

Standard styrsystem för bandvågsprocesser

Henrik Hakala

Examensarbete för ingenjörsexamen (YH)

Utbildningsprogrammet för el- och automationsteknik

Vasa 2022

EXAMENSARBETE

Författare: Henrik Hakala
Utbildning och ort: El- och automationsteknik, Vasa
Inriktning: Automationsteknik
Handledare: Joachim Böling

Titel: Standard styrsystem för bandvågsprocesser

Datum: 12.4.2022 Sidantal: 46 Bilagor: 7

Abstrakt

Detta examensarbete handlar om skapande av ett standardstyrsystem för olika typer av bandvågsprocesser. Arbetet görs åt Electrocon Oy Ab i Kristinestad.

Syftet med arbetet var att skapa ett styrsystem för en vägningsprocess som kan väga material under förflyttning på ett bälte. Denna process ska ersätta den nuvarande processen som väger materialet i en temporär behållare. Produkten som skapas är en första prototyp och kommer att vidare utvecklas vid försäljning av bandvågsprocess.

Med styrsystemet ska användare kunna värma upp processens transportbälten, ange gränser och köra igenom angiven mängd material och kalibrera bandvågen. Utöver detta ska styrsystemet automatiskt övervaka processen, alarmera och logga fel. För ibruktagnin av bandvågen ska en manual skapas.

För styrsystemet används en Siemens S1200 PLC, Beijer X2 Pro HMI och Siemens WP241 bandvågsmodul. Programmet för PLC skapas i TIA-portalen som är en av de mest kända programmeringsmiljöerna, som programmeringsspråk används Function Block Diagram (FBD). För skapande av användargränssnitten på HMI används Beijers egna IX-developer program. Bandvågsmodulen tas i bruk med hjälp av Siemens SIWATOOL programvara.

För testningen av styrsystemet skapades en miniatyrprocess. Testprocessen består av ett litet transportbälte som körs med hjälp av en el-motor. Motorn är utrustad med en växellåda och hastighetsgivare. Hela processen bärs upp av två lastceller som används för vägningen.

Arbetets resultat blev en fungerande första prototyp som uppfyllde sina funktionella krav under tester på miniatyrprocessen med testmaterialet.

Språk: svenska

Nyckelord: bandvåg, PLC, HMI, styrsystem

OPINNÄYTETYÖ

Tekijä: Henrik Hakala
Koulutus ja paikkakunta: Sähkö- ja automaatiotekniikka, Vaasa
Suuntautumisvaihtoehto: Automaatiotekniikka
Ohjaaja: Joachim Böling

Nimike: Standardiohjausjärjestelmä hihnavaakaprosesseille

Päivämäärä: 12.4.2022 Sivumäärä: 46 Liitteet: 7

Tiivistelmä

Tämä opinnäytetyö käsittelee standardinmukaisen ohjausjärjestelmän luomista erityyppisille hihnavaakaprosesseille. Työ tehtiin Electrocon Oy Ab:lle Kristiinankaupungissa.

Työn tarkoituksena oli luoda punnitusprosessi, jolla voidaan punnita materiaalia hihnalla liikkumisen aikana. Tämä prosessi korvaa nykyisen prosessin, jossa materiaali punnitaan väliaikaisessa säiliössä. Luotu tuote on ensimmäinen prototyyppi ja sitä kehitetään edelleen hihnavaakaprosessin myynnin yhteydessä.

Ohjausjärjestelmän avulla käyttäjän tulee pystyä lämmittämään prosessin kuljetushihnat, asettamaan rajoja, ajamaan määrätyn materiaalmäärän ja kalibroimaan hihnavaakayksikkö, lisäksi ohjausjärjestelmän tulee automaattisesti valvoa prosessia sekä hälyttää ja kirjata virheet.

Ohjausjärjestelmä käyttää Siemens S1200 PLC:tä, Beijer X2 Pro HMI:tä ja Siemens WP241 -vaakayksikköä. Ohjelma PLC:lle luodaan TIA-portaalissa, joka on yksi tunnetuimmista ohjelmointiympäristöistä, ohjelmointikielenä käytetään Function Block Diagram (FBD). HMI-ohjelman luomiseen käytetään Beijerin omaa IX-developer-ohjelmaa. Hihnavaakayksikkö otetaan käyttöön Siemens SIWATOOL -ohjelmiston avulla.

Ohjausjärjestelmän testaamiseen luotiin pienoisprosessi. Prosessi koostui pienestä kuljetusvyöstä, jota ajetaan sähkömoottorin avulla. Moottori on varustettu vaihdelaatikolla sekä nopeusanturilla. Koko järjestelmä kannatetaan kahden punnituskenon päällä.

Työn tuloksena syntyi toimiva ensimmäinen prototyyppi. Ohjausjärjestelmäprototyyppi täyttää tälle asetetut toiminnalliset vaatimukset hyvin testauksessa pienoisprosessin ja testimateriaalin kanssa.

Kieli: ruotsi

Avainsanat: hihnavaaka, PLC, HMI, ohjausjärjestelmä

BACHELOR'S THESIS

Author: Henrik Hakala
Degree Programme: Electrical Engineering, Vasa
Specialisation: Automation
Supervisor(s): Joachim Böling

Title: Standard control system for belt weigher process

Date: 12.4.2022 Number of pages: 46 Appendices:7

Abstract

This thesis is about the creation of a standard control system for different types of belt weigher processes. The work is done for Electrocon Oy Ab in Kristinestad.

The purpose of the work is to create a weighing process that can weigh material during movement on a belt, this process will replace the current process that weighs the material in a temporary container. The product that is created is a first prototype and will further be developed when selling a belt weigher process.

With the control system, users should be able to heat the process's transport belt, set limits, drive through the specified amount of material and calibrate the belt weigher, in addition to this, the control system will self-monitor the process, alarm and log errors.

The control system uses a Siemens S1200 PLC, Beijer X2 Pro HMI and Siemens WP241 belt weigher module. The PLC application is created in the TIA-portal which is one of the most well-known programming environments. The programming language used is Function Block Diagram (FBD). For the creation of user interfaces on the HMI, Beijers own IX developer program is used. The belt weigher module is put into service using the Siemens SIWATOOL software.

For the testing of the control system, a miniature process was created. The test process consists of a small transport belt that runs using an electric motor. The motor is equipped with a gear box and a speed sensor. The whole process is placed on two load cells.

The result of the work was a featured first prototype that met the functional requirements well with the test process and test material.

Language: Swedish

Key words: belt weigher, PLC, HMI, control system

Innehållsförteckning

1	Introduktion.....	1
1.1	Syfte och mål.....	1
1.2	Avgränsning.....	2
1.3	Uppdragsgivare	2
2	Teori.....	2
2.1	Bandvåg	3
2.2	Styrsystem	4
2.3	Industriella PLC	4
2.4	Vågmodul	7
2.5	Industriella HMI	7
2.6	Hastighetsensor.....	8
2.7	Viktsensor.....	9
2.8	Frekvensomriktare	10
2.9	TIA-portal	10
2.9.1	Programmeringsspråk.....	11
2.9.2	Datatyper.....	12
2.9.3	Programmeringsblock	13
2.10	IX-developer	14
2.11	SIWATOOL.....	14
3	Bandvågsprocessöverblick	15
3.1	Designprocess	15
3.2	Flödesschema.....	16
4	Kommunikation.....	17
5	Ibruktagning av vågmodul.....	19
6	Lösningförslag.....	20
7	Detaljerad genomgång av system	22
7.1	Signalpresentation	22
7.2	Räkneoperationer.....	24
7.3	Start av systemet.....	27
7.4	Körning	28
7.5	Uppvärmning.....	31
7.6	Alarmering.....	33
7.7	Kalibrering.....	37
7.8	Indikationer	40
8	Testning.....	41
8.1	Testsekvens.....	42
9	Resultat.....	42

10	Diskussion.....	43
11	Källförteckning.....	45

Figurförteckning

Figur 1.	PLC-funktionsprincip. (Hanssen, 2015).....	5
Figur 2.	Minne i en PLC. (Hanssen, 2015).....	5
Figur 3.	PLC-körningssekvens.....	6
Figur 4.	HMI-process-struktur.....	7
Figur 5.	Induktiv givare som varvtalmätare. (Haag, 1998).....	8
Figur 6.	Wheatstonebrygga.....	9
Figur 7.	Användning av töjningsgivare. (Haag, 1998).....	9
Figur 8.	Wheatstonebrygga i L6E-lastcellen. (Zemic, 2020).....	9
Figur 9.	Frekvensomriktarens funktionsprincip. (Haag, 1998).....	10
Figur 10.	Ladder-programexempel.....	11
Figur 11.	FBD-programexempel.....	11
Figur 12.	Exempel på blockuppbyggnad. (Siemens, 2015a).....	13
Figur 13.	Olika typer av OB.....	14
Figur 14.	Designprocess.....	16
Figur 15.	Symbolbeskrivning.....	16
Figur 16.	Flödesschema för programstart.....	16
Figur 17.	Uppkoppling av hastighetsgivare. (Siemens, 2017).....	17
Figur 18.	Uppkoppling av lastcell. (Siemens, 2017).....	17
Figur 19.	Hastighetsstyrning.....	17
Figur 20.	Tvåledarestyrning.....	17
Figur 21.	Treledarestyrning.....	17
Figur 22.	Ihopkoppling av två Siemens moduler. (Siemens, 2018a).....	18
Figur 23.	HMI kontrollerinställningar.....	19
Figur 24.	PROFINET subnät i PLC.....	19
Figur 25.	WP241 funktionsblock.....	19
Figur 26.	WP241 bibliotek.....	19
Figur 27.	Displaystruktur.....	20
Figur 28.	PLC program uppbyggnad.....	21
Figur 29.	Main programmets körordning.....	22
Figur 30.	BVx Counter funktion.....	24
Figur 31.	BVx Weigher funktion.....	26
Figur 32.	Huvuddisplay.....	27
Figur 33.	Språkvalsfunktion.....	28
Figur 34.	Körningens parameterinställningar.....	28
Figur 35.	Körningsfönster.....	29
Figur 36.	Säkerställning av avbrytning av körning.....	29
Figur 37.	Pulsering av signal.....	30
Figur 38.	Process Feeder funktion.....	31
Figur 39.	BVx Feeder funktion.....	31
Figur 40.	Uppvärmningsdisplay.....	32
Figur 41.	Process Preheat funktion.....	32
Figur 42.	BVx Preheat funktion.....	32

Figur 43. I hopslagning av godkänd uppvärmning.....	32
Figur 44. Alarmlogg.....	33
Figur 45. Processalarmfunktion.....	34
Figur 46. BVx alarmfunktion del 2.....	36
Figur 47. BVx alarmfunktion del 1.....	36
Figur 48. Kalibreringsdisplay.....	37
Figur 49. Processkalibreringsfunktion.....	37
Figur 50. Bandvågsspecifik kalibreringsfunktion.....	38
Figur 51. Informationsdisplay.....	40
Figur 52 Processindikationsfunktion.....	40
Figur 53. Testprocess.....	41

Ordlista

HMI	Human Machine Interface
PLC	Programable Logic Controller
LD	Ladder
FBD	Function Block Diagram
OB	Object Block
FCB	Function Block
FC	Function
DB	Data Block

1 Introduktion

Materialvägning är ett viktigt skede i många industrier och därmed behövs många unika lösningar. En bandvåg är en våg som väger material på samma gång som det körs på ett bälte. Användningen av en bandvågsprocess är i många fall snabbare än att överföra allt material till en skild behållare för vägning. En bandvåg ser inte alltid lika ut, olikheter kan förekomma beroende på material som ska vägas och miljö där bandvågen ska användas, i vissa fall kan det även finnas behov av flera bandvågar.

En bandvågsprocess behöver naturligen någon form av styrning för att fungera. Detta dokument kommer handla om skapande av ett standardstyrssystem för bandvågar. Styrsystemet skapas på ett modulerat sätt så att dess implementering i olika typer av processer ska vara så lätt och smidigt som möjligt.

Detta dokument kommer innehålla en teoretisk del som innehåller viktig teori för systemet, så som vad ett styrssystem är, vad en bandvågsprocess är, vilken hårdvara som behövs för processen samt vilken mjukvara som behövs för skapande av olika processfunktioner. Efter den teoretiska delen följer en process- och kommunikationsöverblick där själva processen presenteras i detalj. I kapitlet om kommunikation introduceras hur all hårdvara kopplas upp och kommunicerar med varandra. Efter detta presenteras lösningsförslag, ibruktagning av bandvågen och en detaljerad systempresentation. I systempresentation, genomgås programmets olika delar i detalj. Till sist presenteras testningen för bandvågen och resultat som anhallits genom dessa tester, följt av en diskussion med bland annat förbättringsförslag och vidare arbete.

1.1 Syfte och mål

Målet med arbetet var att skapa en första prototyp av bandvågsstyrssystemet som sedan kunde användas för att snabbare kunna skräddarsy bandvågsprocesser till kundens behov, utan att varje gång behöva skapa ett nytt program.

Styrsystemet ska kunna styra två bälten, huvudmataren och bandvågsmataren. Användaren ska kunna styra processen från ett användargränssnitt, där hen bland annat kan aktivera uppvärmning av bälten, kalibrera bandvågen, granska och kvittera alarm samt ställa in gränser för hur mycket material som ska köras igenom bandvågen. Huvudmataren ska också ha två olika lägen, grovmatning och finmatning/pulsmatning. Vid grovmatningen kör huvudmataren snabbare ända tills genomkörda mängden närmar sig den önskade mängden, varvid läget byts till finmatning eller pulsmatning och det sista materialet körs igenom långsammare för noggrannare slutresultat. Finmatningen ska fungera på samma sätt som pulsmatningen, men med en långsammare hastighet. Pulsmatningen är en ersättande funktion för finmatningen, här används samma motor som för grovmatningen men i stället för att köra den konstant kommer motorn att pulseras. På samma sätt som för finmatningen är pulsmatningens uppgift att köra igenom material med ett noggrannare flöde.

Utöver själva systemet ska en manual för ibruktagnings av bandvågen skapas. Denna manual ska beskriva viktiga parametrar som eventuellt behöver ändras vid skapande av bandvågsprocesser för olika ändamål. Manualen ska också presentera de olika kalibreringarna av bandvågen samt hur dessa ska utföras.

1.2 Avgränsning

Detta arbete avgränsas till korrekt och modulär funktionalitet av de fyra huvudsakliga sekvensernas (alarmen, körnings, uppvärmnings och kalibrering). Det finns fortfarande ytterligare funktioner som bör läggas till och dessa kommer diskuteras i slutet av detta dokument. I och med att detta bara är en första prototyp kommer vi nöja oss med dessa sekvenser.

Funktionaliteten konstateras genom att gå igenom en testsekvens bestående av olika funktionella krav som ska uppfyllas i de olika lägen. Testsekvensen samt dess genomgång presenteras senare i dokumentet i mera detalj.

1.3 Uppdragsgivare

Electrocon är ett familjeföretag som grundades 1996 av Bjarne och Ann-Charlotte Grynngårds. Företaget började småskaligt i eget hem och fem år senare 2001 byggdes ett produktionsutrymme intill riksväg 8. År 2013 tog Tomas Grynngårds över ledarskapet och utnämndes till VD och leder företaget än idag. Nu består företaget av runt 17 anställd som jobbar i ett produktionsutrymme på 1200m². Företagets huvudsakliga arbetsområde är kablage tillverkning och fungerar därmed i huvudsak som underleverantör inom metall- och fordonsbranschen. Utöver kablage och kablagesystem jobbar också företaget med service, reparationer och kretskort.

På service- och reparationssidan erbjuder företaget bland annat felsökning och service av maskiner inom processbehandlingsindustrin samt skogsmaskiner, grävmaskiner och traktorer och reparation av generatorer. Företaget jobbar även med kända styrsystem som Omron, Siemens, Beijer och Crouzet samt stöder med reservdelsanskaffning. Inom kretskortsidan erbjuder företaget planerings, ritnings- och tillverkningstjänster. (Electrocon, n.d.).

2 Teori

I detta kapitel kommer teoretiska aspekter av arbetet behandlas. För att skapa ett styrsystem till någon process behövs en förståelse i vad ett styrsystem är och vad dess uppgift omfattar. Vi behöver även ha en god förståelse i hur själva processen samt dess teknologi fungerar.

Först presenteras hur en bandvåg fungerar och vilka beräkningar som behöver göras för att få fram olika storheter. Till näst behandlas vad ett styrsystem är och vilka uppgifter ett styrsystem tar hand om, efter detta presenteras hårdvaran och mjukvaran för detta projekt.

2.1 Bandvåg

En bandvåg är som tidigare beskrivits en process som väger material medan det körs på en transportör. En bandvåg är till grund och botten inte så olika från en butiksvåg eller en lastbilsvåg. Skillnaden ligger i krafterna som påverkar vägningsystemet, om vi förstår oss på dessa krafter är inte en bandvågprocess svårare att skapa än en vanlig plattformsvåg. En bandvågsprocess fungerar som en typ av integreringsapparat som till exempel en bils hastighetsmätning. I en bil mäts hastigheten och multipliceras med tiden för att få en representation i kilometer per timme. I en bandvåg mäts transportbältets hastighet och multipliceras med materialets vikt för att få fram massflödet. Problemet med att väga materialet genom en bandvågsprocess är att endast en del av massan rör sig på vägningsområdet, därmed kan ett litet fel vid skapandet eller kalibreringen av bandvågen resultera i stora fel i noggrannheten.

För en bandvågsprocess spelar flera olika aspekter en roll i den slutgiltiga noggrannheten. Till dessa aspekter hör processplacering, vägningsområde, bältinställningar och väderförhållanden.

Med processplacering menas i vilket skede av processen som vägning sätts in. Idealt ska vägningen placeras på en horisontell yta med många stödpunkter, om inte det ideala fallet är möjligt bör den bästa kompromissen hittas.

Vägningsområde är också en viktig aspekt i skapande av en bandvågsprocess. Vägningslängden anges ibland av bältleverantören men ibland är man själv tvungen att räkna ut den. Vägningsområdet är den längd som latcellen verkar på, oftast halvvägs till nästa bärande punkt på båda sidorna av cellen.

Bältinställningar är oftast en nyckelfaktor för en korrekt vägning, dvs. bältet måste vara av korrekt storlek för materialet, tillräckligt spänt och bra centrerat. Om bältet inte är av rätt storlek eller tillräckligt bra centrerat kan materialet börja färdas i ett spår vilket kan ge ett litet fel vid vägning. Bältets spänning måste också vara korrekt eftersom ett för löst bälte kommer öka på vägningsområdet och ett för spänt bälte kommer minska på det. Minskning eller ökning av vägningsområdet kommer åter leda till ett litet fel i vägningen och ett stort fel i slutresultatet. För bältspänning är det rekommenderat men inte obligatoriskt att använda en automatisk spännare (GTU). Om inte en GTU används ska bältspänningen granskas regelbundet.

Väderförhållanden spelar också in på processens noggrannhet. Denna aspekt påverkar processer som används utomhus mer än de som används inomhus. Vid temperaturförändringar påverkas bältets styvhet och detta kan leda till problem med hållbarhet och noggrannhet. Detta problem undviks genom att alltid värma upp bältet före användning samt utföra nollpunktskalibrering om nödvändigt.

I vissa fall är det fördelaktigt att installera flera vägningspunkter, dvs. flera lastceller. Användning av flera lastceller ökar på vägningsområdets längd och leder till bättre noggrannhet och repeterbarhet. (Scarrott, 2015).

I detta arbete används en bandvågsmodul, som sköter om tid- och räknefunktioner relaterade till bandvågsoperationer. Denna modul presenteras senare i dokumentet (kapitel 2.4).

2.2 Styrsystem

I dagens läge finns styrsystem överallt, i våra vardagliga hushållsapparater, transportmedel, till och med i våra egna kroppar. I hushållsapparater, som mikrovågsugnar och tvättmaskiner, är styrsystemen inbyggda. Transportmedel, som bilar, fungerar nuförtiden helt och hållet på basis av styrsystem och våra kroppar använder styrsystem för att reglera till exempel vår metabolism och balans. Styrsystem existerar alltså i olika former, mekaniska, elektroniska, biologiska och alla dessa har gemensamt att de styr något för att påverka ett slutresultat. Inom industrin används styrsystem nästan för varje maskin. Tack vare industrins eftersträvan av optimering av tillverkningslinjer har styrsystem utvecklats till en väldigt avancerad nivå.

Ett styrsystems huvudsakliga uppgift är att få maskineri att uppföra sig på ett visst sätt för att få ett visst slutresultat, detta gör systemen genom att manipulera processen med styrsignaler. (Iqbal, 2017).

För att skapa ett styrsystem behövs en del hårdvaror, för att få allt att fungera som planerat. För att få styrsystemet att fungera behöver man något som bestämmer och styr samt något som möjliggör mänsklig kommunikation med systemet. Utöver hårdvara behövs också mjukvara. En del av hårdvarukomponenterna är väldigt avancerade och behöver därmed en mjukvara för att realisera programmering av komponenten. Till följande kommer hårdvaran och mjukvaran för arbetet presenteras.

För hårdvaran beskrivs först PLC:n som är systemets bestämmande och styrande organ, efter detta följer vågmodulen som fungerar som ett tillägg till PLC:n. Efter detta presenteras HMI:n som fungerar som kommunikation mellan människan och styrsystemet. Till sist presenteras hastighetssensorn, viktsensorn och transformatorn. För varje apparatur anges en kort allmän funktionsbeskrivning följt av enhetens uppgift i styrsystemet.

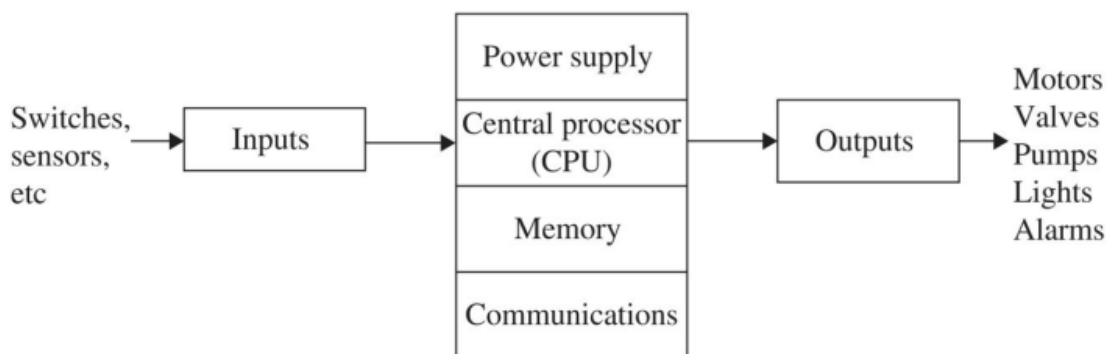
Efter hårdvaran presenteras mjukvaruprogrammen som används för skapande av styrsystemets olika delar. För PLC programmeringen används TIA-portalen som har en hel del aspekter vi behöver ta reda för att skapa ett program. För HMI-programmeringen används IX-developer och för ibruktagningen av bandvågen används SIWATOOL.

2.3 Industriella PLC

PLC eller Programmable Logic Controller är en efterträdare till relä-baserad styrning, där styrningen sköttes av relän, brytare, klockor och räknare. Denna typ av styrning var oftast väldigt klumpig, tog oftast mycket utrymme och var inte väldigt lätt att göra ändringar i. Den första PLCn kallades MODICON (Modular Digital Controller) och skapades till följd av att General Motors som sökte en bättre lösning till reläkontrollers. Språket som användes på denna PLC var Ladder, som baserade sig på reläkontrollernas kopplingsschema. Under den här tiden var PLC:n mycket enkel i och med att dess enda uppgift var att ersätta logiska kontrollers. Med tiden utvecklades diverse funktioner som analoga in- och utgångar samt aritmetiska funktioner. (Hanssen, 2015).

I dagens läge kan man göra nästan allt tänkbart med en PLC och de mera avancerade PLC:n kan jämföras med en dator. För programmering av PLC erbjuds idag ett brett sortiment av programmeringsmiljöer som TIA-portalen och Codesys. Dessa program stöder programmering av PLC genom att innehålla massor av färdiga funktioner, t.ex. matematiska funktioner, jämförelsefunktioner, kommunikationsfunktioner med mera.

PLC:ns funktion kan iaktas i blockschemat nedan (Figur 1). I figuren syns de huvudsakliga delarna för en PLC. En PLC består alltså av ingångar, utgångar, en kraftkälla, en CPU, minne och en kommunikationsmodul.



Figur 1. PLC-funktionsprincip. (Hanssen, 2015).

CPU:n kan sägas vara PLC:ns hjärna. Denna komponent sköter om att utföra instruktioner och kalkyleringar, kontrollera informationsflödet och programmets operation.

Minnet i PLC:n är oftast uppdelat i tre olika delar, ROM, RAM och FLASH EPROM, (Figur 2). ROM, vilket står för read-only memory, används för att lagra operativsystemet och systemdata. ROM-minnet är endast till för att läsas och går inte att manipuleras. På grund av detta används också EPROM, som står för erasable programable read only memory. Denna typ av minne möjliggör systemuppdateringar för operativsystemet. RAM-minnet som står för random access memory används för lagring av själva programmet. RAM-minnet är ett väldigt snabbt minne, men hålls endast aktiv medan PLC:n har spänning. För att undvika tömning av RAM-minnet vid spännings avbrott innehåller PLC:n ett batteri.

ROM	Operating system
	Data
Built-in RAM	Program
	Constants
FLASH EPROM	User program backup

Figur 2. Minne i en PLC. (Hanssen, 2015).

Kommunikationsenheten i PLC:n sköter om all datatrafik som ska in eller ut ur PLC:n. Denna trafik kan till exempel handla om uppladdning av själva programmet eller kommunikation med HMI. Kommunikationsenheten för en PLC är utrustad för att hantera trafik genom olika slags buss-protokoll, såsom AS-i bus, PROFIBUS, Modbus och CANbus. I detta projekt

används PROFINET, som fungerar genom att använda sig av TCP/IP-protokollet. Användningen av denna typ av kommunikation har i dagens läge blivit allt vanligare med PLC. (Hanssen, 2015).

Kraftkällan för PLC:n kan antingen vara en skild modul eller vara inbyggd i PLC, beroende på storlek och tillverkare. De flesta komponenter i PLC:n kör på 5V men på grund av praktiska skäl erbjuder tillverkare kraftkällor i olika storlekar från 220V AC till 24V DC.

Den sista delen av PLC:n är dess in- och utgångar. En PLC kan vara utrustad med olika in- och utgångar beroende på användningsändamål. De vanligaste typerna av in- och utgångar på en PLC är digitala och analoga, men utöver dessa finns många specialiserade typer, som termoelement och encoders. Varje in- och utgång har en unik adress som kan användas i programmet för kontroll av dessa portar.

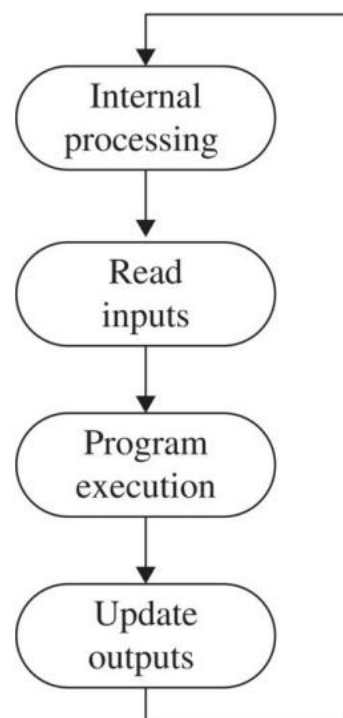
I vanliga fall har PLC:n en sekvens som körs igenom cykliskt (Figur 3), denna sekvens börjar alltid med att granska PLC:ns egna läge. Med att granska sitt eget läge menas t.ex. kontroll av I/O eller kommunikationsmodulen. Om någon av dessa enheter indikerar ett fel ställs en flagga hög. En flagga är en intern boolesk variabel som används för att indikera olika händelser, såsom fel i PLC:n. Om ett kritiskt fel hittas i detta skede kan PLC:n lägga sig i fel-läge och därmed avsluta normal operation.

Efter interna kontrollen läses ingångarna över till interna minnesplatser. Ingångarna läses inte in under körning av det tillverkade programmet utan sparas i dedikerade minnesplatser. Detta görs dels för att det är snabbare och dels för att undvika fel om samma ingång används på flera ställen i programmet.

Efter att ingångarna är lästa körs själva programmet. Här körs huvudcykeln men också övriga avbrott som programmerare definierat. I programkörningsskedet uppdateras interna variabler och utgångsminnen.

När programmet kört klart flyttas värden från utgångsminne till den fysiska utgången. I och med att utgången ändras först när programmet körts igenom, bör det noteras att utgångsminnen inte kan påverkas flera gånger i koden. Om detta görs kommer utgången ha det värde som designerats sist i koden. (Hanssen, 2015).

I detta arbete används SIEMENS S7-1200 som är en standard PLC i industrin. Denna typ av PLC kan köpas med olika typer av in- och utgångar. Modellen som används i detta arbete kommer utrustad med åtta digitala ingångar, två analoga ingångar, sex digitala utgångar och fungerar med en spänning på 24V.



Figur 3. PLC-körningssekvens.

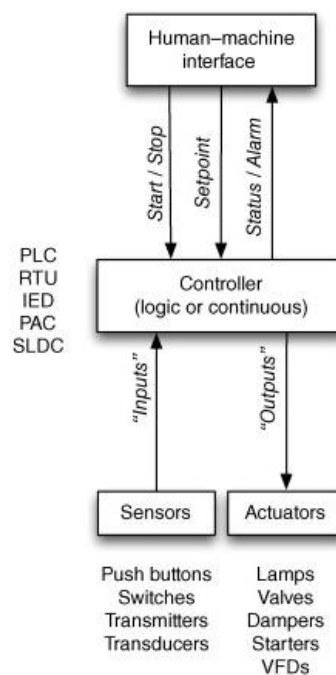
(Hanssen, 2015).

2.4 Vågmodul

I detta arbete används även en tilläggsmodul till PLC:n, som sköter om bandvågens signaler och viktrelaterade kalkyleringar. För bandvågsprocesser som använder sig av en S7-1200 PLC används Siwarex WP241 vågmodulen. WP241 bandvågsmodulen är skapad för funktion med Simatic S7-1200 PLC:n och har tillgängliga funktionsblock för implementering i TIA-portalen. Användning av en PLC är dock inte obligatoriskt, eftersom enheten också kan formas att fungera som en separat enhet. Modulen är utrustad med en kommunikationsenhet som stöder kommunikation via TCP/IP och Modbus RTU. Internt har modulen allt som kan tänkas behövas för övervakning av en bandvågsprocess, allt från flödesräknare till felkodsmeddelande. Enheten är också utrustad med programmerbara digitala in- och utgångar samt en analog utgång. (Siemens, 2017).

2.5 Industriella HMI

HMI, som står för human machine interface, är en term som i industrin används för hård/mjukvaran som tillåter användare kommunicera med olika typer av apparatur, såsom en PLC. En HMI ersätter behovet av knappar, vippor, vridrattar och märklampor och kombinerar alla dessa till en grafisk representation på ett användarfönster. HMI:n kan vara en hårdvara för sig, till exempel en touch-panel eller endast bestå av mjukvara som körs på en dator. HMI:n tillåter användaren att påverka processens olika moment såsom start, stopp och inställning av börvärden. Utöver kontroll förenklar HMI:n också övervakning av processen. Användare kan t.ex. granska felloggar, lägen på ventiler och övervaka processparametrar. En klassisk process struktur med HMI kan iakttas i Figur 4. **Fel! Hittar inte referenskälla..** (Knapp & Langill, 2015).



Figur 4. HMI-process-struktur.

(Knapp & Langill, 2015).

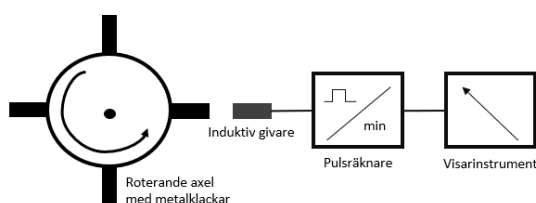
I detta projekt används en touch-panel tillverkad av Beijer. Panelen har många inbyggda funktioner, som alarm- och processloggar, färdiga grafiska symboler som knappar och märklampor. Panelen kan kommunicera genom RS232, RS422, RS485 eller en vanlig internetport (RJ45). Panelen är också utrustad med USB-portar som möjliggör uppkoppling av till exempel en mus för lättare manövrering eller en printer för utskrift av loggar.

2.6 Hastighetssensor

I industrin används många olika teknologier för hastighetsavkänning. En av dessa metoder använder en induktiv lägesgivare. Induktiva lägesgivare hör till digitala givarkategorin, vilket betyder att dessa givare antingen ger ut en hög- eller lågsignal. Givaren fungerar genom att ha en oscillator (en svängkrets) som återkopplas induktivt via ett mätobjekt. När ett elektroniskt ledande objekt, t.ex. en metall, kommer nära givaren försvagas oscillator kretsens amplitud. Denna förändring leder till att vi kan bedöma om ett metalliskt föremål är nära givaren.

Inom industrin används induktiva givare oftast för varvtalsmätning (Figur 5). Detta fungerar genom att ha metalklackar fästa på det roterande mätobjektet. Nära det roterande objektet fästs induktiva givaren som ger ifrån sig en högsignal när en metalklack passerar givaren. En pulsräknare används för att räkna antal höga signaler per minut och ett visarinstrument visar sedan varvtalet.

(Haag, 1998).



Figur 5. Induktiv givare som varvtalmätare. (Haag, 1998).

Induktiva givaren som används i detta arbete är OMRON:s E2A givare. Denna givare har ett mätningsområde på ca 2mm och är kopplad som en normalt öppen (NO) givare. Att givaren är av typen NO betyder att när givaren inte känner av ett objekt, är signalen låg och när ett objekt är inom avkänningsområdet är signalen hög.

I detta arbete används induktiva givaren på ett liknande sätt som vid varvtalsmätningen, men istället för att mäta antal varv per minut används givaren för att bedöma transportbandets hastighet. Genom att veta hur många kuggar det finns på hjulet samt diametern på bältets styrhjul, kan en pulskonstant räknas ut. Pulskonstanten berättar för oss hur många pulser givaren ger per meter. Pulskonstanten kan räknas ut genom ekvation 1. När vi vet pulskonstanten kan hastigheten lösas ut från pulser per sekund, ekvation 2.

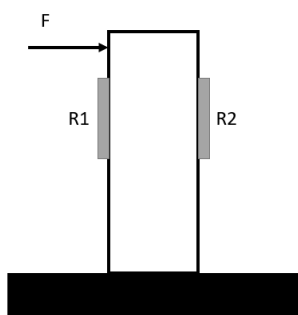
$$\text{Pulskonstant} = \frac{\text{Antal kuggar}}{\text{styrhjulets diameter} \cdot \pi} \quad (1)$$

$$\text{Hastighet} = \text{Pulskonstant} \cdot \frac{\text{pulser}}{\text{tiden i sekunder}} \quad (2)$$

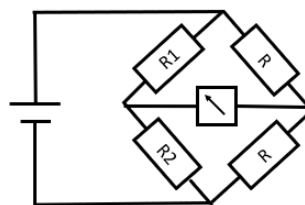
(Siemens, 2015b).

2.7 Viktsensor

I likhet med hastighetsgivaren har också vägning många olika lösningar. En av lösningarna är användning av en trådtöjningsgivare. En trådtöjningsgivare består av en mycket tunn motståndstråd, när tråden utsätts för töjning minskar dess tvärsnittsarea och resistansen ökar. Denna typ av givare limmas oftast på mätobjektet i form av en tunn folie. När mätobjektet utsätts för en kraft påverkas tråden och kraften kan mätas (Figur 7). Ofta kopplas denna givare upp i form av en wheatstonebrygga (Figur 6). Med denna typ av koppling används två töjningsgivare och två fasta motstånd. En spänningskälla kopplas upp i motsatta hörn och spänningen mäts från de två återstående hörnen. På detta sätt får man en noggrann mätning på resistansförändringen.



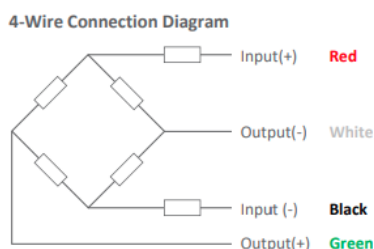
Figur 7. Användning av töjningsgivare.
(Haag, 1998).



Figur 6. Wheatstonebrygga.
(Haag, 1998).

Inom industrin läggs ofta en trådtöjningsgivare in i en stålcylder, vilket kallas för lastcell. Genom att applicera trådtöjningsgivaren i en metallcylder kan man motverka givarens känslighet för temperaturförändringar och få en noggrannare mätning. (Haag, 1998).

I detta arbete används L6E som är en lastcellstyp av tillverkaren Zeimic. Denna lastcell har max vägningskapacitet på 30kg och en noggrannhet på 2.0 ± 0.2 mV. Denna lastcell är uppkopplad i en Wheatstonebrygga och använder sig av fyra ledarsystemet (Figur 8). (Zemic, 2020).

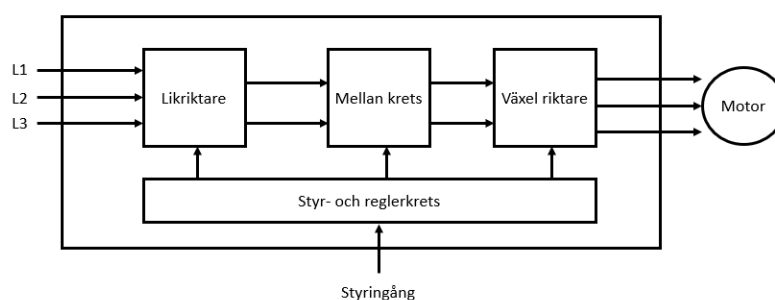


Figur 8. Wheatstonebrygga i L6E-lastcellen.
(Zemic, 2020).

2.8 Frekvensomriktare

I dagens läge sker varvtalsstyrning av växelmotorer med frekvensomriktare. Denna typ av apparatur förändrar nätets fasta frekvens och spänning till en steglös variabel storhet för motorn. Funktionen för en frekvensomriktare är väldigt komplicerad och därför kommer dess funktioner endast kort beskrivas.

Frekvensomriktarens generella funktionsprincip beskrivs i Figur 9. I likriktaren omformas växelspänningen till en likspänning, denna spänning är dock inte helt stabil och därför filtreras den i mellankretsen. Före spänningen skickas till växelriktaren omformas den till en ny växelspänning med justerbar frekvens. Den justerbara frekvensen är proportionell mot motorns varvtal. Styr- och reglerkretsen håller koll på övriga komponenter så att förhållandet mellan spänning och frekvens hålls konstant. (Haag, 1998).



Figur 9. Frekvensomriktarens funktionsprincip. (Haag, 1998).

I detta projekt används Schneiders ATV12 frekvensomriktare. Denna frekvensomriktare fungerar på en fas, för motorer upp till 0,37 kW och har ett ställbart frekvensområde på 0,5 till 400 Hz. (Schneider Electric, 2016).

2.9 TIA-portal

TIA-portal står för Totally Integrated Automation Portal och är Siemens egen programvara. Integrerat i programvaran är SIMATIC STEP 7, vilket är Siemens programmeringsverktyg för industriell automation. I Tia-portalen kan man välja mellan olika programmeringsspråk som följer IEC 61131-3 standarden.

Simatic WinCC är Siemens programmeringsverktyg för HMI-paneler vilket också inkluderas med i TIA-portal. Simatic WinCC används för att skapa användargränssnitt som passar processer som sen kan bli styrda från HMI-panelen. Simatic WinCC fungerar endast för Siemens egna HMI paneler och kan inte tillämpas för andra leverantörers paneler.

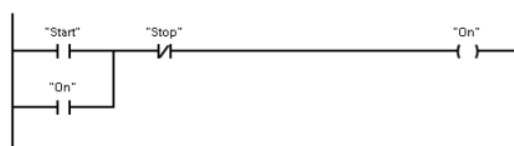
I en SIMATIC-PLC finns det ett operativsystem och ett användarprogram. Användarprogrammet är uppbyggt av fyra olika blocktyper, organisationsblock (Organization block, OB), funktionsblock (Function block, FB), funktioner (Function, FC) och datablock (Data block, DB). Dessa block typer behandlas vidare i kapitel 2.9.3.

Som med de flesta program uppdateras TIA-portalen regelbundet och ibland infogas en större uppdatering, detta leder till en helt ny version av programmet. I dagens läge finns V17 men för detta projekt används V16 på grund av bakåtkompatibilitet. Det är lättare att skapa programmet i en tidigare version och sedan vid behov uppdatera programmet till en nyare version. (Siemens, n.d.b).

2.9.1 Programmeringsspråk

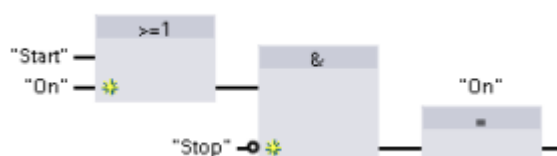
I dagensläge, används inom industrin fem olika programmeringsspråk, när ett PLC-program ska skapas. Dessa språk har fått gemensamma spelregler från IEC-61131-3 standarden, så att de ska ha en likadan struktur runt hela värden. Språken är Structured Text (ST), Sequential Function Charts (SFC), Ladder Logic Diagram (LD), Function Block Diagram (FBD) och Instruction List (IL). De tre överlägset vanligaste språken är ST, LD och FBD.

Ladder är det äldsta grafiska programmeringsspråkets som ännu används. Programmet skapas genom att dra kopplingar mellan olika element som NO och NC kontakter samt Coils. Ladder erbjuder också boxoperationer som används för mera komplicerade operationer som matematiska-, tidsbaserade-, räknebaserade- och konverterandeoperationer. Ett exempel på ett simpelt program kan iaktas i Figur 10. (Siemens, 2015a)



Figur 10. Ladder-programexempel.

Function Block Diagram är också ett grafiskt programmeringsspråk som kan sägas vara Ladders efterträdare. Detta språk bygger på användningen av olika block som utför diverse uppgifter. Denna typ av programmering möjliggör skapande av mycket mera komplicerade program på ett lätt förståeligt sätt. I Figur 11 ser vi ett samma programexempel som för Ladder men nu i FBD. (Siemens, 2015a)



Figur 11. FBD-programexempel.

Structured text som i TIA-portalen kallas Structured Control Language är ett högnivåtextbaserat programmeringsspråk, vars syntax påminner om C eller C++. Ett textbaserat språk har inga grafiska symboler utan allt baserar sig på instruktioner. Programmet består oftast av så kallade conditional statement till exempel IF och CASE

statement med en eller flera instruktioner. Utöver detta finns även loopar som FOR, WHILE och REPEAT. Samma funktion som för Ladder och FBD i ST kan iakttas i Kodexempel 1. (Siemens, 2015a).

Kodexempel 1. ST-programexempel.

```
IF Start=True AND Stop=False THEN
On:= True;

ELSEIF On=True AND Stop=False THEN
On:=True;

ELSE
On:=False;

END_IF;
```

I detta projekt stod valet av programmeringsspråk mellan Ladder, FBD och ST. Ladder uteslöts på grund av att detta programmeringsspråk är föråldrat och inte känns lämpligt för ett program av denna kaliber. ST uteslöts på grund av dess komplexitet och krav på god programmeringskunskap för att skapa och uppdatera programmet. Detta ledde till att FBD valdes på grund av dess möjlighet för komplicerade operationer men ändå lättförståelig struktur.

2.9.2 Datatyper

I PLC-program rör sig data som används för olika syften, denna data har olika former beroende på dess användningsändamål. Det är mycket viktigt att förstå datatypen som används eftersom det mycket lätt kan skapas problem vid felanvändning av datatyper.

De vanligaste data typerna är: Bool, Integer, Real, Character, String och Array. I detta arbete kommer ytterligare datatypen Time att användas. Alla datatyper tilldelas en viss mängd databitar för att kunna representera deras innehåll.

Bool är en datatyp som används för att representera sanna eller falska påståenden och tilldelas en databit vilket betyder att en bool variabel kan antingen vara 1 eller 0 (sant eller falskt).

