

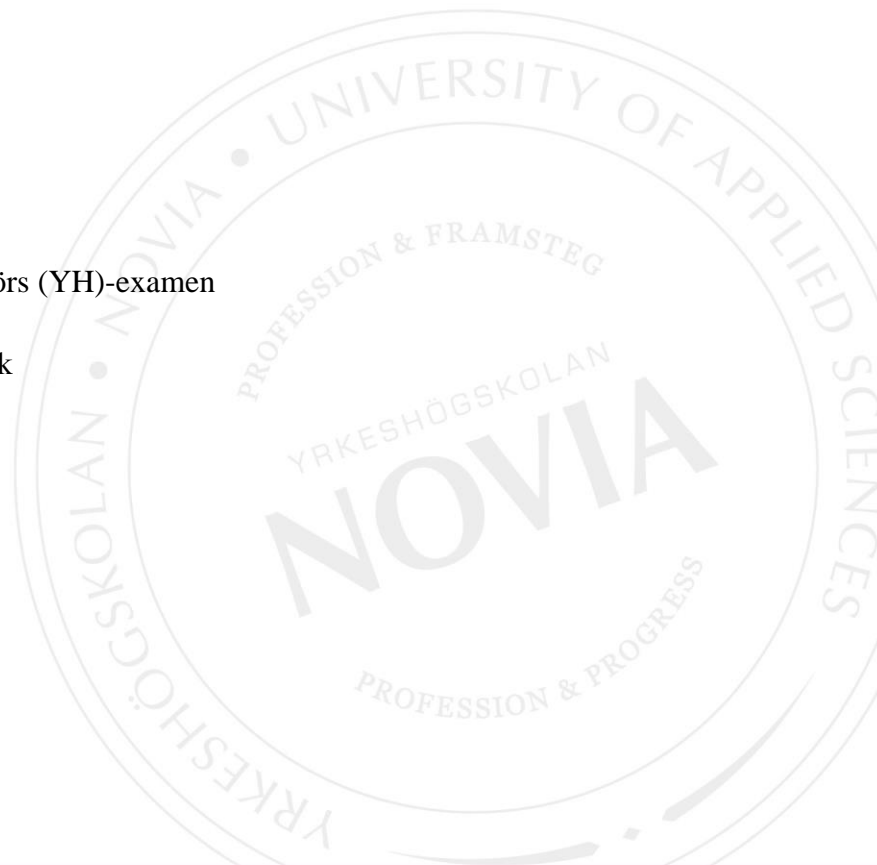
Testutrustning för verifiering av retardationstider

David Andersson

Examensarbete för ingenjör (YH)-examen

Utbildningen Elektroteknik

Vasa 2017



EXAMENSARBETE

Författare: David Andersson
Utbildning och ort: Elektroteknik, Vasa
Inriktningsalternativ: Automation
Handledare: Erik Englund

Titel: Testutrustning för verifiering av retardationstider

Datum: 24.4.2017 Sidantal: 40 Bilagor: 3

Abstrakt

Detta examensarbete behandlar hur man utvecklat testutrustning för uppmätning och verifiering av retardationstider för nyutvecklade maskiner vid företaget LKI Källdman. Maskinerna som tillverkas är plåthanteringsmaskiner inom metallindustrin.

Skyddsutrustningen som finns runt maskinen består bland annat av en ljusridå. Standarden ISO 13855:2010 innehåller olika ekvationer för att räkna ut var ljusridån skall placeras i förhållande till maskinen. En variabel i dessa ekvationer är tiden det tar för maskinen att komma till ett viloläge efter att ljusridån har blivit påverkad vid maxhastighetskörning, alltså retardationstiden.

Arbetet som har gjorts består av programmering av en programmerbar styrenhet (PLC), samt utveckling ett grafiskt användargränssnitt för en lätt interaktion med testutrustningen. Det grafiska användargränssnittet interagerar man med hjälp av t.ex. en bärbar dator. Med testutrustningen verifieras LKI Källdmans maskiners retardationstider med de förinställda inställningarna för maskinen. Av de utförda mätningarna skrivs ett testprotokoll med mätresultat, beräkningar för placering av ljusridån enligt ISO 13855:2010.

Slutligen programmerades testutrustningen och en användarmanual om användargränssnittet skrevs. Två av LKI Källdmans maskiner testades med testutrustningen. I samband med testen skrevs testprotokoll.

Språk: svenska Nyckelord: testutrustning, retardationstid, ljusridå

OPINNÄYTETYÖ

Tekijä: David Andersson
Koulutus ja paikkakunta: Sähkötekniikka, Vasa
Suuntautumisvaihtoehto: Automaatiotekniikka
Ohjaaja: Erik Englund

Nimike: Testauslaite hidastuvuusaikojen todentamiseen

Päivämäärä: 24.4.2017 Sivumäärä:40 Liitteet:3

Tiivistelmä

Tässä opinnäytetyössä kuvataan testauslaitteen kehittämistä. Testauslaite mittaa ja todentaa hidastuvuusaikoja vasta kehittyneille koneille LKI Kälđmanilla. Valmistettavat koneet ovat erilaisia metalliteollisuuden peltikäsittelykoneita.

Konetta ympäröivät turvalaitteet koostuvat muun muassa valoverhoista. ISO-standardi 13855:2010 sisältää erilaisia yhtälöitä, jotka laskevat mihin valoverho tulee sijoittaa koneeseen nähden. Yksi muuttuja näissä yhtälöissä on aika, joka kuluu koneen tullessa lepotilaan valoverhon ensin ollessa vaikutettuna maksiminopeusajolla, eli hidastuvuusaika.

Tehty työ koostuu ohjelmoitavan ohjaimen (PLC) ohjelmoinnista sekä graafisen käyttöliittymän kehittämisestä. Käyttöliittymällä voidaan helposti olla vuorovaikutuksessa testilaitteen kanssa. Graafista käyttöliittymää käytetään esimerkiksi kannettavan tietokoneen avulla. Testilaitteella todennetaan LKI Kälđmanin koneiden hidastuvuusaikoja koneiden oletusasetuksien kanssa. Suoritetuista mittauksista kirjoitetaan testiraportti, joka koostuu hidastuvuusaikojen mittauksien tuloksista ja valoverhon asettamisen laskelmista ISO 13855:2010-standardin mukaan.

Lopuksi ohjelmoitiin testilaite ja käyttöliittymästä kirjoitettiin käyttöohje. Kahta LKI Kälđmanin konetta testattiin testilaitteella. Testauksien yhteydessä kirjoitettiin testiraportti.

Kieli: ruotsi

Avainsanat: testauslaite, hidastuvuusaika, valoverho

BACHELOR'S THESIS

Author: David Andersson
Degree Programme: Electrical Engineering, Vaasa
Specialization: Automation
Supervisor: Erik Englund

Title: Test Equipment for Verifying Deacceleration Time

Date: April 24, 2017

Number of pages: 40

Appendices: 3

Abstract

This thesis involves development of test equipment for measuring and verifying the deacceleration time for newly developed machines at LKI Källdman. The produced machines are types of sheet metal handling machines for the metal industry.

The safety equipment surrounding the machine comprises among other things light curtains. The standard ISO 13855:2010 comprises equations to calculate where light curtains should be placed relative to the machine. One of the variables in these equations is the time it takes the machine to come to rest, after the light curtain has been actuated during the time the machine is running on maximum speed.

The work that has been done is programming a programmable logic controller (PLC) and developing a graphical user interface for simple interaction with the test equipment. The graphical user interface is interacted with for example a laptop. The test equipment is used to verify LKI Källdmans machines' deacceleration time with the machines' preset parameters. Of the preformed measurements, a test protocol is written comprising measurement results and calculations for placing the light curtain according to ISO 13855:2010.

Lastly the test equipment was programmed and a user manual for the user interface was written. Two of LKI Källdmans machines were tested with the test equipment. In conjunction with the tests, test protocols were written.

Language: swedish

Key words: test equipment, deacceleration time, light curtain

Innehållsförteckning

1	Introduktion	1
1.1	Uppgiftsbeskrivning	2
1.1.1	Problembeskrivning	2
1.1.2	Mål	2
1.1.3	Delmål	2
1.2	Uppdragsgivaren	3
1.3	Tidigare utveckling	7
1.3.1	ABB/Jokab safety	7
1.3.2	Tidigare gjorda testprotokoll av LKI:s maskiner	7
1.3.3	Testutrustning vid KWH-Mirka	7
1.4	Avgränsningar	8
2	Maskinsäkerhet 13855:2010	9
2.1	Säkerhetsavstånd	10
2.2	Metodik för att bestämma korrekt position av säkerhetsvakt	12
2.3	Uppmätning av retardationstiden för en maskin	13
2.4	Beräkning av det minsta avståndet	14
2.5	Testprotokoll	15
3	Användbarheten av ett gränssnitt	16
4	Maskinvara	18
4.1	Programmerbar styrenhet (PLC)	18
4.2	Vajermekanism	19
4.3	Pulsgivare	19
5	Programvara	22
5.1	Automation Studio	22
5.2	Programmeringsspråk	22
6	Datakommunikation	24
6.1	Synchronous Serial Interface (SSI)	24
6.2	USB-överföring	25
6.3	Virtual Network Computing (VNC)	27
7	Metoder och tillvägagångssätt	28
7.1	Användning av absolutgivare	28
7.1.1	Konfigurering av absolutgivare	28
7.1.2	Val av mätenhet	28
7.1.3	Integer overflow	30
7.2	Hantering av data	30
7.2.1	Retardationstidsmätning av programmet	31

7.2.2	Loggning av data	31
7.2.3	Statistik beräkning	33
7.2.4	Förberedelse av data till användargränssnittet	33
7.3	Användargränssnitt.....	34
7.4	Spara skärmbilder.....	35
7.5	Praktisk tillställning för retardationstidsmätning av en maskin med testutrustningen.....	36
7.5.1	Fastsättning av vajermekanismen	36
7.5.2	Val av stoppsignal.....	37
7.5.3	Parameter justering för frekvensomriktare.....	38
7.5.4	Retardationstidsmätning.....	39
8	Resultat	40
9	Diskussion	41
10	Källförteckning	43

Bilaga 1.	Användarmanual för testutrustningen
Bilaga 2	Testprotokoll av LIII
Bilaga 3	Testprotokoll av ASF-EU 2 nd station

Ordförklaringar

Absolutgivare	En typ av en pulsgivare som kan mäta en exakt position.
ASF-EU	Advanced Storage for Flying Optics for European market. En av LKI källdmans maskiner.
B&R	Bernecker + Rainer Industrie-Elektronik. Ett österrikiskt företag.
Detekteringszon	Det område framför en maskin där säkerhetsutrustning som t.ex. en ljusridå placeras.
Digital ingång	En logisk signal som kopplas till PLC:n som antingen är på eller av.
Ethernet	Den vanliga internet kabeln som kopplas mellan en dator och ett modem är en Ethernetkabel.
Farozon	Det farliga området på en maskin.
Frekvensomriktare	En apparat som omvandlar frekvensen till motorer, vilket leder till att man kan reglera dess rotationshastighet.
LIII	Loader 3. En av LKI Källdmans maskiner.
Ljusridå	Ett optiskt staket av laserstrålar mellan två stolpar som kan placeras framför en farlig maskin. Ifall någon/något passerar genom stolparna så ges ett alarm till styrsystemet.
Palett	Ett hyllplan på LKI Källdmans maskiner som det placeras plåtar på, och som kan transporteras av maskinerna.
PLC	Programmable logic controller. Det är en programmerbar styrenhet som håller koll på automationen i en process.
Pulsgivare	En grupp av givare som kan läsa av roterande axlar.
Resolution	En förklaring på hur noggrann en pulsgivare är.
Retardationstid	Den tid det tar för maskinen att komma till ett viloläge, efter att en stoppsignal har blivit given.
Säkerhetsrelä	En komponent som säkerställer att man säkert kan ge en stoppsignal till en maskin.
Säkerhetsvakt	En grupp av säkerhetsanordningar som kan placeras runt en maskin.
SSI	Synchronous Serial Interface. Ett protokoll som kommunicerar mellan absolutgivaren och PLC:n.
VNCviewer	En mjukvara till en dator som möjliggör att man kan interagera med PLC:ns användargränssnitt.

1 Introduktion

Automatiserade maskiner ute i industrin kan ha rörelser av laster som vid felhantering eventuellt kan leda till människo-, material- eller miljöskador. P.g.a. detta bör en riskanalys göras för att bestämma möjliga faror för en maskin, och senare bestämma hur man kan motverka detta. Det lättaste sättet är att placera skyddsgaller med dörrar som har givare, runt det farliga området, som kan alarmera om en dörr är öppen, vilket signalerar att stanna maskinen. Ifall detta inte är en praktisk lösning för en specifik maskin så kan man även besluta sig för att använda s.k. ljusridåer. Ljusridån är som ett osynligt staket bestående av ett visst antal laserstrålar mellan två stolpar. Ifall någon av dessa laserstrålar bryts av att någon eller något passerar genom ljusridån, så ges ett alarm vilket i sin tur stannar maskinen.

Fördelen med att använda sig av en ljusridå är att man har en relativ fri tillgång till maskinlinjen utan dörr. Nackdelar med ljusridån är däremot att ett ofrivilligt stopp kan inträffa ifall någon i misstag går igenom strålarna och ljusridån tar oftast mer utrymme än vanliga staket. En annan nackdel är t.ex. ifall en tung plåt lyfts upp med lyftok, så kan det finnas en väsentlig klämrisk med en potentiell dödlig utgång. Plåten lyfts upp och i det läget man går genom ljusridån, så är det inte helt säkert plåten hålls kvar i lyftoket. Till en sådan maskin kan man inte använda en ljusridå. Vid dessa tillämpningar måste tillträde till maskin vara förhindrad med annan utrustning än en ljusridå, t.ex. med staket med låsande grindar.

Enligt den internationella standarden ”*ISO 13855:2010: Placering av tekniska skydd beroende på kroppsdelars hastigheter*” så skall ljusridåer placeras på ett beräknat avstånd från en maskin beroende på några faktorer. Ena faktorn för denna beräkning är tiden det tar för en maskin att komma till ett viloläge, efter att en stoppsignal har blivit given vid maxhastighets körning av maskinen.

Till examensarbetet konstrueras en testutrustning för att mäta upp denna tid och verifiera dessa tider med maskinens förinställda inställningar. Stoppsignalen till detta test kan antingen ges genom att påverka nödstoppet eller ljusridån. Med de uppmätta verifierade tiderna kan ett avstånd beräknas med hjälp av ekvationer från *ISO 13855:2010*, vilket beskriver det minsta avståndet ljusridån skall placeras från maskinen.

Av dessa mätningar skall ett testprotokoll skrivas där det framgår vilken slags utrustning som har blivit använd, hur man har gått tillväga, mätresultat från utförda test, beräkningar

av säkerhetsavstånd enligt ISO 13855:2010 samt bör det även bifogas grafer som beskriver retardationen av hastigheten för maskinen under retardationstidsmätningen.

1.1 Uppgiftsbeskrivning

Under denna rubrik beskrivs uppdragsgivarens bakgrund till behovet av examensarbetet, önskade mål samt delmål av examensarbetet. Vid uppdragsgivaren LKI Källdman så var Patrik Sjöling min handledare under examensarbetets gång.

1.1.1 Problembeskrivning

Skyddsutrustning som stannar maskiner på ett säkert sätt krävs enligt maskindirektivet för att uppfylla CE-krav. Det kan gälla dörrar, galler, ljusridån mm. Det finns standarder som beskriver hur man ska gå tillväga för att uppfylla detta. Retardationstider behöver förstås beräknas men också verifieras och dokumenteras i den tekniska dokumentationen för maskinen. Enligt maskindirektivet skall dokumentationen finnas tillgänglig för myndigheter 10år efter en maskins serieproduktion avslutats.

1.1.2 Mål

Huvudmålet för arbetet är att utveckla rutiner för att verifiera och dokumentera retardationstider för uppdragsgivarens maskiner. En komplett testutrustning med dokumentation för användning bör framtagas. Maskinrörelser bör identifieras för att bestämma vilket sätt man kan mäta retardationstider på den nya maskinen ASF-EU.

1.1.3 Delmål

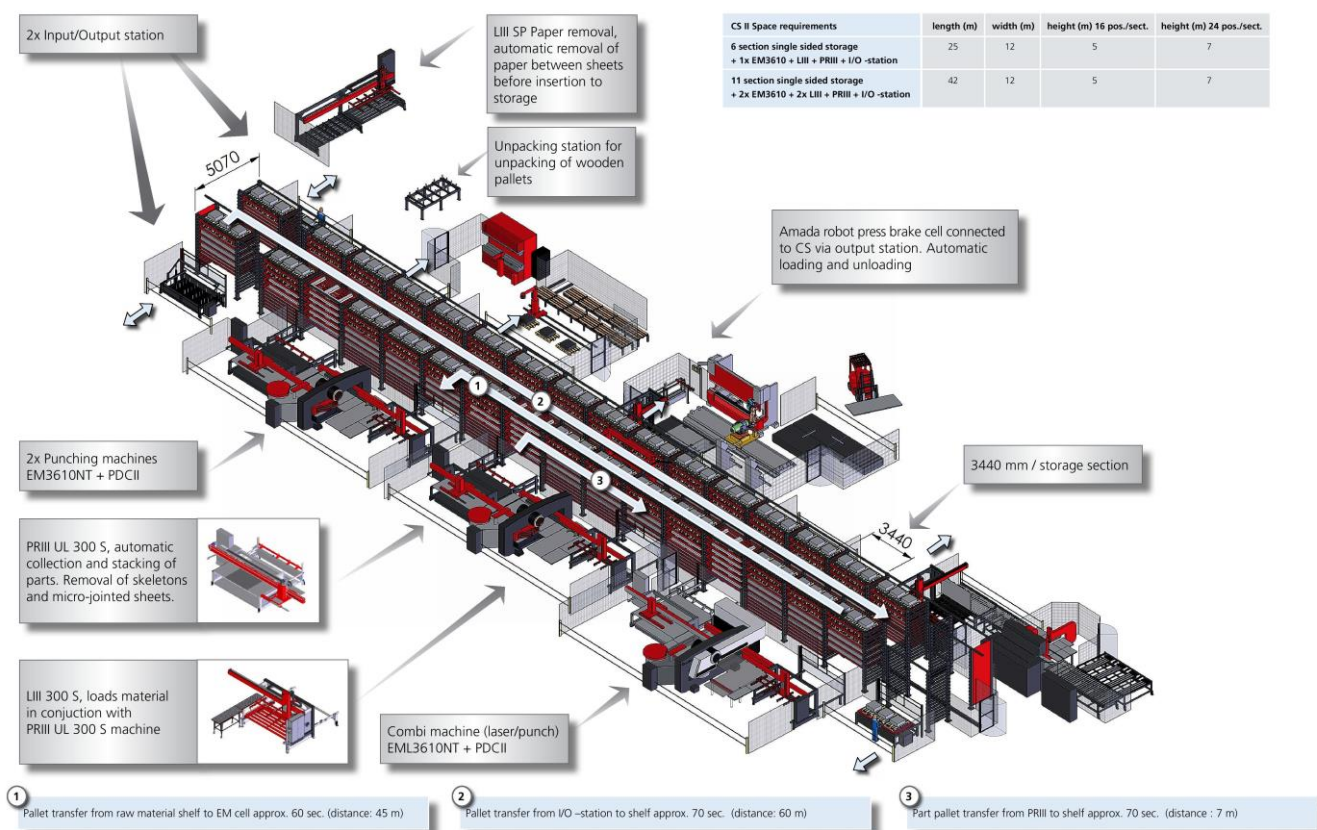
Utvecklingsområden för maskinsäkerhet; vilka utvecklingsmöjligheter som kan passa för uppdragsgivarens maskiner:

- Frekvensomvandlare med STO (Safe Torque off).
- Tätare modell av ljus ridå (flera strålar; möjligt att placera närmare riskällan).
- Safety scanners.
- Några nya produkter på kommande till marknaden.

