



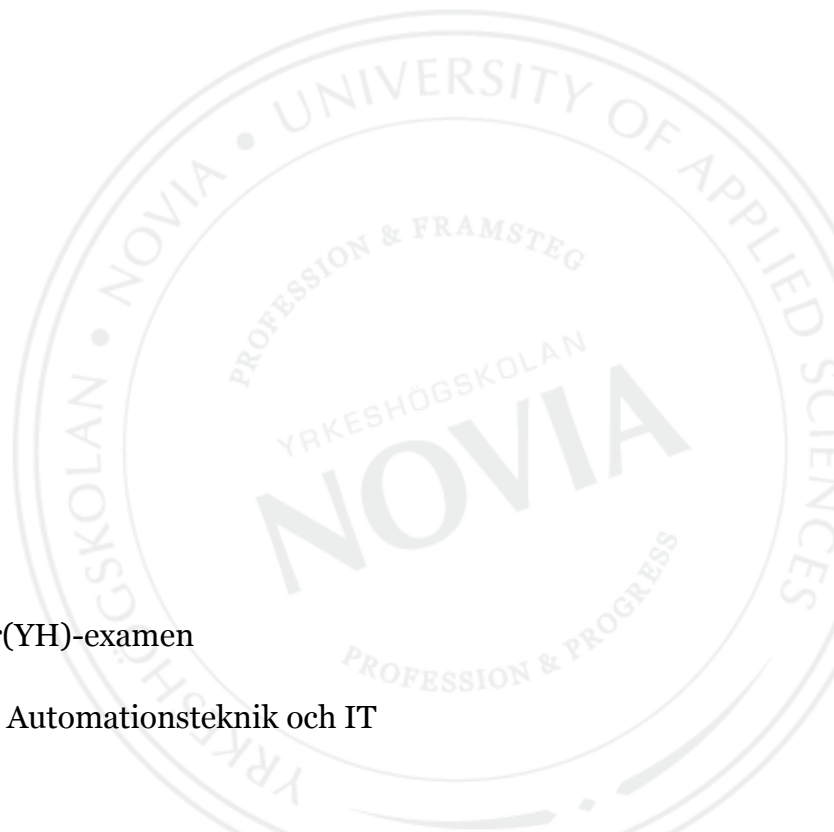
Modernisering av en droptester

Johan Lönnfors

Examensarbete för Ingenjör(YH)-examen

Utbildningsprogrammet för Automationsteknik och IT

Raseborg 2016



EXAMENSARBETE

Författare: Johan Lönnfors

Utbildningsprogram och ort: Automationsteknik och IT, Raseborg

Inriktning/alternativ/Fördjupning: Datorstödd tillverkning

Handledare: Håkan Bjurström

Titel: Modernisering av en droptester

Datum: 11.5.2016

Sidantal: 29

Bilagor: 1

Abstrakt

Detta examensarbete handlar om programmeringen av en programmerbar logik samt en pekskärm till en droptester som används till att testa hållbarheten av ytmonterade komponenter på kretskort.

Dessutom behandlas programmerbara logikers historia, uppbyggnad, val samt deras vanligaste programspråk. Ytterligare behandlas i detta examensarbete industriell Ethernets historia, utrustning, för- och nackdelar samt allmänt hur den används i industriella applikationer. Den Ethernet-baserade fältbussen EtherCAT:s funktionsprincip och fördelar behandlas också.

Jedec standarden JESD22-B111 definierar och beskriver hur hållbarheten på ytmonterade komponenter på kretskort skall testas. Denna standard används för testning av elektronikprodukter som ryms i handen avsedda för konsumentmarknaden. Examensarbetet omfattar hur ett test skall göras enligt denna standard samt vad testrapporten skall innehålla.

Språk: Svenska

Nyckelord: PLC, IEC 61131-3, Jedec, Ethernet

OPINNÄYTETYÖ

Tekijä: Johan Lönnfors

Koulutusohjelma ja paikkakunta: Automationsteknik och IT, Raasepori

Suuntautumisvaihtoehto/Syventävät opinnot: Tietokoneavusteinen tuotanto

Ohjaaja: Håkan Bjurström

Nimike: Drop-testerin modernisointi

Päivämäärä: 11.5.2016

Sivumäärä: 29

Liitteet: 1

Tiivistelmä

Opinnäytetyössä perehdytään ohjelmoitavan logiikan sekä kosketusnäytön ohjelmointiin drop-testerille, jota käytetään piirilevyille asennettujen pintaliitoskomponenttien kestävyystesteihin.

Lisäksi opinnäytetyössä käsitellään ohjelmoitavien logiikkojen historiaa, rakennetta sekä niiden tavallisimpia ohjelmointikieliä. Sen ohella käsitellään teollisuus-Ethernetin historiaa, varustusta, hyötyjä ja haittoja sekä niiden yleistä käyttöä teollisuudessa. Ethernet-pohjaista EtherCAT-kenttäväylän toimintaperiaatetta ja hyötyjä käsitellään myös.

Jedec standardi JESD22-B111 määrittelee ja kertoo, kuinka piirilevyille asennettujen pintaliitoskomponenttien kestävyyttä testataan. Tätä standardia käytetään kun kannettavien kuluttajalaitteiden piirilevyjen komponenttien kestävyyttä halutaan testata. Opinnäytetyö käsittelee, kuinka testi tehdään standardin mukaisesti ja mitä testiraportin pitäisi sisältää.

Kieli: Ruotsi

Avainsanat: PLC, IEC 61131-3, Jedec, Ethernet

BACHELOR'S THESIS

Author: Johan Lönnfors

Degree Programme: Automation Engineering and IT, Raseborg

Specialization: Design and Manufacturing

Supervisor: Håkan Bjurström

Title: Modernization of a Droptester

Date: 11 May 2016 Number of pages: 29 Appendices: 1

Summary

This thesis is about programming a programmable logic controller and a touchscreen used in a droptester. A droptester is used for testing the durability of surface mounted components on printed circuit boards (PCB).

Additionally this thesis covers the history, structure and selection of programmable logic controllers as well as their programming languages. The history of the industrial Ethernet, its devices, benefits, drawbacks and common use in the industry are also dealt with. The operating principle and benefits of the Ethernet-based fieldbus EtherCAT are also brought up.

The Jedec standard JESD22-B111 defines and describes how the durability of surface mounted components on PCB: s used in handheld consumer electronic products shall be tested. This thesis describes how a test is conducted according to this standard and also what should be included in the test report.

Language: Swedish Key words: PLC, IEC 61131-3, Jedec, Ethernet

Innehållsförteckning

1. Inledning.....	1
2. Teori	1
2.1. Programmerbara logiker.....	1
2.1.1. Uppbyggnad	2
2.1.2. Historia.....	2
2.1.3. Val av PLC	3
2.2. Programspråk	4
2.2.1. Ladderdiagram	4
2.2.2. Funktionsblockdiagram.....	5
2.2.3. Instruktionslista	5
2.2.4. Strukturerad text.....	6
2.2.5. Sekventiella funktionsdiagram.....	7
2.3. Industriell Ethernet.....	8
2.3.1. Historia.....	8
2.3.2. Utrustning.....	9
2.3.3. Nackdelar	10
2.3.4. Ethernet och tillverkare	10
2.4. EtherCAT	10
2.4.1. Funktionsprincip.....	11
2.4.2. Fördelar	11
2.5. Jedic JESD22-B111	12
2.5.1. Bakgrund	12
2.5.2. Testmetod.....	12

2.5.3.	Komponenterna och kretskortet	13
2.5.4.	Uppbyggnad	13
2.5.5.	Förberedelser inför test.....	14
2.5.6.	Testet	16
2.5.7.	Testrapport	16
3.	Droptester	17
3.1.	Funktionsprincip.....	17
3.2.	Komponenter.....	17
3.3.	Användning	19
3.4.	Moderniseringens syfte	20
3.5.	Nya Systemet.....	21
4.	Programmering.....	21
4.1.	PLC.....	21
4.2.	Pekskärm	22
4.2.1.	”Main”-fönster	23
4.2.2.	”Setup”-fönster.....	24
5.	Slutsats	25
6.	Källförteckning.....	26
Bilagor	Fel! Bokmärket är inte definierat.	

Bilagor

Bilaga 1

Programmet till droptestern utskrivet i datorprogrammet Sysmac Studio.

1. Inledning

Uppdragsgivare var företaget Kasatek Oy i Svartå, Raseborg. Företaget grundades år 2013 och erbjuder elplanerings-, elinstallation-, programmering- och ibruktagningstjänster av automationsanordningar till alla industrier. Uppdraget var att programmera den programmerbara logiken och pekskärmen till en droptester då den skulle genomgå en modernisering. Droptestern är en produkt som företaget Salon Teknopaja Oy säljer. Den används till att testa hållbarheten på ytmonterade komponenter på kretskort.

2. Teori

2.1. Programmerbara logiker

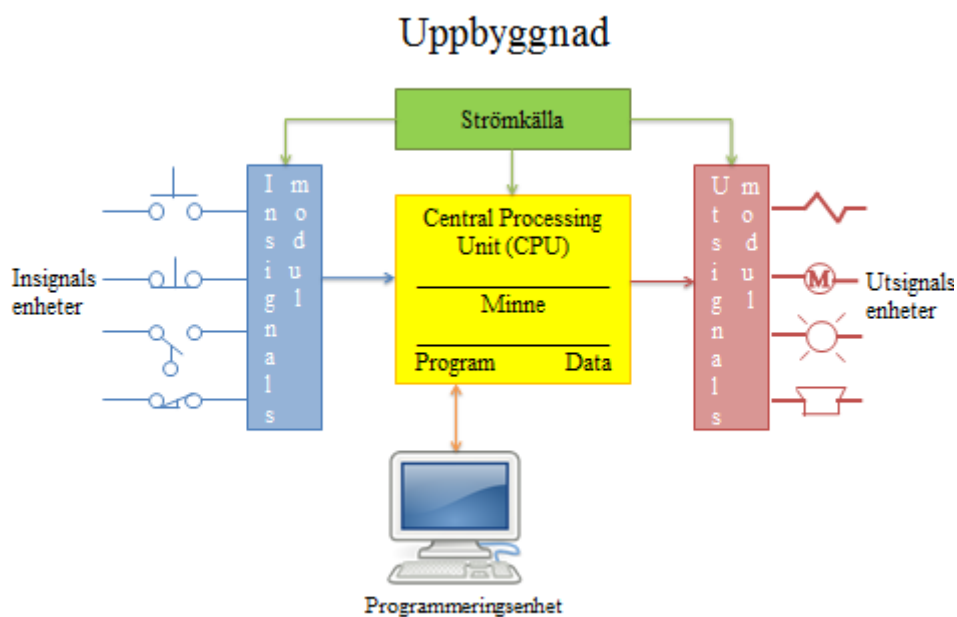
En programmerbar logik (förkortning PLC) är ett industriellt datorstyrsystem som följer med tillståndet av ingångarna och fattar beslut om utgångarnas tillstånd på basen av ett anpassat program. En programmerbar logik är modulär så den lämpar sig för diverse konfigurationer. Den tål också hetta, köld, damm och hög luftfuktighet.



Figur 1. Bild på en Omron NX701-1700 programmerbar logik med strömkälla.

2.1.1. Uppbyggnad

En PLC består av en strömkälla, en CPU, en insignalsmodul samt en utsignalsmodul (se figur 2). En CPU (Central Processing Unit) fungerar som hjärnan i en PLC. Den kan vara en 16 bitars eller 32 bitars mikroprocessor som består av en minneskrets och integrerade kretsar för styrlogiken, övervakning, samt kommunikation. Minnet förser lagringsutrymme åt operativsystemets data som används av CPU: n. I systemets ROM-minne (Read-Only Memory) sparas operativsystemets data permanent. En PLC kräver en programmeringsenhet, t.ex. en dator och programmet sparas i dess RAM-minne (Random Access Memory). Programmet består av information om insignals- och utsignalsenheterna samt värden för timers, räknare samt interna enheter. Ofta förkortas insignal- samt utsignalsmodulen till I/O-modul. Som insignaler kan fungera t.ex. brytare eller sensorer och som utsignaler motorer eller ventiler. (1)



Figur 2. Bild på en programmerbar logiks uppbyggnad.

2.1.2. Historia

Före programmerbara logikerna fanns användes reläer till att kontrollera maskiner. Ett relä består av en spole samt ytterligare en till krets som har en strömbrytare och den kan vara öppen eller slutet. När spolen får ström skapas ett magnetfält. Kraften från magnetfältet ändrar kretsen från att vara öppen till slutet, eller tvärtom. När spolen inte längre får ström ändrar kretsen sitt läge igen.

Ett relä kan användas till att t.ex. kontrollera när en elmotor får ström. Denna sorts relä är ett kraftrelä. Det kan också behövas ett relä till som bestämmer när kraftreläet ska få ström. Detta relä kallas för styrrelä. Utöver detta kunde ännu ytterligare reläer behövas så det kunde alltså finnas många reläer för att styra en maskin. Nackdelen med detta var att alla reläer i systemet måste fungera och vara rätt kopplade. Spolarna kunde ge efter och kontakterna blev också utslitna så felsökning behövdes ofta göras. Felsökningen tog dessutom flera timmar. Dessa maskiner hade också ett strikt underhållsschema som måste följas och de tog också upp en hel del utrymme. Om man ville göra ändringar i systemet måste hela systemet designas på nytt. (2)

År 1968 vann företaget Bedford Associates ett förslag, på begäran av GM Hydramatic, att ersätta fastprogrammerade reläsystem. Produkten hette Modicon som stod för Modular Digital Controller och var den första programmerbara logiken. Ett nytt företag startades av Bedford Associates som tillägnade tid till att utveckla, tillverka och sälja denna produkt. Produktens framgång gav upphov till en global industri som har expanderat kraftigt de senaste 30 åren. (3)

2.1.3. Val av PLC

Det finns olika faktorer som lönar sig att beakta då en PLC väljs. Hårdvaran skall passa ihop oberoende om det är frågan om ett nytt eller ett gammalt system. Miljöförhållandena inverkar på dess prestanda. De flesta styrsystemen arbetar i en temperatur mellan 0 till 55 °C. Antalet separata enheter (logikanordningar som har på/av -lägen) och analoga enheter bestämmer antalet I/O anslutningar som en programmerbar logik behöver. Dessutom lönar det sig att bestämma kraven på PLC: s CPU när antalet RAM-minne som behövs för data manipulering och lagring beräknas. Om data skall lagras under en längre tid skall CPU: s minne vara tillräckligt stort. Vid fjärranslutningar brukas serie- och Ethernet-baserade I/O-moduler användas. Fjärrenheter behövs då PLC: n ligger separat. Serieanslutningar har ett max avstånd på 250 fot och Ethernetanslutningar kan vara upp till 328 fot. Till slut skall också PLC: n förstå programinstruktionerna som anges.(1)

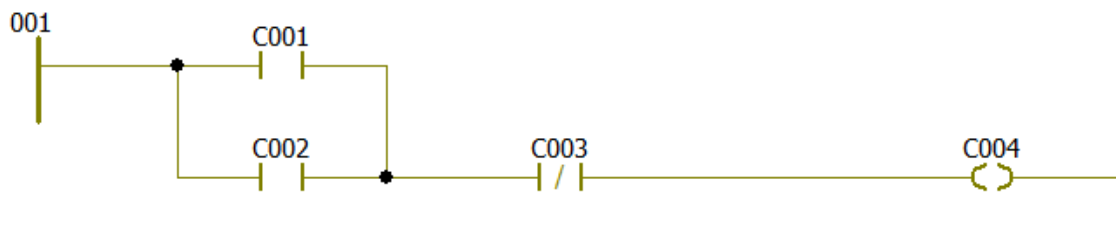
2.2. Programspråk

Organisationen IEC (International Electrotechnical Commission) är en organisation som förbereder och publicerar standarder åt alla elektriska, elektroniska samt relaterade teknologier, inklusive styrsystem. Enligt IEC 61131-3 standarden identifierar organisationen följande fem programspråk samt deras vanligaste förkortningar vid programmering av en PLC; instruktionslista(IL), ladderdiagram(LD), funktionsblockdiagram(FBD), strukturerad text(ST) och sekventiella funktionsdiagram eller sequential function chart (SFC). Det finns olika faktorer som lönar sig att beakta då man väljer programspråk. Alla programmerbara logiker, i synnerhet mikro-PLC: er, har inte tillräckligt med minne eller en tillräckligt hög processorhastighet för att klara av alla programspråk.(4)

2.2.1. Ladderdiagram

Av dessa fem används ladderdiagram mest. Visuellt sett påminner detta programspråk om en serie styrkretsar där vissa insignaler skall ändra läge så att utsignalerna aktiveras(se figur 3). Det är också enkelt att skapa ett program med detta programspråk och enkelt att ändra programkoden i ett senare skede. Nackdelen med ladderdiagram är att när programstorleken växer blir det svårare att läsa och förstå det. Vanligen krävs också funktioner som PID-regulatorer, trigonometri- och data-analyser i styrtillämpningar, men de är svåra att tillämpa i detta programspråk.

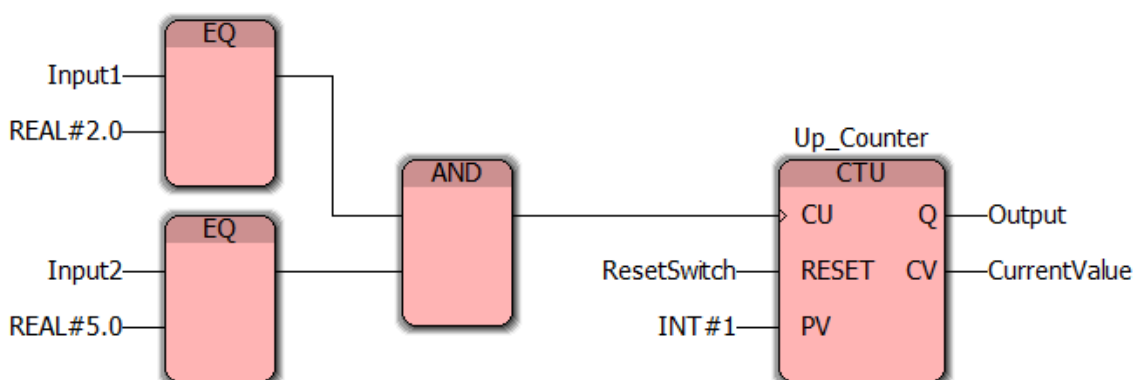
Ladderdiagram passar bäst för tillämpningar där saker ska transporteras och flyttas. Ett exempel på en sådan tillämpning är då en sensor känner av en låda, får det en utsignal att aktiveras. Detta får i sin tur får ett ställdon att röra sig som kan skuffa lådan vidare till ett transportband.



Figur 3. Exempel på ett ladderdiagram.

2.2.2. Funktionsblockdiagram

På senaste tiden har användningen av funktionsblockdiagram ökat. Funktionsblockdiagram är antagligen det till följande mest använda programspråk. Till utseendet påminner det ännu mera om ett kopplingsschema än ladderdiagram. I ett funktionsblockdiagram finns det olika block som är ihopkopplade till varandra i en programsekvens. Det är enkelt att följa med vad som händer. Detta programspråk använder sig av samma instruktioner som ladderdiagram och är också enklare att följa med för de som inte är insatta i relälogik. Funktionsblockdiagram lämpar sig bäst för enklare program som har digitala insignaler, så som fotoelektriska sensorer och som utsignaler exempelvis ventilblock. Detta programspråk lämpar sig också i tillämpningar där ladderdiagram är ideala. Funktionsblockdiagram kräver dock mycket utrymme på skärmen. Dessutom krävs förkunskaper i hur programmet löper framåt eftersom det kan vara svårare att göra korrigeringar senare.



Figur 4. Exempel på ett funktionsblockdiagram.

2.2.3. Instruktionslista

Instruktionslista består av flera rader kod och varje rad representerar en operation. Programspråket har ett steg för steg format, vilket gör att en serie med enkla matematiska funktioner går enkelt att införa. Dessutom kan programmet också flyttas till andra hårdvaruplattformar om instruktioner definierade av IEC-standarden används. Instruktionslista har en snabb utförandehastighet då det inte är ett grafiskt programspråk men p.g.a. detta är det också svårare att förstå vad programmet gör. Programspråket är mycket kompakt och tar upp mindre utrymme på en PLC: s minne. Instruktionslista har samma nackdelar som programspråket ladderdiagram. Då programmet växer blir det

svårare att läsa och förstå det, samt funktioner så som PID och matematiska funktioner är svåra att tillämpa. Det är också tvivelaktigt om dess fördelar i hastighet och storlek är av betydelse med tanke på bearbetningshastigheten och mängden av minne som moderna programmerbara logiker har.

```
1  LD  Input1
2  EQ  REAL#2.0
3  ST  In1
4  LD  Input2
5  EQ  REAL#5.0
6  ST  In2
7  LD  In1
8  AND In2
9  ST  Out
10 LD  Out
11 ST  Counter_Up.CU
12 LD  ResetSwitch
13 ST  Counter_Up.RESET
14 LD  INT#1
15 ST  Counter_Up.PV
16 CAL Counter_Up
17 LD  Counter_Up.Q
18 ST  Output
19 LD  Counter_Up.CV
20 ST  CurrentValue
```

Figur 5. Exempel på en instruktionslista.

2.2.4. Strukturerad text

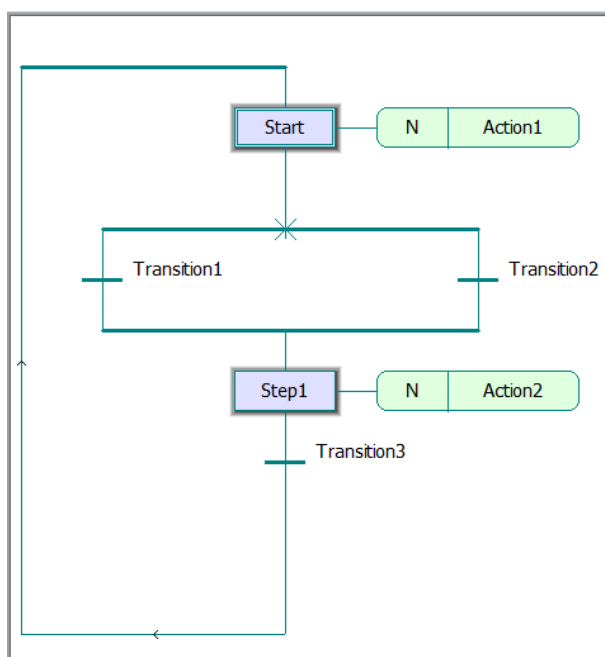
Strukturerad text påminner om programspråken PASCAL och C. Fördelen med detta programspråk är att det går att genomföra data-analyser och andra beräkningar enklare än med ladderdiagram eller instruktionslista. Strukturerad text tar också upp lite plats och har en hög utförandehastighet då det är textbaserat. De flesta program har en inbyggd textredigerare, som gör att det går enkelt att lägga till kommentarer, indrag eller radavstånd för att betona vissa delar av programmet. Det är också möjligt att kopiera och använda texten i andra programmeringsprogram, oftast med endast små ändringar. Detta programspråk är dock inte det bästa när det kommer till felsökning. Användningen av strukturerad text tenderar vara att den innesluts i en programinstruktion där den sedan kallas på. IEC 61131-standarden tillåter användningen av flera än ett programspråk samtidigt. Funktioner som skapas i ett programspråk kan sedan användas i ett annat programspråk.

```
1  In1 := Input1 = REAL#2.0;
2  In2 := Input2 = REAL#5.0;
3
4  Out := In1 & In2;
5
6  Up_Counter(CU:= Out,RESET:= ResetSwitch ,PV:= INT#1);
7  Output :=Up_Counter.Q;
8  CurrentValue :=Up_Counter.CV;
9
```

Figur 6. Exempel på ett program i Strukturerad Text.