Integer används för att representera heltal och är antingen signed eller unsigned. Skillnaden mellan signed och unsigned är möjligheten att representera negativa tal. Med unsigned representeras endast positiva tal och med signed kan också negativa tal representeras. För integer av typen signed är alltså hälften av representabla talen på negativa sidan. Ytterligare finns tre typer av integer, short integer (SInt), integer (Int) och double integer (DInt). Dessa typer tar upp olika mycket minne, Sint tar upp åtta bitar (en byte) och kan representera 256 tal, Int tar upp 16 bitar (två byte) och kan representera 65 536 tal och Dint tar upp 32 bitar (fyra byte) och kan representera 68 719 476 736 tal. (Siemens, 2015a).

Real, ibland kallad float, används för att representera decimaltal. Denna datatyp finns i två storlekar Real och LReal, Real upptar 32 bitar (fyra bytes) och LReal upptar 64 bitar (åtta bytes). Med Real kan tal mellan $-3.40e+38$ till $3.40e+38$ representeras och för LReal $-1.79e+308$ till $1.79e+308$. (Siemens, 2018b).

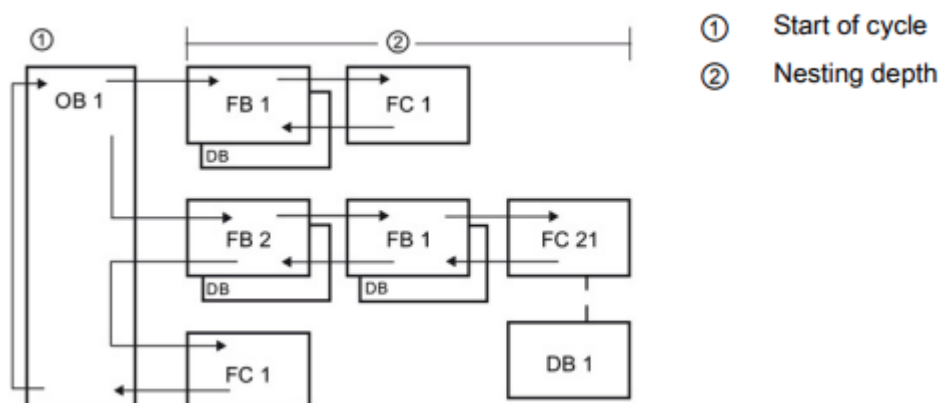
Character (Char) och string är datatyper som används för att representera bokstäver och symboler. Skillnaden mellan dessa två variabeltyper är att char används för att representera en karaktär medan string används för att representera en sträng av karaktärer, till exempel ett ord eller mening. För Char reserveras åtta bitar (en byte) och för string reserveras 256 bytes vilket betyder att man med string kan representera 256 karaktärer. (Siemens, 2018b).

En array är en multi-dimensionell gruppering av samma typs data. Till exempel `ARRAY[1..5,1..3] INT` bygger upp en tabell av storleken 5x3 med data av typen Integer. Datan kan nås genom att använda ett indexvärde som kan vara allt mellan -32768 och 32767 . Till exempel `ARRAY[2,2]` skulle ge oss värdet på andra raden i andra kolumnen. Användning av array gör det lättare att gruppera större mängder data som hör ihop med varandra på ett eller annat sätt.

TIME datatypen är exakt likadan som DInt datatypen, men är dedikerad att visa tiden i millisekunder. Denna data typ används bland annat för att styra on- och off-delays.

2.9.3 Programmeringsblock

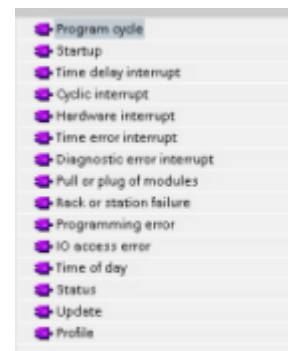
Som tidigare nämnts bygger programmeringen i TIA-portalen på att bygga en struktur av de fyra olika blocken; Organization block (OB), Function block (FB), Functions (FC) och Data block (DB). Programmet byggs genom att ha ett OB som huvudblock. Detta block kan sedan t.ex. köras med vissa intervaller. Inne i OB finns då FC och FB block som innehåller själva programkoden. Utöver programblocken används också globala DB:n, där viktiga variabler vars värden behövs vid nästa cykel sparas. En exempeluppbyggnad illustreras i Figur 12. Fördelen med att använda dessa block är att programmet får en mycket tydlig struktur. Funktioner som ofta måste användas kan lätt kallas på nytt samt felsökningen av programmet och processen förenklas. (Siemens, 2015a).



Figur 12. Exempel på blockuppbyggnad. (Siemens, 2015a).

De fyra nämnda blocken (OB, FB, FC och DB) fungerar på aningen olika sätt och är ämnade för olika ändamål, till följande beskrivs dessa blocks användningsändamål.

OB:n är programmets huvudblock, som kallas av operations- och kontrollsystemet. Enligt Siemens programmeringsinstruktioner beskrivs OB som ett gränssnitt mellan operativsystemet och användarprogrammet. OB:n är ett händelseorienterat block, dvs. blocket körs vid en viss händelse. En händelse kan vara bland annat olika avbrott, tidsintervall eller programcykel. Utöver de uppräknade typer finns många olika funktioner att välja mellan (Figur 13).



Figur 13. Olika typer av OB.

I FC och FB skapas oftast programmets funktioner, alltså den huvudsakliga koden. Skillnaden mellan dessa två block är att FC använder temporära minnet medan för FB skapas ett specificerat DB. Oftast används FC för enkla funktioner som används ofta i programmet, t.ex. räkneoperationer. FB används däremot för uppgifter som inte blir klara vid en programcykel utan värden behöver sparas till nästa cykel.

DB är block ämnade för sparande av data. Datablock skapas automatiskt för FB:n men utöver dessa kan även globala DB:n skapas. Dessa block är skapade för att spara data som hela programmet använder, vilket betyder att DB:n kan kontaktas från programmets alla delar. Globala DB:n kan även användas för kommunikation med andra moduler, t.ex. HMI:n eller andra PLC:n. (Siemens, 2015a).

2.10 IX-developer

IX Developer är en grafisk programmeringsmiljö för Beijer HMI paneler. Med IX-developer bygger man upp själva användargränssnittet, kopplar ihop HMI panelen med olika apparater, skapar dataloggning samt alarmservrar. Detta verktyg är en mycket simpel och användarvänlig programmeringsmiljö. Programmet möjliggör också att skapa användargränssnittet på textnivå, dvs. genom att skriva kod på vad som ska hända. IX-Developer är Beijers egna program och kan endast användas för att skapa användargränssnitt till Beijer-HMI:n. (Beijer, n.d.).

2.11 SIWATOOL

Siwatoool är en programvara skapad för Siemens bandvågs moduler. Programmet används för att effektivt kalibrera modulen, ställa om parametrar samt utföra diagnostik eller felsökning på plats. Programmet möjliggör skapande av backup filer, som smidigt kan laddas upp till nya vågmoduler. Backupfilerna innehåller i detta fall redan alla kalibreringstal, vilket betyder att ingen kalibrering behöver utföras vid byte av vågmodul. Felsökningen med programmet fungerar genom att ha en logg som läser in alla felkoder till användare. Denna logg kan även skrivas ut till PDF för vidare undersökning. Programmet visar upp all data i tre kolumner, en titelkolumn, en kolumn som visar nuvarande värde på datorn (användarfilen) och sista kolumnen som visar värdet i vågmodulen, om vågmodulen är uppkopplad. (Siemens, n.d.a).

3 Bandvågsprocessöverblick

För detta arbete gäller två egentliga processer, en huvudsaklig designprocess och en testprocess. Eftersom detta arbete handlar om skapande av en första prototyp har inte en fullständig process tillverkats, utan för testning har en mindre testprocess skapats. De största skillnaderna mellan design- och testprocessen är dess storlek och innehåll av matartravers. Till följande kommer designprocessen presenteras.

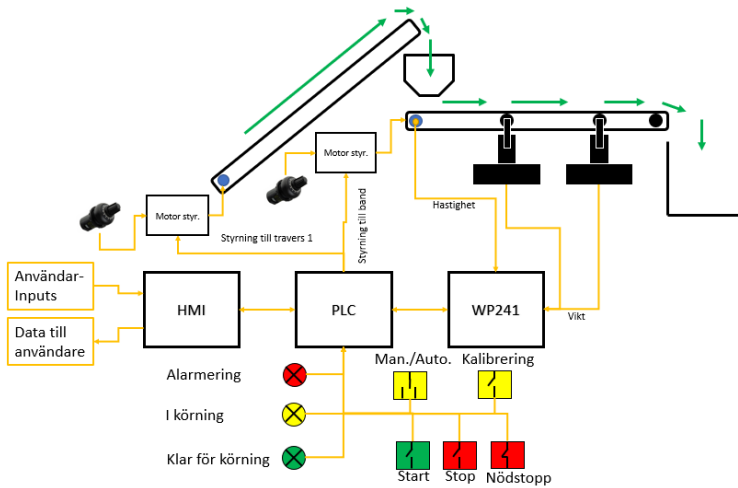
3.1 Designprocess

Designprocessen är den ursprungliga processöverblicken som programmet skapats på basis av. Denna process består av styrsystemet, två transportband med respektive styrningskomponenter, en materialtratt, knappar och indikationsbelysning. Processöverblicken kan iaktas i Figur 14 med symbolbeskrivning i Figur 15.

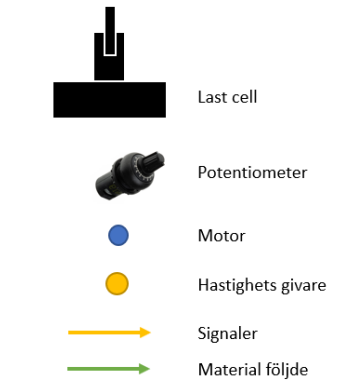
I figuren ser vi styrsystemet på nedre halvan av figuren, styrsystemet består av HMI, PLC och WP241 bandvågsmodulen. Till HMI tas diverse användarsignaler in och viktiga parametrar ges ut för styrning av processen. HMI:n är kopplad till PLC:n med PROFINET kommunikation. PLC:n tar in signaler från de olika knapparna och ger ut signaler för indikationslampor samt styrning av frekvensomriktare. Bandvågsmodulen WP241 tar emot hastighetsdata från hastighetsgivaren och viktdata från lastcellerna. Modulen kommunicerar med PLC över en buss-linje.

Själva processen består av två traverser med varsin motor och frekvensomriktare. Bandvågs bältet som är bältet till höger består av två lastceller som bär upp bältet på mitten samt en hastighetsgivare som mäter bandvågens hastighet. Hastigheten på bälten ska kunna ställs in med en potentiometer som är kopplad till frekvensomriktaren. Materialflödet i processen är ämnat att först köras upp med den första traversen och sedan fällas ner genom en materialtratt. Materialtrattens uppgift är att skapa ett mera stabilt materialflöde till bandvågsbältet. Materialet faller då ner på bandvågsbältet där det vägs vart efter det körs ner till en packningslåda.

Denna process är ett exemplariskt användningsområde som programmet kunde tänkas användas för. Men eftersom olika kunder har olika krav på processer stärker detta behov av ett program som är lätt att tillämpa. Till exempel kunde här, i stället för en materialtratt finnas en sorterare, som fördelar materialet ner på två olika bälten som i sin tur väger och kör materialet separat.



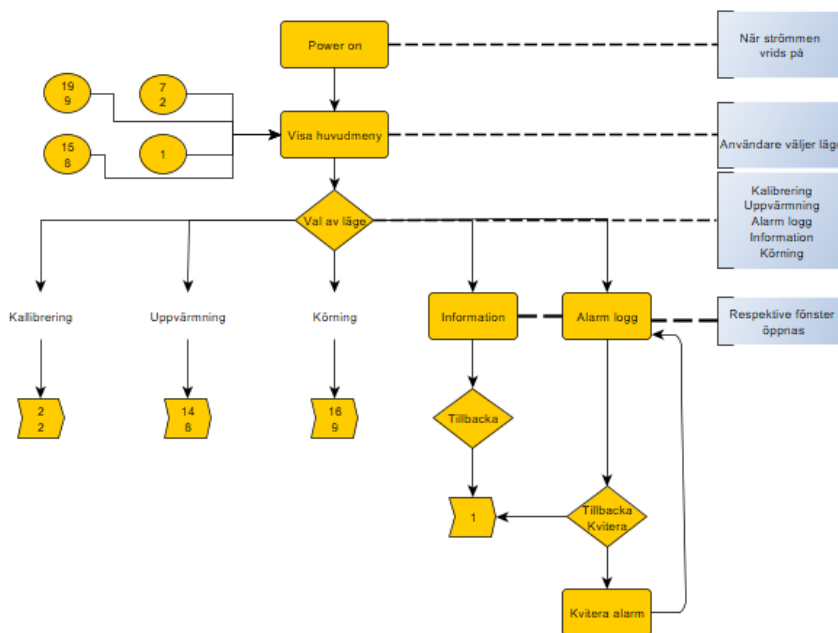
Figur 14. Designprocess



Figur 15. Symbolbeskrivning

3.2 Flödesschema

För att göra programfunktionen tydligare har också ett flödesschema för processfunktionen skapats. I Figur 16 ser vi första sidan av flödesschemat, här beskrivs hur processen ska fungera vid uppstart. Vid start av processen nollas alla körningsvariabler och huvudmenyn öppnas upp. Från huvudmenyn kan användare sedan välja att förflytta sig till kalibrering, uppvärmning, körning, information eller alarmloggen. I informationsläge kan användare granska viktig information angående process och sedan välja tillbaka, detta förflyttar användaren tillbaka till huvudmeny. I alarmloggen kan användare granska vilka alarm som aktiverats och välja att kvittera alarmen och/eller återvända till huvudmeny med tillbakaknapp. Flödesscheman för kalibrering, uppvärmning och körning är aningen mera komplicerade och kan vidare studeras i bilagorna (Bilaga 1).

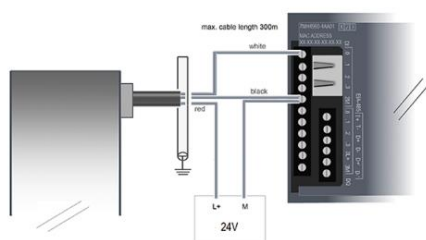


Figur 16. Flödesschema för programstart.

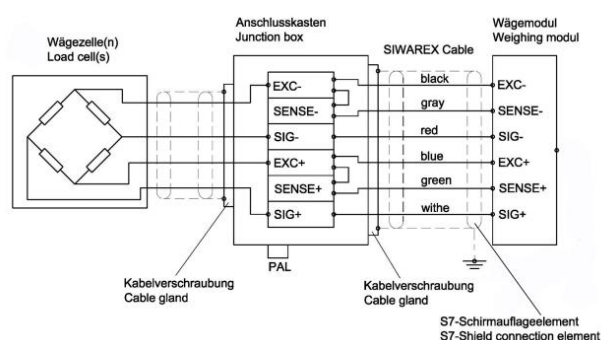
4 Kommunikation

För att styrsystemet över huvud taget ska fungera behöver kommunikation etableras mellan hårdvarukomponenterna. Detta görs på lite olika sätt beroende på komponenten. För givare som inte skickar meddelande utan endast en signal är kommunikationen inte komplicerad. För PLC, HMI och WP241 som skickar mycket mera data mellan varandra krävs ett kommunikationsprotokoll.

För hastighetsgivaren och lastcellen som ska kopplas ihop med bandvågsmodulen finns inte direkt några alternativ utan givarna ska kopplas in på deras designerade ingångar. Hastighetsgivaren kopplas ihop med bandvågsmodulen och en strömkälla enligt Siemens instruktioner i Figur 17 och lastcellen som har fyra ledare, kopplas ihop med bandvågen enligt Figur 18. (Siemens, 2017).

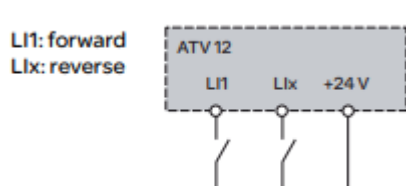


Figur 17. Uppkoppling av hastighetsgivare.
(Siemens, 2017).

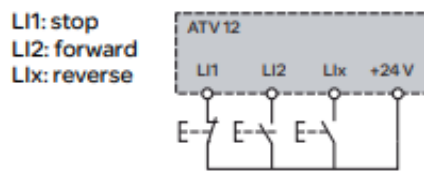


Figur 18. Uppkoppling av lastcell. (Siemens, 2017).

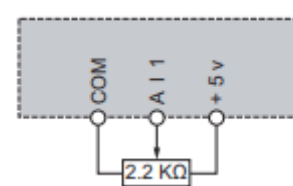
Frekvensomriktaren har också sina designerade ingångar för styrning av motorn, här kan man välja mellan två olika lägen. Det första läget kör motorn fram eller bak på hög signal och slutar köra vid låg (Figur 20). Det andra läget fungerar med pulsstyrning, dvs. en puls startar eller stoppar motorn (Figur 21). I detta arbete används det första alternativet, dvs motorn körs vid hög och slutar köras vid låg. Denna metod valdes eftersom det är lättare att säkerställa att inte motorn lämnar i körningsläge på grund av ett fel eller en bugg i programmet. Styrsignalen tas från en av PLC:ns digitala utgångar. Hastigheten för motorn kan antingen vara konstant eller styras med en analog signal. Uppkopplingen av motorstyrningen illustreras i Figur 19 där en potentiometer kopplas in mellan COM, AI1 och +5V. Med potentiometern kan sedan hastigheten ställas mellan inställt maxvärde och noll. (Schneider Electric, 2016).



Figur 20. Tvåledarestyrning.
(Schneider Electric, 2016).



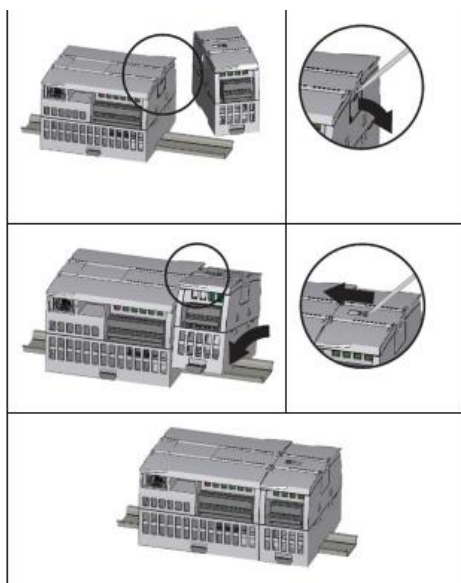
Figur 21. Treledarestyrning.
(Schneider Electric, 2016).



Figur 19. Hastighetsstyrning.
(Schneider Electric, 2016).

För kommunikation mellan bandvågsmodulen och PLC:n används automationssystemets busslinje. Detta betyder att PLC:n och bandvågsmodulen installeras på en din-skens varefter bandvågens bussport öppnas och skjuts fram med en switch på modulens översida. När bussporten är synlig skjuts modulerna ihop. Modulens strömförsörjning kommer inte via bussen utan måste kopplas skilt med egna ledare. Ihopkopplingen av modulerna illustreras i Figur 22

När modulerna är ihopkopplade måste modulen ännu läggas till i TIA-portalen, detta beskrivs senare i dokumentet (kapitel 5). (Siemens, 2018a).

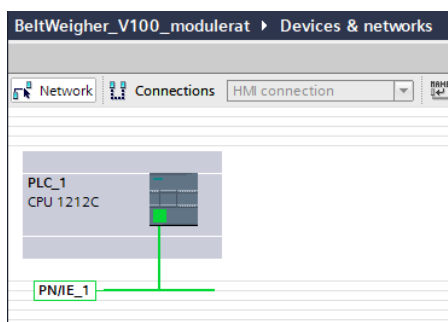


Figur 22. Ihopkoppling av två Siemens moduler. (Siemens, 2018a).

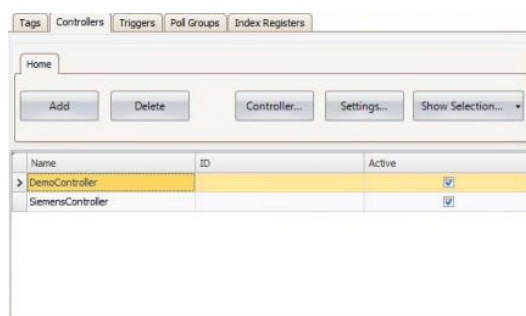
Mellan PLC och HMI skapas kommunikation med PROFINET protokollet. För detta används en standard internetkabel (cat4) med en RJ45 kontakt. HMI:n och PLC:n stöder också kommunikation över RS485 med modbus-protokollet. Modbus uteslöts på grund av att användning av PROFINET är mycket lättare, överföringen av data går snabbare och installationen av en cat4 kabel är smidigare. För att etablera kommunikationen måste en del inställningar fortfarande göras i både TIA-portalen och IX-developer.

För TIA-portalen måste ett PROFINET subnät skapas detta görs i nätverksöversikten (Figur 24). För subnätet behöver inget speciellt fyllas i utöver att PUT/GET checkboxen måste kryssas i. Detta tillåter HMI:n att läsa från och skriva in data i PLC:n.

I IX-developer måste en ny kontroll (PLC) läggas till och några inställningar definieras (Figur 23). Inställningar som ska ställas in angår kommunikationen med PLC:n, här ska bland annat kommunikationstypen, type, rack och slot anges. Som kommunikationstyp anges Iso over TCP/IP, som type anges PG (PUT/GET) som rack anges 0 och som slot anges 1.



Figur 24. PROFINET subnät i PLC.

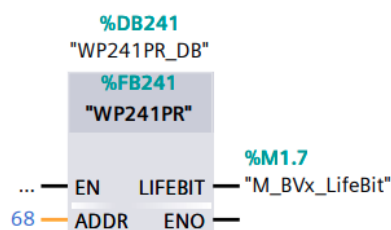


Figur 23. HMI kontrollerinställningar.

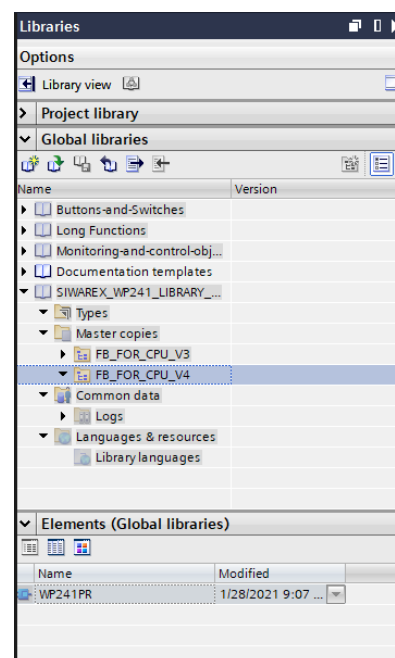
5 Ibruktagnings av vågmodul

Ibruktagnings av SIWAREX vågmodulen görs i huvudsak via SIWATOOL, modulen kan tas ibruk via HMI:ns kalibreringsmeny, men användning av SIWATOOL gör processen märkbart snabbare. För ibruktagningen av WP241 med SIWATOOL har Siemens skapat en detaljerad manual, (Siemens, 2015b). För detta arbete har Siemens dokumentet översatts och förenklats till en svenskspråkig ibruktagningsmanual. Denna manual kan vidare studeras bland bilagorna (Bilaga 2). Manualen innehåller beskrivning på vilka variabler som ska ställas in samt hur inställningen går till. Efter variabel inställningen följer instruktionerna på hur kalibreringen går till. I och med att processen kan vara byggd på lite olika sätt finns olika former av kalibrering. Kalibreringen är uppdelad i hastighetskalibreringar, nollpunktskalibrering, viktkalibreringar och materialtest.

Utöver kalibrering och ibruktagning av den fysiska vågmodulen måste WP241 funktionsblocket importeras till TIA-portalen (Figur 26). Detta görs genom att ladda ner ett bibliotek från Siemens hemsida. När biblioteket importerats till TIA-portalen kan blocket hämtas från global libraries/SIWAREX_WP241_LIBRARY. Funktionsblocket måste väljas enligt vilken version PLC:ns CPU har. När funktionsblocket dragits in i programmet måste ännu startadress definieras, enligt Siemens instruktioner valdes 68 som startadress (Figur 25).



Figur 25. WP241 funktionsblock.

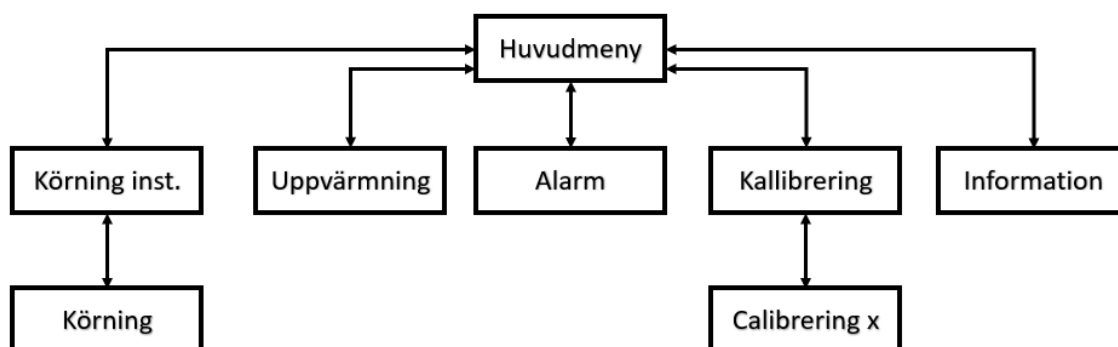


Figur 26. WP241 bibliotek.

6 Lösningförslag

Lösningförslaget bygger på att dela in både HMI- och PLC-programmet i fyra huvudsakliga funktioner; sekvenskörning, uppvärmning, alarmering och kalibrering. Dessa funktioner ska aktiveras från HMI och få godkännande från både HMI:n och PLC:n. Programmets modularitet kommer egentligen in i bilden först i PLC programmet där funktionerna delas upp i bandvågsspecifika funktioner och process-specifika funktioner. Tanken med uppdelningen är att med hjälp av vettiga variabelbenämningar och genomtänkta funktioner, ska flera bandvågsmoduler kunna användas genom att på ett lätt sätt bara duplicera de bandvågsspecifika funktionerna och variablerna.

För HMI ser program- eller fönsteruppläget ut enligt Figur 27. Idén bygger på att ha en huvudmeny varifrån användaren kan manövrera till de olika funktionerna. Bytet från huvudmeny till funktionsspecifika displayen kan därmed användas som godkännande för funktionens körning från HMI sidan. Sekvenskörningen, som är den första funktionen, består av två fönster. I det första fönstret kan man ställa in parametrar och i den andra övervaka processen under körning. Uppvärmnings- och alarmfunktionerna består båda av ett fönster. Till alarmeringsfönstret kommer man från programmets alla delar, ifall ett alarm aktiveras. Till exempel under aktiv uppvärmning eller körning avbryts funktionen vid alarm och HMI fönstret byts. Kalibreringen består av ett stort antal fönster i och med modulens breda kalibreringsmöjligheter. Från kalibreringsfönstren har man bland annat möjlighet att ändra på bandvågens processparametrar, återställa bandvågen till fabriksinställningar eller helt enkelt köra en av kalibreringarna. Informationsfönstret är till för att visa bandvågens namn, total körd vikt och tid både för aktuell dag och sedan ibruktagnig.



Figur 27. Displaystruktur.

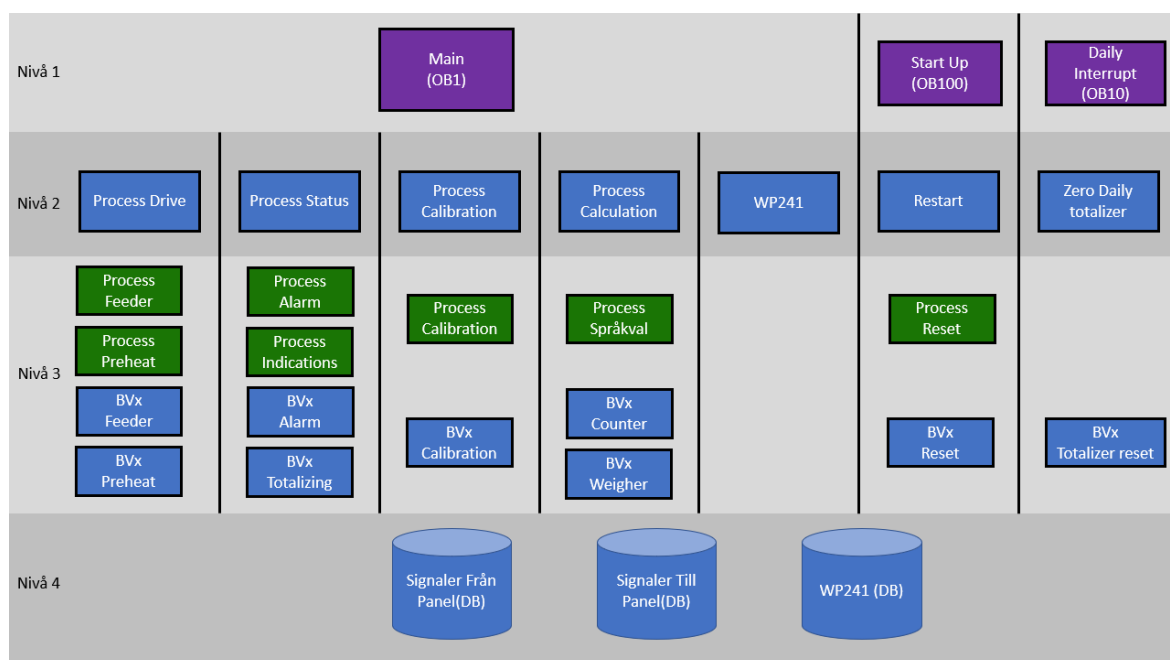
För PLC:n är dess struktur mera komplicerad, här är meningen att dela upp programmet i mindre delar för att lättare kunna modulera och felsöka. Idén med de olika funktionerna är att de så långt som möjligt ska fungera separat från varandra, vid behov ska t.ex. en funktion kunna tas bort. PLC programstrukturen kan delas in i fyra nivåer med tre huvudsakliga grenar (Figur 28).

Den första nivån av består av tre objektblock, main som körs cykliskt, startup som körs vid start av modulen och daily interrupt som körs en gång per dag, klockan 23.00. Main-blocket innehåller programmets huvudsakliga funktioner som behövs för processens användning. Startup-blocket innehåller funktioner som behövs vid uppstart av processen, t.ex. nollning av parametrar. Daily interrupt objektet innehåller endast en funktion som nollar totalräknare.

Den andra nivån består heller inte av någon egentlig programkod utan används för kategorisering av funktioner. Under main-funktionen delas funktioner in i körningsfunktioner, lägesfunktioner, kalibreringsfunktioner och kalkyleringsfunktioner. Utöver detta ligger även bandvågsspecifika funktionsblocket här.

I den tredje nivån ligger själva programfunktionerna, där själva programkoden finns. I denna nivå finns funktioner som körning och uppvärmning. Funktionerna i den tredje nivån är uppdelade i bandvågsspecifika och processspecifika funktioner, detta gör identifieringen av fel lättare och möjliggör kopiering om flera bandvågsmoduler ska användas.

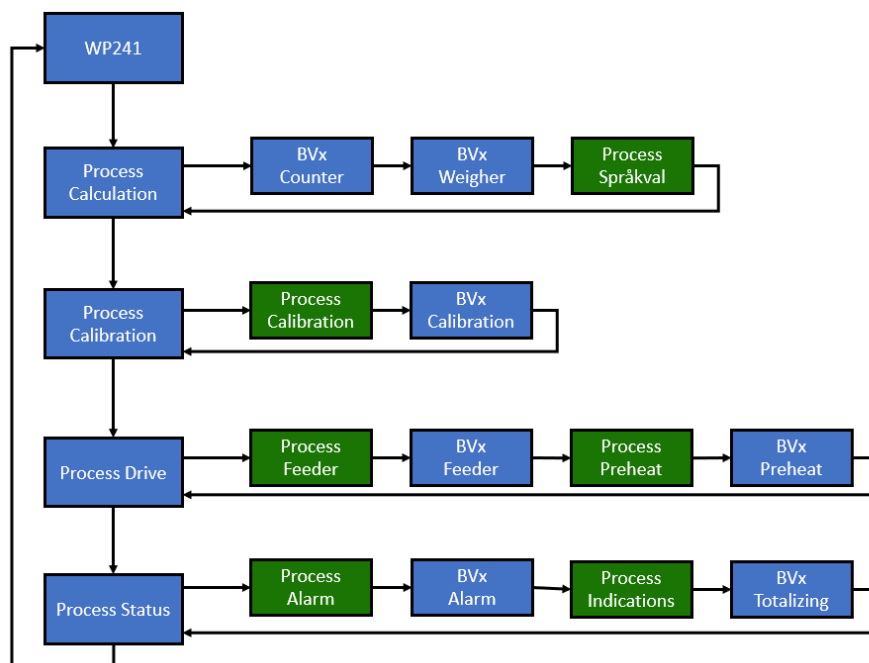
I den fjärde nivån ligger alla globala datablock. Med de globala datablocken avses datablock som hela systemet använder sig av, hit hör signaler till panel, från panel och WP241 datablocket. I programmet finns ytterligare datablock som skapas i samband med funktionsblock. Datablock som skapas för funktionsblocken används inte i ett globalt syfte utan används enbart för datablockets designade funktionsblock.



Figur 28. PLC program uppbyggnad.

I main-blocket som sköter körningen av det huvudsakliga programmet, körs funktionerna i en viss ordningsföljd (Figur 29). I detta program körs först WP241 funktionen, denna funktion tar hand om kommunikationen mellan PLC och WP241-modulen. Efter WP241-funktionen körs processkalkyleringarna. Detta eftersom körningsräknare och vägningskalkyler påverkar processens utgångar är det vettigt att sätta dessa först, så att de

uträknade värden kan användas i samma körningscykel. Som följande kommer kalibreringen, som placeras före körningsfunktioner, eftersom man i vissa kalibreringar behöver köra processen. Efter kalibreringsfunktioner kommer själva körfunktionerna, dvs uppvärmning och processkörningen. Efter körningen kommer processindikationerna, dessa funktioner kommer till sist eftersom dessa är beroende av alla tidigare funktioner, t.ex. om processen sätts i körningsläge aktiveras körningsindikationen under samma cykel.



Figur 29. Main programmets körordning.

7 Detaljerad genomgång av system

Detta kapitel kommer handla om en detaljerad programgenomgång. Genomgången kommer fokusera på nivå två och tre som beskrevs i tidigare kapitel. Först kommer systemets signaler presenteras. Efter styrsignalerna går processens räknefunktioner igenom. Räkneoperationerna presenteras först på grund av dess relevans för resten av programmets sekvenser. Följt av räkneoperationerna presenteras programmets olika skeden steg för steg. För varje del beskrivs först funktionsprincipen av användargränssnittet följt av en helhets presentation av funktionen och till sist en detaljbeskrivning av funktionen.

7.1 Signalpresentation

Signaler som används i systemet kan delas in fem huvudsakliga kategorier baserat på deras användning. Kategorierna är HMI-signaler, PLC interna tags, funktionspecifika tags, IO signaler och bandvågsspecifika tags.

HMI signalerna delas upp i två datablock, signalerna till och från HMI panel. Dessa signaler är informativa variabler som berättar för användaren vad som händer i processen, exempel på dessa är alarmen som loggas i HMI:n och om körningen av något bälte är aktivt. (Bilaga 3.1 och 3.2)

PLC interna tags byggs upp av minnesvariabler för programmets olika skeden. Exempel på dessa tags är bland annat körningsminne, som används för att aktivera motorstyrningsutgångar. (Bilaga 3.3)

Funktions specifika tags är variabler som sparas i funktionsblockens egna datablock. Dessa variabler kan sägas fungera som temporära variabler eftersom viktig information sparas i antingen HMI datablocken eller PLC tag listan.

IO signaler är de signaler som kommer in och ut från PLC:n. In- och utsignalerna presenteras i Tabell 1. Ingångs signalerna består av process-specifika knapptryckningar, som start och stopp. Skapande av dessa signaler genom fysisk knapptryckning i stället för genom HMI-displayn är på grund av att dessa signaler är viktiga för processens drift, därmed är det tydligare att ha en fysisk knapp.

Utgångarna handlar om signalerna för process-styrning och viktiga indikeringar. Process-styrningssignalerna är start av transportbältets motorer och indikationssignalerna. Indikationerna består av felindikering, körningsindikering och startklar-indikering.

Tabell 1. PLC in- och utsignaler.

Variabel namn	Variabel typ	Port ID
Ingångar		
In_Start	Bool	%I0.0
In_Stopp	Bool	%I0.1
In_NodStopp	Bool	%I0.2
In_Kalibrering	Bool	%I0.3
In_DriftMan	Bool	%I0.4
In_DriftAuto	Bool	%I0.5
Utgångar		
Ut_Grov_Mat.	Bool	%Q0.0
Ut_Fin_Mat.	Bool	%Q0.1
Ut_Startklar	Bool	%Q0.2
Ut_Köring	Bool	%Q0.3
Ut_Fel	Bool	%Q0.4
Ut_BVx_Mat.	Bool	%Q0.5

Bandvågsspecifika tags består av variabler i WP241-datablocket. I detta datablock finns väldigt mycket data som delas mellan PLC:n och WP241-bandvågsmodule. I datablocket finns variabler som konstant uppdateras, detta är relevant data som nuvarande flöde, huvudtotalräknare, processindikationer och så vidare. Utöver data som uppdateras konstant finns upp till 48 dataregister (DR) som måste uppdateras manuellt. De mest

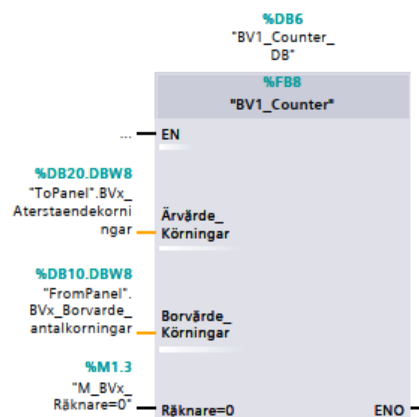
relevanta DR som uppdateras är 30, 33 och 31. DR30 består bland annat av hastigheten och nuvarande vikten på bandvågen, DR33 består av totalräknare S1-S6 och DR31 består av bland annat av pulsinformation för hastighets avkänning. (Bilaga 3.4).

HMI:n har också en taglist, som bygger på olika variabler från PLC-programmet. Tags som används i HMI:n kommer bland annat från WP241-datablocket samt till och från HMI datablocken. Utöver dessa skapas data också internt av HMI:n. Display ID data är något som används endast internt av HMI, dvs denna data representeras inte i något fönster. Denna data är uppdelad i två delar, nuvarande ID och bytes ID, vilket fönster vi nu befinner oss i och vilket fönster vi vill byta till. (Bilaga 3.5).

7.2 Räkneoperationer

Räkneoperationerna skiljer från resten av funktionerna i och med att dessa inte är direkt bundna till något användargränssnitt. Räkneoperationerna består av två bandvågsspecifika funktioner, räknefunktionen och vägningsfunktionen. Räknefunktionen uppgift är att ta hand om körningsräknare, dvs. hur många körningar som utförts. Räkneoperationen är direkt bunden till hur många körningar användaren matat in i parameterinställningskedet, parameterinställningen presenteras under körnings kapitlet. Vägningsfunktion tar hand diverse uppgifter som behövs för att utföra en noggrann vägning. Båda dessa funktioner har skapats som funktionsblock, för att så lätt som möjligt kunna kopiera funktionen för användning till flera bandvågs moduler.

Räknefunktionen är den enklare av de två (Figur 30). Räkneblocket består av tre in/out signaler och en temporär signal. Signaler är deklarerade som in- och ut signaler, eftersom de i funktionen används både som jämförelse och ställbar variabel, dvs. på båda sidorna av diverse block. IO signaler är körningsräknarens är- och börvärde samt en räknare=0 variabel som används för att berätta för processen att alla körningar är gjorda och räknare nu har värdet noll. Inne i funktionsblocket finns fem olika nätverk som har diverse uppgifter.



Figur 30. BVx Counter funktion.

Det första nätverket tar hand om nollning av det temporära räknevärde om körningen avbryts (Bilaga 4.1.1). Denna funktion håller koll på om antingen stopp, nödstopp eller avbrytning från HMI aktiveras under körning. Om något av värden går högt överförs noll till temporära räknevariabeln. Nollvärdet överförs även till variabeln om temporära räknevärde går under noll, eftersom ett negativt antal körningar inte ska vara möjligt. Ytterligare sköter funktionen även om att byta körningsfönstret till parameterfönstret om samma krav uppfylls.

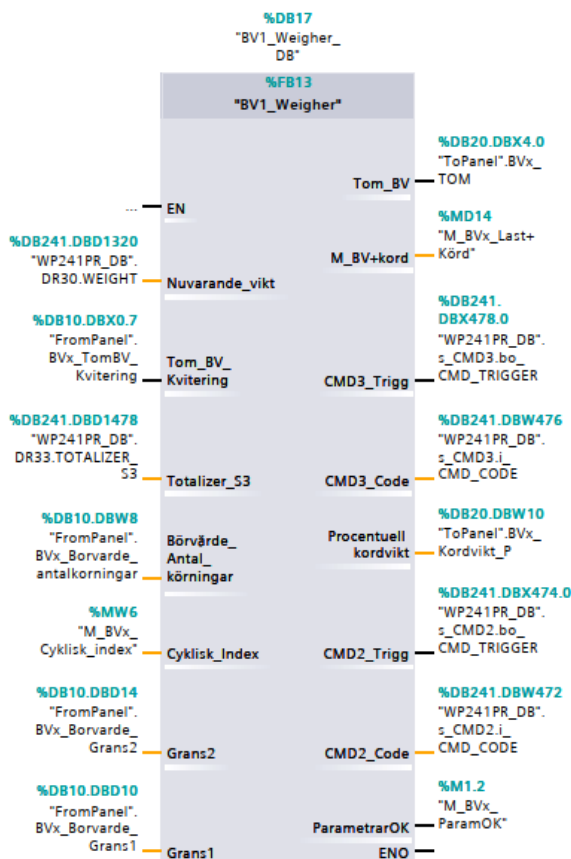
I det andra nätverket överförs börvärdet till temporära räknevariabeln (Bilaga 4.1.2). Börvärdet överförs endast om räknaren=0 variabeln är hög, parametrar är inställda för körning, HMI:n indikerar ok för körning och startknappen trycks.

Det tredje nätverket förhindrar inmatning av ett negativt börvärde (Bilaga 4.1.3). Detta görs genom att använda en jämförelseoperator < som jämför börvärdet med noll, om börvärdet blir negativt överförs noll till variabeln.

Det fjärde nätverket är själva räknaren (bilaga 4.1.4). För denna funktion används ett count down block (CTD), blocket fungerar med tre ingångar och två utgångar. Den första ingången (CD) används för att indikera när räknare ska aktiveras och hit skickas en signal när vikten är uppnådd, körningOK är aktivt och HMI:n indikerar att körning är ok, dvs varje gång vikten uppnås minskas variabelvärdet som är kopplat till den första utgången (CV) med ett. Till utgången CV kopplas ärvärdesminnet som sedan skickas till HMI:n. I HMI:n representeras värdet sedan som återstående körningar. När en signal skickas till den andra ingången (LD) kommer värdet från den tredje ingången (PV) överförs till CV. LD ingången aktiveras när räknaren är noll, parametrar är ok, start har tryckts och HMI:n indikerar att körning är ok eller om körning avbryts, dvs stopp, nödstopp eller avbryt från HMI aktiveras. Till PV ingången kopplas det temporära räknevärdet. Den andra utgången (Q) kommer gå hög när räknaren når noll och aktiverar minnet, räknare=0, och återställer avbrytning från HMI minnet, avbrytningsminnet behöver nödvändigtvis inte vara aktivt men om körningen avbryts via HMI:n måste minnet återställas här.

Det sista nätverket sköter om byte av fönster från körningsfönster till parameterfönster vid slutförd körning (Bilaga 4.1.5). När antingen manuell- eller autokörning klar-minnet går högt, vikten är uppnådd, körningOK är aktivt och HMI:n indikerar att körning är ok kommer fönstret bytas. Manuell- och autokörning klar-minnen aktiveras när respektive läge är inställt och räknare är noll, dessa minnen aktiveras i körningsfunktionen som presenteras senare.

Vägningsfunktionsblocket består av mycket mera in-och utgångar i och med att mycket mera bandvågsspecifika variabler behövs för körningen av flera bandvågsmoduler (Figur 31). De flesta av signalerna är relativt självbeskrivande, men vissa behöver en liten introduktion. CMD_Code och CMD_Trigg har att göra med att skicka kommandon till bandvågsmodulen. Till CMD_Code anges kommandokoden som ska skickas när CMD_Trigg växlas från låg till hög.



