HÖGSKOLAN VÄST

2011-06-02

USB 3 jämfört med 2.0 – En jämförande studie om bulköverföringar

Toni Wissmach
toni.wissmach@student.hv.se

EXAMENSARBETE

Data- och systemvetenskapliga programmet med inriktning mot nätverk

EXAMENSARBETE

Jämförning av bulköverföringar

Sammanfattning

Fokuset i denna studie ligger huvudsakligen på en av de fyra överföringstyper som USB 2 och 3 använder sig av, bulköverföringar. Från den valda typen av överföring kommer tider och hastighet mätas i tolv testfall. Alla testfall delas in i två grupper ett där mätningarna genomförs under tiden media strömmas ifrån en annan USB-enhet för att belasta USB-bussen. Och ett annat där bussen är obelastat. Värden visas och jämförs sedan med hjälp av grafer och histogram där medelvärden, standardavvikelser samt variationskoefficienten har tagits fram.

Resultatet i denna studie visar att USB 3 en högre variationskoefficient i vissa lägen. Mätvärden visar att USB 3 har ett lägre variationskoefficientvärde när testdatan består av många små filer än vad det gör för USB 2. Dock ökar variationskoefficienten för USB 3 när filstorleken blir större jämfört med USB 2 som påvisar en sjunkande variationskoefficient.

Datum: 2011-06-02

Författare: Toni Wissmach

Examinator: Linn Gustavsson Christiernin

Handledare: Christian Ohlsson

Program: Data- och systemvetenskapliga programmet med inriktning mot nätverk

Huvudområde: Datateknik

Utbildningsnivå: Grundnivå/avancerad nivå

Poäng: 180 högskolepoäng

Kurskod: EXC570

Nyckelord: *Superspeed USB, Fullspeed USB, tidsmätning, standardavvikelse, läs- och skrivning till hårddisk, USB-arkitektur*

Utgivare: Högskolan Väst, Institutionen för ekonomi och IT,
461 86 Trollhättan
Tel: 0520-22 30 00 Fax: 0520-22 32 99 Web: www.hv.se

BACHELOR'S THESIS

Jämförning av bulköverföringar

Summary

The focus in this study is on bulk transfer which is one of four transfer types that USB 2 and 3 uses. Due to the chosen transfer type the study contains twelve test cases which are measuring transmission time and bandwidth. Those test cases are later on categorized in two different groups. One is group measuring time and bandwidth while no media is simultaneously streaming to allocate bandwidth on the USB bus. Another one when bandwidth will be allocated on the bus. The result is displayed using graphs and bar charts of which the mean, standard deviation and coefficient of variation is calculated.

The result of this study shows that in some cases USB 3 does not achieve the same low level of values compared to USB 2. The measured values show that USB 3 has a smaller coefficient of variation when test data consists of many small files than for USB 2. However, C.O.V. grows larger for USB 3 when the file size increases compared USB 2 in which case shows decreasing values.

Date:	June 02, 2011
Author:	Toni Wissmach
Examiner:	Linn Gustavsson Christiernin
Advisor:	Christian Ohlsson
Programme:	Applied Computer and Systems Science Program – Data Communication and Networks
Main field of study:	Computer Engineering
Education level:	First cycle
Credits:	180 HE credits
Course code:	EXC570
Keywords	*Superspeed USB, Fullspeed USB, time measuring, standard deviation, read and write to hard drive, USB architecture*
Publisher:	University West, Department of Economics and IT SE-461 86 Trollhättan, SWEDEN Phone: + 46 520 22 30 00 Fax: + 46 520 22 32 99 Web: www.hv.se

Förord

Jag som har genomfört arbetet och är författare till denna studie vill tackar Henrik Nyman Hjern som har ställt upp och tagit sig tid att skriva C koden för programmet datasim. Jag vill även tacka Niclas Eriksson som hjälpte till att felsöka eventuella buggar som uppstod i programkoden.

Innehållsförteckning

1 Inledning 1
1.1 Syfte 1
2 Avgränsningar 1
3 Teknisk Bakgrund 2
3.1 Pakettyper 3
3.2 Överföringstyper 4
3.2.1 Bulköverföringar 4
3.2.2 Styröverföringar 5
3.2.3 Avbrottsöverföring 5
3.2.4 Isokron överföring 5
3.3 Relaterat arbete 5
4 Metod 7
4.1 Testutrustning 7
4.2 Tillvägagångssätt 8
4.3 Testfall 9
4.4 Mätteknik 10
4.5 Utförande av tester 11
4.6 Felmarginaler 11
5 Resultat och Analys 12
5.1 Skrivning 12
5.1.1 Bandbredd 12
5.1.2 Spridning 14
5.2 Läsning 19
5.2.1 Bandbredd 19
5.2.2 Spridning 21
6 Slutsats 26

Figurer

Figur 1. USB 3 arkitekturbeskrivning där USB 2- och USB 3-buss parallellt samarbetar. 3

Figur 2. Flödesschema för läsning där A) visar ingen strömning av media och B) visar strömning av media ifrån USB enheten till datorn. 9

Figur 3. Flödesschema för skrivning där A) visar ingen strömning av media och B) visar strömning av media ifrån USB enheten till datorn. 10

Figur 4. Hastigheter i bandbredd USB 2 & 3 använder vid mätningarna för skrivning av 1 Byte stora filer. För att förtydliga skalan på Y-axeln så rör den sig med ett minimum värde på 0,05 KB/s och ett maximum på 1,45 KB/s. 13

Figur 5. Hastigheter i bandbredd USB 2 & 3 använder vid mätningarna för skrivning av 1 Kbyte stora filer. 13

Figur 6. Hastigheter i bandbredd USB 2 & 3 använder vid mätningarna för skrivning av 1 Mbyte stora filer. 14

Figur 7. A) USB 2 1 Byte skrivning utan strömning. B) USB 3 1 Byte skrivning utan strömning. 15

Figur 8. A) USB 2 1 Byte skrivning vid strömning, B) USB 3 1 Byte skrivning vid strömning. 15

Figur 9. A) USB 2 1 Kbyte Skrivning utan strömning, B) USB 3 1 Kbyte Skrivning utan strömning. 16

Figur 10. USB 2 1 Kbyte skrivning vid strömning, B) USB 3 1 Kbyte skrivning vid strömning. 17

Figur 11. A) USB 2 1 Mbyte Skrivning utan strömning, B) USB 3 1 Mbyte Skrivning utan strömning. 18

Figur 12. A) USB 2 1 Mbyte skrivning vid strömning, B) USB 3 1Mbyte skrivning vid strömning. 19

Figur 13. Hastigheter i bandbredd USB 2 & 3 använder vid mätningarna för läsning av 1 Byte stora filer. För att förtydliga skalan på Y-axeln så rör den sig med ett minimum värde på 0,05 KB/s och ett maximum på 1,45 KB/s. 20

Figur 14. Hastigheter i bandbredd USB 2 & 3 använder vid mätningarna för läsning av 1 Kbyte stora filer. 20

Figur 15. Hastigheter i bandbredd USB 2 & 3 använder vid mätningarna för läsning av 1 Mbyte stora filer. 21

Figur 16. A) USB 2 1 Byte läsning utan strömning, B) USB 3 1 Byte läsning utan strömning. 22

Figur 17. A) USB 2 1 Byte läsning vid strömning, B) USB 3 1 Byte läsning vid strömning. 22

Figur 18. A) USB 2 1Kbyte läsning utan strömning, B) USB 3 1Kbyte läsning utan strömning. 23

Figur 19. A) USB 2 1 Kbyte läsning vid strömning, B) USB 3 1 Kbyte läsning vid strömning. 23

Figur 20. A) USB 2 1 Mbyte läsning utan strömning, B) USB 3 1 Mbyte läsning utan strömning. 24

Figur 21. A) USB 2 1 Mbyte läsning vid strömning, B) USB 3 1 Mbyte läsning vid strömning. 25

Tabeller

Tabell 1: Hårdvara som används i testdatorn 7

Tabell 2. Programvara för mätningarna 7

Bilagor

A. Programkod för DataSim

B. Bashskript

C. Mätdata (tid)

1 Inledning

Redan under 90-talet har Universal Serial Bus även kallat USB varit en standard för att kommunicera med externa enheter. Totalt har fyra versioner släpps av USB. Version 1.0 även kallat *Low Speed* är första officiella versionen och har en hastighet på 1,5 Megabit/sek. Version 1.1 som är nästa steg som använder en bandbredd på 12 Megabit/sek och kallas för *Full Speed*. Dessa två versioner användes mest i samband med mänskliga gränssnittsenheter (Human Interface Device), enheter som möss och tangentbord är exempel på dessa enheter och använts dagligen.[1, 11, 12]

Nästa steg i utvecklingen är version 2.0 såkallad *Hi-Speed* som kom år 2000 och var bakåtkompatibel med föregångarna. Den maximala hastigheten är 480 Megabit/sek vilket öppnade nya möjligheter. Med nya möjligheter menas exempelvis användning utav masslagringsenheter(MSC) vilket startade en ny era där det i majoritet handlar om flashlagringsmedia.[7, 11, 12]

Mellan 2008 – 2009 lanserades den nyaste versionen. *Superspeed* är version 3.0 utav USB-standarden. I kombination av en ny förbättrad USB-arkitektur tillsammans med en ny lösning för USB-kabel som hjälper att maximera det nya systemets potential kan hastigheter som utnyttjar fullduplex uppnå 4,8 Gigabit/sek. Tack vare dessa hastigheter kan otroligt stora mängder data flyttas på en relativ kort tid. Men med så stora förändringar i hastigheten vars grund fortfarande bygger på gamla 2.0 versionen kan vissa frågor uppstå. Frågor som, klarar den gamla grunden av de nya hastigheterna utan att förlora prestanda eller ge en negativ inverkan?[4, 7, 8, 12]

1.1 Syfte

Målet med studien är att undersöka hur överföringstypen bulk av USB 3 jämfört med USB 2 presterar under användning. I syfte med detta skapas testfall som har till uppgift att styra så rätt typ av data erhålls som ska användas för att studera och jämföra prestanda. Med rätt data menas data som berör USB och inte en annan del i systemet. Totalt kommer tolv olika testfall användas. I dessa kommer storleken på data som överförs, riktning samt belastning på USB-bussen vara faktorer som ändras. Mätdatan som erhålls bearbetas sedan till olika mått och figurer som exempelvis standardavvikelse vilket är ett mått på spridning. Mätvärden och andra mått används i slutet av studien för att identifiera och klarlägga prestige, eventuella skillnader eller försämringar mellan USB-versionerna.

2 Avgränsningar

Mätningarna och testdata som använts ifrån testfallen kommer enbart att bygga på bulköverföringar. De andra överföringstyperna avbrott (Interrupt), isokron

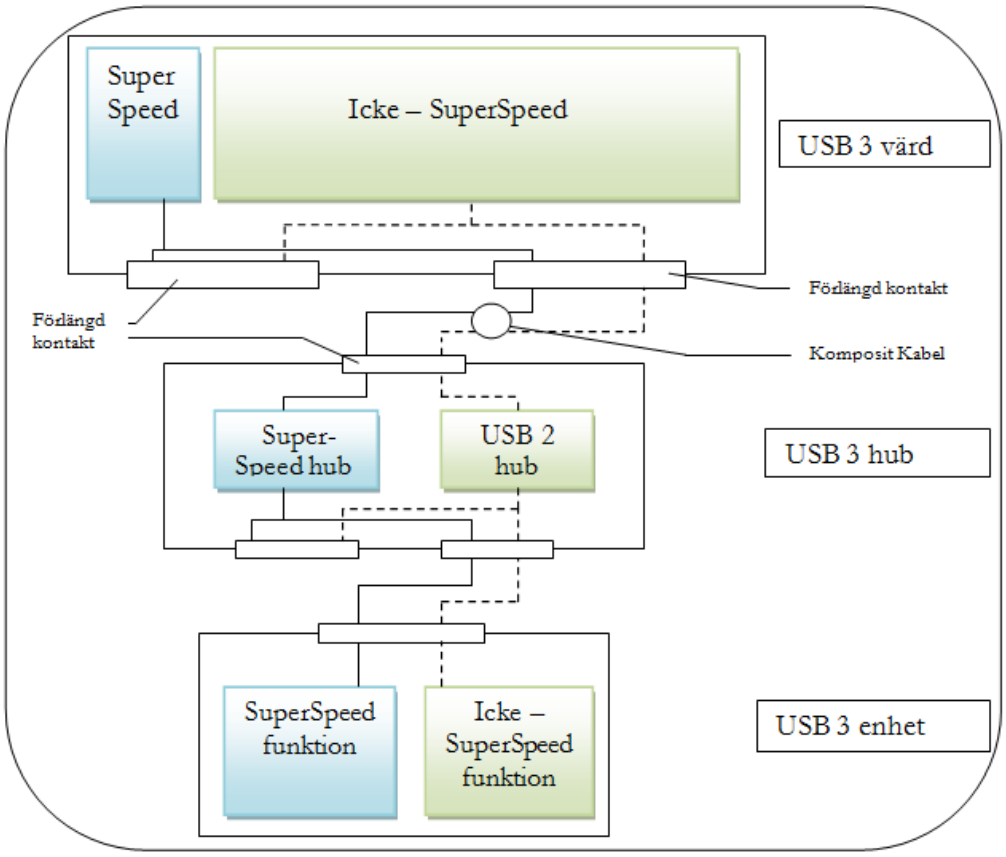
(Isochronous) samt styröverföring (Control) kommer inte göras specifika mätningar på i testerna. Men på grund av bulköverföringens funktionalitet och dess beroende utav de andra överföringstyperna är det rekommenderat att läsa mer om dessa tre typerna i teknisk bakgrund för att få en bättre förståelse.

Sedan kommer det inte finnas några jämföranden mellan tillverkarens angivna maximala hastighet och de uppmätta värdena i denna studie. Detta på grund av utrustningen som använts i denna studie inte stöder dessa hastigheter och på så vis kan inga trovärdiga jämförelser göras.

3 Teknisk Bakgrund

Grunden till USB 3 är den samma som för 2.0. Det vill säga att det är en bussteknik som stöder datautbyte mellan en dator och en mängd olika externa enheter. Dessa enheter delar på bandbredden med hjälp av ett värdbaserat protokoll. USB 3 använder sig av en dualbuss-arkitektur som gör det möjligt för bakåtkompabilitet till USB 2 och lägre. Strukturen bygger på olika nivåer av stjärntopologier där nivå 1 består av en värdenhet, och hubbar vid de nivåerna under värdenheten. Varje nivå är sedan ihopkopplad för att möjliggöra kommunikationen mellan varandra vilket slutar i något som liknar en pyramid eller trästruktur. Dualbuss-arkitekturen ger även möjlighet för samtidig användning av Superspeed och Full-speed.[2, 4 - 6, 8]

Baslinjen i strukturen för topologin är den samma som för USB 2. I dualbuss-arkitekturen är det hopsättningen av kontakten och koaxialkabeln som skapar elektrisk och mekaniskt bakåt- och framåtkompatibilitet för USB 3. Om det inte skulle ha varit möjligt att inkludera både bussinterfacen parallellt av USB 3 och 2 skulle bakåtkompabiliteten som kan köras samtidigt inte vara möjligt. Arkitekturen i figur 1 har även som mål att maximera identifieringshastigheten utav enheter. När det gäller hubbar är det en speciell klass av USB-enheter som har till uppgift att skapa ännu fler anslutningsnoder till bussen bortsett ifrån de som finns tillgängliga från värdenheten. [4 - 6, 8]



Figur 1. USB 3 arkitekturbeskrivning där USB 2- och USB 3-buss parallellt samarbetar.

