

CAM for avanserte verktøymaskiner

Bjarte Halvorsen

Produktutvikling og produksjon
Innlevert: juni 2015
Hovedveileder: Knut Sørby, IPK

Norges teknisk-naturvitenskapelige universitet
Institutt for produksjons- og kvalitetsteknikk

Forord

Dette er en masteroppgave i produksjonssystemer, som er en del av en mastergrad i produktutvikling og produksjon ved Norges teknisk-naturvitenskapelige universitet, Trondheim.

Oppgaven er skrevet av student Bjarte Halvorsen for Institutt for produksjons- og kvalitets-teknikk, NTNU. Problembeskrivelsen er utviklet av Professor Knut Sørby ved IPK, NTNU.

Jeg vil gjerne takke min veileder Professor Knut Sørby for god støtte i prosjektet. Jeg vil også takke Arild Sæther og Bjørn Martin Bendixen for støtten jeg fikk i laboratoriet, og for lisensen til GibbsCam som jeg fikk låne.

Abstract

Advanced tool machines are characterized by their functionality to orientate, in addition to position, the tool relative to the workpiece. The functionality to alter the relative orientation of the tool while machining gives opportunities to machine components with complex geometries without having to re-orientate the workpiece multiple times in the same machine, or between several different machines.

The increased functionality makes it complicated to program the machines manually in the same manner as with conventional tool machines. Given this, the programming often has to be done in a computer program which is specially developed for the task (CAM-systems). The program is then transferred to the tool machine via a translator, called a post processor, which describes how the tool machine has to behave for the component to end up as constructed.

In this master thesis, the functionality for programming machining operations in CAM for two different types of advanced tool machines was researched. One 5 axis milling machine and one mill-turn machine. Several different components were constructed, and machining operations programmed in two different CAM-systems. The functionality for programming machining operations for advanced tool machines was found to be good for both of the tested CAM-systems.

In the thesis, existing post processors for one of the CAM-systems (GibbsCam) was tested, and post processors for the other CAM-system (NX Manufacture) was developed and tested. The existing post processors were found to be lacking in some areas, which was documented. Solutions for the lacking functionality in the post processors were also developed. The post processors that were developed in this thesis were thoroughly tested, and no defects were found that can be linked to any of the post processors.

Real machining tests were executed in both machines with good results.

To secure good knowledge transfer from the project, a demonstration in GibbsCam for programming of turning, 3-, 4- and 5-axis machining operations were held for the employees in the tool machine laboratory in the end of the project.

Sammendrag

Avanserte verktøymaskiner er kjennetegnet ved funksjonaliteten maskinene har for å orientere, i tillegg til å posisjonere, verktøyet relativt til arbeidsstykket. Funksjonaliteten for å endre den relative orienteringen av verktøyet under maskinering gir muligheter for å maskinere komponenter med komplekse geometrier uten å måtte flytte arbeidsstykket flere ganger i samme maskin, eller mellom flere maskiner.

Den økte funksjonaliteten gjør det komplisert å programmere maskinene manuelt på samme måte som konvensjonelle verktøymaskiner. Programmeringen må derfor ofte gjøres i et data-program som er utviklet for formålet (CAM-systemer), og overføres til verktøymaskinen via en oversetter, kalt postprocessor, som beskriver hvordan verktøymaskinen skal oppføre seg for at komponenten skal ende opp slik som den ble konstruert.

I denne masteroppgaven ble funksjonaliteten for programmering av maskineringsoperasjoner i CAM for to forskjellige typer avanserte verktøymaskiner utforsket. En fresemaskin med 5 akser og en flerfunksjonsmaskin som kan både dreie og frese. Flere forskjellige komponenter ble konstruert, og maskineringen av komponentene ble programmert i to forskjellige CAM-systemer. Funksjonaliteten for å programmere maskineringsoperasjoner for avanserte verktøymaskiner ble funnet til å være god for begge de utprøvde CAM-systemene.

