

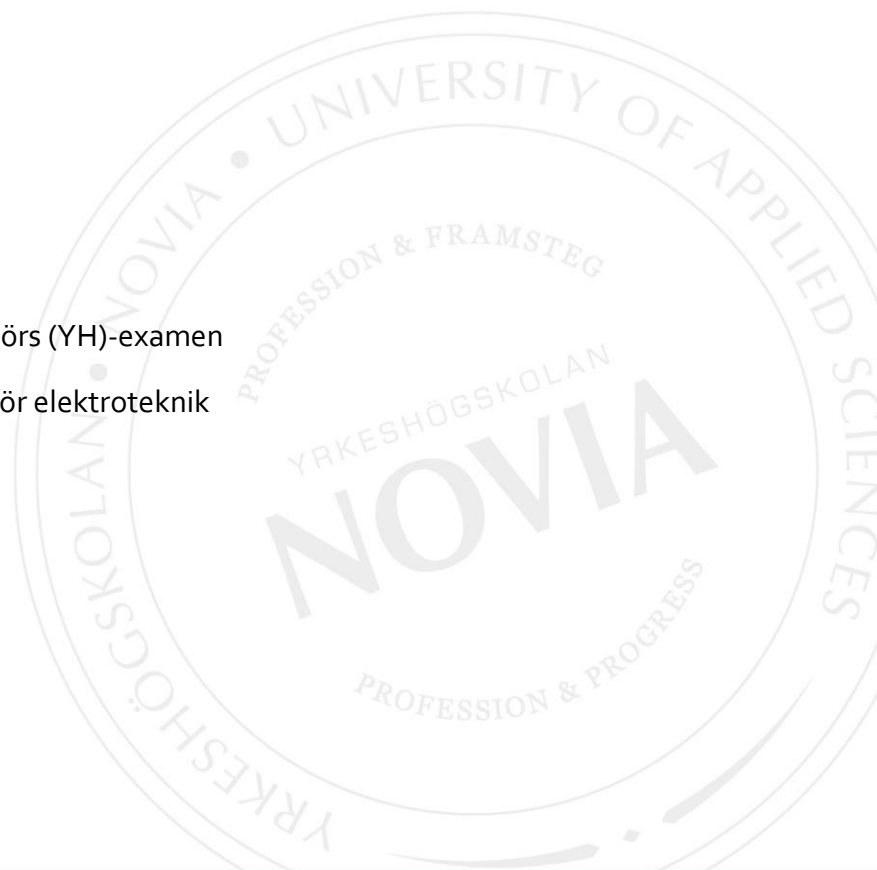
# Implementering av handmanöverdon till en PLC

David Henriksson

Examensarbete för ingenjör (YH)-examen

Utbildningsprogrammet för elektroteknik

Vasa 2019



## EXAMENSARBETE

Författare: David Henriksson  
Utbildning och ort: Elektroteknik, Vasa  
Inriktningsalternativ: Automationsteknik  
Handledare: Erik Englund

Titel: Implementering av handmanöverdon till en PLC

---

Datum: 20.4.2019      Sidantal: 36      Bilagor: 2

---

### Abstrakt

Detta examensarbete behandlar undersökningen hur kommunikationen mellan ett handmanöverdon och en automatisk truck ser ut för att sedan anpassa handmanöverdonet till en Siemens PLC.

Arbetets syfte var att reda ut hur kommunikationen mellan handmanöverdonet MCD och datorn CVC600 i en automatisk truck ser ut och baserat på detta ta fram de programblock som behövs för att kommunicera med enheten på ett enkelt sätt. Programmet skall även avkoda funktionssignalerna från handmanöverdonet och aktivera önskad utgång på en Siemens PLC.

I examenarbetets teoridel behandlas data- och seriell kommunikation och även den maskinvara och programvara som använts i arbetet. I arbetets praktiska del behandlas hur man har gått till väga för att utreda kommunikationen mellan enheterna, hur man genomför ett projekt och var man utför själva programmeringen i Siemens TIA Portal.

Resultatet av arbetet är att man kommit fram till hur kommunikationen ser ut och utifrån detta skapat ett funktionsblock i TIA Portal som kommunicerar med handmanöverdonet och utför en del av de funktioner som kan göras med handmanöverdonet MCD.

---

Språk: svenska      Nyckelord: kommunikation, PLC, handmanöverdon

---

## BACHELOR'S THESIS

Author: David Henriksson  
Degree Programme: Electrical Engineering  
Specialization: Automation Engineering  
Supervisor: Erik Englund

Title: Implementation of Manual Control Device to a PLC

---

Date April 20, 2019      Number of pages: 36      Appendices: 2

---

### Abstract

This bachelor's thesis deals with the examination of how the communication between a manual control device and an automatic guided vehicle looks like and adapt this to a Siemens PLC.

The purpose of the thesis is to figure out how the communication between a manual control device and a compact vehicle controller looks like and on the fundamentals of this create program blocks that are essential to communicate with the unit. The program also needs to decode the function signals from the manual control device and activate desired outputs on the PLC.

The theoretical part of the thesis comprehends data- and serial communication and even hard- and software that are used in the thesis. The practical part of the thesis comprehends the dealing of communication between the two units, how to create a project and where to perform the programming inside Siemens TIA Portal.

The result of the thesis is the knowledge of how the communication works and due to that, the creation of a function block inside TIA Portal that communicates with the MCD and performs a part of the functions that the MCD executes.

---

Language: Swedish      Key words: Communication, PLC, Manual Control Device

---

## Innehållsförteckning

1	Inledning.....	1
1.1	Uppdraget.....	1
1.2	Solving.....	1
2	Kommunikation .....	2
2.1	Datakommunikation .....	2
2.2	Datakommunikationsregler och protokoll.....	3
2.3	Protokollfunktioner .....	4
2.4	Modeller för datakommunikation och protokoll.....	4
2.4.1	Open System Interconnection, OSI – modellen .....	5
2.5	Seriell kommunikation .....	7
2.5.1	Seriella kommunikation standarder .....	8
2.6	Topologier i nätverk.....	10
2.6.1	Busstopologi.....	10
2.6.2	Stjärntopologi .....	10
3	Maskinvara.....	11
3.1	PLC.....	11
3.1.1	Simatic S7-1200 PLC.....	11
3.2	CM 1241 .....	12
3.3	Manual Control Device.....	13
3.4	USB-COM485-PLUS1.....	15
4	Programvara.....	16
4.1	MU Terminal.....	16
4.2	Serial Port Monitor .....	16
4.3	TIA Portal.....	17
4.3.1	OB.....	18
4.3.2	FB .....	18
4.3.3	FC.....	19
4.3.4	DB.....	20
5	Utförande .....	21
5.1	Undersökning av maskinvara och programvara .....	21
5.1.1	Loggning vid Solving.....	22
5.2	Testsystemets uppbyggnad .....	24
6	TIA Portal.....	25
6.1.1	Skapa ett projekt och anpassning av maskinvara .....	25
7	TIA Portal-programmering.....	28

8	Resultat.....	33
9	Diskussion .....	35
10	Litteraturförteckning.....	37
	Bilagor.....	39

## **Ordförklaring**

AGV	Automated Guided Vehicle
MCD	Manual Control Device
CM	Communication Module
CB	Communication Board
PLC	Programmable Logic Controller
CVC	Compact Vehicle Controller
TIA	Totally Integrated Automation
I/O	In och utgångar
OB	Organization block
FB	Function block
FC	Function
DB	Data block
FBD	Function block diagram
LAD	Ladder
SCL	Structured Control Language
SPM	Serial Port Monitor
HMI	Human Machine Interface

# 1 Inledning

Detta examensarbete är gjort åt Ab Solving Oy. Det handlar om en implementering av handmanöverdonet MCD på en PLC. Arbetet består av en teoretisk del om kommunikation, maskinvara och programvara. I den utförande delen behandlas tillvägagångs sättet för att lyssna på kommunikationen mellan ett handmanöverdon och en dator i en automatisk truck samt användningen av Siemens programvara TIA Portal.

## 1.1 Uppdraget

Handmanöverdonet MCD (MCD = **M**anual **C**ontrol **D**evice) Används normalt för att manuellt styra förarlösa truckar (AGV = **A**utomated **G**uided **V**ehicle) när behovet för detta finns. Samma handtag skulle kunna användas på en Siemens PLC-baserad vagn. Genom att använda samma manöverdon skulle både folket på Solving och vana AGV-användare känna igen sig.

Handmanöverdonet kommunicerar med yttervärlden genom ett RS485 gränssnitt. Siemens har kommunikationsmoduler, CM1241 eller CB 1241 som kan användas med ett RS485 gränssnitt.

Uppdraget från Solving är att klargöra hur kommunikationen mellan MCD och datorn i en automatisk truck CVC600 från Kollmorgen ser ut och baserat på detta ta fram de programblock som behövs för att anpassa MCD till en PLC

## 1.2 Solving

Ab Solving Oy är ett företag i Sandsund, Finland som grundades 1977. Sedan starten för 40 år sedan har de specialiserat sig på design och tillverkning av utrustningar för tunga lasthanteringar. Deras produktsortiment behandlar allt från manuellt enkla styrda utrustningar till automatiskt sofistikerade hanteringssystem där både hjul och luftkuddar används.

Solving är idag ledande inom området och har åtagit sig att skapa olika sorters hanteringssystem för komplexa och utmanande applikationer i industrier världen över. Deras produkter finns idag i över 20 olika slags industrier i över 50 länder. Deras materialhanteringssystem kan i princip nästan anpassas till den last som behövs oberoende

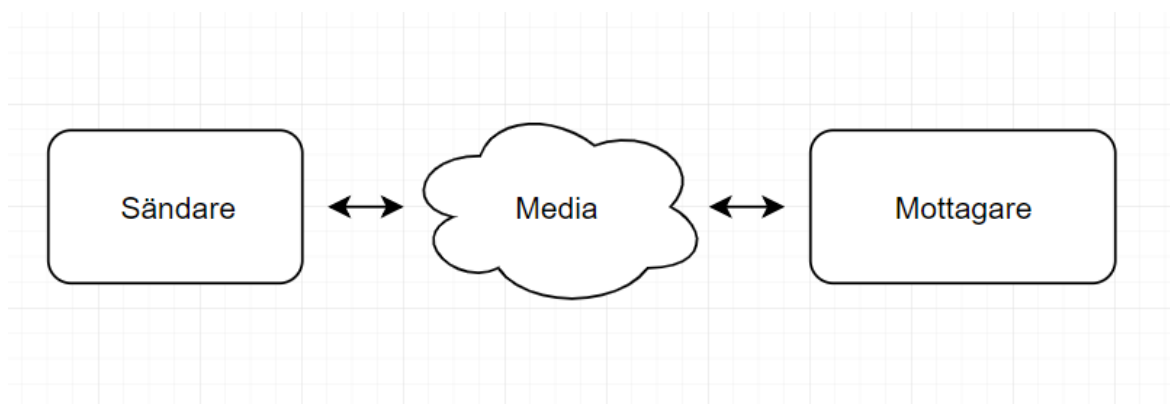
av storlek, vikt eller hanteringsfrekvens. Deras olika produkter består av automatiska truckar, hjultruckar, luftkuddetruckar och flyttsystem. (Solving, 2019)

## 2 Kommunikation

Detta kapitel behandlar teori om datakommunikation, seriell kommunikation med tillhörande delar och information om detta till exempel protokoll, standarder och topologier.

### 2.1 Datakommunikation

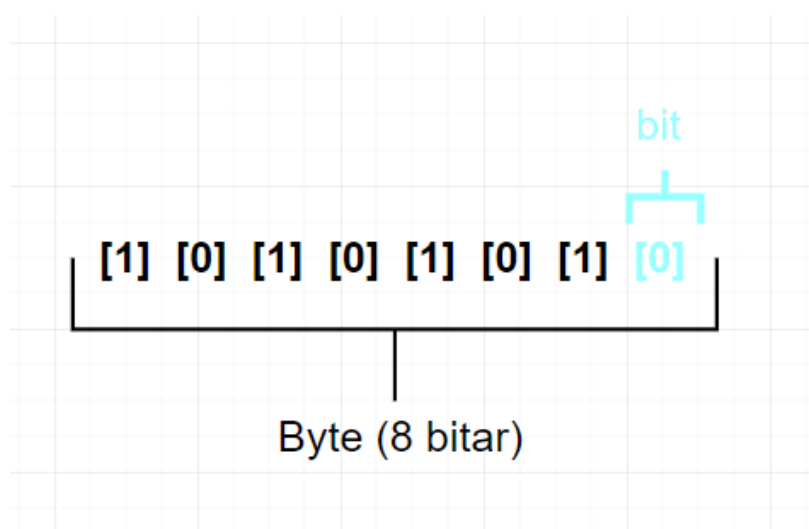
Datakommunikationens system har tre primära komponenter; sändare, medium och mottagare. Sändaren är vanligen den källa där data uppstår, mediet är medlet som används för överföringen och mottagaren är överföringens ända. Dessa enheter kan symboliseras genom att sändaren är en server på internet, mediet är bredbandsförbindelsen och datorn i hemmet är mottagaren. I dagens kommunikationssystem kan mottagaren och sändaren byta roller med varandra så att man kan både sända och ta emot. (Sen, 2014) (Berti, et al., 2012)



**Figur 1. Kommunikationssystemets komponenter.**

Datakommunikation är i dagsläget mestadels digital och handlar i sin tur om att överföra ettor och nollor från punkt a till b. Överföringshastigheten mäts i bitar per sekund (bit/s) medan lagringen av data mäts i det engelska uttrycket byte. Enskilda ettor och nollor ger inte mycket information men om de grupperas i någon form av kod så kan de representera olika tecken, två vanligaste om detta är ”ASCII” och ”unicode”. (Berti, et al., 2012) (Sen, 2014)

Ett tecken från tangentbordet på en vanlig dator skapar en kombination av 8 ettor och nollor. Denna grupp med ettor och nollor kallas för en byte på engelska och det är en sammansättning av orden ”Binary Term”. Användningen av prefix nämns framför ordet byte som till exempel kilo och mega för att det lätt blir stora tal när dessa används. I datakommunikationssammanhang räknas bitar som överföringsenhet och sekund som tidsenhet, av detta fås bit/s eller bps. När det talas om begreppet Mbit/s så handlar det om överföringshastighet och när man talar om MB så handlar det om en fils storlek. (Sen, 2014) (Berti, et al., 2012)



Figur 2. Byte och bit.