1.2 Uppdragsgivaren

Uppdragsgivaren till examensarbetet är LKI Källdman. I dagens läge utvecklas och produceras lösningar för automatiserad hantering av metallplåtar, flexibla tillverkningsystem samt intralogistik som är optimering, automatisering, integrering och hantering av logistik och materiella varor. Mer än 90 % av tillverkningen exporteras, främst till Europa och Nord Amerika. Företaget har cirka 180 anställda.

Ett exempel på ett flexibelt tillverkningsystem kan man se i Figur 1, där de långa rälsen i mitten numrerat "2" är LKI:s "Compact Storage (CSII)" kran, som har hyllor på båda sidorna om sig. Runtomkring detta finns LKI:s "Loader (LIII)" och "Part Remover (PRIII)". När Amadas laser/stans behöver en ny specifik plåt så hämtar CSII-kranen en palett från ett specifikt hyllplan i lagret och transporterar hela paletten vidare till LIII. När paletten har anlänt till LIII så lyfts en plåtskiva åt gången med hjälp av sugproppar och förflyttas in till Amadas laser/stans. Efter att lasern/stansen är klar så plockar och sorterar PRIII de urskurna metallbitarna och det s.k. skelettet med hjälp av sugproppar till paletter. När allt är klart utfört av Amadas maskin så hämtar CSII-kranen paletterna och transporter dem till olika hyllplan i lagret. Från LKI:s input/output(IO) stationerna kan man beställa paletter för att antingen ta ut eller lägga in plåtar till lagret.



Figur 1 Ett exempel på ett flexibelt tillverkningssystem.

Företaget grundas 1979 av Leif Källdman i Lövö, Österbotten och han är fortfarande majoritetsägare till 80 % av aktierna av företaget, medan Amada som är ett japanskt företag med många maskiner inom metallindustrin köpte 20 % av LKI Källdmans aktier år 2009. I dagens läge finns verksamheten kvar i Lövö där det startade, bestående av en produktionshall (A-hall) samt kontor. I grannbyn Bennäs 2km bort finns företagets huvudkontor i kombination med två produktionshallar, en logistik hall och ett måleri. Under våren 2017 startades ett byggprojekt för att utvidga monteringshallen i Bennäs.

Den första hallen som byggdes i Bennäs så var svetshallen (C-hall) som byggdes 2001. Fort därefter krävdes mer monteringsutrymme så en monteringshall (B-hall) byggdes 2005 bredvid C-hallen. 2007 expanderades C-hallen med $3500m^2$ och i samband med detta anslöts C-hallen även ihop med B-hallen för att bilda en logistikhall (L-hall). Under en kort period mellan 2011–2013 så fanns ingen produktion i A-hallen i Lövö utan under denna period var hela produktionen i Bennäs. I dagens läge jobbar cirka 20 personer på produktionen i Lövös A-hall.

Ett programvaruföretag vid namnet Camline i Villmanstrand köptes upp 2012 av LKI Källdman. Personalen där består av 13 personer och företaget fokuserar på lösningar för att förbättra effektiviteten inom industrin.

Från en början 1979 fungerade företaget som underleverantör av svarvning och fräsning. Fem år senare år 1984 tillverkades den första egna produkten ”*Poscontrol*” som var ett programmerbart s.k. bakre anslag av en kantpress. I Figur 2 ser man hur denna produkt såg ut. Ett bakre anslag är en utrustning som placerar en plåt på specifikt önskad position för att senare bocka plåten. Det första automatiska systemet för plåthantering till metallskärning levererades till ABB 1990. Året efter 1991 utvecklades den första in- och utmatnings system av plåtar för stanspressar.



Figur 2 Den programmerbara ”*Poscontrol*” som var den första egna produkten som LKI Källdman tillverkade 1984.

Under åren 1994–1995 utvecklas den första plåthanteringsmaskinen ”*200 manipulator (MP)*” som var kompatibel med Amadas stanspress. Idag säljs inte 200MP mera, men dess efterföljare 300MP, som kan hantera större plåtar säljs fortfarande årligen. Den första versionen av en ”*Part Remover (PR)*” en med unik design av sugproppar som plockar utskurna delar från Amadas stanspress utvecklades 1997.

Det första systemet med lagringsförmåga av plåtar samt försörjning av plåtar till Amadas utrustning utvecklades 2001 som heter ”*ASLUL (Automatic storage with load and unloading)*”. För att få lite större möjlighet av plåtförvaring så utvecklades en ”*Compact storage (CS300)*” kran år 2004 med möjligheten av upp till ett 50 meter långt lager.

2012 utvecklades en ”*LST (Laser Shuttle Table)*” som har två stycken paletter som kan köras in till Amadas laser. Detta betyder då att man kan effektivera laserns dödtid genom att när

en palett är inne i Amadas laser, så kan redan nästa palett förberedas med en ny plåt. (LKI Käldman, 2017)

För att få effektiviserat produktionen av standardlösningar till Amadas laserskärare, så utvecklades ”ASF-EU (*Advanced Storage for Flying Optics for European market*)” år 2014 som är kompatibel med Amadas FO MII NT optiska laser skärare. Detta är modulär maskin vilket betyder att utrymmet i produktionen kan optimeras under tillverkningen. Den består av en kran som kan lyfta plåtar till dess övre del där det finns paletter med lagring av plåtar. I samband med ASF-EU så används även LST för att mata in plåtarna till Amadas laserskärare. Slutligen så finns en modul som förflyttar plåtar mellan kranen och LST paletterna. I Figur 3 ser man hur den ser ut. (Björklund, 2015)



Figur 3 En av de nyastes maskinerna ASF-EU.

1.3 Tidigare utveckling

Här presenteras tidigare utvecklingar som har behandlat samma krav och lösningar som detta examensarbete utgår ifrån.

1.3.1 ABB/Jokab safety

ABB/Jokab safety har en utrustning ”*Smart safety and motion analyser tool*” som är utvecklad specifikt för att mäta maskiners rörelser och tider. Denna produkt tillverkas inte längre, men produktkataloger finns fortfarande kvar på ABBs hemsida. Från denna produktkatalog fås ingen referenskunskap som skulle gynna utvecklingen av detta examensarbete, utan där beskrivs enbart vilka komponenter som fanns av ”*Smart safety and motion analyser tool*” sortimentet.

1.3.2 Tidigare gjorda testprotokoll av LKI:s maskiner

Som referens finns tre tidigare gjorda testprotokoll för verifiering av retardationstid. Det första testprotokollet är från ett test av en ”*LST-MPFLEX*” maskin, utförd av Amada den 26.2.2014 i Frankrike. Det andra testprotokollet är från ett test av en ”*FBS*” maskin, utförd vid LKI Källdman den 19.8.2016. Det tredje testprotokollet är från ett test av en ”*ASF-EU 2nd station*” maskin, utförd vid LKI Källdman den 6.10.2016.

Metodiken för mätningarna från föregående styckes två sistnämnda testprotokoll utförda vid LKI Källdman blev gjorda med hjälp av programvaran Automation Studio och dess diagnostiserings miljö. Problemet med att utföra mätningarna med denna programvara, är att det krävs att en programmerare med kunskap om programvaran är på plats. Målet med examensarbetet är att utveckla en testutrustning som är lätt att använda för vem som helst.

1.3.3 Testutrustning vid KWH-Mirka

Enligt uppgift, så har KWH-Mirka i Jeppo Österbotten tidigare utvecklat liknande testutrustning för maskiner som arbetsgivaren har som mål av detta examensarbete. Det skulle vara väldigt lämpligt att ha en färdig utvecklad testutrustning för att jämföra mätresultaten med den utrustning som blir utvecklad till detta examensarbete. Genom att jämföra dessa två mätvärden skulle man kunna bestämma om de är trovärdiga beroende på hur likadana resultat man får.

Enligt Kari Lund (diskussion via telefon 2.3.2017) som är el ansvarig vid KWH-Mirka i Jeppo, så har de inte kvar testutrustningen, eftersom de inte haft användning av den på några år, så nödvändiga komponenter plockats till projekt där det var av mer användning. P.g.a. detta finns ingen möjlighet att göra s.k. benchmarkings av testutrustningarna för jämförelse.

1.4 Avgränsningar

Till examensarbetet blev det planerat vilka komponenter som skulle behövas, samt att programmera en PLC med ett tillhörande användargränssnitt att interagera med. Dessa komponenter placeras i en låda och till detta väljs en passligt lång kabel som kopplas till absolutgivaren. Under skrivandet av examensarbetet så är programmeringen klar, men det har ännu inte blivit gjort någon låda.

2 Maskinsäkerhet 13855:2010

Maskinfaror är orsaker till potentiella farliga situationer. Farorna kan orsaka mänskliga skador och även förstöra utrustning eller omgivningen. Risk är en kombination av sannolikhet och konsekvens av skadliga effekter av en angiven farlig händelse. Vid maskinplanering så beskrivs risk som sannolikheten till den grad av eventuell skada eller hälsoskador vid en farlig situation. Maskiner skall konstrueras enligt de grundläggande säkerhets- och hälsokraven samt bör tillverkaren hela tiden följa med tekniska utvecklingar som kan förbättra maskinsäkerheten. Säkerhet är maskinens förmåga att fullfölja sin funktion utan att orsaka skador till människor. Maskinen är säker ifall riskerna för maskinen är bedömd att vara acceptabla.

Målet med en riskbedömning är att få ut information om maskinfara, för att skapa en specifikation för en säkerhetskonstruktion. Analysen av de juridiska kraven görs genom att identifiering av en fara och de möjliga olycksscenarierna av liknade existerande maskiner. Det finns olika metoder för att identifiera faror. I ett tidigt utvecklingskede kan man undersöka olyckor och incidenter som har inträffat förut. Dessa metoder kallas för Preliminär Riskanalys (PHA) och Potential Problemanalys (PPA). Ifall man behöver en mer detaljerad riskbedömning av funktioner, moduler och komponenter så kan man tillämpa metoderna Felfunktion- och Effektanalys (FMEA) och Felträdsanalys (FTA). Till processindustrin är det vanligt att det används metoden Risk- och Tillförlitlighetsanalys (HAZOP), men den kan även användas för att utveckla robusta maskiner. Förkortningarna av dessa metoder kommer från dess engelska namn. (Kivistö-Rahnasto, 2000)

Av dessa olika metoder så finns det färdiga formulär som beskriver på vilka sätt farorna skall identifieras, och efteråt ger man en bedömning hur stor farorisen är. Riskbedömning beskrivs i ISO 14121-1 och består i grova tag av fyra saker:

- Specificera kraven för korrekt användning av maskinen.
- Identifiering av möjliga faror samt farliga situationer.
- Riskuppskattning av varje identifierad fara samt farliga situation som kan orsakas av förutspådd felbehandling av maskinen eller av operatören.
- Utvärdera riskbedömningarna till om den är för hög eller inte. Om man utvärderar att sänka på risken bör man implementera fler eller bättre säkerhetsutrustningar. (Leuze electronic, 2017)

Ifall man från riskbedömningen bedömt att implementera ljusridåer eller laseravläsare så specificeras det var och hur de skall placeras i ISO 13855:2010.

Examensarbetet baserar sig på standarden ISO 13855:2010: Placering av tekniska skydd beroende på kroppsdelars hastigheter. Säkerheten för maskiner delas enligt ISO standarden in i tre stycken typer: Typ-A, Typ-B och Typ-C.

Typ-A är en fundamental säkerhetsstandard som gäller alla maskiner, dessa är grundläggande begrepp och principer för planering.

Typ-B är indelad två subkategorier Typ-B1 och Typ-B2. Typ-B1 är specifika säkerhetsåtgärder som t.ex. säkerhetsavstånd, ljudnivåer eller yt-temperaturer. Typ-B2 är implementering av s.k. ”*safeguard detectors*”, som tvåhandsgrepp, låsande grindar, tryckkänsliga apparater, ljusridåer, laseravläsare osv. Dessa ”*safeguard detectors*” refereras som säkerhetsvakter i detta examensarbete.

Platsen där de tre sistnämnda säkerhetsvakterna blir placerade kallas för en detekteringszon, eftersom att de detekterar att en människa antingen passerar förbi zonen, eller att man sträcker sig genom den.

Den tredje och sista Typ-C är maskinspecifika. Dessa är väldigt specifika och gäller enbart en viss maskin eller en grupp av maskiner.

ISO standarden 13855:2010 som examensarbetet utgår ifrån är för Typ-B2 standard. För att dessa standarder skall uppfyllas korrekt krävs att säkerhetsvakterna blir korrekt placerad i förhållande till farozonen på maskinen. För att bestämma dess positioner så finns det en del faktorer som påverkar detta: Tiden det tar för en maskin att komma till ett viloläge efter given stoppsignal, hur långt man kan sträcka sin arm genom eller över säkerhetsvakten mot farozonen, möjligheten av befintlighet av en person mellan säkerhetsvakten och farozonen, icke detekterad åtkomst till farozonen. (ISO 13855:2010, 2010)

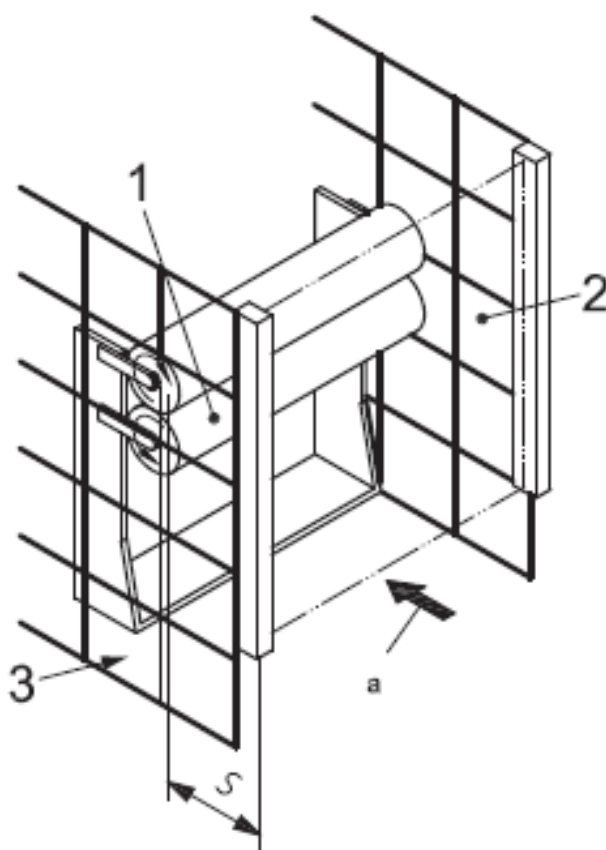
2.1 Säkerhetsavstånd

Från ISO 13855:2010 beskrivs hur man kan kalkylera ut det minsta avstånden mellan farozonerna på maskinen och detekteringszonen där säkerhetsvakten finns. Detta räknas ut i förhållandet till hastigheten av människans kroppsdelar. Värdena som ges för hastigheterna i denna internationella standard är gånghastighet och överkroppens rörelser som är

framtagna och testade i industriell miljö. Annat som spring, fall eller hopp hastigheter tas inte upp i denna internationella standard.

Säkerhetsvakter som täcks av denna internationella standard är ljusridåer, separata laserdetekteringsstrålar, lasersläsare, tryckkänsligskyddsutrustning, tvåhandsmanövrering och sammanlåsande skyddsutrustning. Till examensarbetet behandlas enbart ljusridåerna.

I Figur 4 är detekteringszonen märkt "2", vilket är ljusridåns två stolpar som är fäst på änden av skyddsgallren märkt "3" som omringrar maskinen. Maskinen är alltså farozonen som är märkt "1". Från detekteringszonen till farozonen är det beräkningsbara minsta avståndet, S . Vid beräkning av detta minsta avstånd så tar man i beaktande att föremål eller personer närmar sig detekteringszonen vinkelrätt, alltså riktningen "a".



Figur 4 Bilden beskrivs sträckan mellan farozonen och detekteringszonen.

Avstånden som blir kalkylerade från denna internationella standard gäller enbart ifall säkerhetsvakterna är fastmonterade, så de inte lätt kan tillfälligt flyttas. Vid användning av t.ex. tvåhandsmanövrering så skall säkerhetsvakten inte vara möjlig att flytta närmare farozonen utan verktyg. (ISO 13855:2010, 2010)

2.2 Metodik för att bestämma korrekt position av säkerhetsvakt

Ett exempel på hur man går tillväga när man har en maskin att identifiera och vill välja korrekt skyddsutrustning följer här:

- Identifiera farorna och bedöm riskerna för alla situationer av maskinen enligt ISO 12100-1 och ISO 14121-1.
- Ifall en Typ-C standard existerar för maskinen, så skall säkerhetsvakterna för den maskin specifika standarden användas och avstånden beräknas enligt Typ-B2 standarden.
- Ifall det inte finns någon Typ-C standard, så beräknas avstånden enligt Typ-B2 standarden för den säkerhetsvakt som man valt att använda enligt riskutvärderingen.
- Ifall det finns möjlighet till att man kan undvika säkerhetsvakten genom att gå runt den, så bör tillägg beräkningar utföras utöver den vanliga beräkningen. Av dessa två beräkningar så väljs det säkrare avståndet, alltså det längre avståndet.
- Ifall man i riskutvärderingen beslutit att använda flera säkerhetsvakter, så skall avstånd beräknas till varje individuell säkerhetsvakt.
- Varje möjliga minsta avstånd mellan farozon och detekteringszon skall beräknas, och av dem väljs det längsta avståndet.
- Kontrollera om den beräknade positionen av säkerhetsvakten kan leda till att en person kan förbli odetekterad mellan farozonen och detekteringszonen. Ifall en person kan förbli odetekterad, bör man göra en ny riskbedömning ifall tilläggsåtgärder behövs.