I oppgaven ble også eksisterende postprocessorer til det ene CAM-systemet (GibbsCam) testet, og postprocessorer til det andre CAM-systemet (NX Manufacture) utviklet og testet. For de eksisterende postprocessorene ble det avdekket flere mangler som ble dokumentert. Det ble også funnet fungerende løsninger for manglene. Postprocessorene som ble utviklet i oppgaven ble grundig testet og det ble ikke avdekket feil som kan tilskrives noen av de utviklede postprocessorene.

Til slutt ble en prosedyre og to forskjellige programmer utviklet for å sette riktig nullpunkt for 5-aksemaskinering, og fysiske maskineringstester ble utført i begge maskinene med gode resultater.

For å sikre god kunnskapsoverføring ble det holdt en demonstrasjon i GibbsCam for programmering av operasjoner for dreining, 3-, 4- og 5-aksemaskinering for de ansatte i verktøymaskinlaboratoriet i slutten av prosjektet.

Innhold

Forord	i
Abstract	ii
Sammendrag	iii
1 Innledning	1
1.1 Bakgrunn	1
1.2 Problemstilling	1
1.3 Rapportens oppbygning	2
2 Teori	3
2.1 CAM	3
2.2 Postprosessorer	9
2.3 STEP-NC	24
2.4 Teststandarder for testkomponenter	25
3 Utvikling av postprosessorer for NX	26
3.1 Postprosessor for Deckel Maho 5-akse fresemaskin	26
3.2 Postprosessor for Okuma Genos flerfunksjonsmaskin	28
4 Testmodeller	32
4.1 5-aksefresing	32
4.2 Dreining og 4-aksefresing	34
5 Planlegging av maskineringsoperasjoner i CAM	37
5.1 5-aksemaskinering	37
5.2 Dreining og 4-aksefresing	45
5.3 Oppsummering	70
6 Prosedyre for definering av nullpunkt for 5-aksemaskinering i Deckel Maho DMU 50eVolution	71
6.1 Prosedyre for manuell definering av nullpunkt	71
6.2 Program for automatisk nullpunktsforskyvning	72
6.3 Hjelpeprogram for definering av nullpunkt	73
7 Maskineringstester	75
7.1 5-aksemaskinering av testmodell 2 i Deckel Maho	75
7.2 Maskinering av testmodell 4 i Okuma Genos	78
8 Konklusjon og videre arbeid	80

8.1	Konklusjon	80
8.2	Videre arbeid	81
Tillegg A Liste over elektroniske vedlegg		84
Tillegg B Målebrev for Deckel Maho DMU 50eVolution		85
Tillegg C Programmerte kommandoer		91
C.1	Kommando for kjøring til sikker posisjon - Deckel Maho	92
C.2	Kommando for verktøybytte - Okuma	93
C.3	Kommando for aktivering av auto-låsing av C-aksen - Okuma	94
C.4	Kommando for beskrivelse av rotasjonsretning C-akse - Okuma	95
C.5	Kommando for aktivering av fysisk y-akse - Okuma	97
C.6	Kommando for deaktivering av fysisk Y-akse - Okuma	98
Tillegg D Maskintegninger av testmodeller		99
Tillegg E Maskinkode		106
E.1	Maskinkode generert av postprocessor til NX Manufacture	106
E.2	Uredigert maskinkode generert av postprocessor til GibbsCam	109
E.3	Redigert maskinkode generert av postprocessor til GibbsCam	112