3.1 Pakettyper

USB 3 använder sig av fyra olika pakettyper vilket är samma antal som de äldre versionerna utav USB. LMP (Link Management Packet), TP (Transaction Packet), DP (Data Packet) och ITP (Isochronous Timestamp Packet).[1, 2, 8]

LMP har som ändamål att uppehålla en anslutning. Paketen använder sig av inga adresser och på så sätt kan dessa inte routas. Sådana paket kan exempelvis uppstå ifrån ett hubbportkommando.[8]

TP färdas mellan värdenheten och enheten där dess uppgift är att sköta ändpunktsanslutningar samt kontrollera dataflödet mellan ändpunkterna. När ett TP skickas ifrån enheten fylls fältet ”*Device Address*” med sin egen adress och sänds vidare till värdenheten. Värdenheten kommer sedan att återanvända den mottagna adressen för att sedan identifiera källan. ”*Subtype*” -fältet i ett TP används av mottagaren för att bestämma formatet samt användningen av paketet. Exempel kan vara ACK (Acknowledgement), ERDY(Endpoint Ready), STALL. [8]

DP består av två delar DPH (Data Packet Header) och DPP (Data Packet Payload). DP kan exempelvis skickas från värdenheten och används för att transportera data mellan ändpunkterna.[8]

ITP används för att hålla enheterna synkroniserade med hjälp av tidsinformation som har tagits ifrån värdenheten. Paketen innehåller endast tidsstämplar och ingen adress eller routinginformation. På grund av detta bör och kan mottagaren inte skicka svar med ITP till sändaren. ITP skickas med hjälp av multicast ifrån hubbar där endast deras nedströmsportar använts med länken i ett visst läge "U0", det vill säga aktivt läge.[1, 8]

3.2 Överföringstyper

USB-arkitekturen använder sig av något som kallas ”*Pipe*” vilket i den här studien refereras som "rör" och förbinder USB-enheten med värdenhetens mjukvara. Det är rören som gör det möjligt att flytta data mellan enheter via en minnesbuffert och en ansluten enhet. Data kan flöda till eller ifrån värden, ifall det används två riktningar kommer ett till rör skapas för att ta hand om det motsatta flödet då strömmen bara kan flöda i en riktnig. En ny egenskap som har implementerats i superspeed är att när en ickeisokron nod är upptagen, skickas ett NRDY (Not Ready) svar. Detta kommer innebära att slutnoden måste skicka ett ERDY(Endpoint Ready) meddelande ifall denna ska kunna ta emot data igen. Värdenheten kommer sedan schemalägga nya överföringar vid nästa tillfälle.[8]

3.2.1 Bulköverföringar

Bulköverföringar är avsedda för flyttandet av stora mängder data. Datan är garanterad att komma fram men det finns ingen garanti på bandbredd eller förseningar. Ett användningsexempel kan vara överföring av en datafil. Bandbredden som står till förfogan är den resterande bandbredden som är ledig. Det vill säga bandbredden som inte har allokerats av andra överföringstyper och på så sätt kan den tillgängliga bandbredden varierar kraftigt. I sammanhang med Superspeed kallas det även för ”good effort”.[1, 3, 6, 8]

Den maximala payload-storleken för bulköverföringar är 1024 byte. Ifall datan är större än maxstorleken på paketet (1024 byte) kommer ett flertal överföringar genomföras för att flytta data. Det är värdenheten som garanterar att ingen payload överstiger 1024 byte och samtidigt inte skickar fler paket än maximalt tillåtet. [1, 8]

Som det har nämnts tidigare kan ingen bandbredd allokeras till bulköverföringar. På så vis kan bara den tillfälliga oanvända bandbredden utnyttjas för bulköverföringar. Ifall det finns flera pågående överföringar och bandbredden inte längre räcker till, skapar värden tillfällen för bandbredd till varje slutnod som är grundat på en så kallat *fair access policy*. Ett exempel kan vara att värdenheten minskar datan i strömmen per överföring och rättvist fördelar den resterande tillgängliga bandbredden bland alla pågående bulkströmmar.[8]

3.2.2 Styröverföringar

Varje enhet som ansluts måste först skapa ett standardrör. Standardröret är avsett för styrtrafik som är till för initialisering samt kontroll. Röret används bland annat för att skicka såkallade ”*device descriptors* ”och ändra enhetens beteende. Styröverföringar använder ”best effort” mellan värdenheten och externa enheten där värdenheten styr flödet. Vid high-speed och superspeed kan upp till 20 % av maximala bandbredden reserveras för styröverföringar.[1, 3, 6, 8]

3.2.3 Avbrottsöverföring

För överföringar som sker periodvis, där datamängden är små men kräver garanterat överföring används avbrottsöverföringar. Exempel på ett sådant fall kan vara data ifrån tangentbordet. Denna överföringstyp garanterar maximala serviceintervaller samt försöker igen ifall ett fel skulle uppstå under överföringen. Vid varje serviceintervall som är $2^{(\text{bIntervall-1})}$x $125\mu s$ används tillräckligt med bandbredd för att garantera överföringen. Efter en lyckad överföring kommer ny data skickas tidigast vid nästa serviceintervall. Om mottagaren svarar med ett NRDY meddelande eller en bekräftelse att det inte går att ta emot några paket för tillfället, kommer värdenheten inte försöka igen till dess mottagaren skickat ett meddelande som bekräftar att denna är beredd att ta emot ny data. Efter ett sådant meddelande har tagits emot kommer värdenheten fördubbla servicen inom samma serviceintervall.[1, 3, 6, 8]

3.2.4 Isokron överföring

Denna typ av överföring är avsedd som stöd för strömmar med låg eller ingen feltolerans som använder sig av periodisk trafik inom ett begränsat serviceintervall. Ett exempel vid användning kan vara bild- och ljud data. Tidsinformation skickas separat med ITP som har nämnts tidigare. Isokrona överföringar kan allokera upp till 80 % av den totala bandbredden. Allokeringen forsätter så länge data förses till rören med begränsat latens. Det kan förekomma lägen där isokrona överföringar måste använda länkar som blev inaktiva på grund av energisparlägesfunktionen. Resultatet kan bli förseningar och därmed hinner inte datan fram i tid. För att undvika detta används en specifik mekanism. Mekanismen går ut på att värdenheten skickar ett pingpaket till enheten innan en överföring startas. Enheten svarar med ett pingsvarspaket som talar om att alla länkar till enheten är aktiva.[1, 3, 6, 8]

3.3 Relaterat arbete

Ett av de mest närliggande arbeten till denna studie är en artikel av Patrik Wahlqvist [17]. Artikeln handlar om att visa prestandaskillnad mellan USB 2 och USB 3 med hjälp av att prestandamäta olika typer av USB-minnen. I artikeln visas det tydliga prestandaskillnader mellan de olika versionerna. Artikeln påpekar även att beroende på vilka drivrutiner man använder för USB 3 så kan stora skillnader av prestanda

beskådas. En till sak som nämns är att USB 3 verkar vara mer känslig för prestanda i fråga om processor, minne och hårddisk än USB 2. Dessa två punkter om USB 3 kommer även att tas hänsyn till i denna studie.

En liten nackdel med artikeln är att det inte skrivs om exakt vad för processor, minne och hårddiskar som användes under testningen då dessa delar verkar påverka USB 3 prestandan. Man skriver även inget om vilken typ av prestandamätning samt vad för slags data som användes under testningen.

4 Metod

Innehållet i detta kapitel kommer att ta upp vilka steg som togs för att genomföra studien. Det kommer att ges information om vilken hårdvara samt programvara som har använts, hur varje testfall är uppbyggt i form av en bild och förklarande text. Samt gå igenom hur testerna genomfördes och ta upp eventuella faktorer som kan påverkat mätningarna.

4.1 Testutrustning

Tabellerna nedan visar detaljerad hårdvaru- samt mjukvaruinformation som har används i testfallen. Valet av dessa komponenter beror dels på att ha så hög prestanda som möjligt när mätningarna genomfördes för att minska risken för eventuella prestandaförluster. Samt för vissa komponenter fanns det inget större val vid första kvartalet år 2010 med tanke på vad som fanns tillgängligt på marknaden, exempel är den externa hårddisken som stöder USB 3-standarden. Programmet datasim som användes i dessa testfall är skriven i programmeringsspråket C. För specifik information om koden för testprogrammet se Bilaga A.

Tabell 1: Hårdvara som används i testdatorn

OS	**Typ**	**Version**
Windows 7 Ultimate	x64	6.1 Build 7600
CPU	**Hastighet**	**Cache**
AMD Phenom II X4 945	3000MHz	64+64+4*512+6144 KB
Minne	**Typ**	**Storlek**
Kingston DDR3 HyperX 677 MHz	PC3-10600 DDR3 SDRAM UDIMM	3x2048MB
Hårddisk	**Gränsnitt**	**Storlek/buffer**
WDC WD3200AAKS-00B3A0	SATA II (3Gb/s)	320GB, 16MB
WDBABP0010HCH-EESN (USB 3.0)	USB 3.0/SATA II (3Gb/s)	1TB, 16MB
Moderkort	**Chip**	
ASUS M4A79XTD EVO	AMD 790X (RD780) + SB750	

Tabell 2. Programvara för mätningarna

Program	**Version**
DataSim	v.0.1
VLC	1.1.1
Resouce Monitor	v.6.1.7600

4.2 Tillvägagångssätt

För att underlätta genomförandet av studien, delades den in i olika faser. Varje fas bestod av ett antal delmål. Efter varje delmål hade uppfyllts kontrollerades resultatets innehåll samt att tillvägagångssättet följde beskrivningen som hade angivits. Vid godkänd kontroll ansågs fasen som avslutat. Efter fasen var klar påbörjades en ny fas.

Fas 1. Hårdvara

Delmål 1: Införskaffa stationär dator som testerna utförs på. Införskaffa extern masslagringsenhet som stöder USB 3 samt instickskort med USB 3-portar.

Delmål 2: Installation av hårdvara, detta inkludera mjukvara till kortet.

Fas 2. Skapning av testfall

Delmål 1: Fastställa hur testfallen skall konstrueras för att passa ändamålet.

Delmål 2: Bestämma vilka funktioner programvaran som genomför mätningar måste uppfylla.

Delmål 3: Dokumentation på vad och hur som skall mätas samt hur man går tillväga i testerna.

Fas 3. Programvara

Delmål 1: Införskaffa programvara. Det vill säga OS, programvara samt kodning för testverktyget.

Delmål 2: Installation av all nödvändig programvara för testerna.

Fas 4. Genomförning av testfall

Delmål 1: Skriva ett bashscript som anger parametrar för testprogrammet, kör programmet och iterera körningen ett visst antal gånger.

Delmål 2: Testerna utförs för skrivning och dokumenteras. Detta inkludera testning vid strömning.

Delmål 3: Testerna utförs för läsning och dokumenteras. Detta inkludera testning vid strömning.

Fas 5. Sammanställning

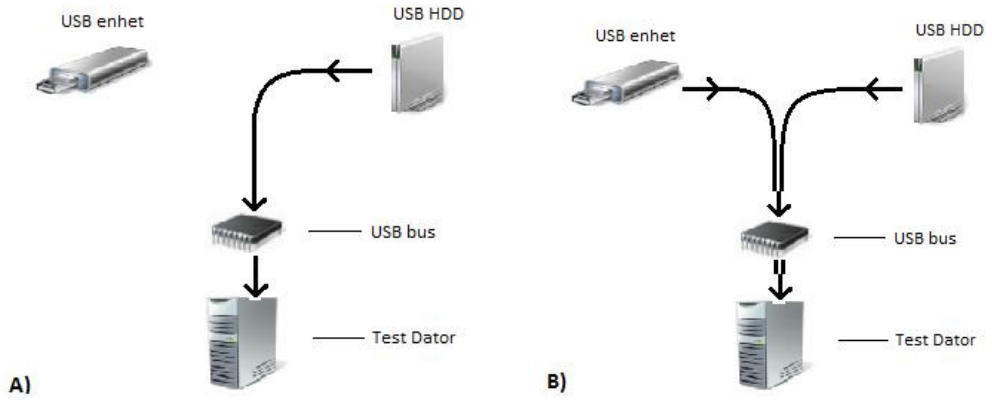
Delmål 1: Mätvärden sammanställs i diagram för skrivning, läsning och med eller utan strömning under test. Medelvärdet och standardavvikelsen togs fram.

Fas 6. Slutfas

Delmål 1: Diagram analyseras, slutsats skrevs.

4.3 Testfall

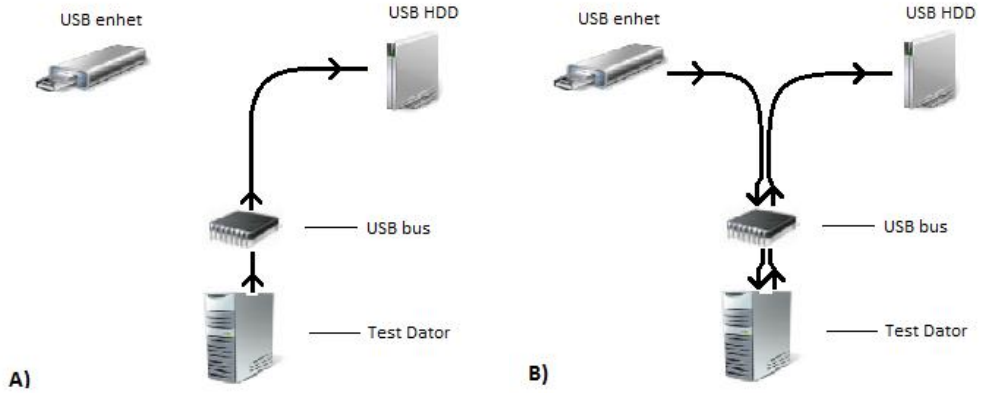
Alla testfall använde samma testdator som testerna kördes på. De huvudsakliga beståndsdelarna är en extern hårddisk som stöder USB 2 och 3 standarden, ett USB 2-minne och ett instickskort som kopplas till testdatorn med hjälp av PCI Express 2 gränssnitt. Vilket ger stöd till testdatorn för användning av USB 3. Externa hårddisken kopplades sedan ihop med testdatorn. Se figur 2 - 3.



Figur 2. Flödesschema för läsning där A) visar ingen strömning av media och B) visar strömning av media ifrån USB enheten till datorn.

I Figur 2 A) avbildas ett testfall där data hämtades ifrån USB-hårddisken vilket i den här studien står för *läsning*. Testfall A gick ut på att datafiler skapades på USB-hårddisken. Sedan kopierades dessa filer till en intern hårddisk. Under tiden data kopierades mättes tiden det tar att flytta filerna. Efter kopieringen hade slutförts raderades all data vid destinationen och källan. Allt detta sköter programmet datasim. I figur 2 B) skedde nästan samma sak, men till skillnad ifrån A) ser ett annat program[1] till att media strömmades ifrån USB-enheten till testdatorn. Median bestod av en film i mkv-format (Matroska Multimedia Container) varav varje ström använde mellan 900 – 1300 KB/s[10]. Totalt spelades 7 filmer samtidigt som använde cirka 6,3 - 9,1 MB/s i bandbredd. Syftet med strömningen var att allokera bandbredd i USB-bussen. På så vis fick bulköverföringarna mindre bandbredd att använda sig av. Samtidigt överfördes samma mängd testdata vid detta tillfälle som vid testfall A).

[1] VLC (VideoLan)



Figur 3. Flödesschema för skrivning där A) visar ingen strömning av media och B) visar strömning av media ifrån USB enheten till datorn.

Figur 3 har samma uppsättning som figur 2 ovan. Den enda skillnaden är att testdata nu skapades på testdatorn och kopierades till USB-hårddisken. Det vill säga data skrevs till USB-hårddisken vilket i den här studien står för *skrivning*.

Denna uppsättning gav samma förutsättningar för både USB-versionerna. Tidmätningarna genomfördes för skrivning och läsning då USB-bussen var obelastad och belastad. Syftet med att det användes en separat extern enhet som strömningsmedia och inte externa testhårddisken var att inte belasta hårddisken i onödan och på så sätt påverka prestandan vid mätningarna.

4.4 Mätteknik

Datasim är ett program skrivit i C och skapades för dessa tester. Programmets huvuduppgifter var att generera data i form av en fil. Datamängden varje fil innehöll kunde ändras enligt önskemål. Sedan skickades filen till en given destination. Till slut mättes hastigheten samt överföringstid i millisekunder. Efter att ha samlat mätvärden skulle en del uträkningar genomföras för att jämföra USB-versionerna.

Medelvärdet som användes i den här jämförelsen är det aritmetiska medelvärdet som räknas fram med formeln $\overline{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$ och är en del av andra uträkningar i resultatet.

Standardavvikelse är ett statistisk mått på hur värden i en mätning avvek ifrån medelvärdet, formeln som användes är $\sigma = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2}$.
Variationskoefficient är förhållandet mellan standardavvikelsen och medelvärdet. Formeln ser ut som $c_v = \frac{\sigma}{\bar{x}}$, svaret anges i procent. Variationskoefficienten gör det möjligt att jämföra liknade system med varandra även om dessa använder olika skalor.[9]

VLC(VideoLan Client) är en mediaspelare som stöder ett antal olika format för både ljud och bild. Bortsett ifrån ett stort antal andra funktioner kan VLC strömma media. Strömningen kan ske både lokalt eller mot ett IP.[15, 16]

4.5 Utförande av tester

Ett Bash-script[2] användes för att mata in parametrar till testprogrammet datasim samt upprepa dessa ett antal gånger. I scriptet anges parametrar för antal filer, storlek på filen samt hur många gånger scriptet skulle itereras. Källa och destination kan ändras genom att ange en adress in ”Källa” för källan och ”Destination” för destinationen. Adressens format såg exempelvis ut som ”`C:\CopyTest\temp`”, varje adress måste sluta med ”`\temp`” för att programmet skulle fungera korrekt. För att ändra antalet filer som skickades, ändrades parametern ”Antal” med ett positivt heltal. Ifall filstorleken skulle modifieras, angavs antal byte i parametern ”Storlek” med ett positivt heltal. Beroende på hur många gånger datasim behöver köras byts ”iterationer” ut mot ett positivt heltal.

4.6 Felmarginaler

Eventuella interferenser som kunde påverka mätvärden borde tas hänsyn till. Hårdvara så som mus och tangentbord tar upp bandbredd även om denna är relativt liten jämfört med den maximala kapaciteten. Man bör dock känna till att dessa används.