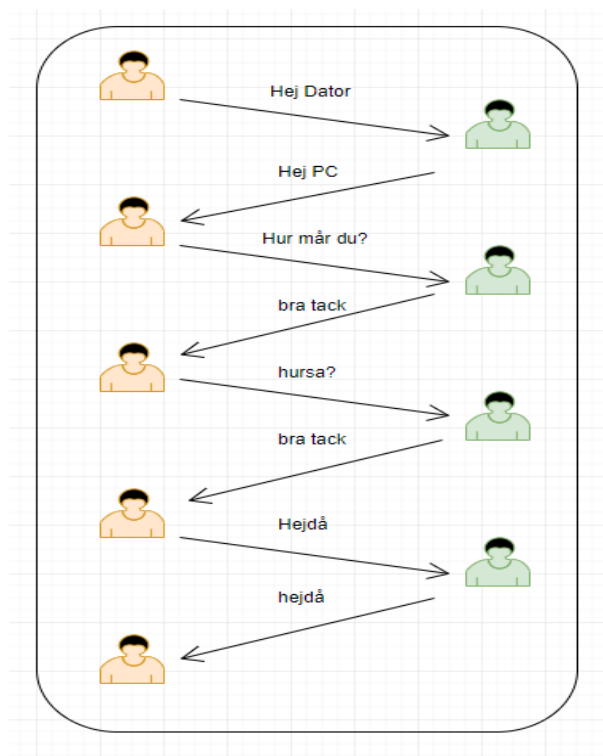
## 2.2 Datakommunikationsregler och protokoll

Ett av de mest centrala begreppen inom datakommunikation är protokoll. I internetvärlden finns det en del begrepp som slutar på ”P” som exempel TCP, IP, HTTP, SNMP, HTTP. P:et står för protokoll (protocol) och detta betyder att det är en överenskommelse mellan två eller flera parter om de regler som gäller under kommunikationen. I det vardagliga livet används kommunikationsprotokoll hela tiden för att genomföra och underlätta en vettig kommunikation.

Detta kan jämföras med två personer som för en dialog med varandra. För att kommunikationen ska starta säger bägge parterna ”hej” och för att avsluta kommunikationen säger de ”Hejdå”. Om den ena parten ej uppfattar det som sägs kan den till exempel skicka ”hursa” varav den andra då kan upprepa vad den sade senast. Uttrycken ”hej”, ”hejdå” och ”hursa” kan jämföras med att vara protokollkoder som kan underlätta tydningen av den överförda informationen. Skulle båda parterna råka tala i munnen på varandra, tystnar båda,



väntar en stund så att någon av dom återupptar diskussionen senare. (Sen, 2014) (Berti, et al., 2012)



**Figur 3. Kommunikation mellan två parter.**

### 2.3 Protokollfunktioner

Protokoll innehåller funktioner som fungerar på olika sätt i olika protokoll. De vanligaste funktionerna som protokollen hanterar är:

- Dialogen, vem som får skicka datameddelanden och i vilken ordning.
- Adresseringen, hur man berättar vem man är och vem man vill kommunicera med.
- Ramformat, hur informationen paketeras eller ”ramas in”.
- Felhantering, hur man ska hantera olika felsituationer som kan uppstå.

### 2.4 Modeller för datakommunikation och protokoll

För att utförandet och felsökning gällande datakommunikation skall gå lättare så har det utvecklats modeller eller så kallade kommunikationsarkitekturer för datakommunikationen.

Det finns många modeller för olika syften men två av de vanligaste är OSI – modellen och TCP/IP. Dessa modeller är uppdelade i skikt för organisering av datakommunikationen. Var och en av skikten har egen funktion och kommunicerar med den ”andra sidans” skikt. (Hallberg, 2009)

#### **2.4.1 Open System Interconnection, OSI – modellen**

OSI modellen definierar metoder och protokoll som behövs för att ansluta en dator till någon annan dator över ett nätverk. Det är en konceptmodell som används för nätverksdesign och inom tekniska nätverkslösningar. OSI modellen erbjuder ett sätt att förstå och visualisera hur datorer kommunicerar till varandra.

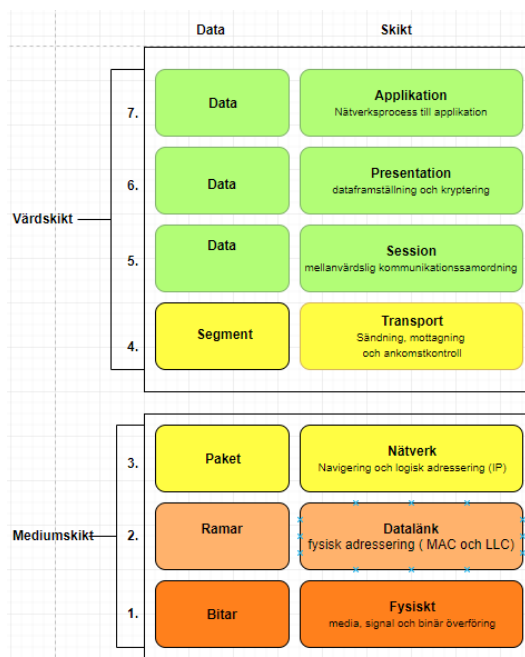
OSI-modellen består av sju olika skikt. Vilket medium som används och hur de skall hanteras placeras i de två skikten som är nederst. I det tredje skiktet placeras adresseringen och hur mottagaren nås. Störningshanteringar sker på skikt 4 och det språk som används i skikt fem och sex. I översta skiktet beskrivs det som man vill säga

Protokoll som berör datakommunikation är skapade från denna modell. Skillnaden som finns är att en del vill kombinera vissa skikt och andra vill flytta delar från ett skikt till ett annat, men skiktningens principen är en överenskommelse. (Hallberg, 2009)

OSI-modellens 7 skikt:

- Applikationsskiktet – 7: Kontrollerar hur operativsystemet och dess applikationer påverkar varandra med nätverket. Applikationerna som används till exempel Microsoft Word är inte en del av applikationsskiktet men arbetet som pågår där så drar den nytta av.
- Presentationsskiktet – 6: Tar data från de lägre skikten och omvandlar detta så det kan presenteras till systemet. Funktionerna som tar plats vid presentationslagret kan inkludera kompression, dekompression, kryptering och dekryptering av data.
- Sessionsskiktet – 5: Definierar anslutningen från en användardator till en nätverksserver, eller från en peer-dator på ett nätverk till en annan peer-dator. Dessa virtuella anslutningar är kallade till sessioner. De är kallade sessioner för att de lägger upp anslutningar som håller för en viss tid.

- Transportskiktet – 4: Hanterar flödet av information från en nätverksnod till en annan. Skiktet säkerställer att paketen är avkodade i den rätta sekvensen och att alla paket är mottagna.
- Nätverksskiktet – 3: Definierar hur datapaketen tar sig från en punkt till en annan i ett nätverk och vad som kommer in i ett paket. Nätverksskiktet använder olika paketprotokoll, till exempel Internet Protocol (IP) och Internet Protocol Exchange (IPX). Dessa paketprotokoll inkluderar källan och destinationen för dirigerad information. Dirigeringsinformationen i varje paket informerar nätverket vart paketets ska skickas för att nå sin destination och berättar den mottagande datorn varifrån paketet härstammar.
- Datalänksskiktet – 2: Definierar standarder som tilldelar förståelse till bitarna som bärs av det fysiska skiktet. Det etablerar ett protokoll som är tillförlitligt genom det fysiska skiktet så nätverksskiktet kan sända sin data. Datalänksskiktet innehåller korrigerande och fel detektering så att en tillförlitlig dataström kan säkerställas. Dataelementen som bärs av datalänksskiktet heter ramar.
- Fysiska skiktet – 1: Definierar egenskaperna hos det fysiska medium som används för nätverksanslutningar. De fysiska lagerspecifikationerna resulterar i ett fysiskt medium, en nätverkskabel som kan överföra en ström av bitar på det fysiska nätverket. Den fysiska anslutningen kan vara punkt till punkt eller multipunkt och det kan bestå av halv-duplex eller full-duplex överföringar. (Hallberg, 2009)



Figur 4. OSI och skikten 1 – 7.

## 2.5 Seriell kommunikation

Konceptet för seriell kommunikation är enkelt. Serieporten skickar och mottar bytes av information en bit åt gången. Detta är långsammare än parallell kommunikation, som tillåter överföring av en hel byte åt gången; men den är mera enkel och kan användas över ett längre avstånd. Längskillnaden kan förklaras genom att med parallell kommunikation så får det inte vara längre än 20m totalt, med inget mera än två meter mellan apparaterna och med seriell kommunikation uppnås längder till 1200m. (National Instruments, 2018) (Contec, 2019)

Denna typ av kommunikation är en metod som använder endera en eller två linjer för att skicka och motta data. När detta tillåter en koppling med få signaltrådar så är en av fördelarna att det inte används mycket material och kostnaden är på den mindre sidan. (National Instruments, 2018) (Contec, 2019)

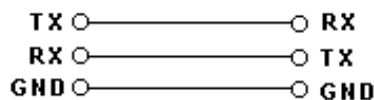
Seriell kommunikation är oftast använt för att överföra ”ASCII” data. Kommunikationen är komplett genom användning av 3 överföringslinjer, jord (ground), överföring (transmit) och mottagning (receive). De viktigaste seriella inställningarna är baud rate, data bitar (data bits), stop bitar (stop bits) och paritet (parity). För att två portar skall kommunicera, så måste dessa parametrar stämma överens. (National Instruments, 2018) (Contec, 2019)

- Baud rate är en hastighetsmätning för kommunikation, den indikerar hur många bitar som överförs per sekund. Till exempel 300 baud = 300 bitar/s eller 9600 baud, 9600 bitar/s. Men ju högre baud rate som används, desto kortare distans får det vara mellan enheterna.
- Databitar är en mätning av de aktuella databitarna i en överföring. När till exempel en dator skickar ett paket information, så behöver inte den aktuella data vara fulla 8 bitar. Standardiserade värden för datapaketet är 5, 7 och 8 bitar och vilken inställning som väljs beror på den information som man vill överföra.
- Stopbitarna är till för att signalera slutet av en kommunikation för ett paket. De mest typiska värden för detta är 1, 1.5 och 2 bitar.
- Paritet är en enkel form av fel koll som används i seriell kommunikation. Paritet kan vara i fyra sorter: lika (even), olika (odd), markerad (marked) och åtskilda (spaced). Att inte använda paritet går att välja. (National Instruments, 2018) (Contec, 2019)

### 2.5.1 Seriella kommunikation standarder

Inom seriell kommunikation så är det oftast tre standarder som det talas om, RS232, RS422 och RS485 där RS står för *Recommended Standard* och dessa tre är då *Electronic Industries Association* (EIA) kommunikationsstandarder. (National Instruments, 2018) (Contec, 2019)

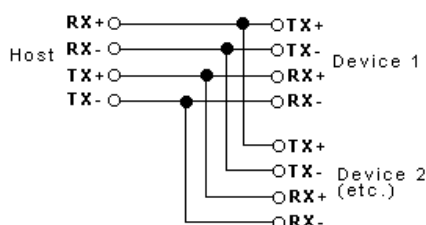
- RS232: Denna standard är brett använd och är för det mesta applicerad på datorer och är även kallat "EIA-232". Vad den kan användas för är till exempel ihopkoppling med en mus, printer, modem och även industriella instrument. RS-232 är begränsad till punkt-till punkt koppling mellan datorers seriella port och applikationer.