Figurer

2.1	Fremgangsmåte fra importert CAD-modell til maskinkode i GibbsCam	6
2.2	Fremgangsmåte fra CAD-modell til maskinkode i NX	8
2.3	Verktøybane med og uten linearisering. [1]	11
2.4	CL-data definerer verktøybanen [2]	12
2.5	B- og C-aksenes posisjon for en verktøybevegelse gjennom singulariteten [2]. . .	12
2.6	Algoritme for postprosessering nær singulariteten, forbedret fremgangsmåte. [2]	14
2.7	Effekten av den forbedrede algoritmen. [2]	14
2.8	Rotasjon av koordinatsystemet $O-xyz$ med vinkelen α om z-aksen [3]	16
2.9	Skisse av en fem-akse maskin med uorthogonale rotasjonsakser sett fra siden. Vist med B-akse i posisjonen $B = 0^\circ$ [2].	18
2.10	Input og output i NX Post [4].	21
2.11	Steg for postprosessering med NX Post [4].	22
2.12	Testkomponent for simultan 5-aksemaskinering [5].	25
3.1	Deckel Maho DMU 50eVolution i verktøymaskinlaboratoriet ved IPK	26
3.2	Okuma Genos L200E-M i verktøymaskinlaboratoriet ved IPK	28
3.3	Skisse av arbeidsstykke og maskineringsplan i OKUMA GENOS L200 E-M . . .	30
3.4	Orientering av de fysiske, linære aksene i OKUMA GENOS L200 E-M	31
4.1	Testmodell 1 - Deckel Maho	32
4.2	Testmodell 2 - Deckel Maho	33
4.3	Testmodell 1 - Okuma	34
4.4	Testmodell 2 - Okuma	35
4.5	Testmodell 3 - Okuma	35
4.6	Testmodell 4 - Okuma	36
5.1	Verktøybaner og simulering for test 1 i GibbsCam.	38
5.2	Verktøybaner og simulering for operasjon 1 i test 2 i GibbsCam.	40
5.3	Verktøybaner og simuleringer for operasjon 2 i test 2 i GibbsCam.	41
5.4	Verktøybaner og simulering for test 1 i NX Manufacture.	42
5.5	Verktøybaner for test 2 i NX Manufacture.	43
5.6	Simulering av maskinering for test 2 i NX Manufacture.	44
5.7	Verktøybaner for testmodell 1 i GibbsCam.	46
5.8	Simulering for testmodell 1 i GibbsCam.	47
5.9	Hjem-posisjon i MDD-fil	49
5.10	Utsnitt av maskinkode for dreining med (t.v.) og uten (t.h.) radiuskompensering.	49
5.11	Verktøybaner for test 2 i GibbsCam.	51

5.12	Simulering for test 2 i GibbsCam.	52
5.13	Utsnitt av maskinkode for fresing med (t.v.) og uten (t.h.) radiuskompensering.	54
5.14	Verktøybaner for test 3 i GibbsCam.	56
5.15	Simulering for test 3 i GibbsCam.	57
5.16	Verktøybaner for test 4 i GibbsCam.	59
5.17	Simulering for test 4 i GibbsCam.	60
5.18	Verktøybaner for test 1 i NX Manufacture	61
5.19	Verktøybaner og simulering for test 1 i NX Manufacture.	62
5.20	Verktøybaner og simulering for test 2 i NX Manufacture.	64
5.21	Verktøybaner og simulering for test 3 i NX Manufacture.	66
5.22	Verktøybaner for test 4 i NX Manufacture.	68
5.23	Simuleringer for test 4 i NX Manufacture.	69
6.1	Program for automatisk nullpunktsforskyvning i Deckel Maho	73
6.2	Hjelpeprogram for posisjonering av nullpunkt for 5-aksemaskinering	74
7.1	Maskinering av testmodell 2 i Deckel Maho DMU 50eVolution	76
7.2	Sluttprodukt fra maskinering av testmodell 2 i Deckel Maho DMU 50eVolution	77
7.3	Maskinering av testmodell 4 i Okuma Genos L200E-M	79

Tabeller

3.1	Postprosessorer for Okuma Genos L200E-M	28
5.1	Mangler funnet ved postprosessor til GibbsCam for Okuma Genos etter test 1 .	47
5.2	Mangler funnet ved postprosessor til GibbsCam for Okuma Genos etter test 2 .	53
5.3	Oversikt over mangler med tilhørende løsninger for postprosessor til GibbsCam for Okuma Genos L200E-M	70

Kapittel 1

Innledning

1.1 Bakgrunn

Verktøymaskinlaboratoriet ved IPK, NTNU, produserer komponenter for bruk i forskning, undervisning, studentprosjekter og til andre formål innenfor universitetet. I tillegg produseres det komponenter til forskningsprosjekter i SINTEF Raufoss Manufacturing AS, som har kontorer og laboratorium i det samme bygget.