Programmet datasim som användes för att genomföra testerna gav inga exakta värden, det skiljer cirka ± 10 ms. Testerna utfördes inte i en ideal testomgivning. Utförandet skedde i en vardaglig miljö, på grund av detta fanns det möjlighet att annan programvara i bakgrunden kunde påverka mätningarna och därmed mätvärdena. Sedan fanns det även en faktor angående flaskhalsen i överföringarna. Vid fullspeed var flaskhalsen själva USB-versionen jämfört med superspeed där flaskhalsen hade förflyttats till ändpunkten, med andra ord den fysiska hårddisken.[4]

Sedan visar det sig även att USB 3-minnen var känsliga för prestanda när det gällde processor, minne samt lagringsmedia. Dock behöver det nödvändigtvis inte vara samma sak för externa hårddiskar. Till sist bör de medföljande drivrutiner nämnas. Beroende på vilka drivrutiner som användes för USB 3 kunde relativt stora prestandaskillnader uppstå.[1]

[2] Se bilaga B

5 Resultat och Analys

Mätvärden har tagits fram från testerna och sammanställts i form av histogram och diagram. Se bilaga C för mer information för spridningsdiagram. Histogrammen representerar tre filstorlekar 1 Byte, 1 Kbyte och 1 Mbyte och har skickats tusen gånger per mätning. Mätningarna har genomförts i två olika testfall, ett där USB-bussen inte belastas när mätningar görs samt ett annat där det belastats, det vill säga bandbredd allokeras med hjälp av en annan överföringstyp då media strömmas. Diagrammen delas upp i två underkapitel, skrivning respektive läsning till USB-hårddisken. Uträkningar som använts för att analysera samt jämföra resultaten för spridning är medelvärde, standardavvikelsen och variationskoefficient. Värdet c_v kommer till största del användas när mätvärden ifrån USB-versionerna jämförs. Den röda kurvan i histogrammen symboliserar normalfördelningen.[9, 13, 14]

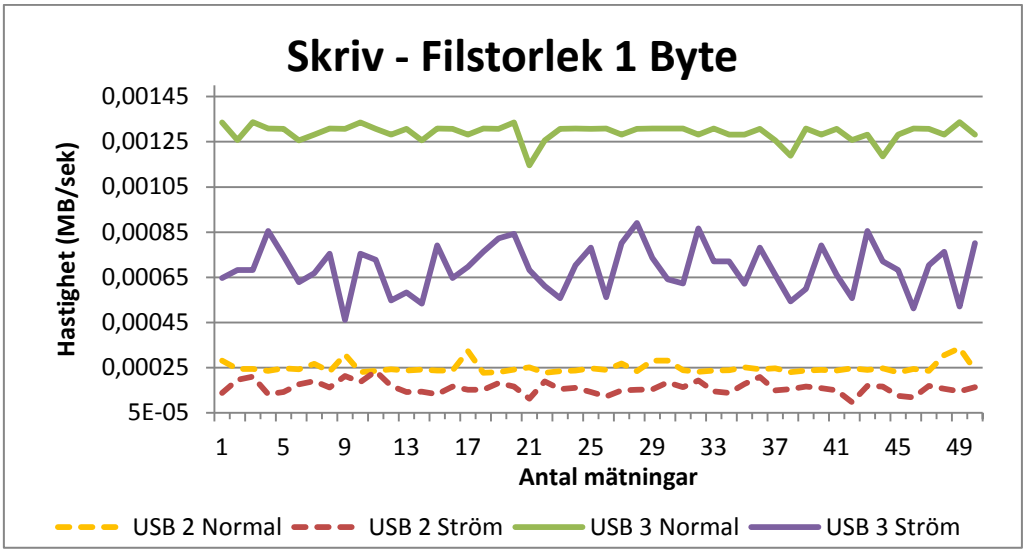
5.1 Skrivning

I detta delkapitel presenteras resultatet med analys för den delen vi kallar för *skrivning* i denna studie. Skrivning delas in i två underkapitel för att få en lättare överblick av figurer och diagram.

5.1.1 Bandbredd

I figur 4 nedan ser vi bandbreddsanvändningen utav skrivmätningar som genomförts. Den röda och gula linjen avbildar bandbredden av USB 2 och grön och lila för USB 3 där den övre linjen avbildar hastigheten vid normalbelastning utav USB-bussen och den nedre det vill säga lila och röd vid belastning med hjälp av strömmande media. Tittar man på USB 2´s bandbreddsanvändning visar sig båda testfall stabila där röd ligger på 0,16 Kb/s i medel och gul på 0,25 Kb/s. USB 3 ligger vid 0,69 Kb/s respektive 1,3 Kb/s i medelvärde. Med detta visar USB 3 en högre maxhastighet på lite mer än fem gånger än USB 2. Men till skillnad från USB 2 är bandbredden vid belastning mer oregelbunden.

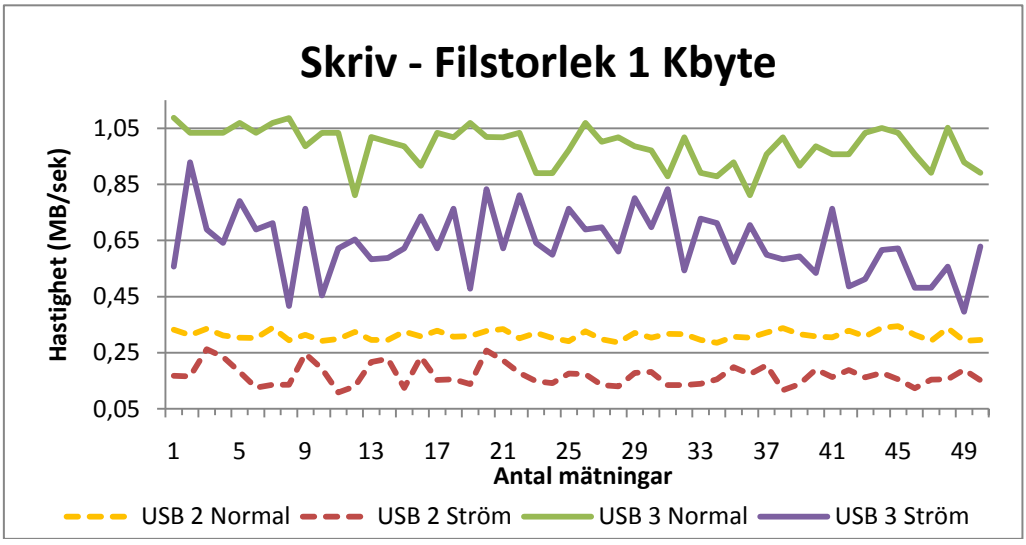
Som man tydligt ser illustreras ett stort gap i hastighet som förväntat. Jämförs varje testfall inom samma version baserat på hastighet så förlorar USB 3 0,61 Kb/s vilket är mycket mer jämfört med USB 2 där det minskar med 0,09 Kb/s. Men det behöver dock inte ses som negativt, tas skillnaden i procent ligger USB 3 på 47 % och 36 % för USB 2. På så vis är skillnaden inte alls lika stor då det skiljer enbart 10 %.



Figur 4. Hastigheter i bandbredd USB 2 & 3 använder vid mätningarna för skrivning av 1 Byte stora filer. För att förtydliga skalan på Y-axeln så rör den sig med ett minimum värde på 0,05 KB/s och ett maximum på 1,45 KB/s.

Figur 5 visar att hastigheter för 1 Kbyte stora filer. Som i figur 4 så har USB 2 betydligt lägre hastigheter än USB 3. Vid denna filstorlek ligger USB 2 på 0,32 MB/s som visas av den gula linjen och 0,17 i medelvärde för den röda. Ser man på USB 3 ligger mätningar på 0,98 MB/s samt 0,64 MB/s. Mätningarna visar att hastighetsskillnaden bara är 3,5 gånger mellan versionerna vilket är en minskning ifrån 1 bytes filer.

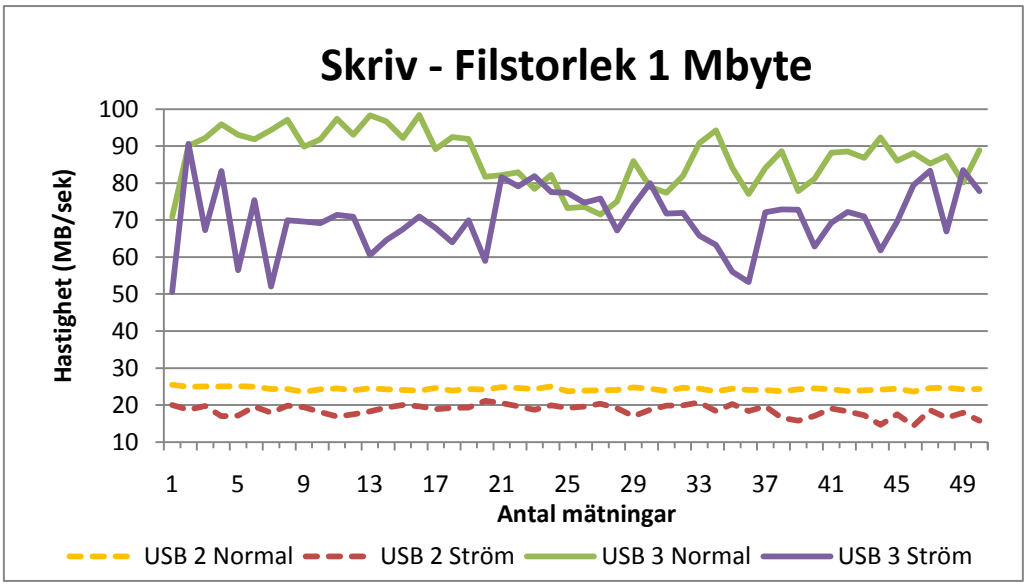
Som vi såg i figur 4 förlorar USB 3 mycket mer hastighet än för USB 2 när bussen belastas med en dataström. I det här fallet minskar versionerna procentuell på 45 % för USB 2 och 35 % för USB 3 i hastighet.



Figur 5. Hastigheter i bandbredd USB 2 & 3 använder vid mätningarna för skrivning av 1 Kbyte stora filer.

För att bandbreddsanvändning för en megabyte filer illustreras värden i figur 6. Som man tydligt kan se så är USB 3's överföringshastigheter väldigt oregelbundna jämfört med föregående mätningar. I vissa fall skär till och med linjerna varandra och visar en högre hastighet när USB-bussen belastas än vid normalbelastning. Hastighetsmässigt ligger USB 3 på 70 MB/s för grönt och grönt på 87 MB/s i medel.

USB 2 visar inga märkbara förändringar, med ett medelvärde på 18,5 MB/s för röd och 24,3 MB/s för gult. Minskningen i hastigheten när USB-bussen är obelastat och belastats för båda versionerna ligger på runt 21 %. Även här är hastigheten ca 3,5 gånger större för USB 3 jämfört med USB 2.

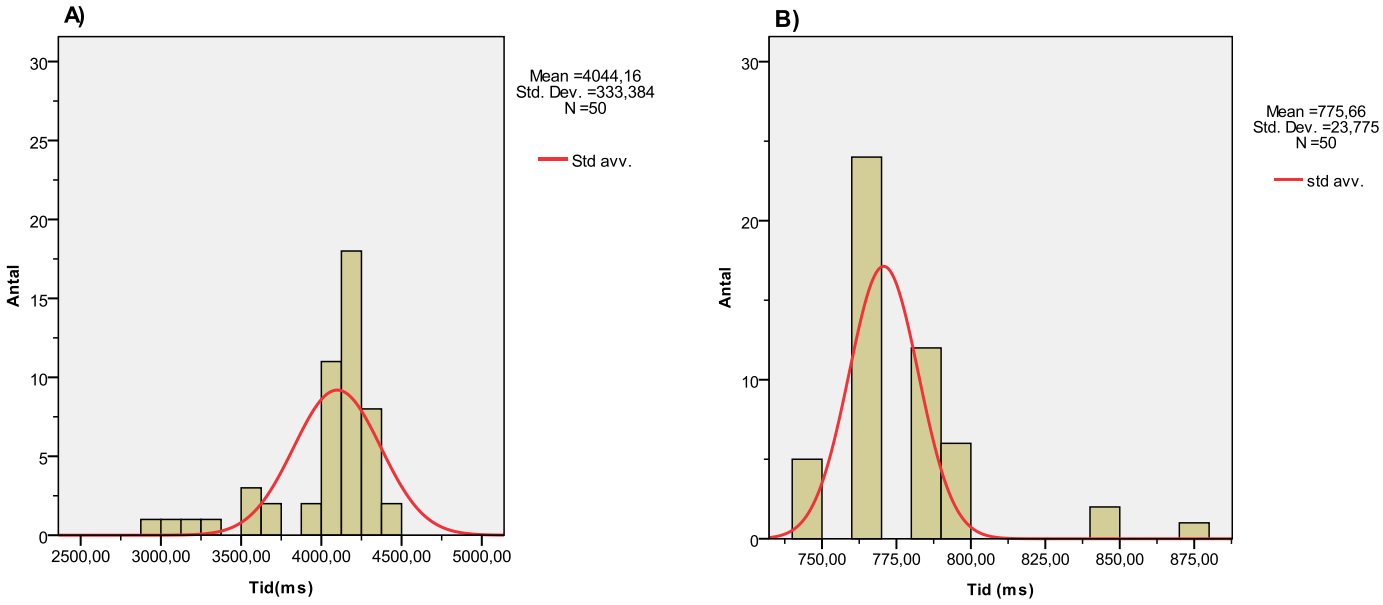


Figur 6. Hastigheter i bandbredd USB 2 & 3 använder vid mätningarna för skrivning av 1 Mbyte stora filer.

5.1.2 Spridning

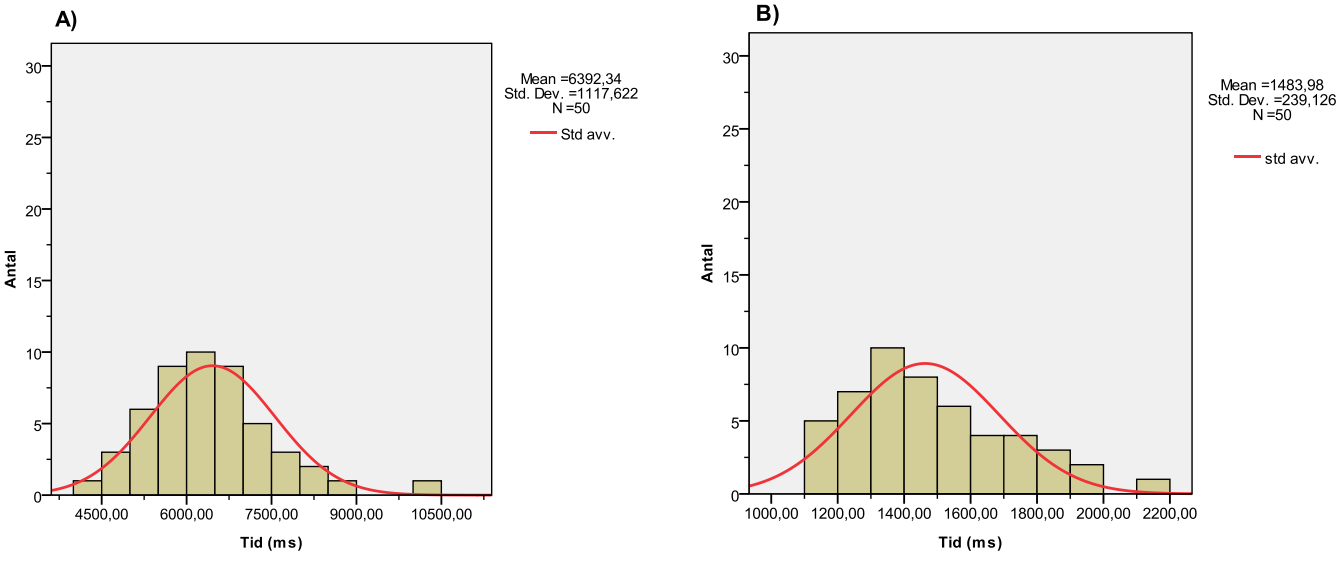
De histogram som presenterar mätningarna under tiden ingen strömning genomfördes är konstruerat enligt testfall figur 3 A) och respektive med strömning enligt testfall figur 3 B).

Figur 7 A) visar att standardavvikelsen ligger på σ = 333,384 ms vilket ger c_v = 8,24 %. Jämförs värdet med figur 7 B) där dess standardavvikelse ligger på σ = 23,775 ms vilket resultera i c_v = 3,06 %, kan man se att USB 2 har sina värden fördelat på en bredare tidsspann. Dock ser man i figur 7 B) ett fåtal värden runt 830 – 885 ms som tog betydligt längre tid än majoriteten.