Figur 31. BVx Weigher funktion

Vägningsblocket består av åtta nätverk, var av det först tar hand om indikering av tom bandvåg (Bilaga 4.2.1). Bandvågen är aldrig fullständigt tom utan växlar mellan några gram positivt och negativt, detta på grund av att systemet inte är kalibrerat för stilla stående vägning. För att fortfarande kunna granska om vågen är tom användes ett IN_RANGE block, där två gränsvärden anges. Nuvarande vikten ska ligga mellan dessa gränser. För detta arbete som inte optimerats till en verklig process användes $\pm 0.02\text{kg}$. När nuvarande vikten ligger mellan gränsvärden aktiveras Tom_BV minnet som berättar för resten av processen att bandvågen är tom. Ytterligare kan minnet aktiveras av den manuella kvitteringen, t.ex. i samband med uppvärmningen.

Det andra nätverket består endast av ett additionsblock (Bilaga 4.2.2). Detta additionsblock räknar ihop bandvågens nuvarande vikt med den körda vikten. Värdet används sedan i körningsfunktionen för att stoppa finmatningen lite före bandvågen, för att förhindra överlopsmaterial på vågen.

I det tredje nätverket skapas en puls för cyklisk inläsning av dataregister. Pulsen skapas genom användning av ett on-delay, off-delay och en pulsvariabel. Pulssignalen är direkt kopplad till CMD3_Trigg variabeln som går hög varje gång en puls aktiveras. Ytterligare är en räknare kopplad till pulsen. CTD räknaren minskar på ett värde från fyra till noll varje gång en puls går hög, när värdet når noll börjar räknaren igen räkna ner från fyra. Detta räknevärde används i nästa nätverk där vi läser i dataregistren (Bilaga 4.2.4). I detta nätverk läses dataregister 30, 31 och 33 in cyklisk med hjälp av räknaren. Dataregister läses in genom att skriva in CMD koden (2000+dataregisters som ska läsas in) och aktivera trigg variabeln. Genom att använda räknevärden som index kan olika dataregister läsas in

cykliskt, varje gång räknaren har värdet 1 kommer dataregister 30 läsas in, när värdet är 2 kommer 33 läsas in och med värdet 3 kommer 31 läsas in. Det fjärde indexvärdet används för kalibreringen.

I det femte nätverket konverteras genomkörda vikten från kilogram till en procentuell andel (Bilaga 4.2.5). Detta görs genom att först dividera genomkörda vikten med börvärdet, genom detta får vi ett normaliserat tal. Efter detta multipliceras talet med 100 och konverteras från real form till integer, detta på grund av att vi inte behöver procentuella talet i decimaltalsnoggrannhet.

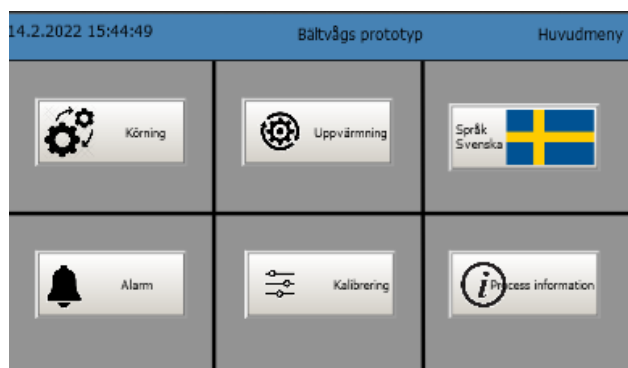
Det sjätte nätverket tar hand om nollning av totalräknare som håller koll på hur mycket vikt som körts igenom (Bilaga 4.2.6). Totalräknaren nollas genom att skicka kommandokoden 671 till bandvågs modulen. Kommandokoden skickas med CMD2 istället för CMD3, detta på grund av att CMD2 kommer prioriteras, dvs CMD3 kommandon som används för registerinläsning kommer pausas medan CMD2 körs. Totalräknaren nollas om vikten är uppnådd, körning_OK-minnet är aktivt och HMI:n indikerar att körning är okej och pulssignalen går hög eller om körningen avbryts genom stopp, nödstopp eller via HMI:n.

Det sjunde nätverket används för att säkerställa att gränsparametrarna inte anges som negativa tal (Bilaga 4.2.7). Om användare försöker ange ett gränsvärde i parameterfönstret kommer detta automatisk att få värdet noll.

Det åttonde nätverket sköter om indikering för att parametrarna är korrekt ifyllda (Bilaga 4.2.8). I denna funktion granskas att parametrarna för körning är över noll, dvs grov- och finmatningsgränserna och antal körningar. Om parametrarna är över noll och HMI:n indikerar att körning är ok aktiveras parametrar_OK-minnet.

7.3 Start av systemet

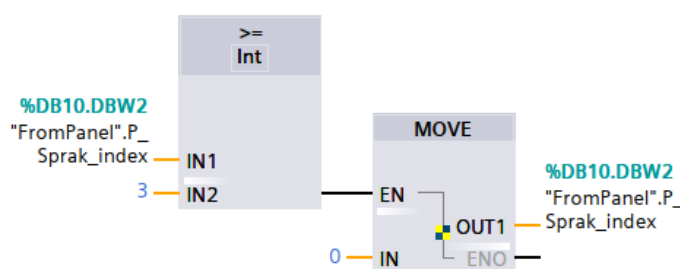
När processen startas börjar användargränssnittet alltid i huvudmenyn (Figur 32). Från huvudmenyn kan användare välja mellan att gå till körningsläge, uppvärmningsläge, alarmloggen, kalibreringen, processinformationen eller bara byta språk. I fönstrets övre del, i det blåa fältet, syns tid och datum, bandvågens namn och vilken display som användaren befinner sig i. Detta informationsfält är den enda gemensamma egenskapen för alla fönster.



Figur 32. Huvuddisplay.

Språkvalet är den enda funktionen som är enbart bunden till huvudmenyn. Funktionen uppgift är att stöda språkvalet. Språkvalet fungerar i stort sätt enbart från HMI:n. Språkvalet skapas med IX-developers multiple language meny. I menyn lägger man först till vilka språk som ska användas och tilldelar dem ett indexvärde. Språken kan därefter översättas i en tabell och här kan en automatisk översättare användas, men bästa resultat erhålls genom att manuellt översätta texten. Språkvalet kan sedan bindas till en indexvariabel som användaren kan påverka, i detta fall ökas indexvärdet varje gång användaren trycker på språk-knappen i huvudmenyn. HMI:n har däremot ingen funktion som kan nolla indexvariabeln utan här kommer PLC funktionen in i bilden.

PLC funktionen består av endast ett nätverk som övervakar att indexvärdet inte överstiger 3 (Figur 33). Funktionen fungerar genom att ha en \leq komparator som granskar värden, om påståendet är sant flyttas värdet 1 till indexvariabeln med ett MOVE block. För denna process är svenska standardspråket med indexvärdet 1, engelska har värdet 2 och finska har värdet 3.



Figur 33. Språkvalsfunktion.

7.4 Körning

När användare navigerar från huvudmenyn till körning hamnar hen först till parameterinställningsfönstret (Figur 34). I detta fönster kan användaren ange grov- och finmatningsgränser, antal körningar samt köra bandvågsbältet och huvudmatarbältet manuellt.

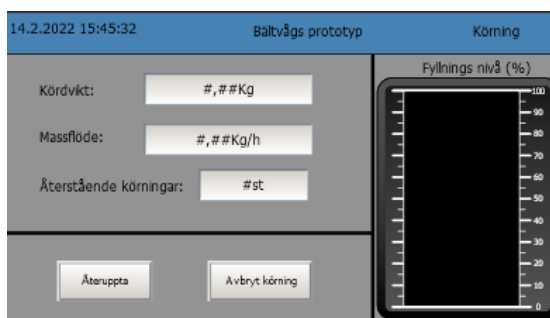
Från detta fönster kan användaren antingen gå tillbaka till huvudmeny eller fortsätta till körningen. För att fortsätta till körningen måste användaren, ha valt manuell- eller autoläge, fyllt i ett värde större än noll på alla parametrar och trycka på den fysiska startknappen. Bandvågen måste också vara tom för att starta körningen, detta kan konstateras från fönstrets nedre hörn där bandvågens nuvarande vikt visas. När bandvågen är startklar indikeras detta av en fysisk märklampa.

14.2.2022 15:45:36		Bältvågs prototyp		ParameterInst.	
Grovmatnings Gräns:	<input type="text" value="#,#Kg"/>	Bandvåg manuell körning			
Finmatnings Gräns:	<input type="text" value="#,#Kg"/>	<input type="button" value="Fram"/>			
Antalkörningar:	<input type="text" value="#st"/>	Matare 1 manuell körning			
		<input type="button" value="Fram"/>			
Nuvarande vikt:	<input type="text" value="#,#Kg"/>	<input type="button" value="Tillbaka"/>			

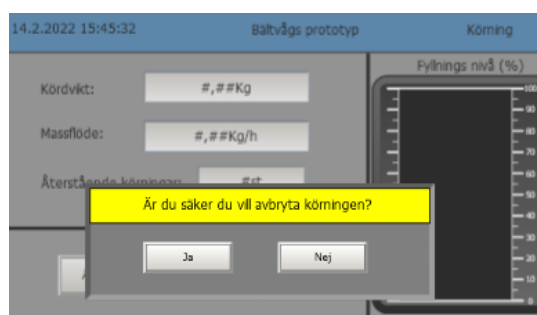
Figur 34. Körningens parameterinställningar.

När användaren tryckt på start kommer fönstret bytas till körningsfönstret där processen kan övervakas (Figur 35). I detta fönster visas den körda vikten, massflödet per timme och återstående körningar. Den körda vikten indikeras också procentuellt i form av en grafisk låda i fönstrets högra kant, lådan fylls med en grön färg och när hela lådan är fylld betyder det att 100 procent av önskade vikten körts. För att stoppa processen kan användare antingen trycka på fysiska stoppknappen eller avbryta körningen via HMI:ns avbryt knapp. Om avbrytningen sker via HMI kommer ett pop-up fönster dyka upp som frågar om processen verkligen ska avbrytas, detta har skapats för att förhindra felklickning (Figur 36). Vid avbruten körning byts fönstret till parameterfönstret. Om användaren vill stanna processen, men inte avbryta körningen, kan HMI pausknappen användas. Om paus har tryckts kan processen återupptas genom att trycka på samma knapp, vars text har bytt till återuppta.

När körningen är klar och antal körningar inte ännu är noll, stannar HMI:n i körningsfönstret och inväntar tryckning av startknapp som startar nästa körning. När antalet körningar är noll byts fönstret automatiskt tillbaka till parameterfönstret.



Figur 35. Körningsfönster.



Figur 36. Säkerställning av avbrytning av körning.

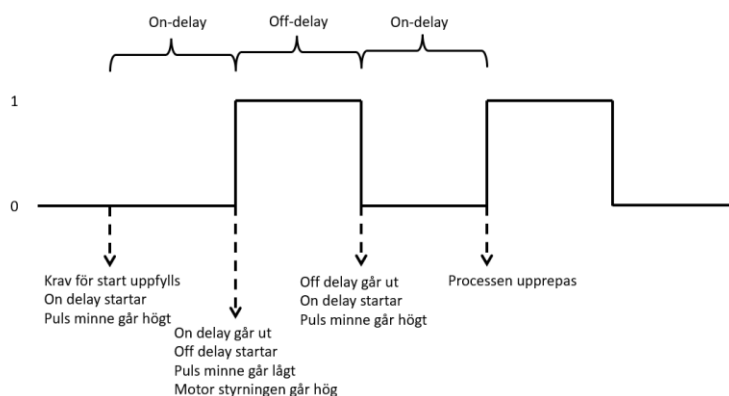
PLC-programmet för körningen är modulärt uppdelat, dvs. en funktion för processen och en funktion för bandvågsprocessen. Bandvågsfunktionen består av olika in- och utgångar som ska kopplas ihop med diverse variabler för kommunikation med resten av programmet. Processkörningsfunktionen består av sex olika nätverk (Figur 38).

Det första nätverket bygger på en SR vipa som aktiverar körning_ok-minne (Bilaga 5.1.1). Detta minne berättar för resten av processen att körningen kan startas. Vippa ställs till högt läge om startknapp påverkas, nödstopp inte är aktiv, HMI:n godkänner körning, parametrar är ok och bandvågen är tom. Tom_bandvåg-kravet kan skippas genom att trycka på start två gånger. Vippa går låg om stoppknappen påverkas, nödstopp aktiveras, körning avbryts från HMI, om alarm aktiveras, körningsräknare når noll eller vikten är uppnådd. I samma nätverk sker också byte från parameterfönstret till körningsfönster och aktivering av manuell- eller automatkörning klar-minnen. Bytet mellan fönstren sker på samma gång som kraven för körning_ok-minnet uppfylls.

Det andra nätverket aktiverar grovmatningens körningsminne (bilaga 5.1.2). Kraven för att grovmatningen ska hållas igång är för det första att inte grovmatningsgränsen är nådd. Jämförelsen här sker mellan gränsvärdet som getts i parameterinställningsfönstret, summan av genomkörda vikten och nuvarande vikten. Det andra kravet är att körning_ok-minnet är aktivt, finmatningen inte är aktiv och paus av körningen inte aktiverats.

Det tredje nätverket sköter om aktivering av finmatningsminnet (bilaga 5.1.3). För denna funktion är kraven nästa samma som för grovmatningen. Här sker jämförelsen mellan summerade värdet och finmatningsgränsvärdet, körning_ok-minnet måste vara aktiv, grovmatningen får inte vara aktiv och paus av körningen får inte vara aktiv.

Det fjärde nätverket sköter om pulsmatningsminnet (bilaga 5.1.4). Denna funktion är ett alternativt val till finmatningen. Startkraven för pulsmatningen är helt samma som för finmatningen, med den enda skillnaden att även en temporär pulsvariabel måste vara låg. Pulseringen sker genom att istället för att aktivera PLC-utgången använda en ON-delay följt av en OFF-delay. När kraven för aktivering är sanna går ON-delayn igång, som släpper igenom signalen efter en på förhand definierad tid. När ON-delayn börjat släppa igenom en signal, ställs den temporära pulsvariabel hög. Detta betyder att ON-delayn inte längre släpper igenom en signal. På samma gång har en OFF-delay gått igång, som håller signalen hög en för definierad tid. Under tiden som OFF-delayn räknar är temporära pulsvariabeln låg och motorstyrningsminnet högt. När OFF-delay räknat klart slutar temporära pulseringsminnet gå lågt, motorn stannar och processen börjar om.



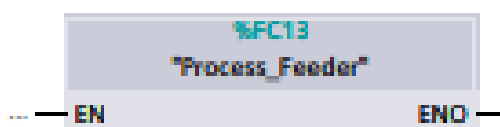
Figur 37. Pulsering av signal.

Det femte nätverket sköter om huvudmatarbältets manuella körningsminne (Bilaga 5.1.5). Minnet hålls aktiv så länge som manuella körningsvariabeln från HMI är högt, både PLC och HMI indikerar att processen är i kalibreringsläge eller fönster ID:n indikerar att HMI parameterinställningsfönstret är öppet.

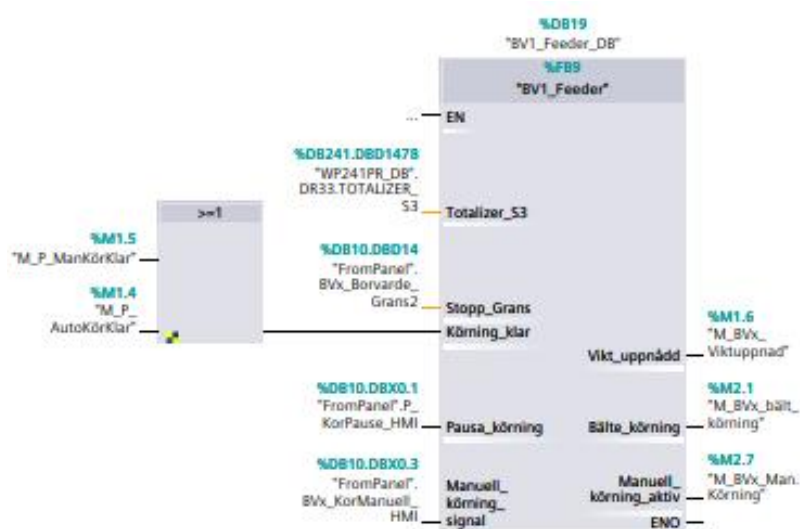
Det sjätte nätverket sköter om styrningen av PLC-utgångarna, som i sin tur startar och stoppar motorerna (Bilaga 5.1.6). PLC-utgångarna hålls aktiva så länge som ett av motorkörningsminne hålls aktiva och nödstoppen inte är intryckt.

Bandvågskörningsfunktionen består av två nätverk (Figur 39). I det första nätverket startas och stoppas bandvågens bälte (Bilaga 4.3.1). Bältet har som körningskrav att körda vikten är mindre än finmatningsgränsen, processkörnings_ok-minnet är högt, körningen inte är klar och att pausande av körning inte är aktivt. Om alla krav är godkända aktiveras bandvågsbältets körningsminne. Ytterligare om viktkravet uppfylls, går ett vikt_uppnåd-minne högt. Detta berättar för resten av processen att önskad vikt är uppnådd.

I det andra nätverket kontrolleras bandvågsbältets manuella styrning (Bilaga 4.3.2). Denna funktion fungerar på samma sätt som huvudmatarens manuella styrning, dvs. kravet är att antingen kalibrering är aktiv eller så ska processen befinna sig i parameterfönstret och manuella körningssignalen ska vara hög.



Figur 38. Process Feeder funktion.



Figur 39. BVx Feeder funktion.

7.5 Uppvärmning

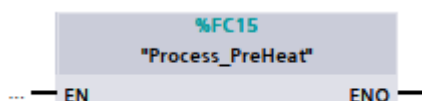
Uppvärmningen sker i två steg. Det första steget är att ställa in tiden och starta uppvärmning. Det andra steget är själva körningen av uppvärmningen. Uppvärmningsprocessen består av endast ett fönster (Figur 40). I fönstrets övre del ställs uppvärmningstiden in, genom en drop-down lista. Den rekommenderade uppvärmningstiden är 30 min, men alternativ för 10 och 20 min har skapats för att användare inte alltid har tid att vänta de rekommenderade 30 minuterna. För att uppvärmningen ska kunna starta, måste bandvågen vara tom. Om uppvärmning ska köras före kalibrering kommer inte bandvågens nuvarande vikt vara noll, därmed kan vikten kvitteras via knapptryckning. För att granska om bandvågen är tom finns en grafisk indikationslampa, som lyser grönt om bandvågen är tom och rött om det finns någon vikt på bandvågen. Under bandvågsviktlampan finns ytterligare en grafisk indikationslampa, som indikerar om uppvärmningen pågår. Uppvärmningen startas genom att trycka på den fysiska startknappen. När uppvärmningen pågår visas tiden i form av en grafisk ruta som fylls med grön färg. I rutan visas tiden i minuter. När den definierade tiden är klar byts fönstret automatiskt till huvudmenyn. Om uppvärmningen ska avbrytas före den planerade uppvärmningstiden, kan användaren trycka på fysiska stoppknappen eller trycka på avbryt

i HMI-fönstret. Om användaren trycker på avbryt via HMI kommer ett pop-up fönster öppnas på samma sätt som för avbrytning av körning. Vid avbruten körning kommer fönstret automatiskt bytas till huvudmenyn.

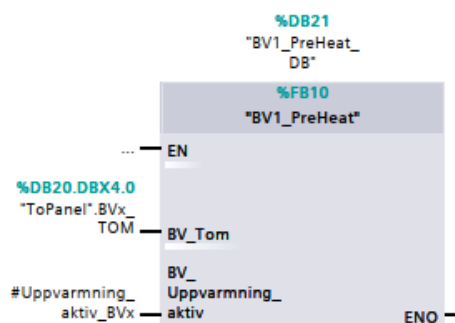
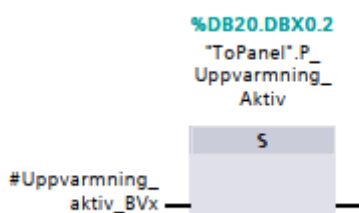


Figur 40. Uppvärmningsdisplay.

PLC programmet finns i körningsfunktionen. Uppvärmningen består på samma sätt som körningen av två funktioner, en bandvägsspecifik och en process-specifik (Figur 41 och Figur 42). För uppvärmningen har också ett nätverk skapats för hopslagning av uppvärmningsindikation (Figur 43). Denna funktion är skapad i förebyggande syfte ifall flera bandvägsmoduler används.



Figur 41. Process Preheat funktion.



Figur 42. BVx Preheat funktion.

Figur 43. I hopslagning av godkänd uppvärmning.

Process-specifika funktionerna består av fyra nätverk. Det första nätverket används för börvärdesbehandling (Bilaga 5.2.1). Börvärdet för uppvärmningstiden förflyttas först till en temporär variabel, denna variabel multipliceras sedan med 60 000 för att konvertera tiden från minuter till millisekunder. Värdet flyttas till sist från temporära variabeln till en PLC tag.

I det andra nätverket finns själva timern och avslutande av uppvärmning (Bilaga 5.2.2). Timern aktiveras när uppvärmning_aktiv-variabeln går hög och börjar räkna ner på det behandlade börvärdet, på samma gång sparas tiden som timern varit aktiv i ärvärdes variabeln. När timern räknat klart eller uppvärmningen avbryts genom stopp, nödstopp,

alarm eller via HMI:n kommer uppvärmningsfönstret bytas till huvudmenyn och uppvärmningsvariabler återställas.

Det tredje nätverket tar hand om aktivering av bältkörningsminnen (Bilaga 5.2.3). Om uppvärmning_aktiv-minnet är aktivt och HMI:n indikerar att processen är i uppvärmningsläge kommer bandvågs- och huvudmatarbältents körningsminne aktiveras.

Det fjärde nätverket behandlar uppvärmningstidens ärvärde (Bilaga 5.2.4). Eftersom tiden från timern indikeras i millisekunder kommer värdet konverteras till minuter för att ge användaren en vettigare representation. Detta görs genom att flytta ärvärdet till en temporär variabel, multiplicera variabeln med 60 000 och flytta värdet till datablocket där HMI:n får sina värden.

Bandvågsspecifika funktionen består av endast en funktion (Bilaga 4.4.1). Denna funktion indikerar om bandvågen ifråga är klar för uppvärmning. Om körning_aktiv-variabeln är låg, bandvågen är tom, HMI:n indikerar att bandvågen är i uppvärmningsläge och startknappen trycks, kommer uppvärmning_aktiv-minnet att ställas till högt läge. Detta minne används sedan i process-specifika funktionen för att starta hela uppvärmningen.

7.6 Alarmering

Alarmeringen byggs huvudsakligen upp i HMI:n, genom skapande av en alarmserver. Alarmeringen består av ett fönster som användaren kan navigera till från huvudmenyn. Systemet kan däremot byta till alarmeringen från programmets alla fönster. Alarmeringsfönstret består av en alarmlogg och en exit-knapp. Alarmloggen i sig själv innehåller fyra knappar, Ack selected, Ack All, Clear och Info. Ack funktionerna kvitterar antingen alla alarmen eller enbart de som är valda. Clear knappen tömmer alla alarm från listan och info-knappen är skapad i förebyggande syfte. Exit knappen tar användaren tillbaka till huvudmenyn. Alla alarm listas upp i Tabell 2. I tabellen listas alarmnamnet och alarmets ursprung, dvs om alarmet kommer från PLC:n eller WP241 modulen.

State	Active Time	Text
Acknowledge	14.2.2022 12:46:50	Kalibrering Akt
Inactive	14.2.2022 15:04:04	Nödstopp aktiv
Inactive	14.2.2022 15:02:05	Nödstopp aktiv
Normal	14.2.2022 15:10:54	BV1 Bandvåg i
Normal	14.2.2022 15:07:36	Kalibrering Akt
Normal	14.2.2022 15:07:29	Kalibrering Akt
Normal	14.2.2022 15:04:47	Kalibrering Akt
Normal	14.2.2022 15:04:36	Kalibrering Akt
Normal	14.2.2022 15:04:30	Nödstopp aktiv
Normal	14.2.2022 14:48:42	Kalibrering Akt
Normal	14.2.2022 14:38:15	Kalibrering Akt
Normal	14.2.2022 14:28:04	Kalibrering Akt
Normal	14.2.2022 14:24:54	BV1 CMD kod
Normal	14.2.2022 14:22:05	BV1 Bandvågs

Active: 1 Inactive: 2 Ack: 1 Normal: 28 Disabled: 0 [32 / 32]

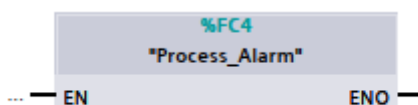
Figur 44. Alarmlogg.

Tabell 2 Alarmlista.

Man-/Autoläge inte invalt.	PLC
Parametrar är inte inställda.	PLC
BVx Bandvågen är inte tom.	PLC
Nödstopp aktiverat.	PLC
BVx CMD-kod kunde inte köras.	WP241
BVx Bandvågshastighetsfel.	PLC
Kalibreringen aktiv.	PLC
BVx ApplID Error.	WP241
BVx ApplIDD Error.	WP241
BVx SFB-fel.	WP241
BVx RdPer Error.	WP241
BVx Life bit-fel.	WP241
BVx Startup-fel.	WP241
BVx Dataoperationsfel.	WP241
BVx Totalräknarfel.	WP241
BVx Modulfel.	WP241
BVx Digitalingångsfel.	WP241

Alarmeringen i PLC-programmet är uppdelat i process-specifika alarm och bandvågsalarm. Process-specifika alarmen omfattar t.ex. att användaren glömt ge in någon variabel före start medan bandvågsspecifika alarmen innehåller alarm från bandvågsmodulen.

Process-specifika alarmfunktionen innehåller sju nätverk (Figur 45). Detta är en standardfunktion som efter sin körning glömmer använda värden. Alarmen i denna funktion är till stor del samlade i en boolean array där varje plats har ett specifikt alarm.



Figur 45. Processalarmfunktion.

Det första nätverket aktiverar manuell-/autoalarm (Bilaga 5.3.1). Om användare försöker starta körning av process utan att ha valt antingen automat- eller manuell-läget kommer alarm-array plats noll gå hög. Alarmet kommer också aktiveras om användare försöker byta läge mitt i körningen.

Nätverk två aktiverar alarm om användare försöker starta en körning utan att ha ställt in alla parametrar (Bilaga 5.3.2). Detta alarm tar upp plats ett i alarm-arrayn och aktiveras om parametrarna inte är inställda, HMI indikerar att processen är i körningsläge och startknappen påverkas.

Nätverk tre alarmerar för att bandvågen inte är tom (Bilaga 5.3.3). Eftersom bandvågen i vissa fall förväntas vara tom har också ett alarm skapats som indikerar för användare att åtgärder måste tas för att starta funktionen. Detta alarm aktiveras om HMI indikerar på körning eller uppvärmning, bandvåg inte är tom, HMI inte befinner sig i körningsläge eller parameterfönster och startknappen påverkas. Detta alarm finns i array-position två.

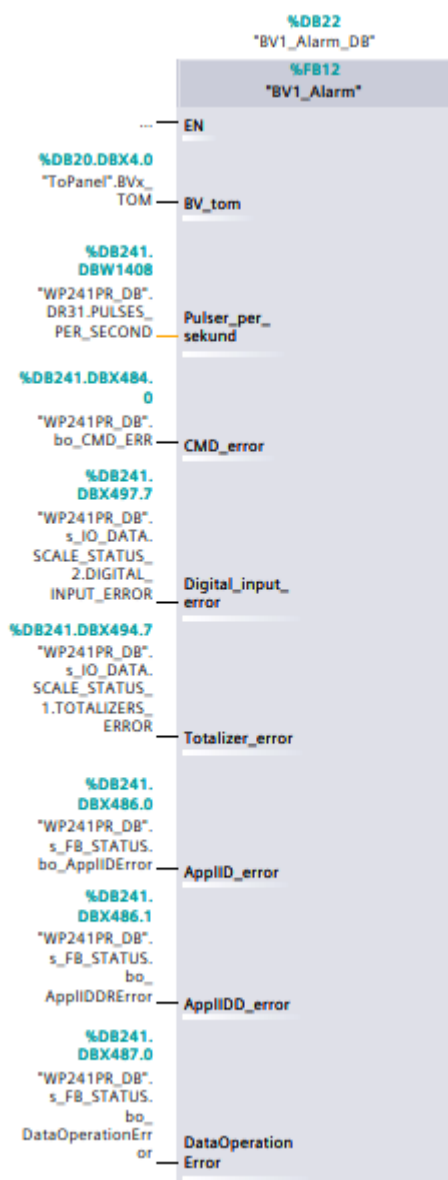
Nätverk fyra är ett alarm som aktiveras om nödstopp knappen påverkas (Bilaga 5.3.4). Alarmet finns i array position tre och aktiveras genast när nödstopp knappen påverkas. Nödstoppknappen bryter strömmen vid aktivering vilket betyder att när signalen är hög är inte nödstoppen aktiverad och när signalen är låg är nödstoppen aktiverad, därmed måste ingången inverteras.

Nätverk fem är ett alarm som aktiveras om användare försöker starta processen, men glömt stänga av kalibreringsläge (Bilaga 5.3.5). Alarmet aktiveras om HMI inte är i huvudmeny, kalibreringsläge från PLC och HMI inte är aktivt och kalibreringsknappen är aktiverad.

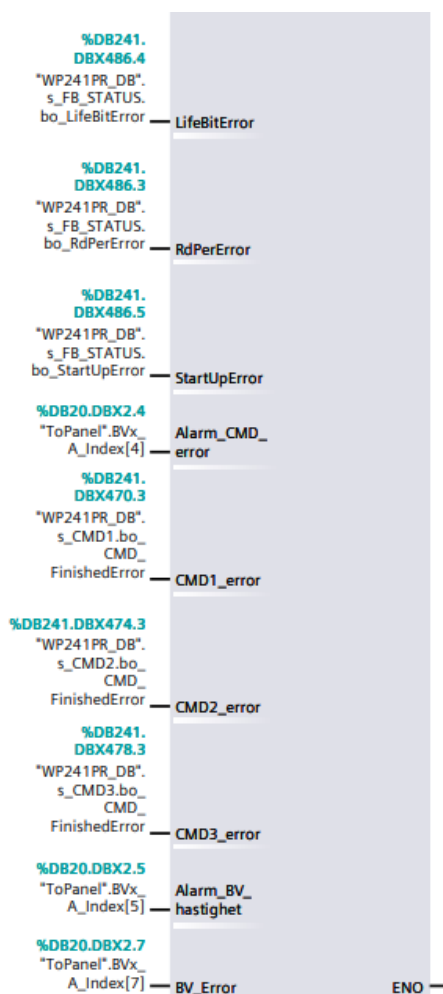
Nätverk sex aktiverar alarmeringar från PLC (Bilaga 5.3.6). Om någon av alarmen i arrayn aktiveras kommer denna funktion aktivera alarmeringsminne, stänga av HMI körning, HMI uppvärmnings och uppvärmning_aktiv-minnen samt byta fönster till alarmlogg

Nätverk sju inaktiverar alla alarmminnen (Bilaga 5.3.7). Om Alarm_Ack minnet aktiveras kommer alla alarm kvitteras. Alarm_Ack aktiveras om användare trycker Ack All eller tillbaka i alarmloggfenstret.

Bandvågsspecifika alarm innehåller tre nätverk (Figur 47 och Figur 46). Denna funktion är modulär, vilket betyder att denna funktion har massor av in- och utgångar som kopplas till dess specificerade variabler. Alarmen läses in separat i HMI:n och därmed behövs ingen aktivering i PLC:n, endast en OR sats som slår ihop alla bandvågsalarm och indikerar i PLC:n att bandvågen har något fel.



Figur 47. BVx alarmfunktion del 1.



Figur 46. BVx alarmfunktion del 2.

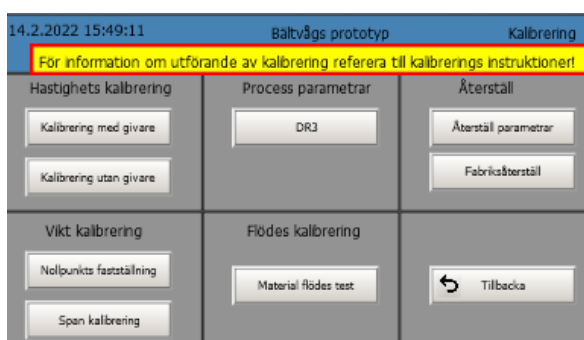
Det första nätverket tar hand om aktivering av CMD-alarmet (Bilaga 4.5.1). CMD alarmet skiljer från resten av alarmeringen i och med att alarmet inte behöver kvitteras och alarmloggen inte öppnas. CMD-alarmet öppnar istället ett pop-up fönster, där användaren kan kvittera att felet noterats.

Nätverk två alarmerar för att bandvågen har ett hastighetsfel (Bilaga 4.5.2). Hastighetsfelet aktiveras om bandvågens hastighet går utanför inställda gränsvärden och bandvågen har varit i körning fem sekunder. Om detta fel uppstår tyder det på att något är fel med motorn eller programmet som styr motorn.

I det tredje nätverket samlas alla bandvågsmodulens fel till ett minne (Bilaga 4.5.3). Detta görs för att underlätta kvittering och alarmering om bandvågsmodulen får något fel.

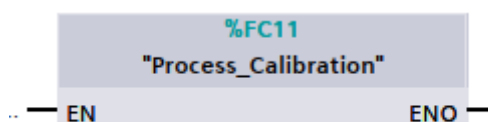
7.7 Kalibrering

Alla bandvågmodulens kalibreringar kan utföras via HMI:n. Kalibreringsfunktionen består av elva fönster. Kalibreringsfönstren byggs upp av ett huvudkalibreringsfönster, från vilket användaren kan navigera till önskad kalibrering (Figur 48). För bandvågen ska hastighet, nollpunkt, vikt och flöde kunna kalibreras, utöver dessa kan användare även ändra basparametrar och återställa enhet tillbaka till försäljnings- och fabriksinställningar. Kalibreringsläget är vanligt vis låst från användare och måste låsas upp genom att påverka en fysisk kalibreringsvippa och stoppknappen. Detta kommer byta färg på kalibreringsknappen i huvudmenyn, vilket i sin tur tillåter användare att komma åt kalibreringsmenyn. För denna bandvågsmodule finns, som tidigare nämnts, flera olika kalibreringar. Dessa kalibreringar körs från kalibreringsmenyn. Kalibreringsspecifika fönster och funktionalitet kan iaktas från HMI instruktionerna (Bilaga 6.5).

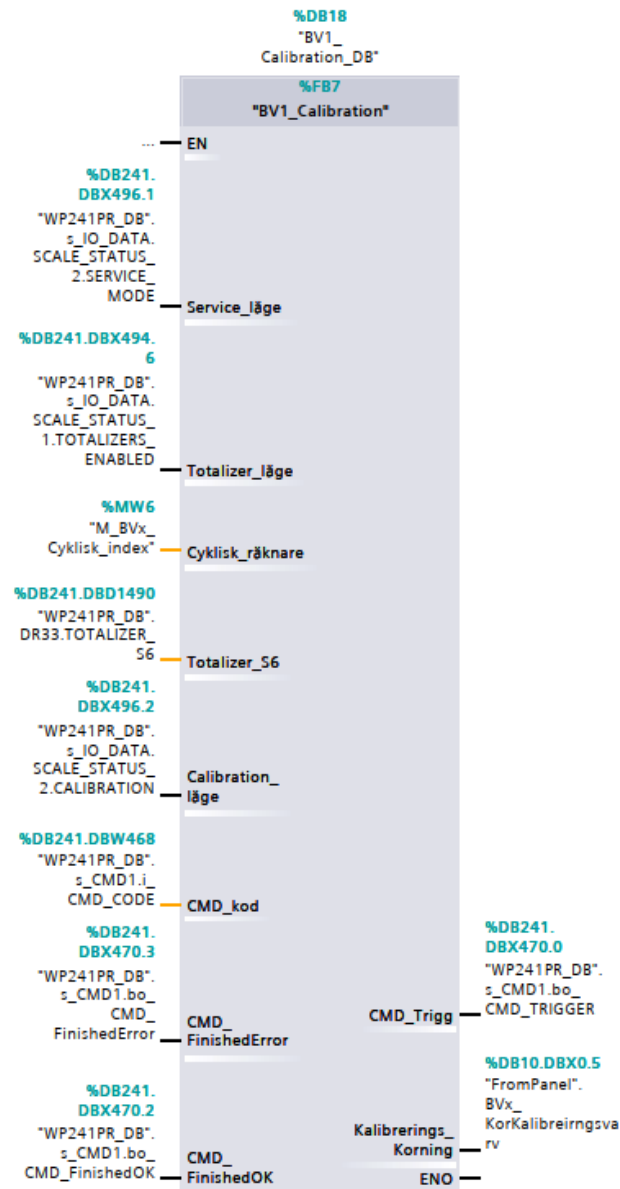


Figur 48. Kalibreringsdisplay.

Även här består PLC programmet av en bandvågsspecifik och process-specifik del. Den process-specifika delen sköter om baskalibreringsfunktioner, medan den bandvågsspecifika delen har att göra med kommunikationen med den bandvågsmodule som ska kalibreras. Process-specifika funktionen är en simpel funktion som glömmer sina värden efter körning (Figur 49). Bandvågsspecifika funktionen har in- och utgångar som länkas ihop med ifrågavarande moduls funktionsblock (Figur 50).



Figur 49. Processkalibreringsfunktion.



Figur 50. Bandvågsspecifik kalibreringsfunktion.

Den process-specifika funktionen byggs upp av endast ett nätverk (Bilaga 5.5.1). I detta nätverk aktiveras kalibrering_PLC-minnet, som berättar för resten av processen att kalibrering pågår. Detta minne hålls aktivt så länge som kalibreringsknappen är påverkad och kalibreringsmenyn är öppen.

Bandvågs-specifika funktionen byggs upp av sex nätverk som sköter om bland annat aktivering av serviceläge, inaktivering av totalräknare och kommunikering av kommandon.

Det första nätverket sköter om aktivering och inaktivering av serviceläget (Bilaga 4.6.1). Serviceläget aktiveras genom att skicka kommandokoden 1 till bandvågsmodulem och inaktiveras genom kommandokoden 2. Koden skickas iväg om serviceläge inte är aktivt, indikation för kalibreringsläge från både HMI och PLC är aktivt och pulssignalen som skapas i fjärde nätverket är högt. Serviceläget inaktiveras om HMI:n indikerar på att användare lämnat kalibreringsmeny, serviceläge är aktivt och ingen pulssignal är hög.

Det andra nätverket sköter om inaktivering och aktivering av totalräknare (Bilaga 4.6.2). När bandvågen sätts i kalibreringsläge inaktiveras totalräknare för att förhindra att kalibreringsmaterialet räknas med i total körda materialet. Totalräknare inaktiveras om serviceläge är aktivt, totalräknare är aktiva och pulssignalen går hög. Totalräknare inaktiveras om HMI:n indikerar på att användaren lämnat kalibreringsmenyn, totalräknare inte är aktiva, serviceläge inte är aktivt och ingen pulssignal är hög.

Det tredje nätverket lagrar den senaste skickade kommandokoden i ett temporärt minne, (Bilaga 4.6.3). När en kommandokod skickats lagras den i en temporär variabel för att förhindra vissa koder från att skickas upprepade gånger, på samma gång återställs även minne som indikerar på att CMD koden är körd. Detta minne används bland annat för att förhindra att serviceläge-kommandokoden skickas upprepade gånger.

Det fjärde nätverket kontrollerar CMD trigg-minnet (Bilaga 4.6.4). Som tidigare beskrevs i vägningsfunktionen måste CMD koden triggas för att skickas till vågmodulem. Denna funktion sköter om trigg-minnet och ett pulsminne för kalibreringen. Pulsminnet och trigg-minnet går högt varje gång som vägningsfunktionens pulseringsindex når 4, kalibreringsläge är aktivt från både HMI:n och PLC:ns sida och CMD koden inte är samma som tidigare CMD kod. Ett undantag har skapat för CMD koder som är över 2000. Dessa koder hanterar inläsning eller skrivning till dataregister och dessa får gärna uppdateras. En off-timer har också inkluderats för övervakning av aktiv kalibrering från HMI och PLC i och med att en sista puls behövs för att inaktivera serviceläge och aktivera totalräknare.

Det femte nätverket utför en kalkylering av korrektionsfaktor, som behövs vid materialtestnings skedet (Bilaga 4.6.5). Korrektionsfaktorn räknas ut genom att dividera det vägda materialet med vad processen fått för värde vid genom körning.

Det sjätte och sista nätverket aktiverar kalibreringskörningen (Bilaga 4.6.6). Kalibreringskörning är något som vissa kalibreringar använder, denna körning är aktiv så länge som bandvågsmodulem indikerar att en kalibrering pågår. Funktionen styr ett bältkörningsminne, som sedan aktiverar motorn i körningfunktionen.

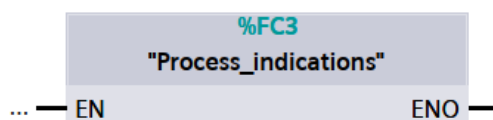
7.8 Indikationer

Indikations funktionen har skapat med tanke på vidare utveckling av prototypen. Denna funktionalitet består av ett fönster som ska ge användaren processinformation, såsom namnet på bandvågen, ibruktagningsdatum, körtid för dagen, körtid sedan ibruktagning, körd vikt för dagen och körd vikt sedan ibruktagning. Detta fönster är endast informationsbaserat och har ingen funktionalitet i processkörningen.

14.2.2022 15:48:54		Bältvågs prototyp	Indikatorer
Vågnamn	Bältvågs prototyp		<input type="button" value="Tilbäcka"/>
Ibruktagnings datum	02.02.2022		
Tid sedan start			
Total kördvikt sedan start:	<input type="text" value="#,##Kg"/>		
Användnings tid			
Total kördvikt sedan ibruktagning:	<input type="text" value="#,##Kg"/>		

Figur 51. Informationsdisplay.

I PLC programmet finns fler funktioner än de som syns på användargränssnittet. Funktionaliteten är uppdelad i process-specifika och bandvågsspecifika indikationer. Processindikationerna byggs upp av tre nätverk som sköter om styrning för de fysiska indikations lamporna (Figur 52).



Figur 52 Processindikationsfunktion.

Det första nätverket sköter om aktivering av Start_klar-indikationslampan (Bilaga 5.6.1). För att indikationsbelysning ska starta måste processen vara antingen i uppvärmnings- eller parameterinställningsskede, inga alarm får vara aktiva och ingen körning får pågå. Start_klar-lampan aktiveras även när processkörningen pausats via HMI:n

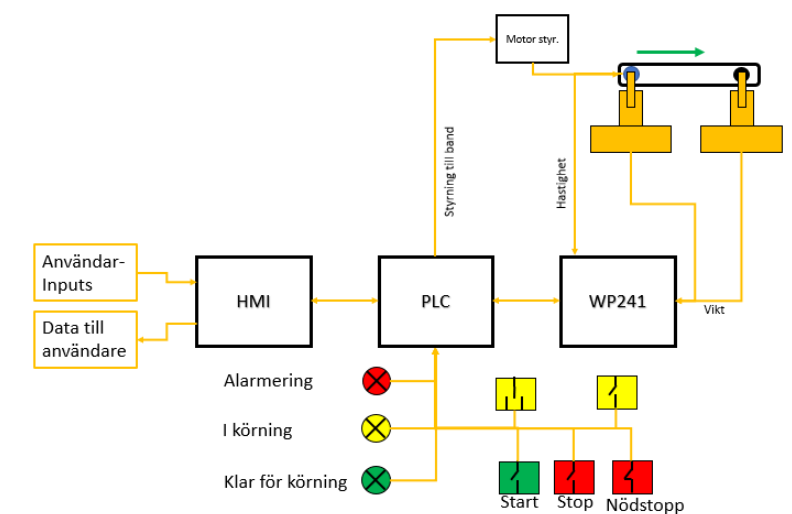
Det andra nätverket sköter om aktivering av Körning_pågår-indikationslampan (Bilaga 5.6.2). Körning_pågår-lampan startas om processkörning, uppvärmning eller någon manuell körning pågår, inga alarm är aktiva och processkörningen inte är pausad.

Det tredje nätverket sköter om aktivering av Alarm_aktiv-indikationslampan (Bilaga 5.6.3). Lampan startar alltid när processen alarmeringsminne är aktivt.