2.2.5. Sekventiella funktionsdiagram

Sequential Function Chart eller sekventiella funktionsdiagram påminner om flödesscheman och dess koncept är enkelt: En låda som innehåller programkod är aktivt tills det efterföljande övergångsvillkoret aktiveras. Övergångsvillkoren innehåller programkod med utsatta villkor som måste uppfyllas för att programmet ska gå vidare. Lådan avaktiveras och nästa låda i sekvensen aktiveras. Programmet börjar med en inledande låda och efter den följer en serie med övergångsvillkor och ytterligare lådor. Detta programspråk lämpar sig bäst för applikationer som har processer som upprepas, t.ex. en produkt som plockas upp från ett område och flyttas till ett annat område jämt och ständigt. Felsökning är enkelt då funktionsdiagrammen är visuella till sin karaktär. Nackdelarna är att detta programspråk lämpar sig inte för alla applikationer. Dessutom går det mängder med tid åt till planering före programmeringen kan göras, annars kan funktionsdiagrammen bli besvärliga att följa. Det går inte heller alls att konvertera detta programspråk till något annat, medan ladderdiagram, strukturerad text, funktionsblockdiagram och instruktionslista går enkelt att konvertera. Om detta programspråk används är det viktigt att hårdvaran har den hastighet och mängd minne som krävs. (4)



Figur 7. Exempel på ett flödesdiagram.

2.3. Industriell Ethernet

Ethernet-baserade nätverk används bl.a. hemma och i kontorsmiljöer till att koppla ihop printrar och datorer. Standarden IEEE 802.3 fungerar som grund för Ethernet. Inom industrin används Ethernet också. Fördelarna är snabbhet, lätthet att designa och bygga, samt låg kostnad för hårdvaran som behövs. Dessa faktorer har också gjort dem populära inom industriella applikationer. Det går enkelt att lägga till eller ta bort noder och felsökning är relativt enkelt att göra. Då Ethernet och relaterade teknologier har blivit mera allmänt förekommande, har också antalet av utbildad personal ökat. (5) Ett Ethernet-nätverk består av noder som skickar data paket åt varandra. Varje nod har en Media Access Control (MAC) adress som specificerar målet och källan för varje datapaket. Alla datapaket skickas till alla noder i nätverket. Noden som datapaketet har som mål, läser datapaketet som har adresserats åt den. (6)

2.3.1. Historia

Före Ethernet fanns, användes egenutvecklade och partvinnade seriekommunikationsprotokoll, t.ex. RS-232 and RS-485, till dataöverföringen i fabriksanläggningar. De hade en halv duplex kommunikation och var deterministiska. Det fanns en bestämd mängd väntetid efter att ett meddelande hade skickats från master-enheten. Tidsinställningen var förutsägbar, således deterministisk, men den var mycket

långsam. Det första industriella datanätverket, Modbus, utarbetades av Hung Yu år 1979 till Modicons programmerbara logiker. Modbus var halv duplex och synnerligen deterministisk men då den hade ett seriellt kommunikationsprotokoll hade den också en långsam dataöverföringshastighet. Med Ethernet är hastigheten mycket snabbare men tidsintervallet, alltså determinismen, för ett förväntat svar är oförutsägbart. (6)

2.3.2. Utrustning

Kablarna som används inom industriellt Ethernet i fabriksmiljö blir mycket mer utsatta än de som befinner sig i kontorsmiljö. Bl.a. är temperaturen annorlunda i fabriksmiljö. Vid relativt kall temperatur är en kabel mer känslig för stötar och själva ledningen eller uttaget kan brista. Vid ännu kallare temperaturer kan kabeln bli skör och brista redan då den böjs. Hetta kan också skada en kabel och med tiden kan hetta orsaka att kabeln blir svagare. Kabelns uttag kan t.ex. smälta då den utsätts för hetta. I fabriksmiljöer blir kablarna också utsatta för kemikalier, störningar och vibrationer. Elektriska och magnetiska störningar som kommer från motorer och högspänningsapparater kan förvränga dataöverföringar i nätverket. Vibrationer kan komma från olika processer och de kan orsaka fränkoppling eller nedbrytning av uttaget.

Av dessa orsaker kan inte Ethernet-utrustning avsedd för konsumentbruk användas till industriellt bruk. Det går att använda sådan utrustning men det är bara en tidsfråga om hur länge den skulle fungera. Kablar och kontakter avsedda för industriellt bruk kan vara förseglade och de består av starkare material. Det finns förseglade RJ45-kablar som kan antingen vara avsedda för lätt eller tungt bruk. Typiskt är att kablarna har en kraftigare låsmekanism än kablar som används för kontorsbruk. Priset på dessa kablar är också högre jämfört med kommersiella komponenter. (7)



Figur 8. Bild på ett RJ45-kontaktpar avsedda för industriellt bruk.

2.3.3. Nackdelar

Datapaketet i Ethernet-nätverk kan kollidera med varandra eller gå varierande vägar till noderna. Detta kan leda till att datapaket inte kommer fram alls eller en variation i dataöverföringstiden. I industriella styrsystem kan detta leda till hastighets- eller dataförluster varpå katastrofala följder kan uppstå. Tiden som det tar för ett datapaket att gå från en nods källa till en nods mål måste således gå att bestämma, alltså processen måste vara deterministisk. (6)

Applikationer som är tidskänsliga har tidigare hanterats genom att använda deterministiska nätverk, där noderna i nätverket är indelade i master- och slavenheter. I industriella nätverk måste även användningsnivån noggrant kontrolleras, då en nivå på över 10 % kan medföra bristfällig prestanda. Eftersom lönsamheten av att använda industriell Ethernet har ökat, har också olika metoder som förbättrar Ethernets dataöverföringsprestanda utvecklats av användare av industriell Ethernet. En metod är att använda sig av switchar eller routrar till att minimera oönskad nätverkstrafik. En annan metod är att använda nyare protokoll som har integrerat prioritering, synkronisering eller andra tekniker som garanterar en snabb leverans av meddelanden. (5)

2.3.4. Ethernet och tillverkare

Användningen av Ethernet i industriella styrsystem fortsätter att öka. De största tillverkarna av styrsystem använder egna versioner av Ethernet-nätverk och Ethernet relaterade protokoll i sina produkter. Ofta har flera tillverkare gjort gemensamma insatser för att utveckla Ethernet baserade standarder och produkter. Dock fortsätter driftskompatibiliteten mellan dem att vara dålig. Exempel på olika Ethernet-baserade fältbussar är EtherCAT, Ethernet Powerlink, Modbus-TCP och PROFINET. (5)

2.4. EtherCAT

EtherCAT är en Ethernet-baserad fältbuss. EtherCAT (Ethernet for Control Automation Technology) utvecklades av Beckhoff Automation GmbH & Co. KG.(8) EtherCAT introducerades år 2003 och blev år 2007 en öppen standard som tillhandahålls av EtherCAT Technology Group. EtherCAT och protokollet Safety over EtherCAT följer också IEC-standarderna IEC 61158 och IEC 61784.(9)

2.4.1. Funktionsprincip

Själva EtherCAT protokollet är uppbyggt med samma ramstruktur som en vanlig Ethernetram. EtherCAT baserar sig på funktionsprincipen "processing on-the-fly". Från mastersidan skickas en ram till varje nod i systemet. Då en nod läser den data som har adresserats till den läggs ny data till i ramen samtidigt. Sedan skickas datapaketet vidare till nästa nod och den sista noden i segmentet skickar ett meddelande tillbaka till mastersidan. Protokollet Safety over EtherCAT ser till att både kontroll och säkerhetsinformation överförs via ett och samma medium i kommunikationssystemet.(8)

2.4.2. Fördelar

Fördelarna med EtherCAT är bl.a. dess snabbhet samt kostnadseffektivitet. Då EtherCAT utvecklades låg fokus på korta cykeltider, låga störningssignaler och låga hårdvarukostnader.(9) Synkroniseringstider med en noggrannhet på nanosekunder kan uppnås, vilket är en enorm fördel vid tillämpningar där målsystemet styrs eller mäts via bussystemet.

Till mastersidan räcker det med ett nätverkskort som har en Ethernet-port. Mastersidan använder en Ethernet Media Access Controller (MAC) som inte kräver ytterligare kommunikationsprocessorer. Mastersidan är den ända noden inom ett segment som skickar en EtherCAT-ram medan de andra noderna endast skickar ramen vidare. Detta förhindrar att oförutsägbara fördröjningar uppstår. Den sista noden i systemet upptäcker att nästa port är öppen och skickar meddelande tillbaka till mastersidan.

Dessuom har EtherCAT inga begränsningar när det gäller nätverkstopologi (ringnät, stjärnnät, bussnät eller en kombination av dessa är möjliga). EtherCAT behöver inga switchar eller hubbar, vilket ytterligare minskar på kostnader. Det är också möjligt att ta bort eller ansluta nya enheter utan att stänga av systemet. Ett EtherCAT-nätverk kan ha stöd för upp till 65 535 enheter och två enheter kan vara på ett avstånd på 100 meter från varandra. Då fiberkabel används är även längre sträckor möjliga. (8)

2.5. Jedec JESD22-B111

Droptestern i detta examensarbete som testar komponenters hållbarhet på kretskort uppfyller bl.a. standarden Jedec JESD22-B111. Föreningen JEDEC Solid State Technology Association upprätthåller och granskar denna standard.

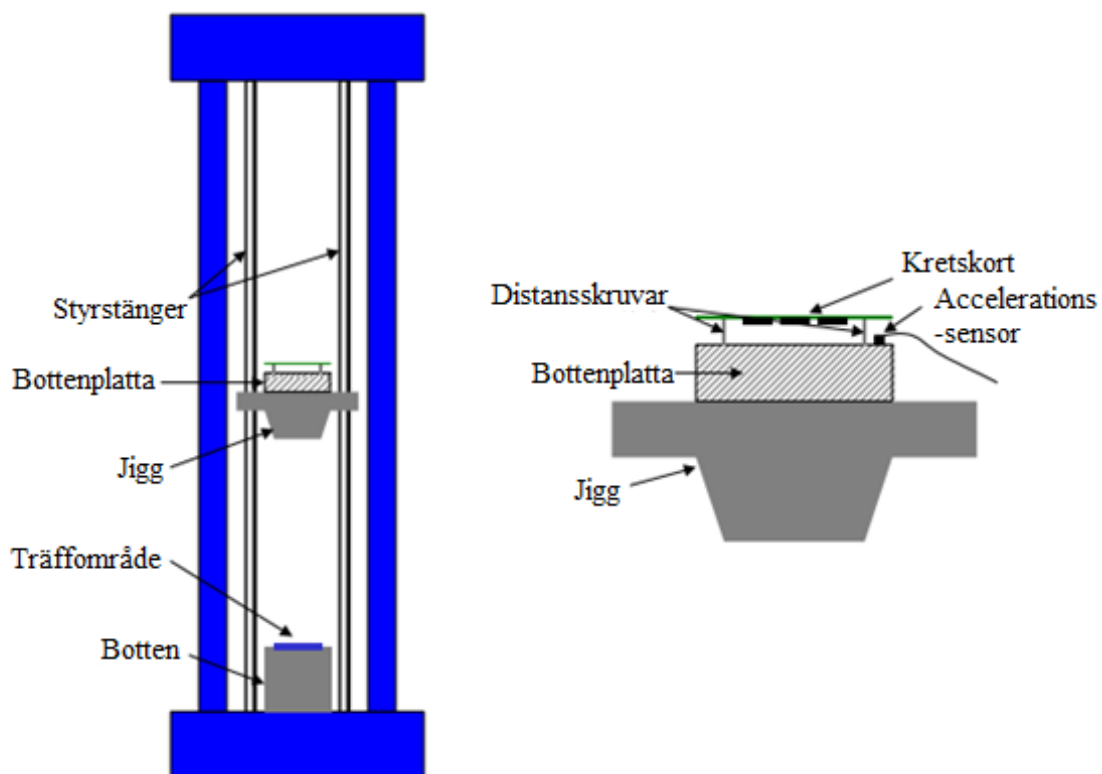
2.5.1. Bakgrund

Elektroniska produkter som ryms i fickan, t.ex. mobiltelefoner, kameror, smartkort, räknare och handdatorer, blir ofta utsatta för olika fall. Ett fall kan orsaka, förutom mekaniska fel, också elektriska fel i kretskortet som är monterat inne i produktens hölje. Detta beror på att en del av energin som uppstår i fallet överförs till kretskortets stödpunkter och vidare till kretskortet. De elektriska felen kan vara att kretskortet spricker helt eller delvis samt sprickor kan uppstå där komponenterna är fastlödda till kretskortet. Främsta orsaken till att sprickor uppstår är att kretskortet böjer sig vid fallet. Då kretskortet böjer sig skapar det en rörelse mellan kortet och dess fastmonterade komponenter som resulterar i fel i antingen komponenterna, i punkterna där de är fastkopplade eller i kretskortet. Faktorer som inverkar på felet är en kombination av kretskortets design, konstruktion, material, tjocklek och ytbehandling, samt komponenternas sammankopplingsmaterial, storlek och avstånd till kretskort.

2.5.2. Testmetod

Denna testmetod som Jedec JESD22-B111 standarden definierar, används till att jämföra och bedöma prestationsförmågan av ytmonterade elektroniska komponenter. Komponenterna används till elektroniska produkter som ryms i handen. Testmiljön består av ett accelerationstest där för mycket böjning av ett kretskort orsakar fel i produktens funktion. Standarden Jedec JESD22-B111 uppfyller inte kraven till falltest som används vid simulering av stötar som kan uppstå vid frakt och hantering av elektroniska komponenter eller kretskort. Dessa krav uppfylls av standarderna JESD22-B104-B och JESD22-B110.

diameter på 3,2 mm och en längd på 10 mm. Bottenplattan monteras fast på jiggen och på bottenplattan monteras det kretskort som önskas testas. Då bottenplattan monteras fast på jiggen skall dess tjocklek och monteringspunkter väljas så att ingen rörelse mellan jiggen eller bottenplattan uppstår under ett test.



Figur 10. Översikt av en droptesters komponenter.

2.5.5. Förberedelser inför test

Före testet skall själva testapparaten monteras på ett stadigt underlag. Utrustningen skall vara försedd med t.ex. en automatisk bromsmekanism, som hindrar jiggen från att studsa efter att jiggen har fallit ner en gång och rört träffområdet. En accelerationsmätare med en liten egenmassa kan sättas fast vid komponent U8A (se figur 8). En trådtöjningsgivare (av typen rektangulär rosett med 45 graders vinkel) skall också monteras fast under positionen U8A (den sidan på kretskortet där komponenterna inte befinner sig) för att utmärka påfrestningar i x- och y-riktningarna samt utmärka de främsta påfrestningarna och deras vinkel. Accelerationsmätaren och trådtöjningsgivaren skall båda kopplas till ett datainsamlingsystem som kan mäta med en sökningsfrekvens på 20 kHz eller högre med en signalbredd på 16 bitar. Ytterligare trådtöjningsgivare kan monteras fast på olika ställen på kretskortet för att helt karaktärisera påfrestningarna av anordningen. Sedan kan

kretskortet monteras på plats. Vid monteringen skall skruvarna spännas fast i ett diagonalt mönster och om brickor används skall deras antal per skruv vara lika. Ytterligare accelerationsmätare kan också monteras vid eller nära stödpunkterna för att försäkra att den ingående stötpulsen till bottenplattan överförs till kretskortet utan störningar. Nu kan jiggen köras till fallhöjden och sedan fällas ner mot träffområdet. Samtidigt mäts stötpulsens kraft och tid. Flera fall kan behöva göras för att nå de önskade värdena (1500 G och en stötpuls som är halvsinusformad med en tid på 0,5 millisekunder). Tiden för stötpulsen börjar mätas från att då 10 % av accelerationens högsta värde nås första gången tills 10 % av accelerationens högsta värde återstår. Justeringar i fallhöjden görs tills de önskade värdena nås. Detta kan också illustreras med nedanstående formel:

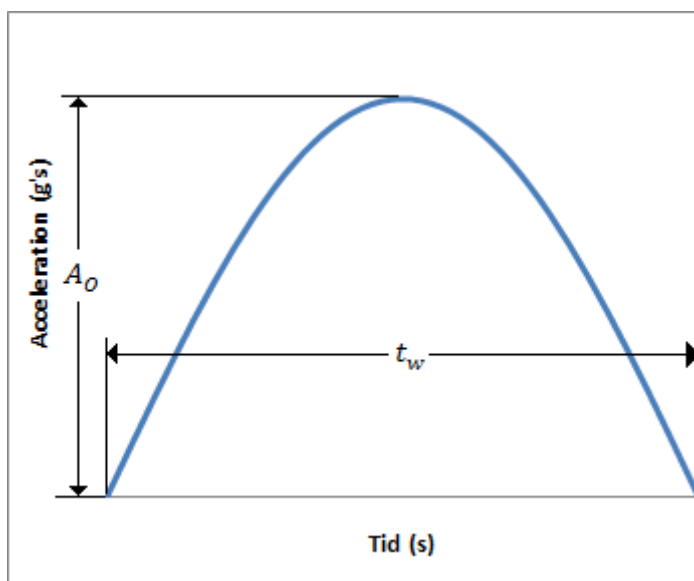
$$A(t) = A_0 \cdot \sin \frac{\pi t}{t_w}$$

Där A är accelerationen, t är tiden, A_0 är högsta uppmätta acceleration vid stötpulsen och tiden t_w är uppmätt tid för stötpulsen (se figur 11).

Följande ekvation kan användas till att beräkna den fallhöjd som behövs för att nå önskad

$$\text{acceleration: } \sqrt{2gH} = \frac{2A_0}{C} \cdot \frac{t_w}{\pi}$$

Där H är fallhöjden, C är studscoefficienten (1,0 vid ingen studsning och 2,0 vid full studsning), accelerationen A_0 och tiden t_w är uppmätta värden. Denna ekvation tar dock inte i beaktan träffytans effekt. För att nå det önskade högsta accelerationsvärde samt längd kan experiment med olika träffytor behövas göras.



Figur 11. Graf över stötpulsen.

När de specificerade parametrarna är nådda kan kretskortets svarsacceleration och belastning mätas. Belastningsnivån skall också räknas ut genom att dividera förändringen i belastningsvärde med tidsintervallet som denna förändring inträffar. Det sätt som kretskortet reagerar på (accelerationen, belastning, belastningsnivå) samt variationen skall dokumenteras och förses med testresultatet. Detta rekommenderas att göra för komponenter som inte har testats tidigare eller om dokumentering saknas för komponenter av liknande storlek.

2.5.6. Testet

Då önskat resultat har nåtts i det förberedande skedet kan kretskortsanordningarna förberedas för testning. Detta innefattar att kablarna löds fast genom de pläterade hålen på ena ändan av kretskortet, dess komponenter är vända nedåt och kablarna till datainsamlingssystemet eller en händelsedetektor kopplas fast. En händelsedetektor är ett mätinstrument som mäter elektriska avbrott i resistans när resistansen är större än 1000 ohm och med en tid på 1 μ s eller längre. Kablarna skall dessutom förses med dragavlastning. Eftersom kretskortets dynamiska respons kan påverkas av massan samt styvheten av kontakten, rekommenderas inte att någon kontakt används och att kablarna löds fast direkt till kortet. Alla kablar skall också vara ur vägen för fallets bana. Om en händelsedetektor används ska dess tröskelvärde i resistans vara inställt på högst 1000 ohm. Resistansen skall mätas före det första fallet och hela tiden under testet för varje anordning. Om en komponent delvis separeras från testkretskortet under ett test utan en förändring i resistans skall även detta betraktas som ett fel. Alla fel skall skrivas upp. Kretskortet som ska testas kan fällas högst 30 gånger. Om det har uppstått fel i 80 % av alla enheter under ett test, skall testet avbrytas. Under testet skall också stötpulsens kraft mätas för varje fall så att den hålls inom toleransvärdet. Om kraften avviker kan antingen justeringar i höjden göras eller träffytans material bytas ut.

2.5.7. Testrapport

I testets rapport skall bl.a. följande saker finnas med; Testanordningen samt kretskortets vikt, utseende, storlek, material och tjocklek. Det behövs också samma information om komponenterna. Dessutom behövs information om testets detaljer såsom fallhöjd, träffytans material och stötpulsprofil samt även testets resultat. Kretskortets

reaktion(accelerationen, belastning, belastningsnivå) skall även finnas med samt resistansen före testet inleddes. Det behövs också information om var på kretskortet ett fel har uppstått och hur många fall som krävdes för ett fel att uppstå. Till slut skall det finnas med vad för utrustning har använts till att upptäcka fel och vilka fel kriterierna var. Det skall även finnas bilder på felet. En dataanalys med medelvärdet och standardavvikelsen över felen som uppstått skall också göras. (10)

3. Droptester

3.1.Funktionsprincip

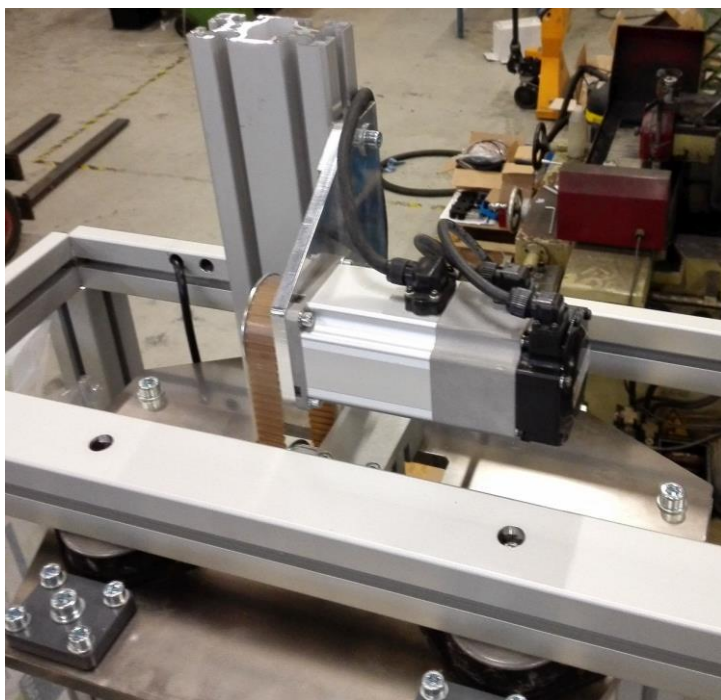
En droptester används till att testa bl.a. halvledarkomponenter, oftast ytmonterade komponenter som monterats på kretskort samt färdiga elektronikkonfigurationer. De flesta av dem används i om bärbara konsumentprodukter. De fälls från en viss höjd enligt t.ex. Jedec standarden JESD22-B111 och med en mekanisk stöt puls på upp till 5000 G. Oftast görs ett falltest med en serie på 1-1000 fall. Testobjektet skruvas fast i en jigg och det finns en änderingsbar fallvikt. Det går att ha en fallvikt på upp till 1,5 kg. Lyftkälken har en elektromagnet som lyfter jiggen. Det finns också en till elektromagnet som kan användas om man vill undvika att jiggen fortsätter att studsas upp och ner efter att den har släppts ner. Max fallhöjd är 1600 mm och apparaten har en höjd på 2500 mm, en bredd på 670 mm och en längd på 468 mm. Dess totala massa är 275 kg. Det finns också en PC som är kopplad till droptestern. Med den kan droptestern också styras med samt kalibreras i PC:ns kalibreringsprogram. Droptestern fungerar med en spänning på antingen 240 V (6 A) eller 110 V (10 A).