Figur 5. RS232 koppling. (Omega, 2019)

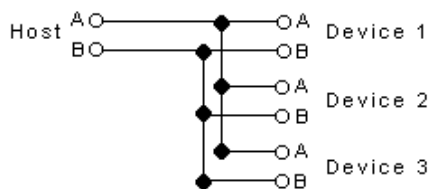
- RS422: Denna standard var presenterad för att komma åt högre datahastigheter att bli överförda över seriella datalinjer än vad som var möjligt med RS232. RS422 använder en differentiell elektrisk signal, i motsats till RS232 som nyttjar

obalanserade signaler refererade till grund. Denna differentiella överföring använder två linjer vardera för sändnings och mottagnings signaler och denna användningsmetod kallas då för "full-duplex". På grund av detta resulterar det i en bättre störnings immunitet och längre avstånd jämfört med RS232. På grund av detta så är denna standard en bättre passning för industriella applikationer. (National Instruments, 2018) (Contec, 2019)



Figur 6. RS422 koppling. (Omega, 2019)

- RS485: Denna standard är en förbättring av RS422 med sin möjlighet att kommunicera med 32 applikationer genom halv-duplex (half-duplex), på två trådar med avstånd upp till 1200m. Data som skickas via RS485 överförs differentiellt med två trådar som är ihoptvinnade (twisted pair). Detta sätt erbjuder en hög störningsimmunitet och möjligheter till långa avstånd mellan applikationer. Störningsimmuniteten och "multi-drop" kapaciteten gör så att RS485 är en av de bättre valen av seriell kommunikation inom industriella applikationer som kräver att många enheter är kopplade med varandra. RS485 är en förbättrad version av RS422, i och med detta så går det att kontrollera alla RS422 enheter med RS485. (National Instruments, 2018) (Contec, 2019)



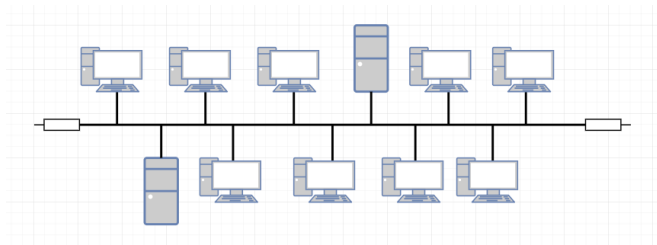
Figur 7. RS485 2-trådkoppling. (Omega, 2019)

## 2.6 Topologier i nätverk

Nätverkstopologier syftar på formen av ett nätverk, hur alla punkter i ett nätverk är ihopkopplade. Nätverk i sig själv kan vara kopplade i flera olika sorters topologier och valet av topologi är oftast ett av de viktigaste valen när planeringen av ett nätverk utförs. Topologierna har olika priser, nivåer av prestanda och pålitlighet. (Hallberg, 2009)

### 2.6.1 Busstopologi

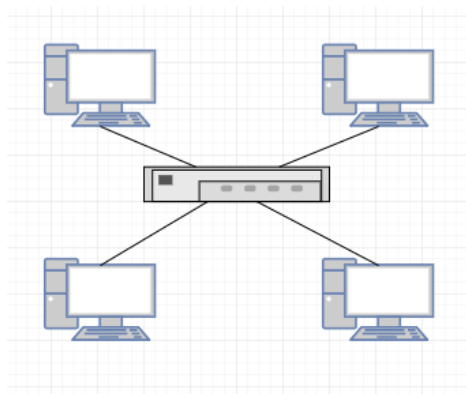
En busstopologi är ett nätverk där en nätverkskabel används från den ena sidan till den andra, med olika nätverksapplikationer som är sammanförda med kabeln vid olika ställen på linjen. I början och i slutet av busslinjen så måste det finnas ett termineringsmotstånd, utan detta motstånd så kommer nätverket inte att fungera. (Hallberg, 2009)



Figur 8. Busstopologi.

### 2.6.2 Stjärntopologi

En stjärntopologi är ett nätverk där alla punkter i nätverket är kopplade till ett centralt ställe, till exempel att ett antal datorer är ihopkopplade till en hubb/switch. Sammankoppling av flera centrala ställen till varandra för att utöka nätverket vidare kan utföras om så behövs. (Hallberg, 2009)



Figur 9. Stjärntopologi.

## 3 Maskinvara

I detta kapitel behandlas en del av maskinvaran som har använts under arbetet och information om den. Valen av maskinvara till projektet var enbart att hitta något som skulle passa till att få en fungerande kommunikation mellan en Simatic S7-1200 PLC och handmanöverdonet MCD samt något för att kunna lyssna på kommunikationen mellan CVC600 och MCD.

En del av den maskinvara som behövdes var en PLC, kommunikationsmodul, strömkälla och handmanöverdonet MCD. Den största delen av maskinvaran hade Solving men maskinvaran för att lyssna på kommunikationen hittades vid Novia. Det som Novia hade var en USB-COM Plus modul användes för lyssningen av kommunikationen.

### 3.1 PLC

En programmerbar logisk styrenhet är en specialiserad dator som används för manövrering av maskiner och processer. De delar därför termer med datorer som processor, minne, programvara och kommunikation. Olikt en personlig dator så är en PLC utvecklad för att överleva i en hård industriell miljöatmosfär och vara väldigt flexibel i hur den arbetar med ingångar och utgångar. (plcdev, 2019)

#### 3.1.1 Simatic S7-1200 PLC

Siemens Simatic S7-1200 är en grundläggande styrenhet som består av teknologiska funktioner och integrerade ingångar och utgångar. De är kompakta och har en utrymmes effektiv konstruktion. Av Simatic S7-1200 serien finns det olika modeller på styrenheterna, till exempel 1211, 1212, 1214, 1215 och 1217.

Till en Simatic S7-1200 styrenhet går det att lägga upp till 8 signalmoduler men detta beror på vilken av 1200 seriens styrenheter som används. Signalmoduler och signalkort kan bli integrerade för att expandera digitala eller analoga ingångar och utgångar utan att påverka den fysiska storleken av styrenheten. Signalmodulerna kan även tillsättas på den högra sidan av styrenheten för att expandera kapaciteten av de analoga eller digitala ingångarna och utgångarna men detta påverkar den fysiska storleken av styrenheten.

På alla Simatic S7-1200 styrenheter kan det tillsättas 3 kommunikationsmoduler som möjliggör användningen av seriell kommunikation samt för anslutningen till olika distribuerade IO-enheter. Dessa kommunikationsmoduler stöder styrenheten i



kommunikationsuppgifter och bygger den fysiska bindningen till en kommunikationspartner, tar över etableringen av bindningen och dataförbindelse och förser den nödvändiga kommunikationsservicen för PLC:ns operativsystem och användarprogrammet. (Siemens, 2019) (Berger, 2013)

Den PLC som användes i detta examensarbete var en Siemens s7-1200 cpu 1215 dc/dc/rly. Denna PLC fanns vid Solving och den passade till sitt ändamål. För mera teknisk information, se bilaga 1.



**Figur 10. Simatic S7-1200. (PLC-city, 2017)**

### **3.2 CM 1241**

CM 1241 är en kommunikationsmodul som tillåter seriell dataöverföring via punkt-till-punkt förbindelse. Dessa moduler är tillgängliga i två koncept, RS232- och RS422/485-gränssnitt, som väljs beroende på de fysiska överföringsegenskaperna. Det finns standardprotokoll för punkt-till-punkt kommunikation (ASCII), Modbus kommunikation, och det universella seriella gränssnittet, *USS drive protocol*. Punkt-till-punkt kommunikation passar för förbindelse mellan till exempel printrar, modem, streckodsläsare och PLC. (Berger, 2013)

Valet av kommunikationsmodul till projektet blev en CM1241 RS422/485 för att det antogs att kommunikationen mellan MCD:n och CVC600 byggdes på en kommunikation via ett RS485 gränssnitt, och denna kommunikationsmodul är byggd för RS422/485. För mera teknisk information om CM1241, se bilaga 2.



**Figur 11. Kommunikationsmodul CM 1241 RS422/485. (Automation24, 2019)**

### 3.3 Manual Control Device

Manual Control Device är ett handmanöverdon för att kunna köra en förarlös truck, AGV, manuellt. På MCD:n finns spakar, en knapp och en potentiometer. I figur 12 är spakarna, knappen och potentiometern numrerade och dessa blir förklarade i denna del av arbetet.

En AGV kan väljas att köras i manuellt läge (MAN), semi-automatiskt (SEA) och automatiskt (AUT). Med MCD:n, väljs ett av dessa lägen genom att byta plats lägesväljaren (2). I semi-automatiskt läge är hastigheten styrd manuellt men styrningen är automatiskt och i automatiskt läge så körs AGV:n helt automatiskt enligt till exempel linjeföljning.

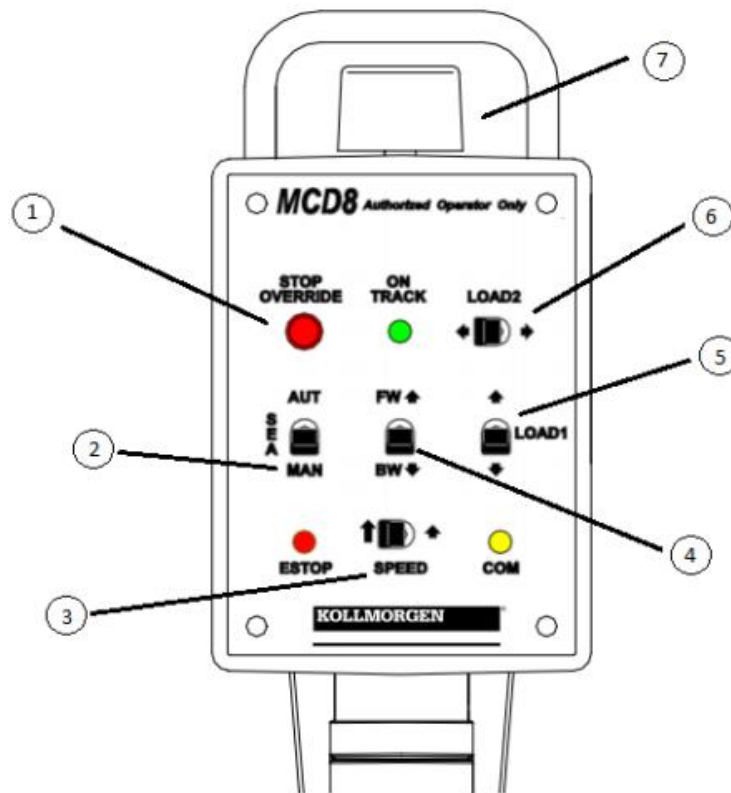
På handmanöverdonet finns även tre olika LED-lampor, en grön LED ”on track” som aktiveras när AGV:n körs längs en linje i SEA eller AUT läget, en röd LED (ESTOP) som aktiveras när något kommer i vägen för en detektor som är placerat på AGV:n samt före det har skett en kvittering av nödstopps knappen som aktiveras vid uppstartning av en AGV och en gul LED (COM) som börjar blinka när datorn i AGV:n och MCD:n börjar kommunicera.

Knappen ”Stop Override (1)” trycks på om AGV:n ska köras fast än ”ESTOP” har aktiverats. Hastighetsspaken (3) kan tryckas endera till vänster eller höger. När spaken trycks till höger så körs den i en långsammare hastighet och när spaken trycks till vänster körs den i en snabbare hastighet.

Finns två olika riktningalternativ, framåt (FW) eller bakåt (BW), detta väljs genom att lägga spaken (4) i det läge som önskas av alternativen. MCD:n har två fjäderbelastade lasthanteringsspakar vid namn ”load1” och ”load2”. Lasthanteringsspak 1 (5) är för

lyftningen och den går att trycka upp och ner. Lasthanteringsspak 2 (6) är för till exempel körning av ett par pallgafflarna utåt eller inåt. Detta görs genom att endera lägga spaken åt höger eller vänster.

På toppen av MCD:n finns en potentiometer (7) som används för styrningen av en AGV. Genom att endera vrida den medsols eller motsols så styrs hjulen på AGV:n. (Turesson, 2016)



Figur 12. Överblick av MCD. (Turesson, 2016)



Figur 13. MCD

### 3.4 USB-COM485-PLUS1

USB-COM485-PLUS1 kommunikationsanordningen erbjuder en enkel metod för anpassning till seriella anordningar med RS485 gränssnitt till moderna USB portar. Detta är en sak som lyckas genom att införliva en industriell standard *FTDI FT232R USB-seriell bridge chip*. Datatrafik kan ses visuellt genom att se på de två LED-lamporna vid namn TXD/RXD. (FTDI chip, 2010)



Figur 14. USB-COM-485-PLUS1 kommunikationsanordning. (FTDI chip, 2010)

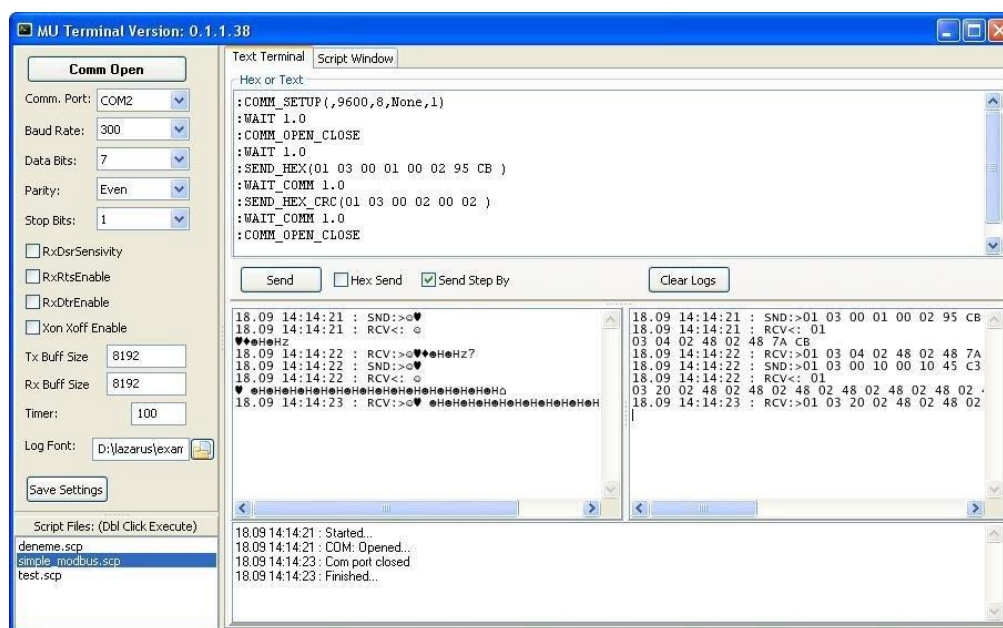
## 4 Programvara

Inom denna del kommer programvaran behandlas. Till detta hör program som användes till arbetets utförande. Programvaran består av programmeringsverktyg och verktyg för lyssning av datakommunikation.