Laboratoriet er utstyrt med maskiner for additiv tilvirkning, koordinatmålemaskin flere industriroboter og forskjellige typer verktøymaskiner. Av CAM-systemer brukes GibbsCam av de ansatte i laboratoriet til 3-aksemaskinering, mens NX Manufacture har blitt brukt av doktorgradsstipendiater og professorer ved NTNU til 5-aksemaskinering.

Av de avanserte verktøymaskinene finnes det postprosessorer til GibbsCam for en Deckel Maho DMU 50eVolution fresemaskin med 5 akser og en Okuma Genos L200 E-M flerfunksjonsmaskin. Postprosessen til 5-aksefresen har blitt testet på et tidligere tidspunkt av en annen student i en masteroppgave, men postprosessen har ikke blitt brukt av de ansatte i verkstedet og det er usikkert om den fungerer som den skal. Postprosessen til flerfunksjonsmaskinen har aldri blitt testet eller brukt. Denne postprosessen bærer navnet til en annen maskin fra samme produsent og det er usikkert om, og eventuelt hvor godt, postprosessen fungerer.

Til 5-aksemaskinering har hovedsakelig en frittstående postprosessor for 5-aksefresen blitt brukt. Postprosessen er utviklet av Professor Knut Sørby ved IPK, NTNU og genererer NC-data fra CL-data som er eksportert fra et CAD-system. Det finnes ingen postprosessorer for NX Manufacturing til noen av maskinene.

1.2 Problemstilling

Oppgaven går ut på å studere funksjonaliteten til CAM-systemer når det gjelder programmering av avanserte verktøymaskiner. Det skal fokuseres på CAM-systemene GibbsCAM og NX Manufacture, og det er spesielt funksjonaliteten for programmering av 5-aksemaskiner og flerfunksjonsmaskiner (dreining/fresing) som skal studeres. Det skal utvikles nødvendige postprosessorer for de aktuelle maskinene og det skal gjøres maskineringstester for å verifisere funksjonaliteten til postprosessorne. Det kan være nødvendig å utvikle prosedyrer for å kontrollere at maskinens nullpunkter settes riktig.

1.3 Rapportens oppbygning

Rapporten begynner med et innledende kapittel som presenterer relevant teori for oppgaven. Deretter beskrives utviklingen av postprosessorene til NX Manufacture for maskinene, og utviklingen av testmodellene som brukes i oppgaven. Arbeidet som ble gjort for planleggingen av maskineringsoperasjonene i CAM-systemene beskrives og den postprosesserte maskinkoden analyseres. Prosedyrer og programmer som ble utviklet for å sette nullpunkt ved 5-aksemaskinering gjennomgås, og resultatene fra de fysiske maskineringstestene presenteres. Til slutt oppsummeres oppgaven i en konklusjon.

Kapittel 2

Teori

2.1 CAM

CAM (computer-aided manufacturing) er en betegnelse på programvare som brukes til å generere verktøybaner for produksjon av komponenter i CNC (computer numerical control)-maskiner. Programvaren har grafisk brukergrensesnitt og ofte simuleringsverktøy som kan vise de genererte verktøybanene relativt til arbeidsstykket og simulere verktøybevegelsene og avirkningen fra råemne til ferdig maskinert komponent. I noen CAM-programmer kan også en modell av verktøymaskinen konstrueres og settes opp med den samme kinematikken som verktøymaskinen som skal utføre maskineringen. Da kan også maskinens bevegelser under maskineringen simuleres. Siemens NX manufacture er et eksempel på CAM-programvare som har støtte for denne typen maskinsimulering. Simuleringsverktøyene er gode hjelpemidler for å avdekke feil i verktøybanene eller avdekke fare for krasj i verktøymaskinen. CAM-programvaren kan være frittstående eller som en integrert del av CAD (computer aided design)-programvare. Selv om CAM blir mye brukt til generering av verktøybaner for fresing og dreining, finnes det også CAM-programvare som har støtte for andre typer bearbeidingsoppgaver, som for eksempel trådnisting.