Bandvågsspecifikaindikationerna omfattar en funktion som tar hand om totalräknarna för processinformation. Denna funktion tar hand om totalräkning av körtid, körd massa och antal körningar för både ifrågavarande dag och sedan ibruktagning. Denna funktion färdigställdes inte och kommer behandlas senare i dokumentet (Kapitel 9).

8 Testning

På grund av att arbetet bygger på en första prototyp och inte skapandet av en process för en kund, har inte resurser använts för att skapa en fullstor bandvågsprocess. Istället har en nerskalad testprocess skapats (Figur 53). Styrsystemet består fortfarande av samma komponenter och fungerar på samma sätt som beskrivits i tidigare kapitel. Själva processen består här av ett kort bälte som körs av en motor. Motorn styrs av en frekvensomriktare. Hastigheten mäts med hjälp av ett kugghjul, som fäst i den roterande axeln och en induktiv givare som ger pulser för varje kugg. Hela systemet bärs upp av två lastceller som väger hela processen. Test kan utföras genom att helt enkelt släppa ner material på bältets ena sida och låta bältet köra igenom materialet.



Figur 53. Testprocess.

Testningen av systemet har genomförts i två egentliga delar. Den första delen handlade om att testa systemets olika delar, vartefter dessa delar blev färdiga eller någon ändring gjordes. Den andra delen handlade om att skapa en testsekvens. Testsekvens gicks igenom för att konstatera en funktionerande helhet.

Den första delen utfördes genom att skapa en lista där problem skrevs upp och åtgärdades steg för steg. Den mest komplicerade delen var den sekventiella körningen, i denna del uppkom de största problemen, detta på grund av att många delar måste samarbeta.

Körningstester utfördes med ett förhandsvägt testmaterial, i detta fall bultar och muttrar. Testmaterialet vägdes till två kilogram och kördes igenom miniatyrprocessen upprepade gånger. Ett stabilt flöde kunde inte skapas, vilket kan ha en inverkan på testningens noggrannhet.

Uppvärmningstestningen var väldigt simpel, här ställdes tiden till tio minuter varefter körningen startades. När timern räknat färdigt säkerställdes att alla uppvärmningsrelaterade variabler återställdes samt att uppvärmningen kunde startas igen utan fel.

Alarmeringen testades genom att aktivera olika alarm vid olika tillfällen. Testningen tog plats till stor del i uppvärmnings- och körningsfunktionerna, här testades alarm som parametrar inte inställda, manuell/auto inte valt, hastighets fel, nödstopp- och kalibreringsläge aktivt.

Kalibreringen testades genom att köra de olika kalibreringarna i tur och ordning. På samma gång som kalibreringarna kördes iaktogs processen från både SIWATOOL och TIA-portalens online-läge.

8.1 Testsekvens

En huvudsaklig testsekvens skapades för att fastställa funktionaliteten. Denna testsekvens fokuserar i huvudsak på att användargränssnittet ska fungera i par med processen. Testsekvensen kan iakttas bland bilagorna (Bilaga 7). Testsekvensen byggs upp, som en tabell med checkboxar. Checkboxarna kryssas i för konstaterad funktionalitet. Sekvensen byggs upp i sex delar, huvudmeny, körning, uppvärmning, alarmering, kalibrering och information. För alla delar testas att de fysiska knapparna utför dess designerade uppgift, indikationslampor visar rätt och alarm fungerar. I och med att kalibreringen består av mycket mera detaljerade uppgifter utfördes noggrannare tester för dessa funktioner. Kalibreringstesterna omfattade testning med hjälp av SIWATOOL och där man kunde konstatera en lyckad kalibrering.

9 Resultat

I detta kapitel presenteras slutresultatet från en objektiv synvinkel och resultat som erhöles genom testningen behandlas. Som förväntat uppnådde prototypen funktionskraven, som ställdes i början av detta arbete. Några delar av programmet färdigställdes inte på grund av brist på tid och testutrustning.

Informationsfönstret är en av funktionerna som inte färdigställdes på grund av tidsbrist. Tanken med funktionen var att användaren skulle kunna se totalkörtid och materialmängd under dagen och sedan ibruktagning. Problem uppkom bland annat vid återställning av totalräknare och loggning av tiden. Återställning av totalräknare löstes till sist genom att skapa ett dagligt avbrott som sker varje dag klockan 23.00. Tiden räckte tyvärr inte till för testning av avbrottet. Några problem som noterades för denna typ av återställning var t.ex. vad kommer hända om strömmen bryts under natten. För att arbeta runt detta problem skapades återställning av totalräknaren vid power-up. Körtiden lämnades också ogjord i och med att ett vettigt sätt att mäta tiden inte hittades, några olika metoder övervägdes. Metoder som kunde tänkas fungera är on- eller off-delay med lite implementering. TIA-portalens har också ett körtidsblock som kunde testas.

Testningen av de övriga funktionerna ledde till förväntat resultat, och funktionerna kunde konstateras fungera bra. Vid testningen sattes hög fokus på att granska minneslägen vid slutet och start av olika funktioner, detta gjordes via TIA-portalens online-läge. När en av de huvudsakliga funktionerna avslutas är det viktigt att alla funktionens minnen återställs. Största delen av problemen, som uppkom vid testningen, var slarvfel vid återställning av minne. Detta ledde till en del olika problem.

Övriga problem som uppkom vid testningen var representation av decimaltal i HMI:n. I arbetets tidigare skede lyckades inte representationen av realtal, detta på grund av att antal decimaler måste anges i HMI:n vid skapande av representationsrutor. Real talet byter också aningen form vid delning av data mellan HMI och PLC. I HMI:n kallas datatypen float medan den i TIA-portalen kallas real.

Testning av kalibrering medförde en del utmaningar, i och med att mycket händer under ytan. För kalibreringen var största problemet säkerställande av att önskad kommandokod körs. Detta kunde konstateras genom att ställa TIA-portalen i online-läge och övervaka WP241-datablocket. Testningen av alla kalibreringar kunde inte utföras, dels på grund av att tiden började nå sin gräns och all utrustning inte fanns tillgänglig.

En mera fysisk iakttagelse, som uppkom vid testningen, var lastcellsplaceringen. I och med att lastcellerna bar på hela bältkonstruktionen och inte endast styrhjulen i mitten av bältet uppkom problem med vägningsområdet. Vid körning av testmaterial kunde iakttas att vikten ökades fast materialet redan fallit av bältet. Den slutgiltiga vikten var ändå rätt tack vare alla korrektioner som bandvågsmodulen gjorde. Tester som utfördes kan ändå inte täcka alla aspekter som processen kommer utsättas för, t.ex. vad som händer när materialflödet är större och mera konstant samt om körningen varar under en längre tidsperiod. För vidare testning kan det finnas skäl att tänka om vad gäller placeringen av lastcellerna, till en mera traditionell lösning.

10 Diskussion

Jag anser att målet för detta arbete uppfylldes. En fungerande bandvågsprototyp skapades, manualen för ibruktagningen gjordes och behov för vidareutveckling analyserades.

Detta arbete bestod av väldigt många delar, som sedan skulle passa ihop med varandra och fungerar som en helhet. Vissa delar gick smidigare än andra och de största osäkerheterna kom fram vid PLC-programstruktureringen. Vid struktureringen var meningen att skapa ett modulärt program, med funktioner som inte skulle vara för beroende av varandra. Idén var att kunna kopiera funktionerna för användning av flera bandvågsmoduler. Denna funktionalitet konstaterades aldrig och flera av funktionerna har ännu rum för optimering. Eftersom detta är en första prototyp och all hårdvara inte existerade för att kunna testa de olika scenariona, drogs en gräns och behoven av vidareutveckling identifierades.

De huvudsakliga vidareutvecklingsbehoven gäller modulering av funktionerna. Om t.ex. två bandvågsmoduler ska användas måste processindikationsminnen slås ihop på samma sätt som för uppvärmning_aktiv-minnet. Dessa hopslagningar måste ännu identifieras och skapas. I användargränssnittet måste också ett sätt att styra de olika bandvågarna skapas. Här kunde t.ex. en dropp-down lista läggas till i informations fliken upp i varje fönster. Med dropp-down listan kunde användaren identifiera och välja vilken bandvåg som påverkas.

För moduleringen finns också infofönstret som kräver vidare utveckling. I detta fönster var tanken att man kunde se körtiden och vikten för en dag och sedan ibruktagning. För skapandet av denna funktion räckte inte tiden till och endast några få tester gjordes. Till totalviktsindikeringen användes totalräknare S1 och S2 i bandvågsmodulen. I och med att

S1 inte kan nollas utan fabriksåterställning, passar denna till användning för körda vikten sedan ibruktagning. S2 användes för att indikera körda vikten för dagen, denna räknare nollas genom att ha ett dagligt avbrott som nollar räknare varje dag klockan 23.00. För mätningen av tiden spekulerades användning av en runt-time funktion som mäter tiden i timmar, eventuellt kunde även en timer i samarbete med en räknare användas. När totalräknare är aktiv är ett minne i bandvågsmodulen högt, detta minne kunde användas till att aktivera tidtagningen.

Utöver vidare utveckling kräver programmet ännu extensiv testning på en verklig process. I och med att testningen utfördes på en miniatyrprocess kan här finnas olika aspekter som inte tagits i beaktande för körning av en hel process. Här är det största bekymret hur processen reagerar när materialet faller från huvudmataren till bandvågsmataren samt, hur denna bandvåg reagerar på byte mellan grov- och finmatningen. Eftersom bandvågen kalibreras för ett visst flöde kan här förekomma problem om flödet minskas. Syftet med flödesminskningen kunde därmed leda till en försämring av noggrannheten istället för en ökning. Med en verklig process kunde även bandvågens noggrannhet och repeterbarhet testas. Testningen av noggrannhet och repeterbarhet är en viktig faktor som ger en bra bild över processen pålitlighet, tyvärr kunde detta inte testas med miniatyrprocessen eftersom ett mera stabilt materialflöde till bandvågsbältet skulle behövts.

I samband med mera extensiva tester kan också behovet av nollpunktkalibreringen bedömas. För bandvågar är det i allmänhet rekommenderat att en nollpunktskalibrering utförs regelbundet och därmed kunde detta ha arbetats in i t.ex. uppvärmningssekvensen.

Realisering av utvecklingsförslag och färdigställning en produkthelhet sker när en beställning på en bandvågsprocess är placerad och en kundspecifik funktionsspecifikation är skapad. Detta kommer antagligen ställa nya krav på systemet. På samma gång som programmet ändras för att möta kundens behov kan utvecklingsförslagen hanteras. I och med att detta arbete lyckats och en fungerande prototyp skapats, kommer färdigställning av en hel bandvågsprocess underlättas avsevärt.

11 Källförteckning

- Beijer. (u.d.). *IX HMI software*. Hämtat från <https://www.beijerelectronics.com/en/Products/software/ix-hmi-software> den 05 03 2022
- Electrocon. (u.d.). Hämtat från <https://electrocon.fi/sv/> den 29 10 2021
- Haag, B. (1998). *Industriell systemteknik*. Lund: Studentlitteratur AB.
- Hanssen, D. H. (2015). *Programmable Logic Controllers : A Practical Approach to IEC 61131-3 Using CoDeSys*. John Wiley & Sons, Incorporated.
- Iqbal, K. (2017). *A First Course in Control System Design*. River Publishers.
- Knapp, E. D., & Langill, J. T. (2015). *Industrial Network Security (seconde edition)*. i *Introduction to Industrial Control Systems and Operations*. International Society of Automation.
- Omron. (u.d.). *Cylindrical proximity meter E2A datasheet*. Hämtat från Omron: https://www.ia.omron.com/data_pdf/cat/e2a_dsheets_d100-e1-01b.pdf?id=1883 den 2022 03 23
- Scarrott, J. (2015). *The Black Art Of Belt Weighing. The QMJ Group, 2*.
- Schneider Electric. (2016). *Getting started with ATV12*.
- Siemens. (2015a). *Easy book*. Hämtat från Siemens industry support: <https://support.industry.siemens.com/cs/document/39710145/simatic-s7-1200-easy-book?dti=0&lc=en-WW> den 23 03 2022
- Siemens. (2015b). *How do I calibrate SIWAREX WP241 with SIWATOOL v7*. Hämtat från Siemens industry support: <https://support.industry.siemens.com/cs/document/109474797/how-do-i-calibrate-a-siwarex-wp241-with-siwatool-v7-?dti=0&lc=en-WW> den 23 03 2022
- Siemens. (2017). *Siwarex WP241 manual*. Hämtat från Siemens industry support: <https://support.industry.siemens.com/cs/document/109750452/siwarex-wp241-electronic-weighing-system?dti=0&lc=en-WW> den 23 03 2022
- Siemens. (2018a). *Siemens S7-1200 Programmable logic controller system manual*.
- Siemens. (2018b). *Programming guidelines for S7-1200/1500*. Hämtat från Siemens industry support: <https://support.industry.siemens.com/cs/document/90885040/programming-guideline-for-s7-1200-s7-1500?dti=0&lc=en-WW> den 23 03 2022
- Siemens. (n.d.a). *SIWAREX SIWATOOL*. Hämtat från Siemens industry support: <https://new.siemens.com/global/en/products/automation/process-instrumentation/weight-measurement/software/siwatool.html> den 23 03 2022
- Siemens. (n.d.b). *Totally Integrated Automation Portal*. Hämtat från <https://new.siemens.com/global/en/products/automation/industry-software/automation-software/tia-portal.html> den 2022 03 03
- Siemens. (n.d.c). *Weighing module WP241*. Hämtat från <https://new.siemens.com/global/en/products/automation/process->

instrumentation/weight-measurement/weighing-electronics-simatic/weighing-module-siwarex-wp241.html den 2022 03 03

Zemic. (November 2020). *L6E loadcell datasheet*. Hämtat från Zemic Europe:
https://www.zemiceurope.com/media/Documentation/L6E_Datasheet.pdf den 03 23 2022

Bilagor

1.	Flödesscheman	5
1.1.	Sida 1 Huvudmeny	5
1.2.	Sida 2 Kalibreringsmeny.....	6
1.3.	Sida 3 Hastighetskalibrering utan givare.....	7
1.4.	Sida 4 Nollpunktskalibrering.....	8
1.5.	Sida 5 Hastighetskalibrering med givare	9
1.6.	Sida 6 Vikt kalibrering	10
1.7.	Sida 7 Materialtest.....	11
1.8.	Sida 8 Uppvärmning.....	12
1.9.	Sida 9 Process körning	13
2	Instruktioner för ibruktagning med SIWATOOL.....	14
2.1	Uppkoppling	14
2.2	Parametrar	15
2.3	Ordningsföljd på kalibreringarna	15
2.4	Fabriksåterställning.....	15
2.5	Parametrar	16
2.6	Hastighets kalibrering	17
2.6.1	Kal 1. Med känd puls konstant.....	18
2.6.2	Kal.2 Med okänd puls konstant.....	18
2.6.3	Kal.3 Användning av känd konstant hastighet.....	19
2.6.4	Kal.4 Användning av okänd konstant hastighet.....	20
2.6.5	Kal.5 Hastighet från extern CPU.....	21
2.7	Nollpunkts kalibrering med tom bandvåg.....	21
2.8	Spankalibrering.....	21
2.8.1	Genom användning av testvikt.....	21
2.8.2	Genom användning av testkedja.....	22
2.8.3	Användning av lastcellsdata	23
2.9	Materialtest	23
2.9.1	Materialtest	23
2.10	Definition av gränsvärden	24
3	Datavariabler.....	25
3.1	Signaler till HMI.....	25
3.2	Signaler från HMI.....	26
3.3	PLC tag-lista	27
3.4	WP241 Datablock	28

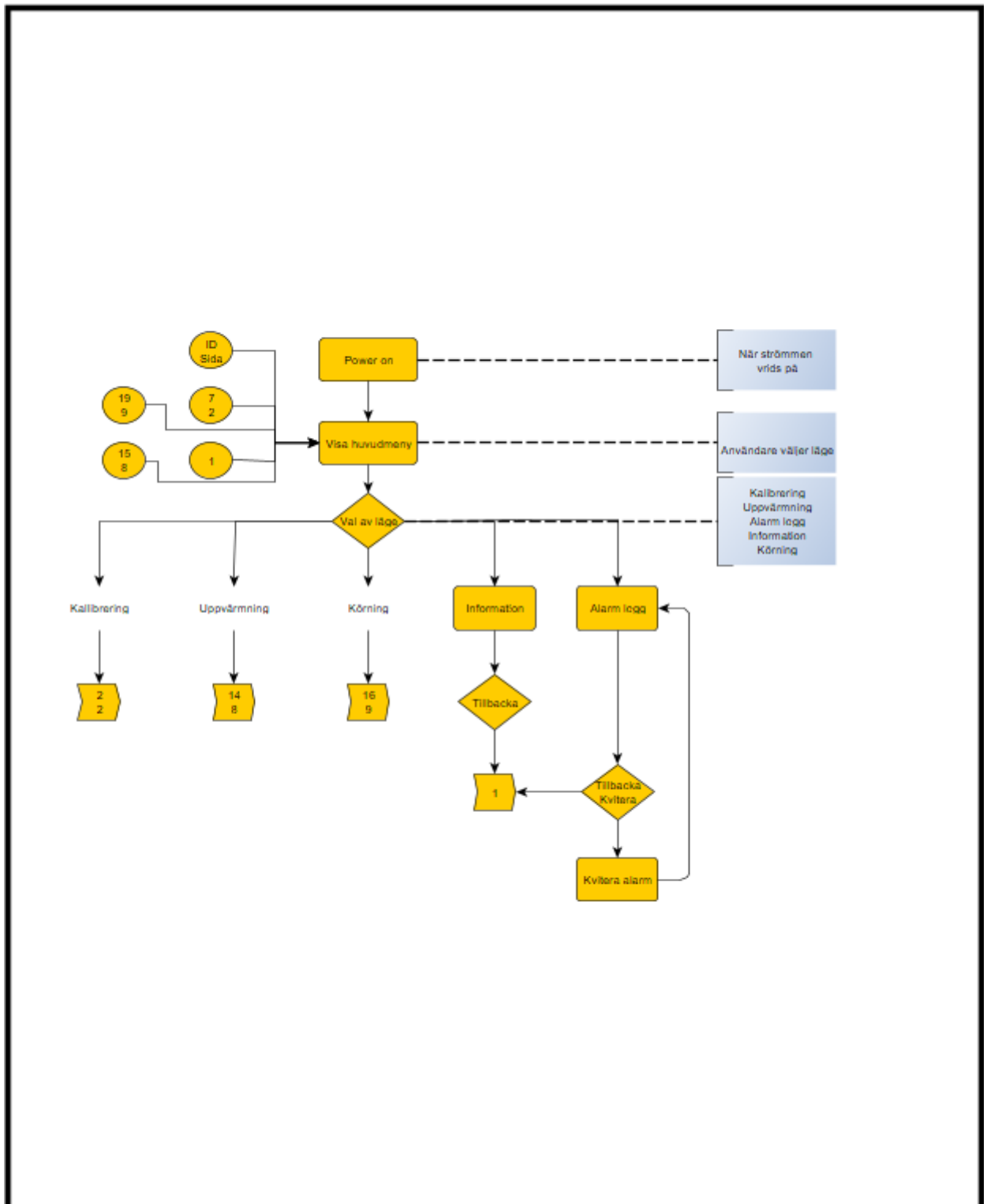
3.5	HMI Tag-lista	31
4	Bandvågs specifika funktioner	33
4.1	Counter	33
4.1.1	Nollning av räknare.....	33
4.1.2	Överföring av börvärde till temp vari.....	33
4.1.3	Negativ tals förhindring	34
4.1.4	Körnings räknare	35
4.1.5	Avslutad körning.....	36
4.2	Weigher.....	36
4.2.1	Tom last cell	36
4.2.2	Summering av nuvarande vikt och genomkörd vikt	36
4.2.3	Trigg pulssignal	37
4.2.4	Cyklisk inläsning av data	37
4.2.5	Uträckning av procentuell genomkörd vikt	38
4.2.6	Nollning av totalräknare S3	38
4.2.7	Negativ gränsvärdes övervakning	39
4.2.8	Parameter indikation	39
4.3	Feeder.....	40
4.3.1	Bandvågs bälte start stopp.....	40
4.3.2	Manuell körning av bandvågs bälte.....	40
4.4	PreHeat	41
4.4.1	Körning ok	41
4.5	Alarm	41
4.5.1	CMD alarm	41
4.5.2	Bandvågs hastighets fel.....	42
4.5.3	Bandvågs fel	43
4.6	Calibration	44
4.6.1	Service läges styrning.....	44
4.6.2	Totaliserar styrning	44
4.6.3	Tidigare komando kod.....	45
4.6.4	Trigg puls	45
4.6.5	Uträkning av korrektions faktorn	45
4.6.6	Kalibrerins körning.....	46
4.7	Restart.....	46
4.7.1	Nollning av param	46
5	Process-specifika funktioner	47
5.1	Feeder.....	47
5.1.1	Godkän körning	47

5.1.2	Körning av grovmatare.....	48
5.1.3	Körning av finmatning.....	48
5.1.4	Körning av pulsmatning.....	49
5.1.5	Manuell körning av M1.....	49
5.1.6	Motorstyrning.....	50
5.2	PreHeat.....	51
5.2.1	Börvärdes behandling.....	51
5.2.2	Klar/avbryt körning.....	52
5.2.3	Körning.....	53
5.2.4	Ärvärdes behandling.....	53
5.3	Alarm.....	54
5.3.1	Man auto inte invalt.....	54
5.3.2	Parametrar inte inställda.....	54
5.3.3	Bandvågen är inte tom.....	55
5.3.4	Nodstopp aktiverat.....	55
5.3.5	Kalibrering aktiv.....	56
5.3.6	Alarm aktivt.....	57
5.3.7	Alarm kvitering.....	58
5.4	Calibration.....	59
5.4.1	Aktivering av kalibrering.....	59
5.5	Indications.....	59
5.5.1	Klar för körning.....	60
5.5.2	I körning.....	60
5.5.3	Alarmering.....	61
5.6	Daily Interrupt.....	62
5.6.1	Nollning av räknare och vägare.....	62
5.7	Language.....	62
5.7.1	Nollning av språk index.....	62
5.8	Restart.....	63
5.8.1	Byte till huvud display.....	63
5.8.2	Nollning av räknare och vägning.....	63
6	Användarinstruktioner för bandvågsprocessens användargränssnitt.....	64
6.1	Körning.....	64
6.2	Uppvärmning.....	66
6.3	Alarm.....	66
6.4	Process information.....	67
6.5	Kalibrering av bandvågsmodul via HMI.....	68
6.5.1	Processparameter inställning.....	69

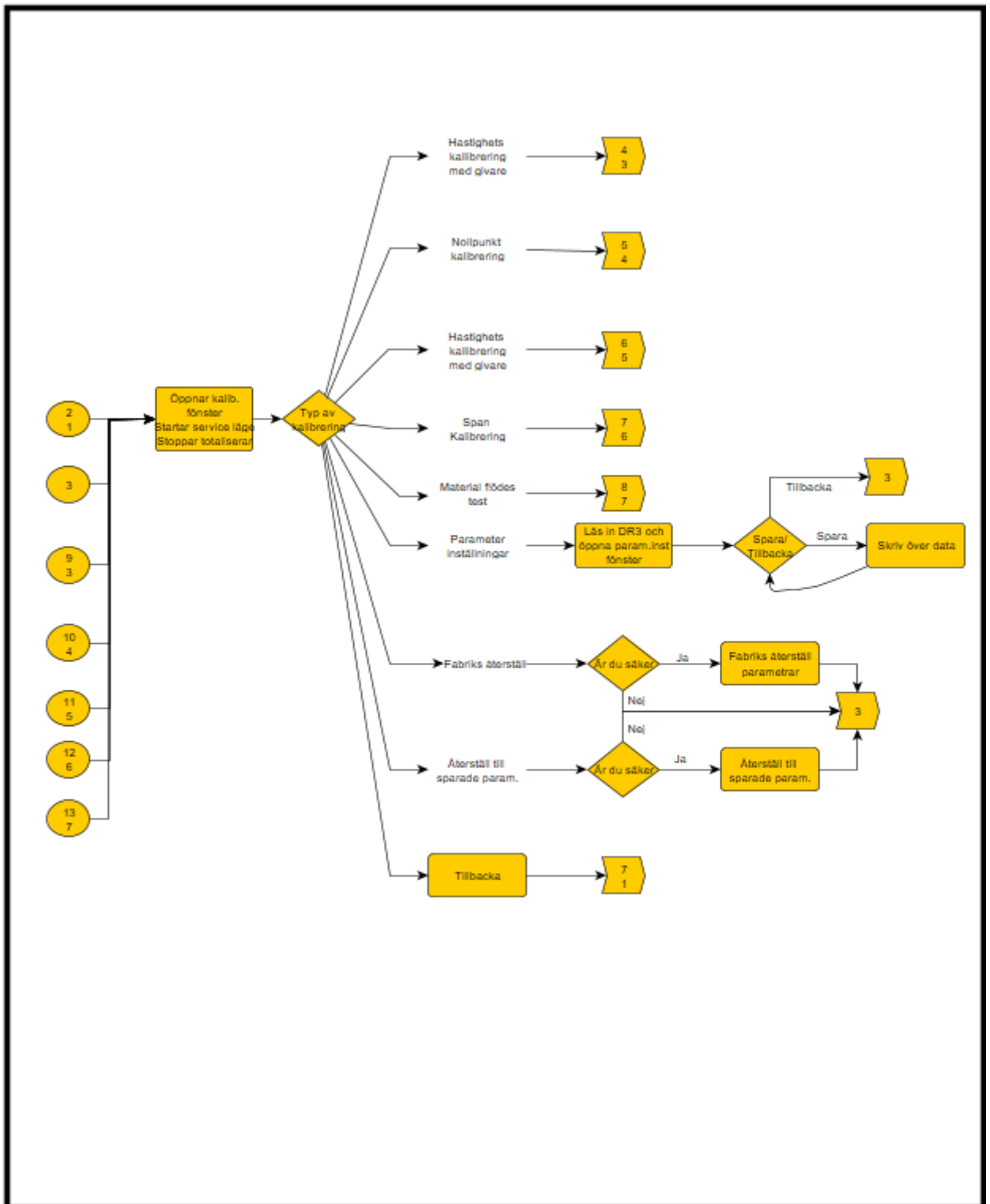
6.5.2	Hastighets kalibrering med givare.....	69
6.5.3	Hastighetskalibrering utan givare	71
6.5.4	Nollpunkts kalibrering.....	72
6.5.5	Span kalibrering	73
6.5.6	Material test	75
7	Test sekvens.....	77
7.1	Start.....	77
7.2	Körning (parameterinställnings skede)	77
7.3	Körning (körnings skede)	78
7.4	Uppvärmning.....	78
7.5	Alarm	79
7.6	Kalibrering.....	80
7.7	Process information.....	80

1. Flödesscheman

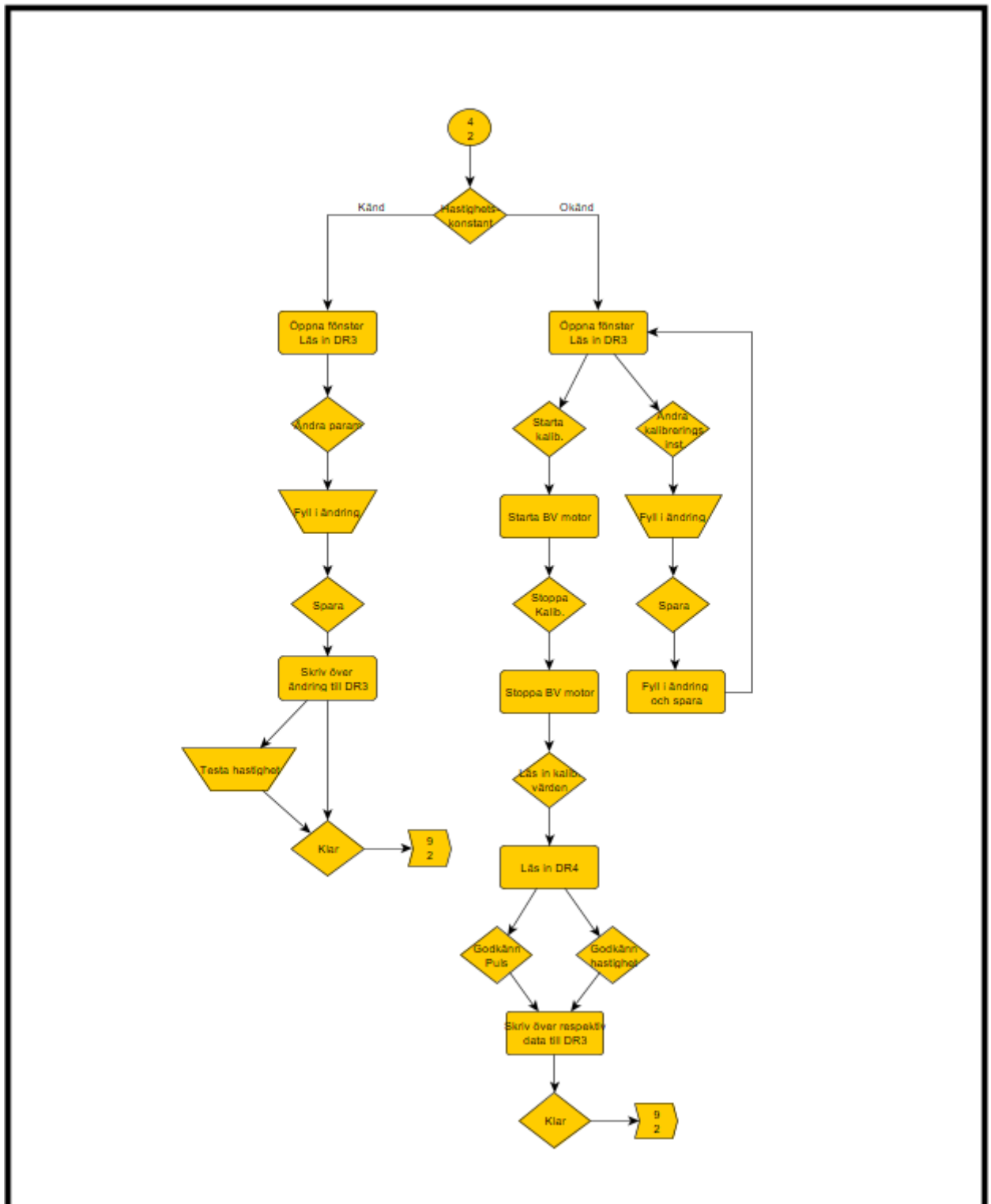
1.1. Sida 1 Huvudmeny



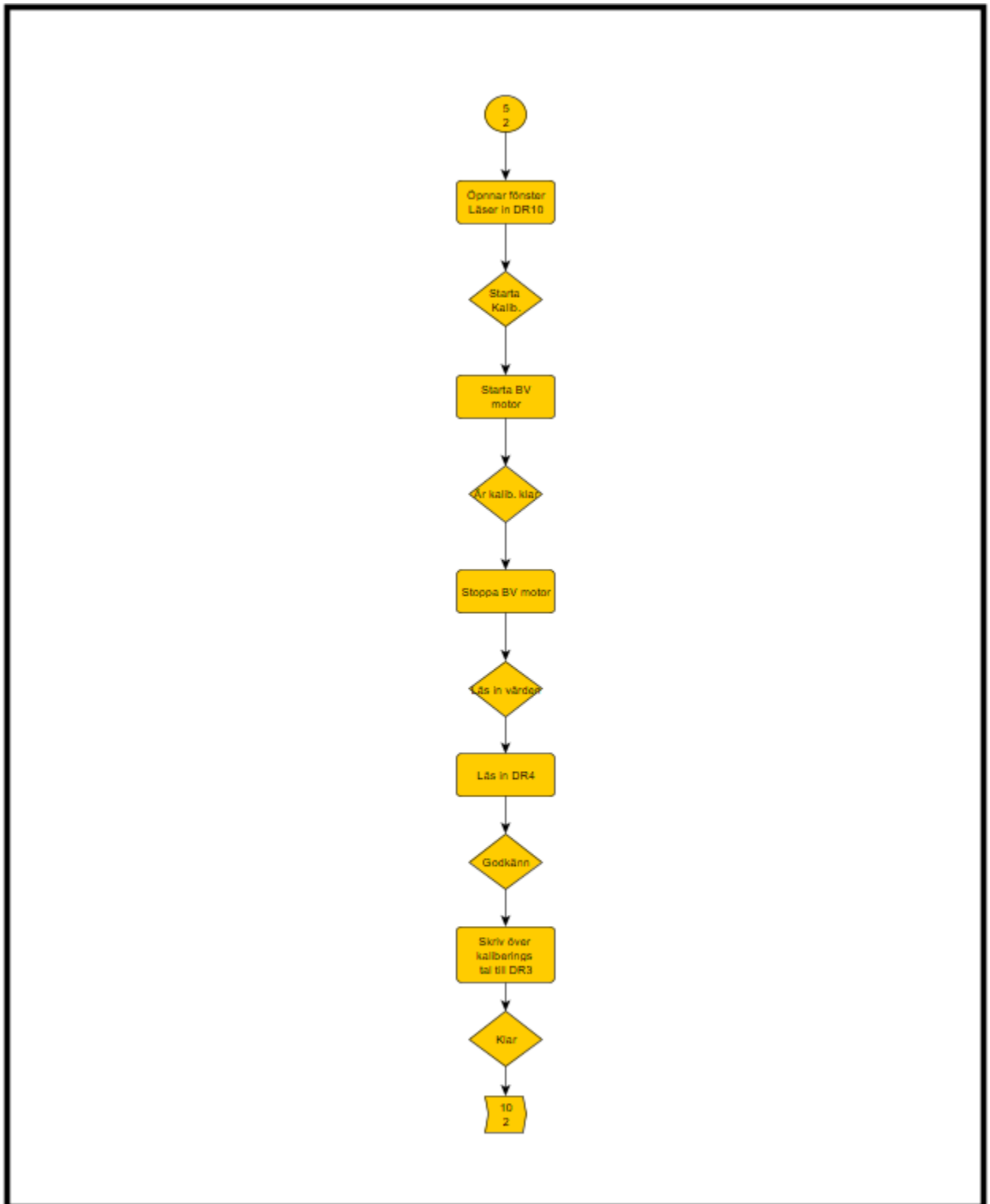
1.2. Sida 2 Kalibreringsmeny



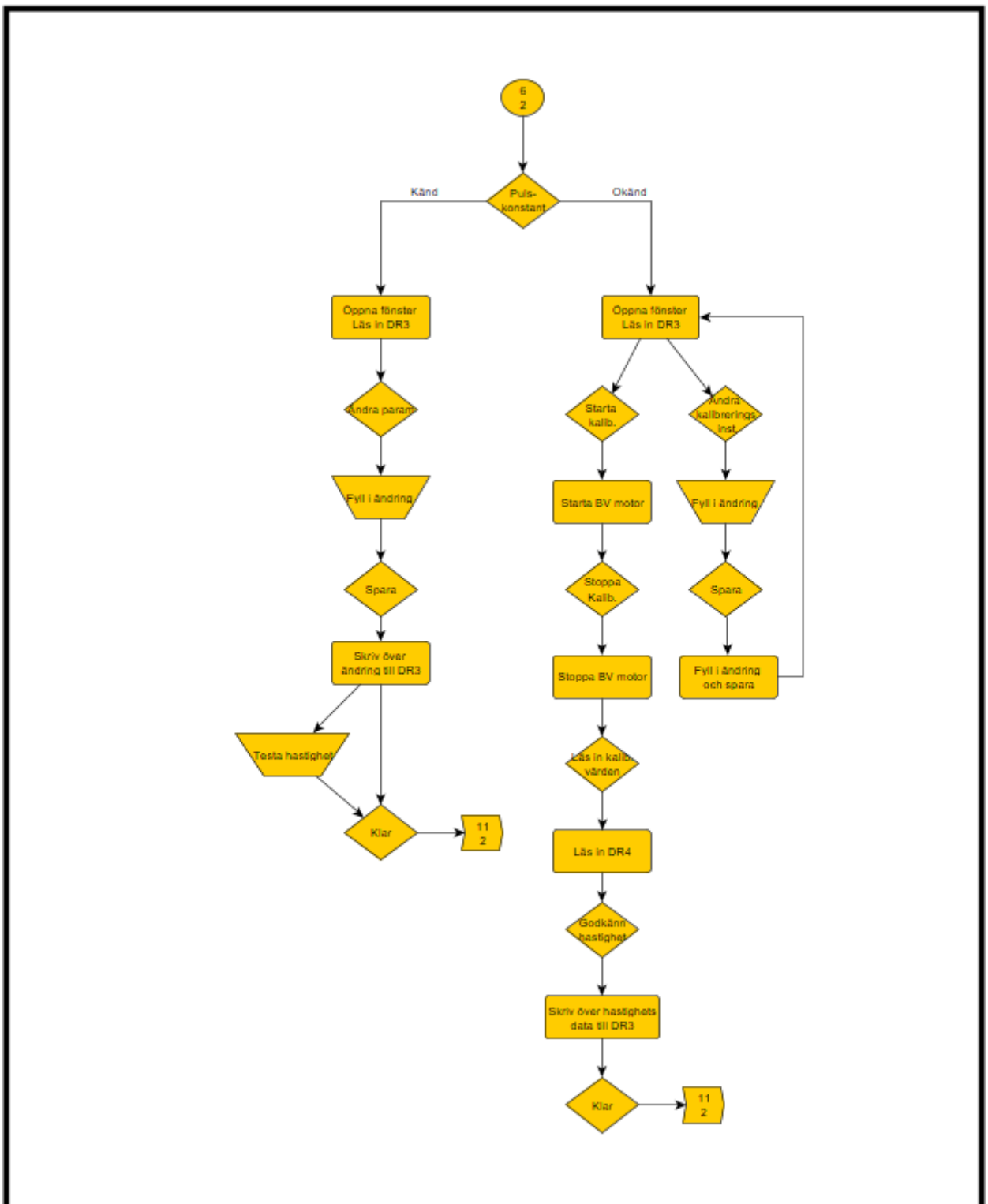
1.3. Sida 3 Hastighetskalibrering utan givare



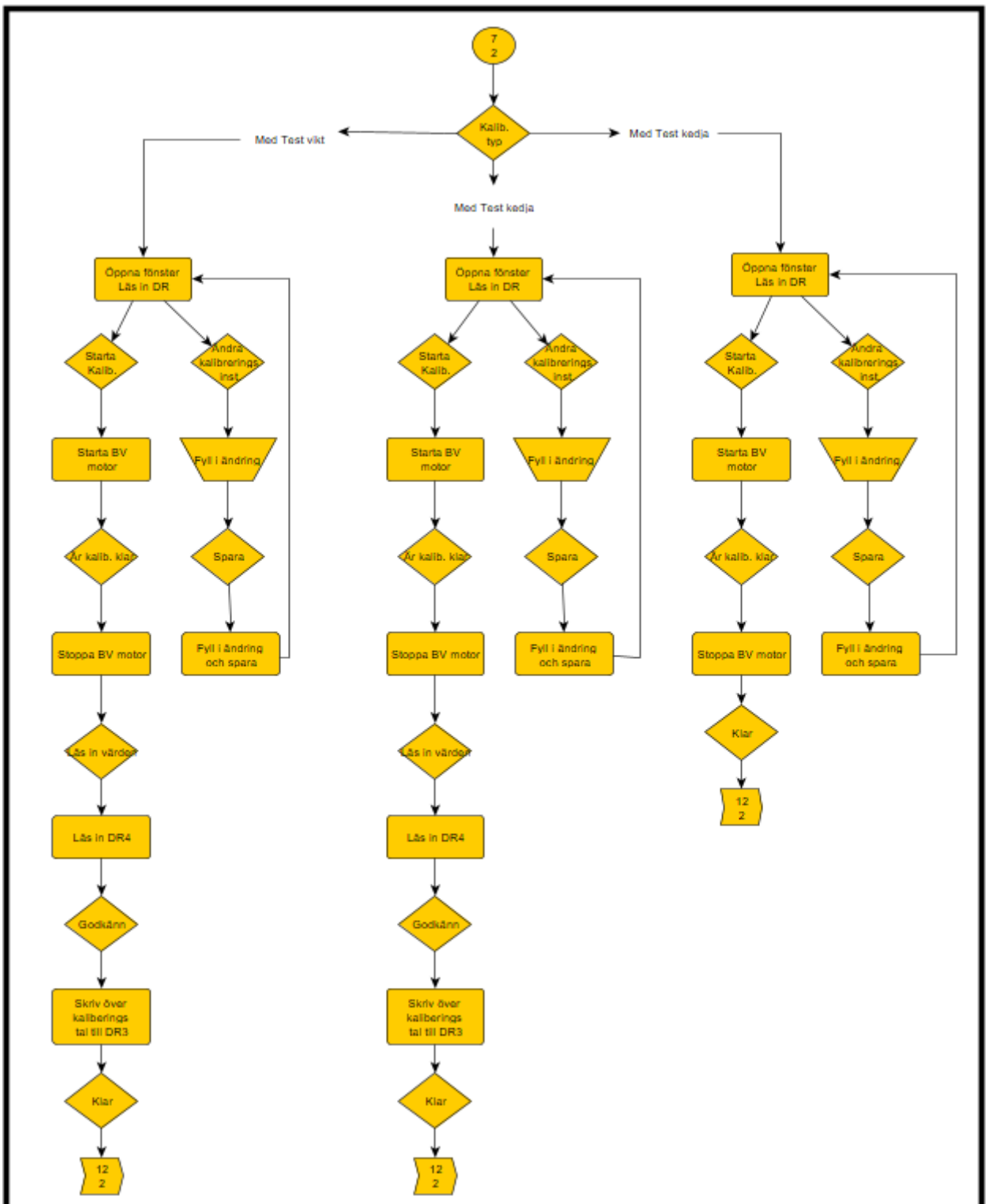
1.4. Sida 4 Nollpunktskalibrering



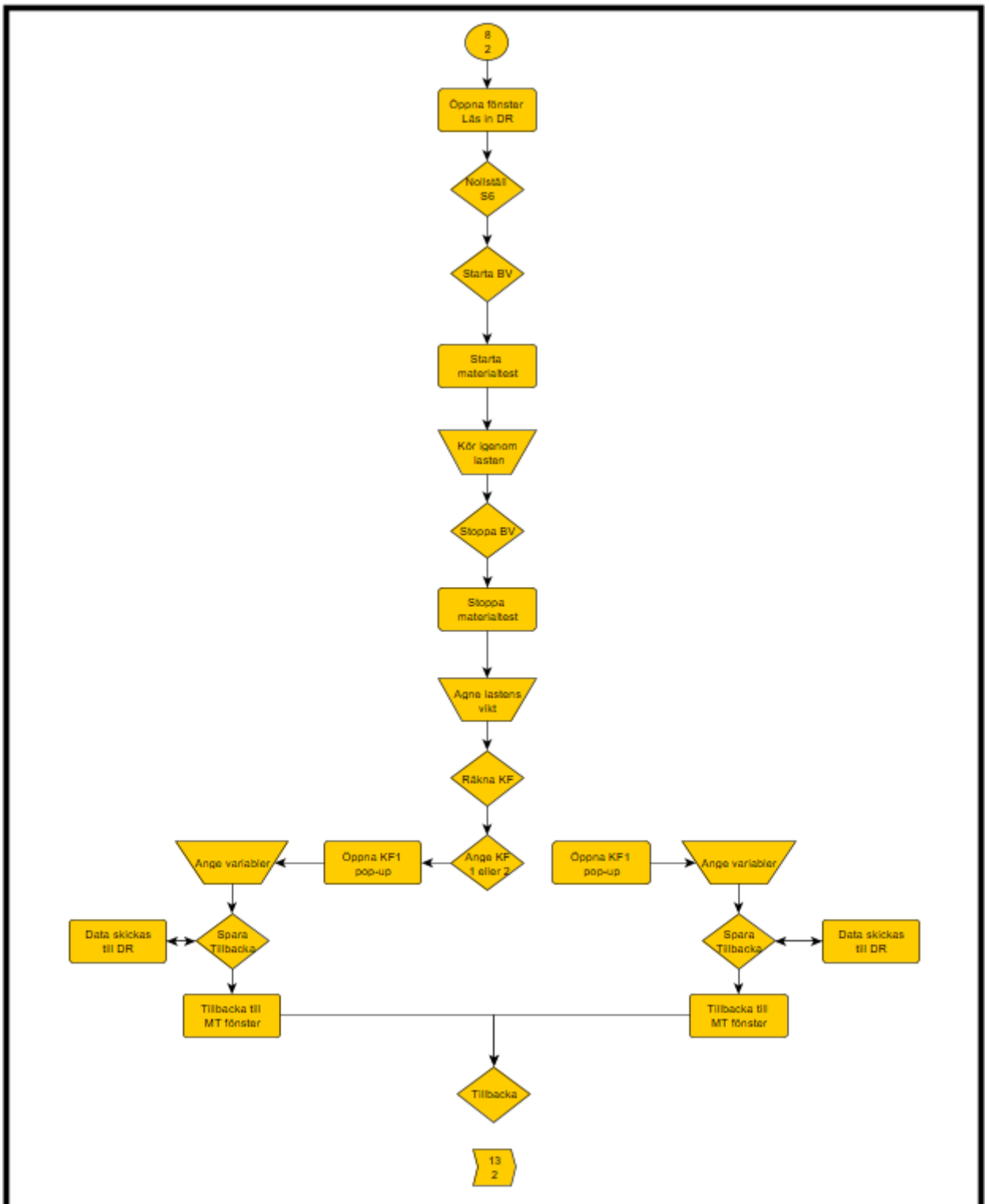
1.5. Sida 5 Hastighetskalibrering med givare



