hier om een online verbinding gaat. Onderaan de editor is een groene, lopende balk zichtbaar. Zie figuur 4-43.

EAD/STL/FBD - [@FC1 tutorial	SIMATIC 300(1)\CPU 315F-2 PN/DP	ONLINE]
🗗 File Edit Insert PLC Debug View	Options Window Help	
Press F1 to get Help.		

Figuur 4-43

Hieronder zijn twee afbeeldingen zichtbaar van netwerk 1 en netwerk 2 tijdens het draaien van het programma (figuren 4-44 en 4-45).







Figuur 4-45

De tutorial is hiermee ten einde. De cursist wordt uitgedaagd het eenvoudige verkeerslichtsysteem uit te breiden met b.v. een mogelijkheid tot het laten knipperen van het gele licht indien het systeem niet operationeel is. Hiervoor is een extra schakelaar nodig.

5. Tutorial S7-Graph

Veel productieprocessen kunnen worden opgelost met een afloop. Een (deel-)product volgt een vast aantal stappen waarin iets gebeurd. Deze stappen zijn goed van elkaar te onderscheiden. Zo'n programma kan goed worden geprogrammeerd als een toestandsmachine. S7-Graph is een taal waarin zo'n toestandmachine kan worden geprogrammeerd.

In deze tutorial wordt uitgelegd hoe een S7-Graph programma ingevoerd moet worden. Eerst wordt de configuratie van de PLC ingevoerd, vervolgens een klein ladder-programma voor conditionering van de ingangen, dan het S7-Graph programma. Als laatste wordt dit programma en enkele variabelen gevolgd met de monitor.

5.1 Project aanmaken

Begin een nieuw project zoals is aangegeven in het vorige hoofdstuk en noem het project tut_graph. Zie figuur 5-1.

New Project			×
User projects	Librari	es Multiprojects	
Name		Storage path	
🞒 sheets		D:\PROJECTS\STEP7\st	neets
🞒 traf_graph		D:\PROJECTS\STEP7\tr	af_gra
🔄 🎒 traf_graph_	simul	D:\PROJECTS\STEP7\tr	af_gr1
🞒 traf_lad		D:\PROJECTS\STEP7\tr	af_lad
🞒 tutorial		D:\PROJECTS\STEP7\tu	itorial
🛃 verkeerslich	ht	D:\PROJECTS\STEP7\v	erkeers
🔲 Add to curren	ıt multi	project	
Na <u>m</u> e:			<u>T</u> ype:
tut_graph			Project 💌
			E Libraru
Storage location			L L HENRY
D:\PROJECTS\	STEP	7	Browse
OK		Ca	ancel Help
<u></u>			

Figuur 5-1

Vul de bekende gegevens in en klik op OK. Het nieuwe project wordt aangemaakt.

Eerst moet de MPI-bus MPI(1) verwijderd worden, deze wordt straks opgehaald uit de configuratie. Indien deze stap wordt overgeslagen, zijn straks twee MPI-bussen zichtbaar.

5.2 Configuratie en Blocks uit PLC halen

In het vorige hoofdstuk is een nieuwe configuratie van de PLC opgezet en geladen in de PLC. Deze configuratie kan teruggehaald worden uit de PLC en zo gebruikt worden in het nieuwe project⁸. Hiervoor moeten de configuratiegegevens ge-*upload* worden vanuit de PLC naar het project. Hierbij worden ook alle programma's mee gekopieerd (OB, FC etc). De symbolenlijst is niet in de PLC opgeslagen, die moet opnieuw worden toegevoegd.

Doe een upload via het menu PLC-Upload Station to PG. Zie figuur 5-2.



Figuur 5-2

Er verschijnt nu een scherm met daarin alle aangesloten PLC's. Midden onderaan het scherm verschijnt een knop **View**. Als er geen PLC's zichtbaar zijn, klik hier dan op. Na een tijdje verschijnt onder Accessible Nodes de PLC die op het prakticum is aangesloten. Selecteer de gegeven PLC en klik op **OK**. Zie figuur 5-3.

Select Node A	ddress				×
Which module o	lo you want to i	reach?			
<u>R</u> ack:	0 🚔				
<u>S</u> lot:	0 🗄				
Target Station:	⊙ <u>L</u> oca	I			
	🔿 Can t	pe reached by n	neans of gatewa	у	
Enter connect	ion to target sta	ation:			
MPI address	Module type	Station name	Module name	Plant designation	
2	CPU 315F	SIMATIC 3	CPU 315F		
Accessible Node	BS				
2	CPU 315F	SIMATIC 3	CPU 315F		
			l atebo		
ОК				Cancel	Help

Figuur 5-3

⁸ Natuurlijk wordt de laatst geladen configuratie opgehaald.

De configuratie wordt nu uit de PLC geladen en in het project opgenomen. Zie figuur 5-4.



Figuur 5-4

In het projectscherm is nu de PLC, de MPI-bus én de Ethernetbus zichtbaar. Als er twee MPIbussen zichtbaar zijn, moet er één verwijderd worden. Dit kan worden gedaan via het netwerkconfiguratie-programma Netpro. De programma- en datablokken zijn ook uit de PLC gehaald. Deze moeten verwijderd worden. Zie figuur 5-5 als voorbeeld.

🔤 tut_graph D:\PROJECTS\S	TEP7\tut_grap			
tut_graph D:\PROJECTS\S tut_graph SIMATIC 300(1) SIMATIC 300	TEP7\tut_grap ∰ System data	OB1 Open Object Cut Copy Paste Delete Insert New Object PLC	Ctrl+Alt+D Ctrl+Alt+D Ctrl+X Ctrl+C Ctrl+V Del	
		Rewire Compare Blocks Reference Data	F	
		Print	+	
		Special Object Propertie	s 🕨	

Figuur 5-5

Nu wordt gevraagd of de blokken verwijderd moeten worden. Klik op Yes. Zie figuur 5-6.

Delete (32	80:794)	×
	This action cannot be undone! The configured diagnostic data will also be deleted.	
	Delete symbolic block names too (Symbolic name of variable tables are always deleted too. Non-unique block names will not be deleted.)	
	Do you really want to delete the selected objects?	
Yes	No	

Figuur 5-6
De configuratie is nu afgerond. Via de programma's Netpro en HWConfig kunnen eventuele wijzigingen aangebracht worden.

5.3 Symbolenlijst opstellen

Natuurlijk moet ook voor deze tutorial een symbolenlijst opgesteld worden. In figuur 5-7 zijn de symbolen vermeld. Let vooral op aantal_gedrukt, dit is een merkerwoord van het type INT.

	rogram	(1) (Symbols) tut_gr	aph\SIMATI	C 300(1)\CP	2U 315F-2 PN/DP
	Status	Symbol	Address 🛆	Data type	Comment
1		drukknop1	I 8.0	BOOL	Drukknop 1
2		drukknop2	I 8.1	BOOL	Drukknop 2
3		knipknop	1 8.2	BOOL	Drukknop voor knipperen geel licht
4		reset	I 8.7	BOOL	Reset systeem
5		voet_gezien	M 8.0	BOOL	Voetganger gezien
6		knip_gezien	M 8.1	BOOL	Knipperschakelaar gezien
7		voet_gezien_flank	M 8.2	BOOL	Flankdetector voet_gezien
8		knip_gezien_flank	M 8.3	BOOL	Flankdetector knip_gezien
9		aantal_gedrukt	MV 20	INT	Aantal keer dat er op een drukknop is gedrukt
10		voet_lampje	Q 8.0	BOOL	Voetganger gezien feedback lamp
11		auto_rood	Q 8.2	BOOL	Verkeerslicht auto rood
12		auto_geel	Q 8.3	BOOL	Verkeerslicht auto geel
13		auto_groen	Q 8.4	BOOL	Verkeerslicht auto groen
14		voet_rood	Q 8.6	BOOL	Verkeerslicht voetganger rood
15		voet_groen	Q 8.7	BOOL	Verkeerslicht voetganger groen
16					

Figuur 5-7

5.4 Een blok vooraf

Eerst wordt een ladderprogramma ingevoerd wat samenwerkt met het S7-Graph programma. Het conditioneert de ingangen, zodat de invoer via de drukknoppen makkelijk kunnen worden verwerkt. Maak hiervoor FC2 aan, vul de gegevens in en voer het ladderprogramma in en klik daarna op **OK**. Zie de figuren 5-8 en 5-9.

Properties - Function			×
General - Part 1 General	- Part 2 Calls Attributes		
<u>N</u> ame:	FC2		
Symbolic Name:	DrukknopConditioner		
Symbol <u>C</u> omment:			
Created in Language:	LAD		
Project path:			
Storage location of project:	D:\PROJECTS\STEP7\tut_grap)	
	Code	Interface	
Date created: Last modified:	12/30/2007 09:06:35 AM	12/30/2007 09:06:35 AM	
Commont:	12/30/2001 03:00:33 AM	12/30/2007 03:00:33 AM	- I
C <u>o</u> nment.			1
			a
		<u> </u>	
ОК		Cancel Help	

Figuur 5-8

FC2 : Drukknopconditioner

Deze FC bestaat uit twee rungs, een voor de voetgangersdrukknoppen en een voor de knipperdrukknop Zie aldaar.

Network 1: Voetganger gezien

Dit netwerk zorgt voor conditionering van het gebruik van de drukknoppen door voetgangers. De flank is nodig omdat als een drukknop continue wordt ingedrukt, de verkeerslichtsysteem "rondjes" blijft draaien. De merker "voet_gezien" wordt gereset in het verkeerslichtsysteem-programma.



Network 2: Knipperschakelaar gezien





Figuur 5-9

5.5 Blokken, stappen en overgangen

Naar de gebruiker toe laat de CPU en bijbehorend geheugen zich afbeelden als programmeerbare blokken. Er zijn verschillende typen, zie hiervoor bijlage 3. Voorlopig is genoeg te weten dat elk gebruikersprogramma moet starten in Organization Block 1 (OB1) en dat S7-Graph programma's worden opgeslagen in een Function Block (FB) met bijbehorend Data Block (DB).

Een S7-Graph programma bestaat uit een aantal stappen (*steps*) en overgangen (*transitions*) tussen de stappen. Dit geheel wordt in S7-Graph een sequencer genoemd. In een stap kunnen diverse operaties worden uitgevoerd, zoals het aanzetten van een uitgang of het starten van een teller. Aan de overgangen kunnen voorwaarden worden verbonden zoals "ga verder als ingang 1 actief is".

De oplettende lezer ziet in dit alles een overeenkomst met toestandmachines uit de digitale techniek.

Eigenlijk is S7-Graph niets anders dan een frontend die de ingevoerde sequencer omzet naar STL. Het is dus eigenlijk een compiler.

5.6 Aanmaken Function Block

Een S7-Graph programma kan alleen in een FB (Function Block) ingevoerd worden. Aan een Function Block wordt een Data Block gekoppeld.

Eerst moet een FB worden aangemaakt waarin het S7-Graph programma kan "leven". Klik in de Simatic Manager links Blocks open. Klik nu met de rechtermuisknop om een **Function Block** in te voegen. Zie figuur 5-10. Er wordt een dialoogvenster geopend waarin de eigenschappen van de FB kunnen worden vastgelegd (figuur 5-11).

Vul bij Name FB2 in. In Symbolic Name kan een naam worden ingevoerd. Deze naam komt dan terug in de symbolentabel. Commentaar gaat in Symbol Comment. Kies verder als taal Graph (Created in Language: GRAPH). Klik op **OK** om de FB aan te maken.

🔄 tut_graph D:\PROJECTS\S	TEP7\tut_grap					-OX
	🚵 System data	€ <mark>FC2</mark>				
Blocks		Cut Copy Paste	Ctrl+X Ctrl+C Ctrl+V			
		Delete	Del		Orașuliaștian Disale	
		PLC			Function Block	
		Rewire Compare Blocks Reference Data Check Block Consistency		, 	Function Data Block Data Type Variable Table	
		Print		•		
		Object Properties Special Object Properties	Alt+Return	•		

Figuur 5-10

Properties - Function Bl	ock	×
General - Part 1 Genera	I - Part 2 Calls Attributes	
<u>N</u> ame:	FB2	🗖 Mul. Inst. Cap.
<u>S</u> ymbolic Name:	Verkeerslichtsysteem	
Symbol <u>C</u> omment:	Dit is het eigenlijke verkeerslic	htsysteem
Created in Language:	GRAPH 💌	
Project path:		
Storage location of project:	D:\PROJECTS\STEP7\tut_g	rap
	Code	Interface
Date created: Last modified:	12/30/2007 09:34:33 AM 12/30/2007 09:34:33 AM	12/30/2007 09:34:33 AM
Comment:		
- <u>-</u>		-
		*
	·	
OK		Cancel Help

Figuur 5-11

5.7 Starten S7-Graph editor

Nu is in de Simatic Manager FB2 te vinden. Dubbelklik hierop om de S7-Graph editor met FB2 te openen. Het openingsscherm is afgebeeld in figuur 5-12.

Voor meer informatie over het editen, zie het document "S7-Graph - Programming Sequential Control Systems", hoofdstuk 6.

🔣 S7 Graph - [FB2,DB2 (Sequencer 1) tut_graph\SIMATIC 300(1)\CPU 315F-2 PN/DP\]	
Eile Edit Insert PLC Debug View Options Window Help	_ @ ×
🗋 😂 🖬 🎒 🗠 • ལ - 🐰 🖺 💼 🗰 🕅 🏙 🚳 💷 🕺 🕺 70% 🔽 💈 🔤	
Hermanent instructions before sequencer	
Block comment	
to. Permanent instructions after sequencer	
t	
TT SI SI	.ep1
Step1	<u> </u>
Transl	
물?	
1	
Compile: tut_graph\SIMATIC 300(1)\CPU 315F-2 PN/DP\\FB2, DB2 - <offlir< td=""><td>ne> *</td></offlir<>	ne> *
Errors occurred.	
Error >>> (T1) without subsequent element.	
Warning >>> (T1) without contents.	
Warning >>> (T1, Condition) Condition is always true.	
1 error(s) found; 3 warning(s) found	
Compile / Decompile Messages / Variables > Addresses > Forward Cross References > Backward Cross Re	lerences /
Press F1 for help.	ine Abs Ins Chg. Interface //

Figuur 5-12

Het scherm is in drie panelen verdeeld. Linksboven is een overzicht van de sequencer inclusief de permanente instrucies te zien (tabblad Graphic). Er kan echter ook voor het tabblad Variables gekozen worden (figuur 5-13). Rechtsboven is het edit-paneel. Hierin worden de aanpassingen verricht. Onder is console-paneel. Hier zijn onder andere systeemberichten te vinden zoals compiler-uitvoer, gegevens van variabelen en gebruikte adressen in een stap.

In het edit-paneel wordt gelijk de eerste stap geplaatst. De dubbele ring geeft aan dat het hier om een beginstap (*initial step*) gaat. Als de sequencer wordt herstart (zie paragraaf 5-13), is dit de eerste stap.

Let op: er moet precies één initial step in het programma aanwezig zijn. Een veel voorkomend probleem is dat tijdens het programmeren de initial step wordt verwijderd, en er geen nieuwe wordt aangemerkt. De sequencer start dan niet als de PLC het programma begint te verwerken.