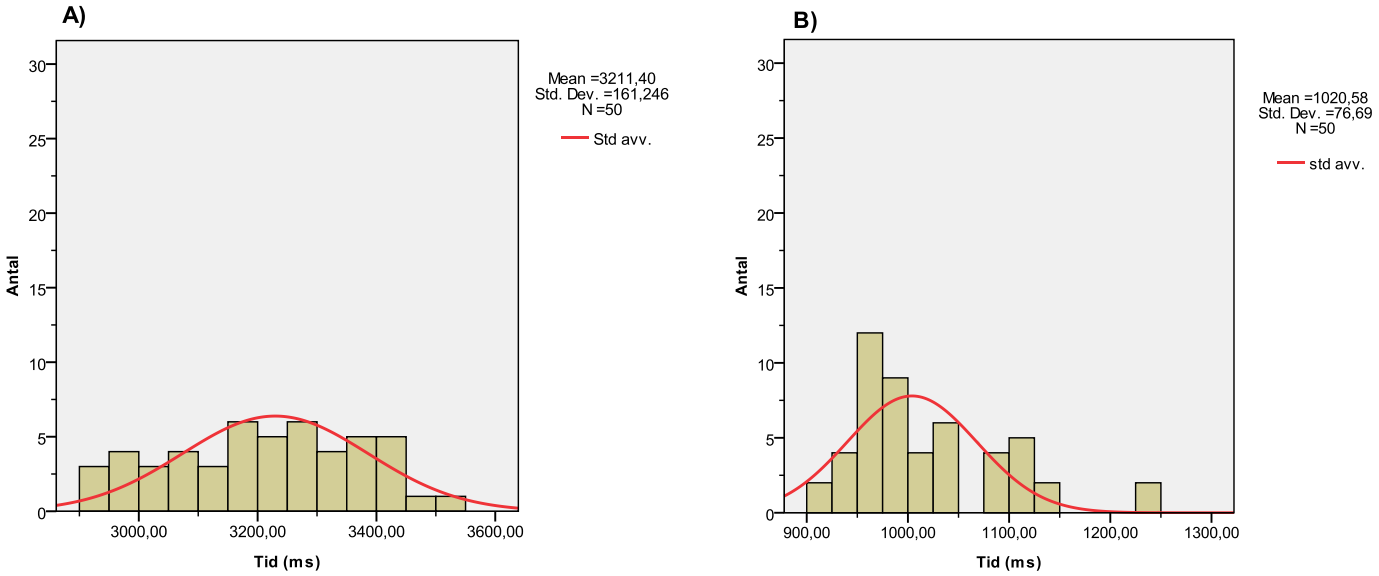
Figur 7. A) USB 2 1 Byte skrivning utan strömning. B) USB 3 1 Byte skrivning utan strömning.

Figur 8 visar inga större avvikelser mellan USB 2 som har c_v = 17,48 % och USB 3 c_v = 16,11 %. Jämförs vardera USB-version med sina egna 1 bytesskrivningarna är ökningen av variationskoefficienten för USB 3, cirka 4-5 gånger högre än för USB 2 där ökningen har fördubblats. Vänder man på det hela presterar USB 3 bättre när bussen inte har någon större form av bandbreddsallokering.



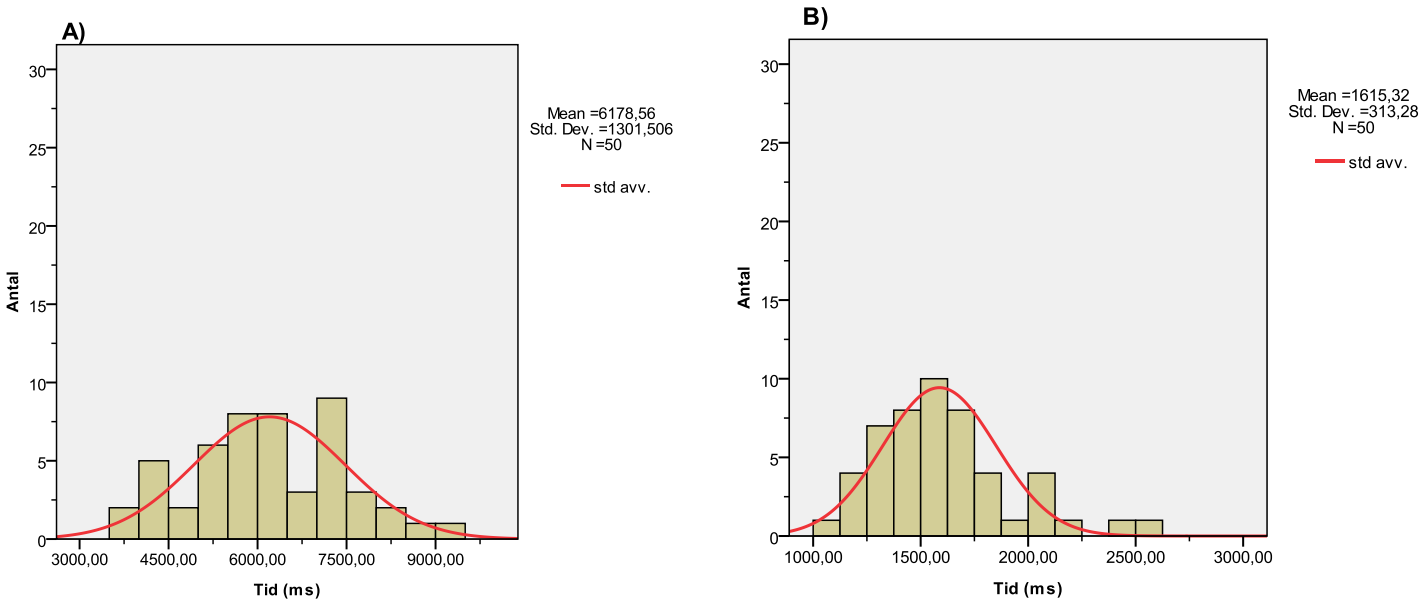
Figur 8. A) USB 2 1 Byte skrivning vid strömning, B) USB 3 1 Byte skrivning vid strömning.

Mätningarna för 1 kilobyte i figur 9 A) med c_v = 5,02 % och B) med c_v = 7,51 % visar att dessa värden är relativt lika i variationskoefficient. Men tittar man på max antalet mätningar i varje stapel så har USB 2 ett max på 6 mätningar vilket är jämfört med B) som illustrerar att i vissa staplar är max antalet dubbelt så högt. B) visar igen cirka 2 – 3 mätningar som är långsammare än resterande.



Figur 9. A) USB 2 1 Kbyte Skrivning utan strömning, B) USB 3 1 Kbyte Skrivning utan strömning.

Jämförs USB 2 och 3 i Figur 10 där det handlar om mätningar för 1 Kbyte skrivningar under tiden bandbredd allokeras är båda histogram väldigt lika varandra. Både max antalet i varje stapel, c_v = 21,06% för A och 19,39% för B samt utseendet på histogrammen ligger nära varandra. Det enda som kan påpekas är att i både figur 9 och 10 har B) ett fåtal mätningar där tiderna skiljer sig från majoriteten och är utanför normalfördelningskurvan.

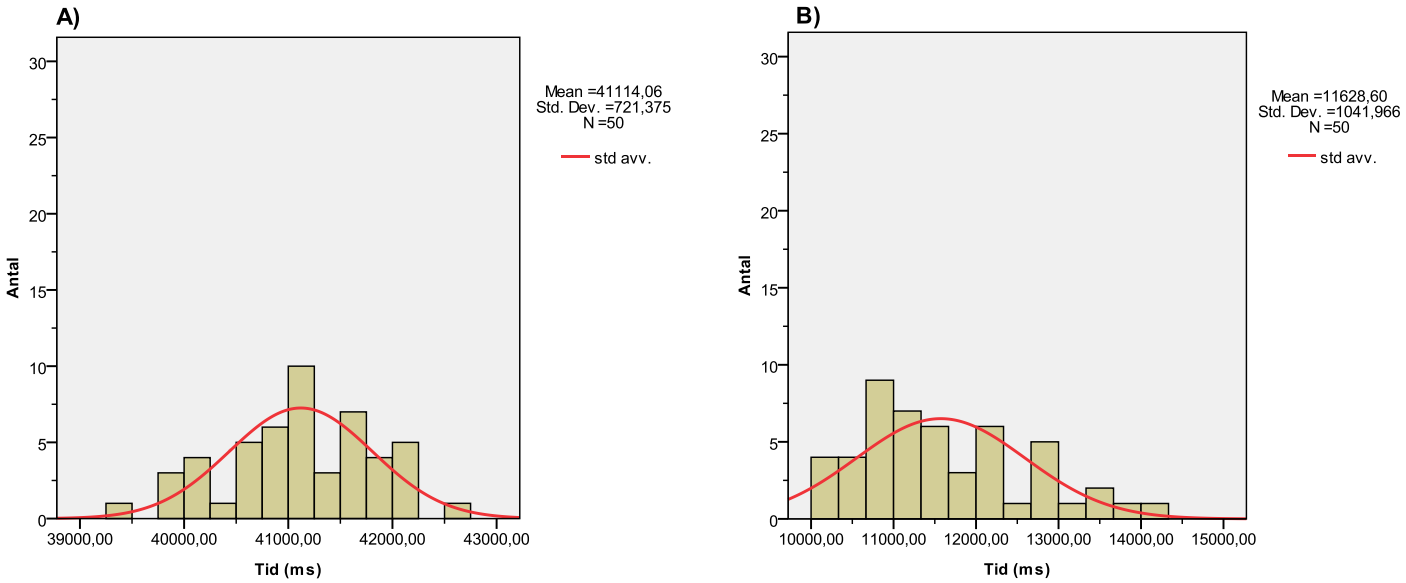


Figur 10. USB 2 1 Kbyte skrivning vid strömning, B) USB 3 1 Kbyte skrivning vid strömning.

Mätningarna med 1 megabyte stora filer i figur 11 visar ett mönster som inte har uppstått tidigare. B) har nämligen en högre standardavvikelse σ = 1 041,96 ms än A) σ = 721,37 ms och detta med en relativ stor marginal. c_v för B) är 8,96 % och enbart 1.75 % för A). Detta kan bero på ett flertal olika orsaker. En orsak kan vara de drivrutiner[3] som ingick när USB 3-produkten köptes. Ett annat kan vara att under testet någon slags programvara[4] i bakgrunden påverkade mätningarna negativt. Slutligen finns det möjlighet att när filstorleken ökade till 1 Megabyte kan USB 3 kräva mer prestanda ifrån systemet vilket inte räckte till. Och som man kom fram till i en artikel som hänvisades under relaterat arbete, verkar USB 3 vara mer känsligt angående systemets prestanda.[17]

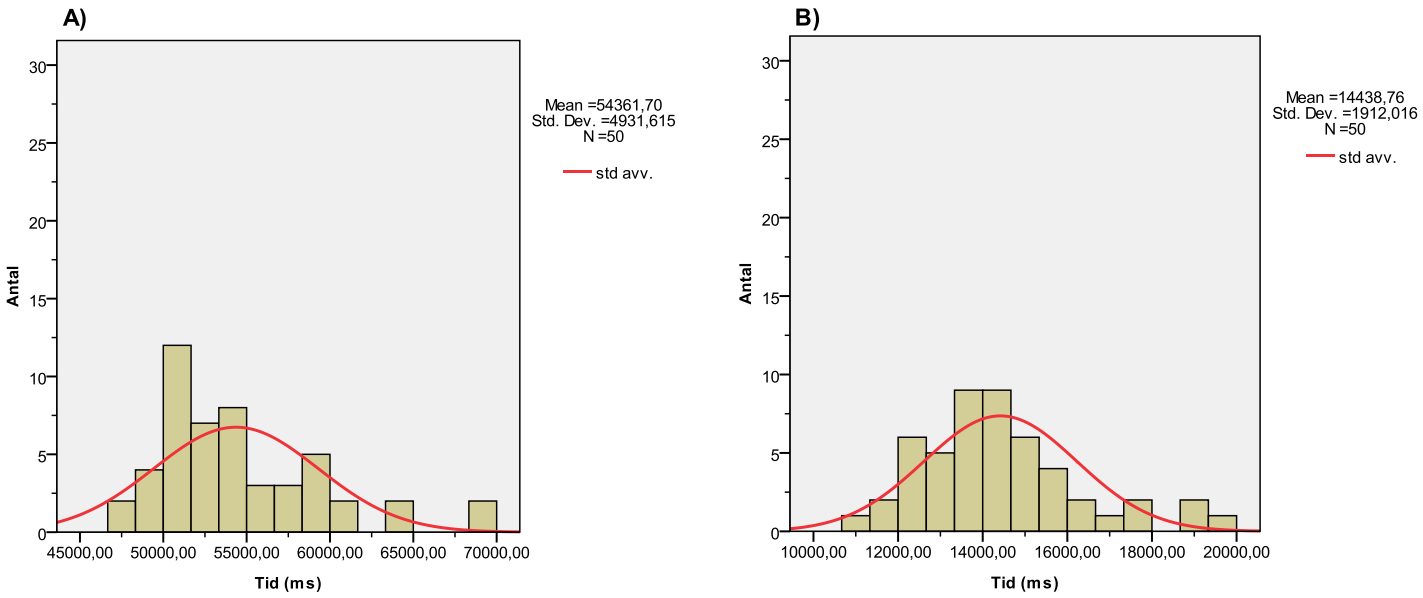
[3] Se kapitel 4.6 Felmarginaler

[4] Se kapitel 4.6 Felmarginaler



Figur 11. A) USB 2 1 Mbyte Skrivning utan strömning, B) USB 3 1 Mbyte Skrivning utan strömning.

Sista testet för skrivning visas i figur 12. Inga märkbara skillnader mellan USB 2 och 3 blev synliga. c_v för A) ligger på 9,07 % och B på 13,24 %. Jämförs dessa värden med de i figur 11, har USB 2 en ökning med nästan 5 gånger vilket låter dåligt. Ser man däremot på alla USB 2-mätningar för skrivning vid strömning är c_v = 9,07 % väldigt lågt och därmed en bra prestering. Samma sak för USB 2-mätningar utan strömning där c_v = 1,75 % i det här fallet och är det lägsta hittills. Däremot kan det inte sägas för USB 3, mellan 1 Mbyte skrivning i figur 11 och 12 ökade c_v cirka 1,5 gånger vilket låter bra. Men med tanke på att USB 3 i figur 11 ligger på nästan 9 % för c_v så är detta inget bra värde. Det betyder att USB 3 vid skrivning av 1 Mbyte stora filer hade procentuellt sett närmare samma prestanda på överföringar vid en ickeallokerat buss som när bussens bandbredd hade allokerats.



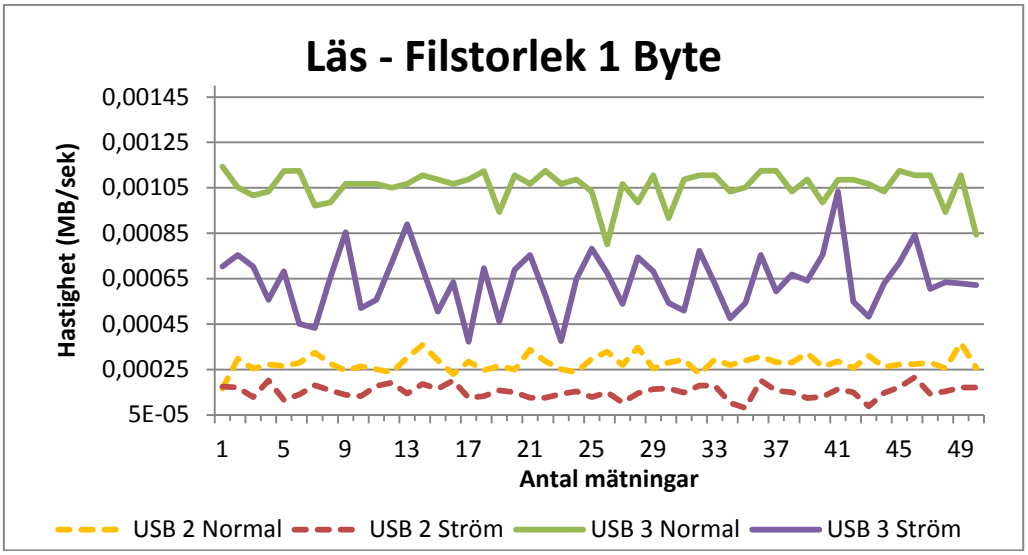
Figur 12. A) USB 2 1 Mbyte skrivning vid strömning, B) USB 3 1Mbyte skrivning vid strömning.

5.2 Läsning

I detta delkapitel presenteras resultatet med analys för den delen vi kallar för *läsning* i denna studie. Läsning delas in i två underkapitel för att få en lättare överblick av figurer och diagram.

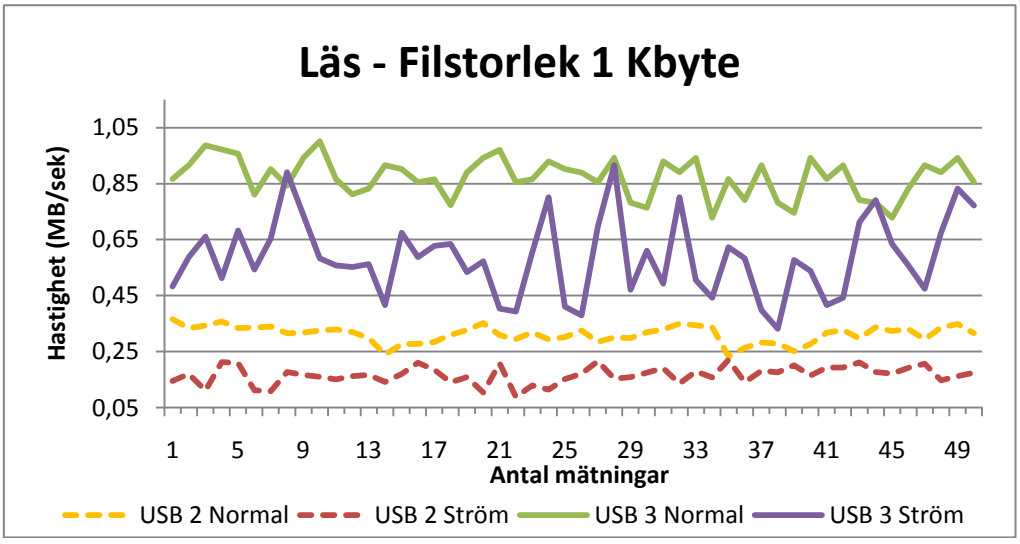
5.2.1 Bandbredd

Som för skrivning så hittar du här bandbreddshastigheter för läsning. Vid läsning av 1 byte stora filer i figur 13 visar USB 2 en medelhastighet på 0,28 KB/s på gult (obelastad USB-buss) och 0,15 på röd (belastad USB-buss). Resterande mätningarna lila samt grön illustrerar USB 3's hastigheter som ligger vid 1,05 och 0,64 KB/s. Mellan de båda versionerna visar USB 3 närmare fyra gånger högre hastigheter än USB 2. Hastighetsminskningen när bussen belastas ligger procentuellt på 39 % för USB 3 och 46 % för USB 2.