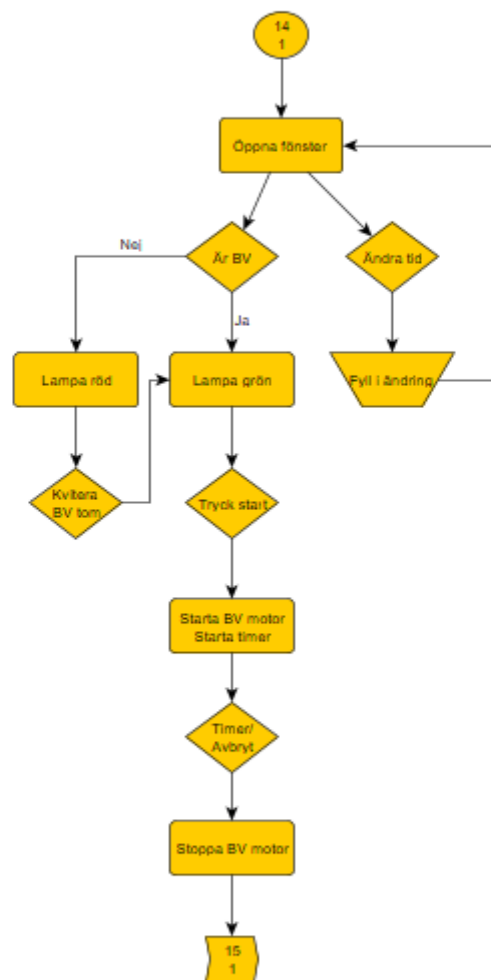
1.6. Sida 6 Vikt kalibrering



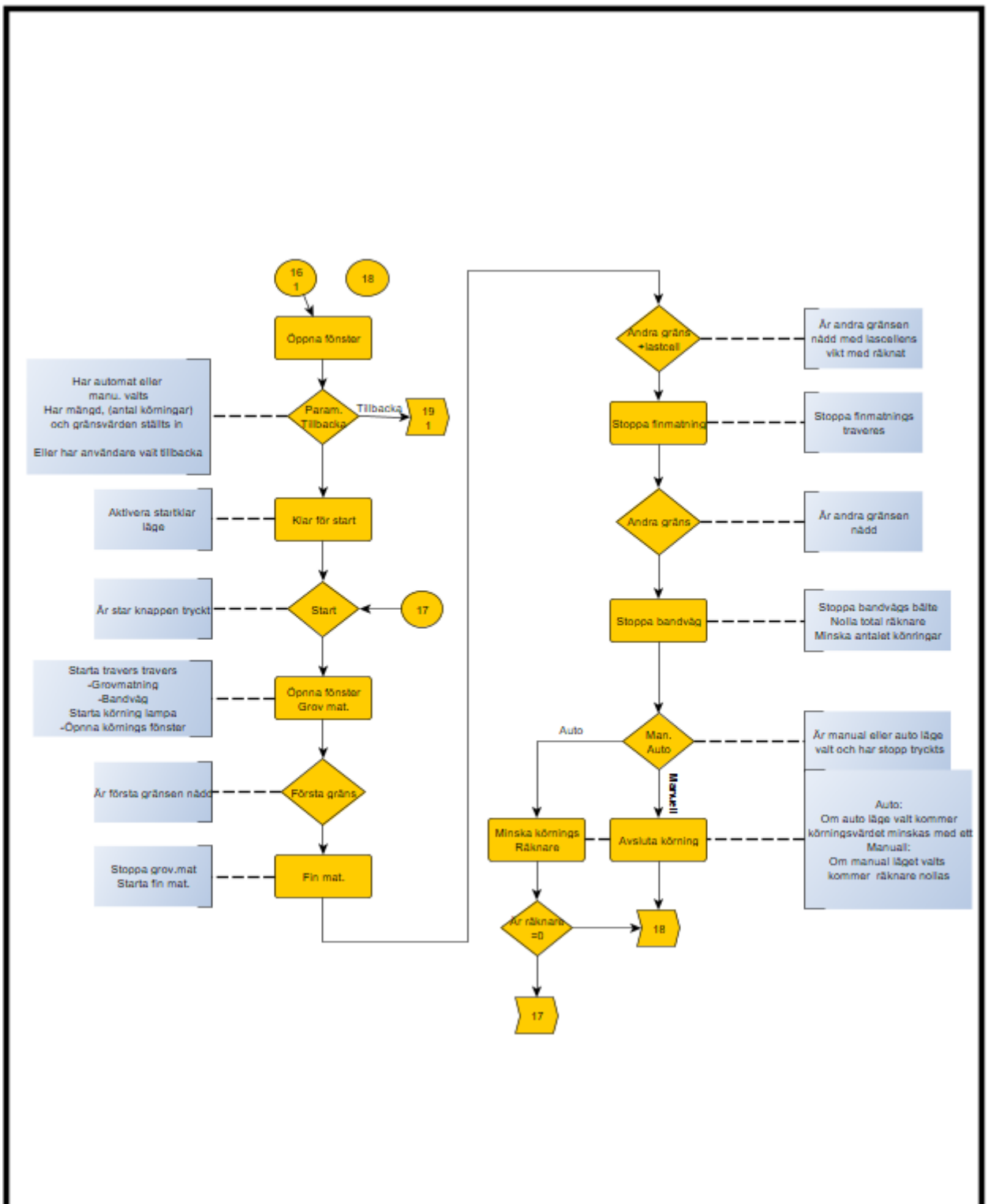
1.7. Sida 7 Materialtest



1.8. Sida 8 Uppvärmning



1.9. Sida 9 Process körning



2 Instruktioner för ibruktagning med SIWATOOL

Denna manual beskriver ibruktagning av bandvågsprocess. Manualen kommer beskriva hårdvarukrav, parameterinställning och olika kalibrerings möjligheter, samt dess ordningsföljd. För ibruktagning av SIWAREX 241 rekommenderas användning av SIWATOOL, därmed kommer denna manual beskriva ibruktagning genom användning av SIWATOOL. Kalibrering genom HMI:n kommer beskrivas under programpresentation

För att följa denna manual krävs en Siwarex WP241 modul, en lastcell, hastighets sensor(valfri), Siemens S7-1200 PLC (valfri) och en Beijer HMI (valfri).

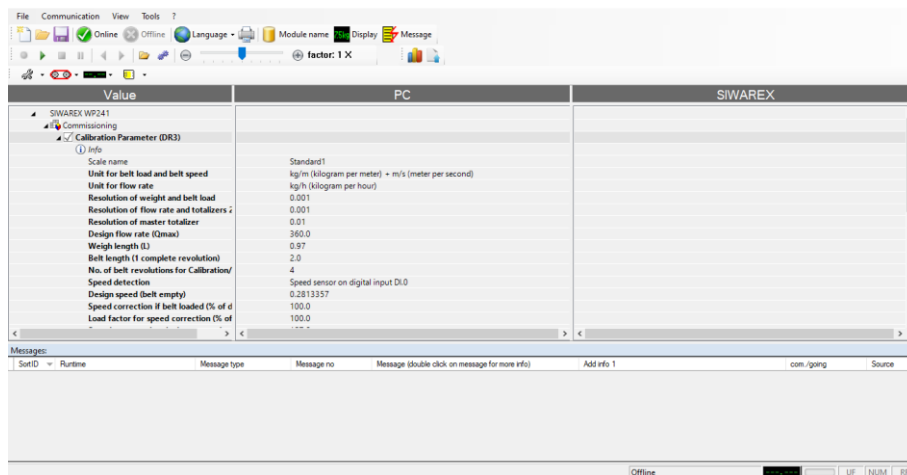
Om modulen används utan ihopkoppling med PLC bör DIP-switch:en lokaliserad på vänster sida av ethernet porten bytas från läge PLC läge till "stand alone" läge. DIP-switchen är per default i PLC-läge (övre läge).

2.1 Uppkoppling

Uppkoppling till modulen sker genom ethernet porten i SIWAREX modulen. Modulen har per default 192.168.0.21 som IP adress, denna adress måste ligga inom samma IP rymd som PC:n, för kommunikation (192.168.0). Om IP-adressen på modulen ska ändras kan man använda Siemens "primary setup tool" verktyget.

Med verktyget kan man söka genom ett nätverk och hitta Siemens apparatur. När apparaturen hittats kan man välja att ändra dess IP-adress.

När IP-adressen överstämmer kan SIWATOOL öppnas, här kan nu online-läge aktiveras (figur 2:1). Ihopkopplingen kan konstateras ner i skärmens högra hörn, (figur 2:1).



Figur 54: Siwatool

2.2 Parametrar

När online-läge är aktivt, hittas SIWAREX parametrarna i listan under med parameter i första kolumn, värde på datorn i andra och värde i SIWAREX-modulenn i tredje. Om värden på PC och SIWAREX inte stämmer med värdet på PC färgas ifrågavarande värde rött.

Värden kan antingen läsas från SIWAREX till PC (read) eller från PC till SIWAREX (write). Vid skrivande eller läsande av värden kan inte enskilda parametrar påverkas utan i detta fall skrivs/läses hela data record (DRx)

2.3 Ordningsföljd på kalibreringarna

Steg 1: Fabriksåterställning

Steg 2: Parametrar som måste matas in

Steg 3: Hastighets definiering

Steg 4: Nollpunkts kalibrering med tom bandvåg

Steg 5: Span kalibrering genom användning av testvikt eller testkedja

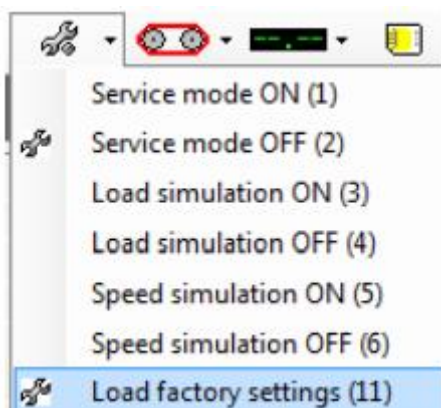
Steg 6: Materialtest för korrekt faktorering

Steg 7: Definition av gränsvärden

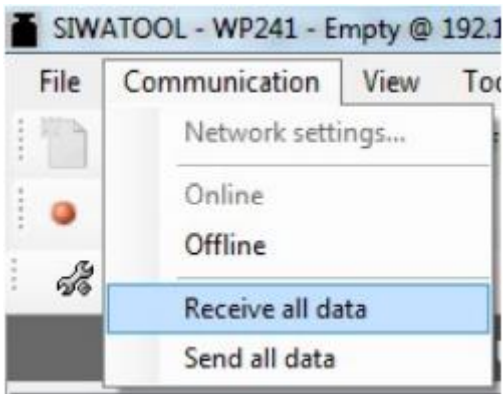
2.4 Fabriksåterställning

Fabriksåterställningen ställer alla SIWAREX parametrar tillbaka till tillverkarens parametrar, även IP-adressen. Fabriksinställningar krävs för att följa denna manual.

För att fabriksåterställa parametrarna ställs modulen i serviceläge, när serviceläge är aktivt indikeras detta nere i programmets högra hörn. När serviceläge är aktivt kan fabriksinställningar laddas in (Figur 2:2). När fabriksinställningarna är inladdade kan alla parametrar läsas in på PC genom "CommunicationàReceive all data" (Figur 2:3).



Figur 55: In läsning av fabriksinställningar

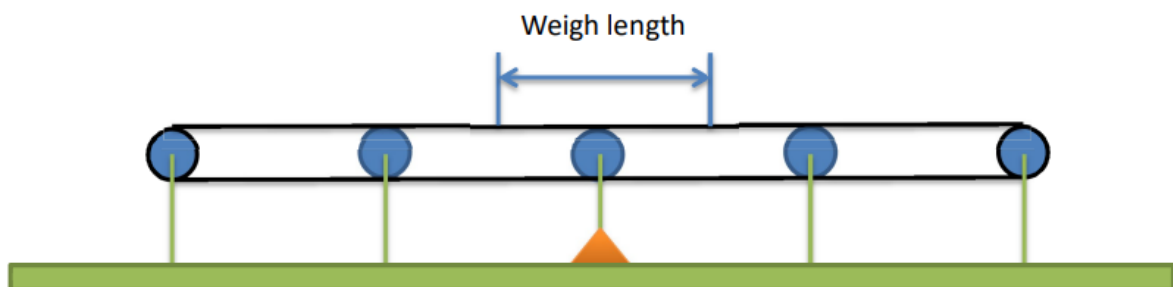


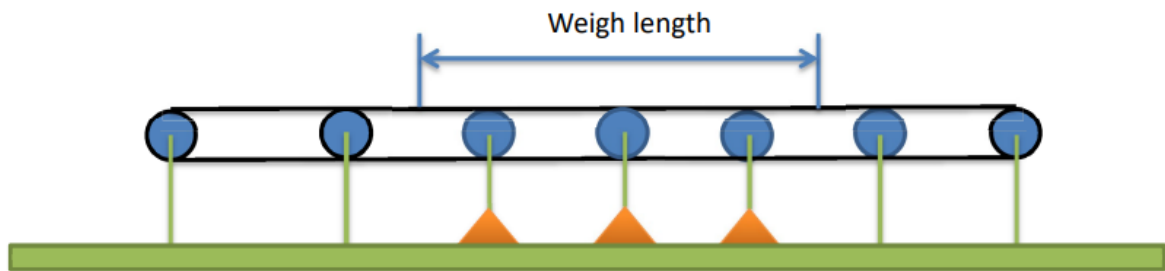
Figur 56: Ner laddning av data till användarfilen

2.5 Parametrar

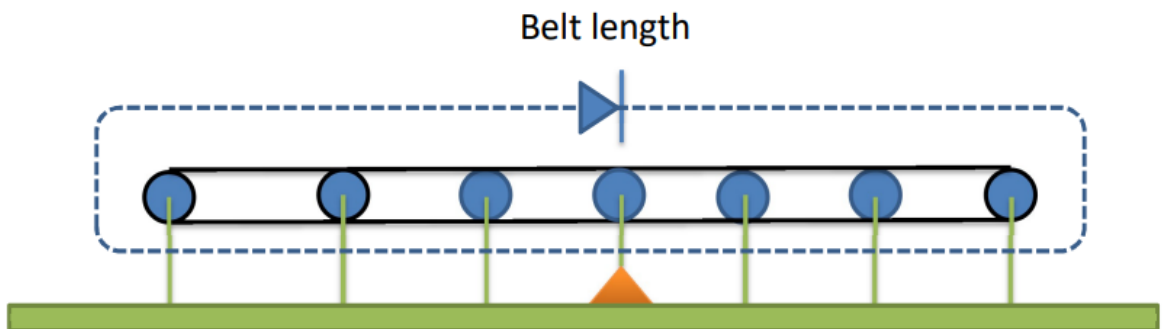
Alla viktiga parametrar för bandvågens drift finns lokaliserade i data record 3 (DR3). Parametrar som bör definieras före kalibrering är:

- Scale Name
 - Vågens namn med 12 karaktärer
- Unit for belt load
 - Lasten enhet, kg/m eller lbs/ft
- Unit for flow rate
 - Materialflödets enhet, t/h (ton per timmer), kg/h (kilogram per timmer), T/h (short ton ton per timmer), TL/h (long ton per timmer) lbs/ft (pound per timmer)
 - Bör noteras att brittiska och metriska måttenheter inte får blandas
- Resolution of weight and belt load
 - Noggrannhet för vikt och bältlast (0.1, 0.01 etc.)
- Resolution of flow rate and totalizer (S2-S6)
 - Noggrannhet för materialflöde och total räknare(0.1, 0.01 etc.)
- Resolution of master totalizer
 - Noggrannhet för huvud totalräknare (0.1, 0.01 etc.)
- Weigh length
 - Viktlängden är längden på det område som lastcellen verkar på, se figur xx.
 - Enheten beror på vald enhet för bältlast, kg/måmeter lbs/ftå fot





- Belt length
 - Bältets totala längd, se figur.



- No. of belt revolution for calibration
 - Antal varv som bältet ska snurra för kalibrering, större antal revolutioner ger bättre noggrannhet.
- Speed detection
 - Hastighetsavkänningen kan ske på tre olika sätt:
 - Med hastighetssensor kopplad på SIWAREX digitala ingång .0
 - Ingen sensor, hastigheten är konstant
 - Hastighetsinformation läses in via DR19

2.6 Hastighets kalibrering

Hastigheten kan kalibreras på ett flertal sätt beroende på vilken typ av hårdvara det är frågan om. Om hastigheten mäts med en givare kopplad direkt till WP241 modulen följs antingen kal.1 eller kal.2 beroende på om vi vet vilken pulskonstant som processen har (pulser per meter).

Om inte processen har någon hastighetsgivare utan vi kör processen med en konstant hastighet följs kal.3 eller kal.4 beroende på om vi vet vad den konstanta hastigheten är.

Om hastigheten ges från en extern källa, t.ex. en PLC följs kal.5

2.6.1 Kal 1. Med känd puls konstant

Parametrar i DR3 som bör fyllas i för denna kalibrering är följande:

- Speed detection
 - För denna kalibrering ska hastighetssensorn vara pulserande och vara inkopplad på SIWAREX modulens digitala ingång .0 (Speed sensor on digital input DI.0)
- Design speed(belt empty)
 - Under "Design speed" ska bältets maximala hastighet anges
- Speed sensor pulses(pulses per m/pulses per foot)
 - För de flesta bältmonterade givare anges denna variabel i givarens datablad men om en axel monterad givare används bör detta värde räknas ut genom ekvationen

$$\text{Pulser per meter} = \frac{\text{pulser per varv}}{(\pi \cdot d)}$$

där d är diametern på bältets styr skjul

När parametrarna är överförda till SIWAREX modulen kan bältet köras och hastigheten kontrolleras i DR30. Hastigheten kan dubbelgranskas genom användning av en tachometer eller alternativt räknas ut genom att dividera längden på bältet med tiden det tar för en revolution.

Fortsätt till nollpunkts kalibrering

2.6.2 Kal.2 Med okänd puls konstant

Steg 1: Markera bältet med t.ex. tejp för att kunna följa hur många varv som bältet snurrat. Välj även en fast punkt som markeringen passerar för att så noggrant som möjligt kunna bedöma antal varv

Steg 2: Starta bältet och låt snurra ett varv, när markeringen passerar fasta punkten aktivera hastighetsdetekteringen "Start speed-/puls detection (70)"

Steg 3: När bältet snurrat det definierade antal varv (No. of belt revolutions for calibration) och markeringen passerar sista gången fasta punkten, stoppa hastighetsdetekteringen "Stop speed-/pulse detection(71)"

När kalibreringen är klar syns de nya värden i DR4, här kan värden granskas före de godkänns och förs över till DR3. Om värden verkar rimliga godkänns de med "Accept pulse parameter (87)" och "Accept (nominal-) speed (85)". Om hastighets detekteringen inte körts på bältets max hastighet ska inte "Accpet(nominal-)speed (85)" köras utan i detta fall ska bältet köras på max hastighet och hastigheten iaktas från DR30 current speed och fyllas i DR3 Design belt speed.

Fortsätt till nollpunkts kalibrering

2.6.3 Kal.3 Användning av känd konstant hastighet

Om hastigheten inte mäts överhuvud taget och endast en konstant används för detta ska kalibreringen ske enligt följande metod.

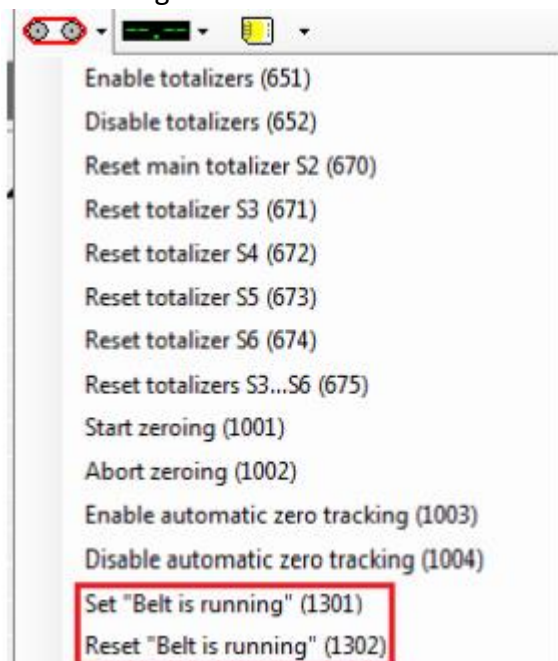
Steg 1: Fyll i DR3 parametrarna "Speed detection" som "No sensor-Preset/Detect constant speed" och "Design speed (belt empty)" som konstanta hastighet som bältet ska köras med.

Steg 2: Berätta för vågmodulen att bältet är i körning

Alternativ 1: Ta in en digital signal på 24V, detta görs genom att ange i DR7 "Assignment digital input DI.0" att vi kommer ta in en bältsignal "Belt signal (ON/OFF)". Detta betyder att när signalen på DI.0 är hög indikeras att bältet är i körning och när signalen är låg är bältet inte i körning.

Alternativ 2: Vi kan också berätta för modulen att bältet är i körningen genom att skicka via en PLC kommando koden 1301 för körning aktivering och 1302 inaktivering

(Alternativ 3:) Under testskede kan vi också ange att bältet är i körning via SIWATOOL, på samma sätt som alternativ 2, genom att aktivera och inaktivera körning med kommando kod 1301 och 1302



Fortsätt till nollpunkts kalibrering

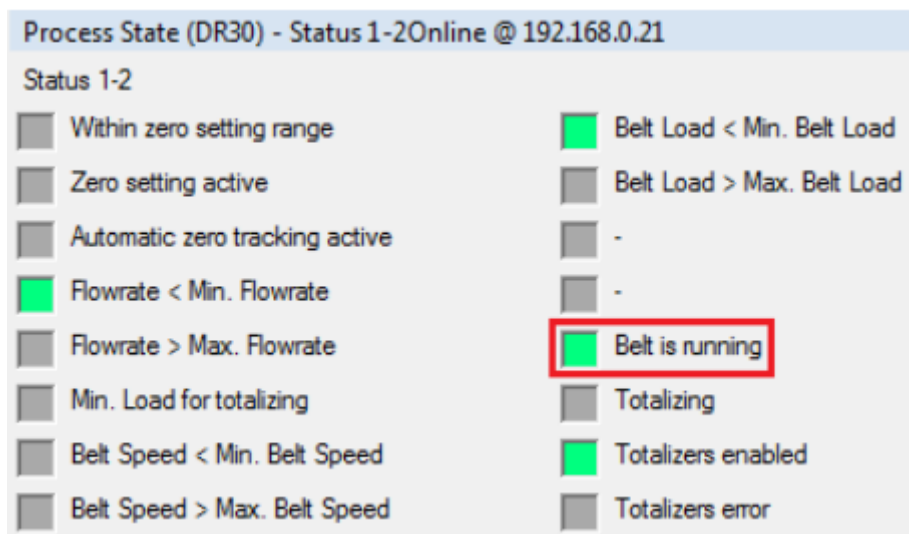
2.6.4 Kal.4 Användning av okänd konstant hastighet

Om hastigheten inte är känd och vi inte har någon givare kan en speciell procedur används för att lösa ut hastigheten:

Steg 1: Ställ DR3 "speed detection" till "No sensor – Preset/Detect constant speed"

Steg 2: Berätta för Våg modulen att bältet är i körning, samma alternativ som i tidigare kalibrering gäller

Steg 3: Starta bältet och granska så att modulen får körningskommandot, detta kan göras genom att gå in på DR30 "Status 1-2" och granska att "Belt is running" lyser grönt.



Steg 4: Markera en punkt på bältet och en fast punkt som bältetmarkeringen passerar.

Steg 5: Starta bältet och låt det rotera ett varv när bältmarkeringen når fasta markeringen, starta mätningen med kommandot 70 ("Start speed-/puls detection")

Steg 6: När bältet roterat det bestämda antalet varv (No. of revolutions for calibration) stoppa mätningen med kommandot 71 ("Stop speed-/puls detection")

När kalibreringen är klar, syns hastigheten i DR4 under "Speed (belt empty)", om värdet verkar lämpliga kan det överföras till DR3 med kommandot 85 "Accept(nominal-) speed"

Fortsätt till nollpunkts kalibrering

2.6.5 Kal.5 Hastighet från extern CPU

Om en färdig uträknad hastighet skickas till modulen från t.ex. en PLC behövs ingen kalibrering, utsett från att DR3 "speed detection" ställs till "External speed value from CPU (via DR19).

Här skickas hastigheten till DR19, alla värden över 0 används för kalkyleringar och om värdet är noll inaktiveras räknare.

2.7 Nollpunkts kalibrering med tom bandvåg

Om basparametrarna i DR3 är inmatade och hastighetsavkänningen fungerar kan noll punkts kalibreringen köras.

Steg 1: Ange DR10 "Switch 50/60Hz". Hit matas lokal nätverkets frekvens in för att förbättra signalfiltrering.

Steg 2: Granska så att bältet är uppvärmt och fungerar, ingen vikt befinner sig på bältet och att serviceläget är aktivt.

Steg 3: Starta bältet och aktivera noll läges kalibrering med kommandot 60 "Start initial zero calibration"

När bältet kört kalibreringsvarven (No. of revolution for calibraton) kommer kalibreringsparametrarna synnas i DR4 under "Inital zero calibration digits" och "Deviation from old initial zero calibration digits(%)". Om värden verkar möjliga kan dessa överföras till DR3 med kommandot 88 "Accept initial zero calibration digits". Efter att värden överförts borde nuvarande bältlast vara 0.0 kg/m

Fortsätt med Span kalibrering

2.8 Spankalibrering

Spankalibrering kan utföras på olika sätt här beskrivs följande tre:

- Genom användning av testvikt
- Genom användning av testkedja
- Genom användning av lastcellsdata

2.8.1 Genom användning av testvikt

Steg 1: Ange DR3 "Calibration weight (span by weight)" till den vikt som används. Enhet beror på vad som valts för "Unit for belt load"

Steg 2: Säkerställ att:

- 1: Nollpunktkalibrering körts

- 2: Bältet är uppvärmt och fungerar
- 3: Testvikter är monterade på vågen
- 4: Bältet är tomt
- 5: Serviceläge är aktivt

Steg 3: Starta bältet och kör spankalibrering med kommando 61 "Start span calibration with test weight"

När bältet körts förhands bestämda antalet varv (No. of belt revolution for calibration) kommer kalibreringsparametrarna finns under DR4 "Calibration weight (calculated)", "Span calibration digits" och Deviation from old span calibration digits in (%)". Om kalibrerings parametrarna verka lämpliga kan dessa överföras till DR3 med kommando 89 "Accept span calibration digits".

När kalibreringsparametrarna överförts till DR3 borde bält lasten visa kalibreringsvikten delta på vägningslängden. Nu kan bältet stoppas, testvikten tas bort och serviceläge aktiveras. I detta sked är det även rekommenderat att skapa en backupkalibrerings fil genom att gå på "Communication" à "Receive all data" à "File" à "Save as..."

Fortsätt med materialtest

2.8.2 Genom användning av testkedja

Steg 1: Ange DR3 "Calibration load (span by test chain)" till den vikt på kedjan. Enhet beror på vad som valts för "Unit for belt load"

Steg 2: Säkerställ att:

- 1: Nollpunktskalibrering är körd
- 2: Bältet är uppvärmt och fungerar
- 3: Testkedjan är monterad och centrerad
- 4: Serviceläge är aktivt

Steg 3: Starta bältet och kör spankalibrering med kommando 65 "Start span calibration with test chain"

När bältet kört de på förhand bestämda antal varv (No. of belt revolution for calibration) kommer kalibreringsparametrarna finns under DR4 "Calibration weight (calculated)", "Span calibration digits" och Deviation from old span calibration digits in (%)". Om kalibreringsparametrarna verka lämpliga kan dessa överföras till DR3 med kommando 89 "Accept span calibration digits".

När kalibreringsparametrarna överförts till DR3 borde bältlasten vara samma som hela kedjans vikt. Nu kan bältet stoppas, testkedjan tas bort och serviceläge inaktiveras. I detta sked är det även rekommenderat att skapa en backupkalibrerings fil genom att gå på "Communication" à "Receive all data" à "File" à "Save as..."

Fortsätt med materialtest

2.8.3 Användning av lastcellsdata

Steg 1: I DR10 ange följande parametrar

- 1: "No. of load cells" antal last celler
- 2: "Averaged characteristic value (mV/V)" Detta värde hittas oftast på själva lastcellen eller i dess datablad om inte ett exakt värde hittas kan ett ungefärligt värde anges. Om två eller flera lastceller används måste ett medeltal för alla lastceller räknas ut.
- 3: "Nominal load of one single load cell" enheten beror på "Unit for belt load"

Steg 2: Säkerställ att:

- 1: Nolllägeskalibrering körts
- 2: Bältet är tomt och fungerar
- 3: Ingen testvikt eller testkedja är monterad
- 4: Serviceläget är aktivt

Steg 3: Starta bältet och kör kommandot 63 "Start automatic span calibration".

När kalibreringen är klar överförs värden automatiskt till DR3. Det är rekommenderat att köra material testet genast efter denna kalibrering för att förbättra processens noggrannhet.

Fortsätt med materialtest.

2.9 Materialtest

Materialtestet kan köras med antingen förhands eller efterhands vägt material. Notera att material testet ska köras med det material och flöde som processen är avsedd för.

Om bältlasten förväntas ha stora förändringar under körning kan det vara skäl kalibrera två korrektionsfaktorer. Vid kalibrering av två korrektions faktorer ska materialtestet köras två gånger med olika bältlast (bält lasten kan övervakas genom att ändra displayn till 740 "Displat belt load (%)", t.ex. kunde första testet köras med ungefär 25% av nominella lasten och andra med ungefär 75% av nominella lasten. När två korrektionsfaktorer angetts kommer bältvågsmodulen automatiskt byta mellan dessa beroende på bältlasten.

2.9.1 Materialtest

Steg 1: Säkerställ att bältet är tomt och starta bältet

Steg 2 (valfritt): Stoppa total räknare S1-S5 genom kommandot 652 "Disable totalizers"

Steg 3: Nollställ totalräknare 6 med kommandot 674 "Reset totalizer S6"

Steg 4: Mata konstant materialet in på bältet och kör det igenom processen.

Steg 5: När allt material körts genom granska värdet på totalräknare S6 i DR33 "Totalizer S6"

Steg 5.5: Om materialet som körts genom processen inte vägts tidigare bör detta göras nu.

Steg 6: Med värdet för totalräknare 6 och den genomkörda vikten kan korrektionsfaktorn räknas ut (formel xx)

Korrektions faktor

$$= (\text{gamla korrektions faktor}) \cdot \frac{\text{genom körda materialtes vikt}}{\text{Total räknare 6 värde}}$$

Gamla korrektionsfaktorns värde kan iaktas i DR5 "Correction factor 1"

Steg 7: Den nya korrektionsfaktorn matas in i DR5 "Correction factor 1"

Steg 7.5: Om två korrektionsfaktorer ska matas in bör både den uträknade korrektionsfaktorn och den procentuella bältlasten matas in, se figur.

Correction Factors (DR5)		
Info		
Belt load factor 1 (% of nominal belt load)	25.0	25.0
Correction factor 1	0.956	0.956
Belt load factor 2 (% of nominal belt load)	75.0	75.0
Correction factor 2	0.989	0.989

2.10 Definition av gränsvärden

Till sista ska gränsvärden matas in, dessa finns i DR6 och är relativt självbeskrivande. Vid osäkerhet använd SIWAREX modulens manual där alla parametrar finns beskrivna.

Denna manual är en förkortad och översatt version av Siemens FAQs dokument angående Kalibrering av SIWAREX WP241 med SIWATOOL V7

3 Datavariabler

3.1 Signaler till HMI

BeltWeigher_V100_modulerat / PLC_1 [CPU 1212C DC/DC/Rly] / Program blocks / HMI_DataBlock

ToPanel [DB20]

ToPanel Properties										
General										
Name	ToPanel	Number	20	Type	DB	Language	DB			
Numbering	Manual									
Information										
Title		Author		Comment		Family				
Version	0.1	User-defined ID								
ToPanel										
Name	Data type	Offset	Start value	Retain	Accessi-ble from HMI/OPC UA/Web API	Writ-able from HMI/OPC UA/ Web API	Visible in HMI engi-neering	Setpoint	Supervi-sion	Comment
▼ Static										
P_KorningOK_PLC	Bool	0.0	false	True	True	True	True	False		
P_Kallibrering_PLC	Bool	0.1	false	True	True	True	True	False		
P_Uppvarmning_Aktiv	Bool	0.2	false	True	True	True	True	False		
P_Uppvarmning_klar	Bool	0.3	false	True	True	True	True	False		
P_A_Aktiv	Bool	0.4	false	True	True	True	True	False		
▼ BVx_A_Index	Array[0..7] of Bool	2.0		True	True	True	True	False		
BVx_A_Index[0]	Bool	2.0	false	True	True	True	True	False		
BVx_A_Index[1]	Bool	2.1	false	True	True	True	True	False		
BVx_A_Index[2]	Bool	2.2	false	True	True	True	True	False		
BVx_A_Index[3]	Bool	2.3	false	True	True	True	True	False		
BVx_A_Index[4]	Bool	2.4	false	True	True	True	True	False		
BVx_A_Index[5]	Bool	2.5	false	True	True	True	True	False		
BVx_A_Index[6]	Bool	2.6	false	True	True	True	True	False		
BVx_A_Index[7]	Bool	2.7	false	True	True	True	True	False		
BVx_TOM	Bool	4.0	false	True	True	True	True	False		
P_DisplayID_val	Int	6.0	0	True	True	True	True	False		
BVx_Aterstaendekorningar	Int	8.0	0	True	True	True	True	False		
BVx_Kordvikt_P	Int	10.0	0	True	True	True	True	False		
P_Arvarde_uppvarmningstid	DInt	12.0	0	True	True	True	True	False		
P_Utraknad_Korrektions_faktor	Real	16.0	0.0	True	True	True	True	False		

3.2 Signaler från HMI

BeltWeigher_V100_modulerat / PLC_1 [CPU 1212C DC/DC/Rly] / Program blocks / HMI_DataBlock

FromPanel [DB10]

FromPanel Properties

General

Name	FromPanel	Number	10	Type	DB	Language	DB
Numbering	Manual						

Information

Title		Author		Comment		Family	
Version	0.1	User-defined ID					

FromPanel

Name	Data type	Offset	Start value	Retain	Accessi- ble from HMI/OPC UA/Web API	Writ- able from HMI/ OPC UA/ Web API	Visible in HMI engi- neering	Setpoint	Supervi- sion	Comment
▼ Static										
P_KorOK_HMI	Bool	0.0	false	True	True	True	True	False		
P_KorPause_HMI	Bool	0.1	false	True	True	True	True	False		
P_KorAvbryt_HMI	Bool	0.2	false	True	True	True	True	False		
BVx_KorManuell_HMI	Bool	0.3	false	True	True	True	True	False		
P_KorManuell_M1_HMI	Bool	0.4	false	True	True	True	True	False		
BVx_KorKalibreeringsvarv	Bool	0.5	false	True	True	True	True	False		
P_Uppvärmning_HMI	Bool	0.6	false	True	True	True	True	False		
BVx_TomBV_Kvitering	Bool	0.7	false	True	True	True	True	False		
P_Kalibrering_HMI	Bool	1.0	false	True	True	True	True	False		
BVx_RaknaKF	Bool	1.1	false	True	True	True	True	False		
P_Alarm_Ack	Bool	1.2	false	True	True	True	True	False		
P_Alarm_Ack_CMD	Bool	1.3	false	True	True	True	True	False		
P_Uppvärmning_Avbryt	Bool	1.4	false	True	True	True	True	False		
P_Sprak_index	Int	2.0	0	True	True	True	True	False		
P_DisplayID_aktuell	Int	4.0	0	True	True	True	True	False		
P_Borvarde_Uppvarmning- stid	Int	6.0	30	True	True	True	True	False		
BVx_Borvarde_antalkornin- gar	Int	8.0	1	True	True	True	True	False		
BVx_Borvarde_Grans1	Real	10.0	0.0	True	True	True	True	False		
BVx_Borvarde_Grans2	Real	14.0	0.0	True	True	True	True	False		
BVx_kordvikt_Korrekt- ion_faktor	Real	18.0	0.0	True	True	True	True	False		

3.3 PLC tag-lista

Name	Data Type	Logical Address
In_Start	Bool	%I0.0
In_Stopp	Bool	%I0.1
In_NodStopp	Bool	%I0.2
In_Kalibrering	Bool	%I0.3
In_DriftMan	Bool	%I0.4
In_DriftAuto	Bool	%I0.5
M_P_KörningOK	Bool	%M1.0
M_Pulsmatning_körning	Bool	%M1.1
M_BVx_ParamOK	Bool	%M1.2
M_BVx_Räknare=0	Bool	%M1.3
M_P_AutoKörKlar	Bool	%M1.4
M_P_ManKörKlar	Bool	%M1.5
M_BVx_Viktuppnad	Bool	%M1.6
M_BVx_LifeBit	Bool	%M1.7
M_P_Pulsmatning_Pulssignal	Bool	%M2.0
M_BVx_bält_körning	Bool	%M2.1
M_P_Grovmatare_körning	Bool	%M2.2
M_P_Finmatning_körning	Bool	%M2.3
M_DataLäsning_PulsSignal2	Bool	%M2.4
M_Grovmatare_Uppvarmning	Bool	%M2.5
M_BVmatare_oppvarmning	Bool	%M2.6
M_BVx_Man.Körning	Bool	%M2.7
M_Grovmatare_Man.Körning	Bool	%M3.0
M_Kalibrering_PulsSignal3	Bool	%M3.2
Z_Weigher	Bool	%M3.3
Z_Counter	Bool	%M3.4
M_BVx_Last+Körd	Real	%MD14
M_P_Borvarde_UVtid	Time	%MD18
M_P_Arvarde_UVtid	Time	%MD22
M_BVx_Cyklisk_index	Int	%MW6
M_BVx_RäknareAntalKörning	Int	%MW8
Ut_Grov_Mat.	Bool	%Q0.0
Ut_Fin_Mat.	Bool	%Q0.1
Ut_Startklar	Bool	%Q0.2
Ut_Köring	Bool	%Q0.3
Ut_Fel	Bool	%Q0.4
Ut_BVx_Mat.	Bool	%Q0.5

3.4 WP241 Datablock

▼ DR30	Struct	1304.0		False	True	True	True	False		DR30 Process Values 1
DR_NO	UInt	1304.0	30	False	True	True	True	False		Data record number
DR_LEN	UInt	1306.0	68	False	True	True	True	False		Length
APPL_ID	UInt	1308.0	104	False	True	True	True	False		Application
DR_VERSION	UInt	1310.0	1	False	True	True	True	False		Version of data record
SCALE_STATUS_1	UInt	1312.0	0	False	True	True	True	False		Scale status word 1
SCALE_STATUS_2	UInt	1314.0	0	False	True	True	True	False		Scale status word 2
OPERATION_ERRORS	UInt	1316.0	0	False	True	True	True	False		Operation error word
TECHNOLOGICAL_ERRORS	UInt	1318.0	0	False	True	True	True	False		Technological error word
WEIGHT	Real	1320.0	0.0	False	True	True	True	False		Current weight
BELT_LOAD	Real	1324.0	0.0	False	True	True	True	False		Current Belt Load
BELT_LOAD_PC	Real	1328.0	0.0	False	True	True	True	False		Current Belt Load (%)
FLOW_RATE	Real	1332.0	0.0	False	True	True	True	False		Current Flow Rate
FLOW_RATE_PC	Real	1336.0	0.0	False	True	True	True	False		Current Flow Rate (%)
SPEED	Real	1340.0	0.0	False	True	True	True	False		Current Belt Speed
SPEED_PC	Real	1344.0	0.0	False	True	True	True	False		Current Belt Speed (%)
MASTER_TOTALIZER	LReal	1348.0	0.0	False	True	True	True	False		Master Totalizer
MAIN_TOTALIZER	Real	1356.0	0.0	False	True	True	True	False		Main Totalizer
RESERVED_56	Real	1360.0	0.0	False	True	True	True	False		Reserved
REFRESH_COUNTER	UInt	1364.0	0	False	True	True	True	False		Weight Refresh Counter
RESERVED_62	UInt	1366.0	0	False	True	True	True	False		Reserved
RESERVED_64	Real	1368.0	0.0	False	True	True	True	False		Reserved
▼ DR31	Struct	1372.0		False	True	True	True	False		DR31 Process state extended 1
DR_NO	UInt	1372.0	31	False	True	True	True	False		Data record number
DR_LEN	UInt	1374.0	54	False	True	True	True	False		Length
APPL_ID	UInt	1376.0	104	False	True	True	True	False		Application
DR_VERSION	UInt	1378.0	1	False	True	True	True	False		Version of data record
DIGITS_UNFILTERED	DInt	1380.0	0	False	True	True	True	False		Unfiltered digits
DIGITS_FILTERED	DInt	1384.0	0	False	True	True	True	False		Digits filtered
CURRENT_LOADCELL_SIGNAL_MV	Real	1388.0	0.0	False	True	True	True	False		Current loadcell input signal (mV)
CURRENT_ANA_OUT	Real	1392.0	0.0	False	True	True	True	False		Current analog output (mA)
PULSES_PER_BELT_REVOLUTION	DInt	1396.0	0	False	True	True	True	False		Pulses per complete belt revolution
RESERVED_24	Real	1400.0	0.0	False	True	True	True	False		Reserved
PULSES_AT_NOMINAL_SPEED	Real	1404.0	0.0	False	True	True	True	False		Pulses/s at nominal speed
PULSES_PER_SECOND	UInt	1408.0	0	False	True	True	True	False		Pulses/s
RESERVED_38	UInt	1410.0	0	False	True	True	True	False		Reserved
RESERVED_40	UInt	1412.0	0	False	True	True	True	False		Reserved
RESERVED_42	UInt	1414.0	0	False	True	True	True	False		Reserved
CURRENT_STATE_DQ_0	Bool	1416.0	0	False	True	True	True	False		Current status DQ.0
CURRENT_STATE_DQ_1	Bool	1416.1	0	False	True	True	True	False		Current status DQ.1
CURRENT_STATE_DQ_2	Bool	1416.2	0	False	True	True	True	False		Current status DQ.2
CURRENT_STATE_DQ_3	Bool	1416.3	0	False	True	True	True	False		Current status DQ.3
RESERVED_44_4	Bool	1416.4	0	False	True	True	True	False		Reserved
RESERVED_44_5	Bool	1416.5	0	False	True	True	True	False		Reserved
RESERVED_44_6	Bool	1416.6	0	False	True	True	True	False		Reserved
RESERVED_44_7	Bool	1416.7	0	False	True	True	True	False		Reserved
CURRENT_STATE_DI_0	Bool	1417.0	0	False	True	True	True	False		Current status DI.0
CURRENT_STATE_DI_1	Bool	1417.1	0	False	True	True	True	False		Current status DI.1
CURRENT_STATE_DI_2	Bool	1417.2	0	False	True	True	True	False		Current status DI.2
CURRENT_STATE_DI_3	Bool	1417.3	0	False	True	True	True	False		Current status DI.3
RESERVED_45_4	Bool	1417.4	0	False	True	True	True	False		Reserved
RESERVED_45_5	Bool	1417.5	0	False	True	True	True	False		Reserved
DIP_SW_1	Bool	1417.6	0	False	True	True	True	False		DIP switch 1
DIP_SW_2	Bool	1417.7	0	False	True	True	True	False		DIP switch 2
NOMINAL_BELT_LOAD	Real	1418.0	0.0	False	True	True	True	False		Nominal belt load
REFRESH_COUNTER	UInt	1422.0	0	False	True	True	True	False		Refresh counter
RESERVED_52	UInt	1424.0	0	False	True	True	True	False		Reserved
▼ DR33	Struct	1454.0		False	True	True	True	False		DR33 Totalizers
DR_NO	UInt	1454.0	33	False	True	True	True	False		Data record number
DR_LEN	UInt	1456.0	44	False	True	True	True	False		Length
APPL_ID	UInt	1458.0	104	False	True	True	True	False		Application
DR_VERSION	UInt	1460.0	1	False	True	True	True	False		Version of data record
MASTER_TOTALIZER_S1	LReal	1462.0	0.0	False	True	True	True	False		Current master totalizer
MAIN_TOTALIZER_S2	Real	1470.0	0.0	False	True	True	True	False		Current main totalizer
RESERVED_20	Real	1474.0	0.0	False	True	True	True	False		Reserved
TOTALIZER_S3	Real	1478.0	0.0	False	True	True	True	False		Totalizer 3
TOTALIZER_S4	Real	1482.0	0.0	False	True	True	True	False		Totalizer 4
TOTALIZER_S5	Real	1486.0	0.0	False	True	True	True	False		Totalizer 5
TOTALIZER_S6	Real	1490.0	0.0	False	True	True	True	False		Totalizer 6
RESERVED_40	Real	1494.0	0.0	False	True	True	True	False		Reserved