3.2. Komponenter

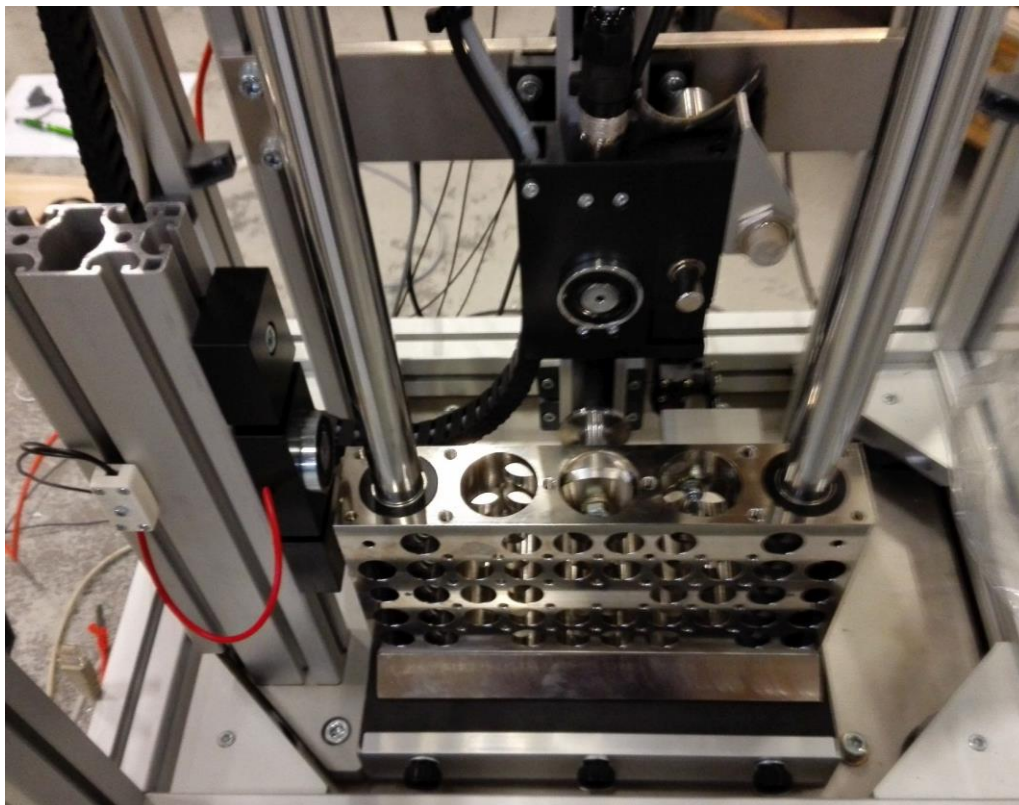
Droptestern har en PLC med en I/O-modul samt strömkälla från företaget Omron som befinner sig på baksidan. Motorn har en effekt på 0,18 kW och är en bromsmotor som befinner sig uppe på droptestern(se figur 12). Vid programmeringen användes dock en annan motor. Motorn är kopplad via en kuggrem till en remdriven linjärenhet med ett utväxlingsförhållande på 15:36. Pekskärmen som är tillverkad av Omron befinner sig på dörren (se figur 14) tillsammans med en nödstoppbrytare. Själva linjärenheten driver lyftkälken som rör sig upp och ned ända tills gränslägena kommer emot. Den har en

elektromagnet, en solenoid och en fotoelektrisk sensor (se figur 13). Magneten används till att lyfta jiggen med kretskortet, sensorn aktiveras då jiggen är tillräckligt nära och solenoiden rör sig ut och in då jiggen skall lyftas. Sensorn aktiveras då lyftkalkens magnet är tillräckligt nära jiggens lyftpunkt. Det finns även en till elektromagnet nere till vänster som skall fånga jiggen efter att den har släppts ner en gång och hindra den från att studsas.

En tredje fotoelektrisk sensor befinner sig nere och den har en spegel fast i dörren. Denna sensor är aktiv då jiggen släpps ner och når botten. Med hjälp av denna sensor får droptestern information om jiggen är nere eller om dörren är öppen, då kontakten till spegeln bryts. Droptesterns dörr är också försedd ett elektriskt lås men det saknades då programmeringen gjordes. Dessutom finns det en accelerationsmätare och en händelse-detektor som kan kopplas in vid kalibreringen.



Figur 12. Bild på motorn.



Figur 13. Bild på lyftkälken och jiggen.

3.3. Användning

Till att börja med trycker man på strömknappen på pekskärmen. Apparats servomotor söker först efter sin position, då ett strömavbrott gör att servomotorn inte längre vet var den befinner sig. Droptestern kör sedan ända upp tills övre gränsen kommer emot och därefter till den senaste valda monteringshöjden. En ny monteringshöjd går att välja. Då man trycker på ”Setup”-knappen på pekskärmen kan man köra manuellt upp och ner. I monteringshöjd kan önskat kretskort monteras på plats. Efter det kan man välja fallhöjd samt köra till fallhöjden. Till sist väljs antalet gånger som testet skall upprepas. Droptestern kan också styras från en dator.



Figur 14. Bild på den gamla droptestern.

3.4. Moderniseringens syfte

Med moderniseringen vill man uppnå en bättre fallhöjdsnoggrannhet ($\pm 0,5$ mm), ökad pålitlighet samt mindre totalkostnader. Den nya droptestern har också en enklare konstruktion samt en enklare konfiguration.

3.5. Nya Systemet

Den nya droptestern har en modernare PLC, pekskärm och servomotor från Omron, som använder sig av fältbussen EtherCAT. De övriga komponenterna är nästan samma som i det gamla systemet. I det nya systemet används inte gränslägesbrytare utan gränslägena bestäms i programmet.

4. Programmering

Vid programmering av logiken användes datorprogrammet Sysmac Studio v 1.10 och vid designen av det grafiska gränssnittet till pekskärmen användes datorprogrammet NB-Designer v 1.38. Programmet gjordes först hemma på testkonfiguration bestående av en programmerbar logik med pekskärm, servoförstärkare samt servomotor tillverkade av företaget Omron. Sedan sattes elmotorn på plats och alla sensorer kopplades till den programmerbara logikens I/O-modul på en prototypversion av en droptester i Salon Teknopajas fabrik i Salo. Programmet testades och ändringar i programmet gjordes så att droptestern fungerade som den skulle, samt enligt företagets önskemål.

4.1. PLC

Programmet gjordes med en kombination av programspråken ladderdiagram och funktionsblockdiagram. Först kopplades servomotorn ihop med droptestern och alla komponenter till logikens insignals- och utsignalsmodul. Sedan laddades testprogrammet upp till den programmerbara logiken och pekskärmen. Då servomotorn kopplades till droptestern konfigurerades motorn så att enheterna stämde överens. I programmet under "Axis Settings" och sedan "Unit Conversion Settings" valdes som enhet mm och 1048576 pulser/varv (antalet pulser som motorn roterar per varv) samt 70,83333 mm/varv (avståndet per varv som motorn utför).

Förutom "Setup"-sektionen och "Alarm"-sektionen är alla av programmets sektioner (se bilaga 1) gjorda så att det finns sekvenser med olika villkor som ska uppfyllas för att programmet skall gå vidare. Programmet börjar i sektionen "Homing" där servomotorn söker efter sin startposition efter att strömmen har satts på. Där finns också definierat när motorbromsen är påslagen. I detta fall är motorbromsen på då servomotorns ström är avslagen och vice versa. Efter detta programmerades droptestern till att först köra upp till

övre gränsen. Det finns en fotoelektrisk sensor monterad uppe som aktiveras då kälken är tillräckligt nära. I programmet definierades detta som den övre gränsen. Detta är den högsta möjliga fallhöjden och det går inte att köra högre upp än så. Den nedre gränsen definierades så att då droptesterns nedre sensorn aktiveras, sparas servomotorns position som nedre gräns i PLC:ns RAM-minne. Sedan kör kälken ner till den monteringshöjd som står på pekskärmen.

Efter detta gjordes programsektionerna "Drop Height" och "Installation Height" som gör det möjligt att köra till en ny monteringshöjd eller fallhöjd. Då fallhöjden är nådd kan själva falltestet börjas. I sektionerna "DropHMI" och "DropPC" sker själva falltestet och upprepas önskat antal gånger. Sektionerna har samma innehåll men bara en av dem används beroende på om droptestern styrs med pekskärm eller PC. Det finns också tre stycken räknare som räknar hur många fall som har gjorts totalt och per test samt hur många gånger det är kvar tills underhållsservice skall göras. I sektionen "Alarm" är olika felmeddelanden definierade, sektionen försöker nollställa eventuella felmeddelanden efter tre sekunder. I "Setup"-sektionen definieras hur snabbt droptestern ska stanna, samt köra manuellt upp eller ner ända tills övre eller nedre gränsen kommer emot.

4.2. Peksärm

Att designa det grafiska gränssnittet för pekskärmen var ganska enkelt. Först väljer man i programmet en knapp eller en display. Sedan anger man dess utseende och address till programmet i PLC:n. Det grafiska gränssnittet designades så att det finns två fönster, "Main" och "Setup", där droptestern kan styras och konfigureras. Det finns också ett meddelandefält där det står i vilket läge kälken befinner sig i, om servomotorn är påslagen och om det finns något fel i systemet. Då meddelandefältet gjordes valdes funktionen "Alarm" under "Parts" i "Project Library Window" och sedan definierades meddelandena (se figur 15) under "Project Database" och sedan "Alarm Setting".

Nere till höger finns en knapp som gör det möjligt att byta mellan "Main" och "Setup" fönstret. Båda fönstren har ett meddelandefält samt en strömknapp som bestämmer när servomotorn har ström. Följande knappar finns också på båda fönstren; Knappen "Stop" som stannar servomotorn, knapparna "Drive to Installation Height" och "Drive to Drop Height" som får droptestern att köra till monteringshöjd eller fallhöjd.

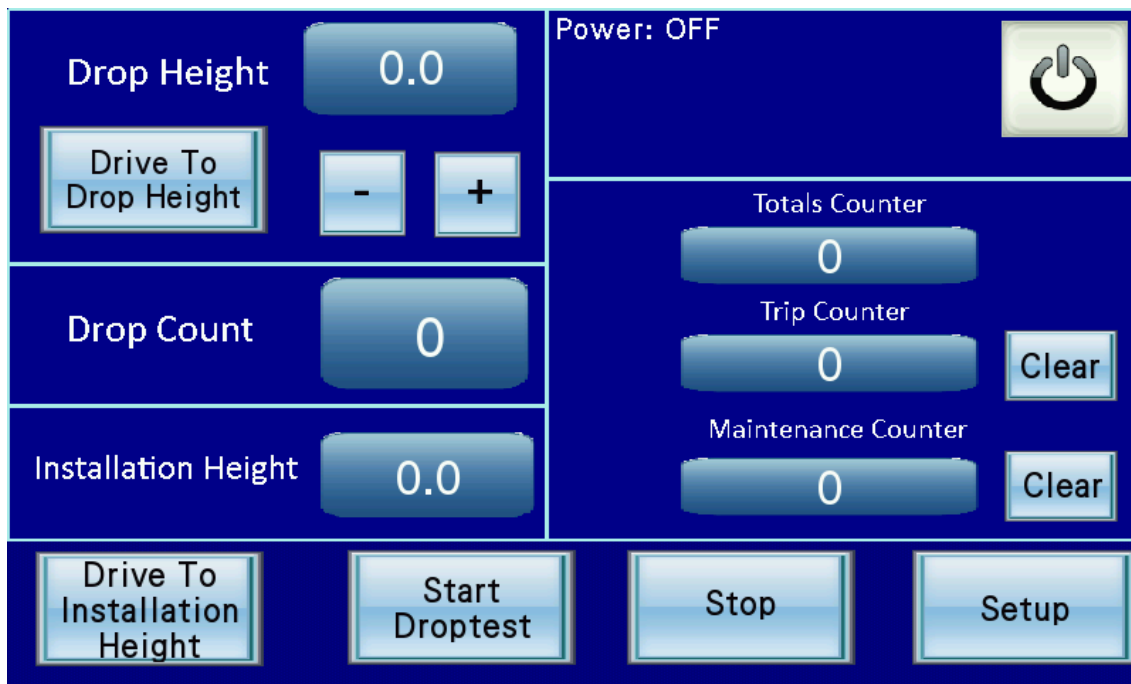
No.	Type	PT No.	Address inform...	Address	Alarm	Message
0	0	0	HMI0:PLC0:1	W_bit:10.01	On	Door Opened
1	0	0	HMI0:PLC0:1	W_bit:10.02	On	Running
2	0	0	HMI0:PLC0:1	W_bit:10.03	On	Ready to start
3	0	0	HMI0:PLC0:1	W_bit:10.05	On	Brake Released
4	0	0	HMI0:PLC0:1	W_bit:10.07	On	Paused
5	0	0	HMI0:PLC0:1	W_bit:210.01	On	Error
6	0	0	HMI0:PLC0:1	W_bit:210.02	On	Installation Height Reached
7	0	0	HMI0:PLC0:1	W_bit:210.03	On	Drop Height Reached
8	0	0	HMI0:PLC0:1	W_bit:210.04	On	Computer In Control
9	0	0	HMI0:PLC0:1	W_bit:210.05	Off	Power: OFF
10	0	0	HMI0:PLC0:1	W_bit:210.05	On	Power: ON
11	0	0	HMI0:PLC0:1	W_bit:210.06	On	Driving to Installation Height...
12	0	0	HMI0:PLC0:1	W_bit:210.07	On	Driving to Drop Height...
13	0	0	HMI0:PLC0:1	W_bit:210.08	On	Max Height Reached!
14	0	0	HMI0:PLC0:1	W_bit:220.06	On	Driving to Installation Height...
15	0	0	HMI0:PLC0:1	W_bit:210.10	On	Upper Limit Reached
16	0	0	HMI0:PLC0:1	W_bit:210.09	On	Driving to Upper Limit...
17	0	0	HMI0:PLC0:1	W_bit:210.11	On	Driving to Drop Height...
18	0	0	HMI0:PLC0:1	W_bit:220.08	On	Max Height Reached!

Buttons: Add, Delete, Delete All, Modify, Import, Export, OK

Figur 15. Alarm-inställningarna i NB-Designer.

4.2.1. "Main"-fönster

I "Main"-fönstret (se figur 16) kan man välja fallhöjd eller monteringshöjd och antalet gånger som testet skall upprepas genom att trycka på deras respektive rutor. Knappen "Start Droptest" gör att testet börjar förutsatt att droptestern befinner sig i fallhöjden. Knapparna "+" och "-" ändrar fallhöjden med 0,1 mm. De behövs då träffområdets material slits ut med tiden vilket leder till att den verkliga fallhöjden ändrar sig också. Då jiggen släpps ner mot träffområdet, finns det en matta som den först möter och mattan slits ut med tiden och måste bytas med jämna mellanrum. Det finns också tre räknare; "Totals Counter" visar antalet gånger som jiggen har släppts ner under ett test, "Trip Counter" visar antalet gånger som jiggen har släppts ner och den räknar på tills "Clear"-knappen trycks. "Maintenance Counter" visar antalet gånger kvar tills underhållsservice skall göras. Det krävs dessutom ett lösenord för att kunna nollställa denna räknare.

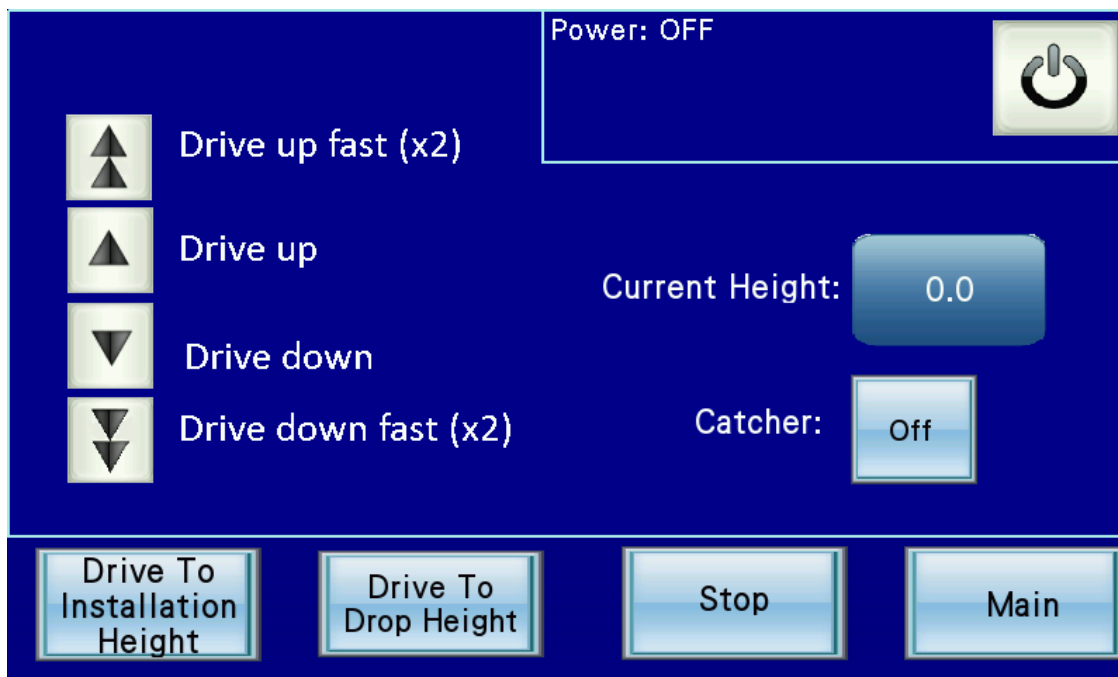


Figur 16. "Main"-fönstret på pekskärmen.

4.2.2. "Setup"-fönster

I "Setup"-fönstret (se figur 17) kan man köra droptestern manuellt upp och ner med pilknapparna. Detta behövs då man kollar i testskedet att kablarna inte är i vägen för fallets bana. Det finns två hastigheter, en långsammare och en snabbare. Droptesterns nuvarande höjden visas också. Det är möjligt att välja om "Catcher"-magneten ska användas.

"Catcher"-magneten är den magnet som befinner sig nere till vänster och har som uppgift att fånga jiggen efter att den har studsat ner en gång.



Figur 17. "Setup"-fönstret på pekskärmen.

5. Slutsats

Det var intressant och lärorikt att göra ett program till droptestern. Allt gick i stora drag utan bekymmer och apparaten fungerade i stort sett som jag hade tänkt när jag gjorde programmet först hemma. Det enda som tog lite mer tid var att få alla världens och knappars adresser i pekskärmen att stämma överens med programmet. Vid eventuella frågor som jag hade angående programmeringen hjälpte personalen på företaget Kasatek till och personalen på Salon Teknopaja vid frågor angående droptesterns funktion. Det som inte blev gjort var att programmera droptestern så att den går att styra från en PC samt kalibreras. Pekskrmen behövdes till en annan av företaget Salon Teknopajas tester så det fick bli till ett senare skede.

6. Källförteckning

(1)Gonzalez C., Engineering Essentials: *What Is a Programmable Logic Controller?* 2015

Tillgänglig: <http://machinedesign.com/engineering-essentials/engineering-essentials-what-programmable-logic-controller> (hämtat 31.3.2016)

(2)AutomationDirect, *History of the PLC*, Tillgänglig:

<http://library.automationdirect.com/history-of-the-plc/> (hämtat 31.3.2016)

(3) Schneider Electric Co, *About Modicon*,Tillgänglig:

<http://www.schneider-electric.co.uk/en/about-us/company-profile/brands/modicon.jsp>

(hämtat 31.3.2016)

(4)Bosch Rexroth Corporation, *Understanding the IEC61131-3 Programming Languages*, 2009, Tillgänglig:

http://www.automation.com/pdf_articles/IEC_Programming_Thayer_L.pdf (hämtat 2.4.2016)

(5) ADVANTECH B+B SMARTWORX, *AN INTRODUCTION TO INDUSTRIAL*

Ethernet, Tillgänglig: <http://www.bb-elec.com/Learning-Center/All-White-Papers/Ethernet/Introduction-to-Industrial-Ethernet.aspx> (hämtat 3.4.2016)

(6) Automation World, *Understanding Ethernet Speed and Determinism*, 2011,

Tillgänglig: <http://www.automationworld.com/networking-amp-connectivity/understanding-ethernet-speed-and-determinism> (hämtat 19.4.2016)

(7) RTA Automation, *What is Industrial Ethernet?* Tillgänglig:

<http://www.rtaautomation.com/industrial-library/what-is-industrial-ethernet/> (hämtat 19.4.2016)

(8) EtherCAT Technology Group, *EtherCAT - the Ethernet Fieldbus*, Tillgänglig:

https://www.ethercat.org/download/documents/ETG_Brochure_EN.pdf(hämtat 31.3.2016)

(9) EtherCAT Technology Group, *EtherCAT -the Ethernet Fieldbus-*, Tillgänglig:

<https://www.ethercat.org/en/technology.html>(hämtat 31.3.2016)

(10) JEDEC Solid State Technology Association, *Board Level Drop Test Method of Components for Handheld Electronic Products*, 2003.

Figurkällor

Figur 1. Bild på en Omron NX701-1700 programmerbar logik med strömkälla. Tillgänglig:

<http://www.omron-ap.co.in/solutions/sysmacstyle/one-machine-controller.html>

Figur 2. Bild på en programmerbarlogiks uppbyggnad. Egen bild.

Figur 3: Exempel på ett ladderdiagram ur datorprogrammet PC Worx 6.30.1202.

Figur 4: Exempel på ett funktionsblockdiagram ur datorprogrammet PC Worx 6.30.1202.

Figur 5: Exempel på en instruktionslista ur datorprogrammet PC Worx 6.30.1202.