Det er flere fordeler ved å bruke CAM-programvare i produksjon. Den automatiske genereringen av verktøybaner i CAM-programvaren kan frigjøre mye tid som tidligere har blitt brukt til å programmere maskinkode manuelt på maskinstyringen, eller i en tekstfil som deretter overføres til maskinstyringen. Dette gir økt produktivitet i tillegg til at feilprogrammering kan reduseres og gi mindre etterarbeid og feilproduksjon. I tilknytning til avanserte verktøymaskiner med flere enn tre akser er CAM-programvare et spesielt kraftig verktøy. Med denne kombinasjonen kan avanserte geometrier som det har vært vanskelig eller umulig å programmere manuelt, som for eksempel turbinskovler, maskineres. Muligheten til å endre orienteringen til verktøyet og/eller arbeidsstykket i verktøymaskiner med flere enn tre akser kan også gi bedre nøyaktighet og produktivitet i tilfeller der det tidligere har vært nødvendig med flere oppspenninger, eller der valg av verktøy eller skjæreforhold ikke har vært optimalt grunnet manglende frihet til å endre orientering mellom verktøyakse og arbeidsstykke under maskineringen.

Verktøybanene som CAM-programvaren genererer baseres på valg som brukeren gjør i programvaren. Drivende geometri, kuttemønster og hvilke faktorer som skal styre orienteringen til verktøyaksen er valg som har stor betydning for utformingen av verktøybanene. I tillegg til generering av verktøybaner kan CAM-programvaren også legge inn andre funksjoner, som spindelhastighet, mating, automatisk verktøybytte og bruk av kjølevæske. En av ulempene

med CAM-programvare til generering av verktøybaner er at brukeren har begrenset mulighet til å endre verktøybanene foruten de valgene som kan gjøres i programvaren. Hvor stor påvirkningskraft brukeren har på utformingen av verktøybanene vil derfor avhenge av den CAM-programvaren som brukes.

Når verktøybanene er definert i CAM-programvaren og parametre som spindelhastighet og mating er lagt inn kan verktøybanene genereres og lagres som CL (cutter location)-data. I CL-data beskrives verktøybanene med posisjoner og orienteringer av verktøytuppens senterpunkt relativt til koordinatsystemet som ble definert som maskinens koordinatsystem, ofte kalt nullpunkt, i programvaren. Formatet på koordinatene er $[x\ y\ z\ i\ j\ k]$, hvor $x\ y\ z$ er koordinatene til verktøytuppens senterpunkt, og $i\ j\ k$ er verktøyets orienteringsvektor [6]. Siden verktøybanene beskrives relativt til maskinens nullpunkt, og ikke tar hensyn til konfigureringen av maskinaksene til verktøymaskinen, er CL-data uavhengig av hvilken type verktøymaskin som brukes til bearbeidingen.

I verktøymaskiner brukes maskinaksenes posisjoner, eller vinkler, til å beskrive verktøyets orientering, i motsetning til i CL-data hvor verktøyets vektor beskriver dets orientering. CL-data må derfor postprosesserer til maskinlesbar kode, NC (numerical control)-data, som beskriver hvilke posisjoner maskinaksene må ha relativt til maskinens nullpunkt for å oppnå ønsket verktøyposisjon og verktøyorientering [7]. CAM-programvare har ofte støtte for å hente inn postprosessorer som er utviklet for å fungere sammen med programvaren. Da kan verktøybanene og annen relevant informasjon postprosesserer direkte til maskinkode uten å måtte eksporteres som CL-data først for deretter å importeres i en frittstående postprosessor. En av ulempene med CAM-programvare og postprosessorer som er skreddersydd for hverandre er at det blir mer kostbart å bytte CAM-programvare, da postprosessorer til alle maskinene må byttes samtidig.