### 4.1 MU Terminal

MU Terminal är ett kommunikationsverktyg för externa enheter som till exempel använder ASCII och MODBUS RTU. Med detta verktyg kan ändring av loggfönstret till sann typ av ASCII utföras. Programvaran kan användas med internet eller lokalt nätverk MODBUS TCP eller HTTP. Denna programvara kan även användas för loggning av serieportar. (Sourceforge, 2012)

Detta var det första programmet som användes för lyssning av kommunikationen i arbetet.

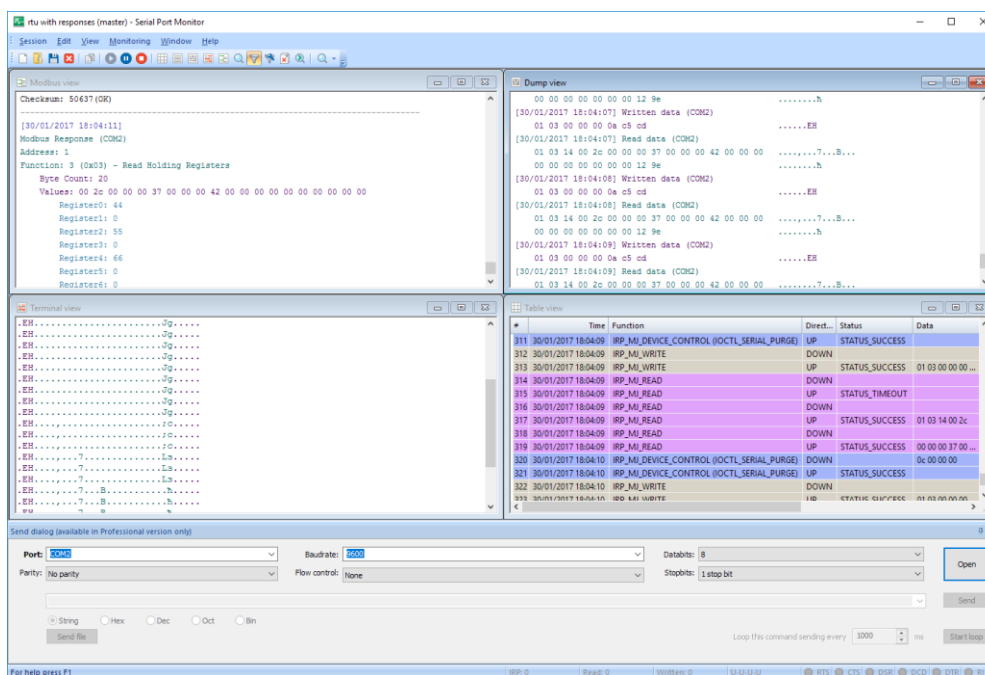


Figur 15. MU Terminals huvudfönster. (Sourceforge, 2012)

### 4.2 Serial Port Monitor

Serial Port Monitor är ett systemverktyg för RS232/RS422/RS485 övervakning och kontroll. COM övervakningen visar loggar och analyserar all seriell portanvändning i ett system. Detta är ett sätt lokalisera problem som dyker upp under till exempel utveckling av seriella kommunikationsapparater. (Eltima, 2018)

Detta program var det andra programmet som användes för lyssningen av kommunikationen i arbetet.



Figur 16. Serial Port Monitor. (Eltima, 2018)

### 4.3 TIA Portal

Programvaran TIA Portal är ett verktyg för programmering av Siemens utrustning. Namnet TIA Portal kommer från Totally Integrated Automation Portal och programmeringsverktyget är uppbyggt av flera programmeringsverktyg och därav kommer namnet TIA Portal. De programmeringsverktyg som TIA Portal består av är Step 7, WinCC, Sinamics Startdrive, Simocode ES och Simotion Scout Tia.

Ett av världens mest kända och använda programmeringsverktyg inom industriell automation är Simatic Step 7. Simatic WinCC är en programvara designad för visualisering och användning av Siemens HMI-produkter och detta verktyg används från enkla operatörspaneler till mera invecklade SCADA-system. För programmering och konfigurering av Siemens frekvensomvandlare och för integreringen av detta i automationslandskap används Sinamics Startdrive. Simocode ES erbjuder hantering av till exempel lågspänningsmotorer med konstant fart och ger omfattande skydd, övervakning och kontrollfunktioner. Simotion Scout kombinerar uppgifter inom rörelse, PLC, teknik och kör konfigurationen i ett enda system.

Alla dessa ovannämnda programmeringsverktyg är integrerat i TIA Portal och detta finns det tillgång till när programvaran används. Före lanseringen av TIA Portal så var dessa enskilda program som användes skilt ifrån sig och detta resulterade ofta i en sämre överblick över de program som skapades. Programmeringsspråken i Simatic Step 7 (TIA Portal) möter kraven som IEC61131-3 ställer. IEC61131-3 standarden är den enda standarden som är giltig över hela världen för programmeringsspråken av programmerbara logiska styrenheter. (Siemens, 2019)

#### 4.3.1 OB

Organisationsblock (Organization Blocks, OBs) är gränssnittet mellan operativsystemet och användarprogrammet. De anropas av operativsystemet och utför följande processer:

- automationssystemets uppstartning
- cyklisk programexekvering
- alarm-styrd programexekvering
- felhantering.

Organisationsblock kan programmeras i det önskemål som anges av användaren och således datorns beteende. Det finns olika optioner för användningen av organisationsblock i programmen, till exempel uppstartning-, cyklisk-, tidsfel- och diagnos-organisationsblock. Dessa organisationsblock kan läggas in i olika projekt och programmeras. (Berger, 2008) (Siemens, 2012)



**Figur 17. Organization block.**

#### 4.3.2 FB

Funktionsblock (Function Blocks, FBs) innehåller underprogram som blir exekverad när ett funktionsblock är anropat av ett annat kodblock.

Funktionsblock är block som lagrar sina värden i instansdatablock så att dessa värden är tillgängliga även efter att blocket är bearbetat. För denna orsak är de även kallad block med ett minne.

Funktionsblock används för uppgifter som inte kan bli implementerad med funktioner till exempel:

- Alltid när *timers* och *counters* behövs i ett block.
- Alltid när informationen behöver lagras i programmet; till exempel när man väljer på förhand ett operationsläge med en knapp.

Ett funktionsblock kan anropas flera gånger vid olika tillfällen och lägen i ett program. Detta underlättar programmeringen av repetitiva, komplicerade funktioner. (Berger, 2008) (Siemens, 2012)



**Figur 18. Function block.**

### 4.3.3 FC

En funktion (function, FC) innehåller ett program som exekveras när andra block anropar en funktion. Dessa funktioner är kodblock utan minne. Data av de temporära variablerna är bortglömda efter att funktionen är bearbetad. Globala datablock kan användas för att lagra data av en funktion.

Exempel på hur funktioner kan bli använda:

- Returnera funktionsvärden till det anropande blocket; till exempel en matematisk funktion.
- Exekvera teknologiska funktioner; till exempel individuella styrningar med binäroperationer. (Berger, 2008) (Siemens, 2012)





**Figur 19. Function.**

#### 4.3.4 DB

I motsats till kodblock innehåller datablock inga instruktioner men är till för att lagra användardata. Detta betyder att datablock innehåller en variabel data som användarprogrammet använder för bearbetning.

Globala datablock lagrar data som kan användas av alla andra block. Den maximala storleken av datablock varierar, beroende på datorn. Strukturen av globala datablock kan bli specificerade om det krävs.

Applikationsexempel:

- Lagrande av informationen i ett varuhussystem. "Vilken produkt är placerad var?"
- Lagrar recept för specifika produkter

Varje FB, FC och OB kan läsa data från ett globalt DB eller skriva data i ett globalt DB. Den data behålls i datablocket även när datablocket är avslutat. (Berger, 2008) (Siemens, 2012)



**Figur 20. Data block.**

## 5 Utförande

Uppgifterna i detta arbete var uppdelade i flera olika delmoment för att slutligen komma fram till ett dokumenterat programblock för att kunna styra en AGV manuellt med en MCD via en Siemens s7-1200 PLC.

Inom detta kapitel kommer inte alla delmoment att behandlas men det kommer berättas till viss del om hur man har gått till väga för att kunna lyssna på en datakommunikation mellan två parter som talar med varandra via ett RS485 gränssnitt samt hur ett projekt i TIA Portal och programmeringen kan se ut och tillverkas.

Själva programmet skrevs enligt de instruktioner som finns i källförteckningen *Programming Guideline* som är en manual från Siemens. (Siemens, 2017)

### 5.1 Undersökning av maskinvara och programvara

Arbetets början var att ta reda på hur kommunikationen mellan handmanöverdonet och datorn CVC600 i en automatisk truck såg ut. Det som Solving hade tagit reda på så att man kunde börja med bearbetningen av hur kommunikationen såg ut var kommunikationens gränssnitt, baudrate, databits, startbitar och stopbitar. Dessa inställningar var nödvändigt att veta under arbetets gång för att få en fungerande kommunikation.

För att lyssna på en kommunikation mellan enheterna behövdes en sorts programvara samt en maskinvara för att kunna anslutas på signaltrådarna som finns mellan MCD:n och CVC600. Till detta gjordes en undersökning av vad som kunde användas för att lyssna på kommunikationen. Det kommunicerades med lärare från Novia och det visade sig att de hade en modul USB-COM485-PLUS1 som är anpassat för kommunikation med RS485 gränssnitt. Samtidigt nämndes det om programvaran MU Terminal kunde tänkas användas för att lyssna på kommunikationen.

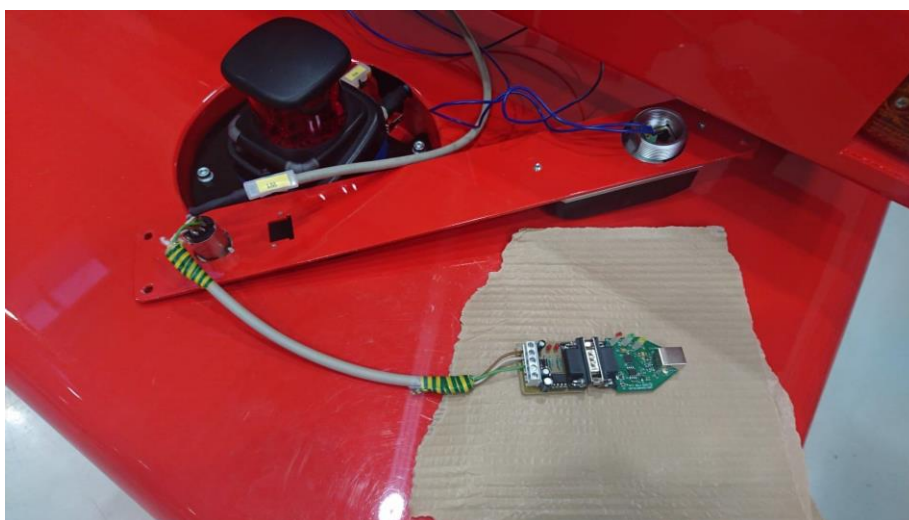


Figur 21. USB-COM485-PLUS1. (FTDI chip, 2010)

Efter bestämmelsen av vilken sorts programvara och maskinvara som skulle kunna användas för att lyssna på kommunikationen så togs dessa delar med till Solving. Där hade dom en AGV som användes för uppvisning så den blev till testningsvagn för kommunikationslyssningen.

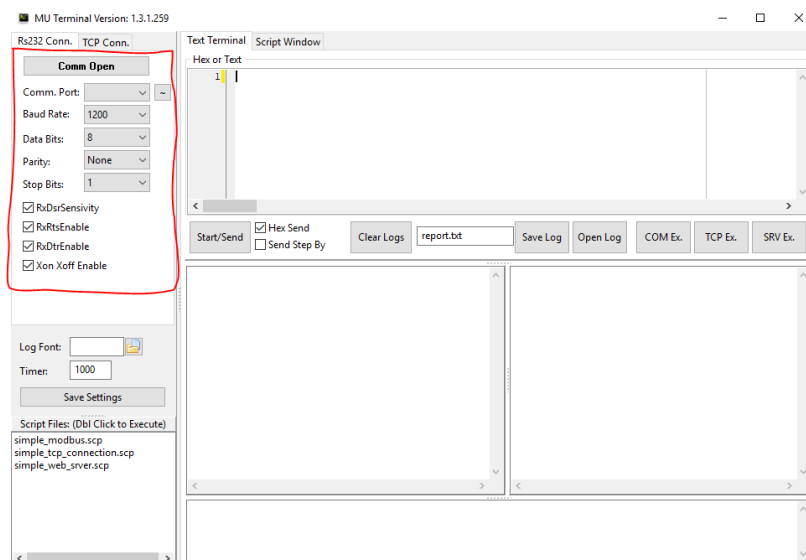
### 5.1.1 Loggning vid Solving

Genom att analysera elskemat för AGV:n, så kunde signaltrådarna och jordningen för handmanöverdonet identifieras. Med denna kunskap löddes tre trådar där uttaget för MCD:n fanns på AGV:n. Efter detta anslöts modulen USB-COM485-PLUS1 till signaltrådarna och jordningen som sedan for via kortet med USB-kabel till ett USB-uttag på en dator med programvaran för kommunikationslyssning.