Figur 6: Exempel på ett program i Strukturerad Text ur datorprogrammet PC Worx 6.30.1202.

Figur 7: Exempel på ett flödesdiagram i datorprogrammet ur datorprogrammet PC Worx 6.30.1202.

Figur 8: Bild på ett RJ45-kontaktpar avsedda för industriellt bruk. Tillgänglig:

<http://www.tme.eu/u/produkt-tygodnia/conecrj45.jpg>

Figur 9. Ritning av kretskortet som används vid testning.

(JEDEC JESD22-B111)

Figur 10. Översikt av en droptesters komponenter. Bild bearbetad från JEDEC JESD22-B111.

Figur 11. Graf över stötpulsen. Egen bild.

Figur 12. Bild på motorn. Egen bild

Figur 13. Bild på lyftkälken och jiggen. Egen bild.

Figur 14. Bild på den gamla droptestern. Salon Teknopaja, USER AND SERVICE GUIDE PWB-Level Drop Tester.

Figur 15. Alarm-inställningarna i NB-Designer. Från datorprogrammet NB-Designer v1.38.

Figur 16. "Main"-fönstret på pekskärmen från datorprogrammet NB-Designer v1.38.

Figur 17. "Setup"-fönstret på pekskärmen från datorprogrammet NB-Designer v1.38.

Droptester

Droptester

Author: Johan Lönnfors
Created: 20.4.2016 14:13:01
Last Modified: 20.4.2016 14:18:20
Comment:

Sysmac Studio Module Version: 1.1.4.95

Droptester

Table of Contents

Droptester	3
1.new_Controller_0	3
1-9.POU's	3
1-9-1.Programs	3
1-9-1-1.Main	3
1-9-1-1-1.Variables	3
1-9-1-1-2.Alarm	6
1-9-1-1-3.Homing	9
1-9-1-1-4.InstallationHeight	14
1-9-1-1-5.DropHeight	17
1-9-1-1-6.DropPC	22
1-9-1-1-7.DropHMI	30
1-9-1-1-8.Setup	38

1.new_Controller_0**1-9.POU's****1-9-1.Programs****1-9-1-1.Main****1-9-1-1-1.Variables**

Name	Data Type	Initial Value	AT	Retain	Constant	Comment
VAR						
Rcset	MC_Reset			False	False	
ZStop	MC_Stop			False	False	
ZSetHome	MC_Home			False	False	
ZPowerOnInst	MC_Power			False	False	
DriveZJumpToRow	LINT			False	False	
DriveZSeqRst	BOOL			False	False	
DriveZ	SeqHandling			False	False	
HomingZJumpToRow	LINT			False	False	
HomeZSeqRst	BOOL			False	False	
HomeZ	SeqHandling			False	False	
Drop	SeqHandling			False	False	
DropJumpToRow	LINT			False	False	
DropSeqRst	BOOL			False	False	
Z_MoveToHomeAuxSQLimit	MC_TorqueControl			False	False	
MCerrD	TON			False	False	
MCErrorRstFB	ResetMCError			False	False	
ECErrorRstFB	ResetECError			False	False	
PLCErrorRstFB	ResetPLCError			False	False	
TripCounterINT	INT			False	False	
TryResetD	TON			False	False	
MaintenanceCounterINT	INT			False	False	
InstH	SeqHandling			False	False	
InstHJumpToRow	LINT			False	False	
InstHSeqRst	BOOL			False	False	
TotalsCounterINT	INT			False	False	
ShockCounterLINT	LINT			False	False	NB-näyttö
ManualMode	MC_MoveJog			False	False	
Delay	TON			False	False	
AlarajallaString	STRING[20]			False	False	
Asennuskorkeuteen	MC_MoveAbsolute			False	False	
Pudotuskorkeuteen	MC_MoveAbsolute			False	False	
Asennuskorkeuteen2	MC_MoveAbsolute			False	False	
NB_master	BOOL			False	False	
AsennuskorkeuteenNB	MC_MoveAbsolute			False	False	
NBDrop	SeqHandling			False	False	
NBDropJumpToRow	LINT			False	False	
nBDropSeqRst	BOOL			False	False	
ViiveHMI	TON			False	False	
PudkorkeuteenNB	MC_MoveAbsolute			False	False	
AskorkeuteenNB	MC_MoveAbsolute			False	False	

Droptester

new_Controller_0 Variables

Main

B						
AjaNoutoonNB	MC_MoveAbsolute			False	False	
Stop	MC_Stop			False	False	
ManualModeb	MC_MoveJog			False	False	
Ylärajaarvo	STRING[20]			False	False	
InitAjaYlös	MC_MoveJog			False	False	
moveNegative	BOOL			False	False	
moveNegativeb	BOOL			False	False	
YlärajaSaavutettu	BOOL			False	False	
initAskorkeuteen	MC_MoveAbsolute			False	False	
Pudkork	LREAL			False	False	
Viive2	TON			False	False	
ZSetPos	MC_SetPosition			False	False	
UusiPudKork	MC_MoveAbsolute			False	False	
AsKorkRes	BOOL			False	False	
AsKorkRes2	BOOL			False	False	
InstHeight	LREAL			False	False	
AxisActPos	STRING[10]			False	False	
AxisActPos2	LREAL			False	False	
Viive3	TON			False	False	
YlärajaSaavutettu2	BOOL			False	False	
NykKorkLreal	LREAL			False	False	
Askorkeus2	LREAL			False	False	
Pudkork2	LREAL			False	False	
InstHeight2	LREAL			False	False	
AjoAlasNB	MC_MoveJog			False	False	
Pudkorkeuteen2NB	MC_MoveAbsolute			False	False	
Viive5	TON			False	False	
Viive6	TON			False	False	
MagneetinViive	TON			False	False	
AjaNoutoonPC	MC_MoveAbsolute			False	False	
AjaNoutoon2NB	MC_MoveAbsolute			False	False	
DropCountEQ	BOOL			False	False	
AxisActPos5	LREAL			False	False	
PudkorkeuteenPC	MC_MoveAbsolute			False	False	
ViivePC	TON			False	False	
Pudkorkeuteen2PC	MC_MoveAbsolute			False	False	
AskorkeuteenPC	MC_MoveAbsolute			False	False	
VAR_EXTERNAL						
NBInput	NB			False		
ZPowerON	BOOL			False		
Prog	ARRAY[0..18] OF members			False		
Z_Axis	sAXIS_REF			True		
Index	INT			False		
Pudota	BOOL			False		PC Komento
DriveStepsZ	DriveCurrentStep			False		
HomingStepsZ	DriveCurrentStep			False		
DropStepsZ	DriveCurrentStep			False		
KpITunnistus	BOOL			False		
TripCounterNB	LWORD			False		
MaintenanceCounterNB	LWORD			False		
MCErrActive	BOOL			False		
ECErrActive	BOOL			False		
PLCErrActive	BOOL			False		
EIPErrActive	BOOL			False		

Bilaga 1

Droptester

new_Controller_0 Variables

Main

Error	BOOL				False	
MCErrCode	DWORD				False	
ECErrorCode	DWORD				False	
PLCErrorCode	DWORD				False	
EIPErrorCode	DWORD				False	
TotalsCounterNB	LWORD				False	
ShockCounter	WORD				False	
PC_Aja_Askorkeuteen	BOOL				False	
InstHStepsZ	DriveCurrentStep				False	
CycleCounter	LINT				False	
PC_Pudotuskorkeuteen	BOOL				False	PC Komento
Alaraja	BOOL				False	Anturi
PoimMagneetti	BOOL				False	PoimintaMagneetti
Magneetti	BOOL				False	
NBDropStepsZ	DriveCurrentStep				False	
PC keskeyty	BOOL				False	
PudotuskorkeusOK	BOOL				False	
PC_master	BOOL				False	
Yläraja	BOOL				False	
StoredHeight	DWORD				False	NB: D220, DropH.
Jarru	BOOL				False	
DropHeightMsg	BOOL				False	Viesti NB-näytölle
InstHeightMsg	BOOL				False	
maxkorkeus	BOOL				False	NB-näytölle
InitInstHeightMsg	BOOL				False	
AsennuskorkeusOK	BOOL				False	
UpperLimitMsg	BOOL				False	Viesti NB-näytölle
UpperLimitRchMsg	BOOL				False	Viesti NB-näytölle
Nykyinenkork	DWORD				False	NB: D240, CurrentH
NB_InstHeight	DWORD				False	
DropHeightMsg2	BOOL				False	
adder	BOOL				False	
subtract	BOOL				False	
maxkorkeus2	BOOL				False	
Solenoidi	BOOL				False	