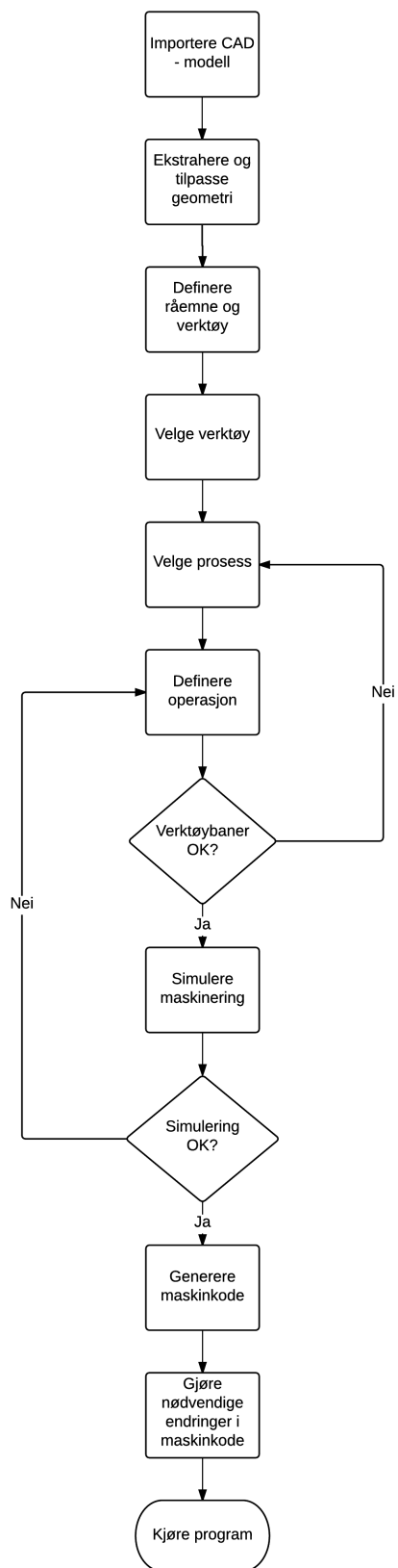
2.1.1 GibbsCam

GibbsCam er et frittstående CAM-system som tilsynelatende er populært i industrien. Programvaren har støtte for dreining, 3-,4- og 5-aksefresing i tillegg til mer uvanlige bearbeidingsmetoder, som trådnisting. Hvilke metoder og funksjoner som kan brukes i GibbsCam bestemmes av lisenspakker. For postprosessering er det integrert en egen postprosessor modul i programmet. I modulen velges postprosessorer for den aktuelle maskinen og hvor utfilen skal lagres. For vanlige brukere må postprosessorer som skal brukes i GibbsCam kjøpes fra en leverandør som utvikler postprosessorer til denne programvaren. Det finnes utviklerverktøy for å lage postprosessorer til GibbsCam, men denne programvaren er tilsynelatende ikke tilgjengelig for vanlige brukere. Uten dette utviklerverktøyet har brukeren ikke mulighet til å gjøre endringer i postprosessorer.

I tillegg til postprosessorer bruker GibbsCam en MDD-fil (Machine Definition Document) når maskinkoden genereres. MDD-filen inneholder en fysisk beskrivelse av verktøymaskinen og i motsetning til postprosessorer har brukeren mulighet til å endre MDD-filen med en MDD-editor som følger med GibbsCam-installasjonen. Editoren ligger ikke i startmenyen og den er vanskelig å finne dersom man ikke vet hvor den ligger i installasjonen. Stien for en vanlig installasjon oppgis derfor her som referanse for fremtidig bruk: `C:\Program Files\Gibbs\GibbsCAM\10.7.19.0-en-US\Bin`.