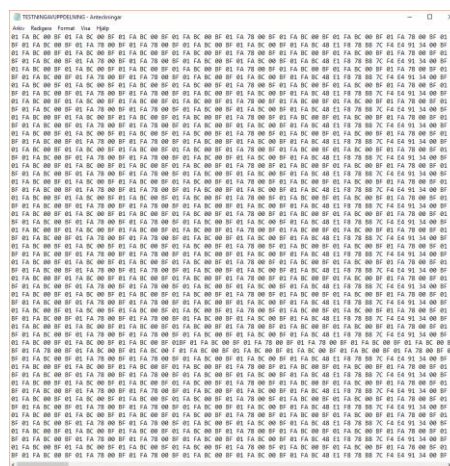
Figur 22. USB-COM485-PLUS1 ihopkopplat med AGV.

När allt praktiska var gjort för lyssningen av kommunikationen var det bara att börja lyssna. Programvaran MU Terminal användes och kommunikationsparametrarna lades in i programvaran. Figur 23 fungerar som exempelbild på var parametrarna ställs in.



Figur 23. MU Terminal parameterinställning.

Det noterades att datapaketet som skickades mellan MCD och CVC skedde med korta tidsintervaller och en stor mängd hexadecimala data uppstod. Dessa små tidsintervaller mellan enheterna ledde till en genomgång av stor mängd data som lades in i en textfil för uppdelning och det var oklarheter hur denna data skulle uppdelas. En del av den data som kom med i en kommunikationslyssning visas i figur 24 och 25.



Figur 24. Textfil som innehåller loggningsdata.

Det noterades att loggningen av data med MU Terminal var en aning rörigt och på grund av detta eftersöktes en annan programvara för att se på datapaketet som skickades mellan enheterna och få en bättre syn och uppdelning av datameddelanden. Internetsökningar gjordes för att hitta lämplig programvara för att lyssna på kommunikationen och resultatet blev Serial Port Monitor.

Serial Port Monitor användes i nästa testning av kommunikationslyssningen, samma kommunikationsparametrar ställdes in och när AGV:n aktiverades och kommunikationen påbörjades så noterades en skillnad med uppdelningen av datameddelanden. Datameddelanden som uppstod genom tryckningar av knapparna på MCD:n var bättre uppdelat och det vara lättare att filtrera bort en mängd av meddelanden som skickades mellan enheterna.

Time	Function	Event	Status	Data	Data length	D. R.	Port	Comments
15:42	MFI_READ	DOWN					3	COM3
15:43	MFI_READ	UP	STATUS_SUCCESS	4178:00	4k		3	COM3
15:44	MFI_READ	DOWN					2	COM3
15:45	MFI_READ	UP	STATUS_SUCCESS	MFI	4		2	COM3
15:46	MFI_READ	DOWN					3	COM3
15:47	MFI_READ	UP	STATUS_SUCCESS	4178:00	4k		3	COM3
15:48	MFI_READ	DOWN					2	COM3
15:49	MFI_READ	UP	STATUS_SUCCESS	MFI	4		2	COM3
15:50	MFI_READ	DOWN					3	COM3
15:51	MFI_READ	UP	STATUS_SUCCESS	4178:00	4k		3	COM3
15:52	MFI_READ	DOWN					2	COM3
15:53	MFI_READ	UP	STATUS_SUCCESS	MFI	4		2	COM3
15:54	MFI_READ	DOWN					3	COM3
15:55	MFI_READ	UP	STATUS_SUCCESS	4178:00	4k		3	COM3
15:56	MFI_READ	DOWN					2	COM3
15:57	MFI_READ	UP	STATUS_SUCCESS	MFI	4		2	COM3
15:58	MFI_READ	DOWN					3	COM3
15:59	MFI_READ	UP	STATUS_SUCCESS	4178:00	4k		3	COM3
16:00	MFI_READ	DOWN					2	COM3
16:01	MFI_READ	UP	STATUS_SUCCESS	MFI	4		2	COM3
16:02	MFI_READ	DOWN					3	COM3
16:03	MFI_READ	UP	STATUS_SUCCESS	4178:00	4k		3	COM3
16:04	MFI_READ	DOWN					2	COM3
16:05	MFI_READ	UP	STATUS_SUCCESS	MFI	4		2	COM3
16:06	MFI_READ	DOWN					3	COM3
16:07	MFI_READ	UP	STATUS_SUCCESS	4178:00	4k		3	COM3

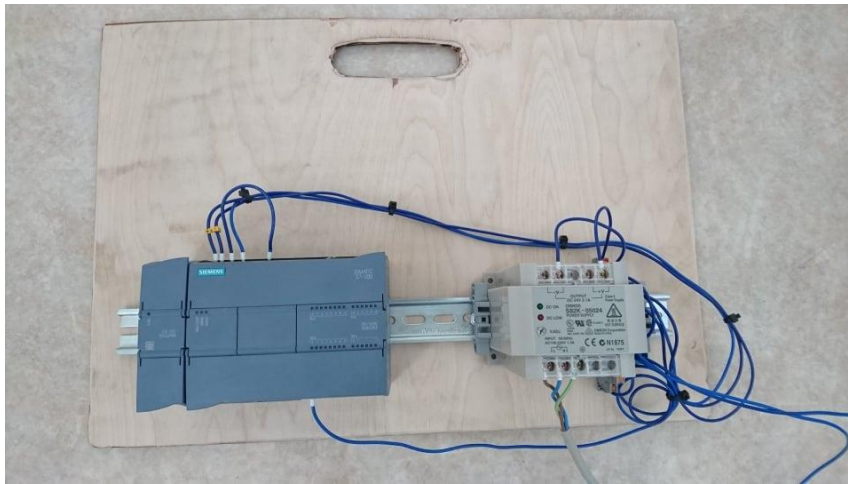
Figur 25. Serial Port Monitor loggning.

Genom att testa alla spakar och kombinationer på MCD:n när det samtidigt lyssnades på kommunikationen så utreddes det hur datameddelanden såg ut när en specifik knapp blev tryckt. De datameddelanden som var avsedda för MCD:s funktioner lades i en skild lista för att sedan användas till en Siemens PLC.

## 5.2 Testsystemets uppbyggnad

På samma gång som loggningen gjordes utfördes även en uppbyggnad av testsystemet. Detta var en mycket enkel sak att få ihop varav det mesta var färdigt ihop lagt. Det som var färdigt sammansatt och ihopkopplat var PLC och strömkällan samt så var uttaget för MCD på en kabel färdigt för att anpassas till kommunikationsmodulen.

Maskinvara och material som behövde läggas till för ett fungerande testsystem var en kommunikationsmodul CM1241, DE-9 stöpsel och ett motstånd för att få en termoresistans på andra sidan av busslinjen för MCD och allt detta placerades på en fanerskiva.



**Figur 26. Testsystem.**

## **6 TIA Portal**

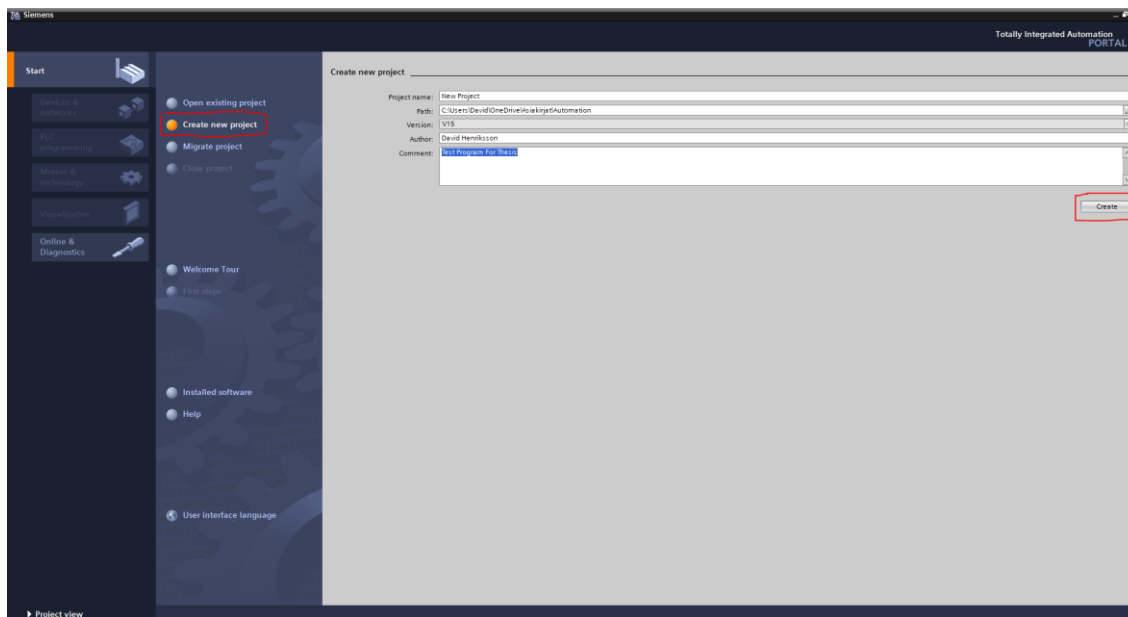
För att få ihop ett lämpligt program så att MCD och PLC talar med varandra användes programverktöget TIA Portal v15 från Siemens. Denna programvara hade Novia till sitt bifogande som studielicens.

Även under detta skede så planerades det vilket programmeringsspråk som skulle användas, de tre som det var att välja över var LAD, FBD och SCL. Det bestämdes för programmeringsspråket LAD för att valet av språk fick väljas på basis av vad som ansågs vara lättast i programmeringsväg. Att programmera i FBD eller SCL fungerar lika bra som att programmera i LAD men för ovana PLC-programmerare kan LAD vara ett programmeringsspråk som är lättare att förstå, se över och använda. FBD har även en enkel överblick men om så önskas kan man byta programmeringsspråk när applikationen är färdig.

För att skapa ett program i TIA Portal så är ett av de första stegen att skapa ett nytt projekt. I tillverkningen av det nya projektet ska maskinvara väljas och maskinvarans anpassningar göras för applikationen som tillverkas.

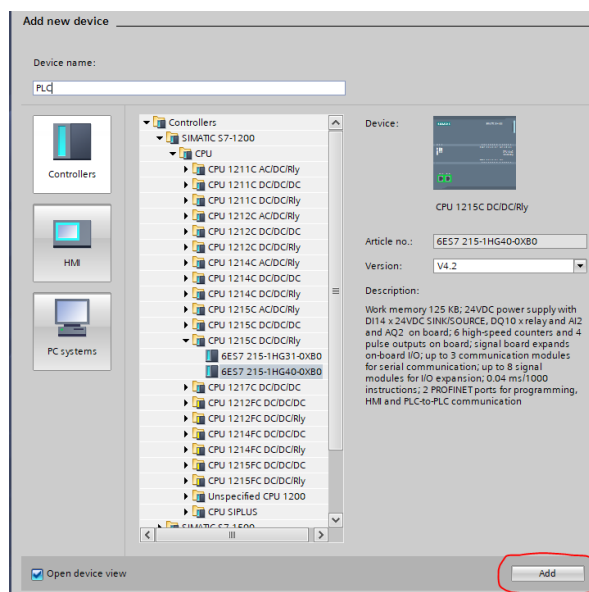
### **6.1.1 Skapa ett projekt och anpassning av maskinvara**

Inledningsvis startades verktyget TIA Portal, detta öppnar fönstret portalvyn och här väljs att skapa ett nytt projekt. Passande projektnamn väljs och tillverkarens namn av projektet skrivs.



Figur 27. Nytt projekt i portalvyn.

När projektet har skapats väljs den PLC som kommer att användas, detta fungerar som en grund av maskinvarukonfiguration. Här valdes PLC:n *CPU 1215C DC/DC/RLY – 6ES7 215-1HG40-0XB0* och informationen om vad det är för PLC hittas på den högra sidan av den fysiska PLC:n som används. När PLC har blivit valt att den används så öppnas nästa del av TIA Portal upp, projektvyn



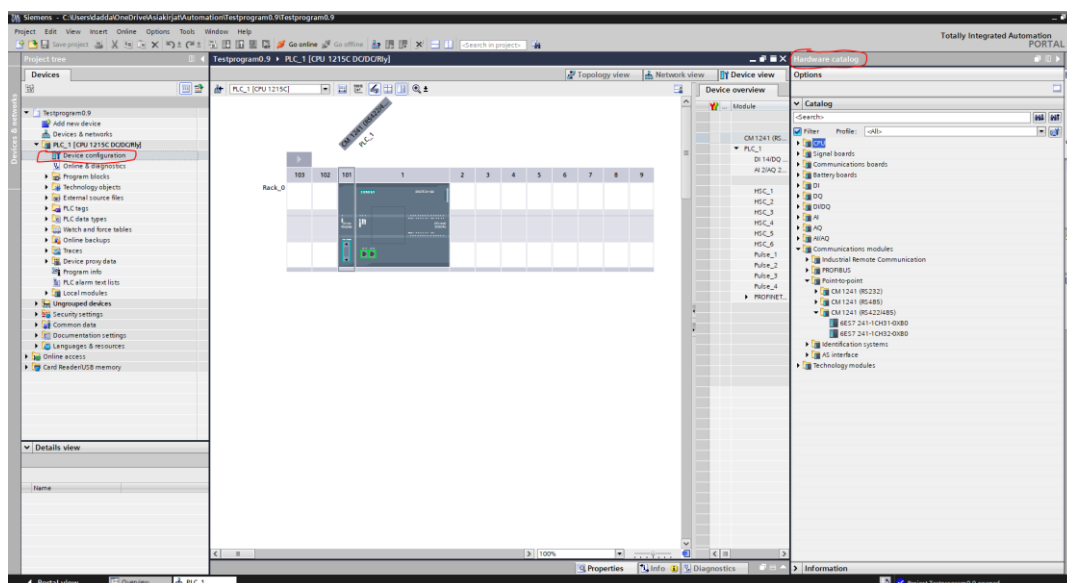
Figur 28. Val av PLC i portalvyn.