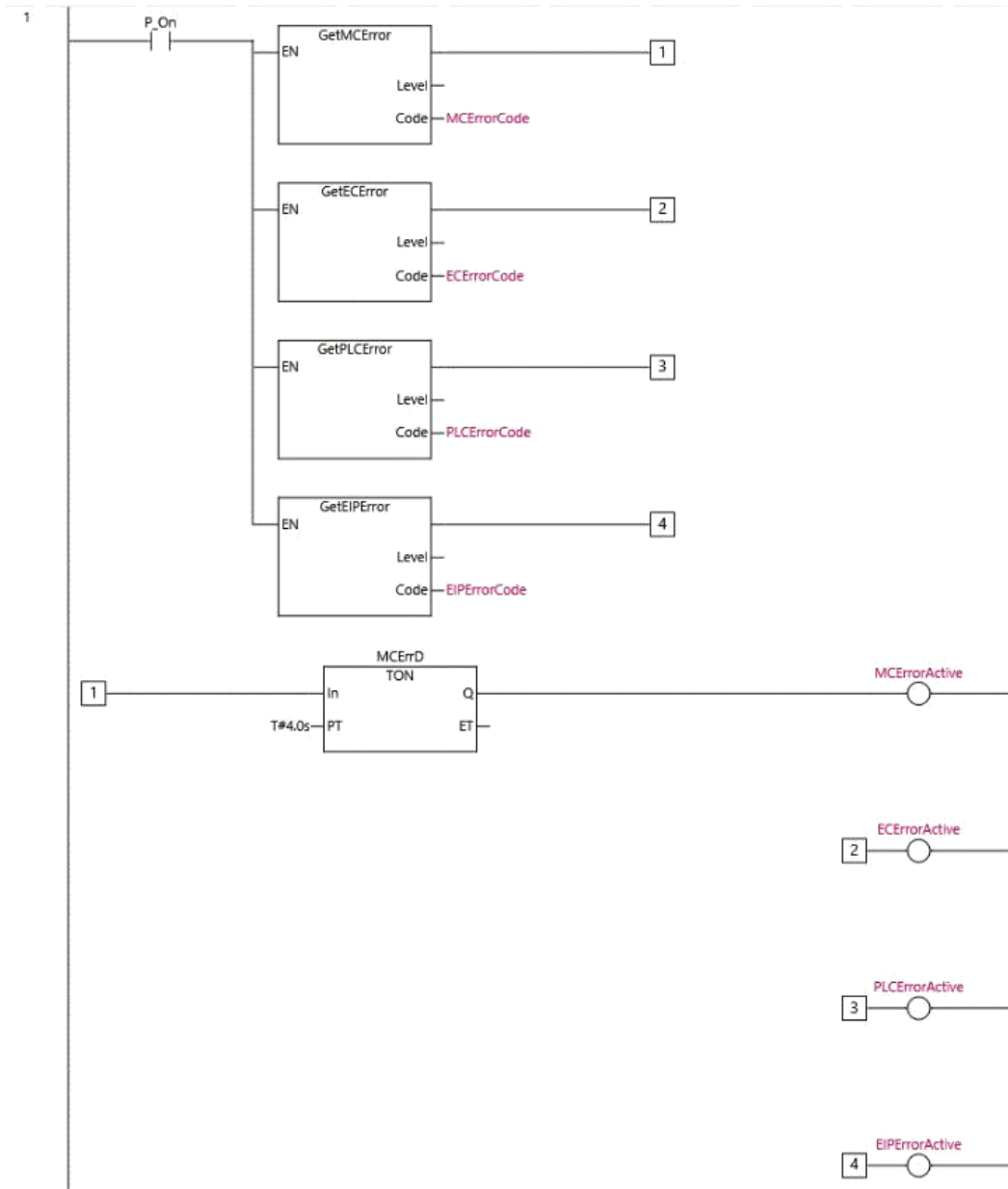
Droptester

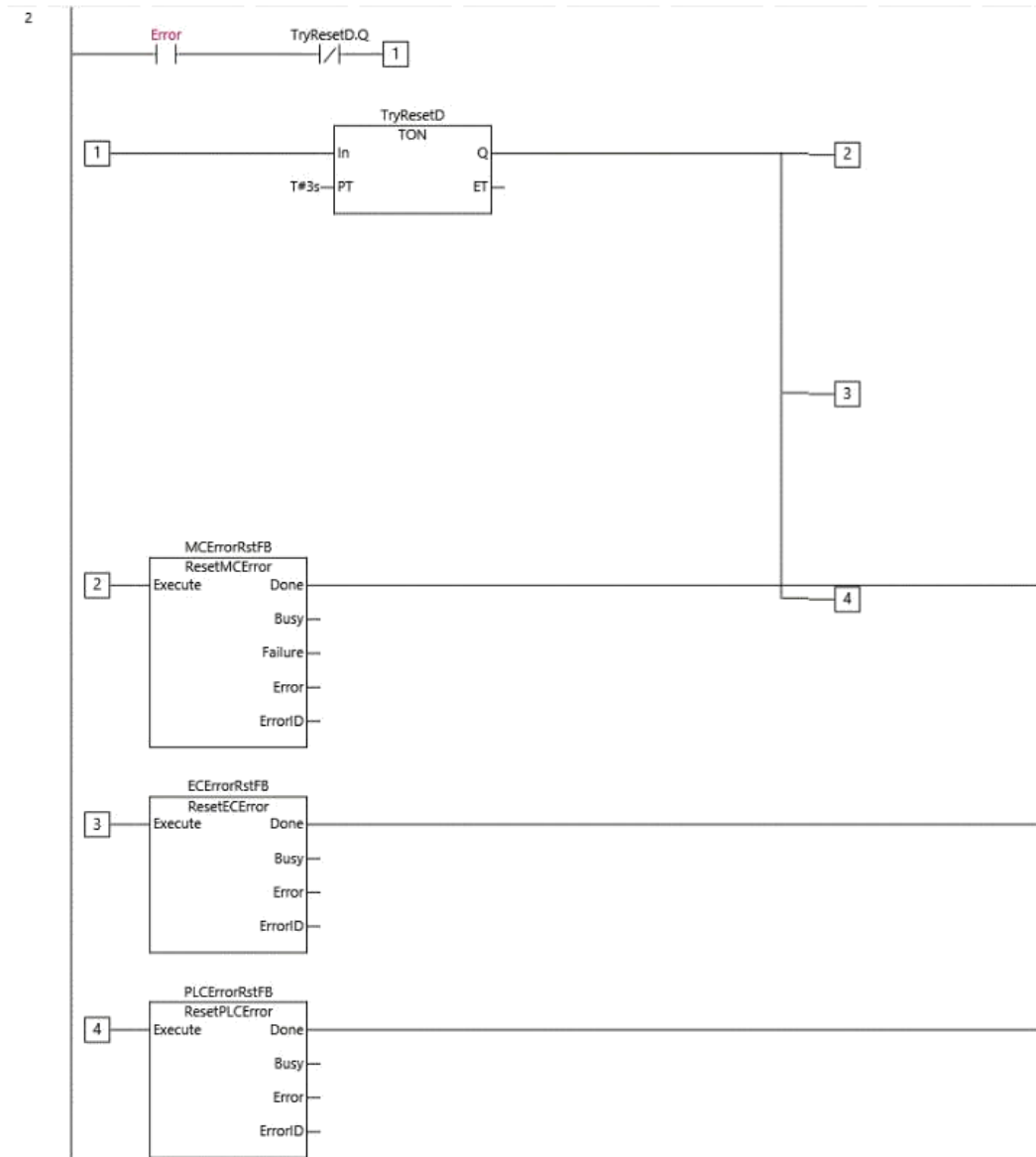
new_Controller_0 Alarm

Main Alarm

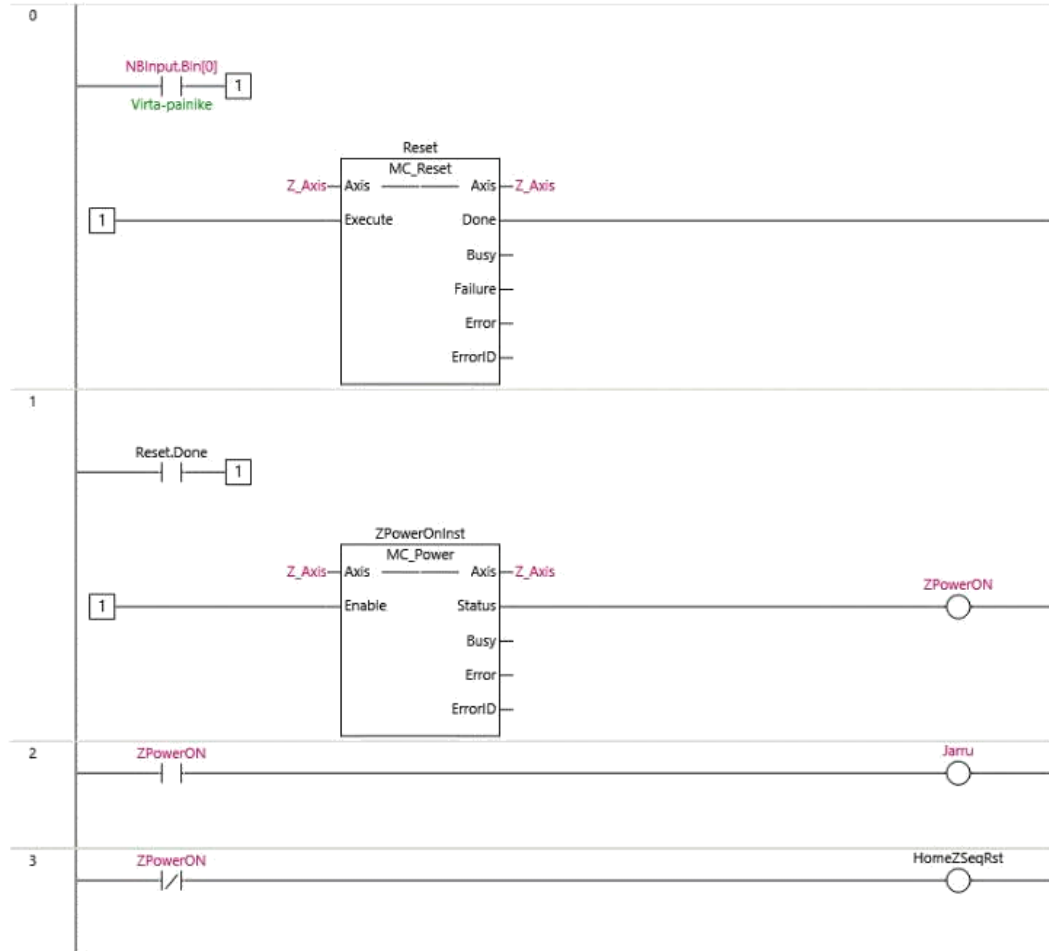
1-9-1-1-2.Alarm

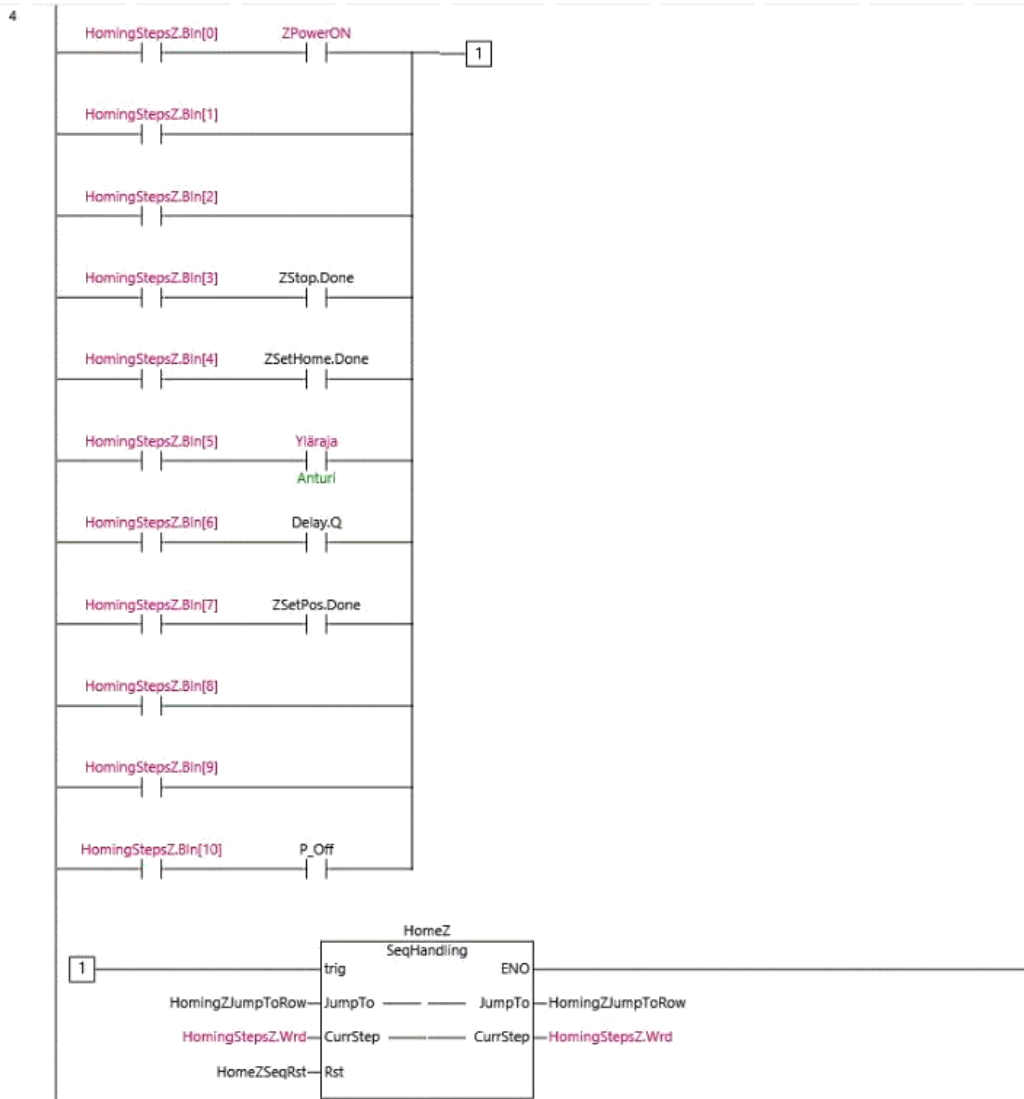


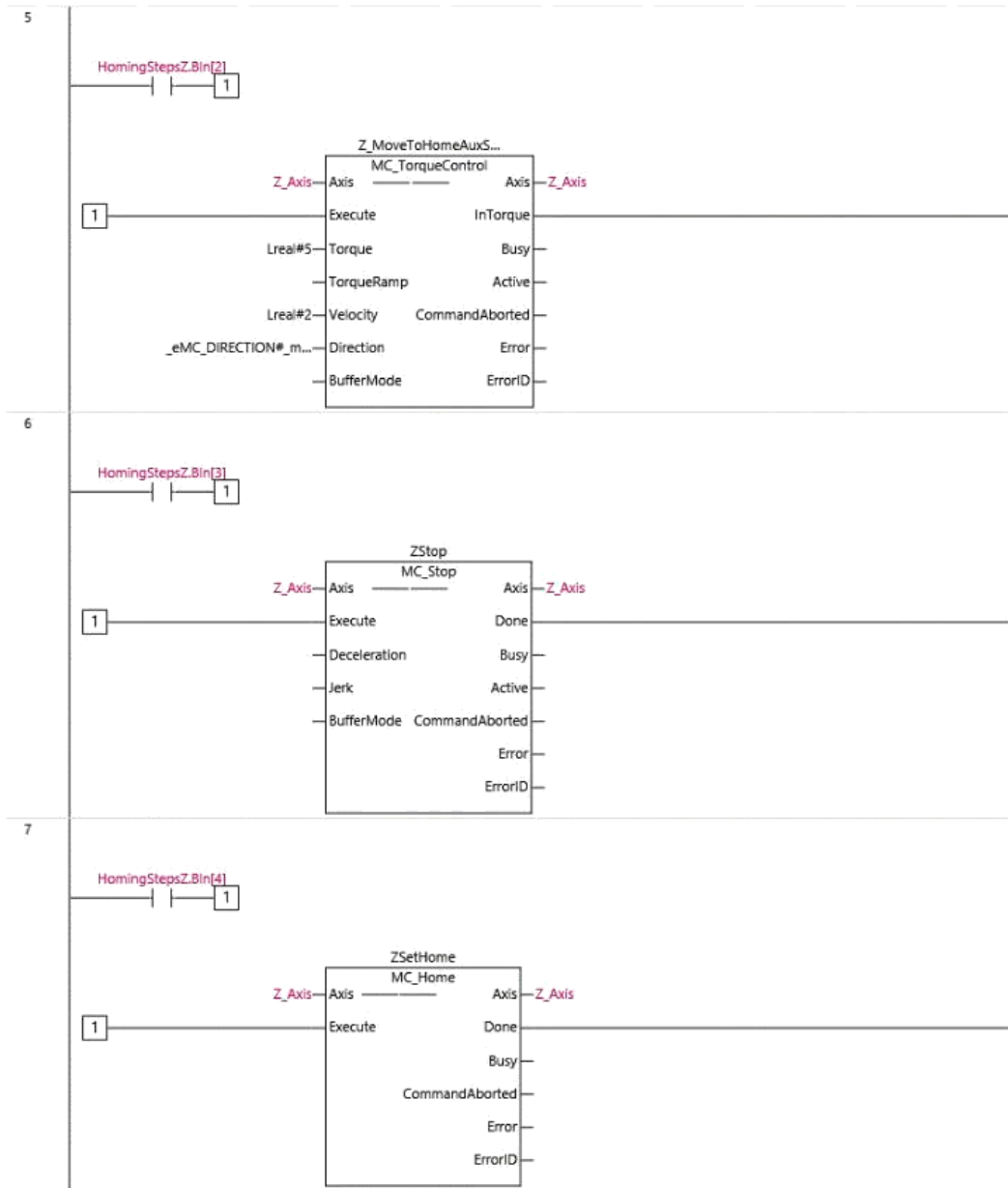


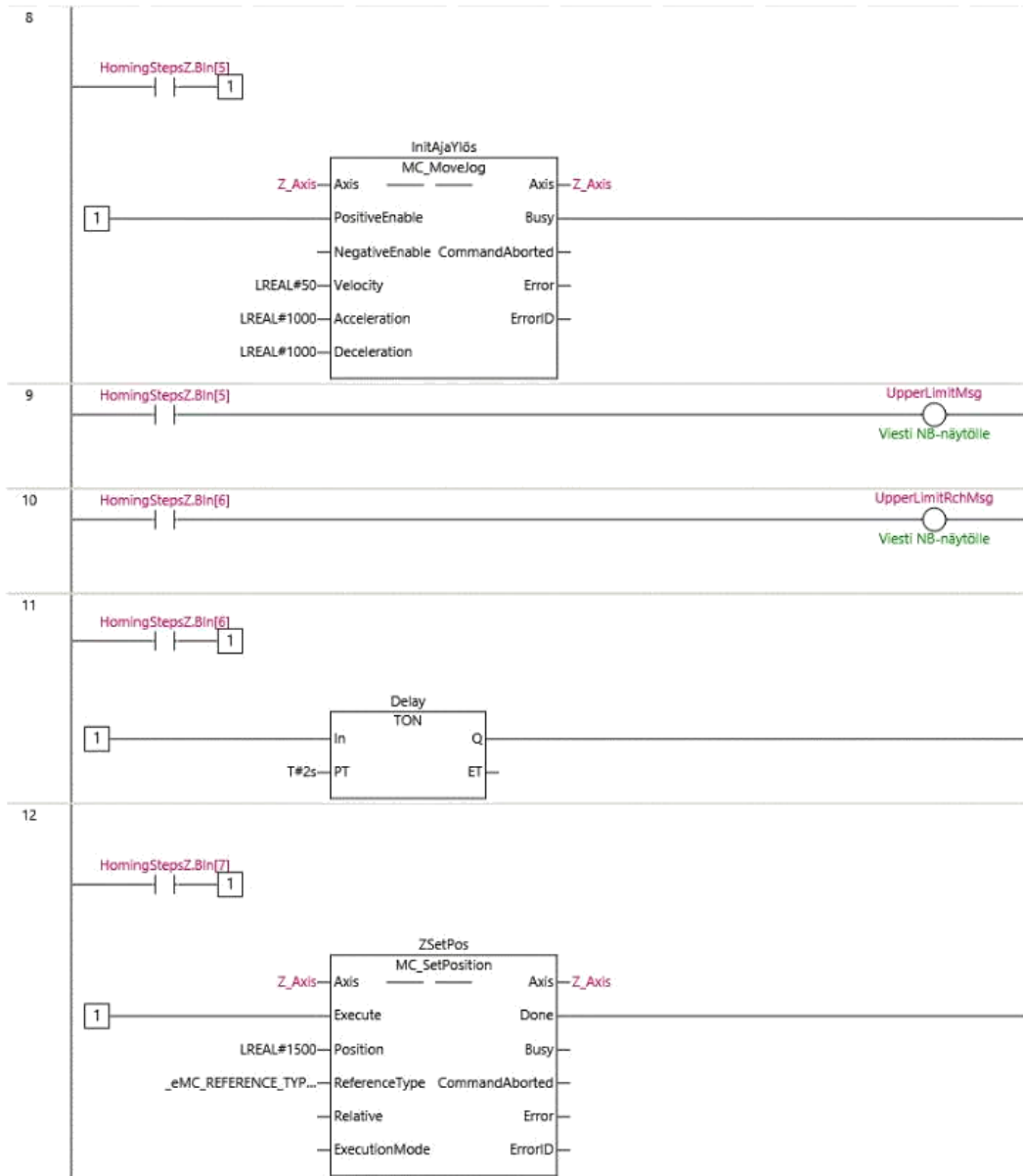


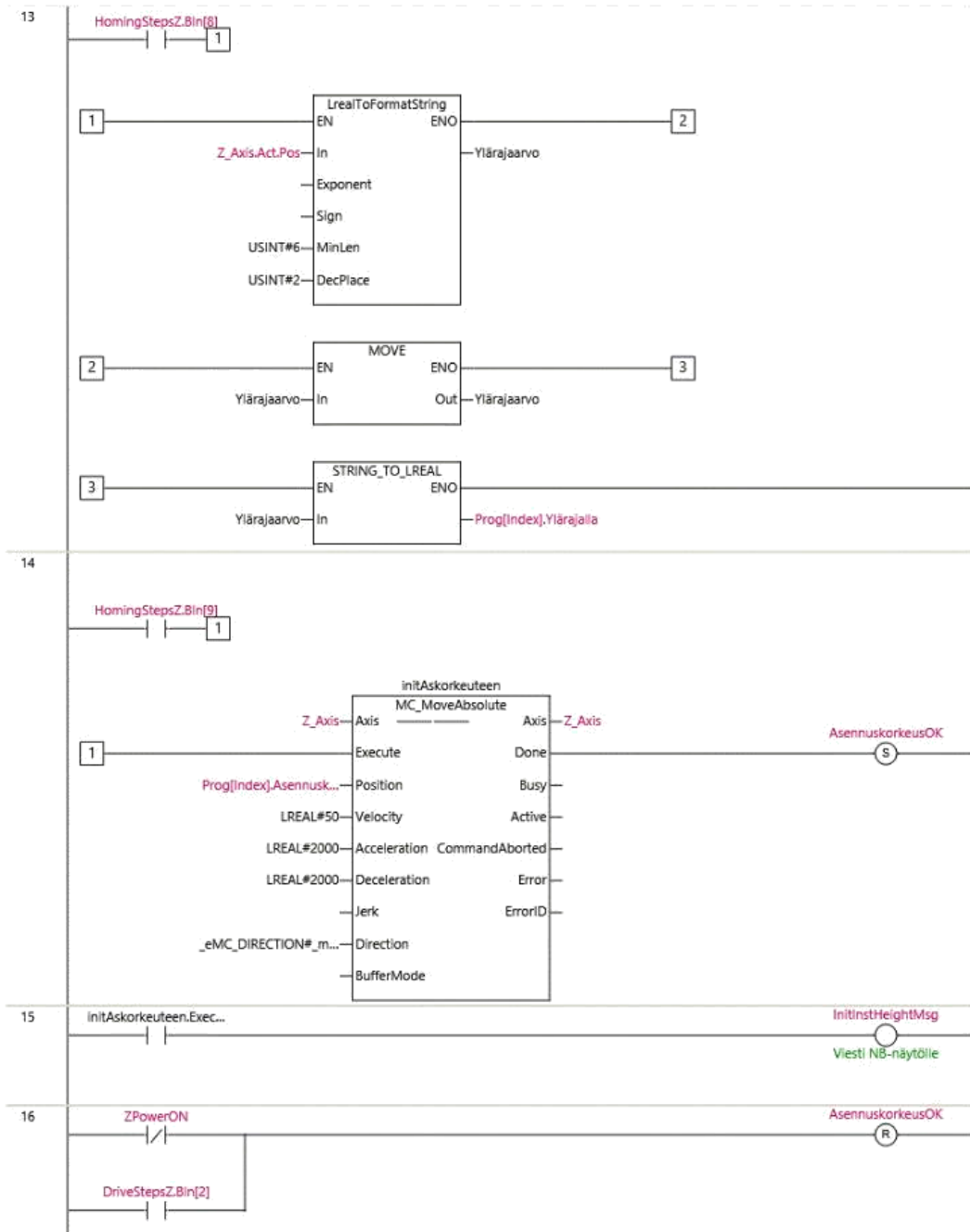
1-9-1-1-3.Homing



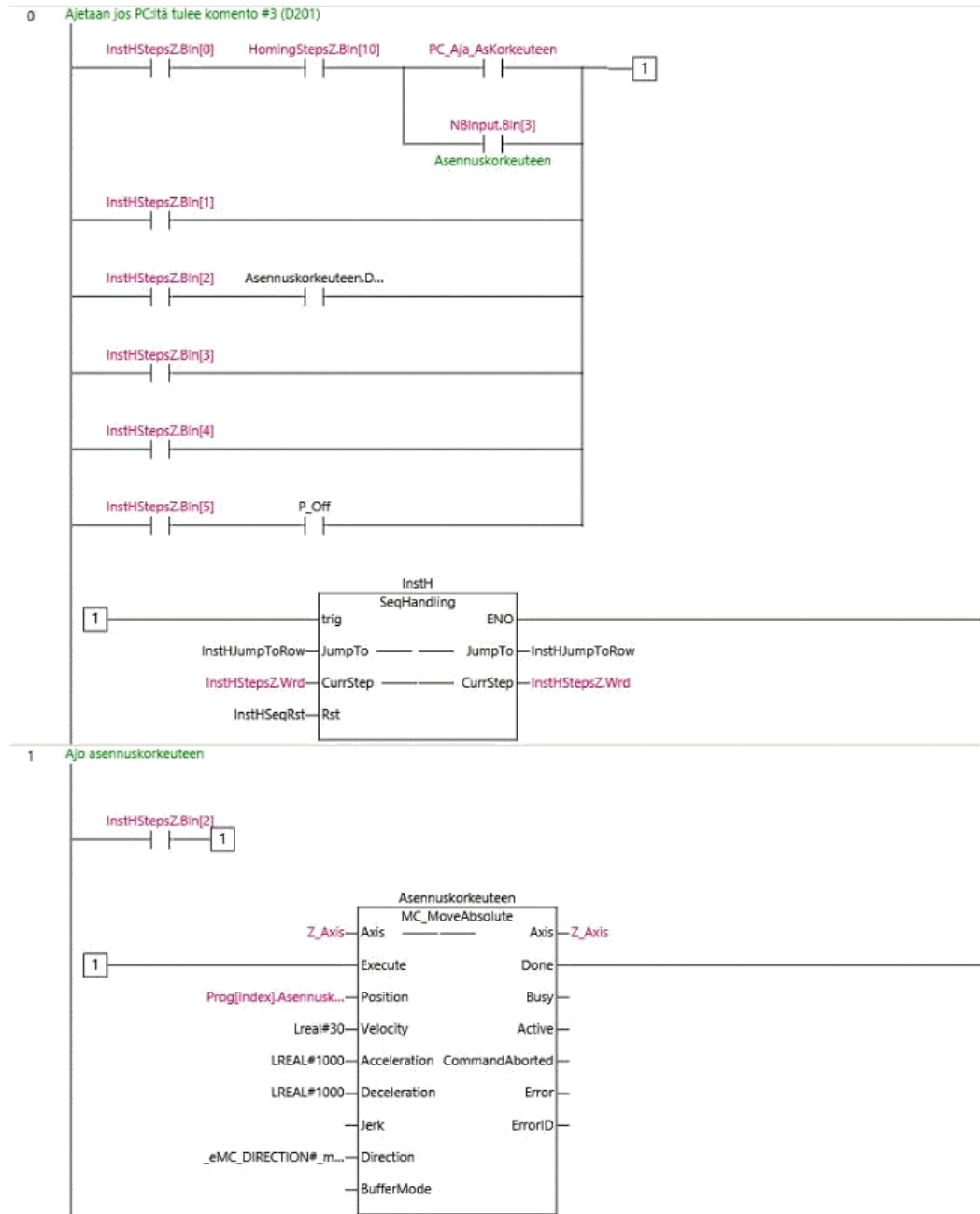








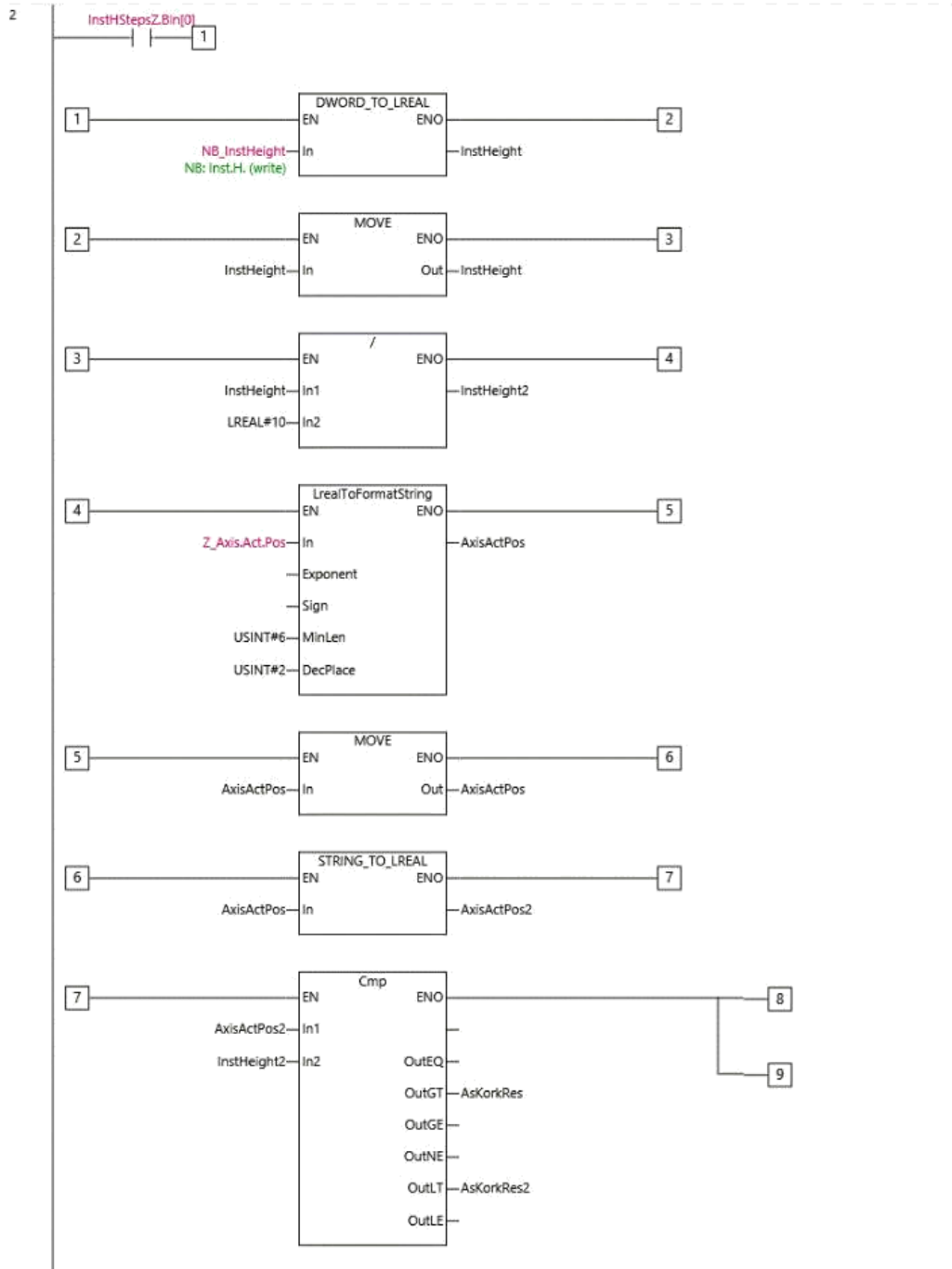