Ett exempel på en tilläggsåtgärd är att man placerar en återställningsknapp utanför området mellan farozonen och detekteringszonen. Positionen där man väljer att placera denna knapp bör ha fri sikt över hela området mellan farozonen och detekteringszonen, så att operatören som kvitterar återställningen kan göra en pålitlig bedömning att ingen längre finns kvar i detta område. (ISO 13855:2010, 2010)

2.3 Uppmätning av retardationstiden för en maskin

För att kalkylera avstånden mellan farozonen och detekteringszonen med de ekvationer som finns i ISO 13855:2010, så bör man mäta tiden från att detekteringszonen har blivit påverkad, till att de rörliga delarna vid farozonen har kommit till ett vilotillstånd. Denna retardationstid kan delas i två delar enligt ekvation (1). T_1 är tiden från att detekteringszonen har blivit påverkad, till att en signal ges att stanna maskinen. T_2 är den tid det tar för maskinen att komma till ett viloläge efter att den fått stoppsignalen vid maxhastighetskörning. T är den totala tiden i sekunder man skall använda i ekvationerna för att beräkna avstånden.

$$T = T_1 + T_2 \quad (1)$$

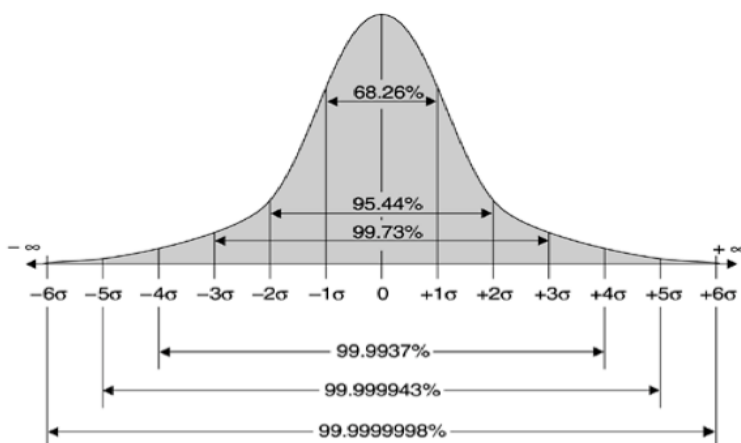
Vid uppmätning av retardationstiden bör detekteringszonen påverkas först vid det tillfälle då det tar den längsta tiden för maskinen att komma till ett viloläge. Det värsta fallet bör utsättas på maskinen när man mäter upp retardationstiden. Detta kan vara faktorer som vikt, temperatur och omkopplingstider av föråldrade komponenter. Vanligtvis väljer man att ha maximal belastning och maximal hastighet på maskinen när man mäter retardationstiden.

När retardationstider mäts bör det göras minst tio gånger för att man skall få maskinens verkliga standardavvikelse. Ett statistiskt sätt att täcka 99,73% av en standardfördelning är att använda medelvärde ± 3 standardavvikelse. Medelvärdet ± 3 standardavvikelse kan även beskrivas som en avvikelse på en av 370. Ifall något av mätvärdena är högre än medelvärdet $+ 3$ standardavvikelse, så bör det högsta mätvärdet användas för att kalkylera säkerhetsavstånden. (ISO 13855:2010, 2010)

Standardavvikelsen (σ) beskriver hur mycket ett visst antal mätvärden avviker från medelvärdet, eller med andra ord hur stor spridning det är av mätvärdena. En beskrivning på hur antalet standardavvikelse täcker en normalfördelad kurva kan man se i Figur 5.

För beräkning av standardavvikelsen används ekvation (2), där n = antalet mätningar, x_i = individuell mätning och \bar{x} = medelvärdet av alla mätningar.

$$\sigma = \sqrt{\frac{1}{n-1} * \sum_{i=1}^n (x_i - \bar{x})^2} \quad (2)$$



Figur 5 Beskrivning på hur antalet standardavvikelse täcker en normalfördelad kurva.

Istället för att beskriva mätningarnas standardavvikelse från medelvärdet, så kan man också beskriva vad som är standardfelet från medelvärdet. Detta beskriver hur noggrant en mätningens medelvärde är jämfört med det sanna medelvärdet. Man kan uttrycka detta som standardavvikelsen av medelvärdet från upprepade mätningar. Detta betyder alltså att desto fler mätningar som görs, desto lägre standardfel får man, vilket beskriver precisionen av mätningarna bättre. Standardfelet (SEM), kan beräknas med ekvation (3).

$$SEM = \frac{\sigma}{\sqrt{n}} \quad (3)$$

2.4 Beräkning av det minsta avståndet

När man skall välja hur man beräknar det minsta avståndet mellan säkerhetsvakten och farozonen, beror det på vilken säkerhetsvakt man använder och hur de är placerade i förhållande till farozonen. Ekvationerna varierar beroende på t.ex. hur ljusridån är monterad. Ljusridån kan t.ex. antingen vara monterad vertikalt, parallellt eller vinklad i förhållande till marken. Ifall det finns möjlighet att ta sig runt eller över en säkerhetsvakt, så behövs ytterligare beräkningar för att bestämma det minsta avståndet. Ifall det finns hinder mellan detekteringszonen och farozonen så skall ett indirekt avstånd beräknas. Alla dessa alternativ grundar sig ändå på ekvation (4) där T är den totala retardationstiden enligt kapitel 2.3, C är intrångsavståndet (mm) av en kroppsdel som t.ex. a av en arm som sträcks förbi säkerhetsvakten fören armen detekteras, S är det beräknade minsta avståndet (mm) man ska placera säkerhetsvakten från farozonen och K är parametern för kroppens rörelsehastighet (mm/s) som väljs beroende av den beräknade sträckan S .

$$S = (K * T) + C \quad (4)$$

Beroende på vilken typ av säkerhetsvakt man gör beräkningar till så har intrångsavståndet C olika värden. När man gör beräkningar av att man sträcker sig genom en skyddsridå så används parameter d , som är minsta diametern som säkerhetsvakten kan detektera (detta beskrivs av tillverkaren) för att räkna ut C , intrångsavståndet. När det däremot finns möjlighet till att sträcka sig över säkerhetsvakten mot farozonen så kan avståndet C väljas ut från en tabell i ISO 13855:2010. Dessa avstånd i tabell 1 har blivit framtagna i en industriell miljö.

Vid beräkning av säkerhetsvakter där det finns möjlighet att sträcka sig över mot farozonen så skall det göras beräkning för att både att sträcka sig genom och över säkerhetsvakten, och den beräkning som resulterar i ett längre avstånd, så skall väljas som det giltiga minsta avståndet. (ISO 13855:2010, 2010)

2.5 Testprotokoll

Efter att man har gjort mätningar av en maskins retardationstider och gjort beräkningar var skyddsutrustningen skall placeras, så skall detta dokumenteras i ett testprotokoll. Förutom att beskriva det beräknade avståndet till en specifik maskin som testats, så bör en del antaganden hur de värsta scenarion har konstaterats och hur metoden för stoppsignal har blivit given till maskinen. Ett fullständigt testprotokoll borde innehålla följande information:

- Identifiering av maskinen samt säkerhetsvakten som har blivit använd.
- Identifiering av personen eller företaget som har utfört mätningarna.
- Beskrivning av testutrustningen som har blivit använd samt metod av mätning.
- Verifikation och kalibrering av testutrustningen.
- Datum på när mätningarna utfördes.
- Den totala retardationstiden.
- Det beräknade minsta avståndet med värden som blivit använd i ekvationerna.
- Antaganden gjorda för mätningarna och beräkningarna.
- Tilläggsinformation om maskinen eller mätningens scenario.

3 Användbarheten av ett gränssnitt

Till examensarbetet behövs ett användargränssnitt så att det ska vara lätt för användare att interagera och göra mätningar med utrustningen. Detta användargränssnitt skall man försöka göra så användbart som möjligt. För att uppnå detta finns det vissa mål. Man kan säga att användbarhet kan uppnås om man uppnår de mål man har för programmet i fråga. Man kan dela upp användbarheten i två kategorier, nytta och användbarhet. Nyttan uppnås ifall funktionaliteten av systemet gör vad den skall. Användbarhet beskriver hur väl man uppnår funktionaliteten. Ofta görs lättanvända användargränssnitt som kan styras från antingen datorer eller paneler för att uppnå maskinens funktionalitet.

Användbarhet kan man dela in i 5 kategorier. Dessa är lärbarhet, effektivitet, minnesförmåga, fel och nöjdhet. Användbarhet mäts normalt genom att ha en kontrollerad testgrupp som skall utföra en förutvald uppgift. Det är värt att påpeka att användbarheten mäts relativt till användarens tidigare kunskap inom liknande miljö. För att avgöra användbarheten för ett system, så behöver det inte alltid vara bäst med ett medelvärdesresultat. Med medelvärde kan det ge väldigt varierande resultat, eftersom det är väl känt att användare har väldigt varierande kunskap. Det är på grund av detta klokare att använda resultaten från alla användare. Ett exempel på detta är att man har en resultatskala från ett till fem, och jämför hur andelen användare som röstat ett eller fem. Man kan ha kriterier att inte fler än hälften har givet mindre än tre.

Lärbarheten av alla system kan representeras två olika sätt. Den första är program som är väldigt intuitiv och användaren kommer fort igång, men det tar längre att nå en expertnivå. Det andra är system som är mer komplicerat att använda från en början, men efter en tid av inläring av grunderna så går det allt snabbare att nå en expertnivå. Det lättaste sättet att mäta hur snabbt en helt ny användare lär sig systemet är att bedöma när användaren har uppnått en viss förståelse av systemet. Den önskade förståelsen kan vara att användaren skall utföra en specifik uppgift med systemet. Viktigt att tänka på är att de flesta användare inte lägger ner så stor tid på att lära sig systemet komplett, utan bara så mycket att man klarar av att utföra det som behövs. Det har konstaterats att det viktigaste ett system bör ha för att vara lätt att lära sig är att felkoder är lätta att tyda, möjligheten att ångra gjorda misstag och att kritiska ändringar kräver en bekräftelse före ändring.

Effektivitet av användning av ett system mäts av användare som klassas som experter av systemet. Man klassificerar ofta en användare till expert när personen har använt systemet

en viss tid. Dessa experter inom systemet får en uppgift de skall utföra. Effektiviteten mäter man genom att ta tid på hur snabbt experterna kan utföra uppgifter.

Minnesförmåga av system är hur bra man kommer ihåg hur man utför uppgifter efter att man inte använt systemet på en längre period. Det är viktigt att systemet är lätt att memorera, eftersom en stor del av användarna inte använder systemet så ofta. Till kontrollgruppen för detta används inte experter, eftersom de vanligtvis använder systemet ofta. Istället testas användare som enbart använder systemet enstaka gånger i året. Alternativt ber man dessa samma användare att utföra en uppgift och i efterhand fråga hur de har gjort.

Fel av systemet kan klassas till hur många fel eller komplikationer användaren har stött på för att utföra en uppgift. Målet är förstås att man skall ha så få fel som möjligt. Man kan förstås inte räkna alla fel till samma kategorier. En del av felen som användaren gör så korrigeras omedelbart efteråt, medan andra förblir kvar i systemet.

Subjektiv tillfredställelse av ett system kan fås av användare på många olika sätt. Generellt sätt så finns det två vanliga metoder av detta. Den första är att man har formulär med frågor där man t.ex. frågar hur väl användaren samtycker om påståendet där en etta är helt av annan åsikt och en femma samtycker man helt och hållet. Ibland kan man bara inte förlita sig på dessa metoder, utan ibland är det även bra att man ställer användarna specifika frågor om hur de upplever att använda programmet. Något som är väldigt viktigt är att man inte är för hastig att fråga användarna om hur de upplever hur systemet är att använda. Man bör låta användarna använda programmet i verkligheten för att få en bra inblick om hur de upplever systemet. Det har konstaterats att det finns användare som vägrar att använda ett system ifall de tycker att manualerna är för långa och komplicerade.

För att ett system skall vara användbar så bör man ha möjlighet att använda det på fler än ett sätt. När en helt oerfaren person börjar använda ett system så är det till en början väldigt stor fördel ifall man lätt kan visuellt navigera sig fram dit man behöver. På detta sätt får användaren en helhet över hur allt hänger ihop. När användaren börjar övergå från att vara en nybörjare till ett måttligt kunnande och slutligen vidare till att bli en expert, så kan det fort bli väldigt långgrandigt att använda samma funktioner som en nybörjare använder sig av. För att få en expert att bli mer effektiv så är en bra funktion i ett system att ha genvägs knappkombinationer för att utföra vissa ändamål. Ifall det enbart skulle finnas dessa knappkombinationer, så skulle en nybörjare ha väldigt svårt att lära sig grunderna. Därför är det bra om man kan kombinera dessa funktioner. På detta sätt kan man sakta men säkert bli mer effektiv hela tiden. (Nielsen, 1993)

4 Maskinvara

Här beskrivs de fysiska komponenter som har blivit använda för detta examensarbete.

4.1 Programmerbar styrenhet (PLC)

För att utföra retardationstidsmätningar så behövs en liten programmerbar dator. Till examensarbetet valdes det en B&R PLC. Denna PLC valdes p.g.a. att uppdragsgivaren använder den från tidigare och eftersom den har de kommunikationskrav som behövs med andra hårdvaror. B&R är en förkortning av Bernecker + Rainer Industrie-Elektronik och är ett österrikiskt företag grundat 1979. De huvudsakliga verksamhetens områden är industriella datorer, visualisering, motorer och motorstyrning.

Det finns några fördelar med de PLC som B&R tillverkar jämfört med andra PLC som t.ex. Siemens S7. För det första så kan man ansluta komponenter som säkerhetselen, frekvensomriktare, pekskärmar, distribuerade stationer och mycket mera. Dessa komponenter kommunicerar enligt B&Rs utvecklande Powerlink, som är baserat på TCP/IP, men har flera lager i OSI modellen vilket säkerställer att inga paket går förlorad. Detta betyder att man säkert kan överföra kritiska signaler som t.ex. stoppkommandon. Alla dessa komponenter kan nu programmeras och konfigureras från ett och samma program och senare överförs till PLC:n. Det enda som varje komponent fysiskt bör konfigureras till, är att ge en adress med hjälp av två små ställ rattar. Jämfört då med många andra tillverkare så skall dessa komponenter programmeras och konfigureras i skilda miljöer och överförs enskilt till komponenterna.

B&R har väldigt kompakta och kapslade distribuerade stationer, som lätt kan placeras närmare själva processen på en maskin, vilket betyder att t.ex. givares och styrans kablar inte enskilt bör dras långa vägar till elskåpet.

PLC:n som blev använd till examensarbetet består av separata moduler som man kan montera ihop med varandra för det ändamål man behöver. Själva processorn var av modellen X20CP1585, som består av en 1.0 GHz processor och 256 MB DDR2 SDRAM. För applikationsminne finns ett s.k. ”CompactFlash” minne som man lätt kan byta ut till valbar storlek. Kommunikation som finns är Ethernet, Powerlink, USB och X2X. Buss systemet X2X är kommunikation som sker mellan de moduler som man kan lägga till och processorn. Till examensarbetet behövdes en modul med SSI protokoll som beskrivs i kapitel 6.1.

Eftersom denna SSI modul X20DC1198 även har två stycken digitala ingångar, så behövs inga extra moduler för att få externa insignaler. (B&R Automation, 2017)

4.2 Vajermekanism

För att mäta maskinens rörelser används en vajer som fungerar som en jojo. Vajern är virad runt en rulle och när vajern dras ut så roteras rullen. Denna vajermekanism ser man i Figur 6. Tillverkaren av vajermekanismen har i specifikationerna beskrivit hur långt vajern dragits ut när rullen har roterat ett varv. Denna position av rullen blir uppmätt av en absolutgivare som är fäst på rullens axel.



Figur 6 SICK Vajermekanism MRA-G130-105D3 med absolutgivare AFM60E-S1AA004096.

Den modell som blev använd till examensarbetet var MRA-G130-105D3, som har en 5 meter lång vajer. På ett roterat varv av rullen kommer vajern 385 mm ut. Noggrannheten beskrivs enligt linjäriteten med en max avvikelse på ± 3 mm av en viss mätning. Denna modell användes eftersom att uppdragsgivaren hade den i sitt lager från förut. Det finns motsvarande vajermekanismer med längre vajer som kan behövas ifall man testar maskiner som har längre sträckor som den rör sig på. (Sick AG, 2017)

4.3 Pulsgivare

En pulsgivare är en elektromekanisk enhet för att övervaka rörelse och position. Det finns roterande och linjära pulsgivare, men i detta kapitel beskrivs den roterande pulsgivaren som används i examensarbetet.

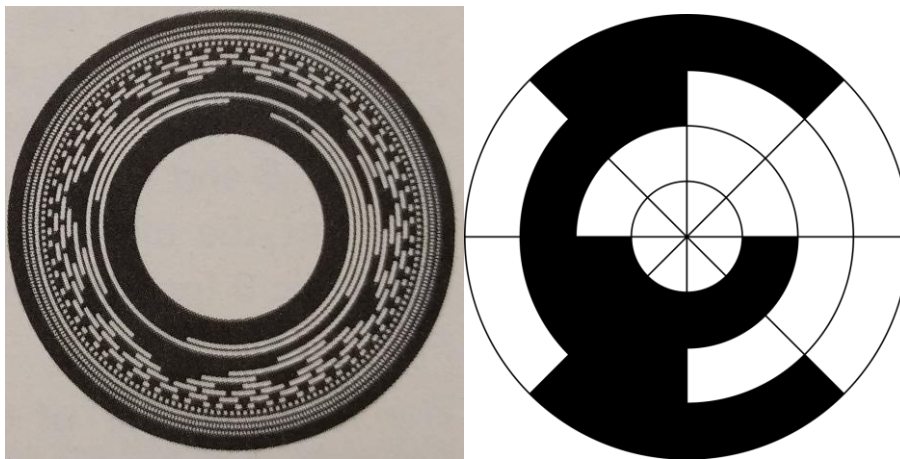
Det finns några typer av roterande pulsgivare men de vanligaste är inkrementell- och absolutgivare. Inkrementella pulsgivare blir oftast använd när enbart den relativa positionen

behövs. För inkrementella pulsgivare är resolutionen definierad som antalet pulser per roterat varv. Absolutgivarna kan delas in i envarvig och flervarviga. Med en varvig absolutgivare så bestäms resolutionen enligt antalet positioner per roterat varv, medan flervarviga absolutgivare även räknar antalet roterade varv.