Figur 13. Hastigheter i bandbredd USB 2 & 3 använder vid mätningarna för läsning av 1 Byte stora filer. För att förtydliga skalan på Y-axeln så rör den sig med ett minimum värde på 0,05 KB/s och ett maximum på 1,45 KB/s.

Vid 1 kilobytesläsning mäts USB 2 medelhastigheter på 0,31 MB/s för gul, 0,17 MB/s för rött. USB 3 med en högre hastighet på 0,87 MB/s för grönt och 0,59 MB/s för lila. Minskningen i hastighet när USB-bussen är belastat och obelastat är 45 % för USB 2 och 32 % för USB 3. I figur 14 visar USB 3 2,8 gånger högre hastighet.

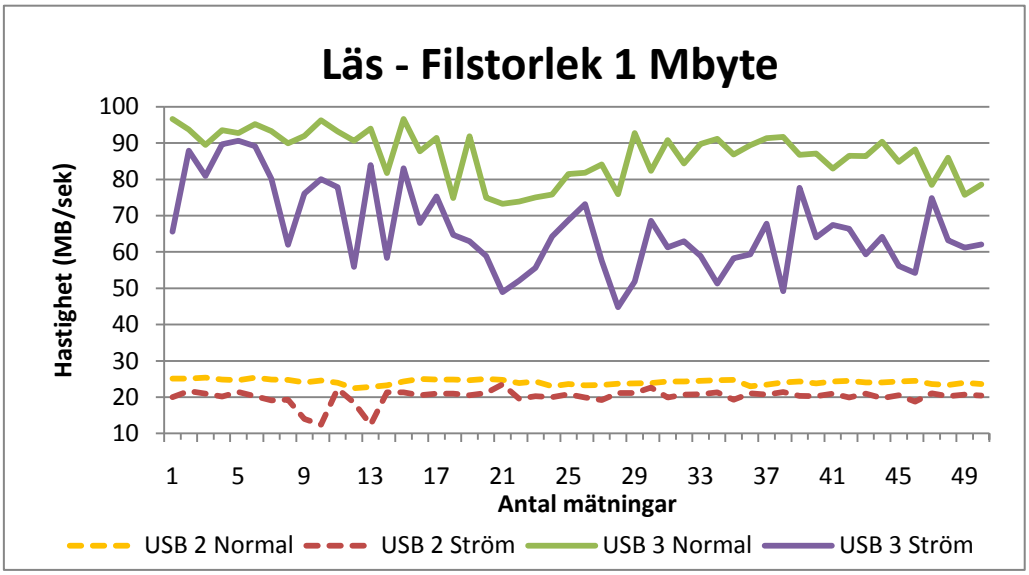


Figur 14. Hastigheter i bandbredd USB 2 & 3 använder vid mätningarna för läsning av 1 Kbyte stora filer.

I sista figuren figur 15 för bandbredd visas läsning av 1 megabyte stora filer. USB 3 visar hastigheter som är 3,6 gånger högre jämfört med USB 2. Medelhastigheter visar vara på 20,12 MB/s för rött, gult 24,17 MB/s, lila 66,33 MB/s samt 86,67 MB/s för grönt. Skillnaden mellan de två mätningarna utav varje USB-version var 19 % för USB

3 och 23 % för USB 2. Ser vi på föregående figurer 14 och 13 har hastighetsminskningen avtagit där USB 2 visar störst skillnad.

Generellt sett på alla bandbreddsmätningar som har mätts i denna studie, har USB 2 varit versionen som har haft den största minskningen i hastighet när USB-bussen belastades. Särskilt vid 1 Kbyte har detta varit synlig. Först vid 1 Mbyte mätningarna stabiliserades minskningen vilket visade relativt samma låga värde för både USB 2 och 3.

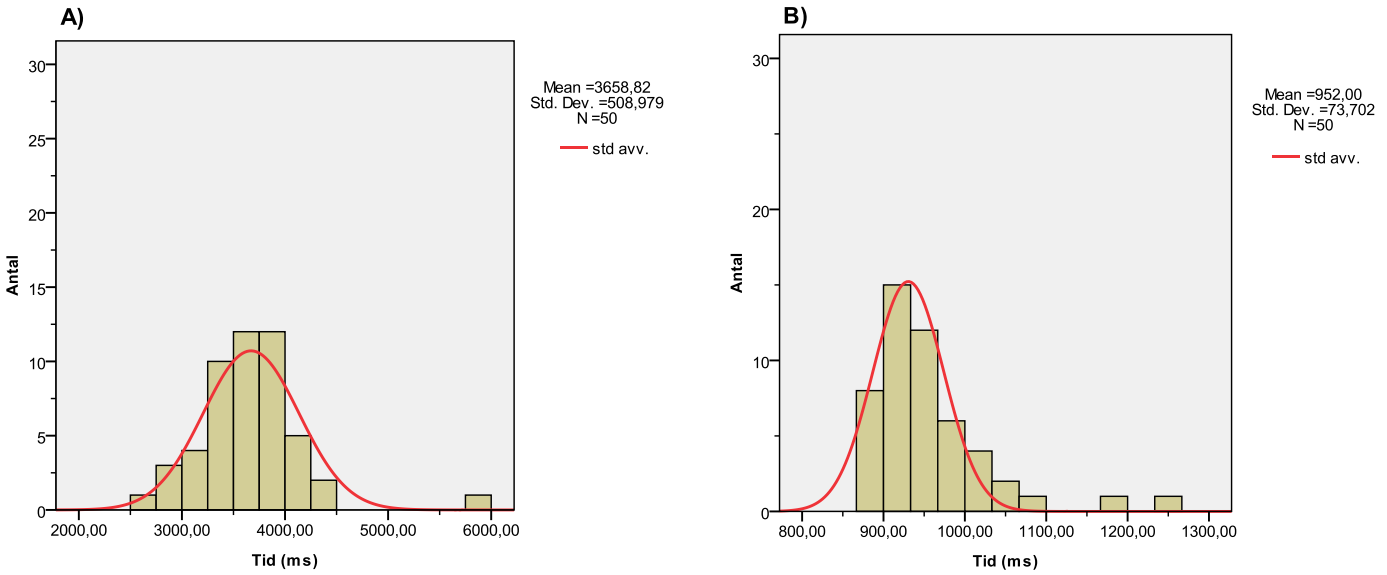


Figur 15. Hastigheter i bandbredd USB 2 & 3 använder vid mätningarna för läsning av 1 Mbyte stora filer.

5.2.2 Spridning

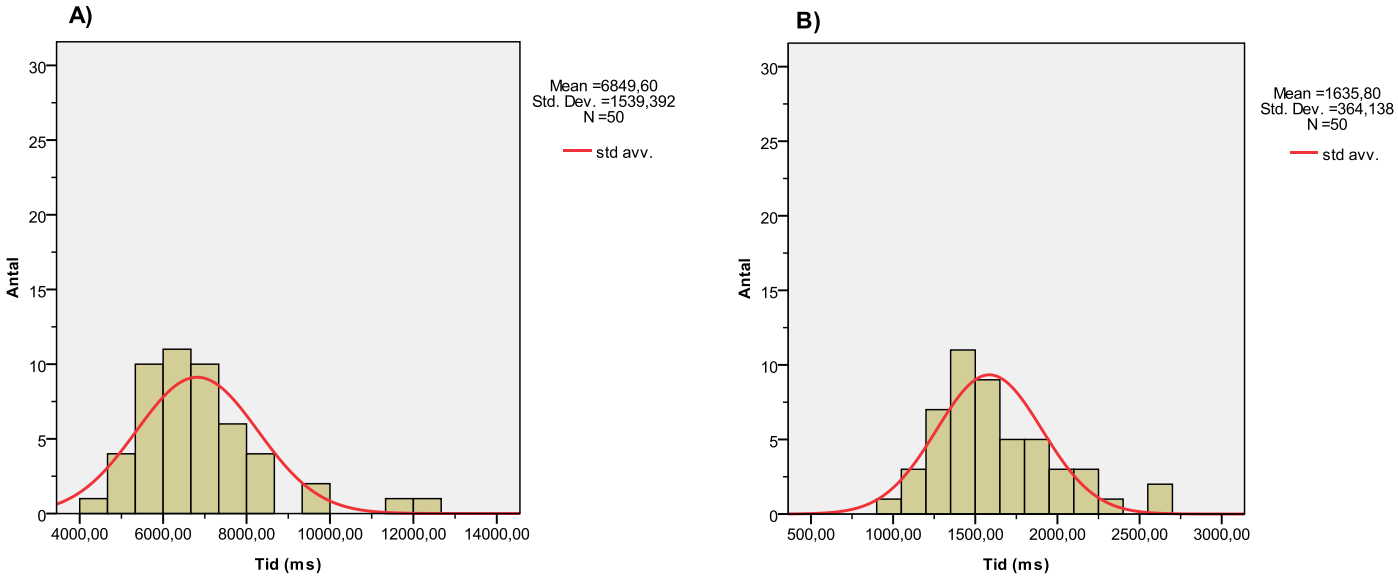
Histogrammen som använder ingen strömning under tiden mätningarna genomfördes är konstruerat enligt figur 2 A), respektive med strömning använts figur 2 B).

Mätningarna för läsning börjar med figur 16 där bortsett ifrån ett fåtal värden som avviker standardfördelningskurvan och variationskoefficienten verkar båda histogrammen likna varandra, med inga större skillnader. A) har c_v= 13,9 % och B) c_v= 7,7 % vilket visar ett större värde vid USB 2 men inget intressant.



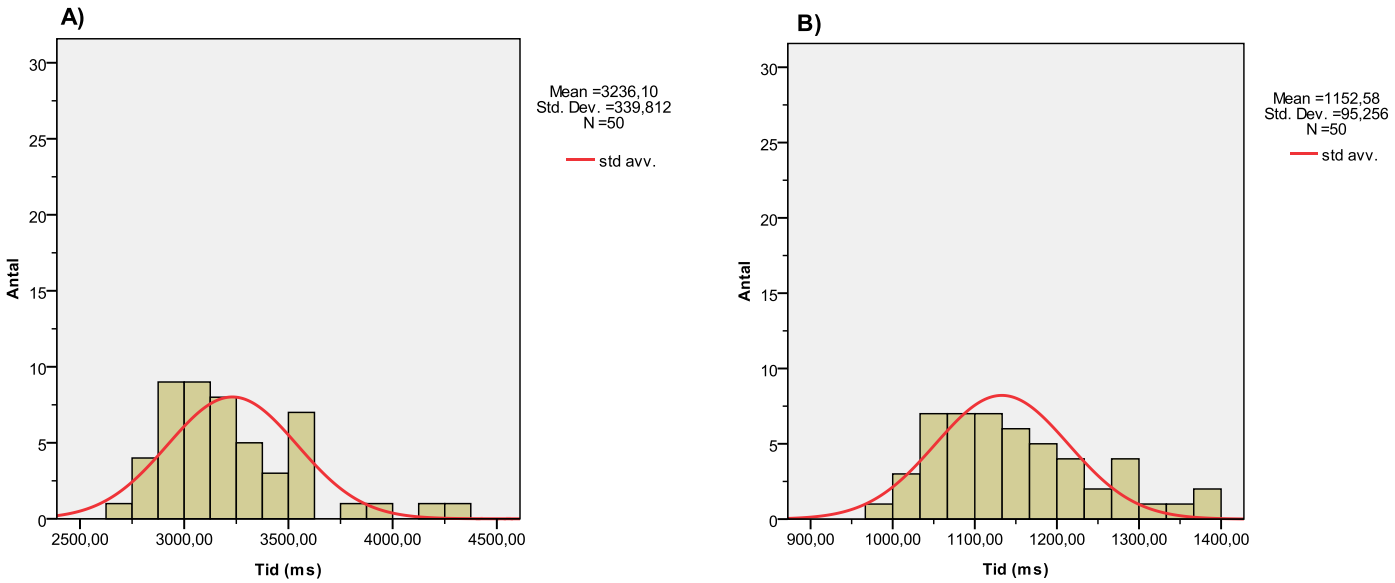
Figur 16. A) USB 2 1 Byte läsning utan strömning, B) USB 3 1 Byte läsning utan strömning.

I figur 17 är A) c_v= 22,4 % och B) c_v= 22,2 %. Som man ser är c_v till båda versioner utav USB väldigt lika varandra med endast en skillnad på 0,2 % mellan dessa. En till sak som man har sett i en del tidigare mätningar och även här, är att det finns ett litet antal mätningar utanför standardfördelningskurvan.



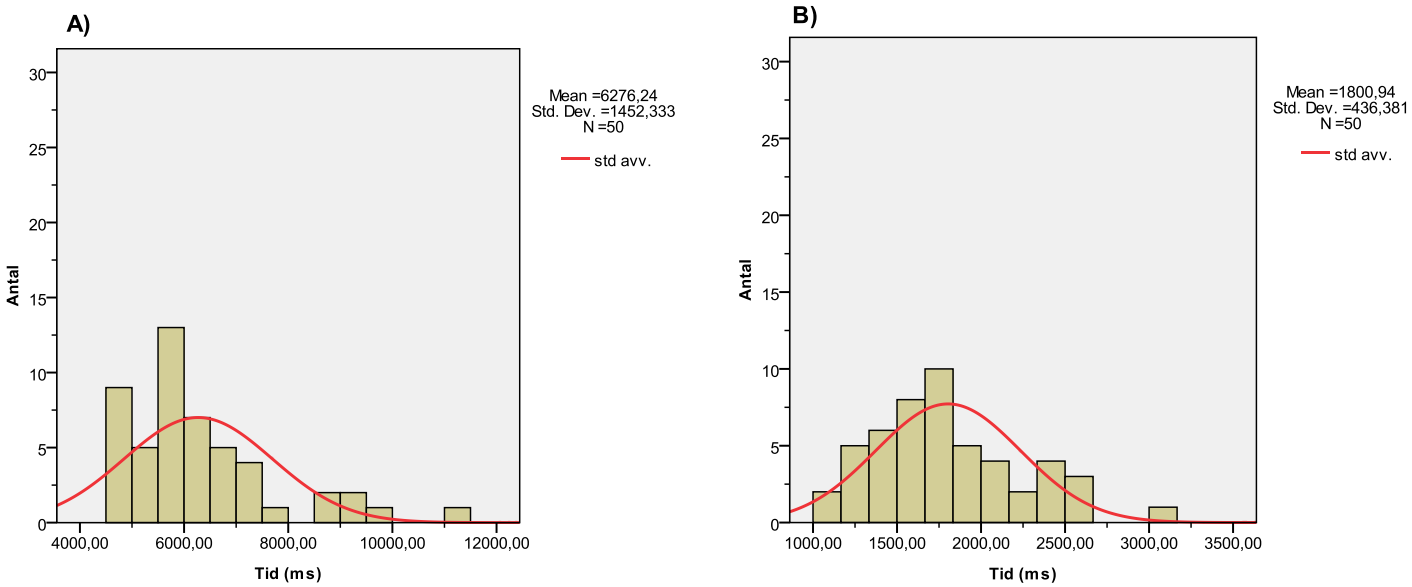
Figur 17. A) USB 2 1 Byte läsning vid strömning, B) USB 3 1 Byte läsning vid strömning.

Sedan vid mätningar av 1 Kbyte stora filerna utan strömning är A) c_v= 10,5 % och B) c_v= 8,2 %. Även här är värdena mellan båda USB versionerna nära varandra.



Figur 18. A) USB 2 1Kbyte läsning utan strömning, B) USB 3 1Kbyte läsning utan strömning.