Figuur 5-13

5.8 Invoeren eerste stap

Het gebruik van de editor vergt, net als bij de laddereditor, enige routine. Als vuistregel geldt: als iets veranderd, ingevoegd of verwijderd moet worden, selecteer dan eerst het onderdeel en voer dan de operatie uit. Namen van variabelen, zoals auto_rood, kunnen zonder aanhalingstekens worden ingevoerd, de editor plaatst ze zelf. Hiervoor moet wel de optie **View**→**Display With**→**Symbol List** aanstaan. In figuur 5-14 is te zien hoe een actie wordt ingevoerd. Selecteer de actiekop, klik op de rechter muisknop en selecteer **Insert New Element**→ **Action**.



Figuur 5-14

In figuur 5-15 is te zien hoe een overgangsconditie, in dit geval een vergelijkfunctie, wordt ingevoerd. Ook dit vergt enige routine.



Figuur 5-15

In figuur 5-16 is de complete stap met acties en overgangsconditie te zien. Programmeer de eerste stap precies als in deze figuur.



Figuur 5-16

5.9 Invoeren volgende stappen

Aan de eerste stap moet nu de vervolgstap gekoppeld worden. Selecteer hiervoor de overgang (niet de conditie), klik op de rechter muisknop en selecteer **Insert New Element**→**Step + Transition**. Een nieuwe stap wordt nu toegevoegd. Zie figuur 5-17.





In figuur 5-18 zijn alle stappen, acties en overgangscondities afgebeeld. Voer deze zo in. Let op de laatste stap (stap 6). Daar wordt een *jump* gemaakt naar stap 2. Boven stap 2 is dan een horizontale pijl te zien. Deze is pas te zien na de plaatsing van de jump.





Als alles ingevoerd is, moet programma nog opgeslagen en gecompileerd worden. Start deze actie via het menu **File** \rightarrow **Save**. Hierna kan de editor worden afgesloten.

In het projectscherm zijn nu FB2 en andere blokken te zien (figuur 5-19). DB2 is de Data Block dat bij FB2 hoort. Dat wordt door de S7-Graph-compiler geregeld. De code in FB2 gebruikt dus DB2 als geheugen. FC72 (Function) is een blok waarin generieke (voor ieder hetzelfde) code wordt geplaatst voor S7-Graph ingevoerde blokken. SFC64 (System Function) wordt door FB2 gebruikt om de verlopen tijd bij te houden.

🔄 tut_graph D:\PROJECTS\S	TEP7\tut_grap			_ 🗆 🗵
tut_graph SIMATIC 300(1) Graph Surces Sources Blocks	System data DB2	FB2	₽ FC72	

Figuur 5-19

5.10 Invoeren OB1

Nu moet nog OB1 ingevoerd worden, die FC2 en FB2 moet aanroepen. Zonder OB1 kan de PLC geen programma draaien. OB1 wordt geprogrammeerd in STL. Zie figuur 5-20.

Properties - Organization	1 Block	X
General - Part 1 General	- Part 2 Calls Attributes	
<u>N</u> ame:	OB1	
<u>S</u> ymbolic Name:	Main Cycle	
Symbol <u>C</u> omment:		
Created in Language:	STL	
Project path:		
Storage location of project:	D:\PROJECTS\STEP7\tut_grap	
	Code	Interface
Date created:	12/30/2007 11:13:14 AM	
Last modified:	12/30/2007 11:13:14 AM	12/30/2007 11:13:14 AM
C <u>o</u> mment:		<u>×</u>
		V
ОК		Cancel Help

Figuur 5-20

Het programma van OB1 bevat twee aanroepen (*call*), één naar FC2 en één naar FB2, die DB2 gebruikt. Let heel goed op de aanroep van FB2. Zodra dit wordt ingevoerd, zal de STL-editor gegevens toevoegen. Dit zijn de formele parameters van FB2. Alleen INIT_SQ moet gekoppeld worden aan symbool reset, de rest wordt niet gebruikt. De reset kan dan gebruikt worden om de sequencer te herstarten. Zie figuur 5-21 voor de inhoud van OB1. Sla de gegevens op en sluit de editor.



Network 2: Title:

Comment:

CALL FB	2 , DB2
OFF SQ	:=
INIT SQ	:="reset"
ACK EF	:=
S PREV	:=
S NEXT	:=
SW AUTO	:=
SW_TAP	:=
SW_MAN	:=
S_SEL	:=
s_on	:=
S_OFF	:=
T_PUSH	:=
s_NO	:=
S_MORE	:=
S_ACTIVE	:=
ERR_FLT	:=
auto_on	:=
TAP_ON	:=
MAN ON	:=

Figuur 5-21

5.11 Laden PLC met programma

Het laden van het programma gaat zoals is beschreven in hoofdstuk 4. Zorg ervoor dat de PLC in STOP staat. Selecteer de blokken OB1, FC2, FB2, DB2 en FC72 en laadt ze in de PLC. Blok SFC64 is reeds in de PLC aanwezig.

5.12 Variabelentabel

Met een variabelentabel is het makkelijk om variabelen (addressen) in de PLC te monitoren. Ze kunnen dan ook gewijzigd worden, zelfs ingangen (denk hierbij aan de *Process Image Table*). Maak in het projectscherm een tabel aan met **Insert New Object**→**Variable Table**. Zie figuur 5-22.

Noot: de variabelentabel kan niet in de PLC geladen worden.

🔄 tut_graph D:\PROJECTS\S	TEP7\tut_grap				
	System data FC72	 OB1 DB2 	GFB2 GFC64	G FC2	
E 🔁 Blocks	Cop Pas Dele Inse PLC	y te tt New Object	Ctrl+C Ctrl+V Del	Organization Block Function Block	
	Rev Com Refu Che	vire Ipare Blocks erence Data ck Block Consistency	••••••••••••••••••••••••••••••••••••••	Function Data Block Data Type Variable Table	
	Obje Spe	set Properties cial Object Properties	Alt+Return		

Figuur 5-22

In figuur 5-23 zijn de eigenschappen weergegeven. Vul in naar eigen inzicht en klik op OK.

Properties - Variable Ta	ble		×
General - Part 1 General	- Part 2 Attributes		_
<u>N</u> ame:	VAT1		
<u>S</u> ymbolic Name:	VAT_1		
Symbol <u>C</u> omment:			
Project path:			
Storage location of project:	D:\PROJECTS\STEP7\tut_grap		
	Code	Interface	
Date created:	12/30/2007 11:24:57 AM	10/00/2007 11/24/57 444	
Last modified:	TZ/30/2007 TT:24:57 AM	12/30/2007 11:24:57 AM	
C <u>o</u> mment:		<u></u>	
		_	
	1	7	
OK		Cancel Help	

Figuur 5-23

Nadat de eigenschappen zijn ingevuld, moeten de variabelen worden ingevoerd. Dit kan door symbolen in te vullen, of direct adressen. Daarna kan voor variabele aantal_gedrukt het afbeeldformaat op DEC gezet worden. Zie figuur 5-24

Sla de tabel op via het menu Table→Exit. Er verschijnt een dialoog als in figuur 5-25.

K	VA	.T1 -	- tut	_graph\SIMAT	IC 300(1)\CPL	J 315F-2 PN	/DP\S7 Proj	gram(1)	
	1	Add	ress	Symbol	Display format	Status value	Modify value		
1		MVV	20	"aantal <u>g</u> edrukt	DEC				
2		Q	8.2	"auto_rood"	BOOL				
3		Q	8.3	"auto_geel"	BOOL				
4		Q	8.4	"auto_groen"	BOOL				
5		Q	8.6	"voet_rood"	BOOL				
6		Q	8.7	"voet_groen"	BOOL				
7		I	8.7	"reset"	BOOL		true		
8									
Г									

Figuur 5-24

Close var	Close variable table (16:5003)						
1	The current variable table '@tut_graph\SIMATIC 300(1)\CPU 315F-2 PN/DP\S7 Program(1)\\VAT_1 ' has not been saved.						
	Do you want to save the variable table?						
<u>Y</u> es	s <u>N</u> o Cancel Help						

Figuur 5-25

5.13 Monitoren van het programma en variabelen

Het S7-Graph-programma kan gemonitord worden, d.w.z. gevolgd worden. De S7-Graph-editor bevat hiervoor een speciale modus. Open de editor door op FB2 te dubbelklikken. Aan de bovenrand is een pictogram met een brilletje te vinden. Klik hier één keer op. De editor zal nu communiceren met de PLC en weergeven in welke stap de sequencer zich bevindt (groen gekleurde stap). Ook de conditie van de transitie wordt weergegeven. Zie figuur 5-26.

De variabelen kunnen op eenzelfde wijze gemonitord worden. Open de variabelentabel VAT1 en klik op het brilletje. De variabelen worden dan continue bijgewerkt met de meest actuele waarden.

Het is ook mogelijk om waarden *in* een variabele te plaatsen, bijvoorbeeld een merker of een uitgang. Ingangen kunnen ook worden aangepast, want de aanpassing wordt gedaan in de *process image table* en natuurlijk niet op de ingangen zelf. Zo kan bijvoorbeeld een reset van de sequencer worden uitgevoerd vanaf de PC als er geen fysieke schakelaar op de ingang is aangesloten. Vul in de kolom Modify Value het getal 1 of het woord true in en klik op de pictogram met de bliksemschicht (naast het brilletje).

Zie voor een overzicht figuur 5-27.

5.14 Extra opdracht

In bijlage 2 is het programma uitgebreid met een knippermogelijkheid. Voer dit gedeelte nog in.



Figuur 5-26

₩ <mark>(</mark> Va	ar - V	AT1											١×
<u>T</u> able	<u>E</u> dit	<u>I</u> n:	sert	PLC	V <u>a</u> riable ⊻iew	<u>Options</u> <u>W</u> ir	ido	w <u>H</u> elp					
-121		2		8	X 🖻 🖬	n 🛛 🗙	2	- 	?	💱 🔐 🛷	66 47	lla	
		_		_									
	_												
		V.	AT1	@	tut_graph\SIM	ATIC 300(1)\(CPI	U 315F-2	PN/DP\S7	Program(1))			
		1	Add	dress	Symbol	Display format	St	atus value	Modify value				
	1		MW	20	"aantal_gedrukt	DEC		6					
	2		Q	8.2	"auto_rood"	BOOL		true					
	3		Q	8.3	"auto_geel"	BOOL		false					
	4		Q	8.4	"auto_groen"	BOOL		false					
	5		Q	8.6	"voet_rood"	BOOL		true					
	6		Q	8.7	"voet_groen"	BOOL		false					
	7		I	8.7	"reset"	BOOL		false	true				
	8												
	9												
		-	-	_			-	_	_		_		
tut_gra	ph\SIN	IAT	IC 30	00(1)\.	\S7 Program(1)					🔶 RUN	At	os < 5.2	-7



6. Tutorial SCL

In deze tutorial wordt de programmeertaal SCL (Structured Command Language)⁹ gebruikt. SCL is een variant op de IEC-taal ST¹⁰. De taal lijkt erg veel op Pascal. Het is een gestructureerde taal waarin mogelijkheden als lussen en beslissingen eenvoudig kunnen worden ingevoerd. Het werken met complexe datastructuren als arrays¹¹ en structures¹² is veel eenvoudiger dan in STL of ladder.

Voor programmeurs van andere talen zoals C zal het werken met SCL eenvoudig uitpakken. Deze tutorial laat slechts een klein deel van SCL zien. Er wordt niet ingegaan op de syntax en mogelijkheden. Zie hiervoor de SCL-handleiding.

6.1 Analoge ingang en temperatuur meten

In deze tutorial wordt gebruik gemaakt van een analoge ingang. Er wordt een thermometer gebouwd met behulp van een NTC-weerstand. De module is al aangesloten op een 24 V voeding. Tevens wordt er een alarmering ingebouwd, die in hoofdstuk 7 weer wordt gebruikt.





⁹ http://www.automation.siemens.com/simatic/industriesoftware/html_76/produkte/software-s7-scl.htm

¹⁰ <u>http://en.wikipedia.org/wiki/Structured</u> text

¹¹ http://en.wikipedia.org/wiki/Array

¹² <u>http://en.wikipedia.org/wiki/Struct</u>

Let op: op de analoge ingang mag maximaal 10 V aangeboden worden! Zie bijlage 4 voor de afleiding van de formule voor de temperatuur. Als alternatief voor de NTC-weerstand kan een potmeter gebruikt worden.

6.2 Project aanmaken

Maak een project aan met de naam tut_scl. Zie figuur 6-2.

New Project		×				
User projects Libra	aries Multiprojects					
Name	▲					
boor_sheet	D:\PROJECTS\STEP7\bo	or_she				
🞒 eerste_keer	D:\PROJECTS\STEP7\ee	rste_k				
🞒 effe	C:\Program Files\Siemens\	Step7\s7proj\effe				
🞒 graphtest	D:\PROJECTS\PLC\proj\g	graphtes				
Iooptest	D:\PROJECTS\PLC\proj\k	ooptest				
read_temp	D:\PROJECTS\STEP7\real	ad_tmp				
🞒 scl_test	D:\PROJECTS\STEP7\sc	Ltest				
l 👘 shoots		aata 🔟				
Add to current mu	ltiproject					
Na <u>m</u> e:		<u>Т</u> уре:				
tut_scl		Project 💌				
		E Library				
Storage location	Storage location					
D:\PROJECTS\STE	D:\PROJECTS\STEP7 Browse					
OK)	Ca	ncel Help				

Figuur 6-2

6.3 Configuratie PLC

De configuratie van de PLC kan opnieuw worden ingevoerd of, net zoals in de vorige tutorial, worden opgehaald uit de PLC. Dit keer wordt wel de naam van de PLC veranderd in SIMATIC_300_THR. Later wordt dit project uitgebreid met een HMI-station (zie hoofdstuk 7).

Voer nu eerst de configuratie in of haal deze op. Start daarna het netwerkconfiguratieprogramma Netpro (dubbelklik op de MPI-bus). Selecteer de PLC-naam en klik op de rechter muisknop. Kies dan **Object Properties**. Vul dan de naam in. Zie figuur 6-3.

Soms verschijnt er een dialoog zoals figuur 6-4. Klik dan op **OK**.

Daarna moet de verandering opgeslagen worden. Doe dit via **Network**→**Save and Compile**. Er verschijnt een dialoog (figuur 6-5). Selecteer Save and check everything.

Figuur 6-6 geeft een beeld van hoe het netwerk er na de naamsverandering uitziet.

Download de configuratie naar de PLC met **PLC**→**Download to Current Project**→**Selected Stations**. Aangezien de PLC-naam anders is dan die in de PLC is opgeslagen, zal figuur 6-7 verschijnen. Hierin is het verschil te zien. Klik op **OK**.