Pulsgivare finns i flera typer som konduktiv, optisk, kapacitiv och magnetisk. Den mest använda av dessa är den optiska pulsgivaren. Den består internt av en skiva som har ringar med spår, som man ser i Figur 7. För varje spår som finns på skivan så finns en LED lampa och en ljusdetektor. Spåren på skivan har små genomskinliga luckor som detekteras av att LED lampan finns på ena sidan av skivan och på andra sidan finns ljus detektorn. Från varje spårs ljusdetektor skickas informationen vidare till en mikroprocessor som beräknar informationen för absolutgivarens tillfälliga position. Denna information kan sen överföras enligt lämpligt protokoll till t.ex. en PLC.

Inkrementell pulsgivare har bara två spår med fyrkantsvågor som är färförskjutna 90° från varandra, även ett tredje spår kan finnas för att bestämma pulsgivarens hemposition. Med detta får man bara en relativ position, dvs man börjar räkna pulser från en startpunkt tills man gjort ett fullständigt roterat varv och kommit tillbaka till samma punkt.

Absolutgivare har däremot många unika spår som producerar en binär Gray kod. Skillnaden mellan vanlig binärkod och Graykod är att den senare nämnda enbart ändrar på en bit mellan varje position. Detta Gray kod spår kan man se i Figur 7. Detta leder till att den kallas till en absolutgivare, eftersom fast strömmen har varit av så kan en exakt position bestämmas. Från position ges en puls, alltså en pulsgivare. Både termerna positioner och pulser används för att beskriva resolutionen för en absolutgivare.



Figur 7 Till vänster är spåren på en skiva som finns inne i en absolutgivare som bestämmer positionerna. Till höger är en förstorad förenklad version där man kan se hur enbart en bit eller ett fält förändras per position/sektion enligt Gray kod.

Den vanligaste metoden absolutgivare använder sig av för att mäta antalet roterade varv, är att ett litet kugghjul är fäst i kombination med absolutgivarens axel och driver ett större kugghjul som i sin tur är fäst i en helt skild skiva med spår i. Detta betyder alltså att när absolutgivarens axel har roterat ett varv så har det stora kugghjulet i sin tur flyttat sin skiva så att en position har förändrats enligt samma metod som beskrivits i förra stycket, och som framgår i Figur 7.

Ifall det t.ex. finns 12 spår som bestämmer den absoluta positionen så betyder det att resolutionen av absolutgivaren uttryckt i tvåpotensform blir 2^{12} eller decimalt 4096 positioner. Detta är alltså då en 12 bitars resolution av absolutgivaren. (Eitel, 2014) (Electro-Craft Corporation, 1980)

Absolutgivaren som användes till examensarbetet var av modellen AFM60E-S1AA004096. Efter som att det är en ”absolut” pulsgivare, så betyder det alltså att den använder sig av Gray kod, detta beskrivs även i tillverkarens datablad. Denna modell har uttryckt i tvåpotensform $2^{12} = 4096$ positioner per roterat varv och kan räkna $2^{12} = 4096$ stycken roterade varv, alltså 12 bitars resolution för positionen och 12 bitars resolution för att räkna antalet roterade varv. Noggrannheten av absolutgivaren är $\pm 0,2^\circ$. Kommunikationen med denna absolutgivare sker över 1 MHz SSI protokoll på som beskrivs i kapitel 6.1. (Sick AG, 2017) (B&R Automation, 2017)

5 Programvara

Här beskrivs om det program som har använts för att programmera PLC:n samt om de olika programmeringsspråks alternativen.

5.1 Automation Studio

Programmeringen av PLC:n görs i programmet Automation Studio som är utvecklad av B&R redan 1997, alltså samma företag som tillverkat PLC:n. Denna programvara är en utvecklingsmiljö som innehåller verktyg för alla skeden i ett projekt. Programmering av PLC, visualisering och frekvensomriktare kan göras bekvämt med hjälp av samma program.

För insättning av ny maskinvara så är nödvändiga komponenter enkelt listade för att göra det simpelt. Detta gör det lättare att skapa lokala och fjärrstyrda maskinvara strukturer. Genom att ställa in alla inställningar i Automation Studio, så har man en bättre överblick.

Visualisering kan skapas i Automation Studio för olika ändamål. Man kan styra dessa visualiseringar från t.ex. från en pekskärm, men det går även att styra samma visualisering genom en s.k. VNC, som det beskrivs i kapitel 6.3.

Diagnostisering av B&R's PLC med hjälp av Automation Studio är väldigt mångsidig. Det finns t.ex. simpel övervakningar av variabler, program och in- och utgångar. Med Automation Studio kan man även diagnostisera PLC över internet i en helt vanlig webbläsare från vilken plats som helst. (B&R Automation, 2017)

5.2 Programmeringsspråk

Automation Studio har många programmerings möjligheter. De olika språk och miljöer som man kan jobba med är de standardiserade IEC 61131-3 språken, C++, Automation Basic, Continuous Function Chart, och Ansi C. Med alla dessa olika alternativ så blir det lätt för en ny användare att använda en miljö som känner sig bekant med.

IEC 61131 är internationell standard för programmerbara styrsystem, som blev publicerad 1993 och är uppdelad in i ett antal delar. Den tredje delen IEC 61131-3 som blev publicerad 2013 beskriver programmeringsspråk. Dessa programmeringsspråk består av två textbaserade språk, instruktions lista och strukturerad text, och tre grafiska språk, Ladder diagram, Funktions Block Diagram och Sekventiell funktionsdiagram. IEC 61131-3 är som

riktlinje för PLC programmering, inga bestämda regler. (PLCopen, 2013) (John & Tiegelkamp, 2010)

Till examensarbetet blev PLC:n programmerad med strukturerad text. Detta är dels p.g.a. att uppdragsgivaren använder sig av detta språk när de har programmerat deras andra maskiner, men jag valde också detta programmeringsspråk eftersom jag aldrig har jobbat med det förut. Strukturerad text baserar sig på programmeringsspråket Pascal och har många likheter med programmeringsspråket C.

6 Datakommunikation

Kommunikationen som finns mellan PC, PLC, PLC moduler och absolutgivare är beskrivna under denna rubrik.

6.1 Synchronous Serial Interface (SSI)

Kommunikationen som sker mellan PLC:n och absolutgivaren fungerar med SSI protokoll. SSI används väldigt mycket inom industrin, speciellt för absolutgivare. Det är ett s.k. punkt-till-punkt protokoll med en master, vanligtvis en PLC eller mikrokontroller och en slave t.ex. en absolutgivare. Positionen av absolutgivaren uppdateras hela tiden och alla signifikanta bitar sparas parallellt i ett shiftregister. PLC:n skickar ett pulståg till absolutgivare och den skickar tillbaka positionen som är i shiftregistret seriellt, börjande med den mest signifikanta biten. När absolutgivare skickat den minst signifikanta biten håller den en liten paus och efteråt uppdateras registret med en ny position. (FRABA Inc., 2013)

Protokollet använder sig också ibland av s.k. monoflop som säger när shiftregistret ”latchar”. Detta betyder alltså när shiftregistret skall låsa alla signifikanta bitar, för att kunna överföra hela protokollets längd seriellt utan att bitarnas tillstånd i shiftregistret förändras konstant.

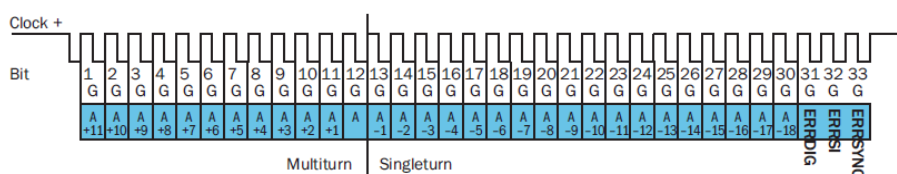
Om man tar absolutgivaren från kapitel 4.3 som fungerar över SSI protokollet, så har den 12 bitars resolution för att beskriva positionen av ett roterat varv och 12 bitars resolution för att räkna antalet roterade varv. Som man ser från Figur 8 så finns antalet roterade varv från bitarna 1–12 av protokollet och positionen per roterat varv finns på bitarna 13–30 av protokollet. Längden av SSI protokollet som alltså behövs för absolutgivaren i kapitel 4.3 blir alltså 24 bitar som ser i Figur 8, eftersom den 24:e biten av kommunikationen motsvarande den 12:e biten av beskrivningen av positionen av ett roterat varv.

Ifall man använder sig av en absolutgivare som t.ex. bara kan räkna 1024 positioner per roterat varv och 1024 antal roterade varv, alltså uttryckt i tvåpotensform 2^{10} så har den 10 bitars resolution. I detta fall för man hålla koll på hur många inledande nollor som finns i protokollet. Längden av protokollet är då 20 bitar lång, men eftersom de två första bitarna i protokollet inte används så är protokollets längd + inledande nollor 22 bitar långt.

Man ställer in dessa värden för noden som skickar pulståget och tar emot data, alltså i detta examensarbete PLC:n. Om man då ställer in att man har 24 bitars protokoll, så maskas resten

bort och kvar lämnar de tre felkodsbitarna som kommer i slutet av protokollet. (B&R Automation, 2017)

30 Bits



Bit 1–12: Position Bits Multiturn

Bit 13–30: Position Bits Singleturn

Bit 31–33: Error Bits

Figur 8 Visuell beskrivning av hur pulståget skicks för en position.

6.2 USB-överföring

För att spara skärmbilder till ett USB-minne som i kapitel 7.4, så bör detta konfigureras. Information om hur PLC:ns USB-överföring sker finns det enbart lite om, men teorin om USB-överföring är ändå samma som för andra enheter. Som exempel på hur USB-överföringen sker, så kommer här en kort förklaring av en PIC mikroprocessors USB-överföring.

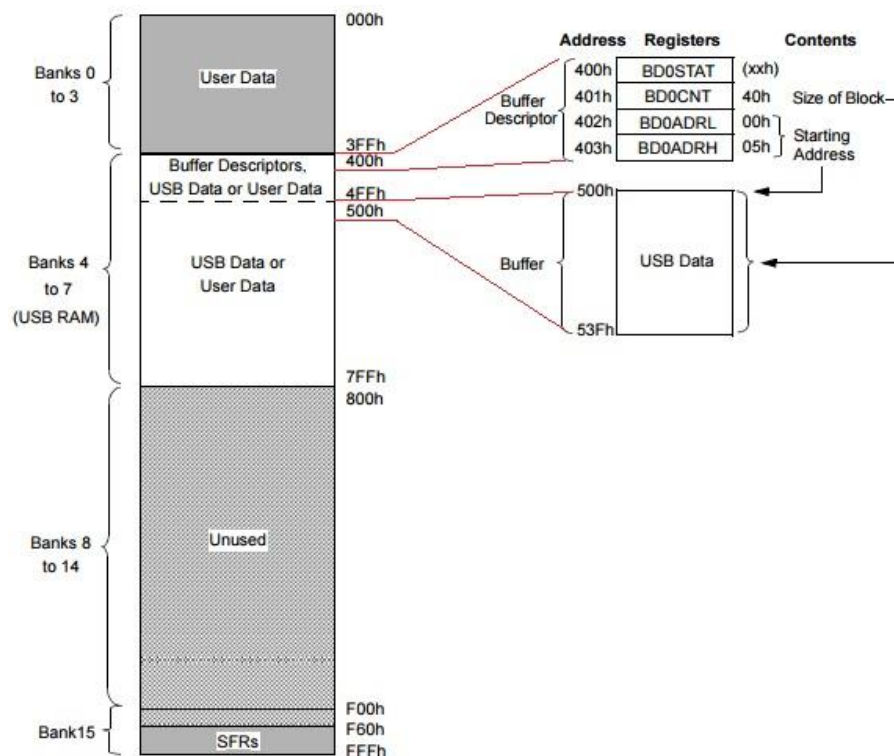
När data överförs från datorns processor (t.ex. som i detta examensarbete är det PLC:n i kapitel 4.1) till en USB enhet, så skrivs data först till USB-RAM bufferten och efter att den har skrivit klart så läser USB enheten vad som finns i bufferten. RAM är en tillfällig minnes lagringsplats och buffert är den tillfälliga mängden data som sparas där så länge överföringen sker.

För att inte datorns processor och USB enheten skall försöka skriva till bufferten samtidigt, eller för att ett flertal USB enheter skall kunna vara anslutna samtidigt, så behövs något som specificerar detta noggrannare. Därför finns det något som kallas för ”*buffer descriptor*” som blir förkortat till BD, eller m.a.o. en buffertbeskrivning. Det finns ett visst bestämt antal av dessa buffertbeskrivningar beroende på viken typ av USB-RAM som används, som kan tilldelas till varje ansluten USB enhet enskilt.

Buffertbeskrivningen innehåller ett status register var det bl.a. framgår vems tur det är att skriva till bufferten. Buffertbeskrivningen innehåller också information om hur stor bufferten är som överförs i USB-RAM och en adress till var i USB-RAM bufferten börjar. Processorn har möjlighet att läsa alla register som finns i USB-RAM. Processorn går turvist

igenom alla buffertbeskrivningarna och kan t.ex. ge USB enheterna tillstånd att skriva till bufferten en efter en. När processorn har ett ställt biten statusregistret att det är en viss USB enhets tur, så skrivs data av USB enheten enligt buffertbeskrivningen och när den är klar så nollställs biten i statusregistret. Nu vet även processorn att USB enheten är klar och kan hitta data som USB enheten har överfört genom adressen i buffertbeskrivningen. Detta fungerar vise versa om överföringen går andra vägen.

I Figur 9 ser man buffertbeskrivningen BD0 till höger, där dess adress går från 400h–403h. Nästa buffertbeskrivning BD1 finns alltså då på adresserna 404h–407h. Alla buffertbeskrivningarna finns på banken 4 och på bankerna 5–7 så finns bufferten. Första registret av BD0 är statusregistret där bl.a. en av bitarna beskriver vems tur det är att skriva till bufferten. I det andra registret beskrivs storleken av bufferten som överförs. De två sista registren beskriver var i bufferten den överförda data finns. Eftersom det är en 8-bitars PIC mikrokontroller så måste informationen om var bufferten befinner sig spjälkas upp i två register. (Microchip Technology, 2009)



Figur 9 Tillämpning av USB-RAM i dataminnesutrymme. Bilden är från ett datablad för en PIC mikroprocessor som klarar av USB-överföring, men principen är den samma.

6.3 Virtual Network Computing (VNC)

Som användargränssnitt till att styra PLC:n så gjordes ett grafiskt användargränssnitt. Detta kan användas om man kopplar t.ex. en pekskärm till PLC:n. Man kan även få samma virtuella skärm med en dator genom att använda öppen VNC programvara.

VNC protokollet är ursprungligen tillämpad för att klienter skall ha en trådlös uppkoppling till servrar. Protokollet fungerar på en rambuffert nivå och fungerar därför på alla operativsystem med grafiskt användargränssnitt och med någon form av kommunikation, som t.ex. TCP/IP. Den noden i protokollet som användaren interagerar med kallas för ”VNC Client” eller ”VNC Viewer”. Den andra ändan som blir styrd kallas för ”VNC Server”. (Richardson, et al., 1998)

Ett exempel av en gratis programvara för Windows är ”TightVNC Viewer”. För att skapa en VNC kommunikation mellan en PC och en PLC så kopplas de ihop med en Ethernet kabel. Konfigureringen för att kommunikationen bör ställas in i både Automation Studio och i Pc:ns Ethernet inställning. Båda enheterna bör ha dylika IP adresser så att enbart de sista siffrorna i punkt-decimal beskrivningen av IP adressen skiljer dem åt.

Det uppstår ett problem med att använda en ”VNC Viewer” med den PLC som valts till examensarbetet i kapitel 4.1. PLC:n kräver att man i ”VNC Viewer” programvaran konfigureras till användning av enbart 256 färger, alltså 8-bitars färg. P.g.a. denna konfiguration så kommer färgerna som man har valt till användargränssnittet inte att stämma överens med de färger som man kommer att se i ”VNC Viewer” programvaran. Som ett exempel på detta är om man har valt att en figur skall vara grå, så kommer den att visas som vit eller om man har valt att en figur skall vara vit så kommer den att visas som blå. Om man väjer att interagera med användargränssnittet med en pekskärm så kommer däremot alla färger att visas i samma nyans som han har valt i Automation Studio.

7 Metoder och tillvägagångssätt

Här beskrivs tillvägagångssätten av de praktiska delarna av examensarbetet.

7.1 Användning av absolutgivare

Här beskrivs hur man i programvaran Automation Studio konfigurerar ett SSI protokoll till en absolutgivare och slutligen formaterar data till lämplig mätenhet.

7.1.1 Konfigurering av absolutgivare

Som det beskrivs om SSI protokollet i 6.1 så skall längden för protokollet konfigureras för att det skall fungera korrekt. Som exemplet av SSI protokollet i kapitel 6.1 för att bestämma protokollets längd för absolutgivaren i kapitel 4.3, så skall protokollets längd vara 24 bitar. Konfigurering av absolutgivaren i programvaran Automation studio, gör man i det område i programmet som behandlar fysiska komponenter. Annan information ställs även in som att absolutgivaren använder gray kod, den fungerar på en viss frekvens och vilken monoflop absolutgivaren är på.