I GibbsCam finnes det verktøy for å bygge opp to- og tredimensjonal geometri, men sammenlignet med fullverdig CAD-programvare, som Siemens NX og SolidWorks, er verktøyene i GibbsCam enkle. Programmet har funksjonalitet for importering av CAD-modeller i forskjellige filtyper. Hvilke filtyper som støttes er bestemt av lisenspakken som brukes. I dette prosjektet har CAD-modeller blitt konstruert i NX og importert i filformatet IGES (Initial Graphics Exchange Specification) til GibbsCam. I importerte modeller i IGES-format kan linjer og punkt ekstraheres fra modellen og benyttes til planlegging av maskineringsoperasjoner. Den ekstraherte geometrien kan også tilpasses i GibbsCam.

I dette prosjektet har den grunnleggende fremgangsmåten som er vist i Figur 2.1 blitt brukt til å planlegge maskineringsoperasjoner fra importerte CAD-modeller i GibbsCam. Etter importering av CAD-modellen ekstraheres og modifiseres geometrien dersom det er nødvendig. Ofte er det mulig å bruke flatene på modellen til å generere verktøybaner, og det er da unødvendig å ekstrahere geometri i form av linjer fra modellen. Dette gjelder spesielt for freseoperasjoner. For dreieoperasjoner er det mer vanlig at geometri må ekstraheres fra modellen og tilpasses. Videre defineres råemnets størrelse og utforming, nullpunktet og verktøyene som skal brukes. Maskineringsoperasjonene bygges opp av en maskineringsprosess og et verktøy, hvor maskineringsprosessen for eksempel kan være grovbearbeiding ("Roughing") eller konturmaskinering ("Contour"). I vinduet for operasjonen velges metode og mønster for maskineringen, maskinparametre som spindelhastighet, kuttdybde og mating i tillegg til flere andre valg, som bevegelse inn og ut av kutt og klaringsdistanser. Til slutt simuleres maskineringen for å kontrollere at verktøybevegelser og avvirkning ser korrekt ut, før operasjonene postprosesseres til maskinkode.