1-9-1-1-4.InstallationHeight



Droptester

new_Controller_0 InstallationHeight

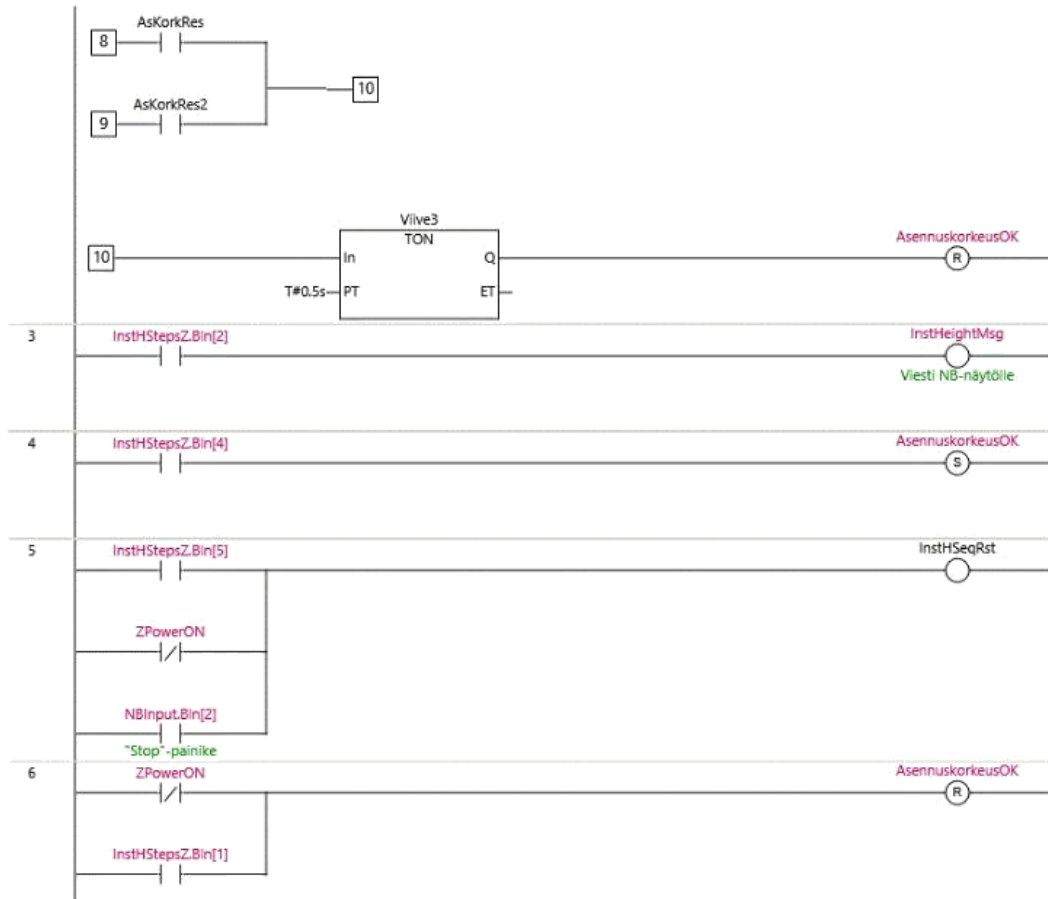
Main InstallationHeight

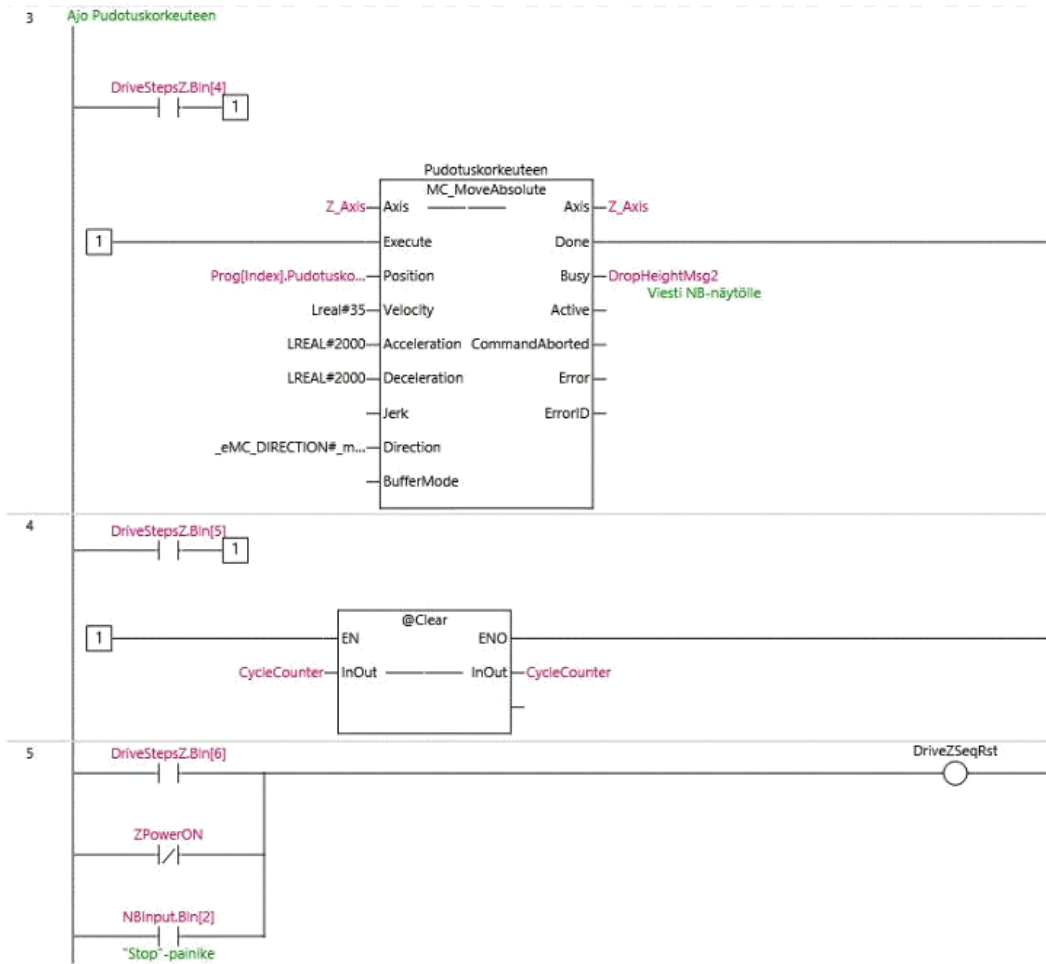


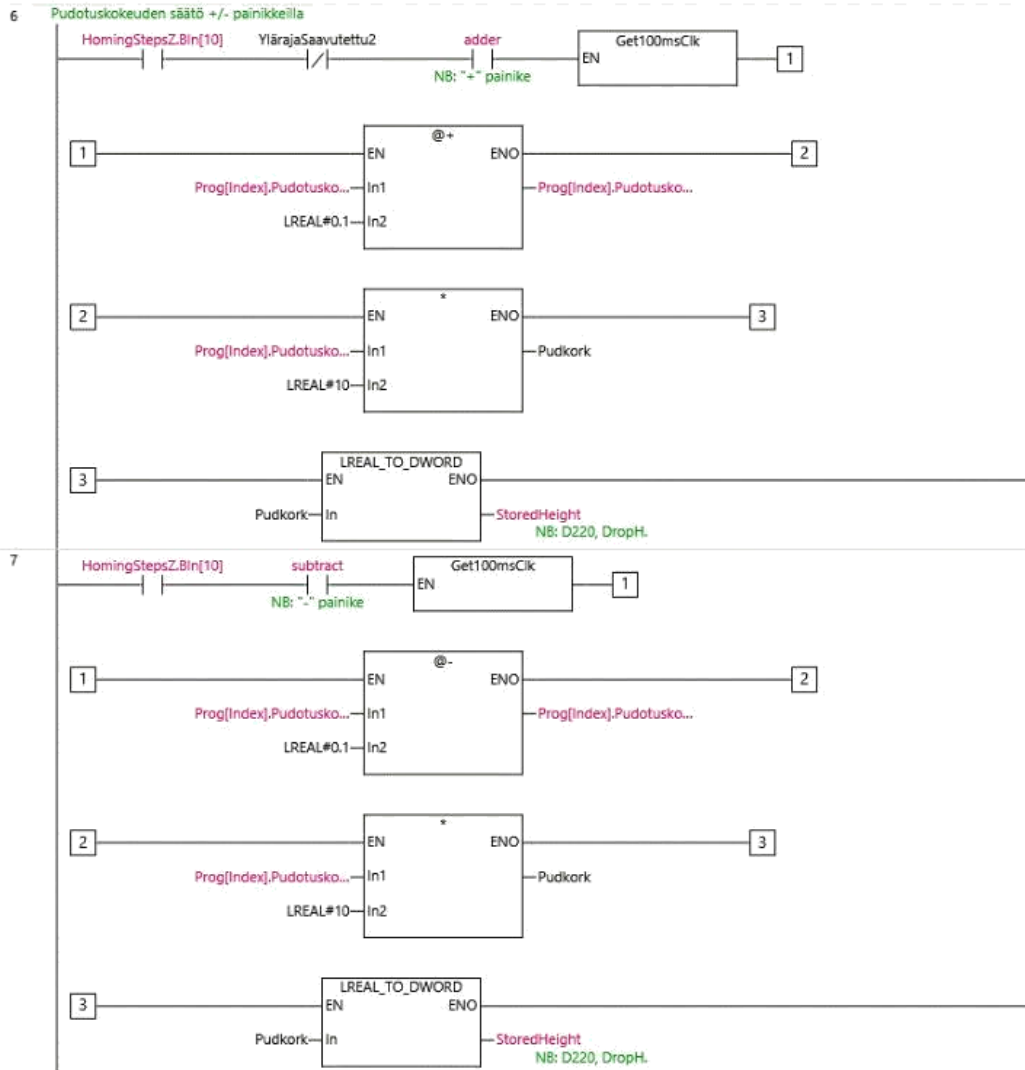
Droptester

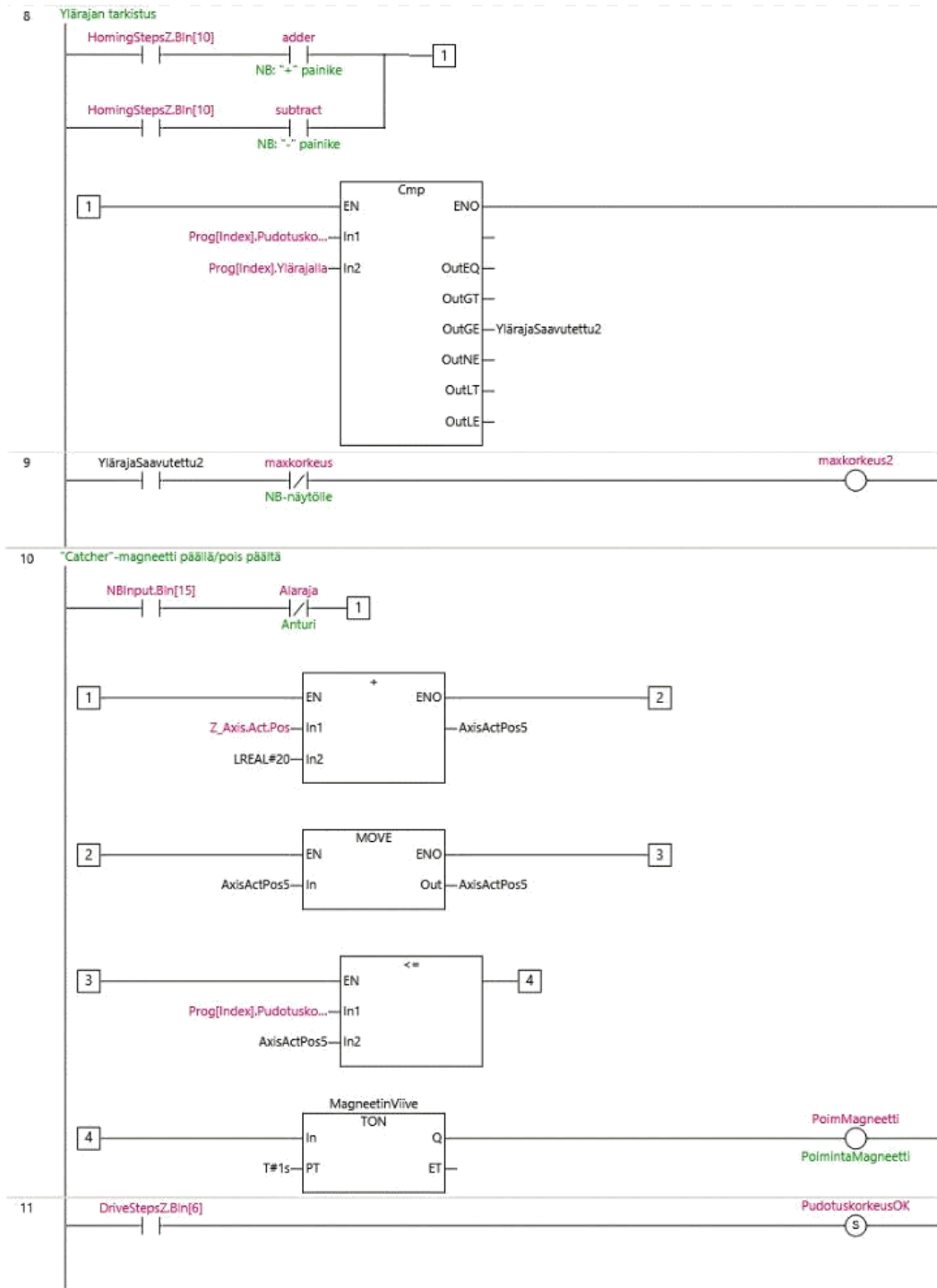
new_Controller_0 InstallationHeight

Main InstallationHeight





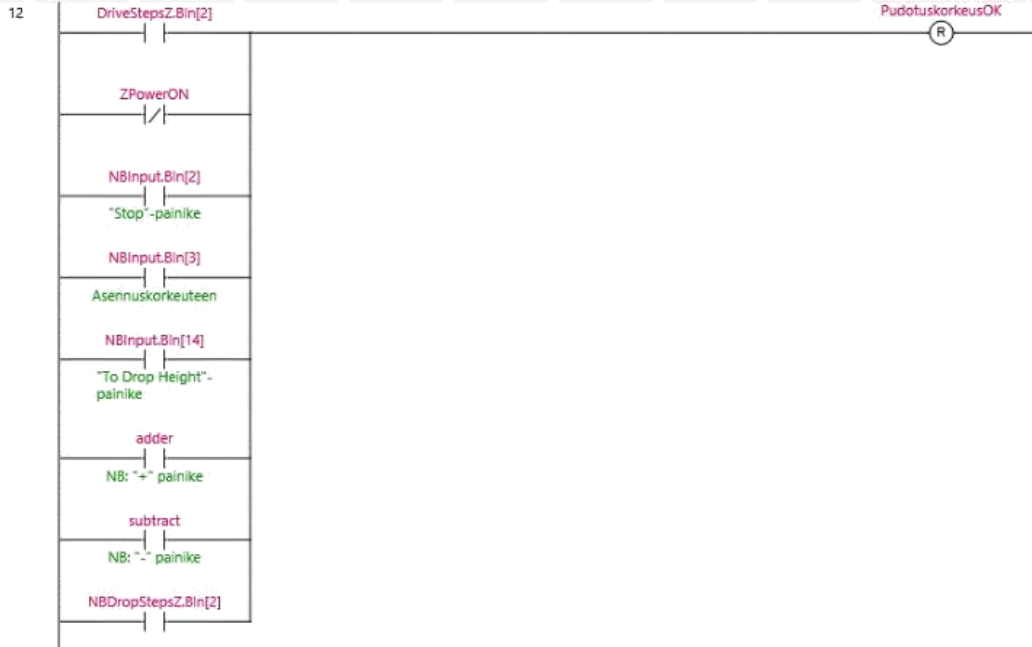




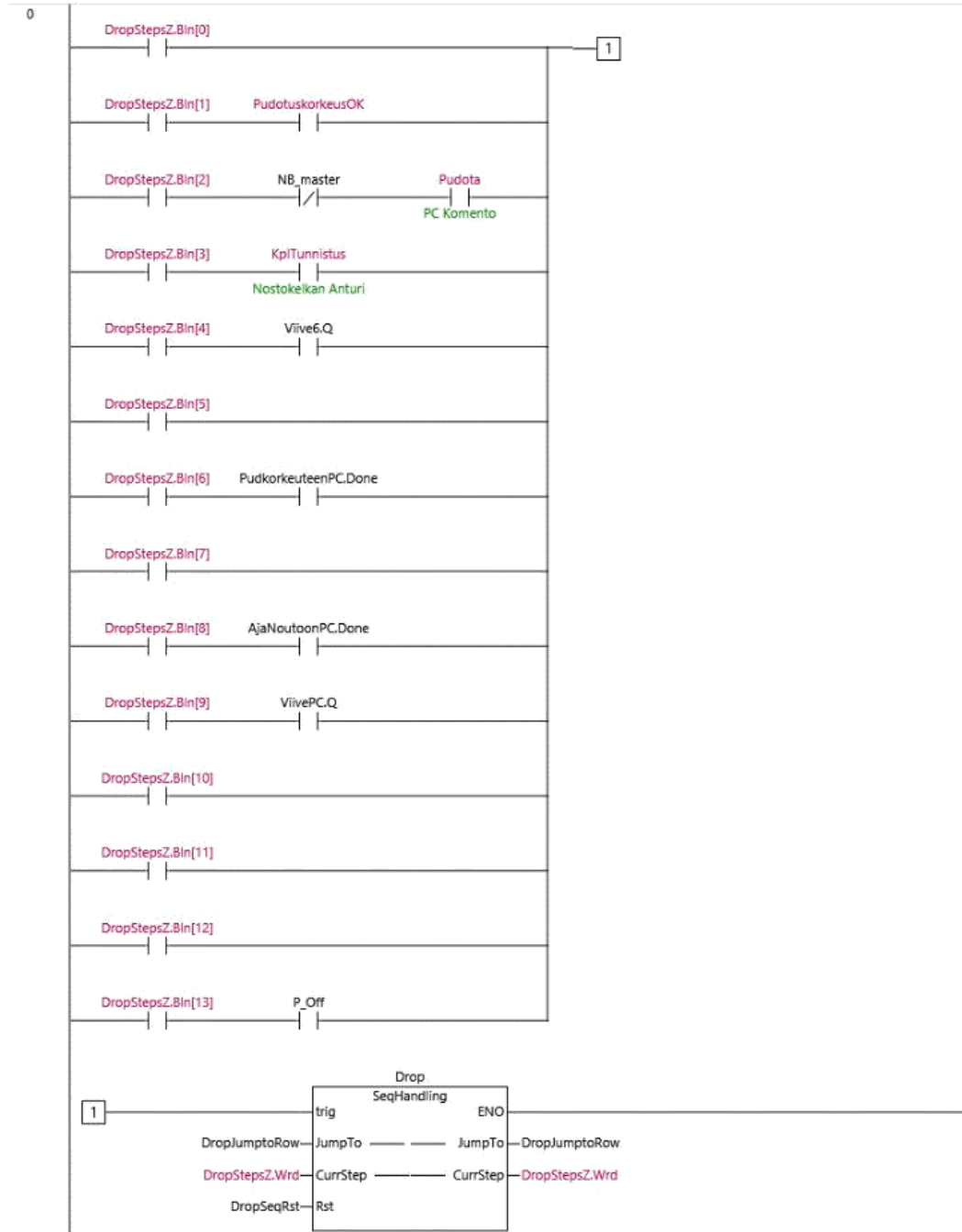
Droptester

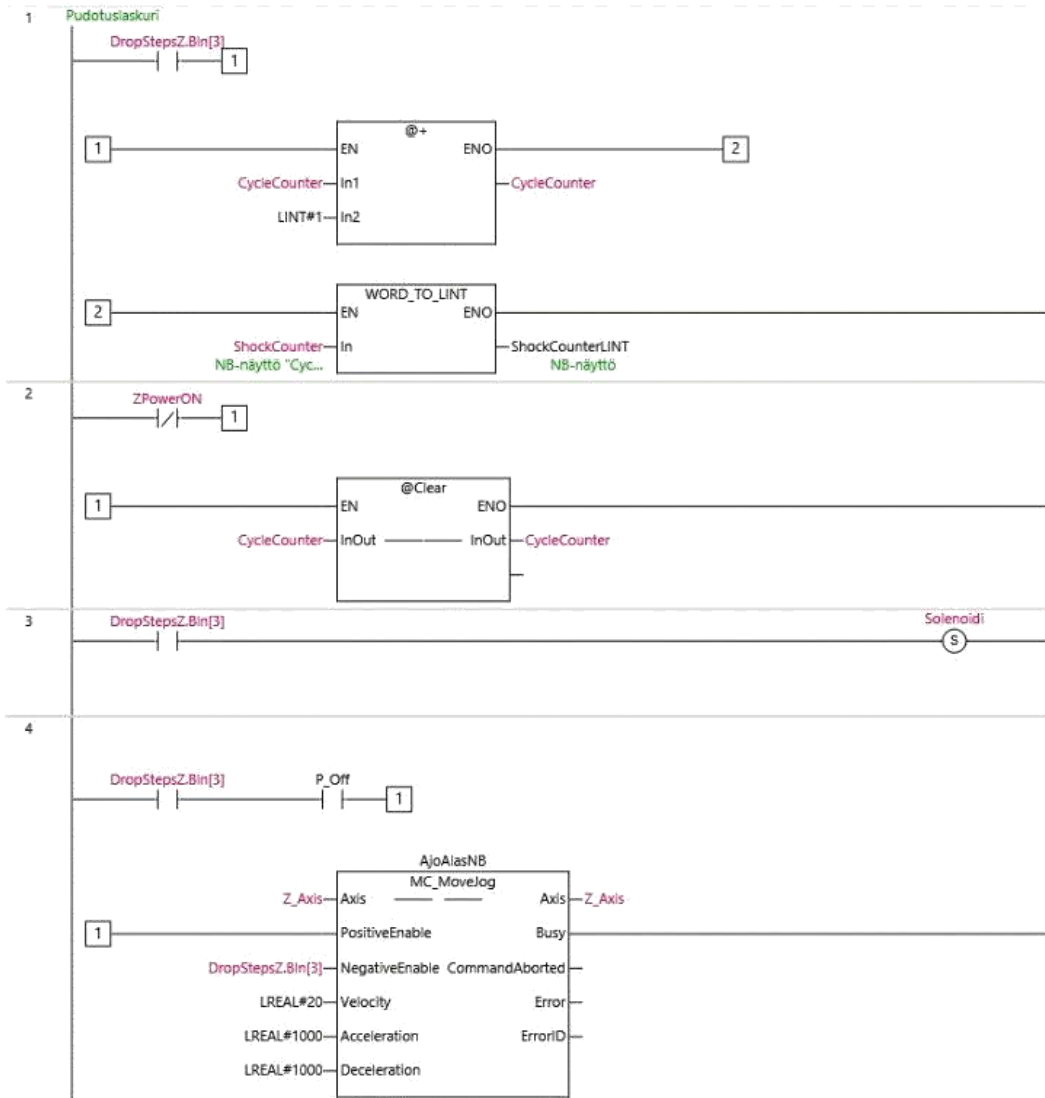