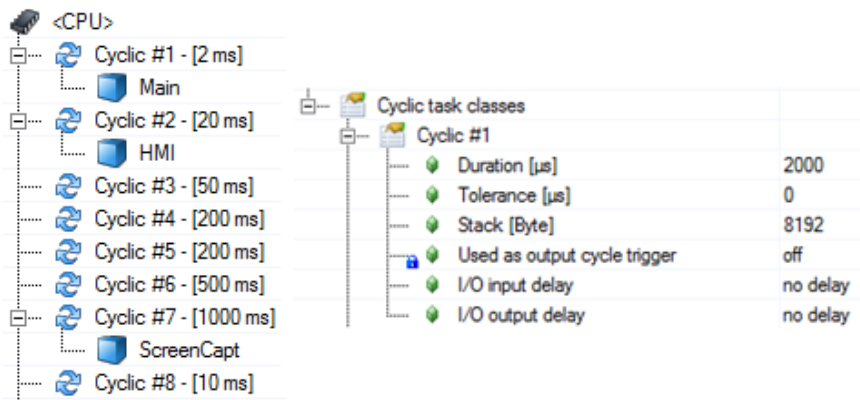
Av denna information ger Automation Studio ut en variabeltyp ”*unsigned double integer*”. Med begreppet ”*unsigned*” beskrivs att variabeln inte visar negativa tal, utan det minsta möjliga värdet är noll. Med begreppet ”*double integer*” beskrivs storleken av variabeln, vilket enligt standard IEC 61131-3 är 4 byte som är samma som 32 bitar. Så det största värdet som man får är $2^{32} \approx 4,3 * 10^9$. Denna variabel som fås ger ett växande heltal när absolutgivarens axel vrids åt ena hållet och när axeln vrids åt andra hållet så minskar variabelns heltal.

7.1.2 Val av mätenhet

För examensarbetet är målet att få uppmätt maskinens hastighet i mm/s under tiden som maskinen fått stoppsignal, tills att den har kommit till ett viloläge. Denna hastighetsförändring skall representeras i en graf.

I programvaran Automation Studio så kan man ställa in åtta olika programcykelgrupper som körs på PLC:n. I varje grupp kan man välja att placera flera olika program att köras. Varje enskild programcykelgrupp har en hel del inställningar som man kan justera, som t.ex. cykeltiden i mikrosekunder och toleransen i mikrosekunder från cykel till cykel. Programmet

”Main” som avläser absolutgivaren, finns under cykel grupp ett och blev inställt till en cykeltid på 2000 mikrosekunder med ± 0 mikrosekunders tolerans, som man ser i Figur 10.



Figur 10 Konfigurering av de olika programmens cykeltider.

Denna cykeltid behöver vara väldigt exakt eftersom att absolutgivaren skall beskriva hastigheten som beräknas enligt ekvation (5). Cykeltiden får inte heller vara för snabb, eftersom då kan det vara att programmet inte märker av att absolutgivaren har rört på sig, fast den i själva verket rör på sig hela tiden. Detta leder till att man inte får ut någon väsentlig information av hastigheten eftersom den kan hoppa från verklig hastighet till 0mm/s och upp tillbaka till verklig hastighet, alltså man får en väldigt hackig graf.

$$\text{hastighet} = \frac{\Delta \text{sträcka}}{\text{tid}} \quad (5)$$

Positionsförändringen av vajermekanismen mellan två programcyklar, alltså $\Delta \text{sträcka}$ kan man beräkna enligt vad tillverkarna har specificerat om vajermekanismen (kapitel 4.2) och om absolutgivaren (kapitel 4.3). På ett roterat varv av vajermekanismens rulle kommer vajern 385 mm ut. Från ett roterat varv ger absolutgivaren 4096 pulser.

För att få fram en sträcka ska man alltså i programmet mäta förändringen av variabeln från absolutgivaren mellan två program cyklar, alltså Δpulser . På basen av detta kan man nu räkna ut hastigheten mellan två program cyklar enligt ekvation (6).

$$\text{hastigheten} = \frac{\text{vajerlängden per roterat varv} * \Delta \text{pulser}}{\text{pulser per roterat varv} * \text{program cykeltid}} \quad (6)$$

7.1.3 Integer overflow

Som det beskrivs om absolutgivaren i kapitel 4.3 så räknas positionerna av absolutgivaren per roterat varv samt antalet roterade varv. När absolutgivarens axel vrids antingen medsols eller motsols får PLC:n information om detta som ett växande eller minskande heltal, som sen sparas till en variabel. När absolutgivaren har räknat max antalet roterade varv enligt dess resolution, så kommer räkningen att börja om från noll.

Beroende på hur absolutgivares axel är roterad när den har blivit monterad på vajermekanismen, så kan det förekomma ett s.k. ”integer overflow”. Detta betyder alltså att absolutgivarens antal roterade varvräkning övergår från det högsta värdet enligt dess resolution till noll, eller tvärt om.

Detta förekom när testutrustningen utvecklades. Värdeförändringen av absolutgivaren mellan två programcyklar blev plötsligt väldigt högt, vilket resulterade till en momentan hög hastighets topp. Hastighetsberäkningen beskrivs i kapitel 7.1.2. När detta problem uppstod så fanns en av firmans programmerare i samma rum. Han hade fått samma problem förut och hade löst detta genom att göra ett eget funktionsblock i Automation Studio som löste detta problem. Som in parametrar för detta funktionsblock ges absolutgivarens resolution för positioner per roterat varv multiplicerat med absolutgivarens resolution för antalet max roterade varv. Av detta får man alltså det högsta värdet som absolutgivaren kommer att räkna till. Funktionsblocket hade även en skalnings funktion, så genom att ge in en skalningskonstant så gav funktionsblocket ut antalet millimeter som vajern har färdats från sin hemposition. Skalningskonstanten beräknades genom att ta absolutgivarens antal positioner per roterat varv delat med längden av vajermekanismen för ett roterat varv. Från detta får man alltså antalet pulser som absolutgivaren ger ut per utdragen millimeter av vajermekanismen. Hastigheten kan nu enkelt beräknas enligt ekvation (5), där $\Delta sträcker$ beräknas enligt förändringen av vad funktionsblocket ger ut för avstånd i millimeter mellan två program cyklar.

7.2 Hantering av data

Här beskrivs hur man mäter ett antal retardationstider av en maskin och hur data samlas in för att senare kunna visa dessa mätningar i en graf.

7.2.1 Retardationstidsmätning av programmet

Tiden det tar för maskinen att komma till ett viloläge från att ha fått stoppsignal från maskinen som testas, kan beräknas med Automation Studios färdiga funktionsblock som beräknar tidsskillnaden mellan två stycken tidsvariabler. Funktionsblocket returnerar resultatet i millisekunder som sparas till en ny variabel.

När programmet läser av att den digitala ingången som i sin tur får en stoppsignal från maskinen som blir testad, så sparas den tillfälliga tiden till en variabel från en intern klocka som kontinuerligt räknar upp i millisekunder.

Efter att en tillståndsförändring har inträffat digitala ingången så väntar programmet på att absolutgivarens hastighet som beskriven i kapitel 7.1.2 har nått hastigheten 0mm/s. När detta uppnås kan man med tidigare nämnda funktionsblock beräkna tiden mellan nuvarande tid och tiden som sparades vid tillståndsförändringen av digitala ingången, för att få retardationstiden T_2 enligt ISO 13855:2010 beskrivet i kapitel 2.3. Från varje ny mätning som görs så sparas retardationstiden i ett nytt element i en vektor som lagrar tiderna från varje mätning.

Det visade sig att den uppmätta hastigheten hade ett litet brus med växlande hastigheter mellan 0–100 mm/s mellan varje programcykel. För att få en giltig retardationstid så valdes att hastigheten skall vara lägre än en viss minimihastighet, under en viss tid. Dessa två parametrar kan justeras från användargränssnittet.

7.2.2 Loggning av data

Eftersom att man vill representera mätningens hastighet och stoppsignal i en graf som en stillbild, så bör alla datapunkter loggas under en mätningens stopprocedur. Som det beskrivs i kapitel 2.3 om ISO 13855:2010, så bör minst tio stycken mätningar göras för att få ett giltigt resultat.

Hur länge man vill logga data för en mätning bestäms enligt ett bestämt antal element i en vektor multiplicerat med cykeltiden för programmet som beräknar loggar data. Både tillståndet av digitala ingången som får stoppsignalen av maskinen som blir testad och hastigheten loggas till två stycken tvådimensionella vektorer. Deklarering av sådana vektorer ser man i Figur 11, där storleken av den första dimensionen är antalet möjliga mätningförsök som går från noll till ”*max_num*” och den andra dimensionen är antalet

datapunkter som skall loggas från en mätning och går från noll till ”*max_vector*”. Variablerna ”*max_num*” och ”*max_vector*” är fördefinierade under variabel deklareringsen.

Cykeltiden för programmet med loggningen är 2ms och det valdes att man vill logga max 3 sekunder av stoppet av maskinen. Därför beräknas antalet datapunkter som behövs för en mätning genom att man tar 3 sekunder delat med 2ms vilket är 1500. Eftersom data börjar sparas från elementet noll, så är det högsta elementet för den andra dimensionen då alltså 1499. Antalet möjliga mätningsförsök valdes till 20 stycken, vilket betyder att det högsta elementet i den första dimensionen är 19.

◆	arrDatapointsVel	REAL[0..max_num,0..max_vector]
◆	arrDatapointsInSignal	BOOL[0..max_num,0..max_vector]

Figur 11 Detta är s.k. tvådimensionella vektorer där digitala ingången och hastighetens alla datapunkter under retardationstiden från alla gjorda mätningar loggas.

Då en begäran för en ny mätning har gjorts från användargränssnittet så börjar loggning av tillståndet av den digitala ingången, samt loggning av hastigheten för absolutgivaren som nämnts i kapitel 7.1.2. Vid varje ny påbörjad programcykel efter detta så sparas värden för digitala ingången och hastigheten i ett nytt element i varsin vektor. Det skulle vara väldigt ansträngande att behöva synkronisera att man först startar maskinen och efteråt begära en ny mätning från användargränssnittet och slutligen ge maskinen en stoppsignal allt under loppet av 3 sekunder. Istället ser programmet ifall tillståndet av den digitala ingången har förändrats när den räknat ett visst antal programcyklar. Ifall ingen tillståndsförändring av digitala ingången har skedd under ett visst antal programcyklar, så återgår loggningen av datadatapunkterna med att skriva över från de första elementen i vektorerna.

Efter att tillståndsförändringen av digitala ingången har skedd och hastigheten har slutligen nått ett viloläge som beskrivet i kapitel 7.2.1, så sparas vilken indexering av element hade i hastighetsvektorn, för att senare kunna visa hastigheten i en passlig tidsskala på den horisontella axeln av grafen. Efter mätningen gjorts så sparas även den högsta hastigheten under mätningen för att senare visa mätningen i en passlig skala på den vertikala axeln i grafen.

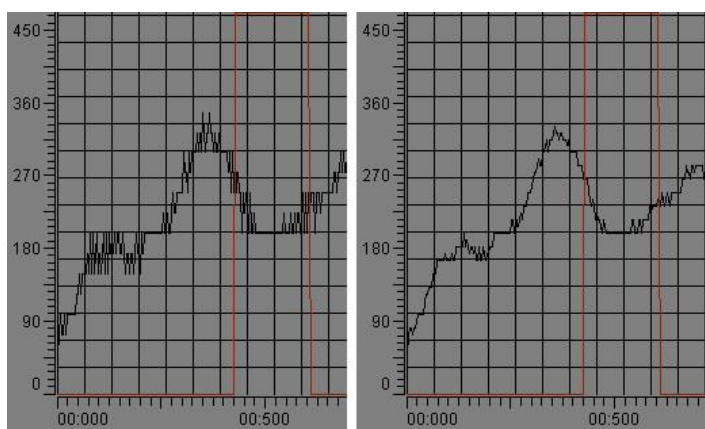
7.2.3 Statistik beräkning

Av de retardationstider som har blivit uppmätta så beräknas medelvärde, standardavvikelse och standardfel enligt kapitel 2.3. För att göra detta gjordes ett eget funktionsblock. Detta gjordes eftersom att samma beräkningar kommer att göras väldigt många gånger och därför är det smidigare med ett funktionsblock. Genom att deklarerera funktionsblocket som en vektor, så kan beräkningarna från varje mätning sparas. Som in parameter till funktionsblocket ges en vektorn med de uppmätta retardationstiderna som nämnts i kapitel 7.2.1, samt ges en variabel som beskriver hur många mätningar som har blivit gjorda.

7.2.4 Förberedelse av data till användargränssnittet

Efter att ett mätningar har gjorts så kan man från användargränssnittet begära att se grafer, tider och beräkningar från varje skild mätning. För att visa hastigheten från en viss mätning i en graf så bör detta ges i en endimensionell vektor. Detta betyder alltså att när en viss mätning begärs att ses från användargränssnittet, så bör datapunkterna från de tvådimensionella vektorerna där datapunkterna från mätningen finns, kopieras till en tillfällig endimensionell vektor som grafen kan läsa.

Som det nämns i kapitel 7.2.1 så har hastigheten en del brus. Ifall man då skall visa en mätning vars högsta hastighet inte är väldigt hög så kommer grafen att vara väldigt hackig. För att lösa detta så kan man beräkna medelvärdet för varje datapunkt tillsammans med föregående och nästa datapunkt i vektorn. Med hjälp av detta får man en jämnare kurva i grafen. Ett exempel på detta kan man se i Figur 12 där den högsta hastigheten enbart är ca 280 mm/s.



Figur 12 Till vänster ser man att hastigheten i mm/s har ett litet brus, medan i den högra bilden ser man att bruset har minskat när man har beräknat medelvärdet för varje datapunkt.

Grafen behöver även andra parametrar som beskriver tiden mellan två mätpunkter vilket beskrivs i kapitel 7.1.2, det högsta värdet av den vertikala axeln och när mätningen blev klar för att få en lämplig tidsskala. Dessa parametrar är sparade i vektorer från alla enskilda mätningar, men kopieras till tillfälliga variabler så att de är i giltig format för grafen.

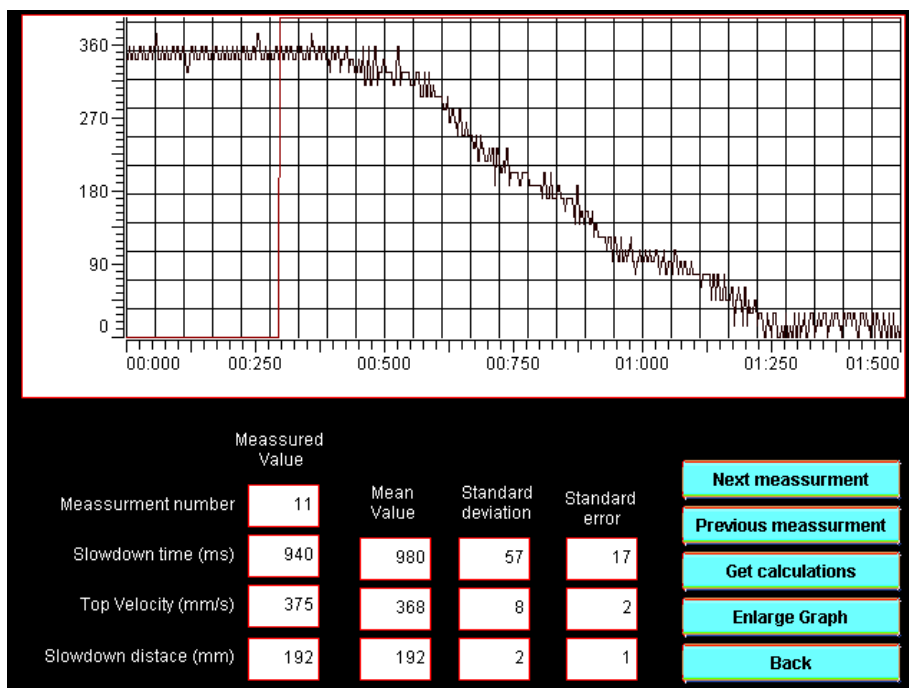
För att visa mättiderna och beräkningarna i användargränssnittet så kopieras de från antingen dess vektorer eller funktionsblock där de är lagrade från mätningarna, till tillfälliga variabler som programmets visualisering klarar av att läsa.

7.3 Användargränssnitt

För att skapa ett grafiskt användargränssnitt kan man skapa allt från B&R:s programvara Automation Studio. Det går även att ha ett flertal grafiska användargränssnitt som styrs från olika enheter. Från dessa användargränssnitt kommer man åt alla globala variabler som även används vid programmeringen av själva PLC programmet. I användargränssnittet kan man använda sig av t.ex. grafer, stapeldiagram, tårtdiagram, värdeskalor, analoga mätare, skutreglage, rullgardinsmenyer, knappar, figurer, numerisk visning, text och knappsatser så att användaren kan mata in värden manuellt. Vid planering av användargränssnittet, så bör man tänka på att det skall vara så användbart som möjligt, som det beskrivs om i kapitel 3.

Det grafiska användargränssnittet i samband med hela PLC programmet överförs och lagras i PLC:n. För att interagera med användargränssnittet som finns i PLC:n så ansluter man en Ethernet kabel från PLC:n till antingen en pekskärm eller till en dator som kör en "VNC Viewer" som det beskrivs om i kapitel 6.3.

I Figur 13 ser man en skärmbild som har blivit tagen från en "VNC viewer" i en dator som är uppkopplad till PLC:n med en Ethernet kabel. Den röda kurvan är när stoppsignalen kom från maskinen som blivit testad och den svarta kurvan är hastigheten av maskinen i mm/s. Under grafen finns information om mätningens retardationstid, topphastighet och sträckan maskinen färdas under retardationstiden. Av dessa tre värden har det även beräknats medelvärde, standardavvikelse och standardfel. En liten sidnotering av bilden är också att färgerna som man ser är inte det som man valt vid utvecklingen av användargränssnittet. Som det beskrivs i kapitel 6.3, så är detta p.g.a. användningen av enbart 256 färger, alltså 8-bitars färg.



Figur 13 Detta är en skärmbild PLC:ns visualisering tagen från dator med programvaran ”VNCviewer” uppkopplat till PLC:n med en Ethernetkabel.

7.4 Spara skärmbilder

Uppdragsgivaren ville att man skulle kunna spara bilder av hastighetsgrafer från mätningen av retardationstiden. Om man använder PLC:n med en dator och en VNC klient som det beskrivs i kapitel 6.3, så är det lätt att ta en skärmbild. Men ifall man styr användargränssnittet från en pekskärm så är det inte lika lätt.