▼ s_CMD1	Struct	468.0		False	True	True	True	False	Command with highest priority
i_CMD_CODE	Int	468.0	0	False	True	True	True	False	
bo_CMD_TRIGGER	Bool	470.0	false	False	True	True	True	False	
bo_CMD_InProgress	Bool	470.1	false	False	True	True	True	False	
bo_CMD_FinishedOK	Bool	470.2	false	False	True	True	True	False	
bo_CMD_FinishedError	Bool	470.3	false	False	True	True	True	False	
▼ s_CMD2	Struct	472.0		False	True	True	True	False	Command with medium priority
i_CMD_CODE	Int	472.0	0	False	True	True	True	False	
bo_CMD_TRIGGER	Bool	474.0	false	False	True	True	True	False	
bo_CMD_InProgress	Bool	474.1	false	False	True	True	True	False	
bo_CMD_FinishedOK	Bool	474.2	false	False	True	True	True	False	
bo_CMD_FinishedError	Bool	474.3	false	False	True	True	True	False	
▼ s_CMD3	Struct	476.0		False	True	True	True	False	Command with lowest priority
i_CMD_CODE	Int	476.0	0	False	True	True	True	False	
bo_CMD_TRIGGER	Bool	478.0	false	False	True	True	True	False	
bo_CMD_InProgress	Bool	478.1	false	False	True	True	True	False	
bo_CMD_FinishedOK	Bool	478.2	false	False	True	True	True	False	
bo_CMD_FinishedError	Bool	478.3	false	False	True	True	True	False	
▼ s_CMD_curr	Struct	480.0		False	True	True	True	False	
i_CMD_CODE	Int	480.0	0	False	True	True	True	False	
bo_CMD_TRIGGER	Bool	482.0	false	False	True	True	True	False	
bo_CMD_InProgress	Bool	482.1	false	False	True	True	True	False	
bo_CMD_FinishedOK	Bool	482.2	false	False	True	True	True	False	
bo_CMD_FinishedError	Bool	482.3	false	False	True	True	True	False	
bo_CMD_ERR	Bool	484.0	false	False	True	True	True	False	
▼ s_FB_STATUS	Struct	486.0		False	True	True	True	False	Status of Function Block
bo_AppIDError	Bool	486.0	false	False	True	True	True	False	
bo_AppIDDERror	Bool	486.1	false	False	True	True	True	False	
bo_SFBErro	Bool	486.2	false	False	True	True	True	False	
bo_RdPerError	Bool	486.3	false	False	True	True	True	False	
bo_LifeBitError	Bool	486.4	false	False	True	True	True	False	
bo_StartUpError	Bool	486.5	false	False	True	True	True	False	
bo_WrongFW	Bool	486.6	false	False	True	True	True	False	
bo_InvalidCMD	Bool	486.7	false	False	True	True	True	False	
bo_DataOperationError	Bool	487.0	false	False	True	True	True	False	
bo_StartUp	Bool	487.1	false	False	True	True	True	False	
w_ErrorCode	Word	488.0	16#0	False	True	True	True	False	
▼ s_IO_DATA	Struct	490.0		False	True	True	True	False	I/O scale data
COORDINATION	Byte	490.0	16#0	False	True	True	True	False	
APPL_ID_ACTUAL	Byte	491.0	104	False	True	True	True	False	
ERROR_CODE	UInt	492.0	0	False	True	True	True	False	
▼ SCALE_STATUS_1	Struct	494.0		False	True	True	True	False	Scale Status Word 1
MIN_LOAD	Bool	494.0	false	False	True	True	True	False	Belt Load < Minimum Belt Load (see DR6)
MAX_LOAD	Bool	494.1	false	False	True	True	True	False	Belt Load > Maximum Belt Load (see DR6)
RESERVED_2	Bool	494.2	false	False	True	True	True	False	Reserved
RESERVED_3	Bool	494.3	false	False	True	True	True	False	Reserved
BELT_STATUS	Bool	494.4	false	False	True	True	True	False	TRUE=running, FALSE=stopped
TOTALIZERS_ACTIVE	Bool	494.5	false	False	True	True	True	False	Totalizers are active
TOTALIZERS_ENABLED	Bool	494.6	false	False	True	True	True	False	Totalizers are enabled
TOTALIZERS_ERROR	Bool	494.7	false	False	True	True	True	False	Totalizers are disturbed
ZERO_RANGE	Bool	495.0	false	False	True	True	True	False	Within zero setting range (see DR6)
ZERO_SETTING	Bool	495.1	false	False	True	True	True	False	Zero setting is performed
ZERO_TRACKING	Bool	495.2	false	False	True	True	True	False	Zero tracking is active

MIN_RATE	Bool	495.3	false		False	True	True	True	False	Flow Rate < Minimum Flowrate (see DR6)
MAX_RATE	Bool	495.4	false		False	True	True	True	False	Flow Rate > Maximum Flowrate (see DR6)
MIN_LOAD_FOR_TOTALIZERS	Bool	495.5	false		False	True	True	True	False	Belt load > Minimum Belt Load for Totalizing (see DR6)
MIN_SPEED	Bool	495.6	false		False	True	True	True	False	Belt Speed < Minimum Belt Speed (see DR6)
MAX_SPEED	Bool	495.7	false		False	True	True	True	False	Belt Speed > Maximum Belt Speed (see DR6)
▼ SCALE_STATUS_2	Struct	496.0			False	True	True	True	False	Scale Status Word 2
CALIBRATED	Bool	496.0	false		False	True	True	True	False	Zero and Span digits are plausible
SERVICE_MODE	Bool	496.1	false		False	True	True	True	False	Service Mode is active
CALIBRATION	Bool	496.2	false		False	True	True	True	False	Calibration command is performed
WRITE_PROTECTION	Bool	496.3	false		False	True	True	True	False	Write Protection Jumper is set
ANALOGUE_OUT_ERROR	Bool	496.4	false		False	True	True	True	False	Analogue Output is disturbed
WARM_UP	Bool	496.5	false		False	True	True	True	False	Warm up timer is running
START_UP	Bool	496.6	false		False	True	True	True	False	Module is starting
ERROR	Bool	496.7	false		False	True	True	True	False	Module is disturbed
REMOTE_TOTALIZER	Bool	497.0	false		False	True	True	True	False	Pulse Signal for Remote Totalizer (see DR7)
RESERVED_9	Bool	497.1	false		False	True	True	True	False	Reserved
LOAD_SIMULATION	Bool	497.2	false		False	True	True	True	False	Belt Load Simulation is active
SPEED_SIMULATION	Bool	497.3	false		False	True	True	True	False	Belt Speed Simulation is active
STOP_WATCH	Bool	497.4	false		False	True	True	True	False	Stop watch is running
TIME_BUFFER_ERROR	Bool	497.5	false		False	True	True	True	False	Internal Time Buffer Failed
TRACE	Bool	497.6	false		False	True	True	True	False	Trace recorder is running
DIGITAL_INPUT_ERROR	Bool	497.7	false		False	True	True	True	False	Error On Digital Inputs
FLOW_RATE	Real	498.0	0.0		False	True	True	True	False	Current Flow Rate
MAIN_TOTALIZER_S2	Real	502.0	0.0		False	True	True	True	False	Main Totalizer S2
dw_RESERVED_1	DInt	506.0	0		False	True	True	True	False	Reserved
dw_RESERVED_2	DInt	510.0	0		False	True	True	True	False	Reserved
OPERATION_ERRORS	UInt	514.0	0		False	True	True	True	False	Operation Errors
TECHNOLOGICAL_ERRORS	UInt	516.0	0		False	True	True	True	False	Technological Errors
DATA_CMD_ERROR_1	UInt	518.0	0		False	True	True	True	False	Data-Operation Error Word 1
DATA_CMD_ERROR_2	UInt	520.0	0		False	True	True	True	False	Data-Operation Error Word 2
DATA_CMD_ERROR_3	UInt	522.0	0		False	True	True	True	False	Data-Operation Error Word 3
DATA_CMD_ERROR_4	UInt	524.0	0		False	True	True	True	False	Data-Operation Error Word 4
ANA_OUTPUT	Real	526.0	0.0		False	True	True	True	False	Value for Analog output
DIGIT_OUTPUT	Word	530.0	16#0		False	True	True	True	False	Values for digital outputs
CONTROL	Word	532.0	16#0		False	True	True	True	False	
r_RESERVED_3	Real	534.0	0.0		False	True	True	True	False	Reserved
ui_APPL_ID	UInt	538.0	104		False	True	True	True	False	

3.5 HMI Tag-lista

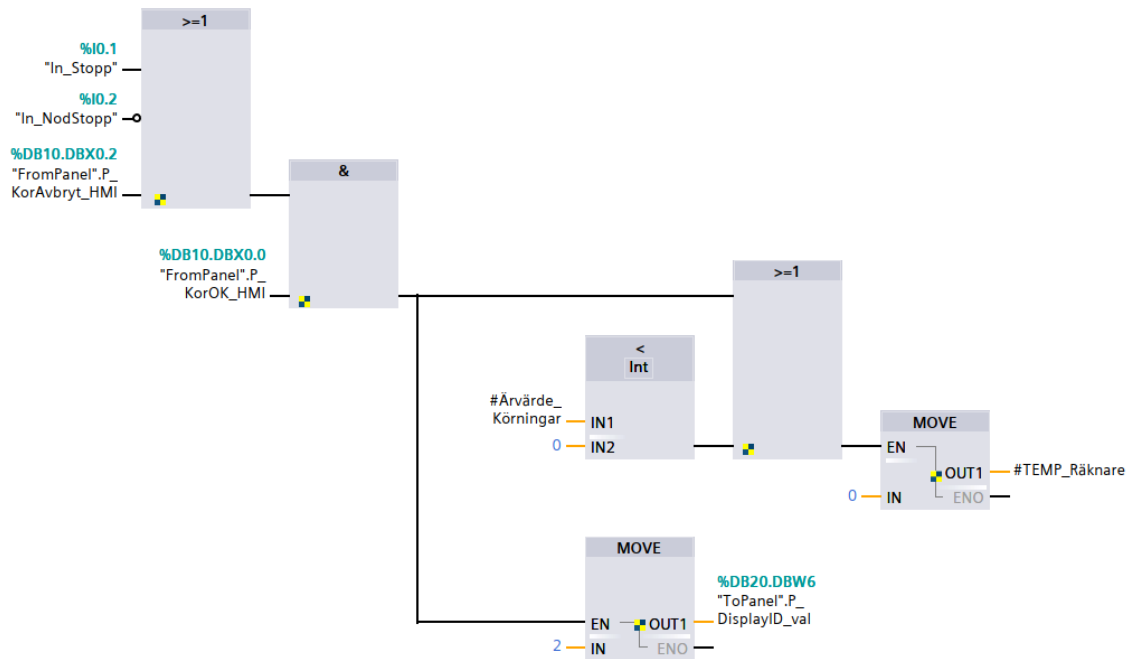
Name	DataType	Address_1
Borvarde_Grans1	FLOAT	DB10.DBD10
Borvarde_Grans2	FLOAT	DB10.DBD14
Kordvikt_KF	FLOAT	DB10.DBD22
Sprak	INT16	DB10.DBW2
Borvarde_UppvarmningsTid	INT16	DB10.DBW6
Borvarde_antalkorningar	INT16	DB10.DBW8
KorningOK_HMI	BOOL	DB10.DBX0.0
KorningPause_HMI	BOOL	DB10.DBX0.1
KorningAvbryt_HMI	BOOL	DB10.DBX0.2
Korning_ManuellBV	BOOL	DB10.DBX0.3
Korning_ManuellM1	BOOL	DB10.DBX0.4
Korning_Kalibrering	BOOL	DB10.DBX0.5
Uppvarmning_HMI	BIT	DB10.DBX0.6
Uppvarmning_TomBV_Kvitering	BOOL	DB10.DBX0.7
Kalibrering_HMI	BIT	DB10.DBX1.0
Kalibrering_RaknaKF	BOOL	DB10.DBX1.1
A_Ack	BIT	DB10.DBX1.2
A_Ack_CMD	BOOL	DB10.DBX1.3
Uppvarmning_Avbryt	BOOL	DB10.DBX1.4
Arvarde_UppvarmningsTid	INT32	DB20.DBD12
Utraknad_korrektions_faktor	FLOAT	DB20.DBD16
Arvarde_kordvikt_p	INT16	DB20.DBW10
Arvarde_antalkorningar	INT16	DB20.DBW8
KorningOK_PLC	BOOL	DB20.DBX0.0
Kalibrering_PLC	BOOL	DB20.DBX0.1
Uppvarmning_aktiv	BOOL	DB20.DBX0.2
Uppvarmning_Klar	BOOL	DB20.DBX0.3
A_Aktiv	BOOL	DB20.DBX0.4
Alarm0	BOOL	DB20.DBX2.0
Alarm1	BOOL	DB20.DBX2.1
Alarm2	BOOL	DB20.DBX2.2
Alarm3	BOOL	DB20.DBX2.3
Alarm4	BOOL	DB20.DBX2.4
Alarm5	BOOL	DB20.DBX2.5
Alarm6	BOOL	DB20.DBX2.6
Alarm7	BOOL	DB20.DBX2.7
BV_Tomm	BOOL	DB20.DBX4.0
DR10_GAIN_LOAD_CELL	FLOAT	DB241.DBD1044
DR10_NOMINAL_LOAD_ONE_LOAD_CELL	FLOAT	DB241.DBD1048
Arvarde_vikt	FLOAT	DB241.DBD1320
DR30_BELT_LOAD_PC	FLOAT	DB241.DBD1328
Arvarde_massaflode	FLOAT	DB241.DBD1332
Arvarde_hastighet	FLOAT	DB241.DBD1340
DR30_SPEED	FLOAT	DB241.DBD1340
DR30_SPEED_PC	FLOAT	DB241.DBD1344
Arvarde_totalkordvikt1	FLOAT	DB241.DBD1462
Arvarde_totalkordvikt2	FLOAT	DB241.DBD1470
Arvarde_kordvikt	FLOAT	DB241.DBD1478
DR33_TOTALIZER_S6	FLOAT	DB241.DBD1490

DR3_RESOLUTION_WEIGHT_LOAD	FLOAT	DB241.DBD568
DR3_RESOLUTION_FLOW_RATE	FLOAT	DB241.DBD572
DR3_RESOLUTION_MASTER_TOTALIZER	FLOAT	DB241.DBD576
DR3_DESIGN_FLOW_RATE	FLOAT	DB241.DBD580
DR3_WEIGH_LENHT	FLOAT	DB241.DBD584
DR3_BELT_LENGTH	FLOAT	DB241.DBD588
DR3_DESIGN_SPEED	FLOAT	DB241.DBD596
DR3_PULSES_PER_LENGTH_UNIT	FLOAT	DB241.DBD608
DR3_CALIBRATION_WEIGHT	FLOAT	DB241.DBD620
DR3_CALIBRATION_LOAD	FLOAT	DB241.DBD624
DR3_CALIBRATION_QUANTITY	FLOAT	DB241.DBD628
DR4_PULSE_PER_BELT_REVOLUTION	INT32	DB241.DBD648
DR4_DESIGN_SPEED	FLOAT	DB241.DBD668
DR4_PULSE_PER_LENGTH_UNIT	FLOAT	DB241.DBD680
DR4_ZERO_CALIBRATION_DIGIT	INT32	DB241.DBD704
DR4_CALIBRATION_WEIGHT	FLOAT	DB241.DBD712
DR4_SPAN_CALIBRATION_DIGITS	INT32	DB241.DBD720
DR4_DEV_FROM_CURRENT_SPAN_DIGITS	FLOAT	DB241.DBD724
DR5_LOAD_FACTOR_1	FLOAT	DB241.DBD756
DR5_CORRECTION_FACTOR_1	FLOAT	DB241.DBD760
DR5_LOAD_FACTOR_2	FLOAT	DB241.DBD780
DR5_CORRECTION_FACTOR_2	FLOAT	DB241.DBD784
DR10_NUMBER_OF_LOAD_CELLS	UINT16	DB241.DBW1038
DR10_SWITCH_50_60_HZ	UINT16	DB241.DBW1040
CMD_kod	INT16	DB241.DBW468
DR3_UNIT_BELT_LOAD	UINT16	DB241.DBW564
DR3_UNIT_FLOW_RATE	UINT16	DB241.DBW566
DR3_NO_OF_BELT_REVOLUTIONS	UINT16	DB241.DBW592
DR3_SPEED_DETECTION	UINT16	DB241.DBW594
CMD_TRIGG	BOOL	DB241.DBX470.0
CMD_Running	BOOL	DB241.DBX470.1
Alarm_CMD_Error	BOOL	DB241.DBX470.3
Alarm_ApplIDError	BIT	DB241.DBX486.0
Alarm_ApplIDDError1	BIT	DB241.DBX486.1
Alarm_SFBError	BIT	DB241.DBX486.2
Alarm_RdPerError	BIT	DB241.DBX486.3
Alarm_LifeBitError	BIT	DB241.DBX486.4
Alarm_StartUpError	BIT	DB241.DBX486.5
Alarm_DataOperationError	BIT	DB241.DBX487.0
DR_BELT_STATUS	BOOL	DB241.DBX494.4
DR_SCALE_STATUS_1_TOTALIZER_ERROR	BOOL	DB241.DBX494.7
Alarm_Totalizer_Error	BIT	DB241.DBX494.7
DR_CALIBRATION	BOOL	DB241.DBX496.2
DR_SCALE_STATUS_2_ERROR	BIT	DB241.DBX496.7
Alarm_ModulError	BIT	DB241.DBX496.7
DR_SCALE_STATUS_2_TIME_BUFFER_ERROR	BIT	DB241.DBX497.5
DR_SCALE_STATUS_2_DIGITAL_INPUT_ERROR	BIT	DB241.DBX497.7
Alarm_DigitalInputError	BIT	DB241.DBX497.7
Alarm_Info	DEFAULT	
Alarm_Ack_All	DEFAULT	
Alarm_Ack_Selected	DEFAULT	
Alarm_Clear	DEFAULT	

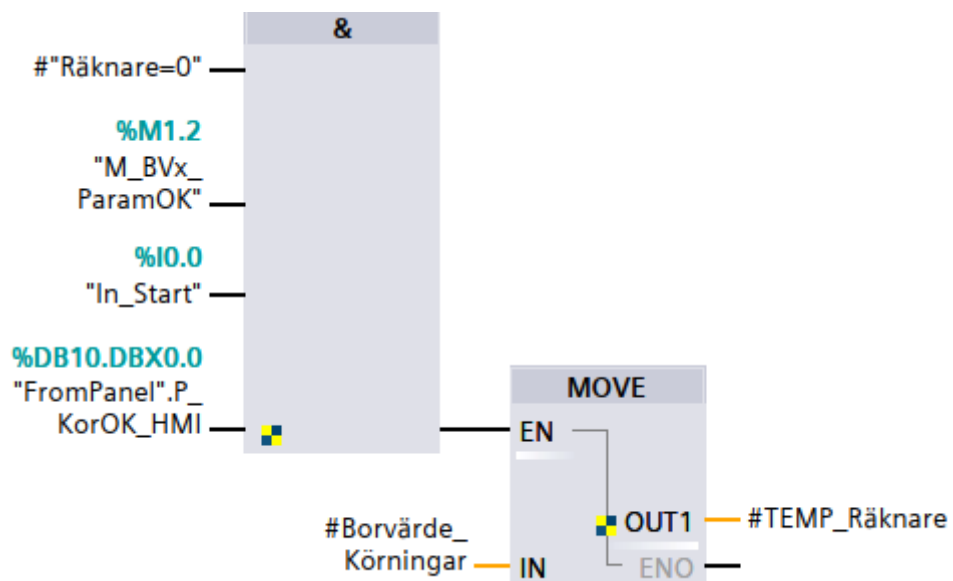
4 Bandvågs specifika funktioner

4.1 Counter

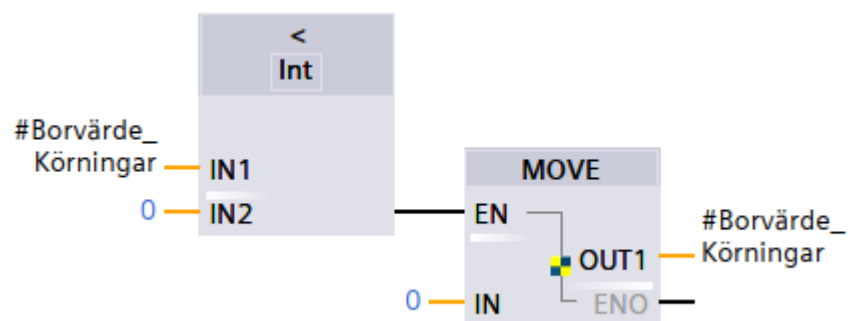
4.1.1 Nollning av räknare



4.1.2 Överföring av börvärde till temp vari



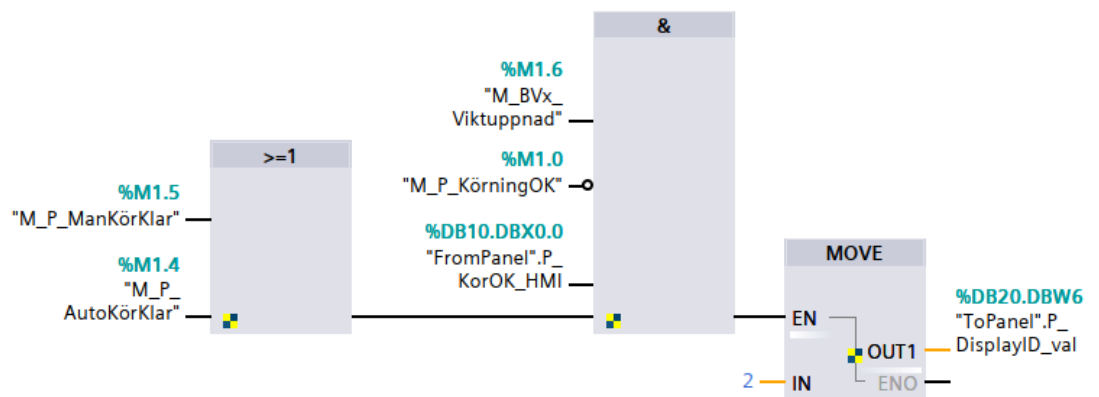
4.1.3 Negativ tals förhindring



4.1.4 Körningsräknare

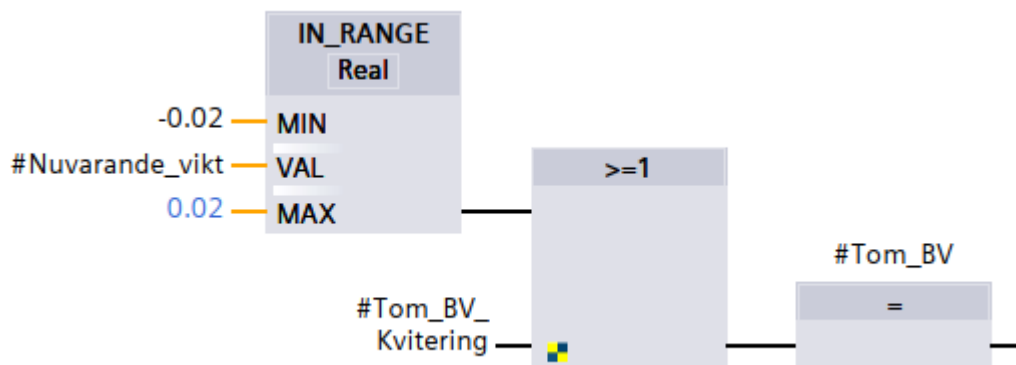


4.1.5 Avslutad körning

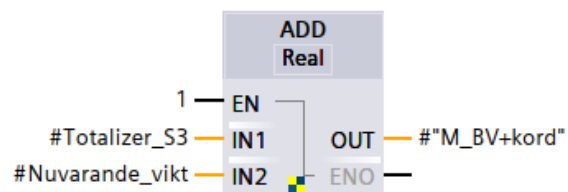


4.2 Weigher

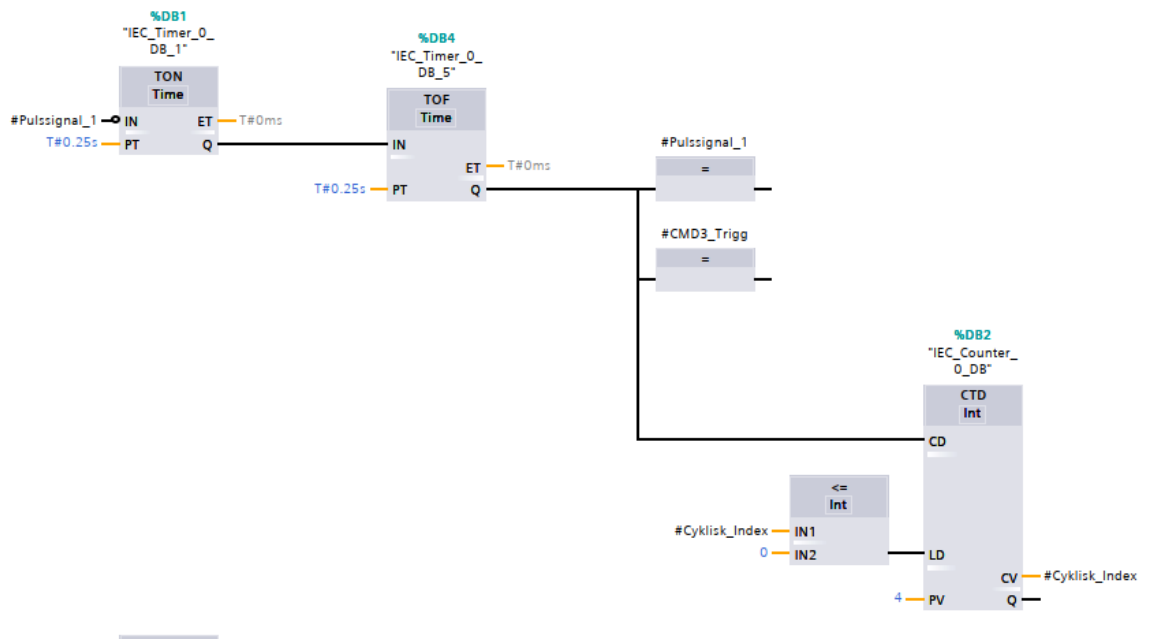
4.2.1 Tom last cell



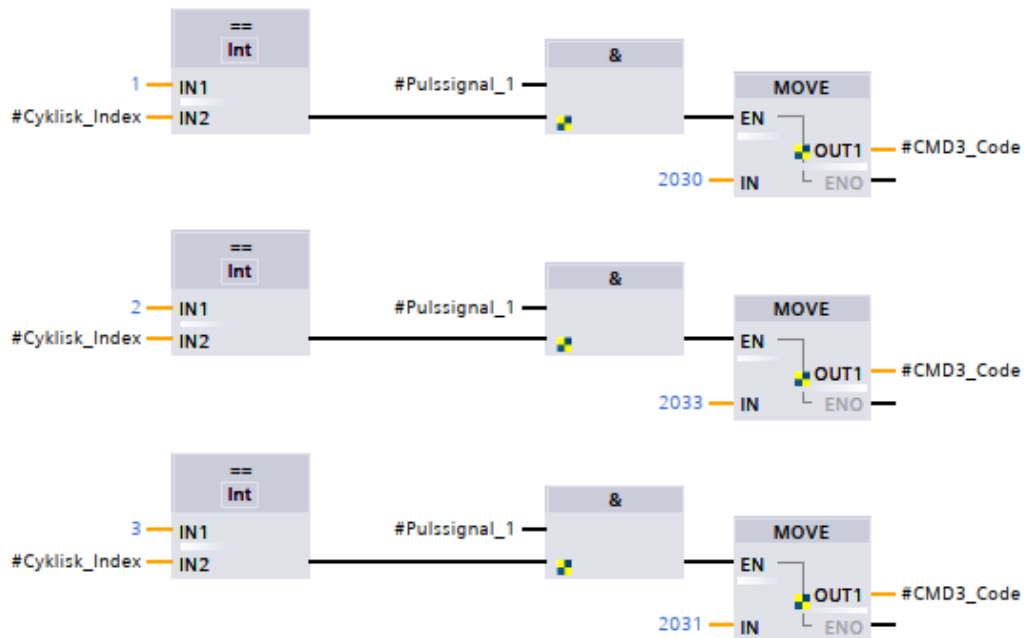
4.2.2 Summering av nuvarande vikt och genomkörd vikt



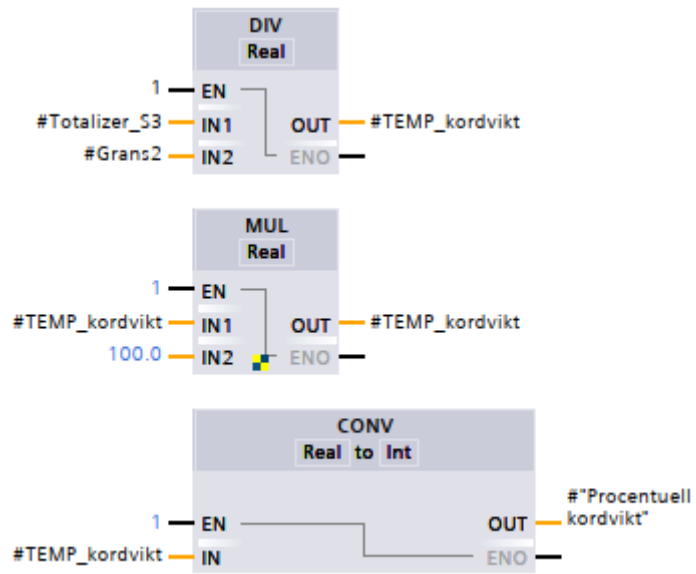
4.2.3 Trigg pulssignal



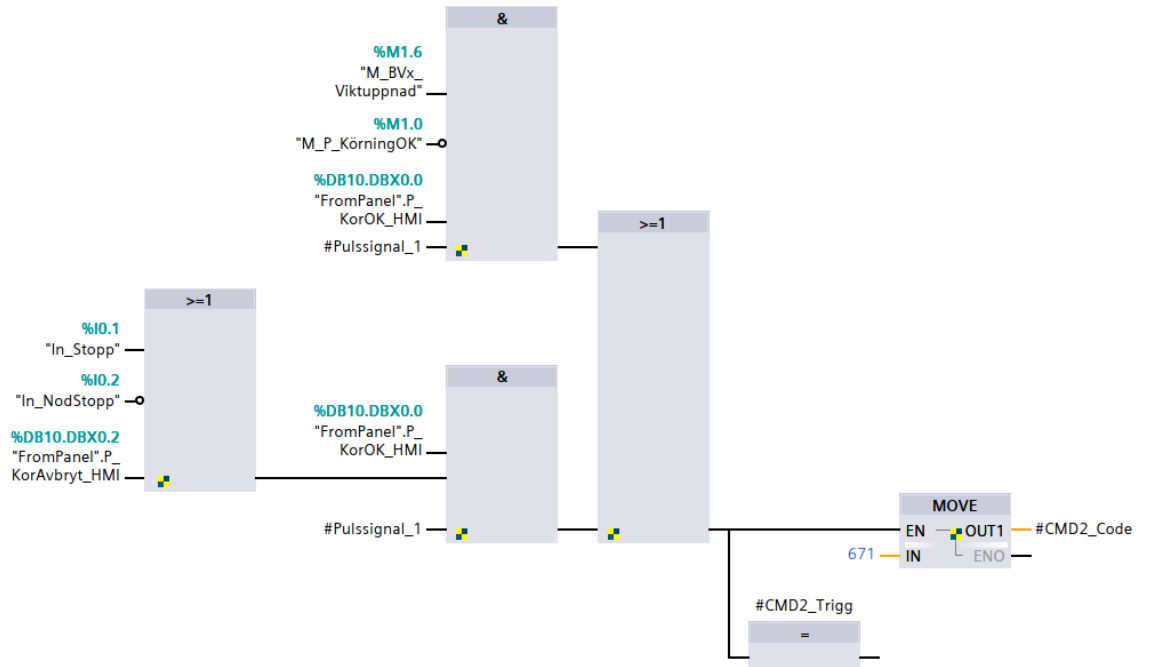
4.2.4 Cyklisk inläsning av data



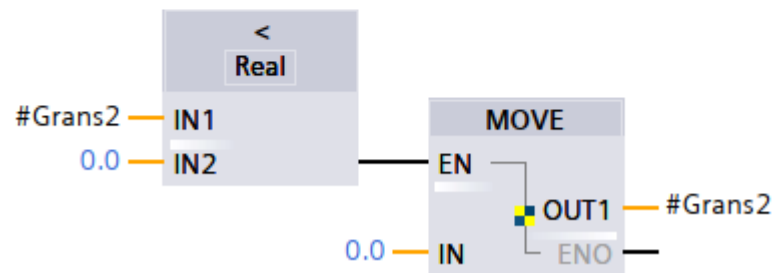
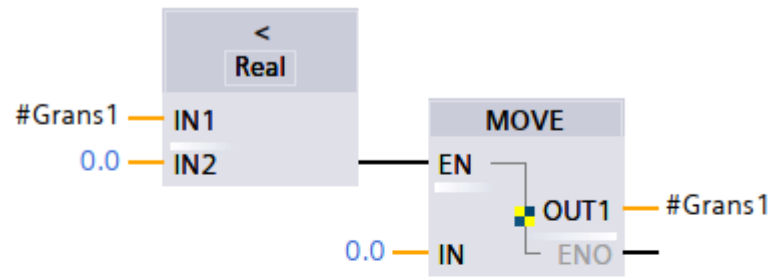
4.2.5 Uträckning av procentuell genomkörd vikt