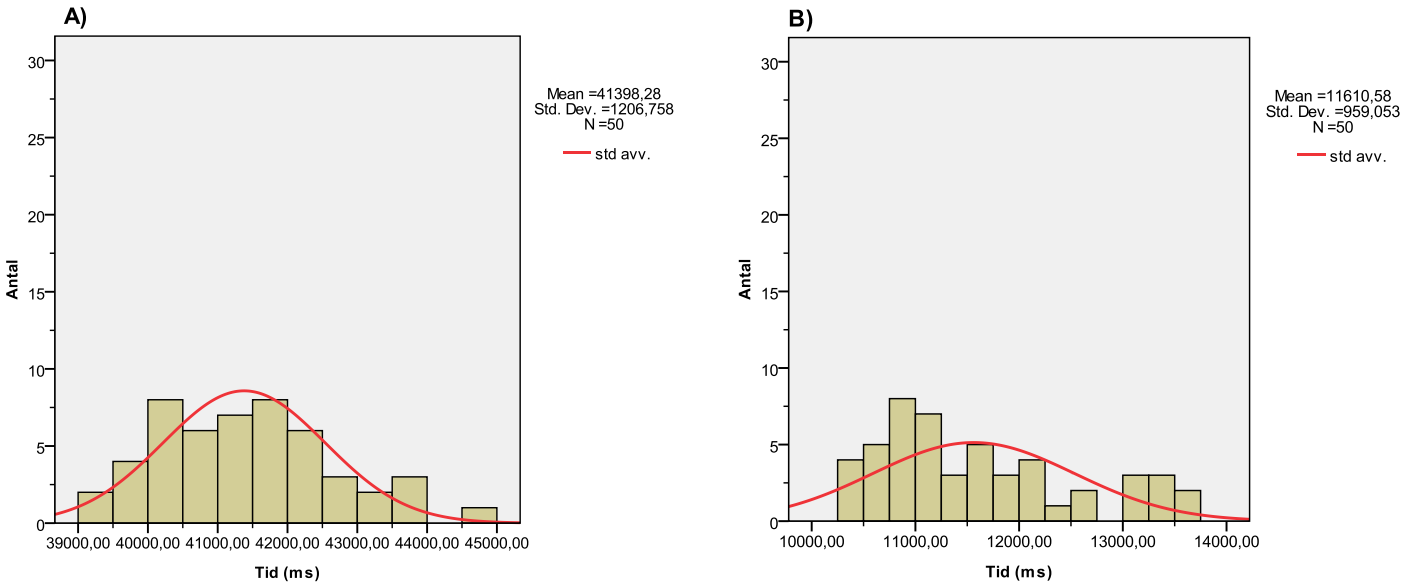
Figur 19 visar att c_v för USB 2 ligger på 23,14 % och USB 3 på 24,23 %. Mätningen repeterar föregående mätningsresultat. Det vill säga mellan USB 2 och 3 uppstod inga större synliga skillnader. Jämförs båda 1 Kbyte mätningar A och B i figur 18 och 19 så ökar c_v vid A) med det dubbla och vid B) upp till tre gånger. Vilket är rimligt om man jämför det med föregående mätningar.



Figur 19. A) USB 2 1 Kbyte läsning vid strömning, B) USB 3 1 Kbyte läsning vid strömning.

Igen i figur 20 vid 1 Mbyte mätningar utan strömning sker större avvikelser i histogrammen. A) har c_v = 2,9 % och B) c_v = 8,2 %. Som man ser är USB 3's c_v värde mycket högre än USB 2 vilket hittills vid läsning har varit tvättom. Sedan har

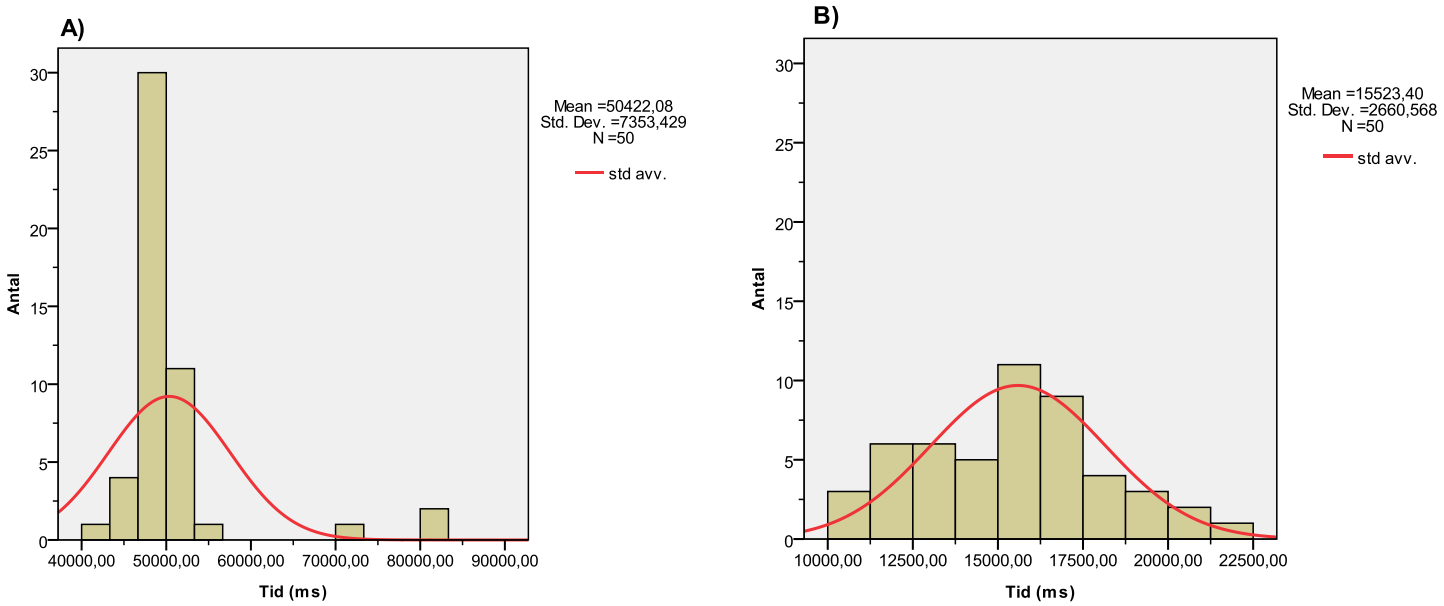
det bildat sig en liten samling värden precis vid gränsen av standardfördelningskurvan vid B), vilket kan vara en av orsakerna till att c_v för USB 3 blev så hög. Dessa värden kan ha orsakats av samma anledning som togs upp tidigare vid figur 11 i kapitel 5.1.2.



Figur 20. A) USB 2 1 Mbyte läsning utan strömning, B) USB 3 1 Mbyte läsning utan strömning.

Mätvärden för 1 Mbyte stora filer vid läsning medan bandbredd har allokerats som figur 21 visar resultatet på, illustreras en stor skillnad mellan USB-versionerna. USB 2 har $c_v = 14{,}6$ % och för USB 3 är $c_v = 17{,}13$ %. Återigen visar USB 3 ett större värde än USB 2. Bortsett ifrån avvikelsen på c_v visar USB 2 tre mätvärden som har hamnat utanför standardfördelningskurvan med en marginal på ca 20000 – 30000 ms ifrån medelvärdet. Vad dessa två saker kan beror på är oklart. En möjlighet är att någon programvara i bakgrunden påverkade mätningen och därmed blev viss data fördröjd.

Det utseendet i figur 21 A) som visar ett max antal på 30 mätningar i samma stapel beror på följande. Skälet är de försenade värden som har hamnat så långt ifrån resterande värden. På grund av dessa värden ökade skalan för Y-axeln vilket resulterade i att 30 mätningar hamnade i samma stapel.



Figur 21. A) USB 2 1 Mbyte läsning vid strömning, B) USB 3 1 Mbyte läsning vid strömning.

För att sammanställa en generell bild av spridningen som hittills har illustrerats i resultatet innan visar USB 2 och 3 följande. När USB-bussen är obelastat har USB 2´s spridning en tendens att minska ju större filen som överförs blir. Vid största filstorleken visar USB 2 väldigt låga värden som ligger på cirka 2-3 % vilket är bra. USB 3 däremot visar något helt annat, nämligen istället för minskningar håller sig spridningen runt 7-9 % med ett undantag på 1 byte stora filer där det mättes till 3 %. Vid belastning av bussen så visar USB-versionerna en helt annorlunda bild. I majoritet håller sig spridningsvärdet för båda versionerna på samma nivå med en maximal skillnad på inte mer än 1-2%. Dock även här finns det ett tillfälle som avviker resterande, spridningen för 1 Mbyte filer. USB 2 ligger runt 3-4 % lägre vid både läsning och skrivning.

6 Slutsats

Med hjälp av de tolv testfall som användes kunde följande saker fastställas. Som förväntat visade USB 3 en betydligt högre överföringshastighet än USB 2. Dock kunde inte dess fulla potential på hastighet visas i denna studie då utrustningen som användes begränsade hastigheten. Sammanfattningsvis visade USB 3 enbart bättre värden på spridning när USB-bussen inte belastades vid den lägsta filstorleken, det vill säga 1 bytes överföringar. Vid de resterande testfall hade båda versionerna antingen liknande värden eller som det visade sig vid 1 megabytesfiler hade USB 2 lägre spridning jämfört med USB 3.

Källförteckning

1. Axelson, Jan. (2009). *USB Complete: The Developer's Guide*. Fourth Edition. Lakeview Research. ISBN: 978-1-931448-08-6
2. Beyond Logic. (2007). *USB in a NutShell – making sense of the USB standard.* [Elektronsik]. Tillgängligt: < http://www.beyondlogic.org/usbnutshell/usb1.htm >. [2010-08-19]
3. Corbet, Jonathan & Rubini Alessandro & Kroah-Hartman, Greg. (2005). *Linux Device Drivers*. Third edition. O`Reilly Media, Inc. ISBN: 0-596-00590-3
4. Harmon, Dan. (2010). *Superspeed USB (USB 3.0): More than just increase of speed.* [Elektronisk]. Tillgängligt: < http://www.eetimes.com/design/analog-design/4201110/SuperSpeed-USB--USB-3-0--More-than-just-a-speed-increase >. [2010-06-13]
5. Hawkesmoor, Nick.(2009).*What makes superspeed tick.* [Elektronisk]. Situation Publishing Ltd. Tillgängligt: <http://www.reghardware.com/2009/05/25/superspeed_usb_3_guide/>. [2010-08-22]
6. Innovative Logic, Inc. (2010). *USB 3.0 Superspeed overview.* [Elektronisk]. Tillgängligt: < http://www.inno-logic.com/resourcesUSB3.html >. [2010-07-20]
7. Intel & Microsoft & Hewlett-Packard. (2000). *Universal Serial Bus Specification.* Revision 2.0. [Elektronisk]. Tillgängligt: <http://www.usb.org/developers/docs/>. [2010-08-18]
8. Intel Corporation & Microsoft Corporation & Hewlett-Packard Company. (2008). *Unsiversal Serial Bus 3.0 Specification.* Revision 1.0. [Elektronisk]. Tillgänligt: <http://www.usb.org/developers/docs/>. [2010-08-18]
9. Jain, Rain. (1991). *The Art of Computer Systems Performance Analysis*. Second Edition. Wiley. ISBN: 0-471-50336-3
10. Matroska.com. (2010). *What is* matroska. [Elektronisk]. Tillgängligt: < http://www.matroska.org/technical/whatis/index.html >. [2011-05-23]
11. McDowell, Steven & D.Seyer, Martin(1999). *USB explained.* USA: Prentice-Hall, Inc. (ISBN 0-13-081153-X)
12. Meyev, Alesey. (2010). *USB 3.0: Theory and Practice.* [Elektronisk]. Tillgängligt: < http://www.xbitlabs.com/articles/storage/display/usb-3.html >. [2010-07-18]
13. Stats4Students. (2010). *Measures of spread.* [Elektronisk]. Tillgängligt: < http://stats4students.com/measures-of-spread-3.php >. [2010-08-19]

14. Techbook Report. (2006). *Standard Deviation in 30 seconds.* [Elektronisk]. Tillgängligt: < http://www.techbookreport.com/tutorials/stddev-30-secs.html >. [2010-08-15]

15. Videolan. (2010). *VLC features.* [Elektronisk]. Tillgängligt: <http://www.videolan.org/vlc/features.html> [2010-07-25]

16. Videolan. Wiki. (2010). *Streaming how to/Easy Streaming.* [Elektronisk]. Tillgängligt: <http://wiki.videolan.org/Documentation:Streaming_HowTo/Easy_Streaming> [2010-07-25]

17. Wahlqvist, Patrik. (2010). *Prestandaskillnad mellan USB 2 och USB 3.* Datormagazin, Nummer 9 2010. Sida 62-63

A. Programkod för DataSim

Koden är skriven i C och är för testprogrammet datasim som använts i denna studie. Utskriften för programkoden i nedan har formaterats något för att passa till mallen som använts. Vid kompilering kontrollera eventuella formateringsfel.

```
#include <stdio.h>
#include <dirent.h>
#include <windows.h>

#define ARGUMENT_COUNT 5
#define TEMP_FILENAME "temp"
#define MAX_FILE_COUNT_LENGTH 6

int directoryExists(const char* path) {
     if (path == NULL)
     return 0;
     DIR *directory;
     directory = opendir(path);
     if (directory != NULL) {
             (void) closedir(directory);
             return 1;
     }
    return 0;
}

int main (int argumentCount, char* arguments[]) {
     unsigned long fileSize;
     unsigned int fileCount, error = 0, i, j;
     char *fromPath;
     char *toPath;

     // Check for errors in arguments
     if (argumentCount != ARGUMENT_COUNT) {
             printf("Wrong number of arguments entered.\n");
             error = 1;
     }
     else {
             char argumentError[] = "Argument %s where not correctly
             detected!\n";
```

```
        if (sscanf(arguments[1], "%lu", &fileSize) <= 0) {
                printf(argumentError, "filesize_in_bytes");
                error = 1;
        }
        if (sscanf(arguments[2], "%du", &fileCount) <= 0 ||
        strlen(arguments[2]) > MAX_FILE_COUNT_LENGTH) {
                printf(argumentError, "file_count");
                error = 1;
        }
        char pathError[] = "The folder for argument %s can't be
        found!\n";
        fromPath = arguments[3];
        if (!directoryExists(fromPath)) {
                printf(pathError, "from_path");
                error = 1;
        }
        toPath = arguments[4];
        if (!directoryExists(toPath)) {
                printf(pathError, "to_path");
                error = 1;
        }
}
// Display program info if errors occured
if (error) {
        printf("\nRequired:\ncopytest filesize_in_bytes
        file_count from_path to_path\n\n");
        printf("Example:\ncopytest 10000 100 \"c:\\from\"
        \"d:\\to\"\n");
        return 0;
};
// Append full path to the temp file
char* fullPath = (void*) malloc(strlen(fromPath) + 1);
strcpy(fullPath, fromPath);
if (fromPath[strlen(fromPath)-1] != '\\')
        strcat(fullPath, "\\");
strcat(fullPath, TEMP_FILENAME);
// Create the test file
FILE *pFile = fopen(fullPath, "w");
if (pFile != NULL) {
        for (j = 0; j < fileSize; j++)
```

```
                putc('0', pFile);
                fclose(pFile);
        }
        // Fix paths
        if (toPath[strlen(toPath)-1] != '\\')
        strcat(toPath, "\\");
        // Move all file paths into an array to reduce impact on the
        actual test
        char* toPaths[fileCount];
        for (i = 0; i < fileCount; i++) {
                // Convert int to string
                char temp[MAX_FILE_COUNT_LENGTH + 1];
                sprintf(temp, "%d", i);
                toPaths[i] = (void*) malloc(strlen(toPath) + 1);
                strcpy(toPaths[i], toPath);
                strcat(toPaths[i], temp);
        }
        // Copy files and measure total time
        DWORD startTime = GetTickCount();
        for (i = 0; i < fileCount; i++) {
                // Create the file with a function from the Win32 api
                for maximum performance
                CopyFile((LPCTSTR) fullPath, (LPCTSTR) toPaths[i],
        FALSE);
        }
        DWORD totalTime = GetTickCount()-startTime;
        // Remove test files
        remove(fullPath);
        for (i = 0; i < fileCount; i++) {
                remove(toPaths[i]);
        }
        // Print results
        printf("Total time taken: %d ms\n", totalTime);
        printf("Time/copy: %f ms\n", totalTime/(double)fileCount);
        printf("Average Speed: %f Mb/s",
        (fileSize*fileCount)/(((double)totalTime)/((double)1000))/((dou
        ble)1000000));


        return 0;
}
```

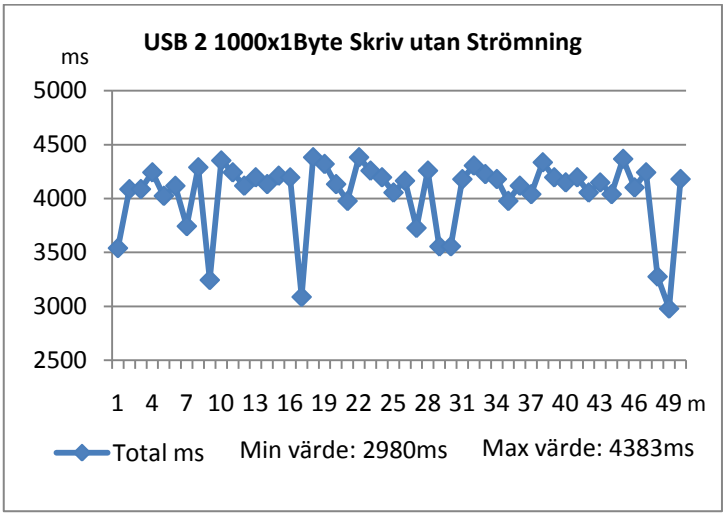
B. Bashskript

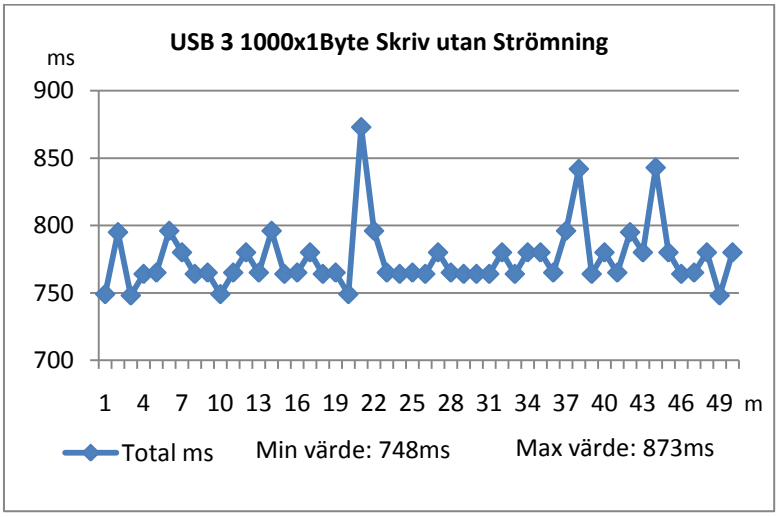
Bashskript använts i denna studie. Variablerna som måste anges i scriptet är storlek, antal, källa och destination för att programmet datasim skall fungera korrekt. Längst ner i skriptet ska det även anges antal iterationer som bestämmer hur ofta skriptet skall anropa programmet datasim.