<mark>않NetPro - [tut_scl (Network) D:\PROJECTS\STEP7\tut_scl]</mark> 양 Network Edit Insert PLC View Options Window Help [같 문 등 6 (今) 목 등 () 수 속 종 종 정 종 정 등 () 1 19	•				- D × _ & ×
	1				므프
MPI			Eind	:	m‡ mi
			Sele	ction of the network	
				ROFIBUS DP	
				PROFIBUS-PA	
SIMATIC 300(1)	Proportion CIMATIC	C 200 Chation		Stations	
	Froperces - STMATTC	Southand Station			
	General Interfaces	Settings			
	<u>N</u> ame:	SIMATIC_300_THR			
	Project path:	tut_scl\SIMATIC 300(1)		A 7	
Ethornot(1)	Storage location	D:\PR0JECTS\STEP7\tut_scl			
Industrial Ethernet	of the project:			v	
	<u>A</u> uthor:				
	Date created:	01/03/2008 08:24:31 PM			
	Last modified:	01/03/2008 08:29:42 PM			
	<u>C</u> omment:			<u> </u>	
				7	
<u> ۱</u>					
To display the connection table, please select a module ca					
OPC server or application). To display the network addres					
	OK		Car	ncel Help	
			PRO)FIBUS-DP slaves for	
l Ready PC	Adapter(Auto)	X 291 Y 136 Insert Cho	57,	M7, and U7 (distribute	a rackj

Figuur 6-3



Figuur 6-4

Save and Compile	×
Compile © Compile and check everything © Compile changes only	
OK Cancel Help	

Figuur 6-5



Figuur 6-6



Figuur 6-7

6.4 Symbolenlijst

De symbolenlijst is erg kort. In figuur 6-8 is de invulling te zien. Let op het adres van symbool ANALOG_IN. Dit is PIW272 en niet IW272. Maak de symbolenlijst aan op de gebruikelijke manier. Verder is er een aantal merkerbits gedefinieerd. Die worden gebruikt voor de alarmering. Merk op dat de symboolnamen volledig uit hoofdletters bestaan. Dit is niet verplicht.

S7 Program(1) (Symbols) tut_scl\SIMATIC_300_THR\CPU 315F-2 PN/DP										
	Status	Symbol	Add	ress 🛆	Data type	Comment				
1		TEMPTOOHIGH	М	18.0	BOOL	True = Temperature too high (output)				
2		TEMPTOOLOW	М	18.1	BOOL	True = Temperature too low (output)				
3		RESETALARMS	М	18.2	BOOL	True = Alarms will reset (input)				
4		ALARMSFF	М	18.3	BOOL	Holding relay for alarms				
5		ALARMSDETEGDE	М	18.4	BOOL	Alarms detected edge				
6		TMEAS	MD	20	REAL	Measured temperature				
7		VMEAS	MD	24	REAL	Measured voltage				
8		RNTC	MD	28	REAL	Calculated resistance of the NTC				
9		ANALOG_IN	PIW	272	WORD	Analog input channel 1, note the use of PI				
10										

Figuur 6-8

6.5 SCL bronbestand aanmaken

Het programmeren van SCL-code kan niet direct via de bekende weg. Bij het aanmaken van een OB, FB of FC is de taaloptie SCL niet beschikbaar. Het aanmaken van een SCL-bronbestand gaat via Sources. In het rechter paneel van het projectscherm moet op de rechter muisknop worden geklik en selecteer dan **Insert New Object** – **SCL Source** (zie figuur 6-9). Er verschijnt een object met de naam SCL Source(1). Verander deze naam in ConvADCTempVolt.

🔄 tut_scl D:\PROJECTS\STEF	7\tut_scl	
tut_scl SIMATIC_300_THR G S7 Program(1) Sources G Blocks	Cut Ctrl+X Copy Ctrl+C Paste Ctrl+V Delete Del	
	PLC • SCL Source	
	Object Properties Alt+Return SCL Compile Control File Special Object Properties GRAPH source	
	External Source	

Figuur 6-9

6.6 Invoeren SCL-code

Dubbelklik op de naam om de SCL-editor¹³ te starten. Er volgt een leeg scherm. Zie figuur 6-10.

SUL - [LonVADUTOTempVolt tut_sch/STMATTL_3UU_THR/LPU_3TSF-2 PN/DP]			
			A
			_
			-
4			▶
Press F1 for help.	Ln 1 Col 1	INS MOD	

Figuur 6-10

Voeg nu via het menu **Insert**→**Block Template**→**FC** de structuurcode voor een FC (figuur 6-11). Deze structuur is te zien in figuur 6-12.



Figuur 6-11

¹³ Met deze editor kunnen ook STL- en Graph-bestanden worden gemaakt, dit wordt niet verder uitgewerkt.

Figuur 6-12

Het programma komt in FC3 en er komt een REAL terug, dus moet dat worden aangepast. In figuur 6-13 is de complete code. In bijlage 5 is de code nog eens terug te vinden, maar nu inclusief commentaar.

```
FUNCTION FC3 : REAL
                                                       // Function 3
CONST
     VPOWER := 24.0; // Power Supply
VANALOG MAX := 10.0; // Max analog voltage
ADCVALUE AT VANALOG MAX := 27648; // ADC value code @ ...
A := 0.0235664; // A parameter in Ohm
B := 3863.65; // B parameter in K
RS := 47000.0; // Series resistor of 47 KOhm
RIN := 100000.0; // Input resistance of analog input
K_TO_C := 273.15; // Convert to Celsius
    RS := 47000.0;
RIN := 100000.0;
K_TO_C := 273.15;
END CONST
VAR_INPUT
     ADC VALUE : WORD;
                                                      // The input from the analog channel
END VAR
VAR_OUTPUT
     VMEAS : REAL;
RNTC : REAL;
                                                       // Voltage measured
                                                       // Calculated resistance of NTC
END VAR
VAR TEMP
     Z : REAL;
                                                     // Conversions
     ZI : INT;
END VAR
BEGIN
     // ADC value to integer
     ZI := WORD_TO_INT(ADC_VALUE);
// Voltage measured
     VMEAS := (1.0*ZI) / ADCVALUE_AT_VANALOG_MAX * VANALOG_MAX;
      // Calculate Z as function of ...
     Z := VMEAS/VPOWER;
      // Rntc in Ohms. Function found with the aid of Maple 10.
     RNTC := - (VMEAS*RS*RIN/ (-VPOWER*RIN+VMEAS*RIN+VMEAS*RS));
     // Temperature in degree C. Function found with the aid of Maple 10.
      // This is the return value of the function.
     FC3 := B / LN( -1.0*Z*RS*RIN / (A*(-RIN+Z*RS+Z*RIN)) ) - K TO C;
END FUNCTION
```

Figuur 6-13

In figuur 6-14 is een deel van het programma terug te vinden in de SCL-editor.

戦 SCL - IConvADCToTemoVolt tut_scl\SIMATIC_300	THR\CPIL315E-2 PN/DP1	
File Edit Insert PLC Debug View Options Window		_ 린 ×
	// Europhice 2	
FUNCTION FCS : REAL	// Function 3	-
CONST		
VPOWER := 24.0;	// Power Supply	
VANALOG_MAX := 10.0;	// Max analog voltage	
ADCVALUE_AT_VANALOG_MAX := 27648;	// ADC value code 0	
A := 0.0235664;	// A parameter in Ohm // D persenter in V	
B := 3003.03; PG := 47000 0.	// B parameter in K // Series resistor of 47 KOhm	
RJ := 47000.0	// Input registance of analog input	
K TO C := 273.15:	// Convert to Celsius	
END CONST		
_		
VAR_INPUT		
ADC_VALUE : WORD;	// The input from the analog channel	
END_VAR		
VAR OUTPUT		
VMEAS : REAL;	// Voltage measured	
RNTC : REAL;	// Calculated resistance of NTC	
END_VAR		
VAR_TEMP		
Z : KEAL; ZT · INT·	// Conversions	
END VAR		
		-
T		Þ
1		
1		
Press F1 for help.	Ln 27 Col 1 IN	S MOD ///

Figuur 6-14

6.7 Compileren SCL-code

Nu het programma ingevoerd en compleet is, moet het gecompileerd worden. De *compiler* vertaalt de SCL-code naar STL, de assemblertaal voor de PLC.

Selecteer het menu **File**→**Compile** (sneller is via **Ctrl+B**). De compiler gaat nu aan het werk. De compileervoortgang wordt weergegeven, zie figuur 6-15. Als blok FC3 al bestaat (en dat komt natuurlijk best vaak voor), zal gevraagd worden of dit blok moet worden overschreven. Klik dan op **Yes** of **All**. Zie figuur 6-16.

Compile: Co	nvADCToTempVolt with 3 blocks
Source:	ConvADCToTempVolt
Block:	FC3
Lines:	0 lines to go
	99%
	Cancel

Figuur 6-15



Figuur 6-16

Eventuele fouten worden gerapporteerd en moeten verbeterd worden. Zie voor een voorbeeld van een mislukte compilatie figuur 6-17.



Figuur 6-17

6.8 Tweede SCL-bronbestand

Er wordt nu een tweede SCL-bestand aangemaakt. De code wordt in FC4 geplaatst en zet twee merkerbits indien de temperatuur te hoog of te laag is.

Maak eerst een nieuw SCL-bestand aan en noem dit CheckMinMax. Zie figuur 6-18.



Figuur 6-18

De code is weergegeven in figuur 6-19. Let vooral op de teruggaveparameter van FC4, deze is van het type VOID, oftewel er is geen teruggaveparameter. De teruggave wordt geregeld via twee VAR_OUTPUT-parameters.

De code van CheckMinMax is zeer eenvoudig. Het zijn slechts twee simpele IF-statements. Hiermee is ook goed te zien dat SCL gebruikt kan worden voor simpele beslissingen. Natuurlijk had de code veel eenvoudige gekund en doorgewinterde PLC-programmeurs hadden dit in LAD opgelost.

In bijlage 5 is de code nog eens afgebeeld.

```
FUNCTION FC4 : VOID
CONST
    C_TEMPTOOHIGH := 30.0;
                                            // Higher bound temperature
    C_TEMPTOOLOW := 15.0;
                                            // Lower bound temperature
END CONST
VAR_INPUT
    TEMPERATURE : REAL;
                                            // The current temperature
END VAR
VAR_OUTPUT
    TEMPTOOHIGH : BOOL;
                                            // Signals temperature too high
    TEMPTOOLOW : BOOL;
                                            // Signals temperature too high
END VAR
BEGIN
   IF TEMPERATURE >= C TEMPTOOHIGH THEN
                                            // If current temperature too high...
                                            // signal it.
       TEMPTOOHIGH := true;
   ELSE
       TEMPTOOHIGH := false;
   END IF;
    IF TEMPERATURE <= C_TEMPTOOLOW THEN
                                            // If current temperature too low...
                                            // signal it.
        TEMPTOOLOW := true;
    ELSE
        TEMPTOOLOW := false;
   END IF;
END_FUNCTION
```



6.9 OB1 programma

Nu de SCL-routine klaar zijn moet ze via OB1 worden aangeroepen met de juiste parameters. Maak OB1 op de bekende wijze aan (via Blocks) en start de LAD-editor. Sleep vanuit linker kader onder FC Blocks FC3 naar edit-kader. Zie figuur 6-20.



Figuur 6-20

Na het plaatsen van het FC-blok geeft de editor automatisch de formele parameters weer met rode vraagtekens op de plaatsen waar de actuele parameters moeten worden ingevoerd. In figuur 6-21 zijn deze ingevuld. Tevens wordt FC4 ook aangeroepen en is er een stukje FBD gemaakt voor de alarmering¹⁴.



Figuur 6-21

Sla OB1 op de bekende manier op.

6.10 Laden PLC programma

Dit gaat weer op de bekende wijze. Zorg ervoor dat de PLC in STOP staat en laadt OB1, FC3 en FC4.

6.11 Monitoren temperatuur

Het monitoren kan eenvoudig met een variabelentabel. Maak de tabel aan op de bekende wijze en ga naar de monitor-modus (klik op het brilletje). Let op dat een aantal gegevens in floating point genoteerd zijn. Zie figuur 6-22.

¹⁴ De lezer wordt uitgedaagd om netwerk 3 te bestuderen.

*	Var -	VAT1					
lat	ble <u>E</u>	dit <u>I</u> ns	ert P <u>L</u> C V <u>a</u> riable	e <u>V</u> iew <u>U</u> ptions	Window Help		
-Æ	<u> </u>) 😂	🖬 🎒 👗 🖣		× <u>*</u> 1	<u>*?</u> 🔮 🚳 🛩	60° 42° //w
K	VA	1 @	tut_scl\SIMATI	C_300_THR\CPU	315F-2 PN/D	P\\$7 □ ×	
	r A	ddress	Symbol	Display format	Status value	Modify value	
1	M	D 20	"TMEAS"	FLOATING_POINT	31.64032		
2	M	D 24	"VMEAS"	FLOATING_POINT	3.101852		
3	M	D 28	"RNTC"	FLOATING_POINT	7543.841		
4	P	W 272	"ANALOG_IN"	HEX	VV#16#2180		
5	M	18.0	"TEMPTOOHIGH"	BOOL	📘 true		
6	M	18.1	"TEMPTOOLOW"	BOOL	false		
7	M	18.2	"RESETALARMS"	BOOL	false	true	
8	M	18.3	"ALARMSFF"	BOOL	📘 true		
9							
-							1
ut_:	scl/SII	MATIC_	300_THR\\\$7 Pr	ogram(1)			RUN

Figuur 6-22

In figuur 6-22 is te zien dat de temperatuur boven de 30 °C is gestegen. De merkers TEMPTOOHIGH en ALARMSFF zijn beide hoog (true). Er is nu een alarm getriggerd. Als de temperatuur nu onder de 30 °C zakt, zal TEMPTOOHIGH weer laag (false) worden, maar ALARMSFF blijft hoog. Om deze laag te krijgen moet RESETALARMS op even hoog en daarna weer laag worden. Dit kan makkelijk via de VAT. Merk op dat de alarmering via merkers gaat en niet via uitgangen. In de tutorial van WinCC Flexible wordt nog terug gekomen op de alarmen.

7. Tutorial WinCC flexible

Met behulp van WinCC flexible kan informatie uit de PLC op een zogenaamde Operator Panel worden afgebeeld en het is mogelijk om het panel informatie in de PLC te brengen. Op deze manier kan een fabriek schematisch worden afgebeeld. Dit visualiseren, controleren en besturen wordt ook wel SCADA¹⁵ genoemd, *Supervisory Control And Data Acquisition*.

In het PLC-lab zijn er echter geen operator panels, hiervoor wordt de PC gebruikt. Deze tutorial borduurt voort op de SCL-tutorial.

In deze tutorial worden slechts een paar mogelijkheden van WinCC Flexible gedemonstreerd. Als eerste wordt in het STEP7-project een HMI-station toegevoegd. Daarna wordt het geconfigureerd, zoals een PLC. Vervolgens worden een paar uitvoervelden en een knop aangemaakt in de WinCC Flexible editor en als laatste wordt de Runtime gestart.

7.1 Aanmaken nieuw project ¹⁶

Begin een nieuw project zoals is aangegeven in het vorige hoofdstuk en noem het project tut_wincc_flex. Zie figuur 7-1. Vul de bekende gegevens in.

New Project		×			
User projects Libraries	Multiprojects				
Name	Storage path				
read_temp	D:\PROJECTS\STEP7	/\read_tmp			
🗃 traf_lad	D:\PROJECTS\STEP7	/\traf_lad			
🞒 tut_graph	D:\PROJECTS\STEP7	"\tut_grap			
but_graph_extended	D:\PROJECTS\STEP7	/\tut_grex			
🞒 tut_scl	D:\PROJECTS\STEP7	"\tut_scl			
🎒 tutorial	D:\PROJECTS\STEP7	Ntutorial			
Add to current multiproj	ect				
Name:		Type:			
but wcc flex					
		Project			
		E Library			
Storage location (path):					
D:\PROJECTS\STEP7		<u>B</u> rowse			
-					
		neel Hele			
Lancei Help					



¹⁵ <u>http://nl.wikipedia.org/wiki/SCADA</u>

¹⁶ Deze stap kan worden overgeslagen als deze tutorial direct volgt op de SCL-tutorial. De projectnaam blijft dan tut_scl. Ga dan verder met paragraaf 7.3.