Figur 29. Information på PLC.

I projektvyn kan man börja med att till exempel skapa programkod under fliken ”Program blocks” eller fortsätta med att anpassa maskinvaran under fliken *device configuration*. För arbetets del så var det ännu att lägga in kommunikationsmodulen CM1241.

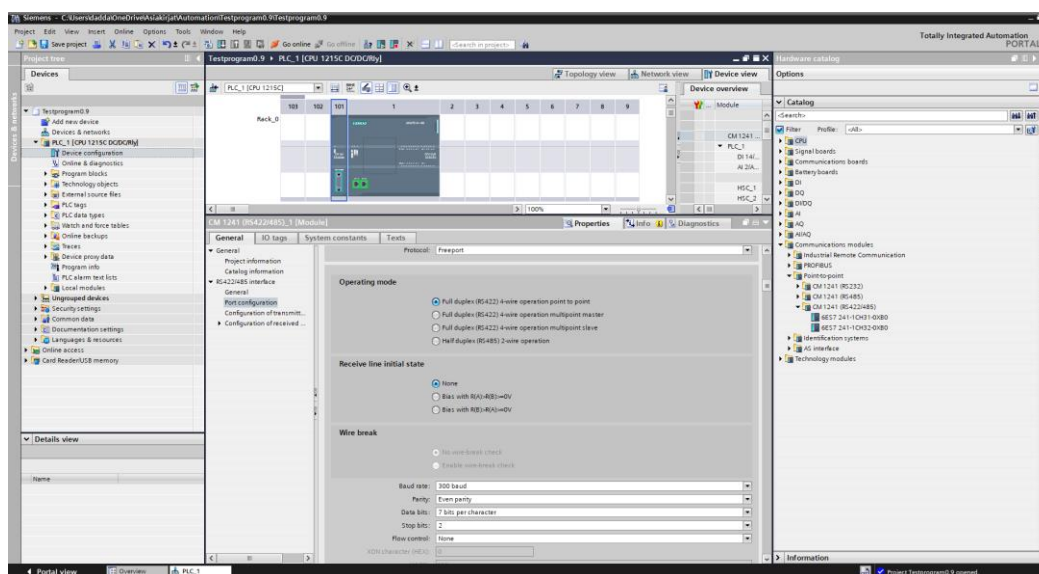
När man rör sig på *device configuration* dyker *hardware catalog* upp på högra sidan i projektvyn, här söker man sig till *communications modules* och väljer den kommunikationsmodul som används, i detta fall punkt-till-punkt kommunikationsmodulen *CM1241 (RS422/485) 6ES7 241-1CH32-0XB0*.



Figur 30. Applicering av maskinvara CM1241.



I *properties* på kommunikationsmodulen ställs parametrarna för CM1241 in. Exempel på dessa är inställningarna för det protokoll som skall användas och *operating mode* för vilket gränssnitt man arbetar med.

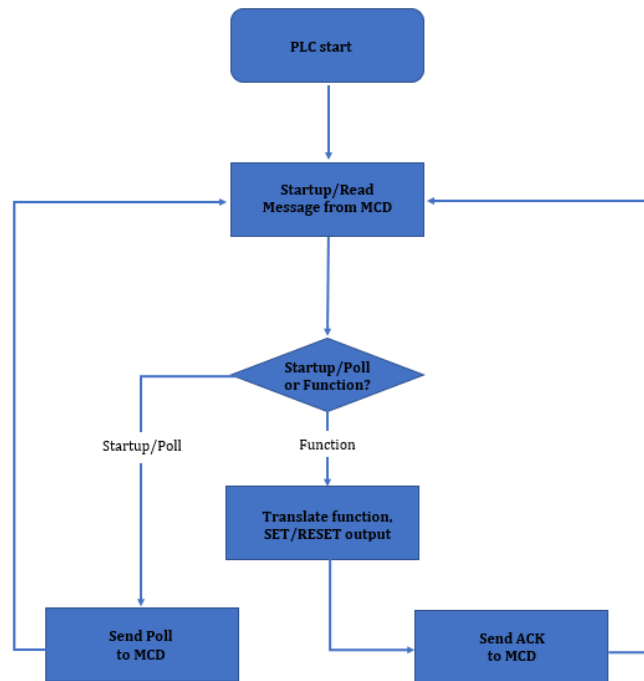


Figur 31. Parameterinställning av CM1241.

## 7 TIA Portal-programmering

Med maskinvaran konfigurerad börjades utvecklingen av ett programblock för att få en fungerande kommunikation mellan MCD och PLC samt aktiverar önskade utgångar. Det som var en viktig del i programmeringen var att dokumentera blocken väl och noggrant så att även andra kan förstå hur det har tillverkats och hur programblocket används.

Det som även gjordes i detta skede var tillverkningen av ett flödesschema som förklarar själva processen i blocket som utvecklades. Flödesschemat fungerar som en överblick av hur programmet utförs och en bättre tillsyn av hur man skall gå till väga för att skapa programblocket samt hanteringen av de meddelanden som skickas till och från MCD:n.



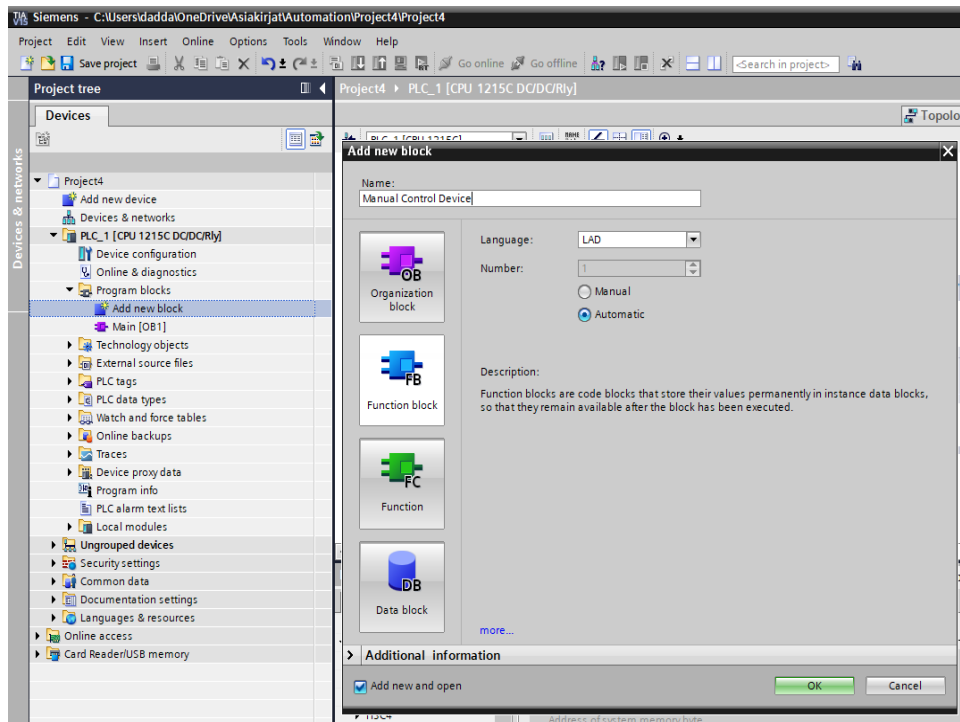
**Figur 32. Flödesschema över programblocket.**

I projektdelen av TIA Portal, påbörjades tillverkningen av ett FB och detta gjordes genom att i projektrådet ta sig fram till fliken programblock. I programblock fliken tillverkas och appliceras nya program.

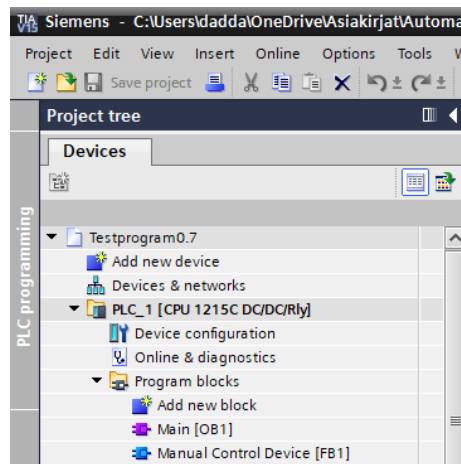
Genom dubbelklickning på *add new block* så kom en ruta upp på skärmen där blocket som ska användas väljs. Det finns OB, FB, FC och DB att välja på.

För projektets del så blev det ett funktionsblock som döptes till *Manual Control Device*. Även här kunde det väljas vilket programmeringsspråk som skulle användas och de tre som gick att välja på är LAD, FBD och SCL.

Efter att typen av block, namn på blocket och programmeringsspråk hade valts var det att fortsätta genom en tryckning på ok. På samma gång tillverkas ett datablock specifikt för det funktionsblock som tillverkades och efter detta så dök funktionsblocket upp i programblock fliken.



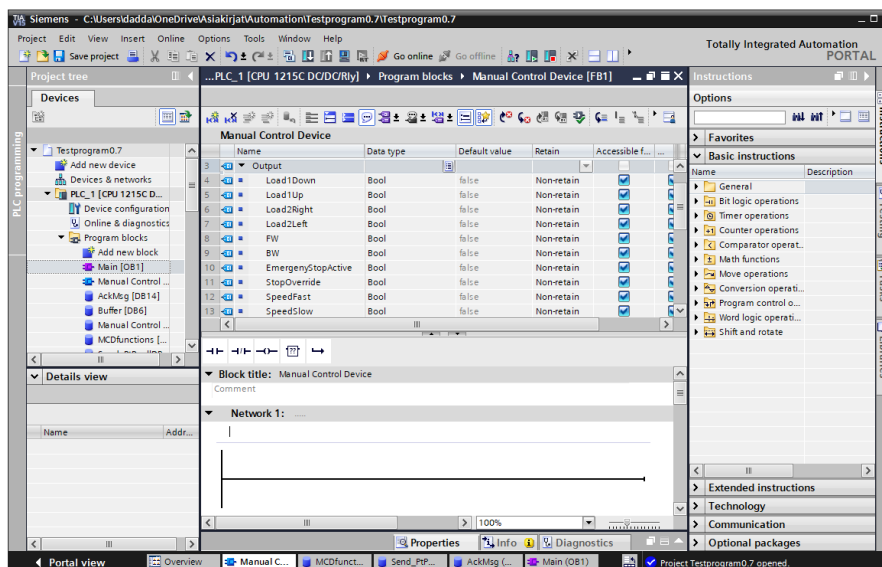
**Figur 33. Skapa ett nytt block.**



**Figur 34. Blocket i programblock filen.**

Då funktionsblocket öppnades så dök en ny ruta upp där programmeringen utfördes och på samma gång öppnades även det skilda datablocket för själva funktionsblocket där tillverkning av ingångar, utgångar och andra variabler utfördes. Här lades det till utgångar som kommer att vara synliga på det slutgiltiga blocket.

Programmeringen utfördes i nätverk med olika instruktioner inuti. De instruktioner som användes i programmeringen finns i en instruktionslista på den högra sidan i TIA Portal när arbete utförs i filen ”*Program blocks*”.



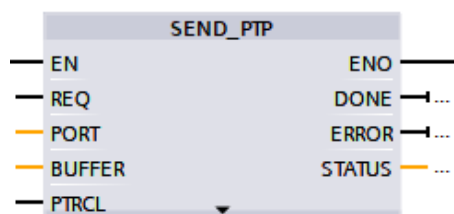
Figur 35. Vy över funktionsblockets insida.

Dessa instruktioner användes för funktionerna som MCD aktiverar, till exempel lasthanteringen, hastigheten samt kommunikationen mellan MCD och PLC.

För programmeringen av själva kommunikationen användes instruktionerna *SEND\_PTP* och *RCV\_PTP* som fanns färdigt i instruktionslistan under *Communication*.

## SEND\_PTP

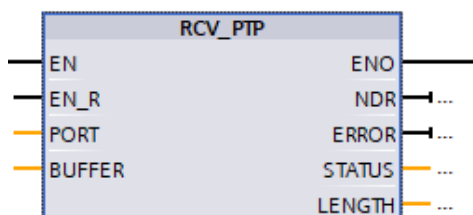
Instruktionen *SEND\_PTP* i TIA Portalen används för att kunna skicka punkt-till-punkt meddelanden. Instruktionen används för att starta en överföring av data, men den utför inte själva överföringen av data. Den data som man vill skicka skrivs in i bufferten av instruktionen, data i bufferten skickas sedan till kommunikationsmodulen till exempel CM1241. Kommunikationsmodulen i sin tur tar hand den data som fås från programmets instruktion (Siemens, 2013)



Figur 36. Instruktionen SEND\_PTP.