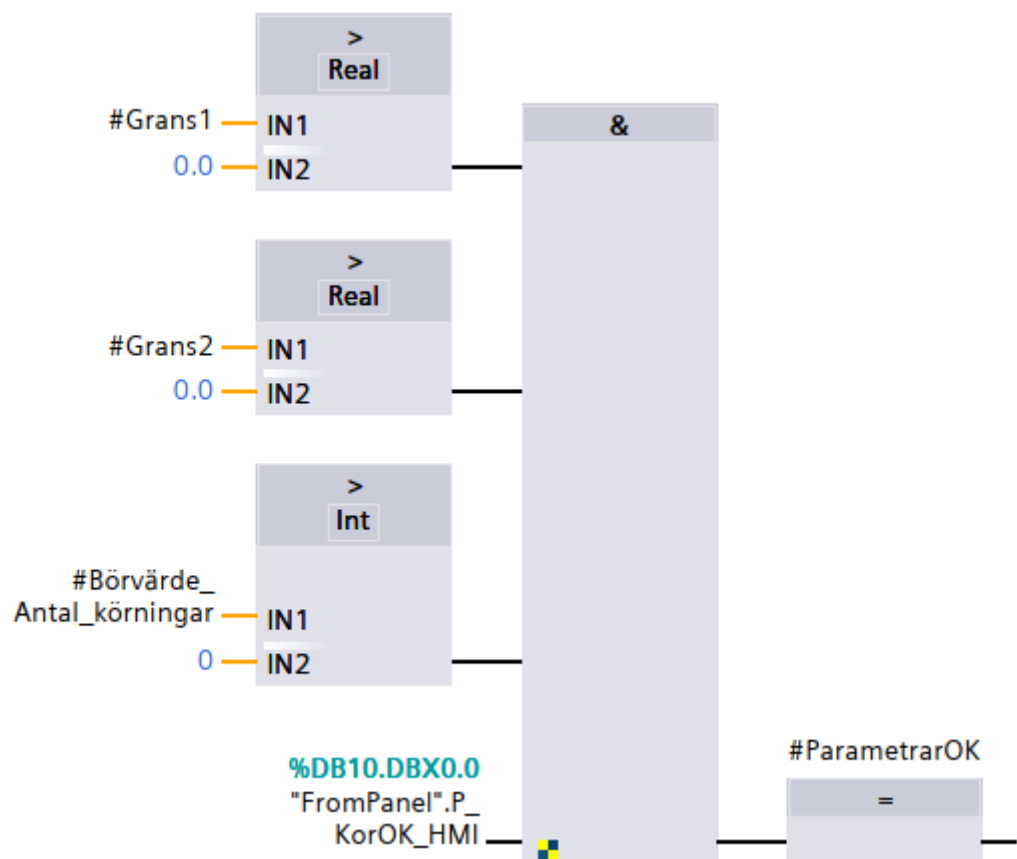
4.2.6 Nollning av totalräknare S3



4.2.7 Negativ gränsvärdes övervakning

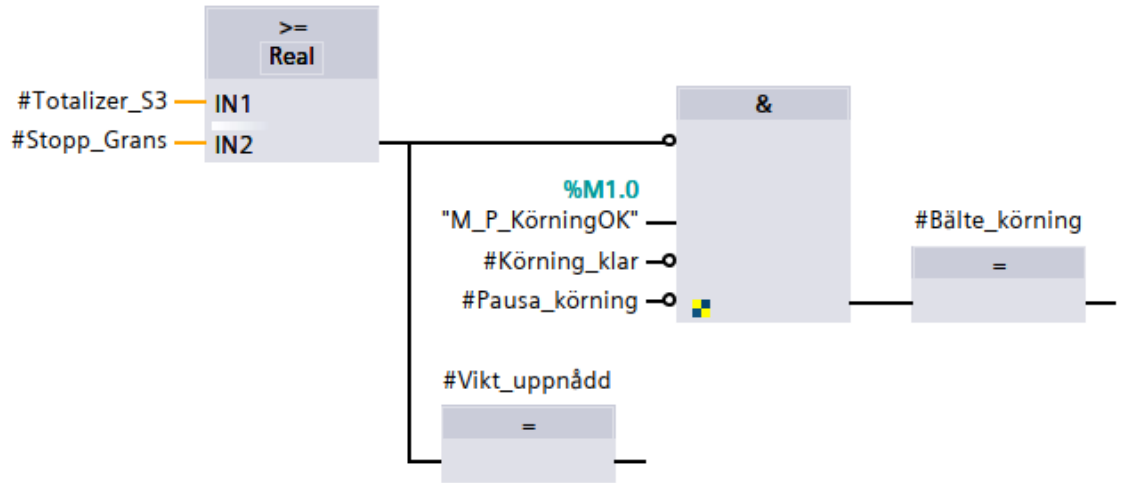


4.2.8 Parameter indikation

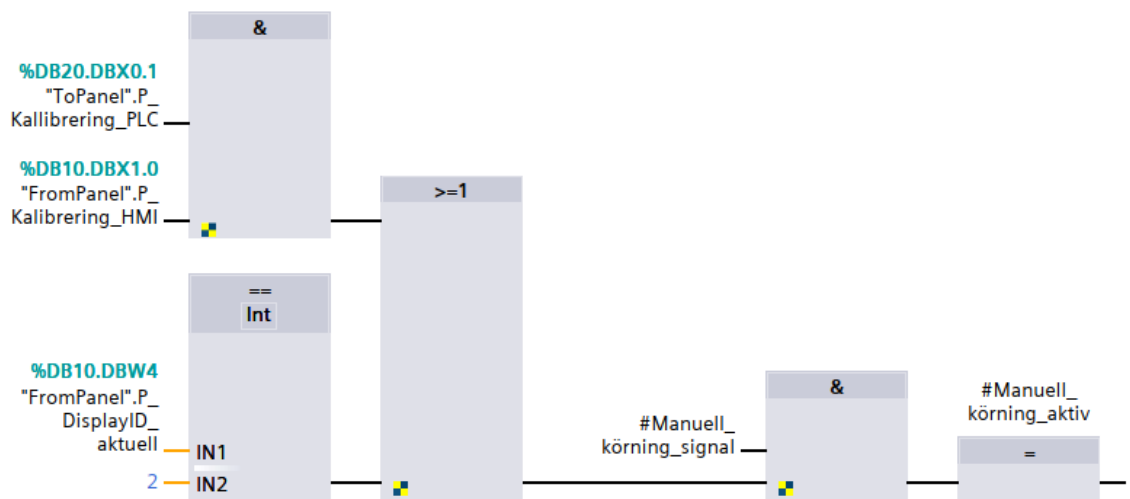


4.3 Feeder

4.3.1 Bandvågs bälte start stopp

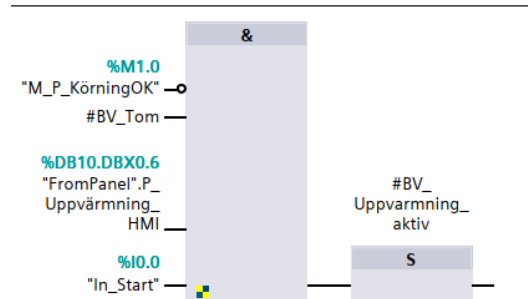


4.3.2 Manuell körning av bandvågs bälte



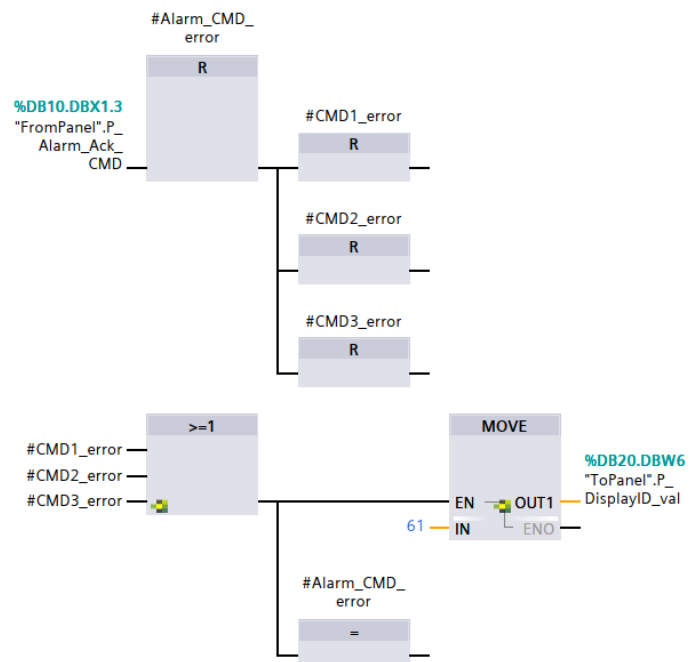
4.4 PreHeat

4.4.1 Körning ok

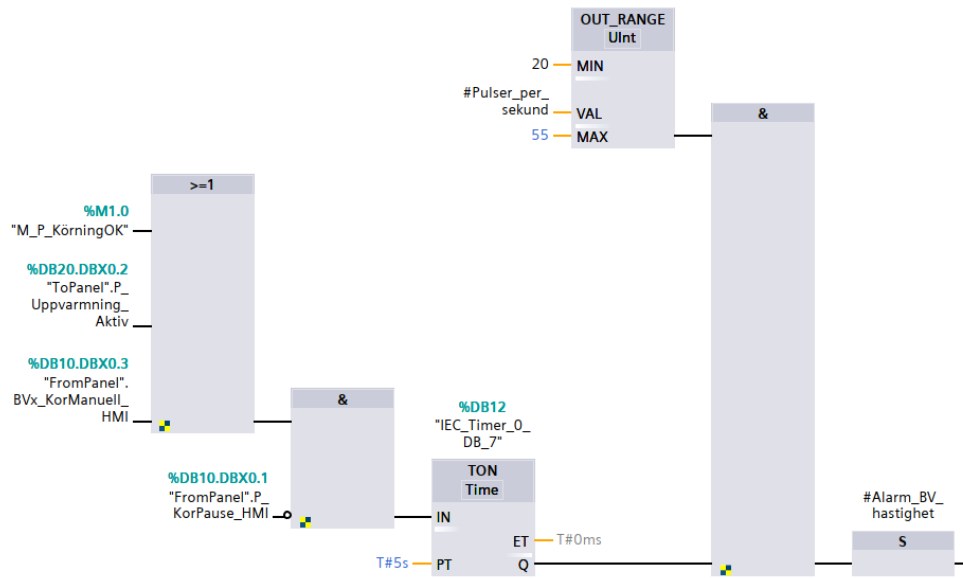


4.5 Alarm

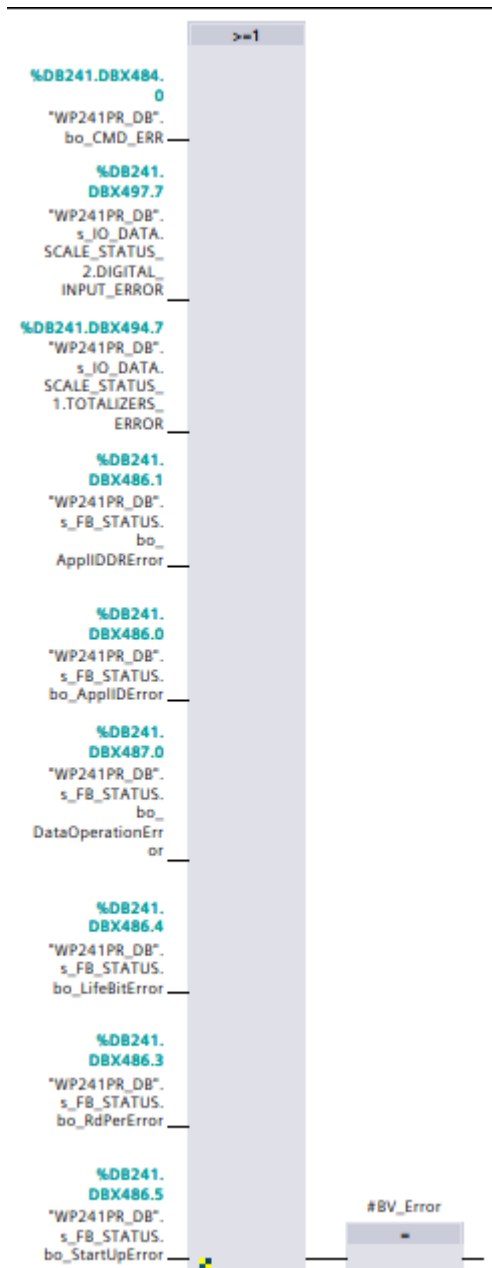
4.5.1 CMD alarm



4.5.2 Bandvågs hastighets fel

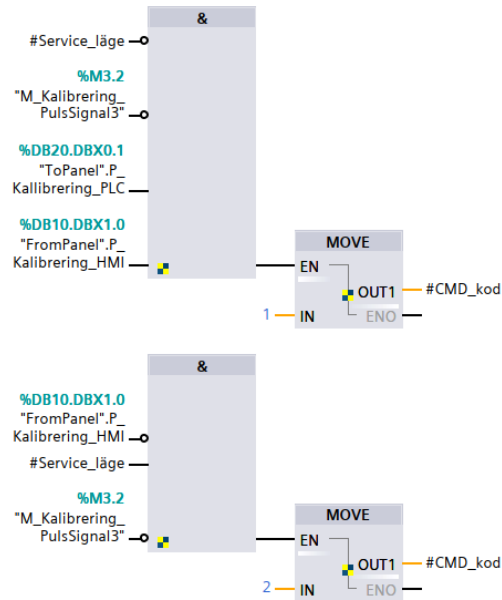


4.5.3 Bandvågs fel

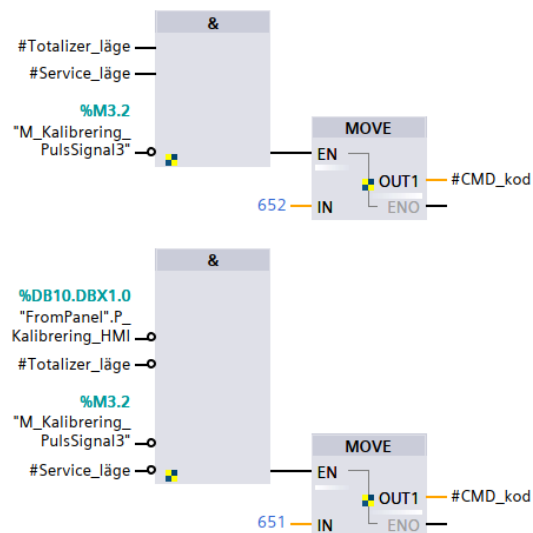


4.6 Calibration

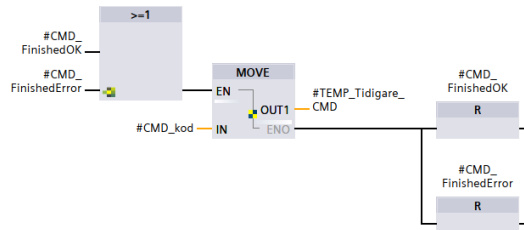
4.6.1 Service läges styrning



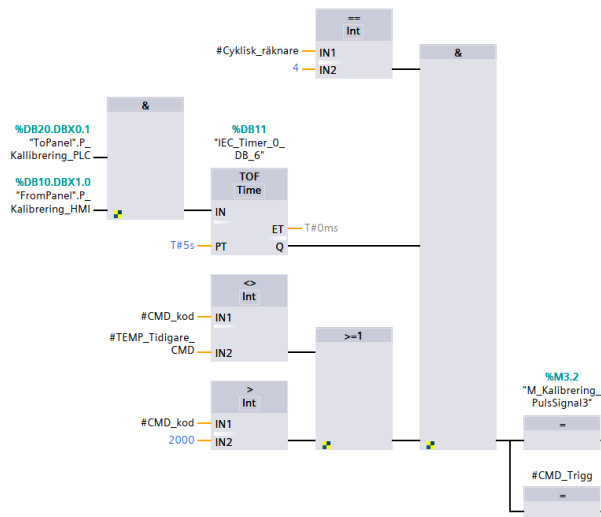
4.6.2 Totaliserar styrning



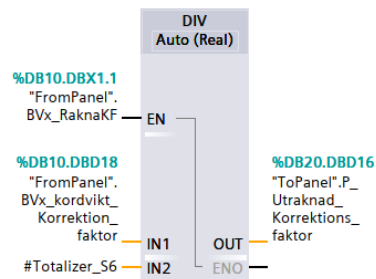
4.6.3 Tidigare kommando kod



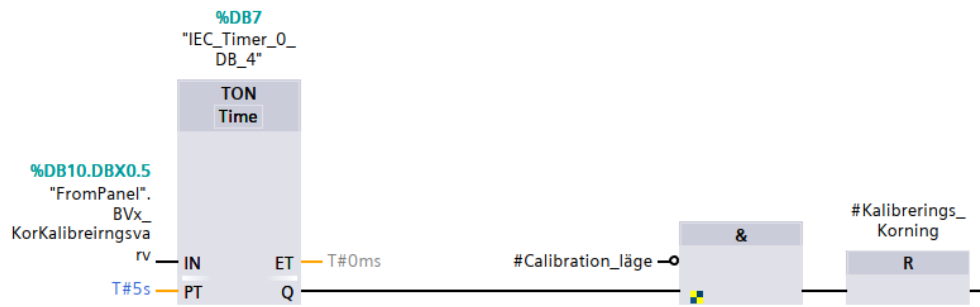
4.6.4 Trigg puls



4.6.5 Uträkning av korrektions faktor

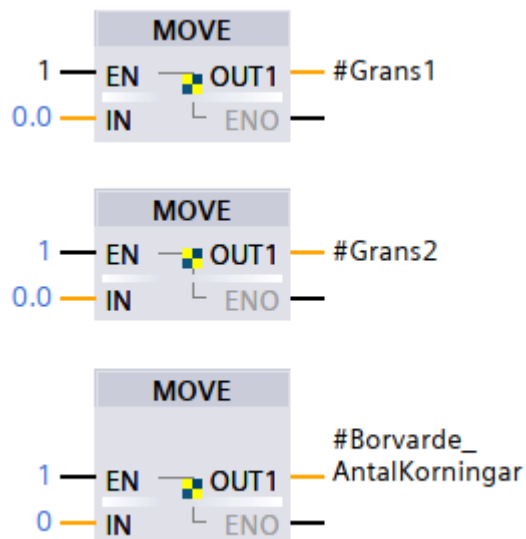


4.6.6 Kalibrerins körning



4.7 Restart

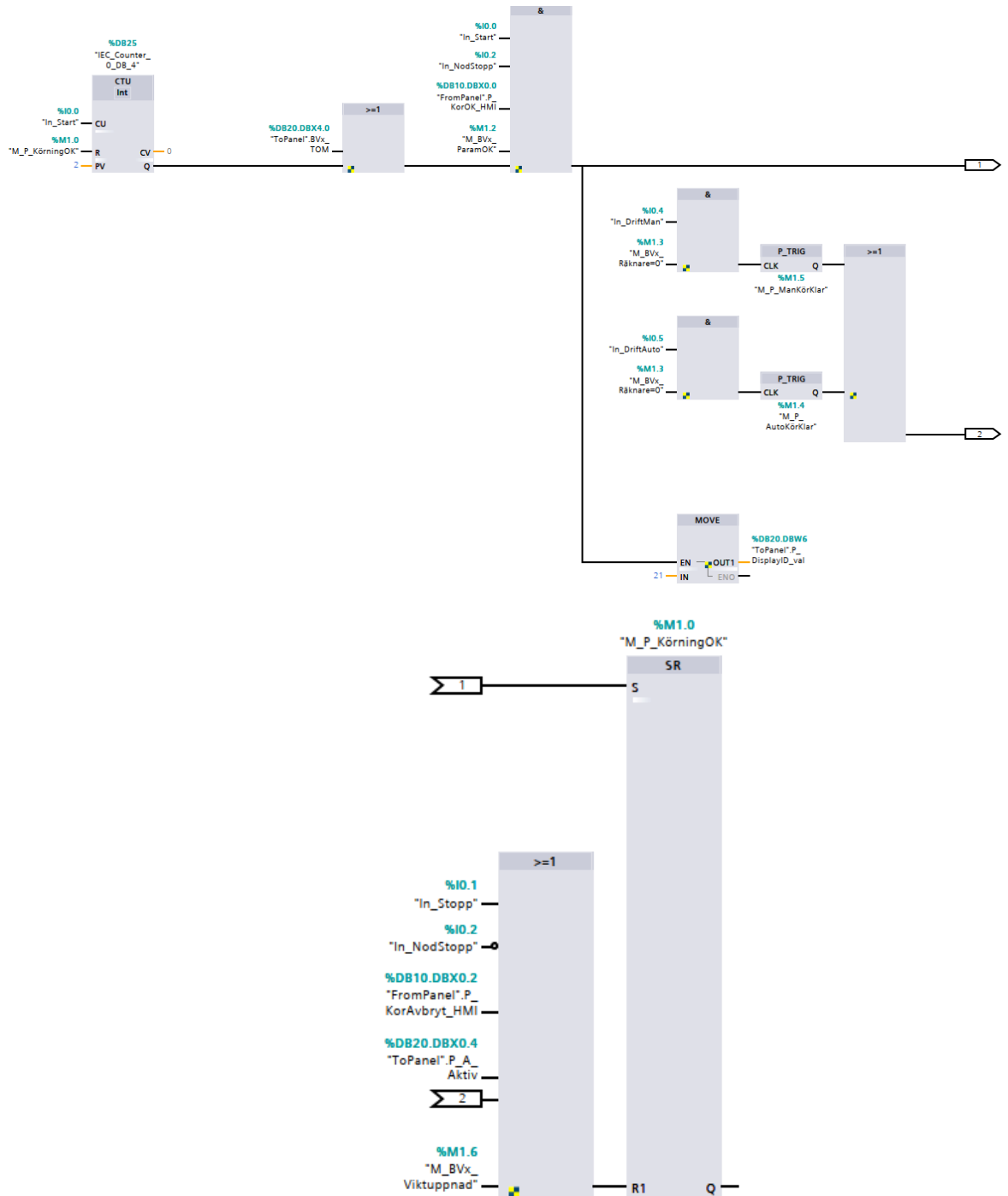
4.7.1 Nollning av param



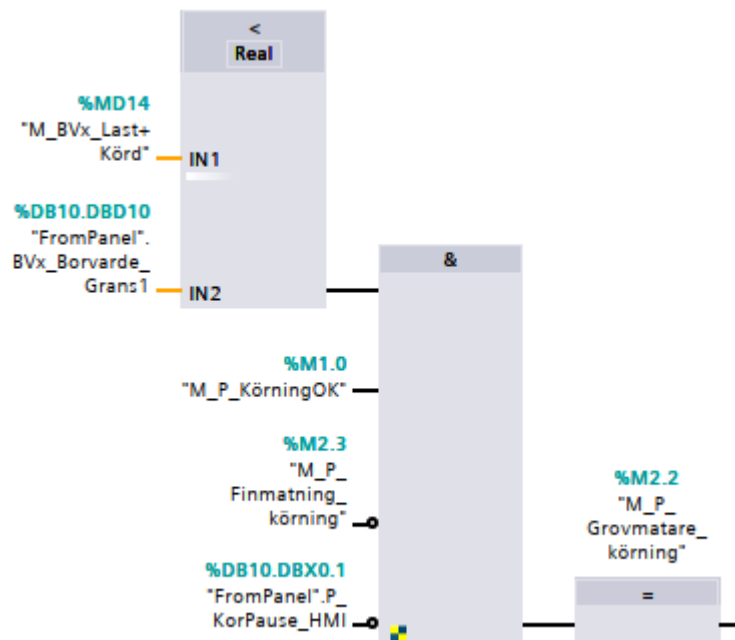
5 Process-specifika funktioner

5.1 Feeder

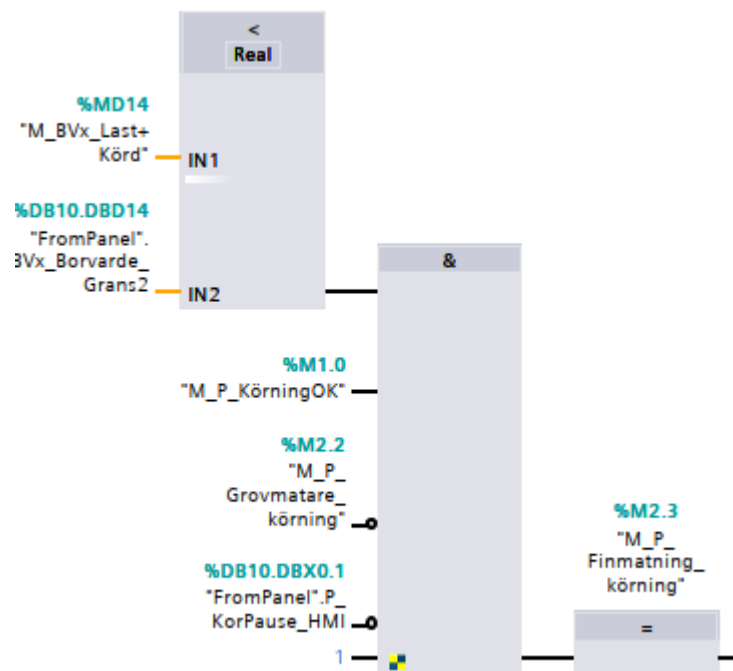
5.1.1 Godkän körning



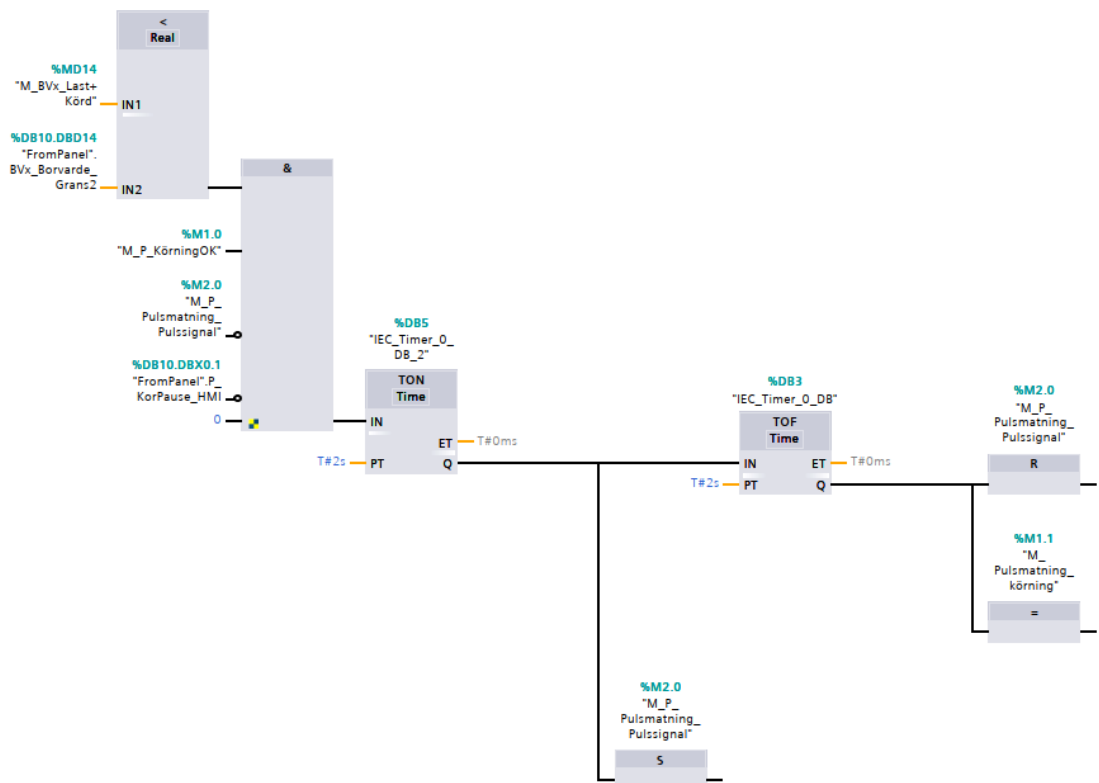
5.1.2 Körning av grovmatare



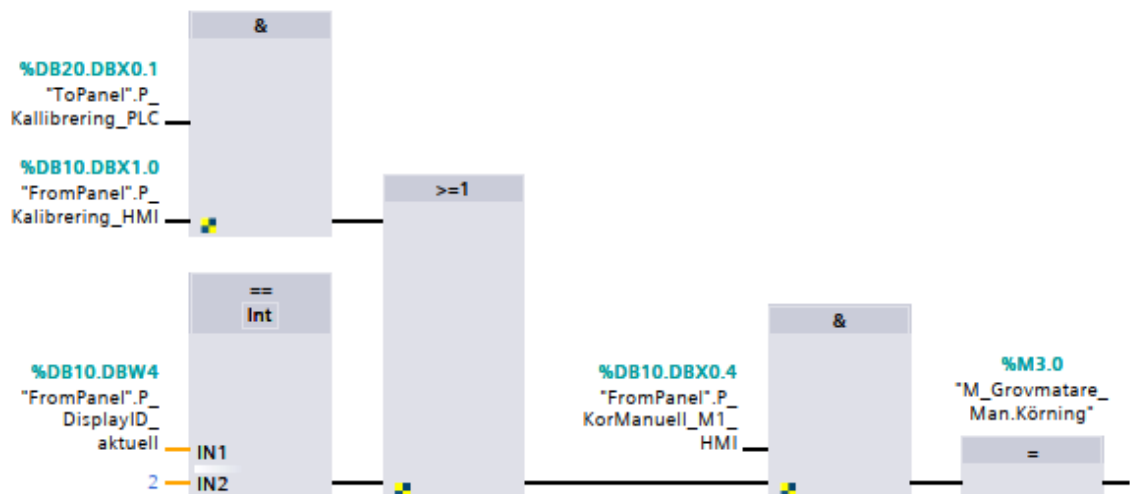
5.1.3 Körning av finmatning



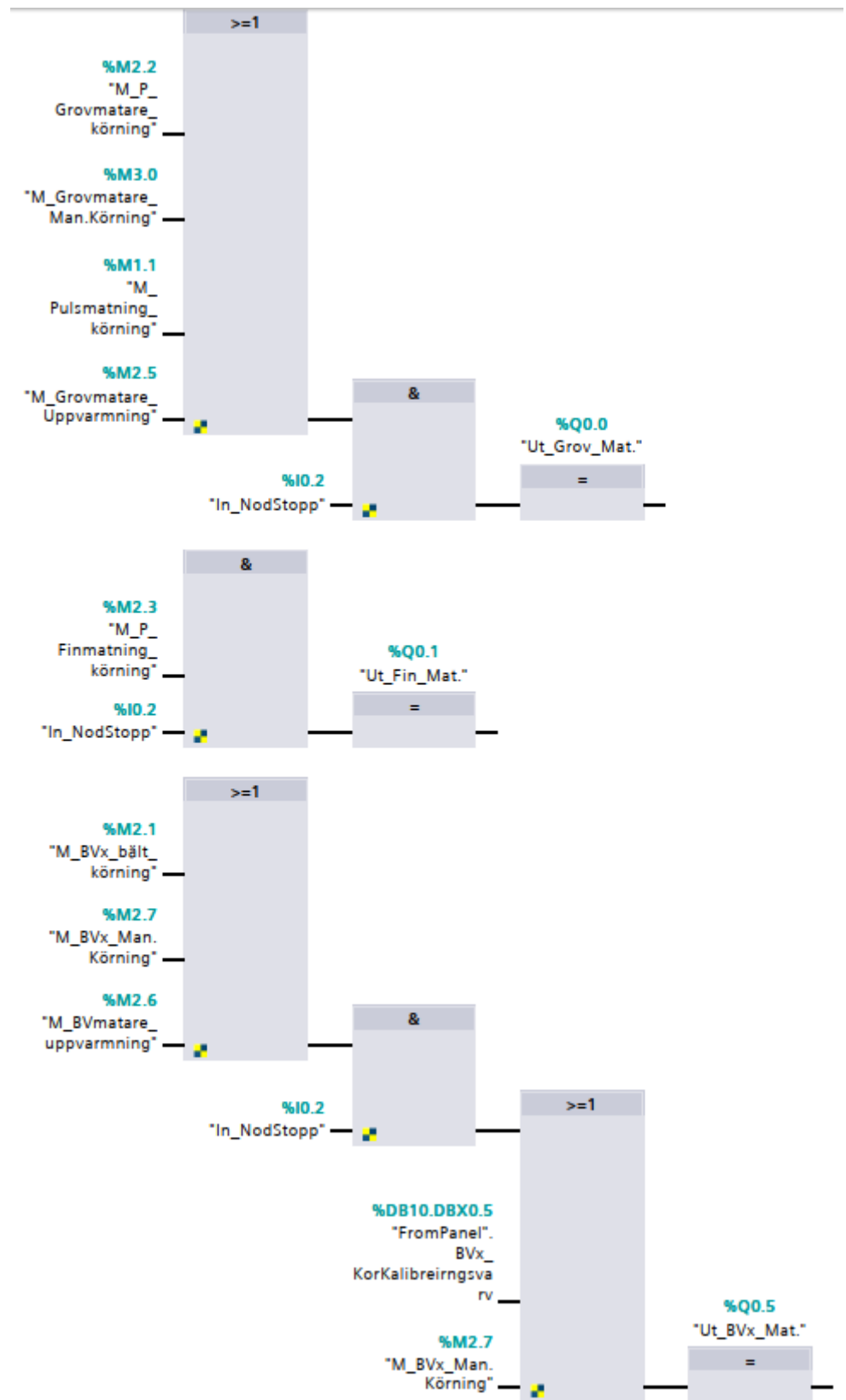
5.1.4 Körning av pulsmatning



5.1.5 Manuell körning av M1

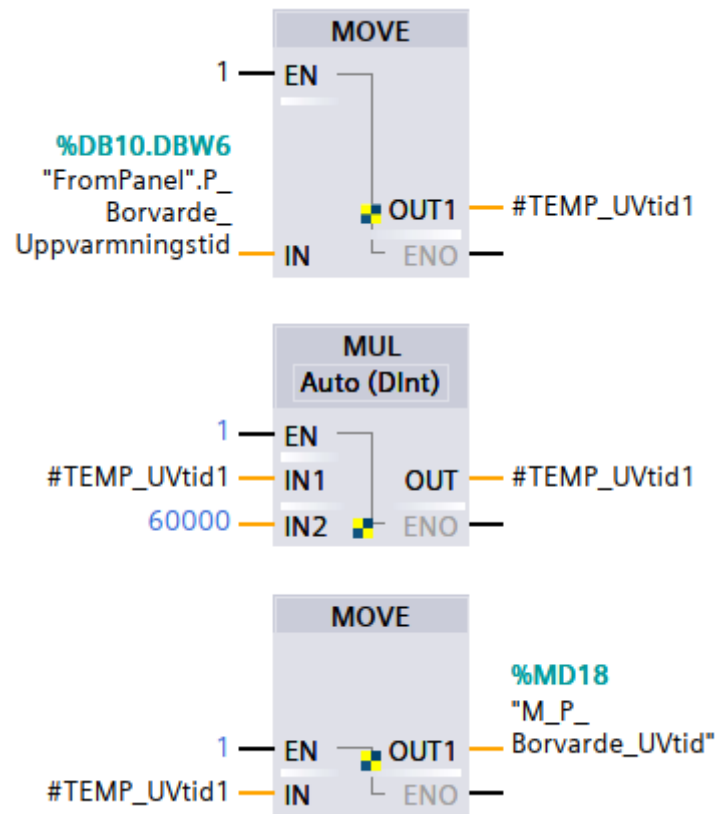


5.1.6 Motorstyrning

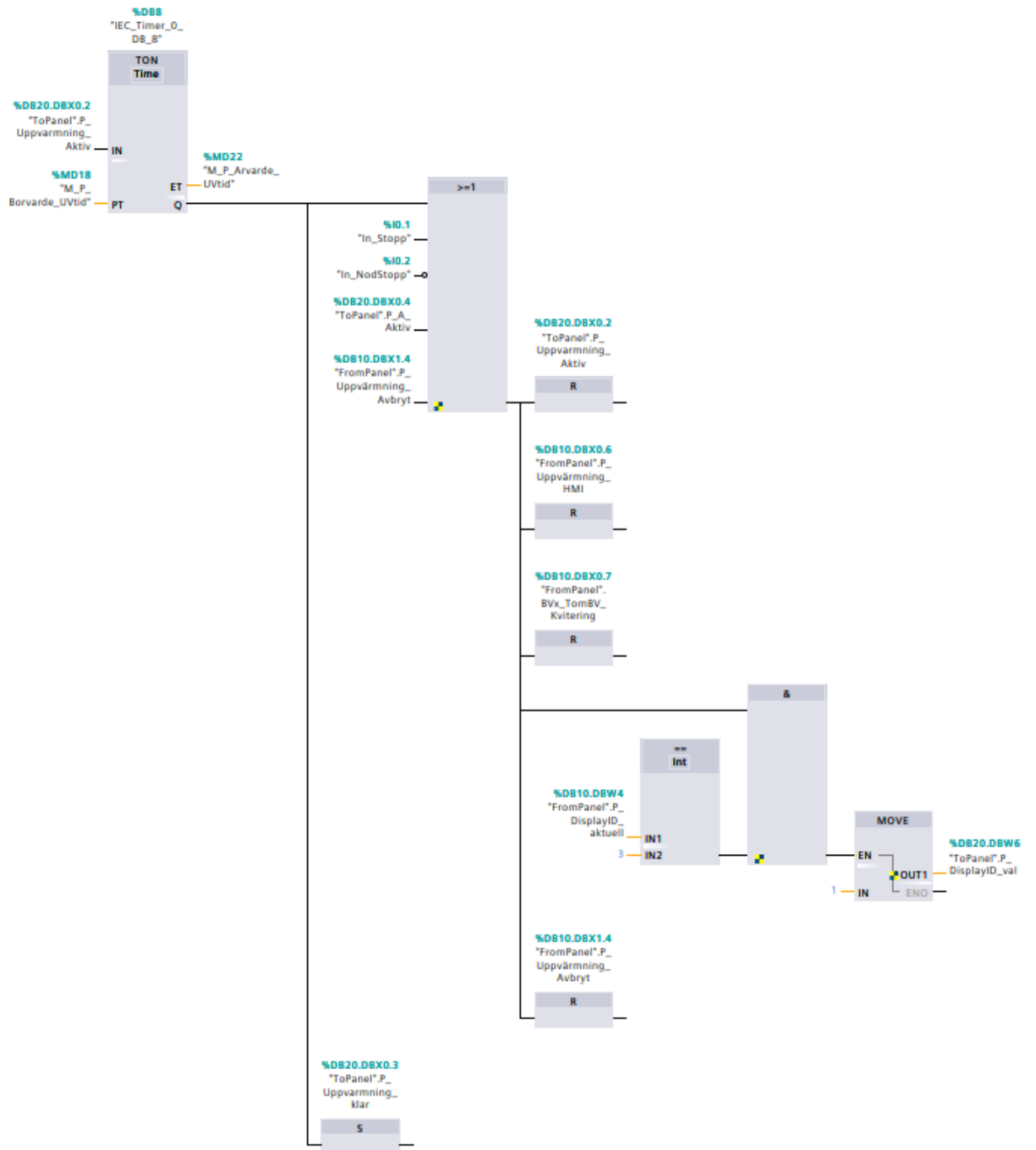


5.2 PreHeat

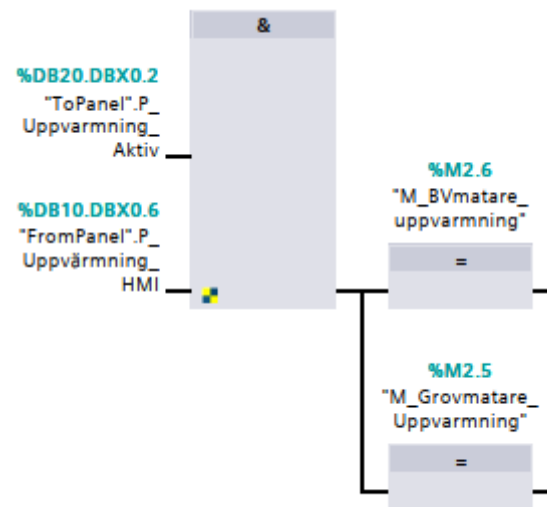
5.2.1 Börvärdes behandling



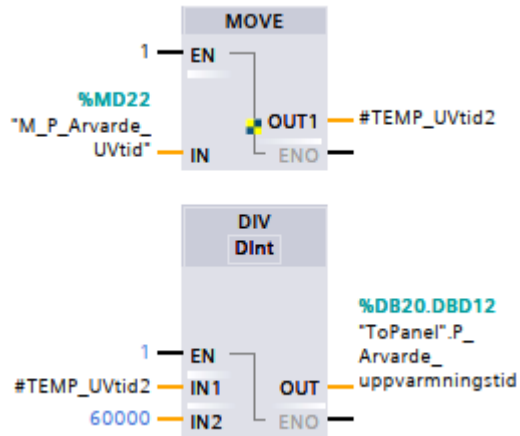
5.2.2 Klar/avbryt körning



5.2.3 Körning

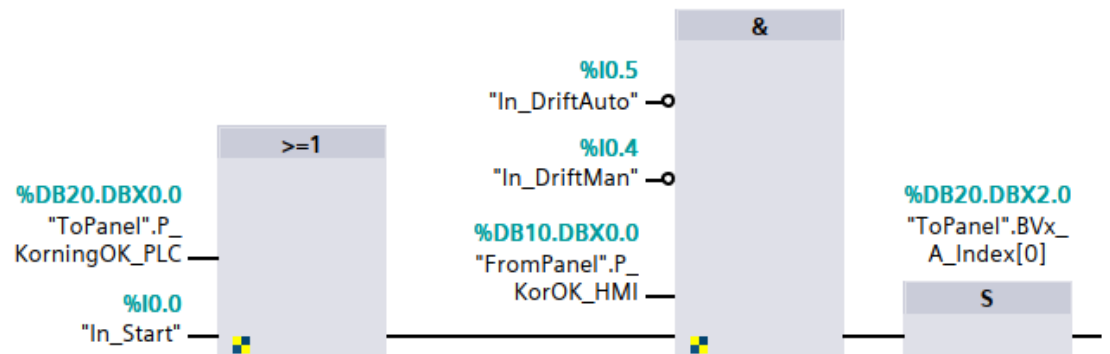


5.2.4 Ärvärdes behandling

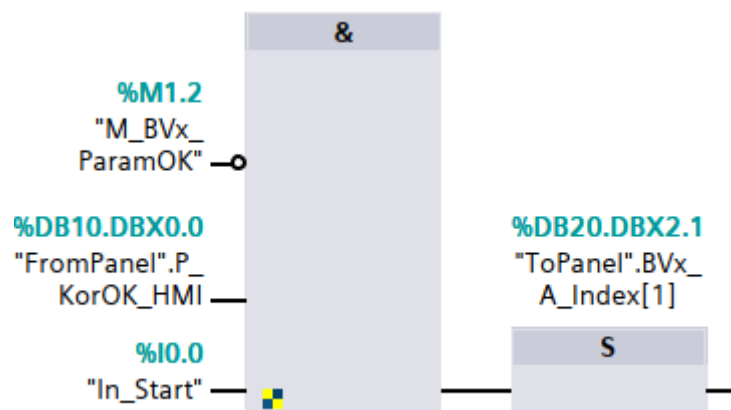


5.3 Alarm

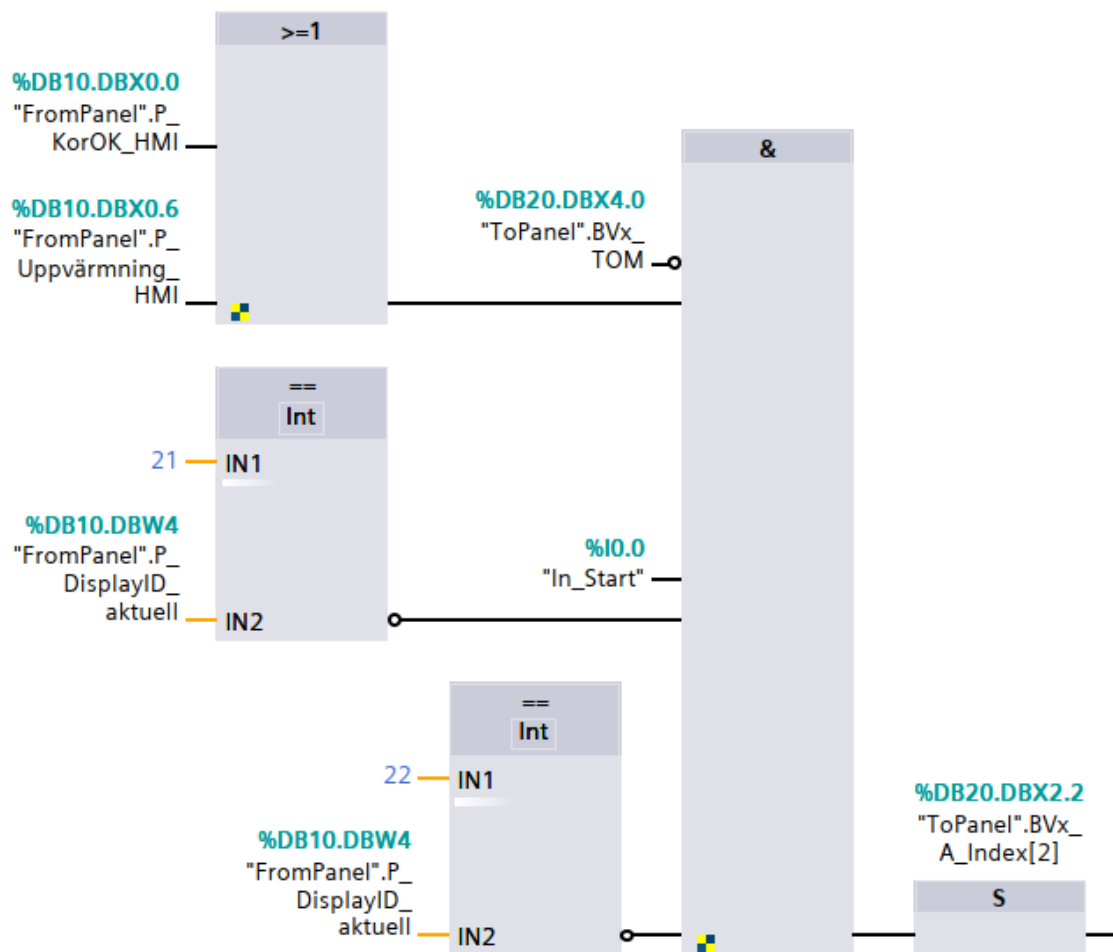
5.3.1 Man auto inte invält



5.3.2 Parametrar inte inställda



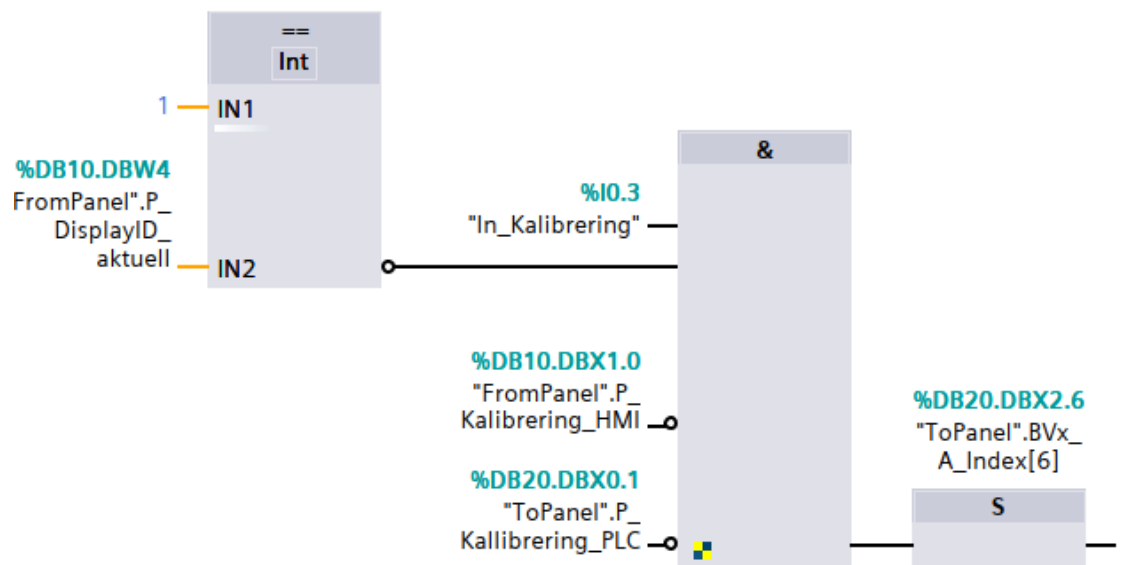
5.3.3 Bandvågen är inte tom



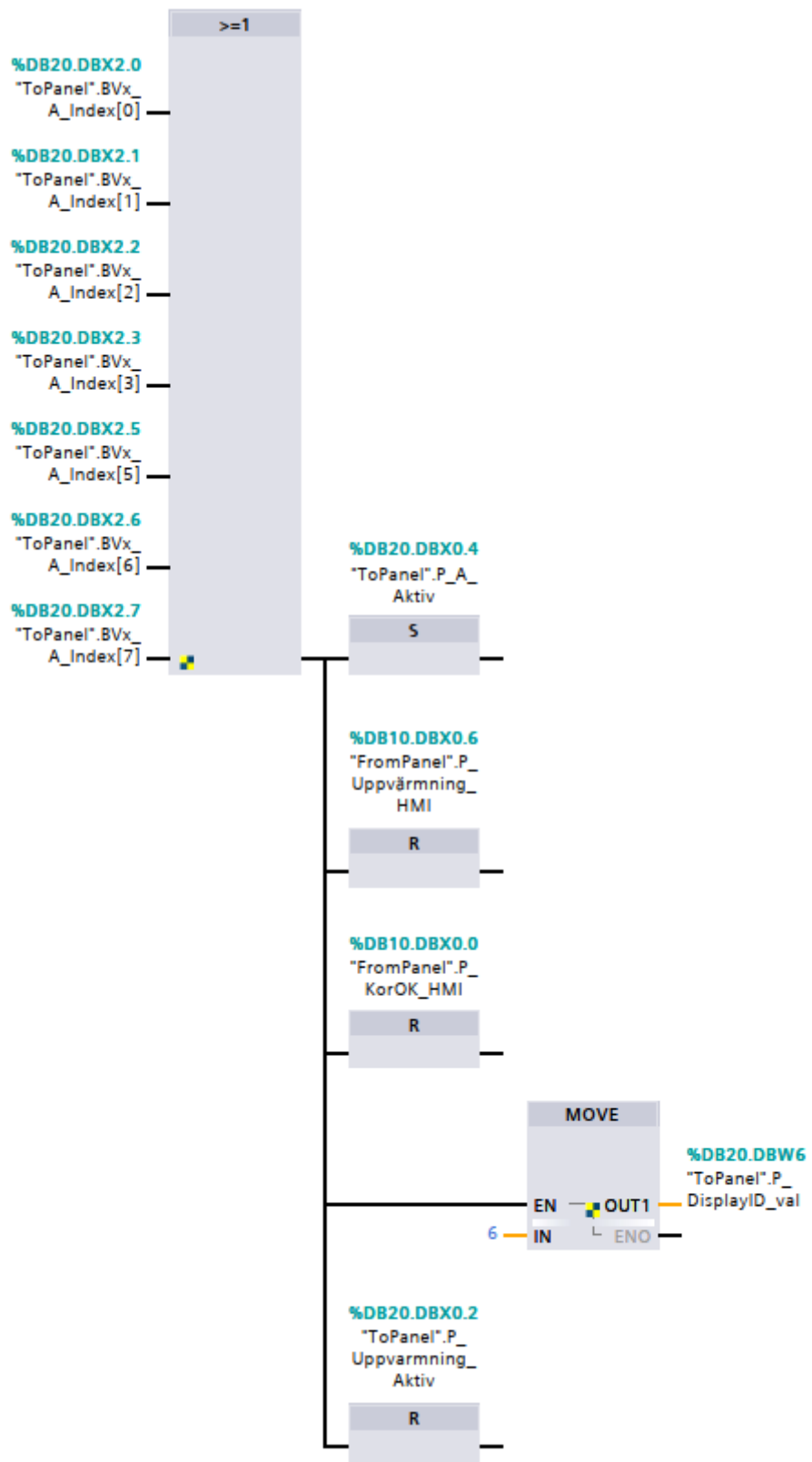
5.3.4 Nodstopp aktiverat



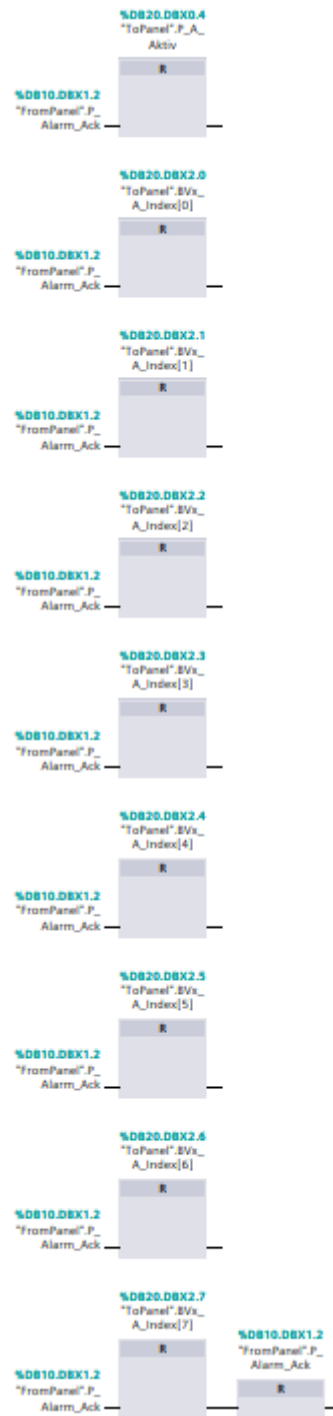
5.3.5 Kalibrering aktiv



5.3.6 Alarm aktivt

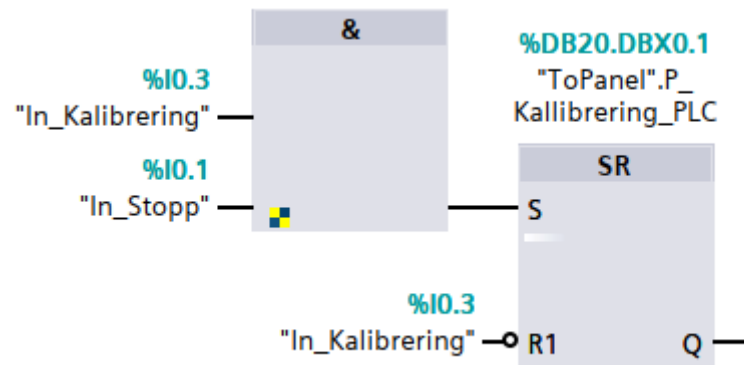


5.3.7 Alarm kvitering



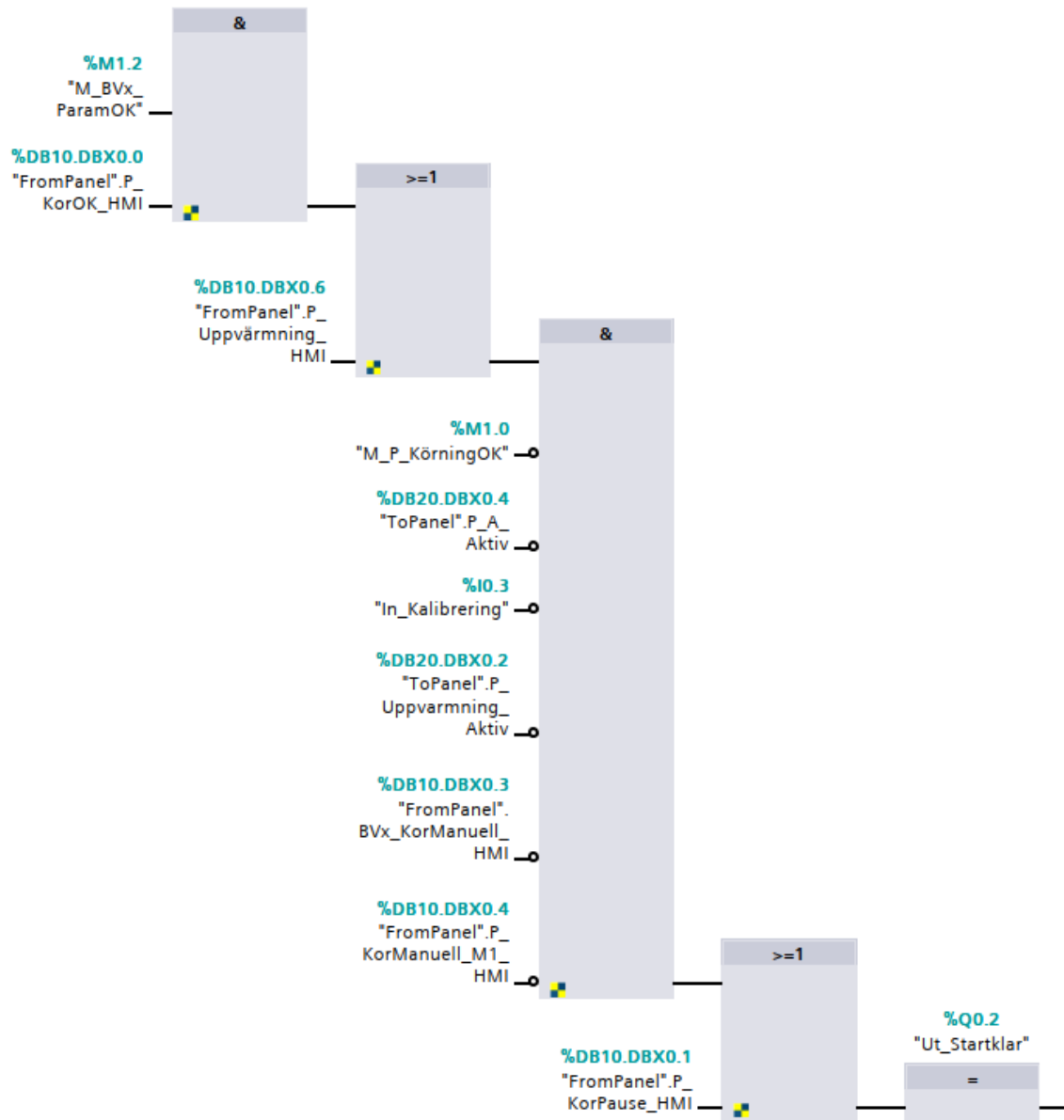
5.4 Calibration

5.4.1 Aktivering av kalibrering

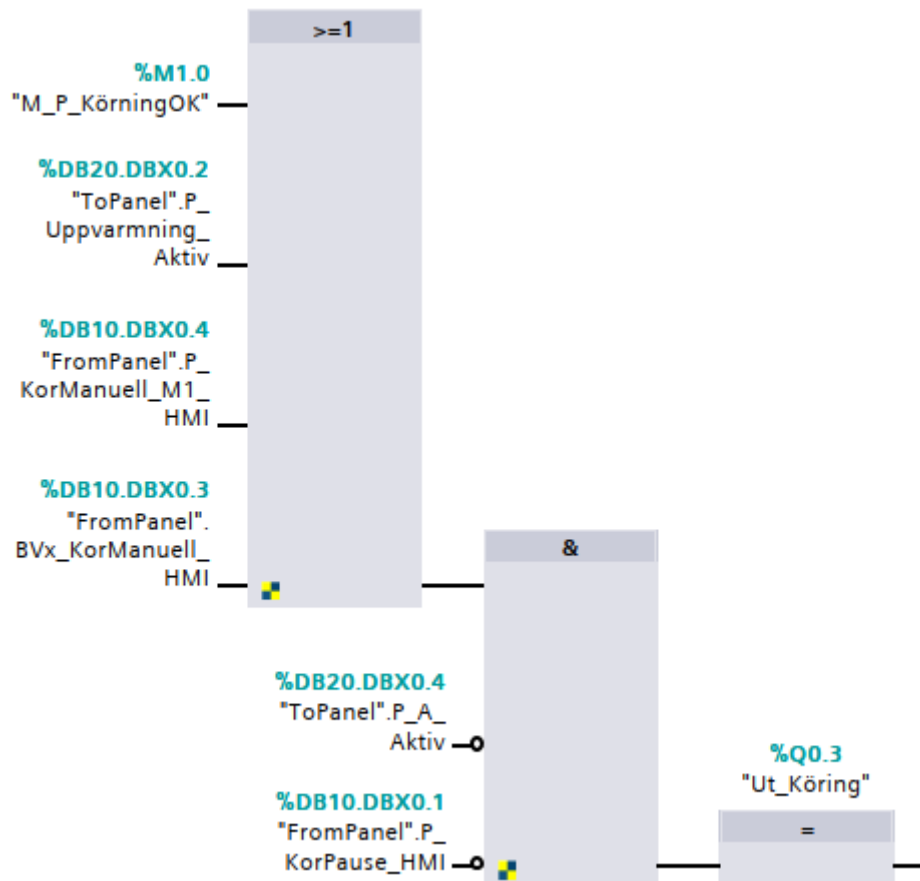


5.5 Indications

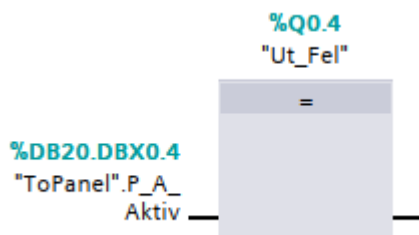
5.5.1 Klar för körning



5.5.2 I körning

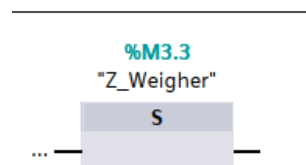
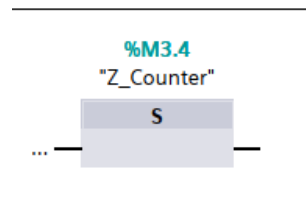


5.5.3 Alarmering



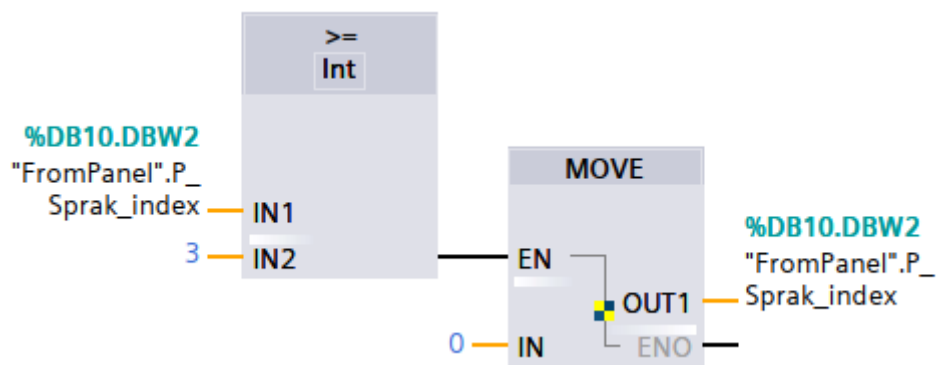
5.6 Daily Interrupt

5.6.1 Nollning av räknare och vägare



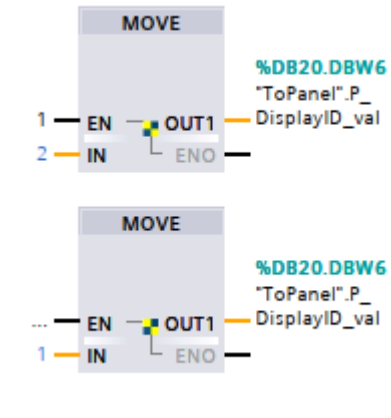
5.7 Language

5.7.1 Nollning av språk index

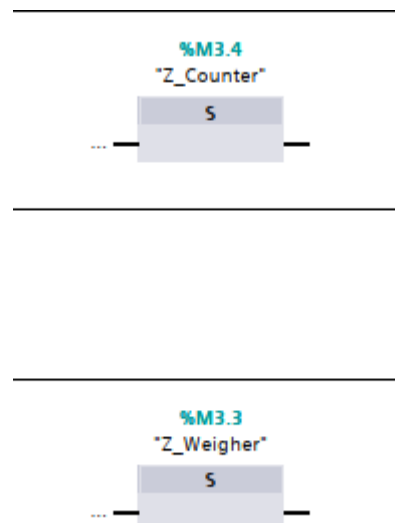


5.8 Restart

5.8.1 Byte till huvud display



5.8.2 Nollning av räknare och vägning



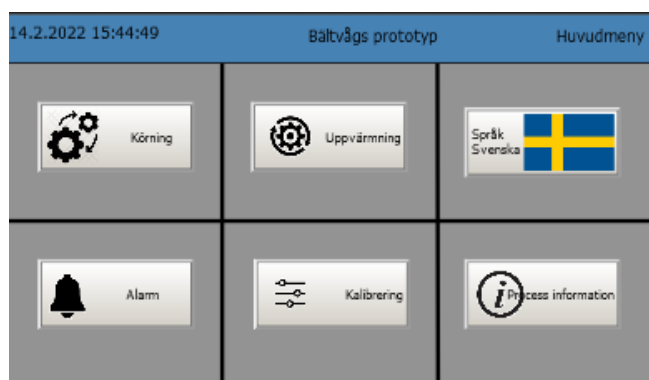
6 Användarinstruktioner för bandvågsprocessens användargränssnitt

Användargränssnitten öppnas alltid upp i huvudmenyn. Från huvudmeny kan användare navigera till processen olika funktioner:

- Körning
- Uppvärmning
- Alarmlogg
- Kalibrering och informationsfönster

Språkvalet sker i huvudmenyn genom att trycka på språk knappen. Ett tryck på knappen scrollar igenom de tre inkluderade språken, svenska finska och engelska.

Kalibrerings knappen är per default låst. För att kunna komma till kalibrerings menyn måste kalibrerings vippan vara aktiv och stoppknappen påverkas, detta öppnar upp användningen av kalibrerings knappen i huvudmenyn.



6.1 Körning

Körning av processen fungerar i två steg, det första steget där användare anger parametrar för körningen och det andra där körningen övervakas.

I parameterfönstret ska två gränser anges ett för grovmatning och ett för finmatning samt hur många körningar som ska göras. För att bandvågen ska kunna startas måste vissa krav vara uppfyllda.

- Alla parameter värden måste vara över noll
- Automat- eller manuell-läget som aktiveras på den fysiska vippan måste också vara inställt
- Vikten på bandvågen måste vara $\pm 20g$, detta kan granskas fönstret nedre hörn.

Från parameterfönstret kan också bältens manuellkörning aktiveras. Detta görs genom att hålla in respektive körningsknapp, i fönstrets högra kant.

Tillbakaknappen kommer byta fönster tillbaka till huvudmenyn.

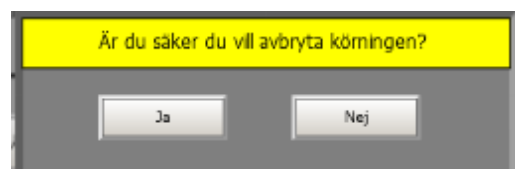
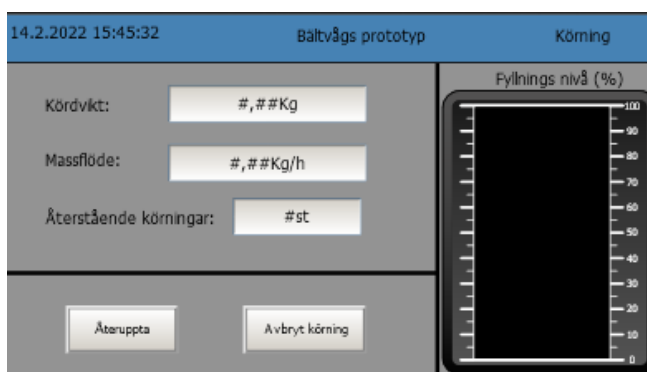


För att starta processen måste fysiska startknappen påverkas. Detta kommer starta bälten och byta fönster till körningsfönstret. I körnings fönstret visas den körda vikten i kg, massflödet i kg/h och återstående körninga. I fönstrets högra kant representeras den kördavikten grafisk som en procentuell summa.

Körningen kan via HMI:n pausas och återupptas genom att trycka på knappen till vänster, med respektive text. Vid paus av körning kommer bälten stoppas tillfälligt och vid återupptagande kommer körningen fortsätta där den slutade.

När en av flera körningar är klar kommer processen stanna och invänta ny startorder och när alla körningar är klara kommer körningen avslutas och fönstret bytas till parameterinställningen. Om startordern ska ges men bandvågen har material på sig måste startknappen dubbelklickas för att starta körningen.

För att avbryta körningen kan antingen den fysiska stoppknappen eller avbryt körning knappen påverkas. Den fysiska stoppknappen kommer genast avbryta körningen och byta fönstret till parameterinställningarna. Avbrytningen via HMI kommer öppna ett pop-up fönster, med alternativen att avbryta eller återgå till körningen.



6.2 Uppvärmning

Uppvärmningen startas genom att trycka på den fysiska startknappen när:

- Uppvärmningstiden är inställd
- Bandvågen är tom eller kvitterats tom

Som uppvärmningstid kan väljas 10, 20 eller 30 min, var av 30 minuter är den renommerade tiden.

Tom bandvåg kan konstateras om indikationen i fönstret lyser grönt. Om bandvågen i verkligheten är tom men inte kalibrerad kan "kvittera tom bandvåg"-knappen användas för att kvittera att bältet är tomt.

När uppvärmningen är startad börjar körningsindikationen lysa grönt och tiden kan iakttas i fönstrets nedre kant.



6.3 Alarm

Alarmeringen kan övervakas från alarmloggen. Här kommer alarm dyka upp med tidsstämpel, lägesindikation (med färg och text) och beskrivning.

- Om alarmet är gult tyder det på att alarmet inte längre är aktivt, men heller inte är kvitterat
- Om alarmet är rött tyder det på att alarmet är fortfarande aktivt och inte kvitterats.

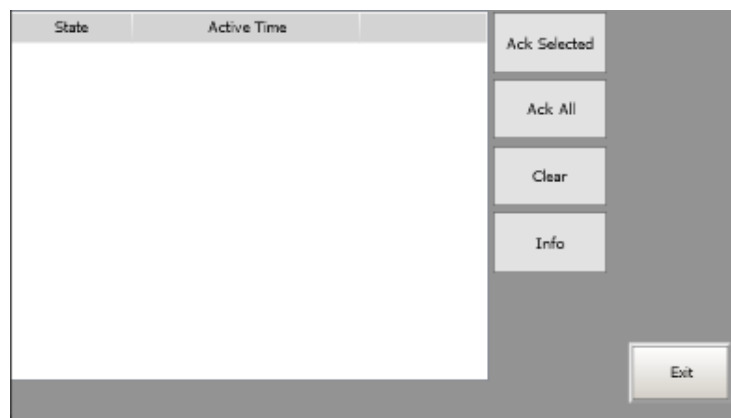
- Om alarmet är grönt tyder det på att alarmet kvitterats

Från alarmloggen kan man välja att, kvittera alla alarm (Ack All), kvittera endast valda alarm (Ack Selected), tömma alarmloggen (Clear) eller få fram mera information (Info) om valda alarmet.

Informations knappen är under arbete

När ett alarm sker under något skede i programmet kommer normal funktion avslutas och alarmloggen öppnas.

Genom att trycka på exit knapp kommer fönstret bytas till huvudmeny och alla alarm i PLC-programmet kommer kvitteras.




6.4 Process information

I processinformationsfönstret kan viktig information om bandvågsprocessen iakttas.

För att återvända tillbaka till huvudmeny kan tillbakaknappen påverkas.

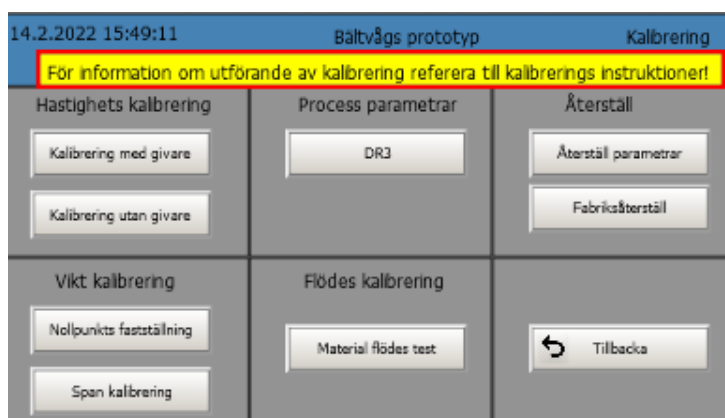
Under arbete

14.2.2022 15:48:54		Bältvågs prototyp	Indikatorer
Vågnamn	Bältvågs prototyp		 Tillbaka
Ibruktagnings datum	02.02.2022		
Tid sedan start			
Total kördvikt sedan start:	#,##Kg		
Användnings tid			
Total kördvikt sedan ibruktagnig:	#,##Kg		

6.5 Kalibrering av bandvågsmodul via HMI

När kalibreringsmenyn öppna startas automatisk bandvågsmodulens serviceläge och totalräknare inaktiveras. Detta kan konstateras från blinkande lampan på bandvågsmodulen.

Från kalibreringsmenyn kan man välja mellan hastighetkalibrering med eller utan hastighetsgivare, nollpunktkalibrering, spankalibrering, materialtest inställning av processparametrar, återställa parametrar till tidigare sked eller fabriksåterställa bandvågsmodulen.



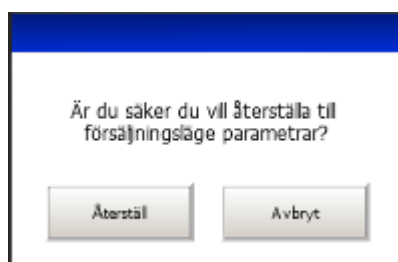
För en full kalibrering ska följande steg tas i ordningsföljd:

1. Inställning av processparametrar
2. En hastighetskalibrering
3. Nollpunktskalibrering
4. Spankalibrering
5. Materialtest

När processparametrar ska återställas kommer ett pop-up fönster dyka upp där systemet vill ha bekräftelse för åtgärden.

En normal återställning av parametrar kommer ändra alla parametrar till försäljningspunkten.

En full fabriksåterställning rekommenderas inte om SIWATOOL inte finns tillgängligt.



För att avsluta kalibrering ska den fysiska vippan stängas av och tillbakaknappen tryckas. Detta kommer byta fönster till huvudmeny, stoppa serviceläge och starta normalläge för

totalräknare. Det är rekommenderat att säkerställa att serviceläget stängts av vid avslutad kalibrering, genom att granska att ingen led blinkar på bandvågsmodulen.

6.5.1 Processparameter inställning

I processparameterfönstret kan viktiga parametrar för processens funktionalitet ändras. Till dessa hör enhets byten, decimal noggrannheter, bältets längd och vägningsområdets längd. Nuvarande värden läses automatiskt in när fönstret öppnas och för att spara ändringar måste sparaknappen tryckas. För att återvända till kalibreringsmenyn bör tillbakaknappen tryckas.

6.5.2 Hastighets kalibrering med givare

När hastighets kalibrering med givare väljs kommer en pop-up meny dyka upp, där man väljer om pulskonstanten är känd eller inte.

Om pulskonstante är känd kommer fönstret till höger dyka upp. För denna kalibrering ska de tre parametrarna i steg 1 fyllas i. Nuvarande värdet läses automatiskt in när fönstret öppnas och om ändringar görs måste sparaknappen tryckas.

I Steg två kan hastigheten testas genom att hålla in "manuell körning"-knappen och observera hastigheten i rutan under.

När kalibreringen är klar kan klar-knappen tryckas, detta kommer byta fönster tillbaka till kalibreringsmeny.

14.2.2022 15:51:52		Bältvågs prototyp		Hastighets kalibrering (känd pulskonstant)	
Steg 1			Steg 2		
Hastighets avkänning: <input type="text"/>			<input type="button" value="Manuell körning"/>		
Design hastighet: <input type="text" value="#,##"/>			Nuvarande hastighet: <input type="text" value="#,#"/>		
Hastighets sensor: (pulser per m) <input type="text" value="#,#"/>			Nuvarande hastighet: (% av design hastigh.) <input type="text" value="#,#"/>		
<input type="button" value="Spara"/>			<input type="button" value="Klar"/>		

För kalibrering med okänd pulskonstant ska först ett rotations antal fyllas i, om ändring på detta värde görs måste sparaknappen tryckas.

14.2.2022 15:52:11		Bältvågs prototyp		Hastighets kalibrering (okänd pulskonstant)	
Steg 1			Steg 2		
Antal rotationer för kalibrering: <input type="text" value="#"/>			Max hastighet (tom väg): <input type="text" value="#,#"/>		
<input type="button" value="Spara"/>			Pulser per meter: <input type="text" value="#,#"/>		
<input type="button" value="Starta pulsdetektion"/>			Pulser per varv: <input type="text" value="#,#"/>		
<input type="button" value="Stoppa pulsdetektion"/>			<input type="button" value="Godkänn Puls"/>		<input type="button" value="Godkänn Hastighet"/>
<input type="button" value="Läs in kalibreringsvärden"/>			<input type="button" value="Klar"/>		

Till näst ska följande steg följas:

Steg 1: När antalet är i fyllt ska bältet markeras med t.ex. tejp, för att kunna följa hur många vara som bältet snurrat. Välj även en fast punkt som markeringen passerar för att så noggrant som möjligt kunna bedöma antal varv.

Steg 1.2: Starta kalibrerings genom att trycka på "Starta pulsdetektion"

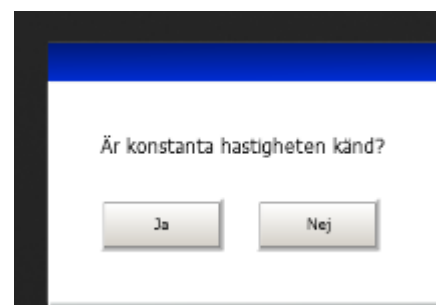
Steg 1.3: När bältet snurrat det definierade antalet vara (No. of belt revolutions for calibration) och markeringen passerar sista gången den fasta punkten, stoppa kalibreringen genom att trycka på "stoppa pulsdetektion".

Steg 2: Granska värden, om de ser ut att vara rimliga, godkänn både puls- och hastighetsvariabler genom respektive knapptryckning.

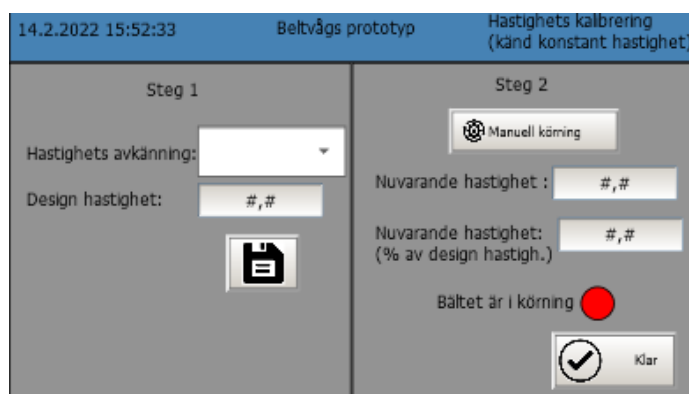
När kalibrering är klar kan klar-knapp påverkas, detta kommer byta fönster till kalibreringsmeny.

6.5.3 Hastighetskalibrering utan givare

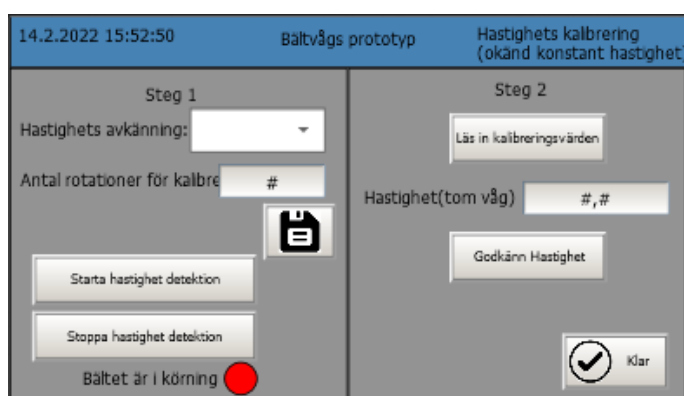
När hastighetskalibrering utan givare väljs kommer en pop-up meny dyka upp där man väljer om den konstanta hastigheten är känd eller inte.



Om den konstanta hastigheten är känd ska parametrar i steg 1 fyllas och sparas. Korrekt hastighet kan testas genom att hålla in "manuell körning"-knappen och granska så att hastigheten är rätt samt att bältet är i körning (grön märkning i fönstrets högra kant).



Om hastighetskonstanten är okänd kan följande steg följas för att kalibrera hastigheten.



Steg 1.1: Ställ DR3 "speed detection" till "No sensor – Preset/Detect constant speed" och välj antal rotationer för kalibrering. Om ändringar i parametrar görs måste sparnappen tryckas