7.2 Configuratie uploaden

In het vorige hoofdstuk is een nieuwe configuratie van de PLC opgezet en geladen in de PLC. Deze configuratie kan teruggehaald worden uit de PLC en zo gebruikt worden in het nieuwe project. Hiervoor moeten de configuratiegegevens ge-*upload* worden vanuit de PLC naar het project. Hierbij worden ook alle programma's mee gekopieerd (OB, FC etc).

Doe een upload via het menu **PLC**→**Upload** Station to **PG**. Zie figuur 7-2.



Figuur 7-2

Er verschijnt nu een scherm met daarin alle aangesloten PLC's. Midden onderaan het scherm verschijnt een knop **View**. Als er geen PLC's zichtbaar zijn, klik hier dan op. Na een tijdje verschijnt onder Accessible Nodes de PLC die op het prakticum is aangesloten. Selecteer de gegeven PLC en klik op **OK**. Zie figuur 7-3.

Let er wel op dat alleen de blokken (Blocks) worden gekopieerd. De symbolentabel (Symbols) en de bronbestanden (Sources) worden niet gekopieerd. Dit komt omdat in de PLC alleen gecompileerde code is opgeslagen. Als er toch veranderingen moeten worden gemaakt aan de programmatuur levert dat problemen op.

Het is mogelijk om de blokken, bronbestanden en symbolentabel te kopieren uit een ander project. Open dit andere project, selecteer Blocks, Sources en Sybols en sleep ze op de juiste plaats in het huidige project.

Select Node A	Select Node Address							
Which module do you want to reach?								
<u>R</u> ack:								
<u>S</u> lot:	0 📑							
Target Station:	⊙ <u>L</u> oca	I						
	C Can I	pe reached by n	neans of gatewa	y				
Enter connect	ion to target sta Module tune	ation: Station name	Module name	Plant designatio	n			
2	CPU 315F	SIMATIC 3	CPU 315F	Thank designate				
Accessible Node	es							
2	CPU 315F	SIMATIC 3	CPU 315F					
lindate								
OK				Cancel	Help			

Figuur 7-3

De configuratie wordt nu uit de PLC geladen en in het project opgenomen. Zie figuur 7-4.

Upload to PG	×
Station:	
SIMATIC 300(1)	
Module:	
[0/2/0] CPU 315F-2 PN/DP	
-	
	Cancel

Figuur 7-4

Het projectoverzicht is nu de PLC én de MPI-bus zichtbaar. Als er twee MPI-bussen zichtbaar zijn, moet er één verwijderd worden. Zie figuur 7-5 als voorbeeld.



Figuur 7-5

7.3 HMI station plaatsen

Voeg nu via het menu **Insert**-Station-SIMATIC HMI Station een HMI-station toe aan het project. Zie figuur 7-6.

SIMATIC N	lanager - [tut_wcc	c_flex D:\PROJECTS\STEP7\
🎒 File Edit	Insert PLC View	Options Window Help
🗅 🧀 👫	Station	1 SIMATIC 400 Station
	Subnet	2 SIMATIC 300 Station
	Program	 3 SIMATIC H Station
	C7 C=6	4 SIMATIC PC Station
	or oorware	5 SIMATIC HMI Station
	a / Block	6 Other Station
	MI7 Sortware	7 SIMATIC S5
	Symbol Table	8 PG/PC
	77	

Figuur 7-6

Er wordt nu een venster geopend waarin het type HMI-station kan worden gekozen. Kies hier voor PC-WinCC flexible Runtime. Zie figuur 7-7.

Nu wordt de WinCC flexible Engineering Station software gestart, zie figuur 7-8.

Let op: dit kost enige tijd. Heb geduld! Ga desnoods even koffie drinken!

Properties - WinCC flexible RT			×
General Device type			
	Select device by name and version		
	Additional operating devices	Device version	-
ОК		Cancel He	P

Figuur 7-7

SIMATIC WinCC flexible
SIMATIC WinCC flexible ES is starting

Figuur 7-8

Hierna is het HMI station toegevoegd aan het project (zie figuur 7-9).

but_wcc_flex D:\PROJECT		
E 20 tut_wcc_flex ⊕-1 SIMATIC 300(1) ⊕ SIMATIC HMI Station(1)	SIMATIC 300(1)	SIMATIC HMI Station(1)

Figuur 7-9

7.4 MPI-netwerk

Nu moeten de PLC en het HMI station verbonden worden via het MPI-netwerk. Dit wordt gedaan via het programma NetPro, wat onderdeel is van STEP7. Dubbelklik op MPI(1) in het projectvenster (figuur 7-9).

Nu wordt NetPro geopend met hierin de MPI-bus, de Ethernet-bus, de PLC en het HMI-station. Deze laatste twee zijn nog niet aangesloten op het netwerk (als dat wel zo is, kan deze stap worden overgeslagen).

Om nu de PLC te koppelen aan de MPI-bus, moet het volgende gedaan worden: klik met de muis op de rode vierkantje en houd de muisknop ingedrukt. Sleep nu naar de MPI-bus en laat de muisknop vervolgens los. De PLC is dan verbonden met het netwerk. Zie figuur 7-10.



Figuur 7-10

7.5 HMI-station configureren

Het HMI-station moet, net als de PLC, geconfigureerd worden. Dubbelklik nu op het HMI station. De hardwareconfiguratie kan nu worden ingevoerd. Zie figuur 7-11.



Figuur 7-11

Noot: Het idee is dat een SIMATIC HMI-station net als een S7-300 een rack en slots heeft. Deze worden Virtual Rack en Virtual Slots genoemd. Hierdoor blijven een HMI station en een S7-300 op dezelfde manier te configureren.

Zoek in het venster aan de rechterkant naar de CP5511-netwerkkaart via SIMATIC HMI Station->CP Profibus->CP5511 en kies de tweede regel. Sleep de geselecteerde regel naar Index 1. Zie de rode pijl.

Noot: Merk op dat in de PC, want daar draait straks de WinCC Flexible Runtime op, helemaal geen CP5511-kaart zit! De PC is met de PLC verbonden via een USB-adapter. De PC zorgt er automatisch voor dat de communicatie op juiste wijze in gang wordt gezet.

Zodra dat gebeurd is, volgt een eigenschappenscherm voor Profibus-instellingen van de CP5511netwerkkaart. Klik nu op **OK** zonder iets te veranderen. Zie figuur 7-12.

Properties -	PROFIBUS int	erface	CP 5511 (F	0/S1)				×
General	Parameters							
<u>A</u> ddress:	E	•		If a subnet is s the next availa	elected, able addre	ss is sug	gested.	
<u>S</u> ubnet: not r	networked					<u>N</u> e	w	
						Prope	erties	
						De	lete	
OK					Cano	el	Help	

Figuur 7-12

De CP5511 is nu zichbaar op Index 1 in het configuratiescherm. Dubbelklik hier nu op om weer de eigenschappen te openen, maar kies nu bij Interface voor de MPI-bus. Zie figuur 7-13.

Er wordt dan een bevestigingsvenster geopend met de melding dat de Profibus-instellingen verloren zullen gaan en dat de netwerkkaart aan de MPI-bus wordt gekoppeld. Klik op Yes. Zie figuur 7-14.

Klik vervolgens in het Interface kader op **Properties**. Nu worden de eigenschappen van de MPI-interface getoond. Als subnet moet nu MPI(1) gekozen worden. In het voorbeeld wordt verder netwerkadres 3 gekozen, maar dit mag ook een ander zijn met uitzondering van 0 en 2. Zie figuur 7-15.

Klik nu een aantal keer op OK en sla de nieuwe configuratie op met $Station \rightarrow Save$ and Compile. Sluit de hardwareconfiguratie af via menu $Station \rightarrow Exit$ of Alt+F4.

Er komt nog een melding of de wijzigingen moeten worden opgeslagen en daarna wordt een compileerslag gedaan.

roperties - CP 5511	2				
General Assignment (Derating Mode Reserve LSAPs Diagnostics				
Short Description: CP 5511 SIMATIC NET CP 5511 PROFIBUS, S7 connections, DP master, DPV1, DP slave, PG functions, DP master class 2, PCMCIA, routing, SIMATIC NET CD 7/2001 SP5					
Order No./ firmware	6GK1 551-1AA00 / V6.0.5				
<u>N</u> ame:	CP 5511				
Interface Iype: PROFIBUS Address: MPI PROFIBUS Properties					
<u>C</u> omment:	×				
OK	Cancel Help				

Figuur 7-13

Object Properties (4502:715)						
<u>!</u>	Do you want to network the module on MPI and lose all the PROFIBUS settings of the module? If you do not want to do this, select 'No'. None of the settings modified in the Properties dialog will then be saved.					
(<u>Y</u> e	s	No		Help		

Figuur 7-14

Properties - MPI interface CP 5511 (R0/S1)				×
General Parameters				
Address: 3				
Highest address: 31				
Transmission rate: 187.5 Kbps				
<u>S</u> ubnet:				
not networked MPI(1) 187.5 Kbps		1	<u>√</u> ew	
		P <u>r</u> o	perties	
		[)eļete	
ОК	Car	ncel	Help	

Figuur 7-15

Nu is de koppeling van het HMI-station en de MPI-bus klaar. In de volgende figuur is dat goed te zien (figuur 7-16).



Figuur 7-16

Selecteer via het menu **Network**→**Save and Compile**. Er volgt een dialoog waarin gekozen moet worden voor **Compile and check everything**. Zie figuur 7-17.

Save and Compile		×					
Compile							
Compile and c	heck <u>e</u> verything						
C Compile chan	C Compile changes only						
	1						
UK	Cancel	Help					

Figuur 7-17

Indien het om een nieuw opgezet project gaat, wordt nog even gevraagd hoe de berichten in de CPU gecodeerd moeten worden (figuur 7-18). Selecteer hier de eerste optie. Deze stap wordt overgeslagen als het al een keer eerder gevraagd is, bijvoorbeeld bij een oud project.



Figuur 7-18

Als alles goed gaat volgt figuur 7-19.

🔥 Outputs	s for consistency check for I	D:\PROJECTS\S	FEP7\tut_v	vcc_\tu 💶 🗙
<u>F</u> ile <u>E</u> dit				
Messag	Message	Туре	Project	Project path
	No errors.			

Figuur 7-19

7.6 Symbolenlijst

Net als in de vorige tutorial moet een symbolenlijst gemaakt worden. Zie ook paragraaf 6.4. In figuur 7-20 worden de symbolen weegegeven.

📥 S7	S7 Program(1) (Symbols) tut_wcc_flex\SIMATIC 300(1)\CPU 315F-2 PN/DP							
	Status	Symbol	Add	ress 🛆	Data type	Comment		
1		TEMPTOOHIGH	М	18.0	BOOL	True = Temperature too high (output)		
2		TEMPTOOLOW	М	18.1	BOOL	True = Temperature too low (output)		
3		RESETALARMS	М	18.2	BOOL	True = Alarms will reset (input)		
4		ALARMSFF	М	18.3	BOOL	Holding relay for alarms		
5		ALARMSDETEGDE	М	18.4	BOOL	Alarms detected edge		
6		TMEAS	MD	20	REAL	Measured temperature		
7		VMEAS	MD	24	REAL	Measured voltage		
8		RNTC	MD	28	REAL	Calculated resistance of the NTC		
9		ANALOG_IN	PIV	272	WORD	Analog input channel 1, note the use of PI		
10								

Figuur 7-20

7.7 Starten WinCC flexible

Zoek in de Simatic Manager naar WinCC flexible RT. Dit is te vinden in het rechterscherm via tut_wcc_flex→SIMATIC HMI Station(1)→WinCC flexible RT. Selecteer deze regel en klik op de rechtermuisknop (contextmenu). Selecteer dan Open Object. Zie ook figuur 7-21.

but_wcc_flex D:\PROJEC	TS\STEP7\tut_v	vcc_		<u>- 0 ×</u>
tut_wcc_flex 	Creens Recipes Reports Runtime Use	Communicati Ma Historical Dai Ma Text and Gra er Administration	on 🧏 Alarm Management ta 🖅 Scripts phics Lists Parice Settings	
WinCC flexible R WinCC flexible R Screens Communicati Airm Manag Accipes Recipes	Open Object Cut Copy Paste	Ctrl+Alt+O Ctrl+X Ctrl+C Ctrl+V		
Tristing Day Tristing Day	Delete Rename Object Properties	Del F2 Alt+Return		

Figuur 7-21

WinCC flexible wordt nu gestart. Dit is te zien aan figuur 7-22.

SIMATIC WinCC flexible - open HMI station		
SIMATIC WinCC flexible ES is starting		



Let op: heb geduld, dit kan enkele minuten duren! Tijd voor een kopje koffie!

In figuur 7-23 wordt een overzicht gegeven van het WinCC Flexible scherm. Aan de linkerkant staan de onderdelen die in het project kunnen voorkomen (niet alles hoeft te worden gebruikt). Rechts staan de objecten die in het project kunnen worden gebruikt. Middenboven is het scherm wat geopend is afgebeeld (dat gekozen zijn aan de linkerkant), en onder een contextgevoelig scherm dat informatie geeft over (objecten op) het scherm erboven.



Figuur 7-23

7.8 Device settings

Klik in het linkerscherm op Device Settings. In het scherm verschijnt nu een lijstje met opties. Eén daarvan is weer Device Settings. Dubbelklik hier op en een nieuw scherm wordt geopend, waarin diverse gegevens kunnen worden ingevuld. Zo dient de schermgrootte te worden aangepast naar 800 bij 600 en kan de naam van de programmeur worden ingevuld. Zorg dat het vinkje bij Full-screen Mode weg is. De naam van het startscherm kan worden geselecteerd. Deze tutorial gebruikt echter maar één scherm. Zie figuur 7-24.
eral	
Device	
Name SIMATIC HMI Station(1)	Start screen_1
Device type WinCC flexible Runtime 1.1.1	Screen resolution 1280x1024
	1600x1200 1290-1024
Author <type hier="" je="" naam=""></type>	12004x768
Comment Mijn eerste WinCC Flexible proj	ject 640x480
	anan Marina Arra Ma
Runtime settings	Runtime services
Use on-screen keyboard 🗖	Sm@rtAccess or Service: Start up Sm@rtServer 🗖
Lock task switching	Sm@rtService: HTML pages 🗖
	Sm@rtAccess: Web service (SOAP) 🗖
Displays limit tooltips 🔽	Sm@rtAccess: SIMATIC HMI HTTP Server
Transfer names 🔽	Act as OPC server
Display script comments 🔽	Name of SMTP server
Project ID 0	Nores of SMTD condex
	SMTP Authentication

Figuur 7-24

7.9 Cyclustijd toevoegen

Het opvragen van de gegevens uit de PLC door het HMI-station wordt cyclisch uitgevoerd. De cyclustijd hiervoor is instelbaar. Een seconde is echter te langzaam en 100 ms is weer zo snel dat het teveel belasting voor de PLC kost. Er wordt een eigen cyclustijd aangemaakt van 200 ms.

Dubbelklik in het rechterscherm op Communications zodat een lijst wordt geopend. Dubbelklik op Cycles om het invoeren mogelijk te maken. Voeg nu een nieuwe cycle toe zoals in figuur 7-25 is opgegeven. Deze cyclustijd zal later worden gebruikt.