För att lösa detta så finns det en hel del färdiga bibliotek som innehåller användbara funktionsblock som är utvecklade av B&Rs Automation Studio. Med att använda tre av dessa bibliotek och dess funktionsblock, så finns det möjlighet att ta en skärmbild av det grafiska användargränssnittet och spara bilden till ett USB-minne som ansluts till PLC:n.

För att dessa bibliotek ska få kontakt med USB-minnet så måste detta fyllas i under den fysiska PLC-konfigurationen, genom att ge USB-minnet ett bestämt namn och en sökväg till den. Namnet på USB-minnet skall ha det samma namn som är givet till USB-minnet när den ansluts till t.ex. en dator. Som sökväg till USB-minnet ger man en valfri ”buffer descriptor” vilket beskrivs i kapitel 6.2. Efter denna konfiguration så är USB-minnet med det valda namnet bunden till dess ”buffer descriptor”.

Från det första biblioteket finns det ett funktionsblock som efter aktivering och kontinuerlig körning, finner det anslutna USB-minnet till PLC:n. Funktionsblocket ställs in så det

fungerar enligt USB masslagring och så att kommunikationen sker enligt *SCSI* kommandon. *SCSI* står för ”*Small Computer System Interface*”, där CPU:n är en klient som begär den anslutna USB-enheten att göra något som att t.ex. läsa data.

Det andra biblioteket som behövs innehåller funktionsblock för att skapa mappar, filer, skriva till filer och många andra funktioner. Först skapas en ny mapp för att lagra alla bilder i en och samma mapp. Som in parameter ger man USB-minnets namn samt ett namn för mappen som skall skapas.

Det tredje och sista biblioteket med dess tre funktionsblock behövs för att spara själva skärmbilden till USB-minnet. Det första funktionsblocket reserverar minne och förbereder och skapar en ny fil att spara bilden till. Som in parameter för detta funktionsblock ges namnet för USB-minnet, namnet för visualiseringsprogrammet och filnamnet som bilden skall få. Även ett s.k. funktionshandtag behövs som in parameter. Till detta funktionshandtag sparas information om den nya tomma fil som skapats och som ges vidare till de två andra funktionsblocken i samma bibliotek.

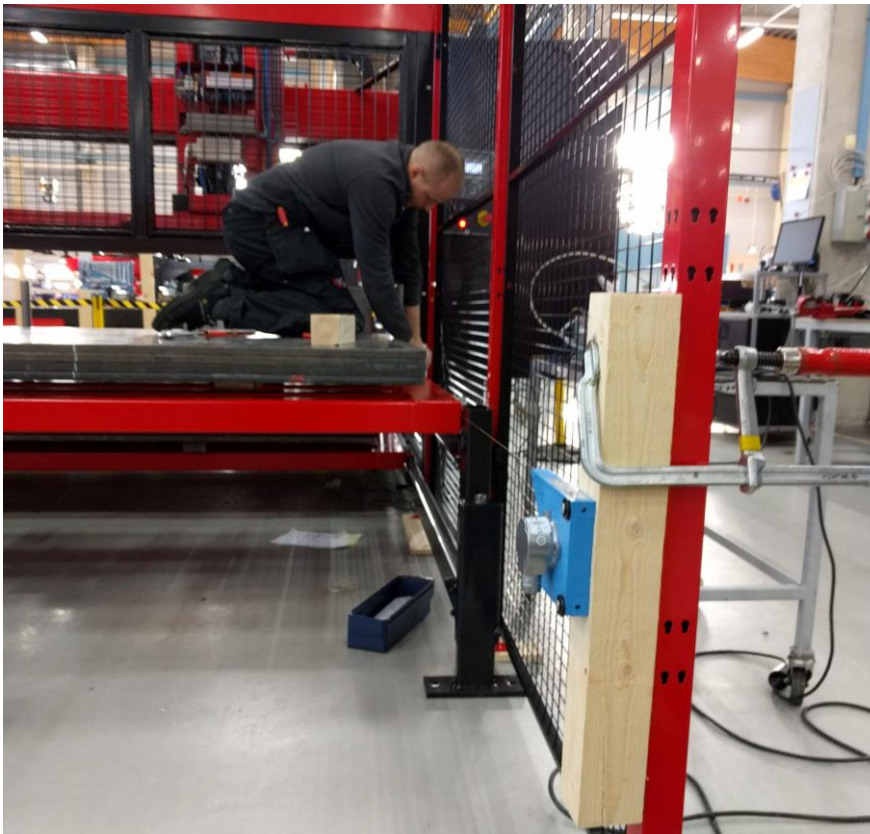
Det andra funktionsblocket sparar själva skärmbilden till filen som blev skapad i tidigare funktionsblock. Detta funktionsblock ges tidigare nämnda funktionshandtaget som in parameter. När den sparat bilden klart till USB-minnet så ges funktionshandtaget till det tredje och sista funktionsblocket som rensar minnet som reserverats för att ta skärmbilden.

7.5 Praktisk tillställning för retardationstidsmätning av en maskin med testutrustningen

När en maskin skall testas i praktiken så skall vajermekanismen vara korrekt monterad och man bör se till att man får en giltig stoppsignal.

7.5.1 Fastsättning av vajermekanismen

Till att börja med när man skall göra en retardationstidsmätning av en maskin så skall man finna en bra position att fästa vajermekanismen (från kapitel 4.3) under mätningen. Enligt tillverkarens specifikationer av vajermekanismen, så får inte vajern dras ut mer än en med en 3° lutning. I Figur 14 ser man att vajermekanismen har fästs i samma höjd som paletten kör horisontellt bort från vajermekanismen. Där gallret slutar i Figur 14 så är den position där säkerhetsvakten, eller närmare bestämt ljusriddan kommer att placeras.

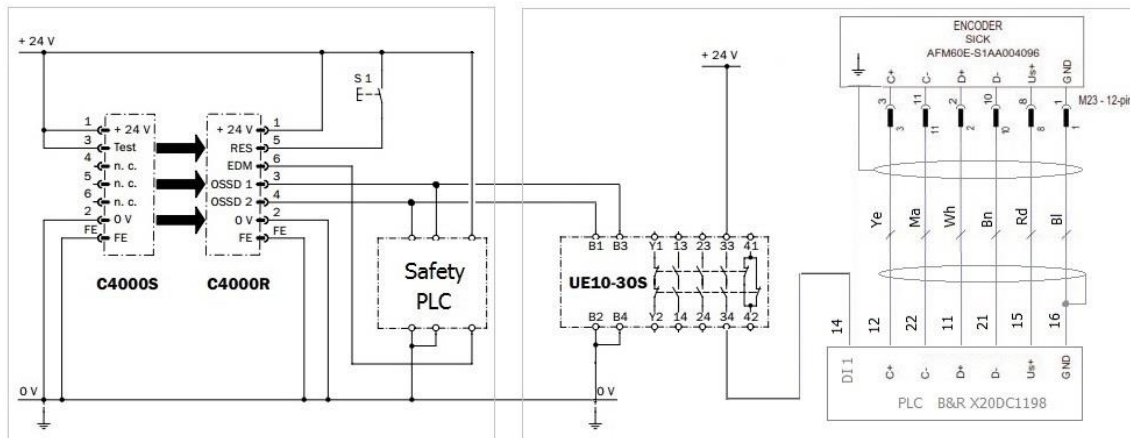


Figur 14 Vajern fästs i paletten av personen i bilden.

7.5.2 Val av stoppsignal

Vid retardationstidsmätning kan man välja att ge stoppsignalen till maskinen på olika sätt. En metod är från ett av maskinens nödstopp, medan ett annat är att direkt påverka ljusridån. För att få en giltig stoppsignal från ljusridån så bör signalen tas så nära källan som möjligt. Ljusridån ger ut två stycken ”*output signal switching device (OSSD)*” signaler för att meddela när ljusridån har blivit påverkad. Dessa två signaler kan t.ex. var kopplad till maskinens säkerhets PLC.

För att få en stoppsignal till testutrustningen tas de två OSSD signalerna, och länkas vidare till ett säkerhetsrelä som finns på testutrustningen. Från kontakterna av detta säkerhetsrelä ges en stoppsignal till testutrustningens PLC. Ett kopplingschema av detta kan man se i Figur 15. Vid maskinens normalkörning så är säkerhetsreläet draget och signalen till testutrustningen är hög. När ljusridån påverkas så bryts säkerhetsreläet, vilket leder till att signalen blir låg på digitala ingången till testutrustningen. Testutrustningens PLC kommer att tolka stoppsignaler som kommer på fallande flank.



Figur 15 Den vänstra lådan i figuren representerar komponenter för maskinen som blir testad. Ljusridåns sändare och mottagare är märkt C4000S och C4000R. I den högra lådan är komponenterna för testutrustningen som säkerhetsrelä, PLC och absolutgivare.

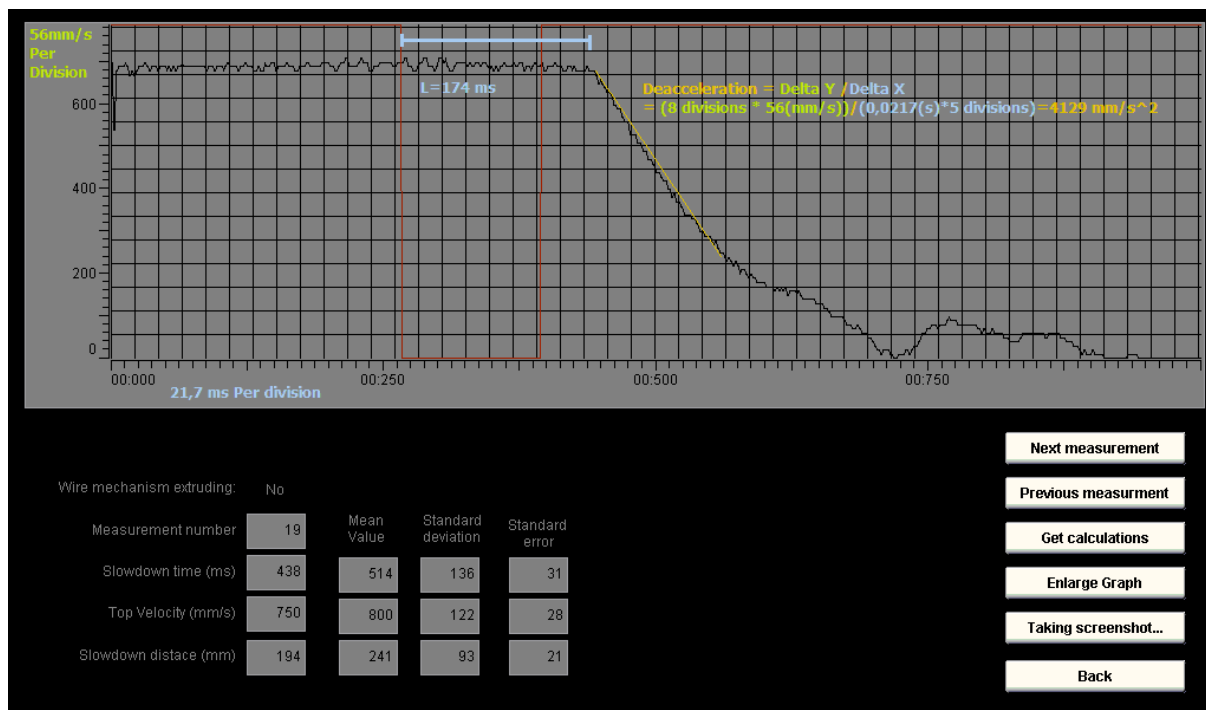
7.5.3 Parameter justering för frekvensomriktare

Ifall man redan har ett bestämt minsta möjligt avstånd för att placera ljusridån så bör man justera parametrarna för frekvensomriktaren som kör motorn, för att få en önskad retardationstid. Som det nämns i kapitel 2.3, så skall man använda medelvärdet plus tre standardavvikelser för retardationstiden i alla beräkningarna i ISO 13855:2010. Man kan börja med att göra en uppskattning att de tre standardavvikelserna kommer att vara 10 % av den tid som skall användas för beräkningarna. Denna uppskattning görs med erfarenhet från tidigare retardationstidsmätning. Genom att kombinera ekvation (1) från kapitel 2.3 och ekvation (4) från kapitel 2.4 så kan man härleda fram en tid som borde vara medelvärdet från alla mätningar, och som kan användas som parameter till frekvensomriktarens retardationstid. Detta kan beräknas enligt ekvation (7).

$$\text{retardationsstid} = \frac{S - C - K * T_1}{K} * 0,9 \quad (7)$$

Man kan börja justera parametrar till frekvensomriktaren som retardationstider, topphastigheter och retardationsaccelerationer. Genom att prova olika parametrar och verifiera stopprocedurerna med testutrustningen så blir det lättare att få fram giltiga parametrar. När den beräknade retardationstiden från ekvation (7) uppnås av maskinen, så upprepar man med ytterligare tio mätningar med de samma parametrarna, så att man kan beräkna det äkta medelvärdet + 3 standardavvikelser. Med dessa mätningar beräknar man det mista avståndet mellan farozonen och detekteringszonen med ekvation (4) från kapitel 2.4, för att se om avståndet är acceptabelt. Ifall det beräknade avståndet blir för långt så justeras parametrarna för frekvensomriktaren, och mätproceduren börjar om.

Ett exempel på detta ser man i Figur 16, där man försöker uppnå en viss retardationstid och man verifierar hur maskinen reagerar på justeringar av frekvensomriktaren. Av grafen ser man hastigheten, men man kan även beräkna och verifiera retardationen för maskinen, vilket är en parameter för frekvensomriktaren.



Figur 16 En retardationstidsmätning har blivit gjord för att verifiera en nyligt justerad parameter till frekvensomriktaren.

7.5.4 Retardationstidsmätning

Före en retardationstidsmätning påbörjas så körs maskinen först tills dess ändläge tills den har kommit till ett viloläge. Nu kan en ny mätning begäras från användargränssnittet. Till följande kör man igång maskinens rörelse som skall mätas, och när maskinens högsta hastighet är uppnådd, eller den situation då den längsta retardationstiden möjligen kan uppstå, så påverkar man antingen ljusridån eller nödstoppet. Testutrustningen loggar nu de följande tre sekunderna, och efteråt är den klar att starta om proceduren för en ny mätning. Detta repeteras minst tio gånger och slutligen så är tiden T_2 från ekvation (1), medeltalet av mätningarna plus tre standardavvikelser enligt ISO 13855:2010 som beskrivs i kapitel 2.3. Av detta kan nu beräknas ett minsta avstånd att placera ljusridån enligt ekvation (4) i kapitel 2.4.

8 Resultat

Under examensarbetet blev det planerat och valt vilka komponenter som skall användas till testutrustningen. För att mäta maskinens hastigheter användes en vajermekanism i kombination med en absolutgivare. En B&R PLC blev programmerad för att toka absolutgivarens data till en hastighet i millimeter per sekund. Retardationstidsmätningarna räknar tidsskillnaden från att ljusridån har blivit påverkad, tills att hastigheten av absolutgivaren står stilla. När retardationstidsmätningar görs så kan upp till 20 stycken mätningar sparas, av vilka medeltal, standardavvikelse och standardfel beräknas för retardationstiden, topphastigheten och retardationssträckan. Av varje retardationstidsmätning fås en skild graf som visar hastigheten under retardationstiden. Efter att minst tio mätningar har gjorts, så kan man från användargränssnittet välja att sträckor räknas ut för att placera ljusridåer enligt ISO 13855:2010. Dessa mätningar görs genom att interagera med ett grafiskt användargränssnitt via en s.k. ”VNCviewer” på en t.ex. bärbar dator. Till denna testutrustning samt användargränssnitt skrevs en tio sidig användarmanual som finns som Bilaga 1.

Två av uppdragsgivarens maskiner testades under våren 2017. Den första av dessa var en LIII som testades 27.2. Av detta test skrevs ett testprotokoll som finns som Bilaga 2. Från det första testet så var målet att se att testutrustningen fungerade som den skulle. Det blev inte ändrat några parameter för maskinen för att ändra på resultatet av retardationstiden. Maskinens frekvensomriktare hade en retardationstids parameter inställd till 0,9 sekunder. Av 12 stycken retardationstidsmätningar som gjordes så avvek två mer märkbart än de andra mätningarna, med ett totalt medelvärde på 980 millisekunder och en standardavvikelse på 54 millisekunder. Ifall man exkluderar de två retardationstidsmätningarna som avvek mer märkbart, så skulle totala medelvärdet vara 957 millisekunder och en standardavvikelse på 15 millisekunder. Orsaken till varför dessa två avvikelser förblir fortfarande oklar. I Figur 13 är en skärmbild av en av mätningarna som gjordes, där man ser hastigheten i grafen. Från kalkylationerna i Bilaga 2 så kan man konstatera att maskinens nuvarande parametrar bör ändras för det aktuella avståndet 1400mm mellan ljusridån och maskinen kan behållas.

Den andra maskinen som testades var en ASF-EU som testades 24.3. Till detta skrevs det även ett testprotokoll som finns som Bilaga 3. Eftersom att ljusridån var placerad 1600mm från maskinen, så kalkylerades det att retardationstiden borde vara $< 0,4$ sekunder. Av första retardationstidsmätningen konstaterades tiden vara 0,77 sekunder. Maskinens parametrar

justerades flera gånger under dagen för att minska på retardationstiden. Slutligen nåddes en retardationstid på 0,388 sekunder. Eftersom det började bli brist av tiden på den dagen så fanns det inte tid att göra tio mätningar till, för att räkna ut hur stor standardavvikelsen blir. I Figur 16 är en skärmbild av en av mätningarna, där man ser hastigheten i grafen.

9 Diskussion

Arbetet påbörjades redan under hösten 2016 med att fundera ut vad man kunde använda sig av för komponenter. De komponenter som väljs skulle finnas i arbetsgivarens komponentlager från förut. Det beslöts i ett tidigt skede att vajermekanismen i kapitel 4.2 skulle användas. Vajermekanismen har en absolutgivare som kommer färdigt monterat på den och som fungerar med ett SSI protokoll. PLC:n av märket B&R som även fanns i komponentlagret, har möjligheten för en SSI tilläggsmodule. Det kom på diskussion att man eventuellt kunde byta ut absolutgivaren till en vanlig pulsgivare, eftersom då kunde en Mitsubishi PLC användas istället. Orsaken till detta var att det skulle vara lättare att använda sig av Mitsubishis PLC. Det valdes ändå till slut att B&R:s PLC skulle användas.

Det blev inte så stora framsteg från början av hösten 2016. Efter att det i november 2016 beslöts vilka komponenter som skulle användas, så kunde den störta delen av examensarbetet påbörjas, nämligen programmeringen.