new_Controller_0 DropHeight

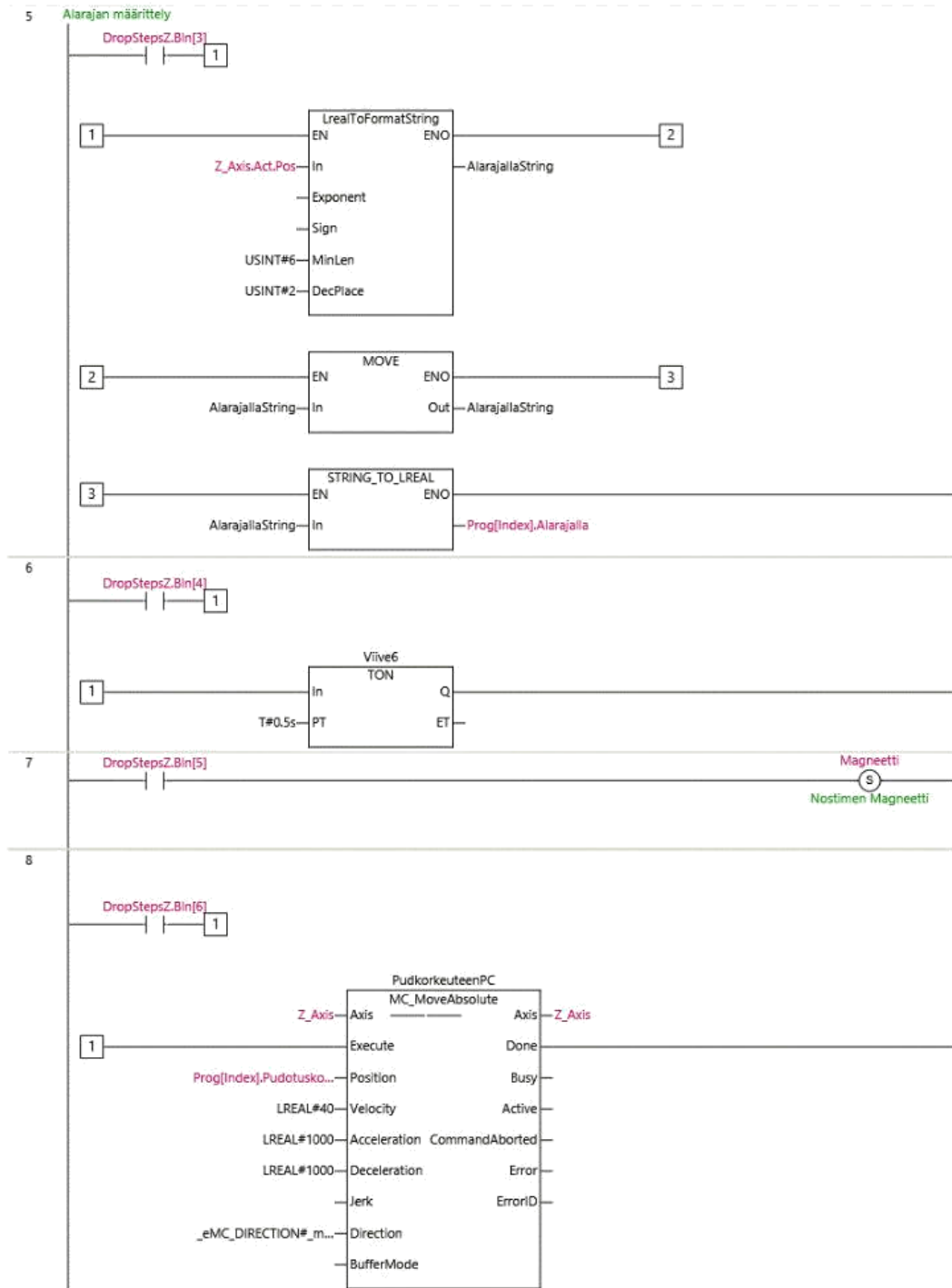
Main DropHeight

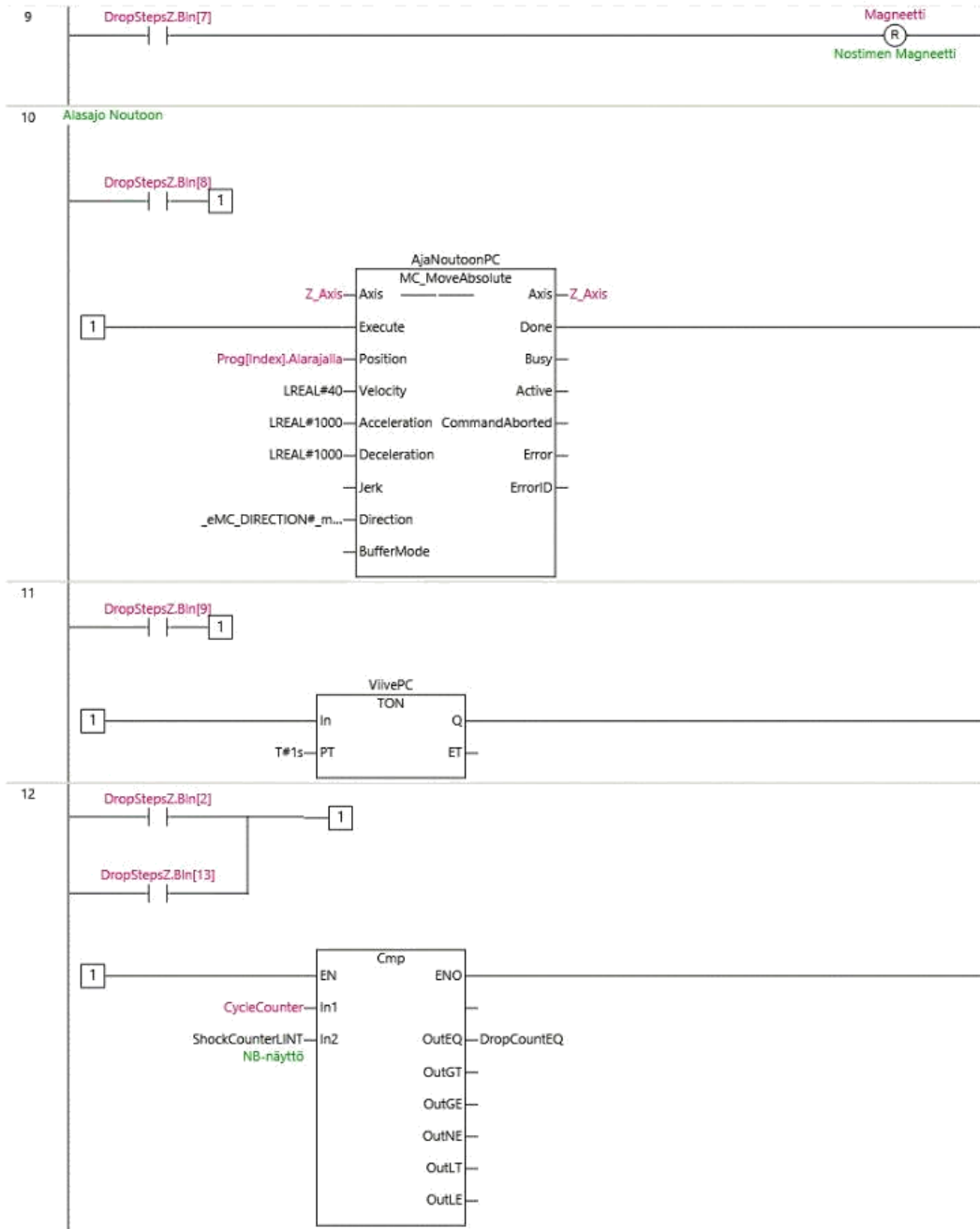


1-9-1-1-6.DropPC





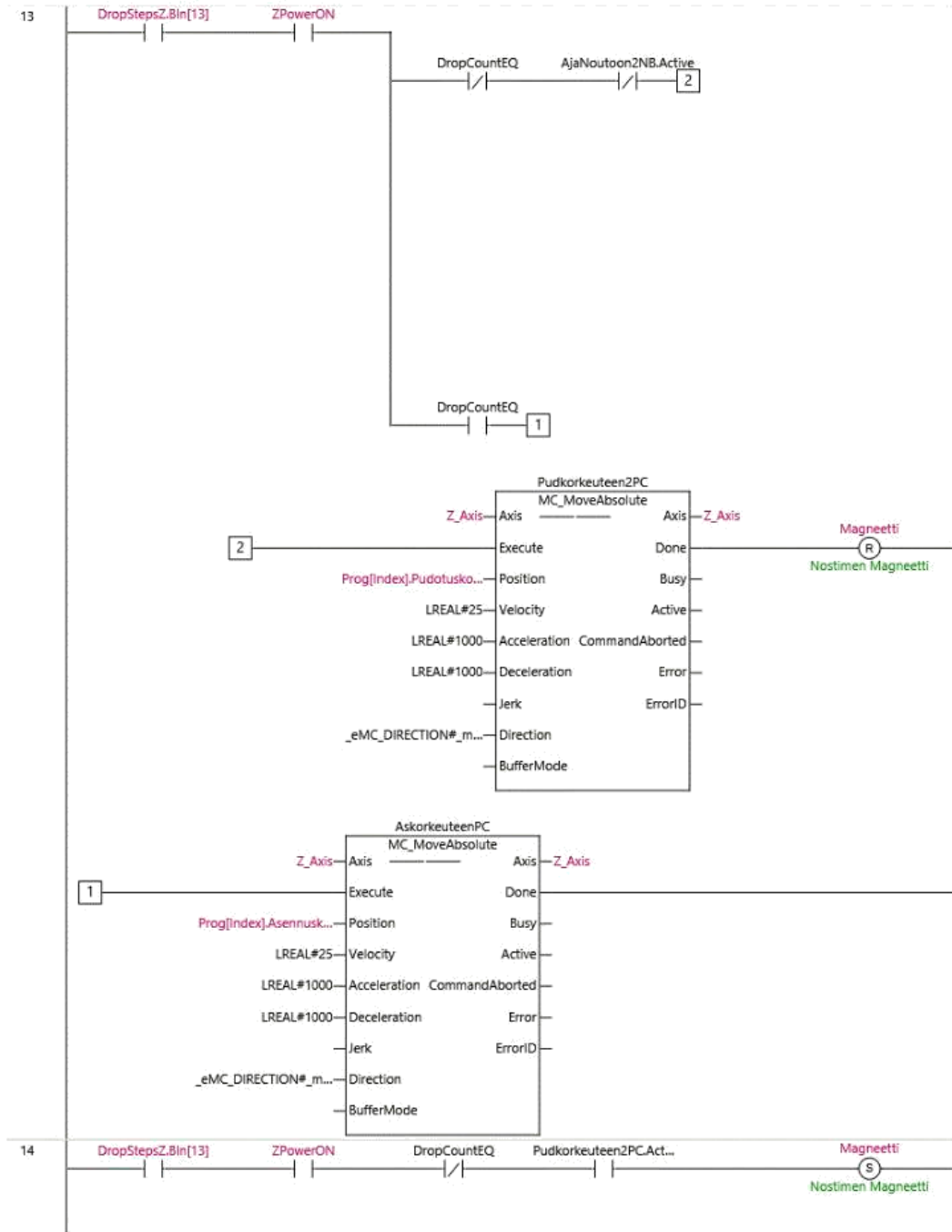




Droptester

new_Controller_0 DropPC

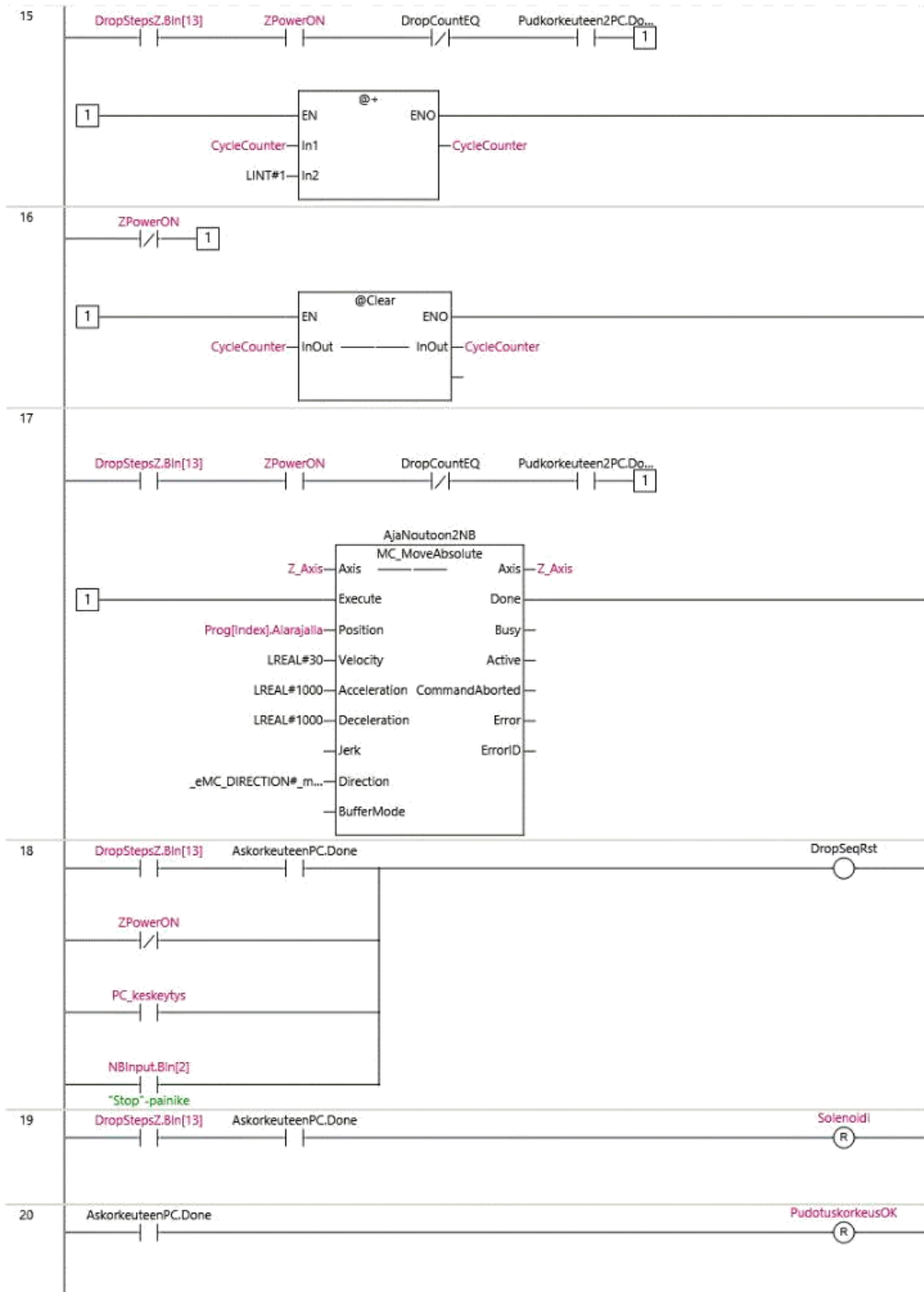
Main DropPC



Droptester

new_Controller_0 DropPC

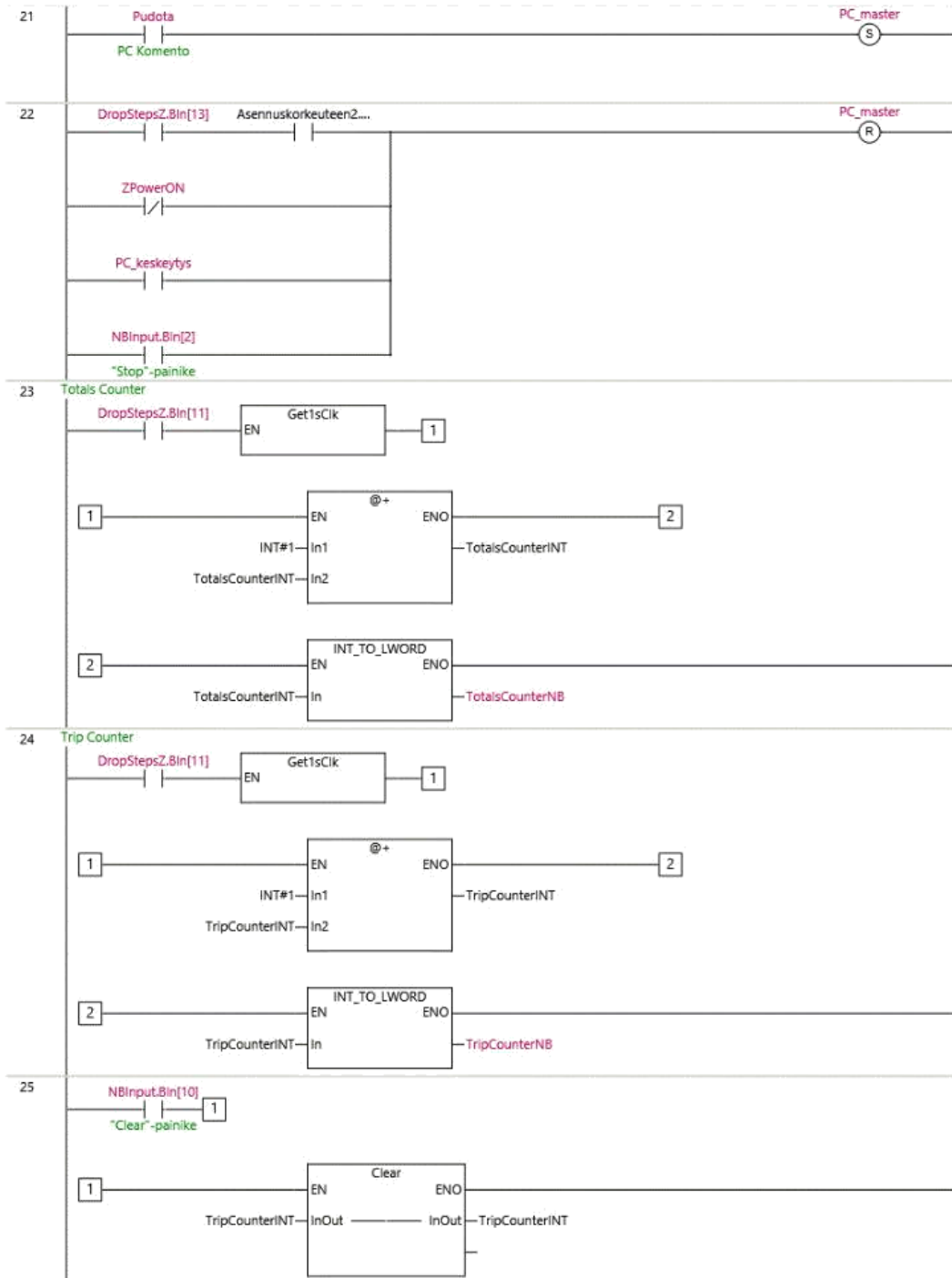
Main DropPC

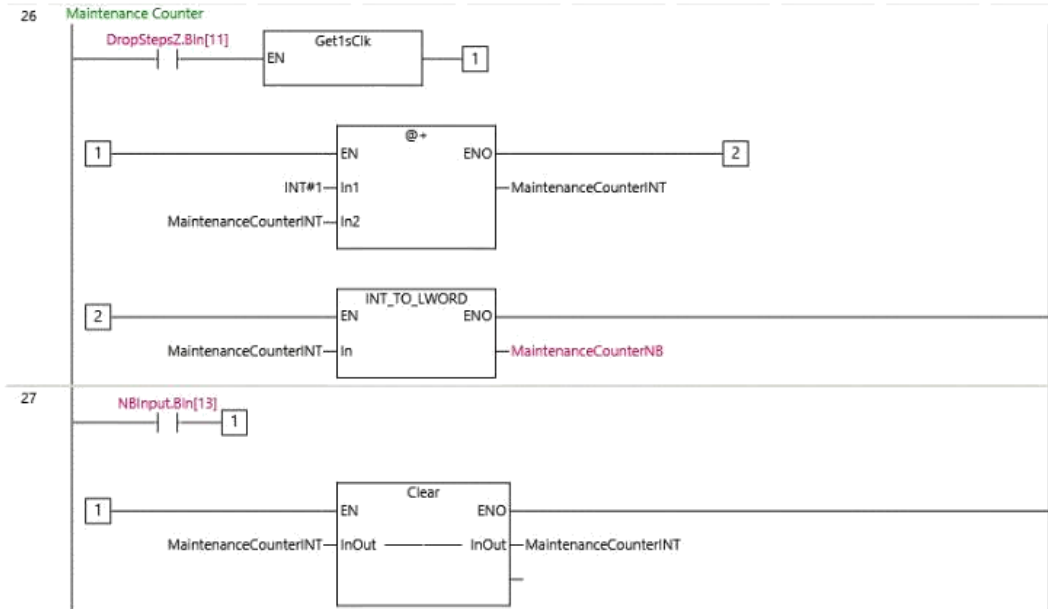


Droptester

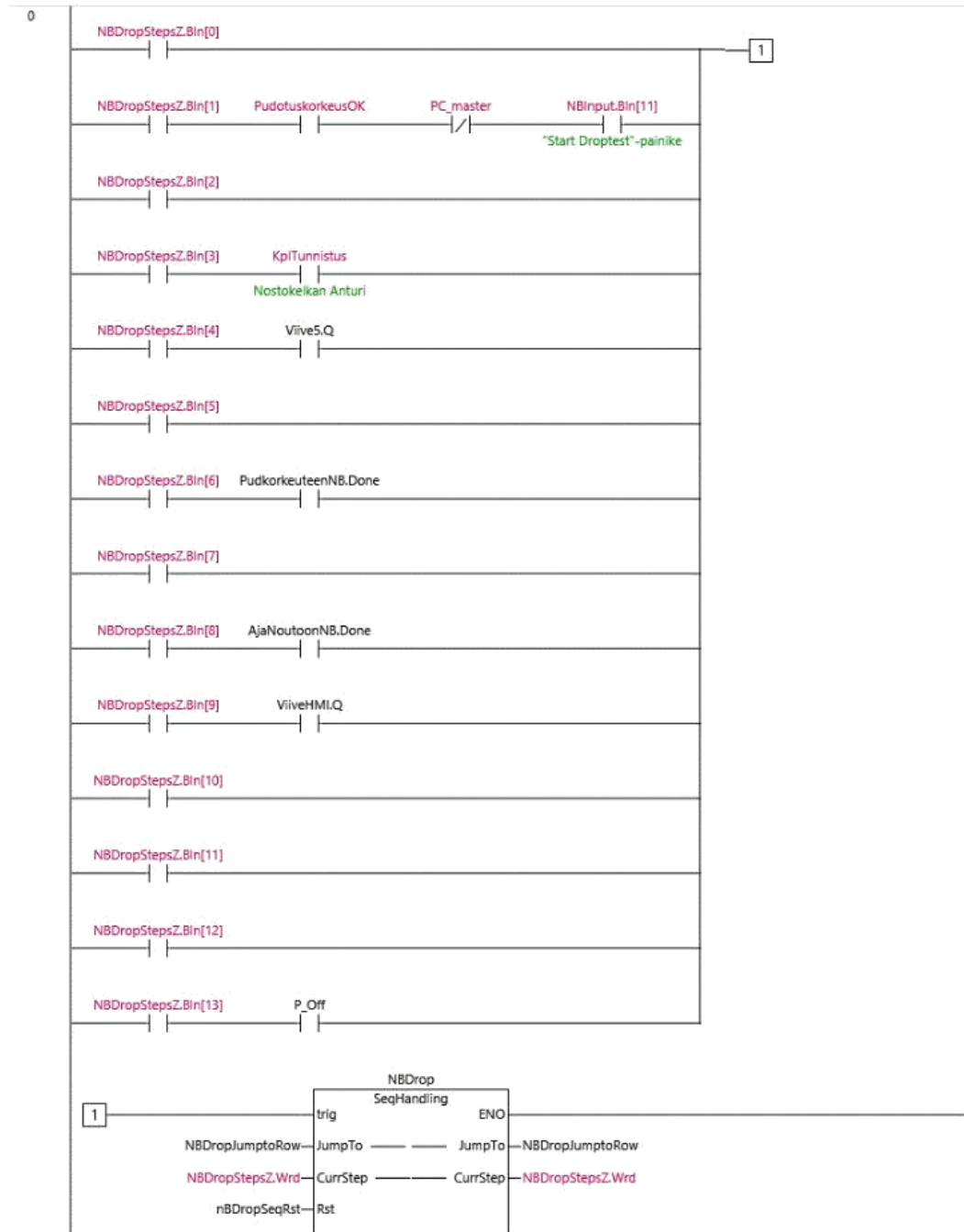