Screen_1	Device Settin	igs SPCycles	۲ کا کا ک
Cycle time	Cycle unit	Name	Last changed Comment
1	Hour	1 h	10/31/2008 8:5
1	Minute	1 min	10/31/2008 8:5
1	Second	1 s	10/31/2008 8:5
10	Second	10 s	10/31/2008 8:5
100	Millisecond	100 ms	10/31/2008 8:5
2	Second	2 s	10/31/2008 8:5
5	Second	5 s	10/31/2008 8:5
500	Millisecond	500 ms	10/31/2008 8:5
200	Millisecond	200 ms	10/31/2008 8:0 Self defined 200 ms cycle



7.10 Verbinding met PLC maken

In hetzelfde lijstje staat ook Connections. Hier kunnen verbindingen met PLC's worden opgenomen, maximaal acht. In het lab is er maar één gekoppeld aan de MPI-bus.

Dubbelklik op Connections om het scherm te openen. Automatisch wordt de verbinding in de eerste regel gepresenteerd onder de naam Connection_1. Let er op dat de verbinding actief moet zijn, anders werkt het geheel niet. Zie rode pijl in figuur 7-26.

Name	Active	Communication drive	r Station	Partner	Node	Online
Connection_1	On	▼ SIMATIC S7 300/400	 \tut_wcc_flex\SIM 	▼ CPU 315F-2 Pf	N/DP 🔻 MPI/DP	▼ On
rameters Are	a pointer					
rameters Are WinCo Ru	a pointer	ce			Station	
Type TTY Ru TYPE TTY Ru Ru Ru Ru Ru Ru Ru Ru Ru Ru	E flexible Interfa CP 55: Baud rate Address	ce III ▼ HMI device	Profile MPI Highest station address ()	etwork FSA)	Address	PLC device

Figuur 7-26

Onder in de figuur is diverse informatie beschikbaar zoals de interface en adressen.

7.11 Tags toevoegen

Via tags kan informatie uit de PLC gehaald worden en in het WinCC flexible systeem verder worden verwerkt. Tags kunnen direct zijn zoals M0.0 of I125.2, maar er kan ook gebruik gemaakt worden van de symbolentabel.

Dubbelklik in het rechterscherm op Tags. Het Tags scherm wordt geopend. Open nu het contextmenu (rechtermuisknop) en kies Add Tag. Zie figuur 7-27.

Screen_1	Device !	Settings	Cycles S ^m Con	nections 🔩 Tags
Name	Conne	ection	Data type	Symbol
X	Cu <u>t</u>	Ctrl+X]	
rn i	<u>С</u> ору	Ctrl+C		
<u> </u>	F <u>l</u> at Copy			
6	<u>P</u> aste	Ctrl+V		
	<u>R</u> eplace	Ctrl+E		
~=	Add Tag			
×	<u>D</u> elete	Delete		
<u>(</u>)()	Cross-Referen	nce		
121	<u>R</u> ewire			
	R <u>e</u> connect		-	
	4		-	

Figuur 7-27

Nu wordt automatisch een standaard invulling aan de tag gegeven, die veranderd moet worden. Klik aan de rechterkant van het vakje onder de kolom Symbol op het pijljte voor een *dropdown*menu (zie de rode pijl). Er wordt nu een klein scherm getoond met daarin het STEP7-project. Vouw dit helemaal uit totdat het onderdeel Symbols te zien is. Aan de rechterkant kan nu uit de symbolen gekozen worden. Zie figuur 7-28.

ï

	Screen_1	Device Settings	Cycles s	Connections real Tags	Address	Array co	A Aunt	GS Acc	
	199_4			Project Project Project Project Stut_wcc_flex Gm SIMATIC Gm SIMTIC	300(1) 315F-2 PN/DP 7 Program(1) 3 Symbols 3 DB	ALARMSDETEGDE ALARMSDETEGDE ALARMSFF ANALOG_IN RESETALARMS RNTC TEMPTOOHIGH TEMPTOOLOW TMEAS VMEAS	Info M PIW M MD M MD MD	18.4, BOOL 18.3, BOOL 272, WOR 18.2, BOOL 28, REAL, 18.0, BOOL 20, REAL, 24, REAL,	
-		•						Þ	

Figuur 7-28

De tag-naam en het datatype worden automatisch overgenomen uit de symbolentabel. Op deze manier kunnen ook variabelen uit Data Blocks worden gekoppeld aan een tag.

Als laatste moeten en cyclustijden worden aangepast; die staan standaard op 1 seconde. Onder de kolom Acquisition cyle kunnen deze worden ingevuld. Kies voor de 200ms cyclustijd. Zie figuur 7-29.

		-	TAGS		
	Array count	Acquis	sition cycle Con		
-	1	1 s	🕶 Moto		
	1	Icon	Name	Info	
	1	-	1 h	1 Hour	
	-	-	1 min	1 Minute	
	1		1 s	1 Second	
			10 s	10 Second	
			100 ms	100 Millisecond	
		-	2 s	2 Second	
			200 ms	200 Millisecond	
		-	5 s	5 Second	
			500 ms	500 Millisecond	
		<<	< New		X

Figuur 7-29

Voeg nu alle tags toe. Zie figuur 7-30.

	Screen_1 📮 Dev	vice Settings 📑	Cycles <mark>"S</mark> [™] Co	onnections 🛛 🔫 💻 Tags			
							TAGS
	Name	Connection	Data type	Symbol	Address	Array count	Acquisition cy
	TEMPTOOHIGH	Connection_1	Bool	TEMPTOOHIGH	M 18.0	1	200 ms
	TEMPTOOLOW	Connection_1	Bool	TEMPTOOLOW	M 18.1	1	200 ms
	RESETALARMS	Connection_1	Bool	RESETALARMS	M 18.2	1	200 ms
	ALARMSEF	Connection_1	Bool	ALARMSEE	M 18.3	1	200 ms
	ALARMSDETEGDE	Connection_1	Bool	ALARMSDETEGDE	M 18.4	1	200 ms
	TMEAS	Connection_1	Real	TMEAS	MD 20	1	200 ms
	VMEAS	Connection_1	Real	VMEAS	MD 24	1	200 ms
	RNTC	Connection_1	Real	RNTC	MD 28	1	200 ms
	ANALOG_IN	Connection_1	Word	ANALOG_IN	PIW 272	1	200 ms
-r		•					F

Figuur 7-30

7.12 Scherm opbouwen

Het laatste onderdeel is het scherm. Hierop zijn buttons, I/O fields, sliders, switches, bars etc. te plaatsen.

Op het scherm worden drie I/O fields geplaatst. Daarop worden de temperatuur (TMEAS), de spanning over de NTC (VMEAS) en de weerstandswaarde van de NTC (RNTC) afgebeeld.

Daarna worden er die cirkels geplaatst die gebruikt gaan worden als signaleringslampen (één van de leuke trucs die je met objecten kan doen).

Vervolgens wordt er een knop geplaatst waarmee het alarmsignaal kan worden gehonoreerd (*acknowlegde*, in jargon ook kwiteren genoemd).

Als laatste wordt er een "ouderwedse" meter geplaats waarop de temperatuur te zien is.

Plaats nu als eerste een I/O Field op het scherm. Selecteer in het eigenschappenscherm onderaan General. Selecteer als Mode nu Output, en bij Tag de tag TMEAS. Het formaat is Decimal en het patroon is 999. Zie de rode pijlen in figuur 7-31.



Figuur 7-31

Dit wordt ook gedaan voor de tags VMEAS en RNTC. Bij RNTC moet het patroon echter op 99999.9 staan, anders kan straks de waarde niet worden gerepresenteerd. Plaats ook tekst (Textfield) bij de I/O Fields. Zie alvast figuur 7-32.

Om de alarmen weer te geven worden cirkels geplaatst die dienst doet als signaleringlampen. De kleur van de cirkel is aan te passen via de waarde van een tag. Zo kan de cirkel groen worden als een tag de (binaire) waarde 0 heeft en rood voor de waarde 1. Dit wordt gedaan via een *animation*.

Sleep vanuit het rechterscherm een circel naar het grijze vlak. Onderaan zie je alle eigenschappen van de circel. Klik op Animations en daarna op Appearance. Vul onder Tag de naam van TEMPTOOLOW in, hier wordt weer een scherm geopend zoals in figuur 7-28. Selecteer daarna het type Bit (STEP7 noemt het een Bool).

Onder het kopje Value moeten nu twee regels worden aangemaakt. Dit kan het beste worden gedaan door per regel even te dubbelklikken. Er verschijnen nu twee regels met 0 en 1 onder Value, maar nog niet de goede kleuren. Door te dubbelklikken op een kleurvlakje verschijnt een selectieschermpje. Kies dan voor rood en groen. Zie figuur 7-32.



Figuur 7-32

Plaats nu een knop (Button). Onderaan worden de eigenschappen van de knop weergegeven. Vul deze in zoals is weergegeven in figuur 7-33.



Figuur 7-33

Klik nu op Events. Kies nu onderdeel Press. Onder de lijst Function List kan nu voor een voorgeprogrammeerde functie worden gekozen. Kies uit de lijst nu de functie SetBit. Zie figuur 7-34.

Button_1 (Butte	on)	? ×
General Properties Animations		Function list
 Events Click Press Release Activate Deactivate Change 		



Er wordt nu een regel aangemaakt met de functie SetBit. Deze moet op een tag werken, maar die is nog niet ingevuld. Op de tweede regel staat achter Tag (InOut) nu <No value>. Klik hier nu op en vul de tagnaam RESETALARMS in. Er komt een een selectiescherm zoals in figuur 7-28. Zie figuur 7-35.

Button_1 (But	ton)		?					
 General Properties 	×+		Function list					
Animations	1	🖃 SetBit						
		Tag (InOut)	<no value=""></no>					
Press	2	<no function=""></no>	🖃 🚧 Project	Icon	Name	Info)	
Release			🚊 🛄 SIMATIC HMI Station(1)	-	ALARMSDETEGDE	M	18.4, BOOL	
 Activate 			 🖻 😽 Communication	-200	ALARMSEE	M	18.3, BOOL	
 Deactivate 			 	-1	ANALOG_IN	PIW	272, WOR	
Change			🖮 🎒 tut_wcc_flex	~=	RESETALARMS	M	18.2, BOOL	
				-	RNTC	MD	28, REAL,	
			E- 📓 CPU 315F-2 PN/DP	-	TEMPTOOHIGH	М	18.0, BOOL	
			 S7 Program(1)	-12	TEMPTOOLOW	M	18.1, BOOL	
			मिल्स्नी Symbols	-200	TMEAS	MD	20, REAL,	
				-21	VMEAS	MD	24, REAL,	
			>>> New					V

Figuur 7-35

Figuur 7-36 toont het resultaat.

Button_1 (But	tton)			?
 General Properties 	×			Function list
Animations	1	🖃 SetBit		
Events Click		Tag (InOut)	RESETALARMS	
🔶 Press	2	<no function=""></no>		•
Release				
 Activate Deactivate 				
 Change 				

Figuur 7-36

Hetzelfde moet gedaan worden maar nu voor Events-Release en de functie ResetBit. Zie figuur 7-37.

Button_1 (Butto	n)		(? ×
 General Properties 	×		Function list
Animations	1	🖃 ResetBit	
Events Click		Tag (InOut)	RESETALARMS
Press	2	<no function=""></no>	•
🔶 Release			
 Activate Departments 			

Figuur 7-37

Als laatste wordt een *gauge* geplaatst, die er uitziet als een ouderwedse manometer. De meter heeft een bereik van 10 °C tot 40 °C, en drie kleurenbanden voor het aangeven van de status.

De gauge is in het linkerscherm te vinden onder Enhanced Objects. Plaats de gauge boven de knop. Na plaatsing ziet de gauge er uit als in figuur 7-38.



Figuur 7-38

Vul onder General van het eigenschappenscherm van de gauge in zoals is opgegeven in figuur 7-39. Selecteer daarna Properties en vul deze in zoals is aangegeven in figuur 7-40. Tevens is te zien hoe de gauge er uitziet.

Gauge_1 (Ga	iuge)	? ×
 General Properties Animations 	General	General
Events	Label Temp Unit C	 Display decimals Display peak value
	Process Tag TMEAS Cycle 200 ms	Background graphic

Figuur 7-39

Gauge_1 (Gau	ige)	(? (x	
General Properties		Scale	
Appearance	Range		
 Appearance Layout Text Scale Misc Animations 	Value Max 40 Min 10	Tag Cycle Angle Image: Cycle 45 Image: Cycle -225	20.0 ^{25.0} 20.0 ^C 30.0 15.0 35.0
	Scale	Danger & Warning	
	Gradation 5	Display normal range 🔽 Warning range start 25 🛛 🗹 Danger range start 30 🕅 🗹	



7.13 Starten Runtime

Nu alles ingevuld is, kan een *consistency check* gedaan worden. Hiermee wordt gekeken of het project consistent is. Start de check via Project→Compiler→Check Consistency. Indien het project in orde is, wordt niets gemeld. Zie figuur 7-41.

Output			P X
Time	Category	Description	
23:34:31.42	Compiler	Compiling started	
23:34:31.42	Compiler	Linking target 'SIMATIC HMI Station(1)'	
23:34:31.42	Compiler	Number of PowerTags used: 7	
23:34:31.42	Compiler	Succeeded with 0 error(s), 0 warning(s),	
23:34:31.42	Compiler	Time stamp: 1-11-2008 23:34 - used 66112 bytes	
23:34:31.85	Compiler	Compiling finished!	

Figuur 7-41

Start nu de runtime via Project→Compiler→Start Runtime. Het "programma" wordt nu gecompileerd en de runtime wordt gestart. Dit kan lang duren. In figuur 7-42 is de Runtime te zien. Probeer nu de temperatuur boven de 30 °C te krijgen. Probeer daarna het alarm te kwiteren.



Figuur 7-42

7.14 Afsluiten Runtime

De Runtime kan worden afgesloten door de toetscombinatie **Alt+F4** in te drukken.

8. Tutorial Data Block

In deze tutorial wordt even kort gedemonstreerd hoe een data block wordt opgezet en gebruikt. Als eerste wordt een data block aangemaakt binnen de Simatic Manager, daarna wordt het ingevuld met een array. Er wordt een kort stukje SCL-code gegeven waarin het data block wordt gebruikt.

Als project wordt het SCL-project of het WinCC Flexible project gekozen.

Noot: bij deze tutorial wordt verwacht dat de cursist al enige kennis heeft van STEP7 en de vorige tutorials. Een aantal handelingen wordt niet meer expliciet uitgelegd. Houdt daar rekening mee.

8.1 Aanmaken Data Block

Open één van de genoemde projecten en maak een nieuw data block aan in de Simatic Manager. Zie figuur 8-1.

Kies als naam DB1. Eventueel kan een sybolische naam gekozen worden. Zie figuur 8-2.



Figuur 8-1

Properties - Data Block		X
General - Part 1 General	- Part 2 Calls Attributes	1
Name and type:	DB1 Shared DB	•
<u>S</u> ymbolic Name:		
Symbol Comment:		
Created in <u>L</u> anguage:	DB	
Project path:		
Storage location of project:	D:\PROJECTS\STEP7\tut_wcc_	
	Code	Interface
Date created: Last modified:	11/05/2008 08:06:18 PM 11/05/2008 08:06:18 PM	11/05/2008 08:06:18 PM
C <u>o</u> mment:		×
		Cancel Help

Figuur 8-2

Bevestig met **OK** het aanmaken van het data block. Er wordt nu een venster geopend met daarin één declaratie (dit is de *initial view* van het data block). Zie figuur 8-3.