Programmeringsspråket som använts från tidigare till B&R:s PLC av uppdragsgivaren var strukturerad text. Eftersom jag inte var så bekant med detta programmeringsspråk från tidigare, så studerade jag ett tidigare programmerat program till en av uppdragsgivarens andra maskiner. Ganska fort kunde jag konstatera att programmeringsspråket påminner mycket på andra programmeringsspråk som jag kunde från tidigare. Programmeringen gick överlag väldigt bra under examensarbetet. Det ända större problemet var när det uppstod ”*integer overflow*” av absolutgivarens räkning, vilket det beskrivs om i kapitel 7.1.3.

Eftersom det fanns lite tid till att testa utrustningen med uppdragsgivarens maskiner så är det lite svårt att tolka hur noggrant utrustningen mäter. Efter att några fler tester blir gjorda så kanske man kan konstatera hur noga resultaten blir. Som det nämnts om tidigare i kapitel 1.3.3, så finns det andra som har gjort liknade testutrustningar. Det som skulle vara bra någon gång i framtiden är att hitta någon annan som har liknade testutrustning och få jämföra resultaten från båda testutrustningarna för att kunna konstatera ifall man får samma resultat.

Resultatet av examensarbetet uppfyller nästan alla mål, som beskrivs i kapitel 1.1.2. Vid skrivandes stund av examensarbetet så är programmeringen klar och har det blivit utfört test av uppdragsgivarens maskiner med hjälp av testutrustningen som har utvecklats. Det som inte ännu är klart är att designa en låda att montera testutrustningens alla komponenter i. Det påbörjades aldrig någon undersökning om delmålen i kapitel 1.1.3 som uppdragsgivaren gav, eftersom att det blev ont om tid under våren 2017.

Något som jag skulle göra annorlunda om jag skulle börja om på nytt, skulle vara att lägga mer tid på examensarbetet under hösten 2016 istället för att göra majoriteten av examensarbetet våren 2017. Det är planerat att man skall lägga cirka 400 timmar på examensarbetet. Jag har under examensarbetets gång försökt hålla en dagbok på tider som jag har lagt ner. I skrivandes stund så har cirka 350 timmar lagts på examensarbetet. Ungefär hälften av denna tid är lagt på den praktiska delen av examensarbetet.

Det har varit intressant att bekanta sig med ett nytt programmeringsspråk och att lära sig en hel del andra nya saker. Jag är själv nöjd med resultatet och arbetsgivaren värkar också vara det. Jag kommer att börja jobba vid LKI Källdman efter att jag utexamineras. Då kommer även det som fortfarande är oklart av testutrustningen att göras klart.

10 Källförteckning

B&R Automation, 2017. *Automation Studio*. [Online]
Available at: https://rebrand.ly/Automation_Studio
[Använd 24 Februari 2017].

B&R Automation, 2017. *Control and I/O systems; X20DC1198*. [Online]
Available at: <https://rebrand.ly/X20DC1198>
[Använd 2 Mars 2017].

B&R Automation, 2017. *X20 Systems*. [Online]
Available at: <https://rebrand.ly/X20CP1585>
[Använd 24 Februari 2017].

Björklund, D., 2015. *Kvalitetssäkring av ASF-EU genom analys och optimering av testrutiner*, Vasa: Lärdomsprov inom elektroteknik. Yrkehögskolan Novia.

Eitel, E., 2014. *Machinedesign*. [Online]
Available at: https://rebrand.ly/Basics_of_Rotary_Encoders
[Använd 19 Februari 2017].

Electro-Craft Corporation, 1980. *DC motors, speed controls, Servo systems : an engineering handbook*. u.o.:Elsevier Ltd.

FRABA Inc., 2013. *Implementation of SSI Master interface*, Hamilton: u.n.

ISO 13855:2010, 2010. *Safety of machinery - Positioning of safeguards with respect to the approach speeds of parts of the human body (ISO 13855:2010)*, u.o.: u.n.

John, K.-H. & Tiegelkamp, M., 2010. *IEC 61131-3: Programming Industrial Automation Systems*. u.o.:Springer-Verlag Berlin Heidelberg.

Kivistö-Rahnasto, J., 2000. *Machine safety design; An approach fulfilling European safety requirements*. Tampere: Technical Research Centre of Finland, VTT Publications.

Leuze electronic, 2017. *Safety of machinery, risk analysis and risk assessment*. [Online]
Available at: https://rebrand.ly/risk_analysis_and_risk_assessment
[Använd 27 Januari 2017].

LKI Källdman, 2017. *History of LKI Källdman*. [Online]
Available at: https://rebrand.ly/LKI_History
[Använd 2 Mars 2017].

Microchip Technology, 2009. *PIC18F4550 Data Sheet*. [Online]
Available at: <https://rebrand.ly/PIC18F4550>
[Använd 8 Mars 2017].

Nielsen, J., 1993. *Usability Engineering*. u.o.:Toppan Publishing.

PLCopen, 2013. *Introduction into IEC 61131-3 Programming Languages*. [Online]
Available at: <https://rebrand.ly/IEC-61131-3>
[Använd 1 Mars 2017].

Richardson, T., Stafford-Fraser, Q., Wood, K. R. & Hopper, A., 1998. *Virtual Network Computing*, u.o.: u.n.

Sick AG, 2017. *Absolute encoders*. [Online]
Available at: <https://rebrand.ly/AFM60E-S1AA004096>
[Använd 27 Februari 2017].

Sick AG, 2017. *Accessories*. [Online]
Available at: <https://rebrand.ly/MRA-G130-105D3>
[Använd 27 Februari 2017].



**Manual for test equipment
to measure stop time**



Contents

1. Mounting of wire draw mechanism.....	2
2. Wiring of test equipment.....	3
3. Predetermined minimum distance	3
3.1. Example 1.....	4
4. Connect to the PLC as a VNC client	4
5. Change settings of the test equipment	4
6. Perform new measurements	5
6.1. Undo a requested measurement.....	6
6.2. Delete previously performed measurement	6
6.3. Delete all measurements.....	6
7. View performed measurements	6
8. Take screenshot and save it to a USB device	7
9. Get calculations of the minimum distance.....	8
9.1. Choose a calculation type	8
9.2. Choose which time to use	8
9.3. Insert parameters.....	9
9.4. Result page of calculations.....	9

1. Mounting of wire draw mechanism

The wire draw mechanism shall be mounted orthogonally towards a hazard zone, as can be seen in Figure 1. According to the manufactures specification, the wire should be pulled out with an inclination $< 3^\circ$.

The wire must be fasten properly to the machine so the wire cannot snap back. If the wire is able to snap back the wire draw mechanism may be harmed for good.

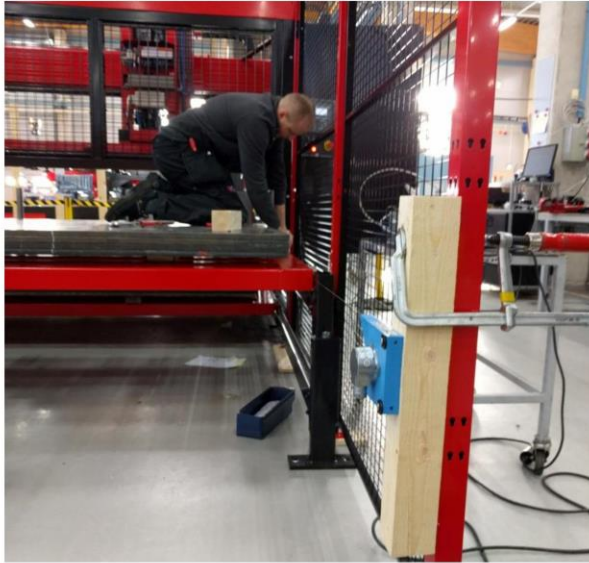


Figure 1 The wire draw mechanism is mounted to measure movement of the pallet changer orthogonally away from the wire draw mechanism.

2. Wiring of test equipment

A stop command can be given to the Light Beam Safety Device or to the emergency stop. The stop signal shall be taken from the Light Beam Safety Device's OSSD signals and connected to the test equipment's safety relay. The test equipment's PLC detects stop signals on the digital input that are of falling edge, so the signal should be coming from a normally open contact of the safety relay, as can be seen in Figure 2.

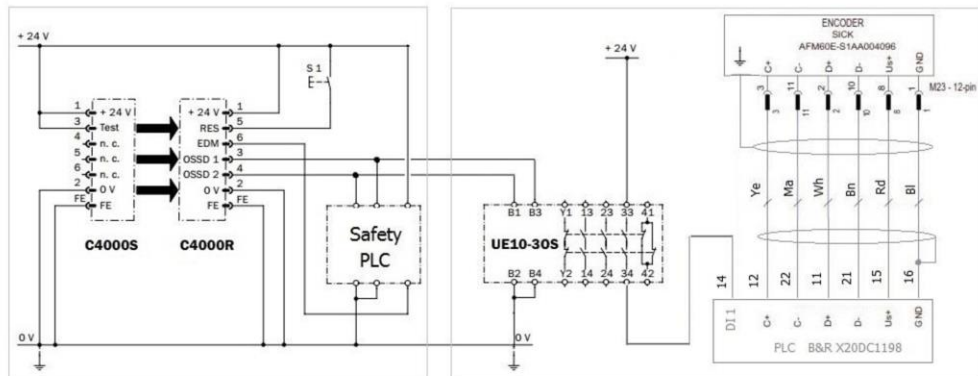


Figure 2 In the left box is the machine that is tested. In the box to the right is the components of the test equipment. The absolute encoder AFM60E-S1AA004096 is wired to the module PLC module X20DC1198 which also gets the stop signal from the safety relay UE10-30S.

3. Predetermined minimum distance

If no minimum distance between the hazard zone and the detection zone have been determined, this section can be skipped and continue from chapter 4.

When a minimum distance S is predefined, T_2 can be calculated per equation (1)

$$S = (K * (T_1 + T_2)) + C \rightarrow T_2 = \frac{S - C - K * T_1}{K} \quad (1)$$

Since T_2 shall equal to the mean + 3 standard deviations, it can be assumed that the 3 standard deviations is roughly 10 % of T_2 . The a maximum ramp-down time parameter can then be calculated according to equation (2)

$$\text{rampdown time} = \frac{S - C - K * T_1}{K} * 0,9 \quad (2)$$



3.1. Example 1

The Light Beam Safety Device sender "M40S-034000AR0" and receiver: "M40E-034000RT0" are placed 1500mm from the hazard zone. According to the manufactures specification the response time is 10ms and the beam separation is 400mm. The stop signal from the Light Beam Safety Device is given to a safety relay "Sick UE10-30S" which has a response time of 10ms.

Since the detection capability of the Light Beam Safety Device is 400mm, calculations shall be made according multiple separate beams where $K = 1600\text{mm/s}$ and $C = 850\text{mm}$. T_1 is the time it takes to give a stop signal, after the safeguard has been actuated, i.e. the combined response time of the safety equipment. The maximum ramp-down time parameter can then be calculated per equation (3).

$$\text{rampdown time} = \frac{1500\text{mm} - 850\text{mm} - (1600\text{mm/s} * 0,02\text{s})}{1600\text{mm/s}} * 0,9 = 0,348\text{s} \quad (3)$$

4. Connect to the PLC as a VNC client

To connect to the PLC, the PC's Ethernet IPV4 address must be 192.168.10.X where "X" can be any number except 123. To change TCP/IP settings on a Windows PC, follow these steps: Go to network connections → Right click on Local Area Connection i.e. Ethernet and select properties → Double click IPV4.

The PLC user interface can be accessed from a computer using a software called "ThightVNC". In the properties of the software, check the option to force 256-color display. This should be used for slow connections with limited bandwidth. To connect to the PLC, enter IP address with port number: 192.168.10.123:5900

5. Change settings of the test equipment

Parameters of the equipment used during the measurements is possible to change in the settings menu, which can be accessed from the settings button on main menu, as seen in Figure 3.

Specification of the encoder and the wire draw mechanism can be found from the manufacturer.

Since the velocity measured with the encoder and wire draw mechanism has some fluctuating noise, two parameters can be adjusted to counteract that to get a trustworthy measured stop time. If the measured velocity is bellow a predefined velocity during a predefined amount of time, the movement of the wire draw mechanism can be considered as stopped.

6. Perform new measurements

The worst-case scenario for the stopping time of the machine should be used (to the extent it is realistic). When determining this scenario, factors such as tool weight, temperature, switching times of valves and ageing of components should be considered. In most cases, the worst-case scenario is when the maximum speed of the machine occurs while transporting maximum load.

When the machine is in a position ready to perform a new measurement, a new measurement can be requested by pressing the button "Get new measurement", as can be seen in Figure 3. Now the machine can be started. When the worst-case scenario has been achieved i.e. the maximum speed, a stop command can be given either to the Light Beam Safety Device or the emergency stop. Data will then be logged for the next three seconds.

Once the button label has changed from "Measurement in progress" to "Get new measurement", the measurement is done and the procedure can be repeated for another new measurement. Above the button "Get new measurement" there is a counter of how many measurements have been made, as can be seen in Figure 3.

One measurement is not sufficient for calculating the minimum distance. At least 10 measurements are required. However, the max number of measurements is limited to 20. A statistical way of covering 99,7 % of all individuals in a normally distributed population is to calculate the mean value + 3 standard deviations. The highest measured value or the mean plus three standard deviations, whichever is the greater, should be used in the calculation of the minimum distance.

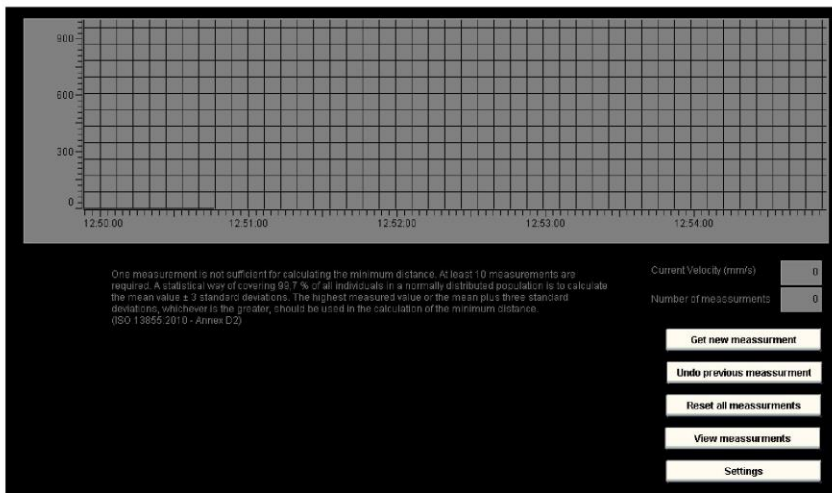


Figure 3 the main menu of the user interface. The graph in the picture is a continuous online graph that simply illustrates the current speed in mm/s of the encoder.



6.1. Undo a requested measurement

Once the button "Get new measurement" has been pressed, the label of the button below will change to "Undo". If no new measurement was wanted, pressing "Undo" will stop the current measurement procedure.

6.2. Delete previously performed measurement

If previous measurement was performed inadequately, it is possible to delete it by pressing the button "Undo previous measurement", which can be seen in Figure 3.

6.3. Delete all measurements

If one would like to start from scratch, pressing the button "Reset all measurements" removes all currently saved data from the measurements that have been done so far.

7. View preformed measurements

Once one or more measurements have been made, graphs, measured values and statistical calculations can be examined by pressing the button "View measurements" in the main menu, as seen in Figure 4.

By pressing the buttons "Previous measurement" and "Next measurement", data from different measurements will update in the graph and the numeric boxes.

Below the graph is following information:

- Wire mechanism extruding. This will be displayed as true or false of which direction the wire was traveling during the measurement.
- Measurement number. When pressing the "Previous measurement" and "Next measurement", this numeric value will indicate which measurement attempt is currently displayed.
- Slowdown time. This is the time in milliseconds between the incoming stop signal until a minimum velocity considered as a stop has been reached.
- Top velocity. The highest reached velocity during the measurement.
- Slowdown distance. This is the distance that the wire mechanism has traveled during the measurement.

There are four columns of data displayed below the graph

- In the first column is the value from the specific measurement chosen by pressing the buttons "Previous measurement" and "Next measurement".
- In the second column is the mean value of the all the previous measurements, up until the currently displayed measurement.
- In the third column is the standard deviation of the all the previous measurements, up until the currently displayed measurement.
- In the fourth column is the standard error of the all the previous measurements, up until the currently displayed measurement.

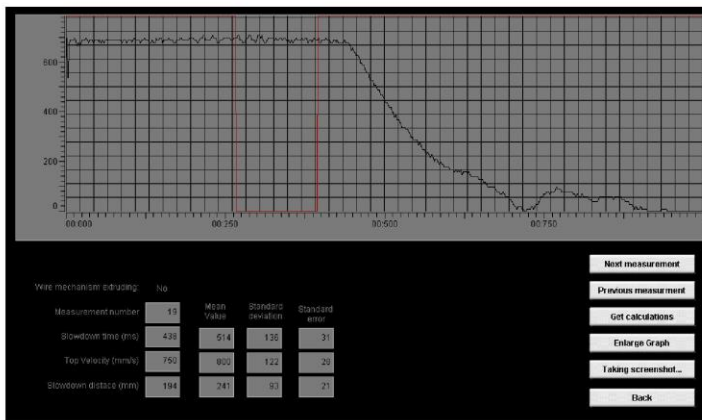


Figure 4 Example of how a measurement might look like. The Graph displays the velocity of the encoder in mm/s. The red trend is the stop signal given from the machine that is being tested.

8. Take screenshot and save it to a USB device

Screenshots can be captioned and saved to a USB device connected to the PLC by pressing the button "Screenshot". After pressing the button the screenshot procedure takes roughly 15 seconds whereupon a message is displayed if the process was successful.

In order for the PLC to recognize the USB device, the device name must be "DEVUSB" and the device must be formatted as a FAT32 file system, as can be seen in Figure 5.

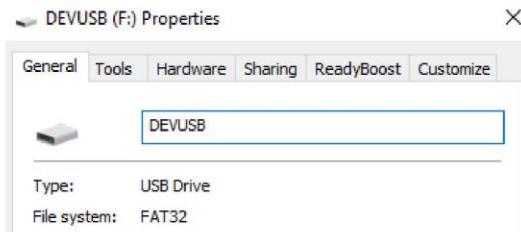


Figure 5 Properties for a USB device when plugged in to a Windows computer.