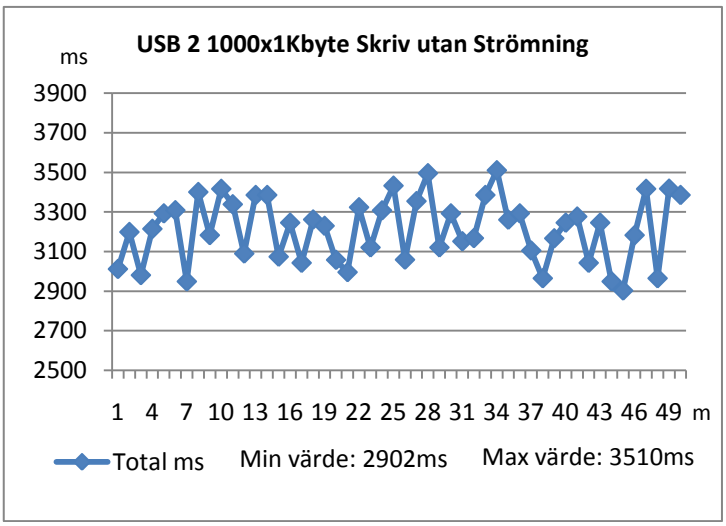
```
@echo off
if not exist temp mkdir temp
if not exist temp2 mkdir temp2
echo Kopierar antal St storlek Byte stora filer
Set Count=1
:LOOP
echo Count= %Count%
copytest.exe storlek antal "Källa" "Destination"
echo.
echo.
Set /A Count=%Count%+1
If %Count% lss iterationer GoTo :LOOP
Pause
```

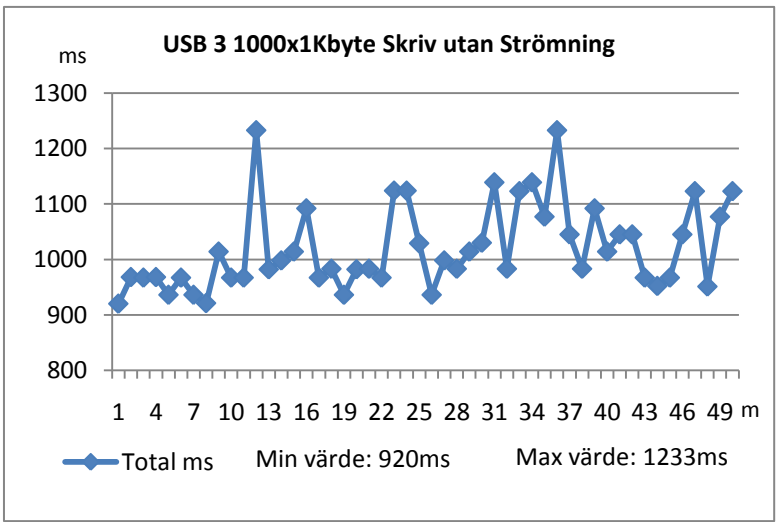
C. Mätdata (tid)

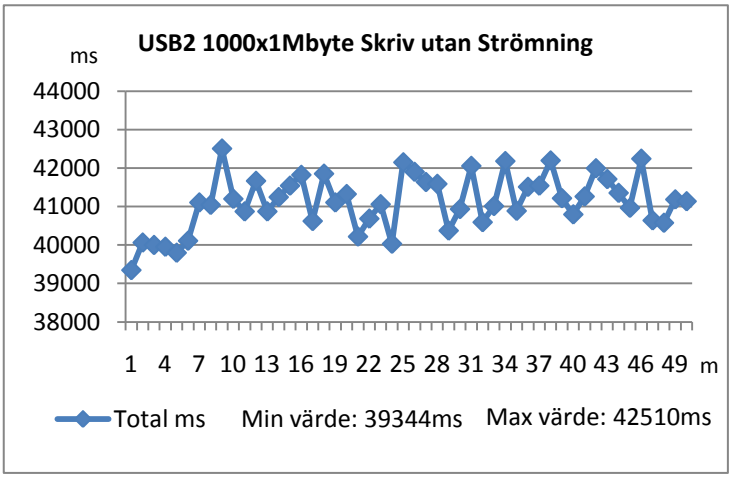
I alla grafer står Y-axeln för tid i millisekunder och X-axeln för antal mätningar som står för ”m”. Graferna illustrerar mätvärden ifrån de olika mätningar som har genomfört under arbetet.

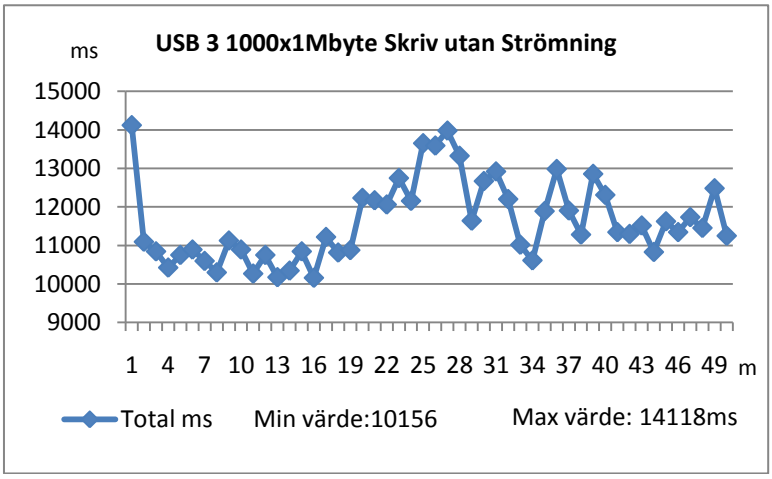


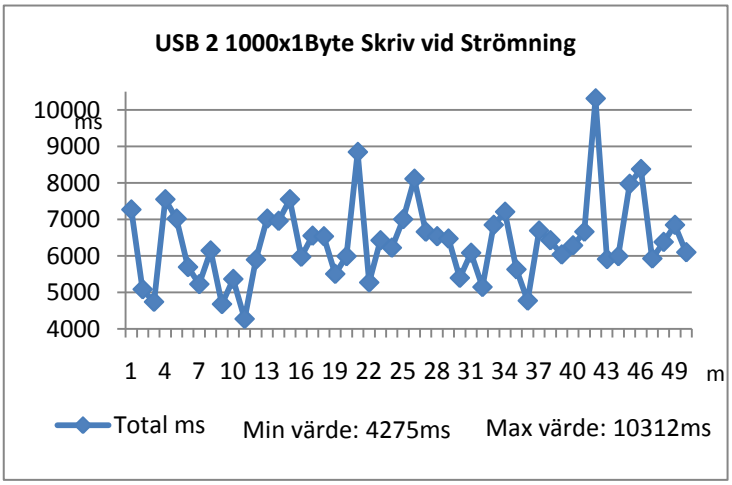


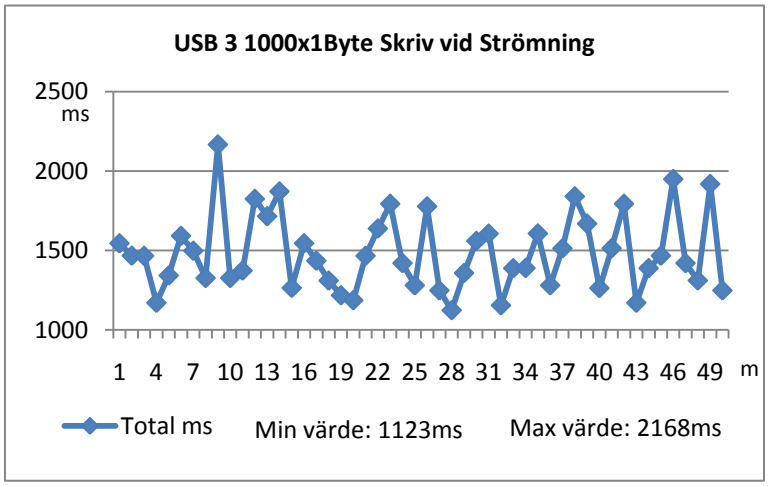


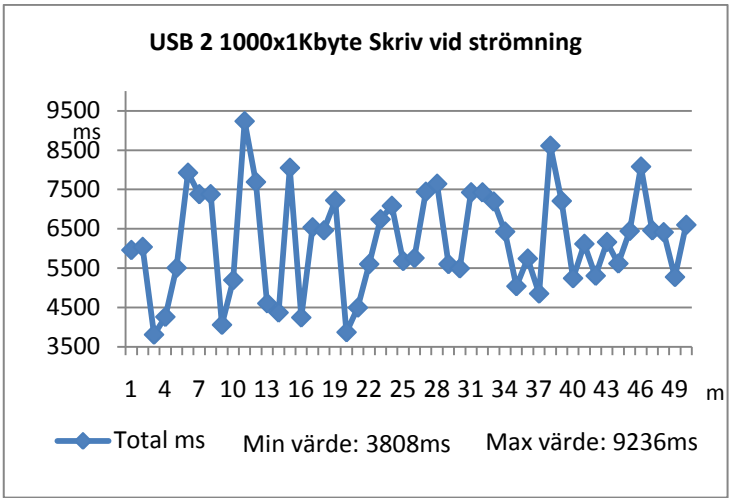


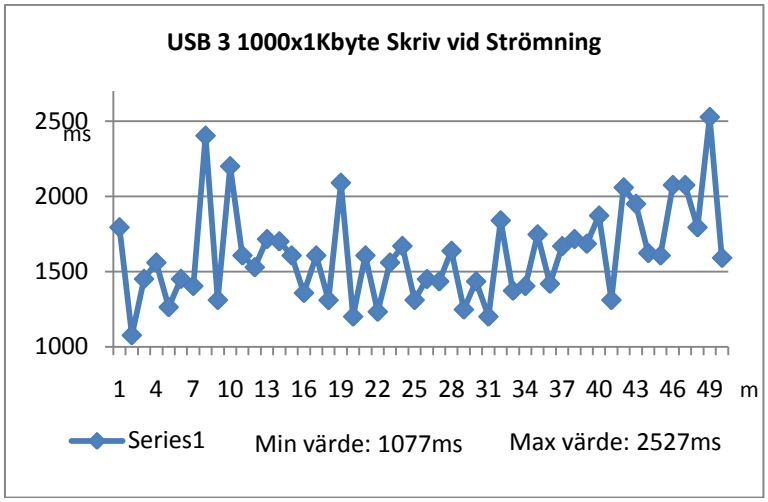

USB2 1000x1Mbyte Skriv utan Strömning
ms
44000
43000
42000
41000
40000
39000
38000
1 4 7 10 13 16 19 22 25 28 31 34 37 40 43 46 49 m
Total ms
Min värde: 39344ms
Max värde: 42510ms

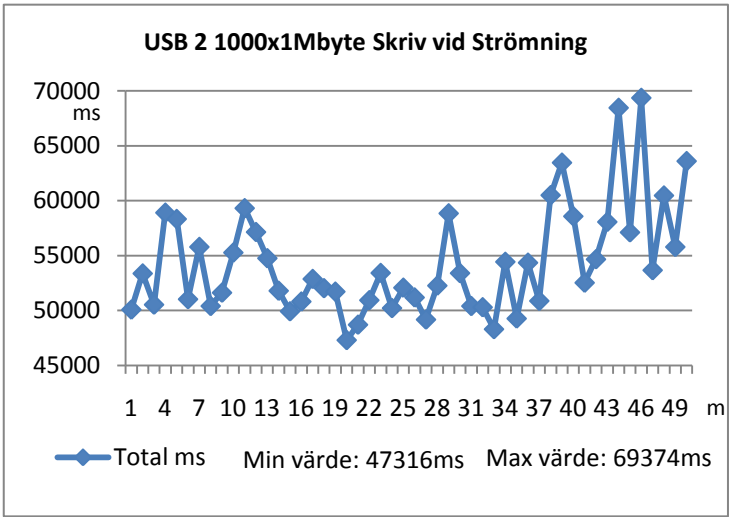

USB 3 1000x1Mbyte Skriv utan Strömning
ms
15000
14000
13000
12000
11000
10000
9000
1 4 7 10 13 16 19 22 25 28 31 34 37 40 43 46 49 m
Total ms
Min värde:10156
Max värde: 14118ms

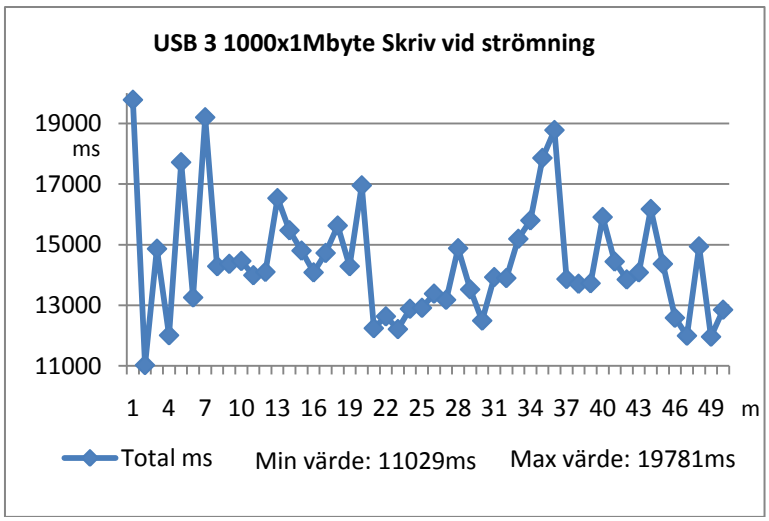

USB 2 1000x1Byte Skriv vid Strömning
10000
ms
9000
8000
7000
6000
5000
4000
1 4 7 10 13 16 19 22 25 28 31 34 37 40 43 46 49 m
Total ms
Min värde: 4275ms
Max värde: 10312ms

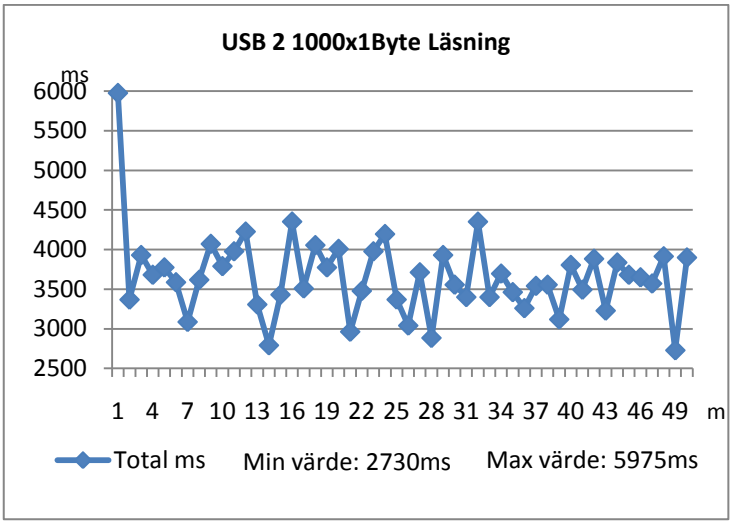

USB 3 1000x1Byte Skriv vid Strömning
2500
ms
2000
1500
1000
1 4 7 10 13 16 19 22 25 28 31 34 37 40 43 46 49 m
Total ms
Min värde: 1123ms
Max värde: 2168ms