KAD/STL/FBD - DB1						_ 🗆 🗵	
<u>File Edit Insert PLC D</u> ebug	\underline{V} iew $\underline{0}$ ptions	<u>W</u> indow <u>H</u> elp					
🗋 🗁 🔓 🔚 🎒 🐰 🖻	6 0 0 G	67 🛍 🖂 🗣 6	r !« »! 🔲 🖪	▶?			
DB1 tut_wcc_flex\SIMATIC 300(1)\CPU 315F-2 PN/DP							
+- 👭 Libraries	Address	Name	Туре	Initial value	Comment		
	0.0		STRUCT				
	+0.0	DB_VAR	INT	0	Temporary placeholder variable		
	=2.0		END_STRUCT				
						•	
Program							
Press F1 to get Help.					© offline Abs < 5.2 Insert		

Figuur 8-3

Vul de eerste regel over de initial view heen. Zie figuur 8-4.

DB1 tut_wcc_flex\SIMATIC 300(1)\CPU 315F-2 PN/DP							
Address	Name	Туре	Initial value	Comment			
0.0		STRUCT					
+0.0	temperatuur	array[110]	0.0	De laatste tien temperatuur monsters			
=2.0		END_STRUCT					
					F		



Na een druk op de return-toets verschijnt onder array[1..10] een leeg wit vakje. Vul daar het woord REAL in en sluit af met een druk op de return-toets. Het geheel wordt nu door de editor aangepast in de juiste vorm: de keywords worden met hoofdletters geschreven en de initiële waarde van reals wordt in wetenschappelijke notatie omgezet. Zie figuur 8-5.

DB1 tut_wcc_flex\SIMATIC 300(1)\CPU 315F-2 PN/DP						
Address	Name	Туре	Initial value	Comment		
*0.0		STRUCT				
+0.0	temperatuur	ARRAY[110]	0.000000e+000	De laatste tien temperatuur monsters		
*4.0		REAL				
=40.0		END_STRUCT				
•					•	

Figuur 8-5

Voeg er nu nog een integer bij met de naam index. Let op dat deze als initiële waarde 1 moet hebben. Zie figuur 8-6.

■ DB1 "I	DB1 "DB1" tut_wcc_flex\SIMATIC 300(1)\CPU 315F-2 PN/DP\\DB1					
Address	Name	Туре	Initial value	Comment		
0.0		STRUCT				
+0.0	temperatuur	ARRAY[110]	0.000000e+000	De laatste tien temperatuur monsters		
*4.0		REAL				
+40.0	index	INT	1	Waar komt een nieuwe meetwaarde?		
=42.0		END_STRUCT				

Figuur 8-6

Sluit de editor af door middel van **File**→**Save**.

8.2 Een stuk je SCL-code

Nu moet er natuurlijk iets met dat data block gedaan worden. Er volgt nu een stuk je SCL-code dat twee onderdelen bevat. Het eerste deel zorgt ervoor dat een gemeten temperatuurwaarde wordt opgeslagen in de array, het tweede onderdeel sommeert de tien opgeslagen waarden uit de array en deelt het door tien. Zo wordt het rekenkundig gemiddelde berekend.

Maak een SCL-broncodebestand aan met de naam RecordAndAvgTemp. Vul de SCL-code in zoals in opgegeven in figuur 8-7. Compileer de code en kijk onder Blocks of FC8 is aangemaakt. Zie bijlage 5 voor de volledige code.

```
FUNCTION FC8 : REAL
VAR INPUT
    record temp : BOOL;
    current temp : REAL;
END VAR
VAR_TEMP
    loop_index : INT;
    cumu_temp : REAL;
END VAR
BEGIN
    IF record temp THEN
        DB1.temperatuur[DB1.index] := current temp;
        DB1.index := DB1.index + 1;
        IF DB1.index > 10 THEN
           DB1.index := 1;
        END IF:
    END IF:
    cumu temp := 0.0;
    FOR loop index:= 1 TO 10 BY 1 DO
        cumu temp := cumu temp + DB1.temperatuur[loop index];
    END FOR:
    FC8 := cumu temp / 10.0;
END FUNCTION
```

Figuur 8-7

8.3 Symbolen, OB1 en downloaden

Het functieblok moet gestart worden vanuit OB1. Dit wordt gedaan door middel van een schakelaar op de simulatiemodule. De functie geeft een real-waarde terug. Die moet ergens worden opgeslagen. Vul de symbolentabel aan met nieuwe variabelen: twee booleans genaamd RECORDSWITCH en RECORDSWITCHEDGE en real genaamd TEMPAVG. Let er op dat de geheugenplaatsen niet overlappen met andere variabelen!

Vervolgens moet in OB1 nog de functie-aanroep geprogrammeerd worden. Pas OB1 aan zodat de functie wordt aangeroepen (waarom is de flankdetector nodig?). Zie figuur 8-8.



Figuur 8-8

Download nu de nieuwe en aangepaste blokken OB1, DB1 en FC8. Omdat de volgorde van downloaden niet vast staat, kan het zijn dat FC8 vóór DB1 geladen worden. Als nu de PLC in RUN staat én FC8 wordt direct na laden uitgevoerd (OB1 is dus net geladen), dan gaat de PLC in STOP omdat DB1 nog niet geladen is en FC8 een aanroep doet naar DB1. Zet daarom de PLC eerst in STOP. Na het downloaden natuurlijk de PLC weer in RUN zetten.

8.4 Monitoring

Nu kan de SCL-code getest worden. Hiervoor moet het data block geopend worden. Klik na openen op het bekende brilletje. Er volgt een waarschuwing. Klik op **Yes**. Zie figuur 8-9.



Figuur 8-9

Nu volgt de data view in *online mode* van DB1. Hierin zijn de actuele waarden van de diverse variabelen te zien. Merk op dat index op 2 staat en alle array-onderdelen een waarde (anders dan de initiële waarde 0.0) hebben. Zo te zien is er al meer dan tien keer een waarde opgeslagen. Hier wordt immers gebruik gemaakt van een circulaire buffer¹⁷. Zie figuur 8-10.

K LAD/STL/FBD - @DB1								
<u>File Edit Insert PLC Debug</u>	g ⊻ie	ew <u>O</u> ptions	<u>W</u> indow <u>H</u> elp					
□ 🗳 🖫 🗐 🦂 🕫 🖻 🗢 ⇔ 📴 📩 🗢 🗣 🔐 🔣 📜 🛄 👷								
Comparison of the second								
🕀 🦉 Libraries		Address	Name	Туре	Initial value	Actual value	Comment	
		0.0	temperatuur[1]	REAL	0.000000e+000	23.12653	De laatste tien temperatuur monsters	
		4.0	temperatuur[2]	REAL	0.000000e+000	32.53973		
		8.0	temperatuur[3]	REAL	0.000000e+000	29.49942		
		12.0	temperatuur[4]	REAL	0.000000e+000	28.68213		
		16.0	temperatuur[5]	REAL	0.000000e+000	27.49399		
		20.0	temperatuur[6]	REAL	0.000000e+000	26.72531		
		24.0	temperatuur[7]	REAL	0.000000e+000	25.97418		
		28.0	temperatuur[8]	REAL	0.000000e+000	25.23962		
		32.0	temperatuur[9]	REAL	0.000000e+000	24.87827		
€ <u><</u>		36.0	temperatuur[10]	REAL	0.000000e+000	24.16684		
		40.0	index	INT	1	2	Waar komt een nieuwe meetwaarde?	
Program E Call str.	Ľ							
Press F1 to get Help.							Abs < 5.2 Rd	



Natuurlijk moet ook nog de gemiddelde temperatuur gemonitord worden. Dit kan met via een VAT. Zie figuur 8-11.

¹⁷ <u>http://en.wikipedia.org/wiki/Circular_buffer</u>

8 (V	/ar - V	AT2					<u>_ D ×</u>
<u>T</u> able	e <u>E</u> dit	: <u>I</u> nse ∼Ins	ert P <u>L</u> C V <u>a</u> riable ⊻ ⊐ut est utesta	iew <u>O</u> ptions <u>W</u> in ≂I I I ≥∠I	dow <u>H</u> elp		
		<u></u>			* * *	<u>₩</u> 66° ₩	
66	VAT2	@	ut_wcc_flex\SIMA1	ric 300(1)\CPU 3	315F-2 PN/DP		
	C Ad	dress	Symbol	Display format	Status value	Modify value	
1	1	8.0	"RECORD_SWITCH"	BOOL	true		
2	М	18.0	"TEMPTOOHIGH"	BOOL	false		
3	М	18.1	"TEMPTOOLOW"	BOOL	false		
4	М	18.2	"RESETALARMS"	BOOL	false		
5	м	18.3	"ALARMSFF"	BOOL	📘 true		
6	М	18.4	"ALARMSDETEGDE"	BOOL	false		
7	MD	20	"TMEAS"	FLOATING_POINT	21.45813		
8	MD	24	"VMEAS"	FLOATING_POINT	4.351852		
9	MD	28	"RNTC"	FLOATING_POINT	11691.38		
10	MD	32	"TEMPAVG"	FLOATING_POINT	26.8326		
11	PM	/ 272	"ANALOG_IN"	HEX	VV#16#2F00		
12							
_	_	_					J
ut_wo	cc_flex	\SIMA	TIC 300(1)\\S7 Prog	am(1)		•	RUN //



8.5 Instance Data Block

In deze tutorial is gebruikt gemaakt van een *shared data block*. Het data block wordt echter alleen door FC8 gebruikt; geen ander programmablok werkt met de data uit DB1. Dan is het makkelijker om een Function Block (FB) te gebruiken met daaraan gekoppeld een *instance data block*. Zo kan de declaratie van de array en index gewoon binnen de FB gedaan worden; hiervoor moeten STAT-variabelen worden aangemaakt. Deze STAT-variabelen komen dan in de Instance DB terecht. Bij referentie naar de STAT-variabelen hoeft de DB-*qualifier* niet meer te worden opgegeven.

In bijlage 5 is de code van FB8 en DB8 te vinden. FB8 lijkt erg veel op FC8 uit figuur 8-7. FB's hebben echter geen teruggaveparameter maar moeten dat met een OUTPUT-parameter doen. De declaratie van de array en index wordt nu gedaan binnen de beschrijving van de FB. Het data block DB8 is leeg. Dat is ook de bedoeling, want de parameters van FB8 worden automatisch in DB8 geplaatst. Wel moet worden opgegeven dan DB8 bij FB8 hoort. Merk op dat de in- en outparameters ook in het data block aanwezig zijn. Zie figuur 8-12.

🔢 DB8	DB8 tut_wcc_flex\SIMATIC 300(1)\CPU 315F-2 PN/DP							
	Address	Declaration	Name	Туре	Initial value	Actual value	Comment	
1	0.0	in	record_temp	BOOL	FALSE	FALSE	Should we record?	
2	2.0	in	current_temp	REAL	0.000000e+000	0.000000e+000	The current temperature	
3	6.0	out	avg_temp	REAL	0.000000e+000	0.000000e+000	The average temp returns	
4	10.0	stat	temperature[1]	REAL	0.000000e+000	0.000000e+000	Special treat: 10 items of temperature initialized to 0.0	
5	14.0	stat	temperature[2]	REAL	0.000000e+000	0.000000e+000		
6	18.0	stat	temperature[3]	REAL	0.000000e+000	0.000000e+000		
7	22.0	stat	temperature[4]	REAL	0.000000e+000	0.000000e+000		
8	26.0	stat	temperature[5]	REAL	0.000000e+000	0.000000e+000		
9	30.0	stat	temperature[6]	REAL	0.000000e+000	0.000000e+000		
10	34.0	stat	temperature[7]	REAL	0.000000e+000	0.000000e+000		
11	38.0	stat	temperature[8]	REAL	0.000000e+000	0.000000e+000		
12	42.0	stat	temperature[9]	REAL	0.000000e+000	0.000000e+000		
13	46.0	stat	temperature[10]	REAL	0.000000e+000	0.000000e+000		
14	50.0	stat	index	INT	1	1	Index into the array, initialized to 1	
•							F	

Figuur 8-12

9. Tips, tricks & troubleshoot

In dit hoofdstuk worden wat zaken behandeld die niet direct in een tutorial thuishoren maar wel handig zijn om te weten. Zo wordt uitgelegd hoe je projecten kan archiveren en gearchiveerde projecten kan inlezen. Het veranderen van PC/PG interface is ook nog wel eens nodig als bijvoorbeeld met de Ethernet-verbinding moet worden gewerkt. De PLC kan in STOP gaan na een fout, maar wat is er nu gebeurd? Dat is te zien via diagnostiek.

9.1 Projecten archiveren

Via de Simatic Manager is het mogelijk om projecten te archiveren. Alle bestanden van het project worden dan door middel van compressie in één bestand geplaatst. Dit archief, wat in feite een gewoon ZIP-bestand is, kan dan worden opgeslagen op bijvoorbeeld een USB-stick. Noot: maak nooit zelf een archief aan, altijd via de Simatic Manager. Noot: er mogen géén spaties in map- en bestandsnaam voorkomen.

Sluit eerst het project af door middel van **File**→**Close**. Zie figuur 8-1.



Figuur 9-1

Kies daarna File→Archive. Zie figuur 9-2.



Figuur 9-2

Vervolgens wordt gevraagd om een project te selecteren. Selecteer het juiste project en klik op **OK**. Noot: het project moet afgesloten zijn! Zie figuur 9-3.

oser projects	Libraries Sample projects Multiprojects
Name	Storage path
🞒 ВМА	C:\Program Files\Siemens\Step7\S7Proj\Bma
🎒 efkes	C:\Program Files\Siemens\Step7\s7proj\efkes
🞒 read_temp	D:\PROJECTS\STEP7\read_tmp
🎒 testme	C:\Program Files\Siemens\Step7\s7proj\testme
🗗 testuur	C:\Program Files\Siemens\Sten7\s7proi\testyuz
	C. V Togram Files (Stemens (Step) Vst proj Veskyz
tutorial	D:\PROJECTS\STEP7\tutorial
tutorial	D:\PROJECTS\STEP7\tutorial
ser projects:	D:\PR0JECTS\STEP7\tutorial
ser projects:	D:\PROJECTS\STEP7\tutorial Selected 1
ser projects: braries:	D:\PROJECTS\STEP7\tutorial Selected 1
ser projects: braries: ample projects:	Selected Browse Browse

Figuur 9-3

Nu wordt er gevraagd wat de naam is van het nieuw te maken archief. Kies een naam die nog niet bestaat, anders wordt het oude archief zonder pardon overschreven. Zie figuur 9-4.

Archiving - S	elect an archive			?	×
Save jn: 🔀	STEP7	•	🗧 🗈 📛	* 🎟 -	
boor_she eerste_k Final graphtes looptest read_tmp	È SCL_BOOK È scl_test È sheets È STL_BOOK È traf_gr1 È traf_gra	traf_lad tut_grap tut_grex tut_scl tut_scl tutorial verkeers	Bla Loc Re. Rlo SCI Sle	at.zip pptest.zip ad_tmp.zip _ops.zip L_Book.zip ufdet.zip	
•					Þ
File <u>n</u> ame:	Tutorial.zip			<u>S</u> ave	
Save as <u>t</u> ype:	PKZip 12.0-Archive (*.:	zip)	•	Cancel	

Figuur 9-4

Dan wordt er gevraagd of er gearchiveerd wordt op diskettes (jawel!). Kies No. Zie figuur 9-4.