Steg 1.2: Markera en punkt på bältet och en fast punkt som bältetmarkeringen passerar.

Steg 1.2: Tryck på "starta hastighets detektion".

Steg 1.3: Granska så att bältet är i körning, detta indikeras av märklampan i fönstrets nedre hörn. Lampan lyser grönt när bältet är i körning och rött när bältet inte är i körning.

Steg 1.4: När bältet roterat det bestämda antalet gånger (antal rotationer för kalibrering) stoppa kalibreringen genom att trycka på "Stoppa hastighets detektion".

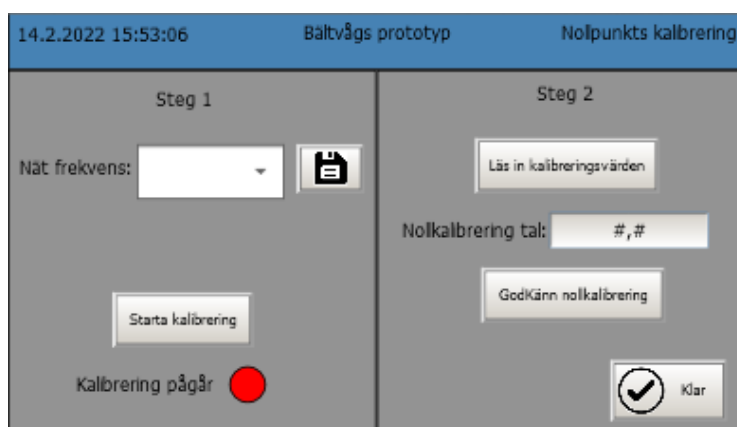
Steg 2: Läs in kalibreringsvärden genom att trycka på "Läs in kalibreringsvärden"

Steg 2.1: Om hastighet verkar rimlig, godkänn värdet genom att trycka på "Godkänn hastighet".

När kalibrering är klar tryck på klar-knapp som byter fönstret till kalibreringsmenyn.

6.5.4 Nollpunkts kalibrering

För att utföra nollpunktskalibrering ska följande steg följas.



Steg 1.1: Ange den lokala nätfrekvensen.

Steg 1.2: Spara värden, om ändring gjorts.

Steg 1.3: Starta kalibrering

Steg 2.1: När kalibrering kört klart, läs in kalibreringsvärden genom att trycka på knappen.

Steg 2.2: Om värdet verkar möjligt, godkännvärdet genom att trycka på "godkänn nollkalibrering".

När kalibreringen är klar tryck på klar-knappen för att komma tillbaka till kalibreringsmenyn.

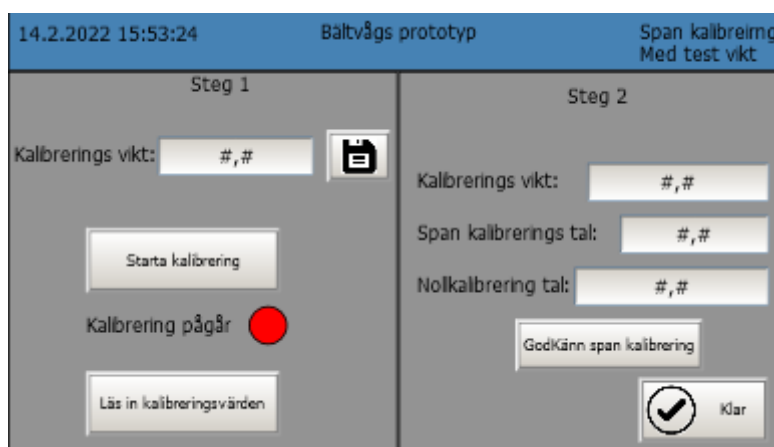
6.5.5 Span kalibrering

När spankalibrering väljs från kalibrerings menyn kommer ett pop-up fönster dyka upp, här kan användare välja mellan tre olika typer av span kalibrering.

- Testvikt
- Testkedja
- Lastcellsdata



För kalibrering med testvikt följ följande steg:



Steg 1.1: Ange kalibreringsvikt med den enhet som processen körs med (per default i kg)

Steg 1.2: Säkerställ att:

- 1: Nollpunktkalibrering körts
- 2: Bältet är uppvärmt och fungerar
- 3: Testvikter är monterade på vågen
- 4: Bältet är tomt
- 5: Serviceläge är aktivt

Steg 1.3: Påbörja kalibrering, genom att tryck på "starta kalibrering"

Steg 1.4: Säkerställ att kalibrering pågår, genom grön indikationslampa i fönstret.

Steg 1.5: När bältet stannar och "kalibrering pågår"-lampan blir röd, läs in kalibreringsvärden genom att trycka på "Läs in kalibreringsvärden".

Steg 2: Om kalibreringsvärde verkar rimliga, godkänn spankalibreringen genom knapptryckning.

När kalibreringen är klar tryck på klar-knappen för att komma tillbaka till kalibreringsmenyn.

För kalibrering med viktkedja följ följande steg:

14.2.2022 15:53:50 Bältvågs prototyp Span kalibrering
Med vikt kedja

Steg 1

Kalibrerings vikt: #,#

Starta kalibrering

Kalibrering pågår

Läs in kalibreringsvärden

Steg 2

Kalibrerings vikt: #,#

Span kalibrerings tal: #,#

Nollkalibrering tal: #,#

Godkänn span kalibrering

Klar

Steg 1.1: Ange kalibreringsvikt med den enhet som processen körs med (per default i kg)

Steg 1.2: Säkerställ att:

- 1: Nollpunktkalibrering körts
- 2: Bältet är uppvärmt och fungerar
- 3: Testkedjan är monterad och centrerad
- 4: Serviceläge är aktivt

Steg 1.3: Påbörja kalibrering, genom att tryck på "starta kalibrering"

Steg 1.4: Säkerställ att kalibrering pågår, genom grön indikations lampa i fönstret.

Steg 1.5: När bältet stannar och "kalibrering pågår"-lampan blir röd, läs in kalibreringsvärden genom att trycka på "Läs in kalibreringsvärden".

Steg 2: Om kalibreringsvärden verkar rimliga, godkänn spankalibrering genom knapptryckning.

När kalibreringen är klar tryck på klar-knappen för att komma tillbaka till kalibreringsmenyn.

För kalibrering med test vikt följa följande steg:

14.2.2022 15:54:16 Bältvågs prototyp Span kalibrering
Med last cells data

Steg 1

Antal last celler: #,#

Medle Karakteristiska värde: #,#

Nominella spänning för en last cell: #,#

Starta kalibrering

Kalibrering pågår

Klar

Steg 1.1: Ange följande parametrar

1: Antalet lastceller

2: "Averaged characteristic value (mV/V)" Detta värde hittas oftast på själva lastcellen eller i dess datablad, om inte ett exakt värde hittas kan ett ungefärligt värde anges. Om två eller flera lastceller används måste ett medeltal för alla lastceller räknas ut.

3: "Nominal load of one single load cell" enheten beror på "Unit for belt load"

Steg 1.2: Spara ändringar

Steg 1.3: Säkerställ att:

- 1: Nollpunktskalibrering körts
- 2: Bältet är tomt och fungerar
- 3: Ingen testvikt eller testkedja är monterad
- 4: Serviceläget är aktivt

Steg 1.4: Starta kalibrering genom knapptryckning. Säkerställa att kalibrering pågår, genom att indikationslampan blir grön.

Steg 1.5 vänta tills kalibrering avslutas och märklampan blir röd

När kalibreringen är klar överförs värden automatiskt till DR3. Det är rekommenderat att köra materialtestet genast efter denna kalibrering för att förbättra processens noggrannhet.

När kalibreringen är klar tryck på klar-knappen för att komma tillbaka till kalibreringsmenyn.

6.5.6 Material test

För att utföra materialtest följ följande steg:

The screenshot shows a software interface for a material test. At the top, it displays the date and time '14.2.2022 15:54:39', the device name 'Bältvågs prototyp', and the current mode 'Material Test'. The interface is divided into two main sections: 'Steg 1' and 'Steg 2'.
Steg 1: This section is for zeroing the scale. It features a button labeled 'Nollställ total räknare 6'. Below this, there are three input fields: 'Total räknare 6:' with a '#, #' format, 'Nu varande last (%):' with a '#, #' format, and two buttons at the bottom: 'Starta båndvåg' and 'Stoppa båndvåg'.
Steg 2: This section is for calculating correction factors. It starts with an input field for 'Genom körd vikt:' in '#, #' format, followed by a button 'Räkna ut korrektions faktor'. Below that is another input field for 'Korrektions faktor:' in '#, #' format. At the bottom, there are two buttons: 'Ange korrektions faktor 1' and 'Ange korrektions faktor 2'. A 'Klar' button with a checkmark icon is located at the bottom right of the interface.

Steg 1.1: Nollställ totalräknare 6

Steg 1.2: Startprocessen genom att trycka på "Starta bandvåg"

Steg 1.3: Mata konstant materialet in på bältet och kör det igenom processen.

Steg 1.4: När allt material körts igenom, stoppa processen genom att trycka på "stoppa bandvåg"-knappen.

Steg 1.5: Om materialet som körts genom processen inte vägts tidigare bör detta göras nu.

Steg 2.1: Ange materialets vikt i "genom körd vikt"-fönstret



Steg 2.2: Tryck på "räkna ut korrektionsfaktor"-knappen (om detta inte görs manuellt)

Steg 2.3: Tryck på ange korrektionsfaktor 1 eller 2 beroende på om två korrektionsfaktorer ska användas. Detta kommer öppna ett pop-up fönster.

Steg 2.4: I pop-up fönstret anges lastfaktorn, tidigare korrektionsfaktor och uträknad korrektionsfaktor

Steg 2.5 spara värden och stäng pop-up fönster

När kalibreringen är klar tryck på klar-knappen för att komma tillbaka till kalibreringsmenyn.

Uträknad KF	#, #
Last faktor 1 (& av nominell last)	#, #
Korrektions faktor 1	#, #
	 Tilbaka

7 Test sekvens

7.1 Start

<input type="checkbox"/>	Start gör inget
<input type="checkbox"/>	Stopp gör inget
<input type="checkbox"/>	Nödstopp gör inget
<input type="checkbox"/>	Kalibrering gör inget
<input type="checkbox"/>	Manuell/auto läge gör inget
<input type="checkbox"/>	Indikationer inga aktiva
<input type="checkbox"/>	Vid power-up börjar öppnas huvud disp.
<input type="checkbox"/>	Kalibrering och stopp aktiverar kalibrerings val
<input type="checkbox"/>	Språkval fungerar
<input type="checkbox"/>	Fönster. Byte till övriga skärmar fungerar

7.2 Körning (parameterinställnings skede)

<input type="checkbox"/>	Start gör byter till körnings disp.
<input type="checkbox"/>	Stopp gör inget
<input type="checkbox"/>	Nödstopp aktiverar alarm
<input type="checkbox"/>	Kalibrering aktiverar alarm
<input type="checkbox"/>	Manuell/auto läge gör inget men måste vara aktivt när start trycks
<input type="checkbox"/>	Indikationer klar för körning efter inmatade parametrar
<input type="checkbox"/>	Tillbaka fungerar
<input type="checkbox"/>	Manuell körning av BV fungerar
<input type="checkbox"/>	Manuell körning av M1 fungerar
<input type="checkbox"/>	Vikten visas korrekt
<input type="checkbox"/>	När modulen startas är alla parametrar noll

7.3 Körning (körnings skede)

<input type="checkbox"/>	Start gör inget före körning klar och ny körning inväntas
<input type="checkbox"/>	Stopp avbryter körning och byter fönster. Till parameterinställning
<input type="checkbox"/>	Nödstopp aktiverar alarm och avbryter körning
<input type="checkbox"/>	Kalibrering aktiverar alarm och avbryter körning
<input type="checkbox"/>	Manuell/auto om läge byts mitt i körning aktiveras alarm och körning avbryts
<input type="checkbox"/>	Indikationer körning pågår när i körning och klar för körning när ny körning inväntas
<input type="checkbox"/>	Pausa körning fungerar
<input type="checkbox"/>	Avbryt körning fungerar
<input type="checkbox"/>	Vikten visas rätt
<input type="checkbox"/>	Hastighet visas korrekt
<input type="checkbox"/>	Flöde visas korrekt
<input type="checkbox"/>	Start av Fin/pulsmatning fungera
<input type="checkbox"/>	Start av Grov matning fungerar
<input type="checkbox"/>	Start av BV matning fungerar
<input type="checkbox"/>	Byte mellan grov och fin/pulsmatning fungerar
<input type="checkbox"/>	Stopp av grov matning
<input type="checkbox"/>	Stopp av fin/pulsmatning fungerar
<input type="checkbox"/>	Stop av BV matare fungera
<input type="checkbox"/>	Genomkörd vikt stämmer
<input type="checkbox"/>	Start av ny körning i automat läge fungerar

7.4 Uppvärmning

<input type="checkbox"/>	Start påbörjar uppvärmning
<input type="checkbox"/>	Stopp avbryter uppvärmning
<input type="checkbox"/>	Nödstopp aktiverar alarm och avbryter uppvärmning

<input type="checkbox"/>	Kalibrering aktiverar alarm och avbryter uppvärmning
<input type="checkbox"/>	Manuell/auto läge gör inget
<input type="checkbox"/>	Indikationer klar för körning när BV är tom och start inväntas
<input type="checkbox"/>	Indikation i körning aktiv när uppvärmning aktiv
<input type="checkbox"/>	Avbryt öppnar pop upp
<input type="checkbox"/>	Pop upp Ja avbryter uppvärmning
<input type="checkbox"/>	Pop upp Nej stänger Pop upp
<input type="checkbox"/>	Tid inställning fungerar
<input type="checkbox"/>	Kvittering av tom BV fungerar
<input type="checkbox"/>	Uppvärmnings tid visas i indikations baren
<input type="checkbox"/>	Uppvärmningen avslutas när tiden är noll

7.5 Alarm

<input type="checkbox"/>	Start gör inget
<input type="checkbox"/>	Stopp gör inget
<input type="checkbox"/>	Nödstopp gör inget men startar alarm
<input type="checkbox"/>	Kalibrering gör inget men startar alarm
<input type="checkbox"/>	Manuell/auto gör inget
<input type="checkbox"/>	Indikationer om ett alarm aktivt indikeras fel
<input type="checkbox"/>	Alarm 1 fungerar (Man/Auto inte invalt och start påbörjad)
<input type="checkbox"/>	Alarm 2 fungerar (Parametrar inte inställda och start påbörjad)
<input type="checkbox"/>	Alarm 3 fungerar (Bandvågen inte tom och uppvärmning eller start påbörjad)
<input type="checkbox"/>	Alarm 4 fungerar (CMD error)
<input type="checkbox"/>	Alarm 5 fungerar (Fel från BV)
<input type="checkbox"/>	Alarm 6 fungerar (BV hastighets fel)
<input type="checkbox"/>	Alarm 7 fungerar ()

7.6 Kalibrering

<input type="checkbox"/>	Start gör inget
<input type="checkbox"/>	Stopp gör inget
<input type="checkbox"/>	Nödstopp alarmerar inte
<input type="checkbox"/>	Kalibrering alarmerar inte
<input type="checkbox"/>	Manuell/auto gör inget
<input type="checkbox"/>	Indikationer inga aktiva
<input type="checkbox"/>	Hastighets kalibrering för känd pulskonstant fungerar
<input type="checkbox"/>	Hastighets kalibrering för okänd pulskonstant fungerar
<input type="checkbox"/>	Hastighets kalibrering för konstant hastighet fungerar
<input type="checkbox"/>	Hastighets kalibrering för okänds hastighet fungerar
<input type="checkbox"/>	Viktkalibrering med test vikt fungerar
<input type="checkbox"/>	Vikt kalibrering med test kedja fungerar
<input type="checkbox"/>	Vikt kalibrering med lastcells data fungerar
<input type="checkbox"/>	Material test fungerar (med både en och två korrektions faktorinställningar)
<input type="checkbox"/>	Fabriksåterställning fungera
<input type="checkbox"/>	Återställning till försäljningspunkt fungerar
<input type="checkbox"/>	DR3 Parameter inställningar fungerar
<input type="checkbox"/>	Tillbaka fungerar
<input type="checkbox"/>	Service läge aktiveras och av aktiveras vid påbörjande och avslutande av kalibrering
<input type="checkbox"/>	Alla CMD kommandon fungera

7.7 Process information

<input type="checkbox"/>	Start gör inget
<input type="checkbox"/>	Stopp gör inget
<input type="checkbox"/>	Nödstopp alarm men gör inget

<input type="checkbox"/>	Kalibrering gör inget
<input type="checkbox"/>	Manuell/auto gör inget
<input type="checkbox"/>	Indikationer inga aktiva
<input type="checkbox"/>	Tillbaka fungerar
<input type="checkbox"/>	Total kördvikt för all tid är korrekt
<input type="checkbox"/>	Total kördvikt sedan power upp är korrekt
<input type="checkbox"/>	Tid sedan power-up är korrekt
<input type="checkbox"/>	Tid sedan ibruktagning är korrekt