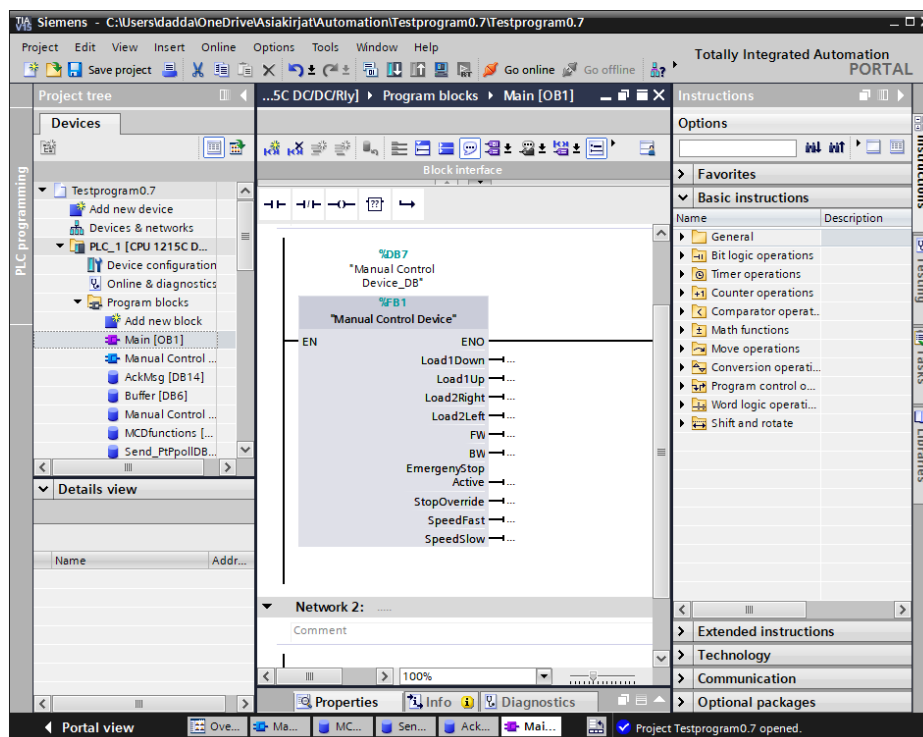
## RCV\_PTP

Instruktionen RCV\_PTP kan användas för mottagning av datameddelanden. Den fungerar så att den inspekterar om det finns något meddelande mottaget i en kommunikationsmodul, till exempel CM1241. Om ett meddelande är tillgängligt så blir det överfört från kommunikationsmodulen till PLC:n. Med denna instruktion så tillåts mottagningen av ett skickat meddelande. (Siemens, 2013)



Figur 37. Instruktionen RCV\_PTP.

Själva programmet innehållande instruktioner som utför MCD-funktioner fanns nu inom ett funktionsblock som kan testas genom att dra in det till ett organisationsblock. Detta block består endast av utgångar och dessa utgångar är spakarna och knappen som finns på MCD:n.



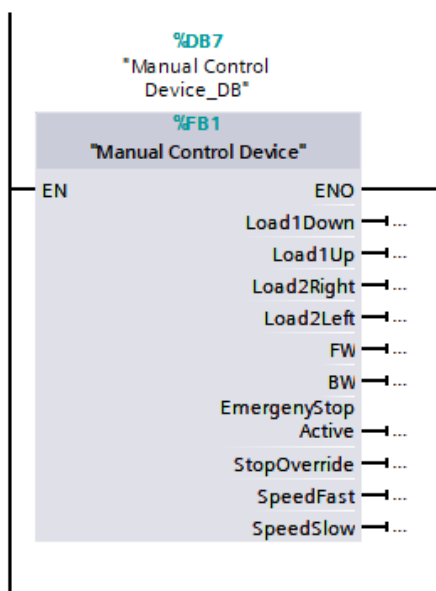
Figur 38. Funktionsblock i organisationsblock.

## 8 Resultat

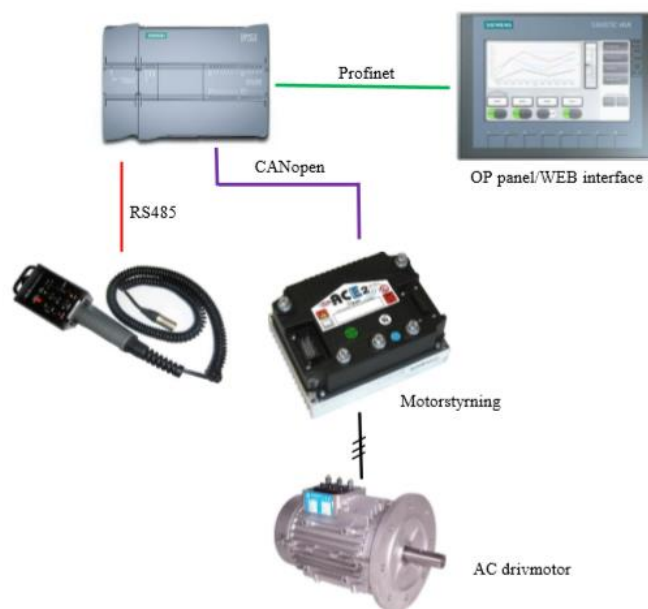
Målet med detta arbete var ett dokumenterat programblock där man skulle kunna styra en PLC med ett handmanöverdon för att sedan kunna anpassa detta till en AGV så att det i framtiden skall gå att ta fram flera Siemens maskinvaruanpassade truckar. Att detta görs genom ett RS485 gränssnitt har sina fördelar när det inte finns mycket utrymme i en truck för extra maskinvara och när detta då använder endast två trådar och alla av handmanöverdonets ingångar styrs genom datameddelanden som skickas i dessa trådar.

Det som projektet resulterade i var ett dokumenterat funktionsblock som talar med MCD och avkodar datameddelanden som sedan styr olika utgångar som då kan anpassas till motorer och annat tillbehör som truckarna använder sig av. Alla funktioner blev inte färdiga men en fungerande kommunikation mellan PLC och MCD kunde tas fram och det bevisade att detta kan gå att utföras till fullo. De datameddelanden som kommer in till CM1241 från MCD far vidare till PLC. Där läses meddelanden av och aktiverar önskad utgång på funktionsblocket.

Det som ännu saknas i funktionsblocket är själva styrningen via potentiometern och ingångar för nödstopp, mjukstopp och LED-lamporna *ESTOP* samt *ON TRACK*. Detta funktionsblock kommer att vidare utvecklas för att kunna användas på en testtruck i framtiden.



Figur 39. FB1 Manual Control Device.



Figur 40. Helhetsbild för fungerande testtruck.

## 9 Diskussion

Examensarbetet har varit en intressant utmaning att utföra i tanke med att man började i princip från noll med en väldigt liten kunskap inom programmeringsverktyget TIA Portal samt även utförde loggningar mellan två enheter och såg hur de talade med varandra för att sedan försöka sig på att anpassa den ena enheten att tala med en annan sorts enhet.

Det var ett tidskrävande arbete på grund av att ingen visste i början av projektet hur kommunikationen skulle se ut, vilket protokoll det använde samt hur själva datameddelanden var uppbyggt och vad som faktiskt var vad i dessa meddelanden, så en stor del av tiden användes till undersökningen av detta för att ta reda på en lång rad med loggningsdata. Det kan även hända att det inte skulle alls blivit till något om man ej hade fått hjälp från de som hade tillverkat själva handmanöverdonet. De bistod oss med ett papper som innehåll information av deras protokoll och visade hur deras kommunikation fungerade och hur det var uppbyggt och hur man skulle tolka de datameddelanden som skickades.

När man hade fått en förståelse över hur de kommunicerade och hittat vilka meddelanden som hör ihop med sin funktion var det att börja försöka utveckla programmet i TIA Portal. För att få en början på TIA Portal så blev det mycket läsande på de manualer som Siemens bidrar med och forskning på Siemens hemsida och forumet hur man kan utföra olika kommunikationsprogram och förslag till dessa samt då även filmer på Youtube. Sen kunde man börja med att tillverka och köra ett enkelt testprogram i TIA Portal för att få en bättre förståelse över programmeringsverktyget.

Efter allt detta märkte jag själv vilken stor nytta och lärdom man får utav ett arbete i den här kalibern, att söka efter information, se hur en kommunikation kan vara uppbyggd, tolka och bena ut hur meddelanden endast bestående av bytes kan fungera, anpassa detta till ett programmeringsverktyg som inte har någon alls koppling med den typ av applikation som lades till och kunskapen inom TIA Portal som utvecklades ser jag endast som goda kunskaper att ha i arbetslivet inom automation.

Det som skulle rekommenderas är att programmera i ett annat programmeringsspråk. LAD är inte det optimalaste programmeringsspråket när man utför programmering där man skall hantera datameddelanden som skickas över ett seriellt gränssnitt. Programmeringsspråket som skulle ha varit mera optimalt och föredra för denna typ av hantering är SCL.



Tanken var även att testa detta på en mini-truck som Solving har byggt upp så arbetet med att få en fungerande kod och alla av handmanöverdonets funktioner i programblocket fortsätter och tanken är att blocket skall fungera till fullo inom kort.

## 10 Litteraturförteckning

Automation24, 2019. *Siemens CM 1241 RS422/485 - 6ES7241-1CH32-0XB0*. [Online]  
Available at: <https://www.automation24.biz/siemens-cm-1241-rs422-485-6es7241-1ch32-0xb0>  
[Använd 31 Mars 2019].

Berger, H., 2008. *Automating with STEP 7 in LAD and FBD*. u.o.:Publicis Corporate Publishing.

Berger, H., 2013. *Automating with SIMATIC S7-1200*. u.o.:Publicis MCD Verlag, Germany.

Berti, V., Björkman, M., Nordén, L.-Å. & Lindgren, A., 2012. *Datakommunikation*. u.o.:Liber.

Codrey, 2018. *What is RS232 Protocol and How it Works?*. [Online]  
Available at: [https://www.codrey.com/embedded-systems/rs232-serial-communication/#How\\_RS232\\_Communication\\_Works](https://www.codrey.com/embedded-systems/rs232-serial-communication/#How_RS232_Communication_Works)  
[Använd 31 Mars 2019].

Contec, 2019. *Serial communication standards*. [Online]  
Available at: <https://www.contec.com/support/basic-knowledge/daq-control/serial-communicatin/#anc-02>  
[Använd 14 Mars 2019].

Eltima, 2018. *COM port monitoring software*. [Online]  
Available at: <https://www.eltima.com/products/serial-port-monitor/>  
[Använd 14 Mars 2019].

FTDI chip, 2010. *Functional Description*. [Online]  
Available at:  
[https://www.ftdichip.com/Support/Documents/DataSheets/Modules/DS\\_USB-COM485-PLUS1.pdf](https://www.ftdichip.com/Support/Documents/DataSheets/Modules/DS_USB-COM485-PLUS1.pdf)  
[Använd 13 3 2019].

Hallberg, B., 2009. *Networking, A Beginner's Guide, Fifth edition*. New York: McGraw-Hill.

National Instruments, 2018. *National Instruments*. [Online]  
Available at: <http://www.ni.com/white-paper/11390/en/#toc6>  
[Använd 14 Mars 2019].

Omega, 2019. *Differences and Typical Applications*. [Online]  
Available at: <https://www.omega.co.uk/techref/das/rs-232-422-485.html>  
[Använd 31 Mars 2019].

PLC-city, 2017. *CPU 1215C*. [Online]  
Available at: <https://www.plc-city.com/shop/en/siemens-simatic-s7-1200-cpu-1215c.html>  
[Använd 31 Mars 2019].

plcdev, 2019. *How PLCs Work*. [Online]

Available at: [http://www.plcdev.com/how\\_plcs\\_work](http://www.plcdev.com/how_plcs_work)

[Använd 14 Mars 2019].

Sen, S. K., 2014. *Fieldbus and Networking in Process Automation*. u.o.:CRC Press inc.

Siemens, 2012. *Siemens TIA Portal Modules*. [Online]

Available at:

[https://w3.siemens.com/mcms/sce/en/advanced\\_training/training\\_material/tia-portal\\_modules/pages/default.aspx](https://w3.siemens.com/mcms/sce/en/advanced_training/training_material/tia-portal_modules/pages/default.aspx)

[Använd 14 Mars 2019].

Siemens, 2013. *PtP communication instructions used on distributed I/O of an S7-300/400*. [Online]

Available at:

<https://support.industry.siemens.com/cs/mdm/69124220?c=50139645195&lc=en-FI>

[Använd 14 Mars 2019].

Siemens, 2017. *Programming guideline*. [Online]

Available at:

[https://cache.industry.siemens.com/dl/files/040/90885040/att\\_915463/v1/81318674\\_Programming\\_guideline\\_DOC\\_v15\\_en.pdf](https://cache.industry.siemens.com/dl/files/040/90885040/att_915463/v1/81318674_Programming_guideline_DOC_v15_en.pdf)

[Använd 13 Mars 2019].

Siemens, 2019. *Overview of the SIMATIC S7-1200 modules*. [Online]

Available at:

<https://new.siemens.com/global/en/products/automation/systems/industrial/plc/s7-1200.html>

[Använd 14 Mars 2019].