Archive - Options	×
Archive That Goes across Diskettes:	
No	•
Incremental Archiving	
🗖 Reset <u>A</u> rchive Bit	
Check Consistency	
OK Cancel He	lp

Figuur 9-5

Daarna wordt het project gearchiveerd in een nieuw geopende DOS-box. Hier is geen figuur van.

9.2 Projecten inlezen

Gearchiveerde projecten kunnen weer worden ingelezen. Daarna zijn ze te openen via de Simatic Manager.

Klik in de Simatic Manager op File-Retrieve. Zie figuur 9-6.



Figuur 9-6

Selecteer nu het juiste archief en klik op **Open**. Zie figuur 9-7.



Figuur 9-7

Nu moet worden opgegeven waar de bestanden van het archief worden geplaatst. Zoek een geschikte directory en klik op **OK**. Zie figuur 9-8.



Figuur 9-8

Het archief wordt geopend en de bestanden worden ingelezen. Dit is te zien in een aparte DOSbox die gestart wordt (geen figuur). Nadat het archief helemaal geplaatst is volgt een melding. Klik op **OK**. Zie figuur 9-9.

Retrieve	(3020:58)	×
į)	The retrieved data have been stored in the project directory 'D:\PR0JECTS\STEP7\effe'.	
🗖 Don	ot display this <u>m</u> essage again.	
10		

figuur 9-9

Dan wordt gevraagd of het project gelijk geopend moet worden. Klik op **Yes** om te openen. Zie figuur 9-10.





Het project wordt geopend. Zie als voorbeeld figuur 9-11.



Figuur 9-11

9.3 PG/PC interface

De PG/PC¹⁸ interface wordt voor diverse doeleinden gebruikt. Het is het logische koppelstuk tussen de PG/PC en de PLC- en HMI-hardware. De PLC kan er mee worden geconfigureerd en geprogrammeerd, en kan er mee worden gevolgd door middel van een *online* verbinding. HMI-stations kunnen er mee worden geprogrammeerd.

Hardwarematig zijn er aan de PG/PC-kant verschillende mogelijkheden. Er kan gekozen worden voor zogenaamde CP-kaarten, communicatiekaarten van Siemens. Denk hierbij aan de CP5511 en CP5611. Het kan een seriële interface zijn, of een Ethernet-aansluiting. In het lab zijn ook USB-adapters beschikbaar.

Afhankelijk van de PLC en communicatiemogelijkheden moet de juiste interface worden ingesteld. Hieronder volgt een korte tutorial over het selecteren van een nieuwe interface.

9.3.1 PC/PG interface via de USB adapter

Kies in de Simatic Manager Options→Set PG/PC Interface. Zie figuur 9-12.





¹⁸ De term PG staat voor Programmier Gerät. De term PG wordt eigenlijk gebruikt om de PC/laptop/computer aan te geven waarmee de PLC geconfigureerd en geprogrammeerd wordt.

Er wordt een venster geopend waarin de huidige geselecteerde interface te zien is. In dit geval is het de PC Adapter (USB) in Auto modus (figuur 9-13). De adapter zoekt dan zelf uit of er een verbinding kan worden met de PLC kan worden gelegd via MPI, Profibus of PtP. Gebruik deze modus liever niet. Het kan problemen opleveren met het online volgen van variabelen en programmacode.

Set PG/PC Interface	×
Access Path LLDP	
Access Point of the Application: S70NLINE (STEP 7)> PC Adapter(MI (Standard for STEP 7)	PI) 🔽
Interface <u>P</u> arameter Assignment Used: PC Adapter(MPI)	P <u>r</u> operties
C Adapter(Auto) C Adapter(MPI) C Adapter(PPI) C Adapter(PROFIBUS)	Copy Dejete
[Parameter assignment of your PC adapter for an MPI network]	
Add/Remove:	Sele <u>c</u> t
	Cancel Help

Figuur 9-13

Als de adapter niet werkt, kan het zijn dat een verkeerde poort is ingesteld. Klik dan op **Properties**. Er opent nu een nieuw venster (figuur 9-14). De parameters moeten precies als in de figuur ingesteld zijn.

Properties - PC Adapter(Auto)		×
Automatic Bus Profile Detection Local Connect	tion	
Station Parameters		
Address:	0	3
Iimeout:	30 s	-
Network Parameters		
Start Network Detection		
<u>OK</u>	Cancel	Help

Figuur 9-14

Onder het tabblad Local Connection kan de huidige poort gewijzigd worden. Deze moet als USB ingesteld staan. Zie figuur 9-15.

Properties - PC Adapter(Auto)				×
Automatic Bus Profile Detection Loc	cal Connect	ion		
Connection to:	USB	•		
Apply settings for all modules			_	
OK		Cancel	Help	

Figuur 9-15

9.3.2 PC/PG interface omzetten

De PLC in het lab zijn geschikt om via Ethernet te communiceren. Dat heeft als voordeel dat de netwerkcomponenten die al beschikbaar zijn zoals routers en switches, gebruikt kunnen worden. De PLC is configuren, programmeren en monitoren via Ethernet.

Kies in de Simatic Manager **Options**→**Set PG/PC Interface**. Zie figuur 9-12. Selecteer nu de TCP/IP interface van de Ethernetkaart waarmee de PLC bereikt kan worden (figuur 9-16).

Set PG/PC Interface	×
Access Path LLDP	
Access Point of the Application: S70NLINE (STEP 7)> TCP/IP -> Rea (Standard for STEP 7)	altek RTL8139 Family💌
Interface Parameter Assignment Used:	Properties
TLP/IP -> Realtek RTL8T39 Family	
CP/IP → Intel(R) 82567LM-3 Gig. TCP/IP → NdisWanIp TCP/IP → Realtek RTL8139 Family TCP/IP(Auto) → Intel(R) 82567LM- (Assigning Parameters to Your NDIS CPs with TCP/IP Protocol (RFC-1006))	Liagnostics
Interfaces Add/Remove:	Sele <u>c</u> t
OK]	Cancel Help

Figuur 9-16

Noot: omdat er veel verschillende Ethernetkaarten bestaan, zal figuur 9-16 enigzins afwijken van de werkelijkheid.

Klik nu op **OK**. Er volgt een dialoog, zie figuur 9-17. Klik op **OK** om de Ethernet-interface te gebruiken.

Warning	×
2	The following access path(s) was (were) changed:
~	S7ONLINE (STEP 7) => TCP/IP -> Realtek RTL8139 Family
	OK Cancel Help

Figuur 9-17

9.4 PLC diagnostiek

Als de PLC onverwacht in STOP gaat, is het meestal goed fout. Maar waarom is de PLC in STOP gegaan? Er zijn veel verschillende problemen mogelijk. Een BCD-conversie kan zijn mislukt, een *I/O-slave* kan offline gegaan zijn, er kan een programmeerfout zijn gemaakt.

De PLC registreert al deze gebeurtenissen. Ze kunnen worden opgevraagd via het diagnostische venster. Selecteer in de Simatic Manager eerst de PLC. Klik nu op

PLC→**Diagnostic/Setting**→**Module Information**. De diagnostics venster wordt geopend op tabblad General. Selecteer tabblad Diagnostic Buffer. In figuur 9-18 is te zien dat de PLC in STOP ging omdat een Profinet-module verwijderd werd.

(Module	e Informa	ation - CP	U 315F-2 PN/	'DP)		
Path: tut	_profinet\{	STAT_PRO)F\CPU 315F-2 F	PN/ Oper	ating mode of the CPU	J: (STOP
Status: 😿	Error			Not a	a force job	
Perf	formance [Data	Commur	nication	Stacks	Identification
Gener	al	Diagnosti	c Buffer	Memory	Scan Cycle Time	Time System
<u>E</u> vents:		🗖 Eilte	r settings active	, r Tir	me including CPU/local	I time difference
No.	Time of d	ay	Date	Event		▲
1	09:03:18.	639 PM	11/06/2008	STOP cause	d by removing/inserting	g module (OB not 💻
2	09:03:18. 09:02:38.	635 PM 427 PM	11/06/2008 11/06/2008	PROFINET I Mode transiti	O module removed/car on from STARTUP to F	nnot be addressed RUN
4	09:02:38.	423 PM	11/06/2008	Request for r	manual warm restart	
5	09:02:38.	341 PM	11/06/2008	Mode transiti	on from STOP to STAF	RTUP
6	09:01:59.	062 PM	11/06/2008	PROFINET I	0: station return	
7	09:00:04.	237 PM	11/06/2008	STOP cause	d by stop switch being	activated
8	08:43:56.	007 PM	11/06/2008	Mode transiti	on from STARTUP to F	RUN
<u>D</u> etails o	in Eivent:	2 of 187	,		Event ID:	: 16# 3951
PROFIN station r IO syste Input ad no relev Request	IET IO mo number: m ID: 100 Idress: E ance for u ted OB: Re	dule remov 1 Ser (Z3): 81 emove/inse	ed/cannot be ac 01 ert OB (OB83)	ldressed		▲
Sav	/e <u>A</u> s	<u>s</u>	ettings	Open <u>B</u> loc	sk	Help <u>o</u> n Event
Close		<u>U</u> pdate	Print			Help

Figuur 9-18

9.5 Online blokken in de PLC verwijderen

Het is mogelijk om de blokken die ooit naar de PLC zijn verstuurd (download) te verwijderen. Deze blokken staan namelijk niet echt in de PLC maar op de MMC-kaart. Als de PLC in RUN gaat, worden de benodigde blokken in het werkgeheugen van de PLC geladen. Nu kan het zijn dat er zoveel blokken op de MMC-kaart staan, dat nieuwe blokken niet verstuurd kunnen worden. Via de *online block view* kunnen (niet benodigde) blokken verwijderd worden.

Selecteer in de Simatic Manager de map Blocks. Klik dan op **View**→**Online** of klik op het Online-pictogram (figuur 9-19).



Figuur 9-19

Er opent zich een venster waarin de online blokken te zien zijn. Selecteer alle blokken, behalve de System Data, SFC's en SDB's en verwijder ze. Zie figuur 9-20 en 9-21.

tutorial D:\PROJECTS\ST	EP7\tutorial_ONL	INE)			
🖃 🎒 tutorial	🚵 System data	📾 <mark>081</mark>	FC2	🕞 SFBO	
🗄 🔛 SIMATIC 300(1)	🕞 SFB1	🕞 SFB2	🕞 SFB3	🖽 SFB4	
📄 📆 CPU 315F-2 PN/DP	🕞 SFB5	🕞 SFB32	🕞 SFB52	🕞 SFB53	
⊡ ⊡ S7 Program(4)	ҧ SFB54	🖽 SFB75	🖽 SFB81	🖽 SFC0	
Blocks	SFC1	🖽 SFC2	🖽 SFC3	🖽 SFC4	
	🕞 SFC5	🖽 SFC6	🖽 SFC7	SFC11	
	SFC12	🖽 SFC13	🖽 SFC14	🕞 SFC15	
	SFC17	🖽 SFC18	🖽 SFC19	🕞 SFC20	
	SFC21	🖽 SFC22	🚍 SFC23	🕞 SFC24	
	SFC28	🖽 SFC29	🚍 SFC30	🕞 SFC31	
	SFC32	🖽 SFC33	🖽 SFC34	🕞 SFC36	
	🖽 SFC37	🖽 SFC38	👝 SFC39	🕞 SFC40	
	🖽 SFC41	🖽 SFC42	🖽 SFC43	🕞 SFC44	
	🖽 SFC46	🖽 SFC47	🖽 SFC49	🕞 SFC50	
	SFC51	🖽 SFC52	🚍 SFC55	🖽 SFC56	
	🕞 SFC57	🖽 SFC58	🚍 SFC59	🚌 SFC64	
	🖽 SFC65	🖽 SFC66	🖽 SFC67	🗊 SFC68	
	🕞 SFC69	🖽 SFC70	🖽 SFC71	🕞 SFC72	
	🗗 SFC73	🖽 SFC74	🖽 SFC81	🖽 SFC82	
	🕞 SFC83	🖽 SFC84	🖽 SFC101	🕞 SFC102	
	SFC103	🗊 SFC105	🚍 SFC106	SFC107	
	SFC108	🗊 SFC109	🖽 SFC112	🖽 SFC113	
	SFC114	🖽 SFC126	🚍 SFC127		

Figuur 9-20



Figuur 9-21

9.6 Clock memory byte

In de PLC is het mogelijk om één byte te reserveren als *clock memory*. De PLC genereert in elke bit een kloksignaal van verschillende frequenties. Alle kloksignalen hebben een duty cycle van 50%. Het gebruik van de clock memory wordt tijdens de configuratie opgegeven.

Start HWconfig en selecteer de CPU-module. Dubbelklik om de eigenschappenvenster te openen. Selecteer daarna het tabblad Cycle/Clock Memory. Vink de optie Clock memory aan en pas eventeel het adres aan. Standaard wordt MB0 gebruikt. Zie figuur 9-22.



Figuur 9-22

Hieronder staat een tabel met alle frequenties en periodetijden. Om bijvoorbeeld een knipperlamp voor alarmering te maken met een frequentie van 1 Hz moet bit M0.5 gebruikt worden als MB0 het clock memory byte is.

Bit	7	6	5	4	3	2	1	0
Freqentie (Hz)	0,5	0,625	1,0	1,25	2,0	2,5	5,0	10,0
Periode (s)	2,0	1,6	1,0	0,8	0,5	0,4	0,2	0,1

Tabel 9-1

10. Bijlagen

Bijlage 1 – LAD-programma verkeerslichtsysteem

SIMATIC

tutorial\ SIMATIC 300(1)\CPU 315F-2 PN/DP\...\FC1 - <offline>

11/07/2008 07:34:55 PM

 FC1 - <offline>

 ""

 Name:
 Family:

 Author:
 BRW

 Version:
 1.0

 Block version:
 2

 Time stamp Code:
 11/07/2008 07:31:36 PM

Tıme	stamp	Code:	11/	0772008	07:31	:36	ЪМ
		Interface:	12/	27/2007	10:34	:55	AM
Lengt	hs (b	olock/logic/da	ata):	00220	00118	000	00

Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC1 Ve:	- 3lock: FC1 Verkeerslichtsysteem			
Zeer eenvoudig voetoversteekplaats met verkeerslichten.				
	 > > auto			
voet	+	oo o +		
dinger o oo d gr r ro u oo k ed 1 n		<pre>> o rood o geel o groen </pre>		
I8.0 = drukknop 1 I8.1 = druknkop 2 I8.7 = RESET (bv. power-on reset)				
Q8.0 = voetganger ziet zijn aanvraag beantwoord Q8.2 = auto rood Q8.3 = auto geel Q8.4 = auto groen Q8.6 = voet rood Q8.7 = voet groen				
M8.0 = geheu M8.1 = tijdn M8.2 = tijdn M8.3 = geheu	.0 = geheugen voetganger_wil_oversteken .1 = tijdmeting geel is klaar (T0) .2 = tijdmeting groen is klaar (T1) .3 = geheugen flank drukknop 1 en 2			

Page 1 of 3

tutorial\ 11/07/2008 (SIMATIC 300(1)\CPU 315F-2 PN/DP\...\FC1 - <offline>

Network: 1 Voetganger gezien Wil een voetganger oversteken? Vang de actie in. Flankdetectie is nodig omdat voet gezien naar 1 gaat als RESET van 1 naar 0 gaat en SET-actie nog steeds 1. Nu moet SET-actie eerst 0 worden.