9. Get calculations of the minimum distance

Once more than ten measurements have been made (as mentioned in chapter 6), a reliable minimum distance can be selected by pressing the button "Get calculations" on the page where the measurements graphs are displayed, as can be seen in Figure 4.

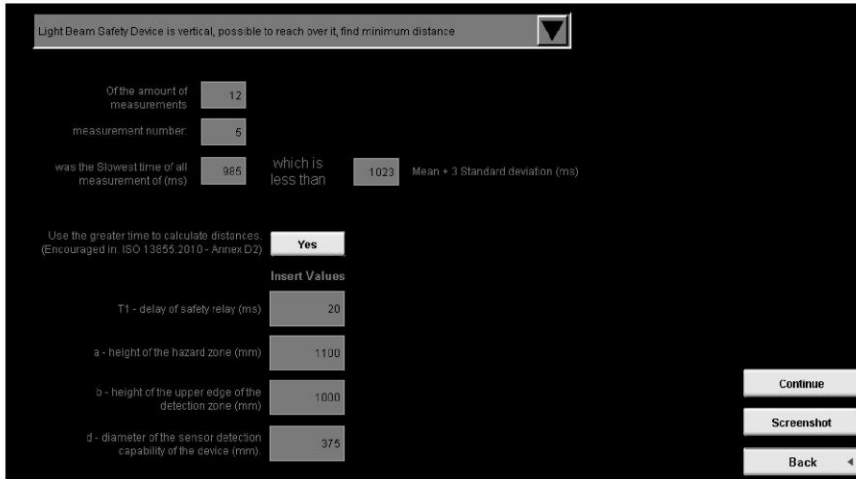


Figure 6 Example of how it looks like when choosing which calculations to make.

9.1. Choose a calculation type

Depending on which setup of safeguards is used, there are different calculations for them. By pressing the dropdown menu in the top of the screen, a calculation of your choice can be made, as can be seen in Figure 6.

- Light Beam Safety Device is parallel to ground, Find minimum distance.
- Light Beam Safety Device is parallel to ground, Find minimum height.
- Light Beam Safety Device is parallel to ground, Find minimum detection diameter.
- Light Beam Safety Device is vertical, not possible to reach over it, find minimum distance.
- Light Beam Safety Device is vertical, possible to reach over it, find minimum distance.
- Light Beam Safety Device is vertical, possible to reach over it, find minimum LBSD height.
- Laser scanner, find minimum distance.

9.2. Choose which time to use

An evaluation is displayed on the user interface where the slowest measured time is compared with the mean + 3 standard deviations. In ISO 13855:2010 it is mentioned that the greater time of these two shall be used as the time to calculate the minimum distance.

By leaving the option "Use the greater time to calculate distances" as a yes, will automatically calculate with the greater time.

9.3. Insert parameters

After choosing an option as described in chapter 9.1, several input parameters can be entered. If a "not used" label is displayed beside a box, simply just skip those and only enter parameters with valid labels. The possible parameters are:

- Time T_1 . As also described in example 1 in chapter 3.1, is the time it takes to give a stop signal, after the safeguard has been actuated.
- A-height. This is the height of the hazardous moving machine part.
- B-height. The height measured from the floor to the upper most beam of the Light Beam Safety Device.
- D-diameter. This is the minimum detection capability of the Light Beam Safety Device.

After entering the required parameters, the results will be given by pressing the button "continue", as can be seen in Figure 6.

9.4. Result page of calculations

An example of how a result might look like can be seen in Figure 7. The calculation performed was on a vertically mounted Light Beam Safety Device with the possibility to reach over it. Two different calculations were made, reach through and reach over. The result is that the distance calculated from reaching over is greater than reaching through, therefore the S_{RO} distance shall be used as the minimum distance to place the vertically mounted Light Beam Safety Device from the hazard zone. From Figure 7, $S_{RO} = 1487\text{mm} + 321\text{mm} = 1808\text{mm}$.

	minimum distance (mm)	+3 standard deviation (mm)	ISO 13855:2010 Equation number:	T1 (ms)	T2 (ms)	T total (ms)	+3 standard deviation (ms)
(SRT) reach through detection zone of Electro-sensitive protective equipment	1681	321	Equation (6) $S = (1600 \times T) + 850$	20	980	1000	23
(SRO) if possible circumventing of electro-sensitive protective equipment by reaching over the detection zone	1487	321	Equation (12) $S = (1600 \times T) + CRO$				
Not used	0						
CRO - value from table 1 in ISO 13855:2010	750						
Result: Use SRO distance, since reaching over is greater than reaching through							

Figure 7 The result page displays the calculated minimum distance to place the Light Beam Safety Device.



Date: (27.2.2017)

Stop time measurement of LIII

The aim of this test is to measure the stopping time of:

- A loaded pallet changer traveling at its maximum speed horizontally orthogonally towards a vertically mounted light guard.

In addition to measuring the stopping time, the motion of the measurements is recorded to a graph in mm/s.

Pallet changer inverter parameters: Ramp-down time: 0,9 s
Maximum sheet weight used during testing: 6000 kg
Height of hazard zone, a \approx 1000 mm

Equipment used

The speed sensor used is a 5m wire draw mechanism "MRA-G130-105D3" attached to an absolute encoder "AFM60E-S1AA004096" with a total deviation of $\pm 3,2$ mm. The signals are recorded using a VNC viewer connected to a "X20CP1585" PLC with a measurement cyclic time of 4ms and a tolerance of ± 0 ms. The test equipment is configured per the manufactures specifications that the wire mechanism draw length is 385 mm per revolution, and the absolute encoder has a resolution of 4096 steps per revolution. The speed or velocity, v is calculated a following:

$$v = \frac{(\text{wiredrawlength} / \text{revolution}) * (\Delta \text{encoder steps} / \text{measuring cycle})}{(\text{encoder steps} / \text{revolution}) * \text{measuring cyclic time}}$$

Stopping command used for this test was from an emergency stop connected to a safety relay "Sick UE10-30S" with a response time of 10ms.

Light curtain sender and receiver: "M40S-034000ARO" and "M40E-034000RTO"

Response time: 10ms

Height of light guard, b: 972 mm

Number of beams: 3

Beam separation: 400 mm

Beam diameter: 23mm

i.e. Detection capability: Beam separation- Beam diameter=377mm

Possible to reach over the light curtain: Yes



Measurement results

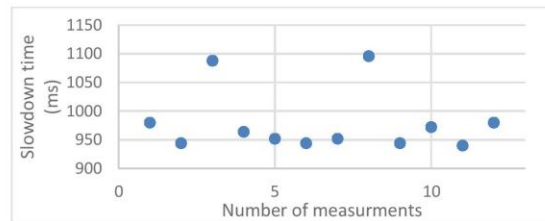
Measurement number	Slowdown time (ms)	Top velocity (mm/s)	Distance of slowdown motion (mm)
1	980	359	189
2	944	375	192
3	1088	359	194
4	964	359	190
5	952	375	192
6	944	375	189
7	952	359	193
8	1096	375	194
9	944	375	192
10	972	359	192
11	940	375	192
12	980	359	191

Slowdown time:

Mean (ms): 979,67

Standard Deviation (ms): 54,35

Standard Error (ms): 15,69



A statistical way of covering 99,7 % of all individuals in a normally distributed population is to calculate the mean value ± 3 standard deviations.

$$T_{tot} = T_1 + T_2$$

T_1 = response time of light guard and safety relay: 10ms + 10ms

$$T_{tot} = (10ms + 10ms) + 980ms + (3 * 54ms) \approx 1163ms \approx 1,163s$$



Calculations according to ISO 13855:2010

Distance reach through:

$$\text{Equation 2} \rightarrow S_{RT} = (K * T) + C_{RT}$$

Detection capability: 377mm

Calculate according to chapter 6.2.5-multiple beams i.e. Equation 5.

$$\rightarrow S_{RT} = (1600mm * T_{tot}) + 850mm$$

$$\rightarrow S_{RT} = (1600mm * 1,163s) + 850mm \approx 2710mm$$

Since it is possible to reach over the detection zone, calculations are made according to chapter 6.5 Equation 12.

$$\rightarrow S_{RO} = (1600mm * T_{tot}) + C_{RO}$$

C_{RO} value can be found in chapter 6.5 table 1.

Height of detection zone, b: 972 mm

Height of hazard zone, a \approx 1000 mm

$$\rightarrow C_{RO} = 1200mm$$

$$S_{RO} = (1600mm * 1,163s) + 1200mm \approx 3060mm$$

Conclusion of calculation

Since $S_{RO} > S_{RT}$, the $S_{RO} \approx 3060mm$ is the minimum distance to place the detection equipment from the hazard zone.

Currently the detection zone is placed 1400 mm from the hazard zone. To keep this distance, changes must be made.

Solutions

There are two possible solutions to keep the detection zone at the distance $S_{RT} = S_{RO} = 1400mm$.

Option 1:

T_2 shall be the mean + 3 standard deviations. The 3 standard deviations can be estimated to be roughly 10 % of T_2 .

The parameter for the inverter maximum ramp-down time, RDT can be calculated from equation 5:

$$S_{RT} = (1600mm * (T_1 + T_2)) + 850mm$$

$$\rightarrow T_2 = \frac{S_{RT} - 850mm - (1600mm * T_1)}{1600mm}$$

$$RDT = \frac{1400mm - 850mm - (1600mm * 0,02s)}{1600mm} * 0,9 \leq 0,291s$$



Required height of the detection zone can be calculated:

$$S_{RO} = 1400mm$$

$$\text{Using } S_{RT} = (1600 * T_{Tot}) + C_{RT}$$

Setting $S_{RO} = S_{RT}$, which implies $C_{RO} = C_{RT} = 850mm$.

Height of hazard zone, $a \approx 1000$ mm

Minimum height of a safeguard, b can be found from ISO 13855:2010 chapter 6.5 table 1. Using the safer alternative, the minimum height of the safeguard, b should be 1300mm.

Option 2:

$$S_{RT} = S_{RO} = 1400mm.$$

Choose a safeguard, b with a detection capability, $d \leq 40mm$, and equation 4 can be used. T_2 shall be the mean + 3 standard deviations. The 3 standard deviations can be estimated to be roughly 10 % of T_2 . The parameter for the inverter maximum ramp-down time, RDT can be calculated from equation 4:

$$S_{RT} = (1600 * (T_1 + T_2)) + 8 * (d - 14)$$

$$T_2 = \frac{S_{RT} - (8(d - 14)) - (1600mm * T_1)}{1600mm}$$

$$RDT = \frac{1400mm - (8(14 \dots 40mm - 14)) - (1600mm * 0,02s)}{1600mm} * 0,9$$

$$RDT = 0,652 \dots 0,769s$$

Setting $S_{RO} = S_{RT}$, which implies,

$$C_{RO} = C_{RT} = 8((14 \dots 40mm) - 14mm) = 0 \dots 208mm$$

Height of hazard zone, $a \approx 1000$ mm

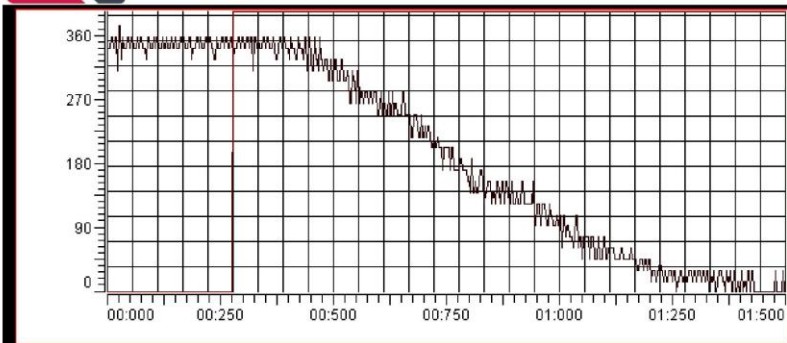
Minimum height of a safeguard, b can be found from ISO 13855:2010 chapter 6.5 table 1. Using the safer alternative, the minimum height of the safeguard, b should be 1600mm.

Testing by

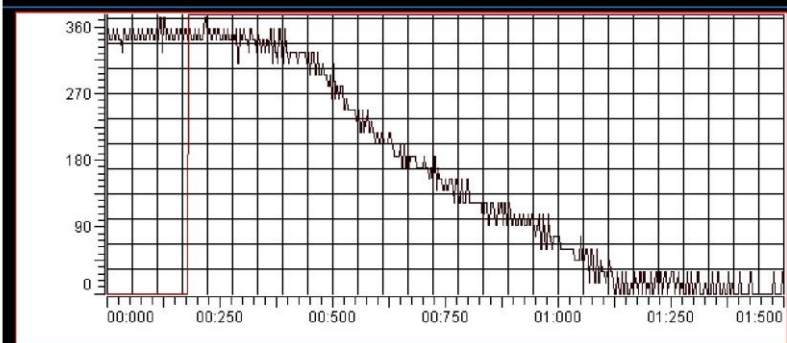
Dan, Nka 27.2.2017

Appendix

Screenshots of the measurements made.



	Measured Value	Mean Value	Standard deviation	Standard error	
Measurement number	1				Next measurement
Slowdown time (ms)	980	0	0	0	Previous measurement
Top Velocity (mm/s)	359	0	0	0	Get calculations
Slowdown distance (mm)	189	0	0	0	Enlarge Graph
					Back



	Measured Value	Mean Value	Standard deviation	Standard error	
Measurement number	2				Next measurement
Slowdown time (ms)	944	962	25	18	Previous measurement
Top Velocity (mm/s)	375	367	11	8	Get calculations
Slowdown distance (mm)	192	190	2	1	Enlarge Graph
					Back



TEST PROTOCOL - ASF-EU 2nd station [1]

Date: (24.3.2017)

Stop time measurement of ASF-EU 2nd station

The aim of this test is to measure the stopping time of:

- Cutting pallet traveling from ASF-EU towards 2nd station orthogonally to a vertically mounted light beam safety device.

In addition to measuring the stopping time, the motion of the measurements is recorded to a graph in mm/s.

Height of hazard zone, a ≈1000 mm

Equipment used

The speed sensor used is a 5m wire draw mechanism "MRA-G130-105D3" attached to an absolute encoder "AFM60E-S1AA004096" with a total deviation of ±3,2 mm. The signals are recorded using a "X20CP1585" PLC with a measurement cyclic time of 2ms and a tolerance of ±0ms. The test equipment is configured per the manufactures specifications that the wire mechanism draw length is 385 mm per revolution, and the absolute encoder has a resolution of 4096 steps per revolution. The speed or velocity, v is calculated as following:

$$v = \frac{(\text{wiredrawlength} / \text{revolution}) * (\Delta \text{encoder steps} / \text{measuring cycle})}{(\text{encoder steps} / \text{revolution}) * \text{measuring cyclic time}}$$

The stopping command was given to the light beam safety device sender and receiver: "M40S-034000ARO" and "M40E-034000RTO", which has the specifications:

A response time of 10ms, height of light beam safety device is 972 mm and a beam separation of 400 mm. It is possible to reach over the detection zone.

The light beam safety device's OSSD signals were connected to a safety relay "Sick UE10-30S" with a response time of 10ms. The measurement equipment's PLC gets the stop signal from the safety relays contacts.



Measurement requirements

A statistical way of covering 99,73 % of all individuals in a normally distributed population is to calculate the mean value ± 3 standard deviations. It is estimated that the 3 standard deviations is roughly 10 % of the measured stop time.

$$T_{tot} = T_1 + T_2$$

$T_1 = \text{response time of safety equipment: } 10\text{ms} + 10\text{ms}$

$$\text{Equation 2 from ISO 13855:2010} \rightarrow S = (K * T_{tot}) + C$$

$$\rightarrow T_2 = \frac{S - C - (K * T_1)}{K}$$

$$\text{Mean stop time} = \frac{S - C - (K * T_1)}{K} * 0,9$$

The light beam safety device around ASF-EU 2nd station is placed 1600mm from the hazard zone. The light beam safety device can be classified to the category multiple beams, therefore $K = 1600\text{mm/s}$ and $C = 850\text{mm}$.

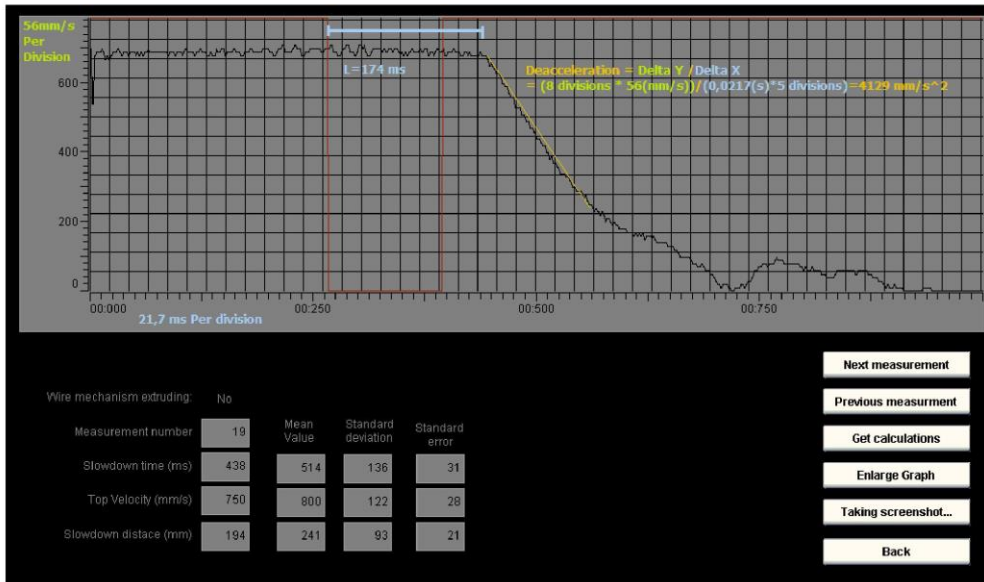
$$\text{Mean stop time} = \frac{1600\text{mm} - 850\text{mm} - (1600\text{mm/s} * 0,02\text{s})}{1600\text{mm/s}} * 0,9$$

The mean stop time should be $< 0,4\text{ s}$



Stop time measurement and inverter parameter adjustment

The inverter has a delay, L of 160 ... 170 ms after it has been given a stop signal. The inverter parameters top speed and deceleration are adjusted to achieve the required stop time of < 0,4 s.



Testing by

DAn, PSj, ESm - 24.3.2017