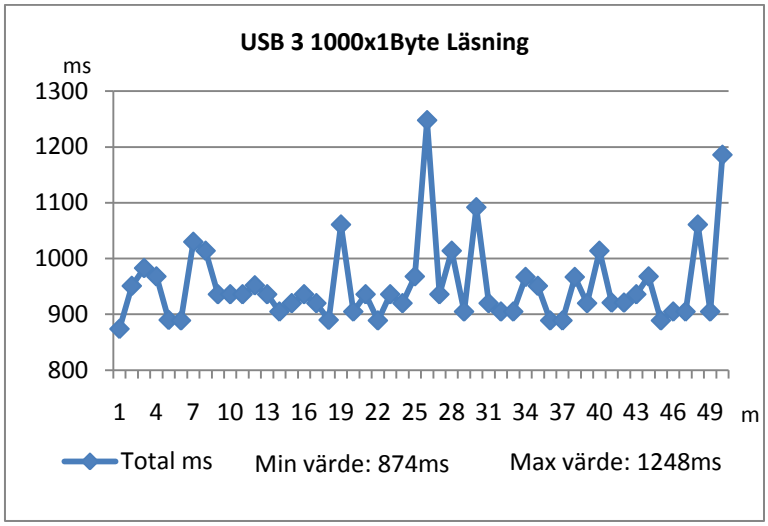

USB 2 1000x1Kbyte Skriv vid strömning
9500
ms
8500
7500
6500
5500
4500
3500
1 4 7 10 13 16 19 22 25 28 31 34 37 40 43 46 49
m
Total ms
Min värde: 3808ms
Max värde: 9236ms

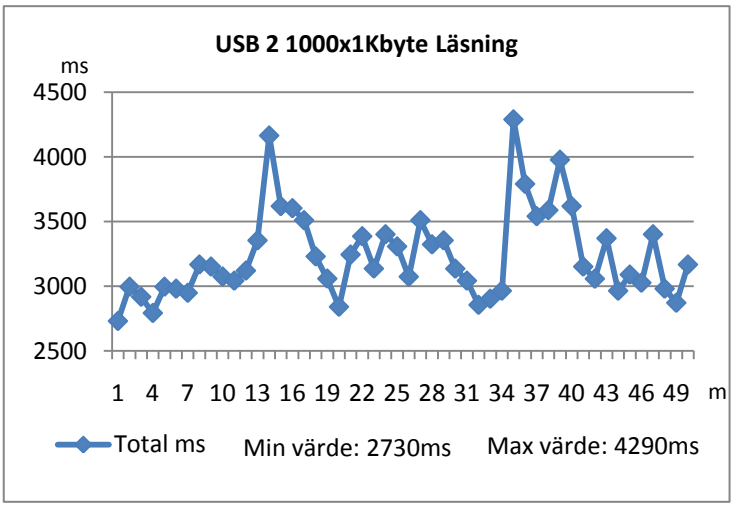

USB 3 1000x1Kbyte Skriv vid Strömning
2500
ms
2000
1500
1000
1 4 7 10 13 16 19 22 25 28 31 34 37 40 43 46 49
m
Series1
Min värde: 1077ms
Max värde: 2527ms

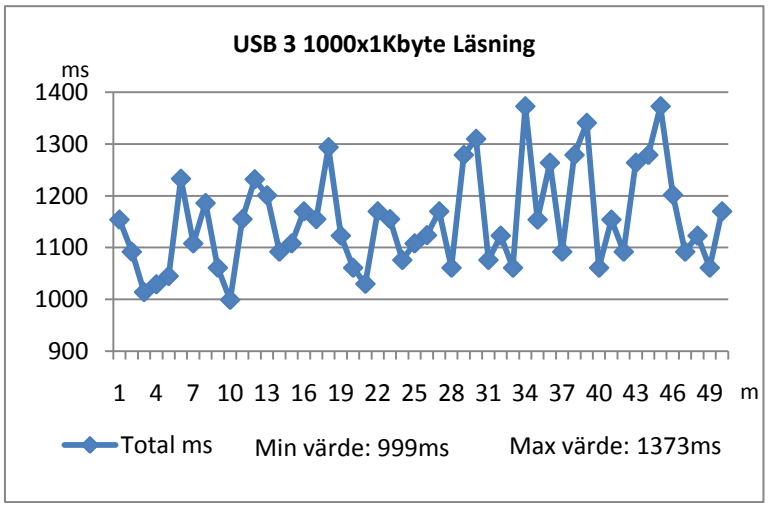

USB 2 1000x1Mbyte Skriv vid Strömning
70000
ms
65000
60000
55000
50000
45000
1 4 7 10 13 16 19 22 25 28 31 34 37 40 43 46 49
m
Total ms
Min värde: 47316ms
Max värde: 69374ms

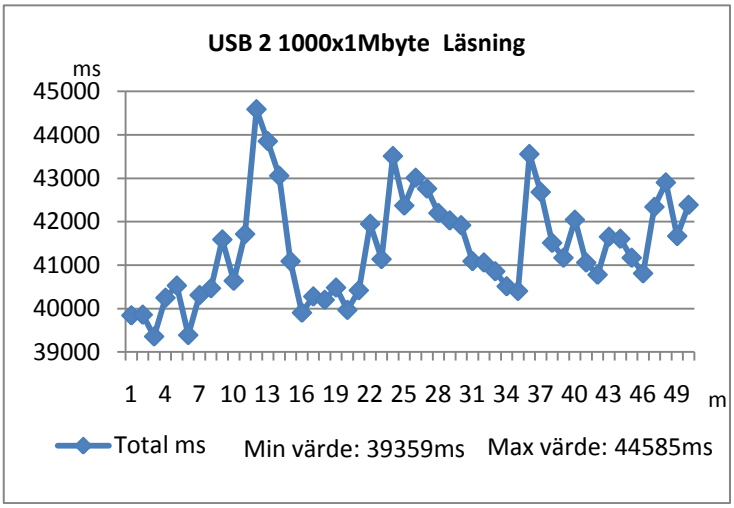

USB 3 1000x1Mbyte Skriv vid strömning
19000
ms
17000
15000
13000
11000
1 4 7 10 13 16 19 22 25 28 31 34 37 40 43 46 49
m
Total ms
Min värde: 11029ms
Max värde: 19781ms

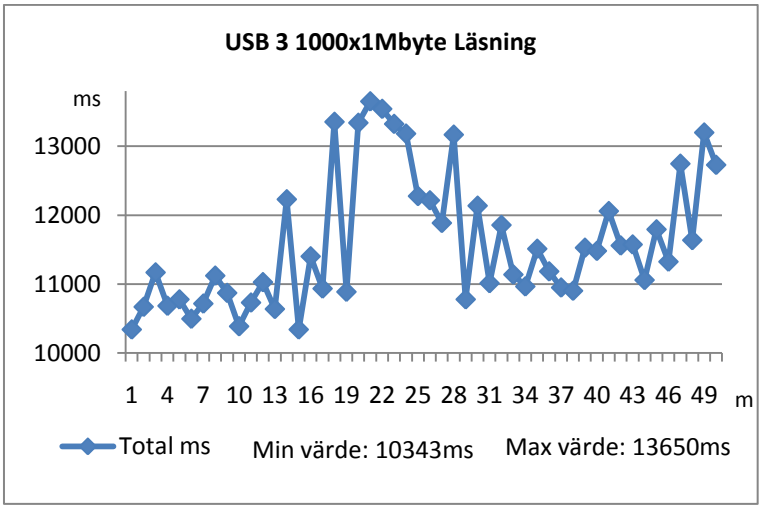

USB 2 1000x1Byte Läsning
ms
6000
5500
5000
4500
4000
3500
3000
2500
1 4 7 10 13 16 19 22 25 28 31 34 37 40 43 46 49
m
Total ms
Min värde: 2730ms
Max värde: 5975ms

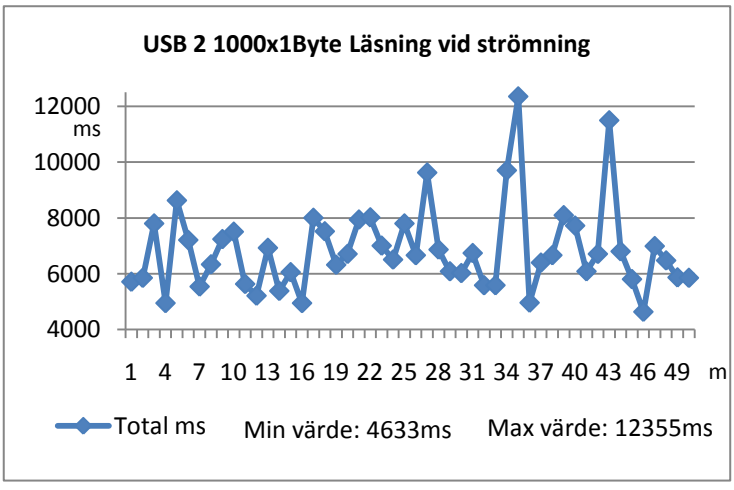

USB 3 1000x1Byte Läsning
ms
1300
1200
1100
1000
900
800
1 4 7 10 13 16 19 22 25 28 31 34 37 40 43 46 49
m
Total ms
Min värde: 874ms
Max värde: 1248ms

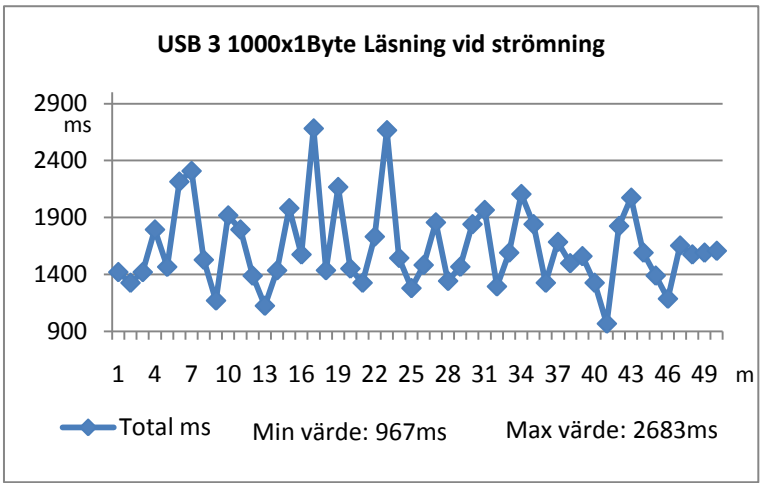

USB 2 1000x1Kbyte Läsning
ms
4500
4000
3500
3000
2500
1 4 7 10 13 16 19 22 25 28 31 34 37 40 43 46 49
m
Total ms
Min värde: 2730ms
Max värde: 4290ms

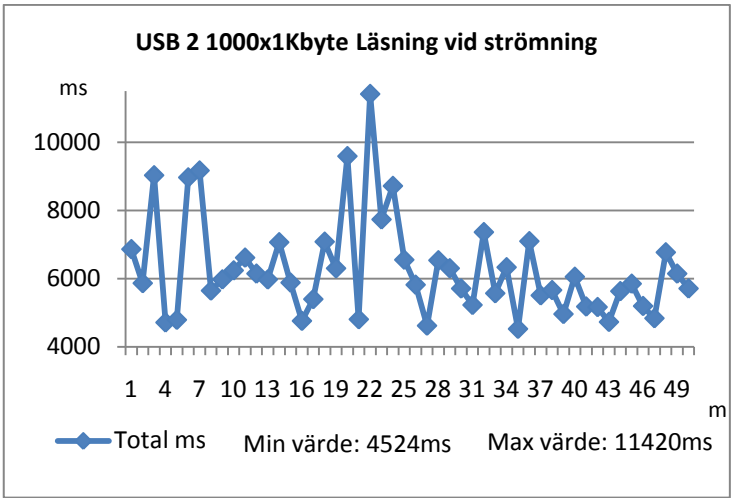

USB 3 1000x1Kbyte Läsning
ms
1400
1300
1200
1100
1000
900
1 4 7 10 13 16 19 22 25 28 31 34 37 40 43 46 49
m
Total ms
Min värde: 999ms
Max värde: 1373ms

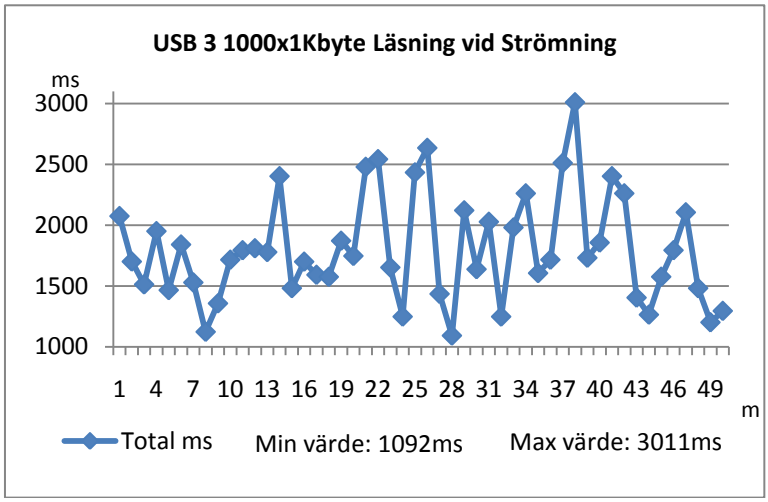

USB 2 1000x1Mbyte Läsning
ms
45000
44000
43000
42000
41000
40000
39000
1 4 7 10 13 16 19 22 25 28 31 34 37 40 43 46 49
m
Total ms
Min värde: 39359ms
Max värde: 44585ms

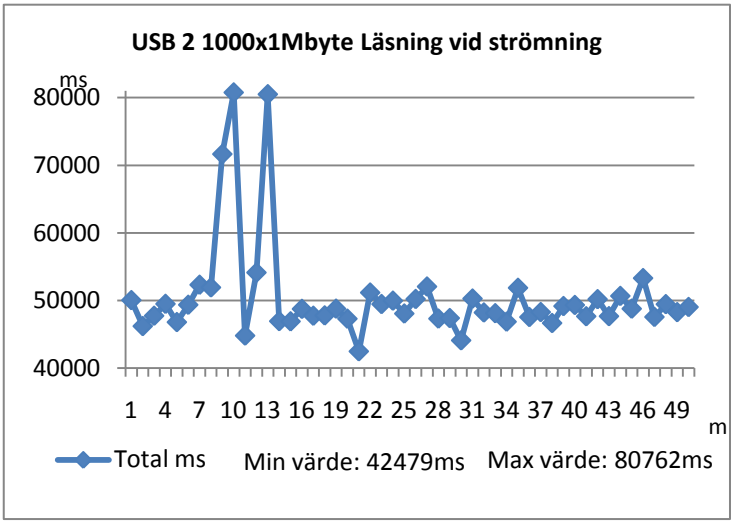

USB 3 1000x1Mbyte Läsning
ms
13000
12000
11000
10000
1 4 7 10 13 16 19 22 25 28 31 34 37 40 43 46 49
m
Total ms
Min värde: 10343ms
Max värde: 13650ms

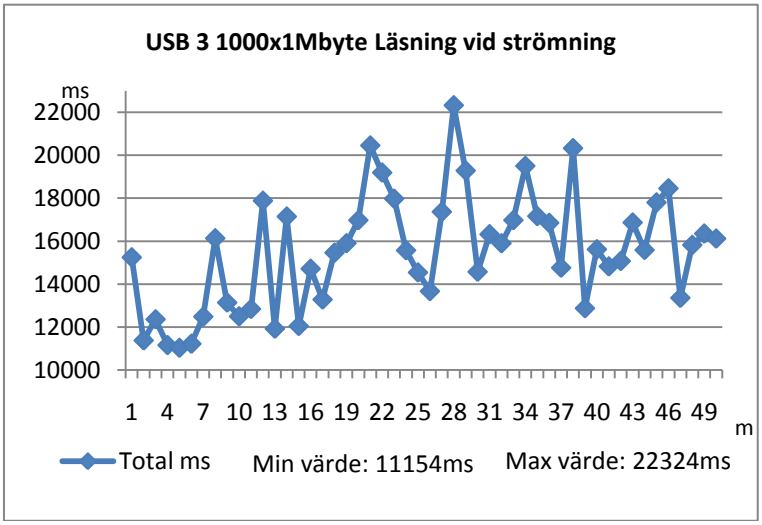

USB 2 1000x1Byte Läsning vid strömning
12000
ms
10000
8000
6000
4000
1 4 7 10 13 16 19 22 25 28 31 34 37 40 43 46 49
m
Total ms
Min värde: 4633ms
Max värde: 12355ms


USB 3 1000x1Byte Läsning vid strömning
2900
ms
2400
1900
1400
900
1 4 7 10 13 16 19 22 25 28 31 34 37 40 43 46 49
m
Total ms
Min värde: 967ms
Max värde: 2683ms


USB 2 1000x1Kbyte Läsning vid strömning
ms
10000
8000
6000
4000
1 4 7 10 13 16 19 22 25 28 31 34 37 40 43 46 49
m
Total ms
Min värde: 4524ms
Max värde: 11420ms


USB 3 1000x1Kbyte Läsning vid Strömning
ms
3000
2500
2000
1500
1000
1 4 7 10 13 16 19 22 25 28 31 34 37 40 43 46 49
m
Total ms
Min värde: 1092ms
Max värde: 3011ms


USB 2 1000x1Mbyte Läsning vid strömning
ms
80000
70000
60000
50000
40000
1 4 7 10 13 16 19 22 25 28 31 34 37 40 43 46 49
m
Total ms
Min värde: 42479ms
Max värde: 80762ms


USB 3 1000x1Mbyte Läsning vid strömning
ms
22000
20000
18000
16000
14000
12000
10000
1 4 7 10 13 16 19 22 25 28 31 34 37 40 43 46 49
m
Total ms
Min värde: 11154ms
Max värde: 22324ms