Network: 2 Timer voor auto geel Eerste tijdmeting. Meet de tijd dat het gele autoverkeerslicht actief is en houdt voetgangerslicht op rood.



Network: 3 Timer voor auto groen

Tweede tijdmeting. Meet de tijd dat het auto verkeerslicht rood is en het voetgangerslicht op groen. Hier zit een gevaarlijke constructie in. Timer_groen stuurt timer_groen_klaar aan als het klaar is, maar reset zichzelf dan ook na 1 scan cycle. Netter zou zijn om een extra timer te nemen, die bv. 100 ms te laten draaien en die de andere timers te laten resetten.



Page 2 of 3

Network: 4	Verkeerslicht auto groen
Is voet_gezien rood. Wederzijd	gelijk aan 0? Dan hebben we autolicht groen en voetgangerslicht s uitgesloten met volgend netwerk.
"voet_gezien"	"auto_groen" (S)
	"voet rood"
	(s)
	"auto_rood"
	"auto_geel"

Network: 5	Verkeerslicht auto geel
Is voet_gezien (rood. Wederzijds	gelijk aan 1? Dan hebben we autolicht geel en voetgangerslicht s uitgesloten met vorig netwerk.



Network: 6 Verkeerslicht auto rood	
------------------------------------	--

Hebben we timer_geel_klaar is 1? Dan hebben we autolicht rood en voetgangerlicht groen. Als timer_geel_klaar is 1, dan worden de acties van de vorige twee netwerken ongedaan gemaakt. Dat komt omdat de PLC z'n outputs pas aan het eind van de huidige scan cycle aanpast (bij Siemens eigenlijk aan het begin, maar goed).

"voet_groen" ____(R)____|



Bijlage 2 – Graph programma

SIMATIC

tut_graph_extended\SIMATIC_300_TRAF_EXT\
CPU 315F-2 PN/DP\...\FB2, DB2 - <Offline>

01/06/2008 12:26:57 PM



Page 1 of 2



01/06/2008 12:26:58 PM

Page 2 of 2

Bijlage 3 – PLC Blocks, input- en outputadressen, memory

De PLC beschikt over een aantal blocks, waarin programma en data gestructureerd kan worden.

Organization Blocks (OB)

OB's zijn de interfaces tussen het OS en het gebruikersprogramma. De belangrijke OB is OB1: *cyclic operation block*, voer het programma "steeds" uit. Daarnaast zijn er tal van andere OB's.

Dus: om een programma draaiend te krijgen moet ALTIJD OB1 gebruikt worden, ook al doet deze niets anders dan FB's of FC's aanroepen

Function Blocks (FB)

Dit zijn blokken met parameteroverdrachtmogelijkheden en eigen geheugen. Dit geheugen voor dataopslag ligt in een DB en wordt persistent genoemd, dat wil zeggen dat een FB zijn eigen data kan bewaren over aanroepen heen. Voor C-programmeurs is het te vergelijken met static variabelen.

Functions (FC)

Dit zijn blokken die ook parameteroverdrachtmogelijkheden hebben, maar geen eigen geheugen. Nadat een functie klaar is, wordt alle data gewist. Typische voorbeelden zijn het berekenen van de wortel van een getal, grootste gemene deler. Functions kunnen waarden retourneren.

Data Blocks (DB)

Dit zijn de blokken waarin de data wordt opgeslagen. Ze kunnen maximaal 16420 bytes groot zijn.

System Function (SFC), System Function Blocks (SFB)

Deze zijn als FC en FB, maar nu met voorgeprogrammeerde functies, bijvoorbeeld communicatie over de bus.

Input- en outputadressen

De digitale ingangen lopen van I0.0 t/m I1.7. Hierin zijn 0 en 1 het input adresbyte en .0 en .7 het bitnummer binnen een adres. Er zijn in totaal 16 ingangen.

De digitale uitgangen lopen van Q0.0 t/m Q1.7. Hierin zijn 0 en 1 het output adresbyte en .0 en .7 het bitnummer binnen een adres. Er zijn in totaal 16 uitgangen.

De analoge ingangen zijn PIW272 t/m PIW278. De uitgangen zijn te vinden op PQW272 en PQIW274. Merk op dat het om PI en PQ gaat (buiten de process image table om) en dat het words zijn. Bv. PIW278 bestaat uit byte 278 en 279.

De ingangen van de simulatiemodule lopen van I8.0 t/m I8.7, maar dit kan veranderd worden. De uitgangen lopen van Q8.0 t/m Q8.7

Memory

Het geheugen loopt van M0.0 tot M2047.7. Het geheugen is ook byte-, word- en long wordadresseerbaar. Het is een algemeen geheugen dat gedeeld wordt met andere OB's, FB's en FC's.

Bijlage 4 – Enige formules voor temperatuurmeting

Een NTC-weerstand heeft een weerstandswaarde die zeer goed benaderd wordt met de formule:

$$R_{\rm NTC} = A \cdot e^{\frac{B}{T}}$$
(1)

A en B zijn hierin parameters van de NTC.

De overdracht van de spanningsdeler uit figuur 6-1 is als volgt te berekenen

$$Z = \frac{V_{\rm NTC}}{V_{\rm POWER}} = \frac{R_{\rm NTC} //R_{\rm IN}}{R_{\rm S} + R_{\rm NTC} //R_{\rm IN}}$$
(2)

Hierin is V_{POWER} de spanning van de voeding, V_{NTC} de spanning over de NTC, R_{NTC} de weerstand van de NTC, R_{IN} de ingangsweerstand van de analoge ingang en R_S de serieweerstand.

Invullen en uitwerken levert de functie

$$Z = \frac{R_{IN} \cdot A \cdot e^{\frac{B}{T}}}{R_{S} \cdot R_{IN} + R_{S} \cdot A \cdot e^{\frac{B}{T}} + R_{IN} \cdot A \cdot e^{\frac{B}{T}}}$$
(3)

Hieruit is T te herleiden:

$$T = \frac{B}{\ln\left(-\frac{Z \cdot R_{s} \cdot R_{IN}}{A\left(-R_{IN} + Z \cdot R_{s} + Z \cdot R_{IN}\right)}\right)}$$
(T in K) (4)

V_{NTC} is echter ook gelijk aan:

$$V_{\rm NTC} = \frac{ADC_{\rm VALUE}}{ADC_{\rm VALUE_MAX}} \cdot V_{\rm ANALOG_MAX}$$
(5)

Hierin is ADC_{VALUE} de huidige waarde van de ADC, ADC_{VALUE_MAX} de ADC-waarde bij de maximale analoge ingangsspanning en V_{ANALOG_MAX} de maximale analoge ingangsspanning. Z is dan eenvoudig te bepalen uit het eerste deel van (2) en in te vullen in (4).

De volgende waarden voor diverse parameters zijn:

$V_{POWER} = 24,0 V$
$V_{ANALOG_MAX} = 10,0 V$
$ADC_{VALUE_MAX} = 27648$
$^{\circ}C = K - 273,15$

Bijlage 5 – SCL-code temperatuurconversie en min-max testen

FUNCTION FC3 : REAL // Function 3 TITLE = 'Convert ADC To Temp' VERSION: '1.0' AUTHOR: THR NAME: TCONV FAMILY: CONV 11 // This routine converts the raw temperature data from the self // made NTC temp sensor to the temp in degree C. 11 11 The transfer function for Vout is: +24 V 11 11 Vout = Z * 24 V 11 11 11 where Rs 11 Rntcp 11 Z = -----// Rs + Rntcp ---- Vout 11 -+-11 where 11 11 Rntcp = {A * exp (B/T)} // Rin // Rin Rntc with A and B being parameters of Rntc, 11 11 and Rin being the input resistance of 11 the analog input. // + 11 11 === 11 $\ensuremath{{//}}$ The tranfer function has to be rewritten to a function with // T explicit. Note that T is in Kelvin. // The raw ADC value has a maximum of C600 hex at 10.0 V and // a minimum of 0000 hex at 0.0 V. // Please note that with a high power supply voltage, the self // heating effect of the NTC is considerable. 11 CONST VPOWER := 24.0; // Power Supply VANALOG MAX := 10.0; // Max analog voltage ADCVALUE_AT_VANALOG_MAX := 27648; // ADC value code @ ... A := 0.0235664; // A parameter in Ohm B := 3863.65; // B parameter in K RS := 47000.0; // Series resistor of 47 KOhm // Input resistance of analog input
// Convert to Celsius RIN := 100000.0; K_TO_C := 273.15; END_CONST VAR_INPUT ADC_VALUE : WORD; // The input from the analog channel END_VAR VAR_OUTPUT VMEAS : REAL; // Voltage measured

RNTC : REAL; // Calculated resistance of NTC END_VAR VAR_TEMP // Conversions Z : REAL; ZI : INT; END_VAR BEGIN // ADC value to integer ZI := WORD_TO_INT(ADC_VALUE); // Voltage measured VMEAS := (1.0*ZI)/ADCVALUE_AT_VANALOG_MAX * VANALOG_MAX; // Calculate Z as function of ... Z := VMEAS/VPOWER; // Rntc in Ohms. Function found with the aid of Maple 10. RNTC := -(VMEAS*RS*RIN/(-VPOWER*RIN+VMEAS*RIN+VMEAS*RS)); // Temperature in degree C. Function found with the aid of Maple 10. // This is the return value of the function. FC3 := B / LN(-1.0*Z*RS*RIN / (A*(-RIN+Z*RS+Z*RIN))) - K TO C; END FUNCTION FUNCTION FC4 : VOID TITLE = 'Check Minimum and Maximum Temps' VERSION: '1.0' AUTHOR: BRW NAME: CMM FAMILY: CHK CONST C_TEMPTOOHIGH := 30.0; // Higher bound temperature C TEMPTOOLOW := 15.0;// Lower bound temperature END CONST VAR_INPUT TEMPERATURE : REAL; // The current temperature END_VAR VAR_OUTPUT TEMPTOOHIGH : BOOL; // Signals temperature too high TEMPTOOLOW : BOOL; // Signals temperature too high END_VAR BEGIN IF TEMPERATURE >= C TEMPTOOHIGH THEN // If current temp too high... TEMPTOOHIGH := true; // signal it. ELSE TEMPTOOHIGH := false; END_IF; IF TEMPERATURE <= C_TEMPTOOLOW THEN // If current temp too low... TEMPTOOLOW := true; // signal it. ELSE TEMPTOOLOW := false; END_IF; END_FUNCTION

```
FUNCTION FC8 : REAL
TITLE = 'Record temperature and calculate average'
VERSION: '1.0'
AUTHOR: BRW
NAME: RECCALC
FAMILY: TEMP
VAR INPUT
                                 // Should we record?
   current_temp : BOOL;
_VAR
                                    // The current temperature
END_VAR
VAR_TEMP
                                    // Used as loop index
    loop_index : INT;
    cumu_temp : REAL;
                                    // Cumulative temp so far
END_VAR
BEGIN
    // If we must record a new temperature
    IF record_temp THEN
        // Do record it
        DB1.temperatuur[DB1.index] := current_temp;
        // Advance the index
        DB1.index := DB1.index + 1;
        // But if it is past the end, flip it to the begin
        IF DB1.index > 10 THEN
            DB1.index := 1;
        END_IF;
    END IF;
    // Make sure we start at 0.0
    cumu_temp := 0.0;
    // Visit all array items and sum them
    FOR loop_index:= 1 TO 10 BY 1 DO
        cumu_temp := cumu_temp + DB1.temperatuur[loop_index];
    END_FOR;
    // Calculate average
    FC8 := cumu_temp / 10.0;
END_FUNCTION
```
```
FUNCTION_BLOCK FB8
TITLE = 'Record temperature and calculate average'
VERSION: '1.0'
AUTHOR: BRW
NAME: RECCALC
FAMILY: TEMP
    record_temp : BOOL; // Should we record?
current_temp : REAL; // The current
VAR
VAR INPUT
                                     // The current temperature
END_VAR
VAR_OUTPUT
                                     // The average temp returns
    avg_temp : REAL;
END_VAR
VAR
    temperature : ARRAY[1..10] OF
                    REAL := 10(0.0); // Special treat: 10 items of
temperature initialized to 0.0
    index : INT := 1;
                                      // Index into the array, init. to 1
END_VAR
VAR_TEMP
    loop_index : INT;
                                    // Used as loop index
                                     // Cumulative temp so far
    cumu_temp : REAL;
END_VAR
BEGIN
    // If we must record a new temperature
    IF record_temp THEN
        // Do record it
        temperature[index] := current_temp;
        // Advance the index
        index := index + 1;
        // But if it is past the end, flip it to the begin
        IF index > 10 THEN
            index := 1;
        END_IF;
    END IF;
    // Make sure we start at 0.0
    cumu_temp := 0.0;
    // Visit all array items and sum them
    FOR loop_index:= 1 TO 10 BY 1 DO
        cumu_temp := cumu_temp + temperature[loop_index];
    END_FOR;
    // Calculate average
    avg_temp := cumu_temp / 10.0;
END FUNCTION BLOCK
DATA_BLOCK DB8 FB8
// Empty data block
BEGIN
END_DATA_BLOCK
```

Bijlage 6 – referenties

Bij het samenstellen van deze tutorial is gebruik gemaakt van de volgende documentatie:

Simatic Manuals

- Programming with STEP 7, edition 03/2006, ordernummer A5E00706944-01
- S7-GRAPH V5.3 for S7-300/400: Programming Sequential Control Systems, edition 02/2004, ordernummer A5E00290656-01
- S7-GRAPH V5.3 for S7-300/400: Programming Sequential Control Systems Getting Started, edition 02/2004, ordernummer A5E00290666-01
- S7-SCL V5.3 for S7-300/400, Edition 01/2005, ordernummer A5E00324650-01
- Statement List (STL) for S7-300 and S7-400 Programming, edition 03/2003, ordernummer A5E00706960-01
- S7-300 Automation System: Module Data, edition 02/2007, ordernummer A5E00105505-05
- S7-300 CPU 31xC and CPU 31x, Technical Specifications, edition 12/2006, ordernummer A5E00105475-07
- Ladder Logic (LAD) for S7-300 and S7-400 Programming, edition 03/2006, ordernummer A5E00706949-01
- Configuring Hardware and Communication Connections STEP 7, edition 03/2006, ordernummer A5E00706939-01

Boeken

- Berger, Hans, Automating with STEP7 in STL and SCL, 3rd edition, ISBN 3-89578-243-2, ordernummer A19100-L531-B893-X-7600. Er is nu een vierde editie.
- Berger, Hans, Automating with SIMATIC, 3rd edition, ISBN 3895782769, ordernummer A19100-L531-B927-X-7600

Websites

- PLC open <u>http://www.plcopen.org/</u>
- Siemens Automation Website -<u>http://support.automation.siemens.com/WW/livelink.exe?func=cslib.csinfo&lang=en&siteid=</u> <u>cseus&aktprim=0&extranet=standard&viewreg=WW&objid=10805055&treeLang=en</u>

Componenten

• Betatherm NTC - <u>http://nl.farnell.com/betatherm/10k3a542i/thermistor-ntc/dp/9707271</u> - let op: dit is een vervanger voor de originele NTC, met iets andere parameters.