Siemens, 2019. *The intelligent choice for your automation task: SIMATIC controllers*.

[Online]

Available at:

<https://new.siemens.com/global/en/products/automation/systems/industrial/plc.html>

[Använd 14 Mars 2019].

Solving, 2019. *Företaget*. [Online]

Available at: <https://www.solving.com/foretaget/>

[Använd 31 Mars 2019].

Sourceforge, 2012. *MU terminal*. [Online]

Available at: <https://sourceforge.net/projects/muterm/>

[Använd 18 Mars 2019].

Turesson, J., 2016. *Remote Control of an AGV*, Göteborg: CHALMERS UNIVERSITY OF TECHNOLOGY.

## **Bilagor**

Bilaga 1 Siemens s7-1200 cpu 1215 dc/dc/rly.

Bilaga 2 CM1241 RS422/485

# SIEMENS

## Data sheet

## 6ES7215-1HG40-0XB0



SIMATIC S7-1200, CPU 1215C, COMPACT CPU, DC/DC/RELAY, 2 PROFINET PORT, ONBOARD I/O: 14 DI 24V DC; 10 DO RELAY 2A, 2 AI 0-10V DC, 2 AO 0-20mA DC, POWER SUPPLY: DC 20.4 - 28.8 V DC, PROGRAM/DATA MEMORY: 100 KB

General information	
Engineering with	
• Programming package	STEP 7 V13 SP1 or higher
Display	
with display	No
Supply voltage	
Rated value (DC)	
• 24 V DC	Yes
permissible range, lower limit (DC)	20.4 V
permissible range, upper limit (DC)	28.8 V
Load voltage L+	
• Rated value (DC)	24 V
• permissible range, lower limit (DC)	5 V
• permissible range, upper limit (DC)	250 V
Input current	
Current consumption (rated value)	500 mA
Current consumption, max.	1 500 mA
Inrush current, max.	12 A; at 28.8 V DC
Encoder supply	
24 V encoder supply	
• 24 V	L+ minus 4 V DC min.
Power losses	
Power loss, typ.	12 W

Memory	
Type of memory	EEPROM
Work memory	
• Integrated	125 kbyte
• expandable	No
Load memory	
• Integrated	4 Mbyte
• Plug-in (SIMATIC Memory Card), max.	with SIMATIC memory card
Backup	
• present	Yes; maintenance-free
• without battery	Yes
CPU processing times	
for bit operations, typ.	0.085 µs; / instruction
for word operations, typ.	1.7 µs; / instruction
for floating point arithmetic, typ.	2.5 µs; / instruction
CPU-blocks	
Number of blocks (total)	DBs, FCs, FBs, counters and timers. The maximum number of addressable blocks ranges from 1 to 65535. There is no restriction, the entire working memory can be used
OB	
• Number, max.	Limited only by RAM for code
Data areas and their retentivity	
retentive data area in total (incl. times, counters, flags), max.	10 kbyte
Flag	
• Number, max.	8 kbyte; Size of bit memory address area
Process image	
• Inputs, adjustable	1 kbyte
• Outputs, adjustable	1 kbyte
Hardware configuration	
Number of modules per system, max.	3 comm. modules, 1 signal board, 8 signal modules
Time of day	
Clock	
• Hardware clock (real-time clock)	Yes
• Deviation per day, max.	+/- 60 s/month at 25 °C
• Backup time	480 h; Typical
Digital inputs	
Number of digital inputs	14; Integrated
• of which, inputs usable for technological functions	6; HSC (High Speed Counting)

integrated channels (DI)	14
m/p-reading	Yes
<b>Number of simultaneously controllable inputs</b>	
all mounting positions	
— up to 40 °C, max.	14
<b>Input voltage</b>	
• Rated value (DC)	24 V
• for signal "0"	5 V DC at 1 mA
• for signal "1"	15 VDC at 2.5 mA
<b>Input current</b>	
• for signal "1", typ.	1 mA
<b>Input delay (for rated value of input voltage)</b>	
for standard inputs	
— Parameterizable	Yes; 0.2 ms, 0.4 ms, 0.8 ms, 1.6 ms, 3.2 ms, 6.4 ms and 12.8 ms, selectable in groups of four
— at "0" to "1", min.	0.2 ms
— at "0" to "1", max.	12.8 ms
for interrupt inputs	
— Parameterizable	Yes
for counter/technological functions	
— Parameterizable	Yes; Single phase : 3 at 100 kHz & 3 at 30 kHz, differential: 3 at 80 kHz & 3 at 30 kHz
<b>Cable length</b>	
• shielded, max.	500 m; 50 m for technological functions
• Unshielded, max.	300 m; For technological functions: No
<b>Digital outputs</b>	
Number of digital outputs	10; Relays
integrated channels (DO)	10
short-circuit protection	No; to be provided externally
<b>Switching capacity of the outputs</b>	
• with resistive load, max.	2 A
• on lamp load, max.	30 W with DC, 200 W with AC
<b>Output delay with resistive load</b>	
• "0" to "1", max.	10 ms; max.
• "1" to "0", max.	10 ms; max.
<b>Switching frequency</b>	
• of the pulse outputs, with resistive load, max.	1 Hz
<b>Relay outputs</b>	
• Number of relay outputs, integrated	10
• Number of relay outputs	10
• Number of operating cycles, max.	mechanically 10 million, at rated load voltage 100,000
<b>Cable length</b>	

• shielded, max.	500 m
• Unshielded, max.	150 m
<b>Analog inputs</b>	
Number of analog inputs	2
Integrated channels (AI)	2; 0 to 10 V
<b>Input ranges</b>	
• Voltage	Yes
<b>Input ranges (rated values), voltages</b>	
• 0 to +10 V	Yes
• Input resistance (0 to 10 V)	≥100k ohms
<b>Cable length</b>	
• shielded, max.	100 m; twisted and shielded
<b>Analog outputs</b>	
Number of analog outputs	2
Integrated channels (AO)	2; 0 to 20 mA
<b>Output ranges, current</b>	
• 0 to 20 mA	Yes
<b>Cable length</b>	
• shielded, max.	100 m; shielded, twisted pair
<b>Analog value creation</b>	
<b>Integration and conversion time/resolution per channel</b>	
• Resolution with overrange (bit including sign), max.	10 bit
• Integration time, parameterizable	Yes
• Conversion time (per channel)	625 µs
<b>Encoder</b>	
<b>Connectable encoders</b>	
• 2-wire sensor	Yes
<b>1st interface</b>	
Interface type	PROFINET
Physics	Ethernet
Isolated	Yes
Automatic detection of transmission speed	Yes
Autonegotiation	Yes
Autocrossing	Yes
<b>Functionality</b>	
• PROFINET IO Device	Yes
• PROFINET IO Controller	Yes
<b>PROFINET IO Controller</b>	
• Transmission rate, max.	100 Mbit/s



• Number of connectable IO devices, max.	16
• Prioritized startup	
— Number of IO Devices, max.	16
<b>PROFINET IO Device</b>	
<b>Services</b>	
— Shared device	Yes
— Number of IO controllers with shared device, max.	2
<b>Communication functions</b>	
<b>S7 communication</b>	
• supported	Yes
• as server	Yes
• As client	Yes
<b>Open IE communication</b>	
• TCP/IP	Yes
• ISO-on-TCP (RFC1006)	Yes
• UDP	Yes
<b>Web server</b>	
• supported	Yes
• User-defined websites	Yes
<b>Number of connections</b>	
• overall	16; dynamically
<b>Test commissioning functions</b>	
<b>Status/control</b>	
• Status/control variable	Yes
• Variables	Inputs/outputs, memory bits, DBs, distributed I/Os, timers, counters
<b>Forcing</b>	
• Forcing	Yes
<b>Diagnostic buffer</b>	
• present	Yes
<b>Traces</b>	
• Number of configurable Traces	2; Up to 512 KB of data per trace are possible
<b>Integrated Functions</b>	
Number of counters	6
Counter frequency (counter) max.	100 kHz
Frequency meter	Yes
controlled positioning	Yes
PID controller	Yes
Number of alarm inputs	4
<b>Galvanic isolation</b>	

<b>Galvanic isolation digital inputs</b>	
• Galvanic isolation digital inputs	500V AC for 1 minute
• between the channels, in groups of	1
<b>Galvanic isolation digital outputs</b>	
• Galvanic isolation digital outputs	Relays
• between the channels	No
• between the channels, in groups of	2
<b>Permissible potential difference</b>	
between different circuits	500 V DC between 24 V DC and 5 V DC
<b>EMC</b>	
<b>Interference immunity against discharge of static electricity</b>	
• Interference immunity against discharge of static electricity acc. to IEC 61000-4-2	Yes
— Test voltage at air discharge	8 kV
— Test voltage at contact discharge	6 kV
<b>Interference immunity to cable-borne interference</b>	
• Interference immunity on supply lines acc. to IEC 61000-4-4	Yes
• Interference immunity on signal lines acc. to IEC 61000-4-4	Yes
<b>Surge immunity</b>	
• on the supply lines acc. to IEC 61000-4-5	Yes
<b>Immunity against conducted interference induced by high-frequency fields</b>	
• Interference immunity against high-frequency radiation acc. to IEC 61000-4-6	Yes
<b>Emission of radio interference acc. to EN 55 011</b>	
• Limit class A, for use in industrial areas	Yes; Group 1
• Limit class B, for use in residential areas	Yes; When appropriate measures are used to ensure compliance with the limits for Class B according to EN 55011
<b>Degree and class of protection</b>	
Degree of protection to EN 60529	
• IP20	Yes
<b>Standards, approvals, certificates</b>	
CE mark	Yes
UL approval	Yes
cULus	Yes
RCM (formerly C-TICK)	Yes
FM approval	Yes
<b>Marine approval</b>	
• Marine approval	Yes
<b>Ambient conditions</b>	

<b>Free fall</b>	
• Drop height, max. (in packaging)	0.3 m; five times, in dispatch package
<b>Ambient temperature in operation</b>	
• Min.	-20 °C
• max.	60 °C; Number of simultaneously activated inputs or outputs 7 or 5 (no adjacent points) at 60 °C horizontal or 50 °C vertical, 14 or 10 at 55 °C horizontal or 45 °C vertical
• horizontal installation, min.	-20 °C
• horizontal installation, max.	60 °C
• vertical installation, min.	-20 °C
• vertical installation, max.	50 °C
<b>Storage/transport temperature</b>	
• Min.	-40 °C
• max.	70 °C
<b>Air pressure acc. to IEC 60068-2-13</b>	
• Storage/transport, min.	660 hPa
• Storage/transport, max.	1 080 hPa
• Permissible operating height	-1000 to 2000 m
<b>Relative humidity</b>	
• Operation, max.	95 %; no condensation
• Permissible range (without condensation) at 25 °C	95 %
<b>Vibrations</b>	
• Vibrations	2G wall mounting, 1G DIN rail
• Operation, checked according to IEC 60068-2-6	Yes
<b>Shock test</b>	
• checked according to IEC 60068-2-27	Yes; IEC 68, Part 2-27 half-sine: strength of the shock 15 g (peak value), duration 11 ms
<b>Pollutant concentrations</b>	
— SO2 at RH < 60% without condensation	SO2: < 0.5 ppm; H2S: < 0.1 ppm; RH < 60% condensation-free
<b>programming</b>	
<b>Programming language</b>	
— LAD	Yes
— FBD	Yes
— SCL	Yes
<b>Cycle time monitoring</b>	
• can be set	Yes
<b>Dimensions</b>	
Width	130 mm
Height	100 mm
Depth	75 mm

Weights	
Weight, approx.	585 g
<b>last modified:</b>	12.03.2015

# SIEMENS

## Data sheet

## 6ES7241-1CH32-0XB0



SIMATIC S7-1200, COMMUNICATION MODULE CM 1241,  
RS422/485, 9 PIN SUB D (FEMALE) SUPPORTS MESSAGE  
BASED FREEPORT

Supply voltage	
Rated value (DC)	
• 24 V DC	Yes
permissible range, lower limit (DC)	20.4 V
permissible range, upper limit (DC)	28.8 V
Power losses	
Power loss, typ.	1.2 W
Interfaces	
Number of interfaces	1
Interface physics, RS 422/RS 485 (X.27)	Yes
Point-to-point	
• Cable length, max.	1 000 m
Integrated protocol driver	
— ASCII	Yes; Available as library function
— USS	Yes; Available as library function
Ambient conditions	
Free fall	
• Drop height, max. (in packaging)	0.3 m; five times, in dispatch package
Ambient temperature in operation	
• Permissible temperature range	-20 °C to +60 °C horizontal mounting, -20 °C to 50 °C vertical mounting, 95% humidity, non-condensing
• Permissible temperature change	5°C to 55°C, 3°C / minute
Ambient temperature during storage/transportation	
• Min.	-40 °C

• max.	70 °C
<b>Air pressure acc. to IEC 60068-2-13</b>	
• Operation, min.	795 hPa
• Operation, max.	1 080 hPa
• Storage/transport, min.	660 hPa
• Storage/transport, max.	1 080 hPa
<b>Relative humidity</b>	
• Permissible range (without condensation) at 25 °C	95 %
<b>Software</b>	
<b>Runtime software</b>	
<b>Target system</b>	
— S7-1200	Yes
<b>Dimensions</b>	
Width	30 mm
Height	100 mm
Depth	75 mm
<b>Weights</b>	
Weight, approx.	155 g
<b>last modified:</b>	20.04.2015