new_Controller_0 DropPC

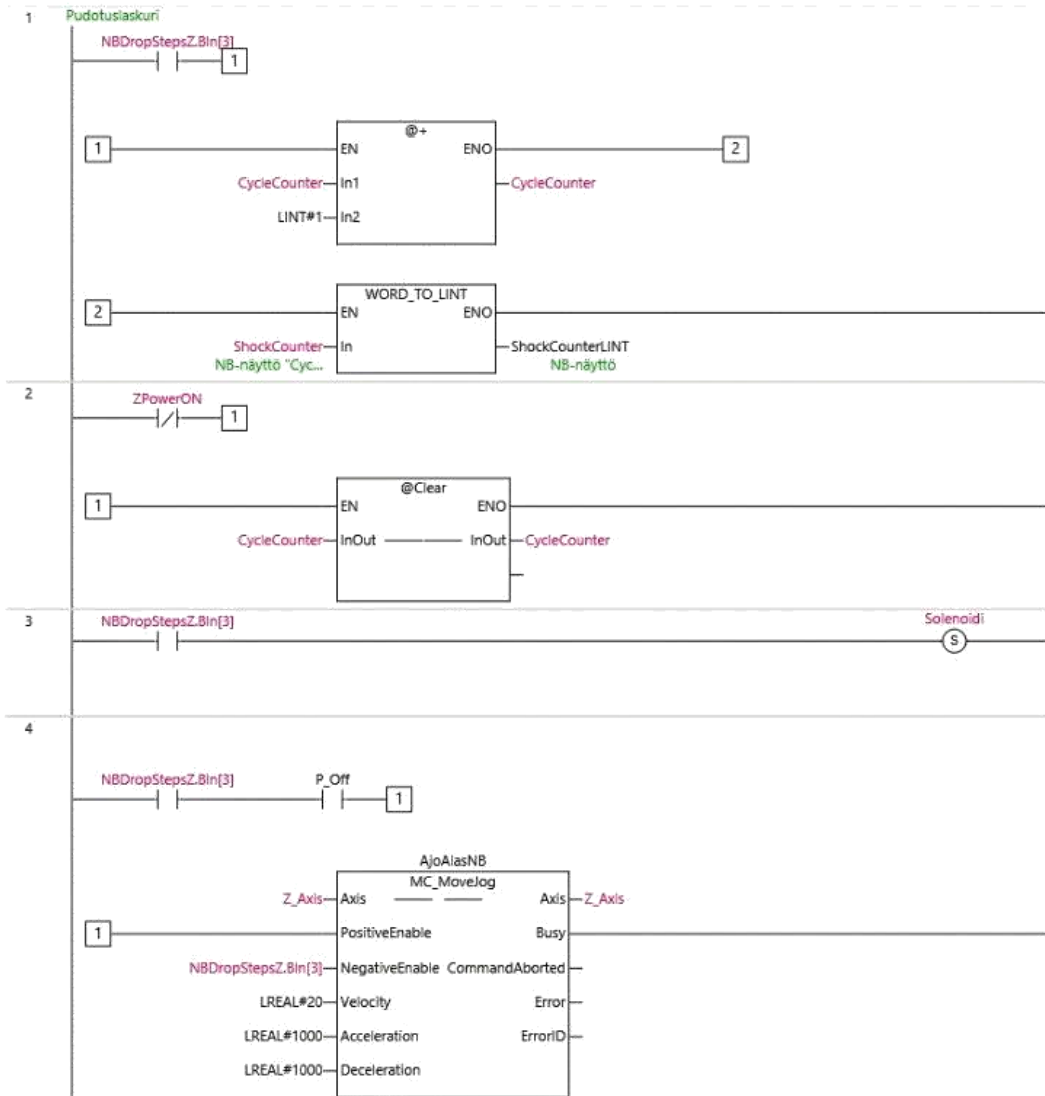
Main DropPC

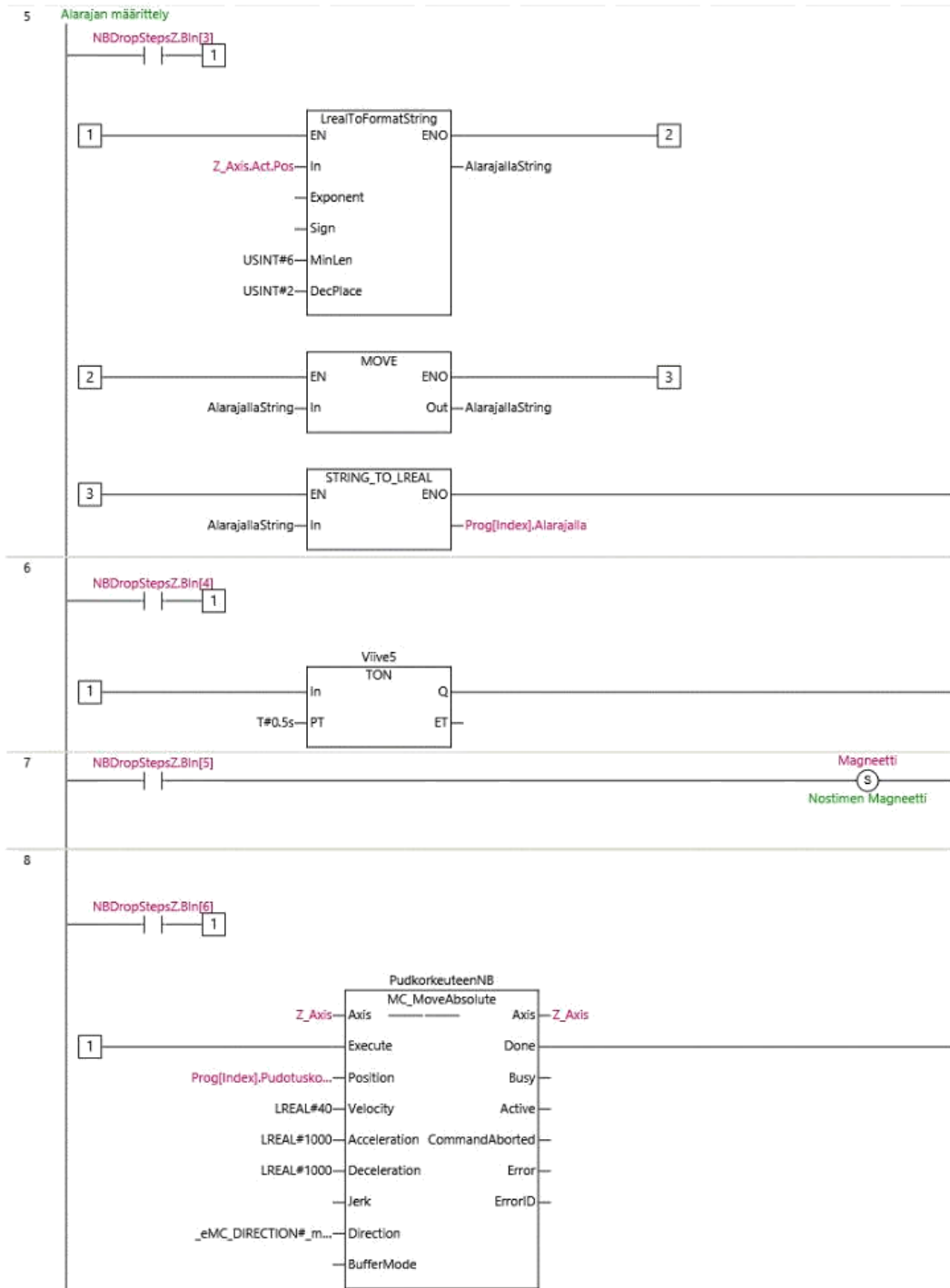


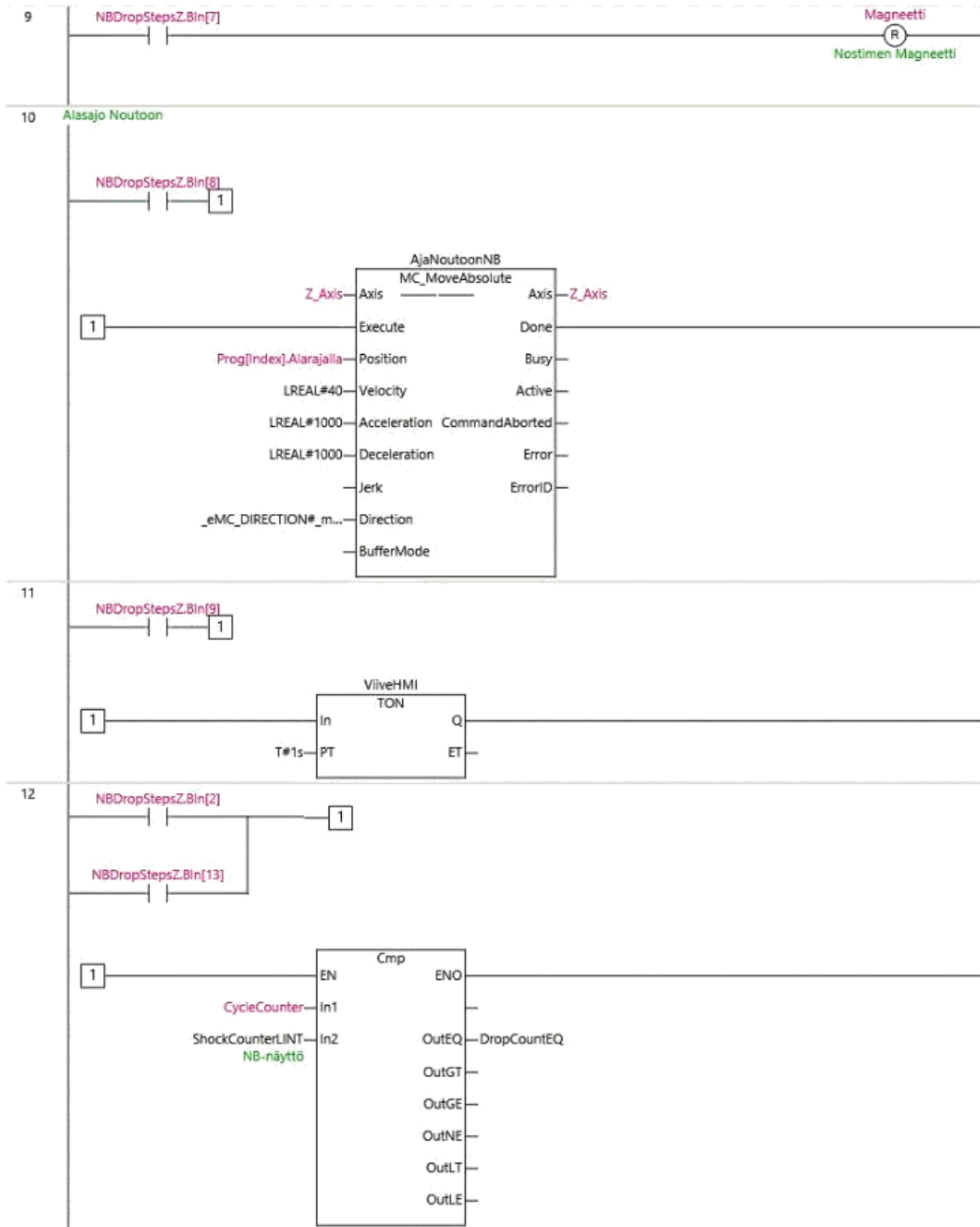


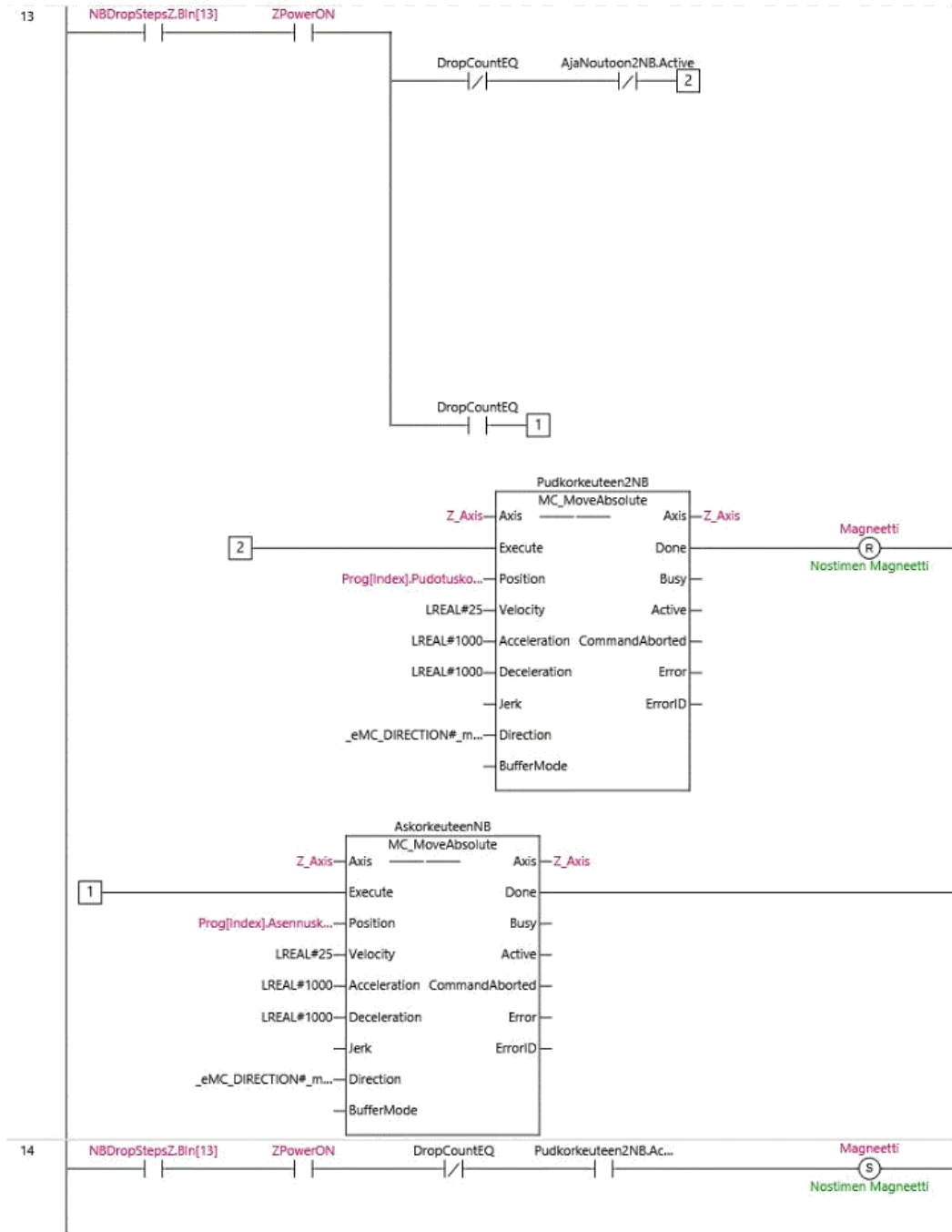
1-9-1-1-7.DropHMI







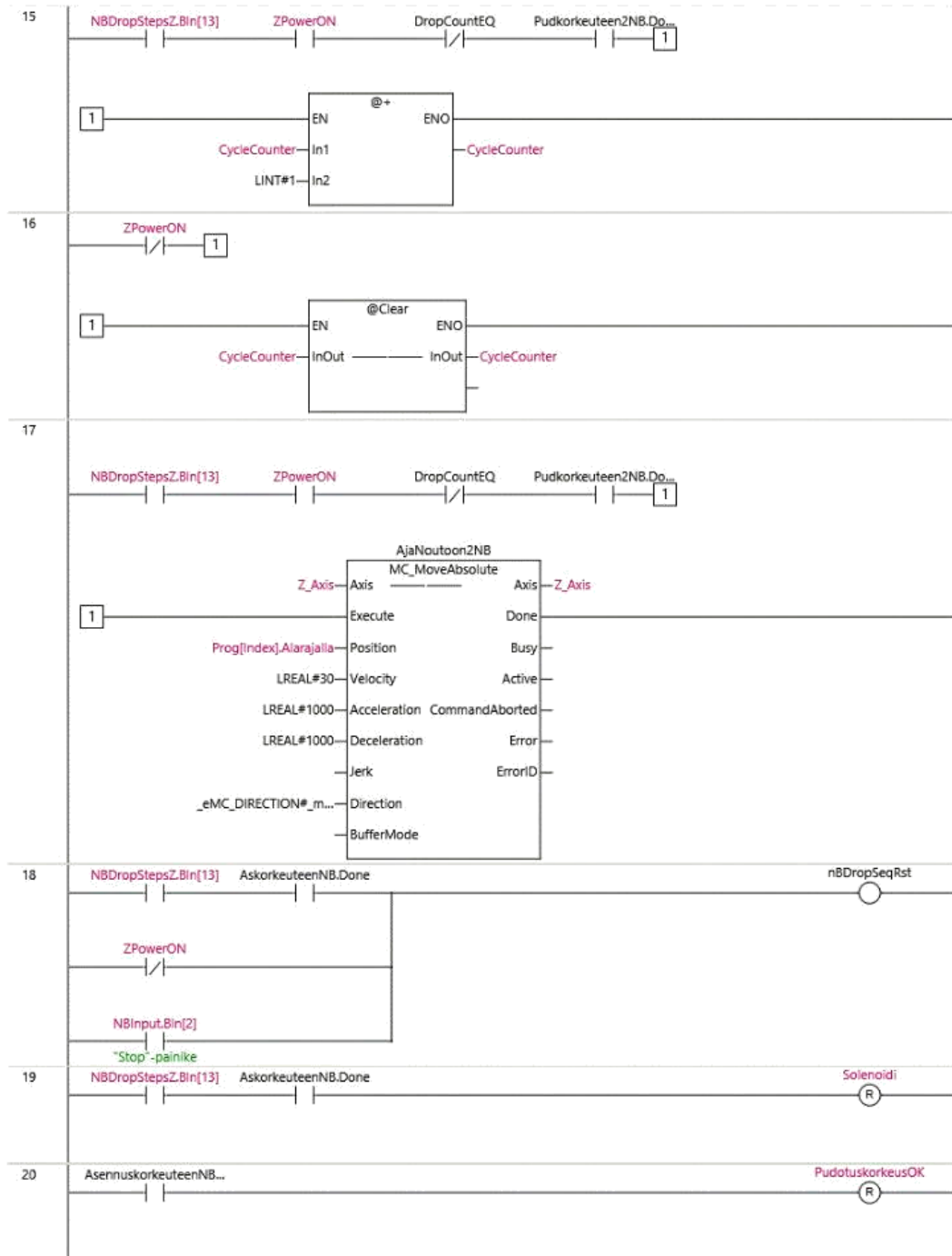




Droptester

new_Controller_0 DropHMI

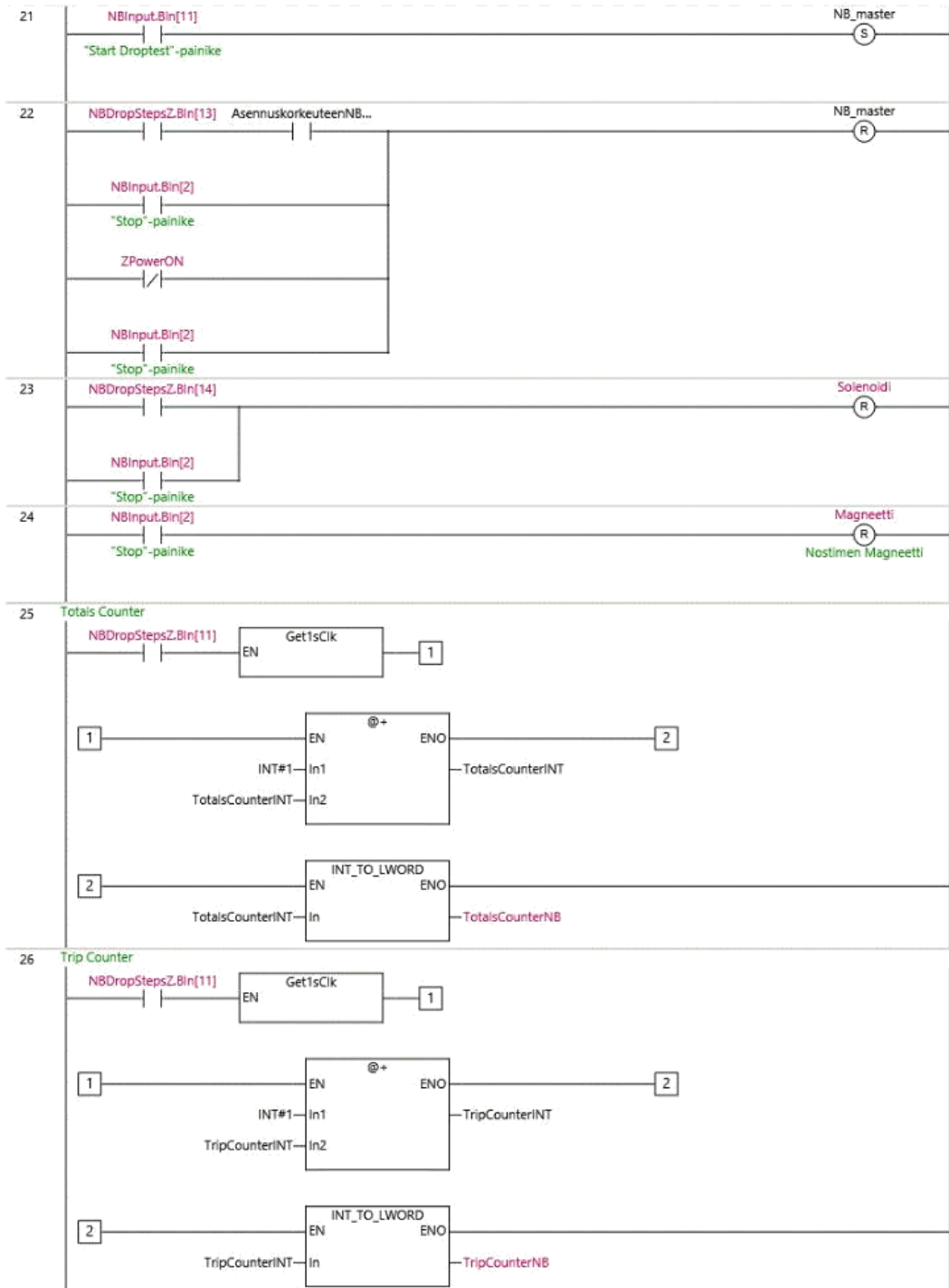
Main DropHMI

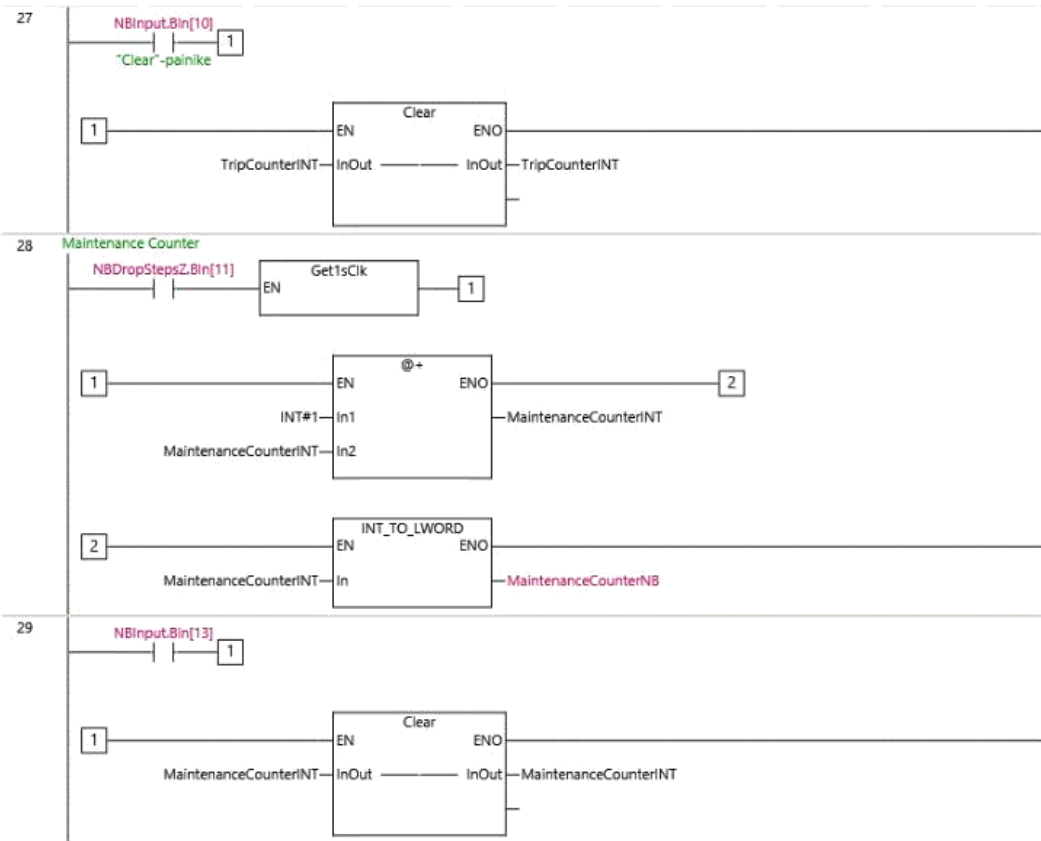


Droptester

new_Controller_0 DropHMI

Main DropHMI





1-9-1-1-8.Setup