Figur 2.1: Fremgangsmåte fra importert CAD-modell til maskinkode i GibbsCam

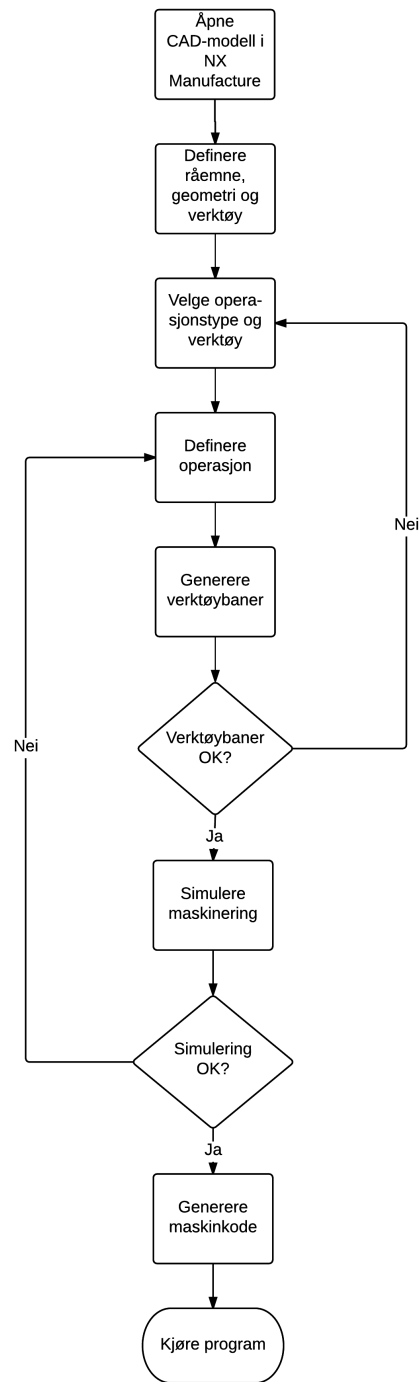
2.1.2 NX Manufacture

NX Manufacture er en CAM-modul i Siemens NX, som er et integrert system for produktdesign (CAD), engineering (CAE) og produksjon (CAM). I NX Manufacture kreves det kun noen få museklikk å forflytte seg mellom CAD og CAM, og endringer i CAD-modellen oppdateres automatisk i CAM-modellen. Den korte veien mellom CAM og fullverdig CAD i et integrert system gjør det lettere og raskere å optimalisere konstruksjoner med hensyn til hvordan de skal produseres.

I motsetning til GibbsCam er postprosessorer for NX Manufacture basert på åpne script, og utviklerverketøyet for å bygge eller endre postprosessorer for NX Manufacture følger med i installasjonen. Postprosessorer for bruk med NX Manufacture kan også kjøpes av profesjonelle utviklere. En postprosessormodul i NX Manufacture kjører den valgte postprosessoren og genererer maskinkode direkte fra programmet. I tillegg til utfilen, som inneholder den genererte maskinkoden, vises maskinkoden i et dialogvindu i NX Manufacture. Dette gjør det raskt og enkelt å kontrollere maskinkoden for feil før programmet overføres til maskinstyringen.

Flytskjemaet i Figur 2.2 viser fremgangsmåten som har blitt brukt til å planlegge maskineringsoperasjoner i NX Manufacture i denne oppgaven. Når konstruksjonene åpnes i NX Manufacture defineres råemnet, geometri og verktøy først. Geometrien kan være for eksempel være klaringsplan eller koordinatsystemer som skal brukes. Deretter planlegges maskineringsoperasjonene som skal gjøres ved å velge operasjonstype. Dette kan for eksempel være lommefresing med operasjonen "Floor and Wall". Valg av operasjonstype og metode er avgjørende for hvordan verktøybanene blir utformet og hvilke valg brukeren kan eller må gjøre for å definere hvordan maskineringen skal utføres. Deretter velges overflatene som skal maskineres, begrensninger, maskinparametre og en rekke andre faktorer. Til slutt kan verktøybanene genereres og simuleres med eller uten materialavvirkning. Maskineringen kan også simuleres med arbeidsstykket oppspent i en verktøymaskin. Maskinsimuleringen viser hvordan maskinens akser beveger seg under maskineringen og er et godt verktøy for å kontrollere at for eksempel oppspenningsverktøy ikke kommer til å kolliderer med maskin eller verktøy under maskineringen. I biblioteket i NX Manufacture finnes det noen ferdige eksempelmodeller for maskiner, i tillegg til at brukeren selv konstruere og bygge opp en modell av sin egen verktøymaskin.

Et annet, nyttig verktøy i NX Manufacture kalles Used Defined Events (UDE's). UDE's er funksjonalitet som er programmert inn i postprosessoren og som kan kalles opp fra NX Manufacture. Dette kan for eksempel være en kommando om å låse rotasjonsaksen før en freseoperasjon i en flerfunksjonsmaskin, eller å definere hvilken verktøyspindel som skal brukes, slik at den rette postprosessoren for oppgaven hentes inn. UDEene er tilgjengelige ved å høyreklikke på en operasjon, eller en gruppe med operasjoner, og velge "start events".



Figur 2.2: Fremgangsmåte fra CAD-modell til maskinkode i NX

2.2 Postprosessorer

Postprosessen utgjør leddet mellom CAM og verktøymaskin og er stedet hvor maskinspesifikk NC-data genereres fra nøytral CL-data. For at CNC-maskiner skal kunne utføre bevegelsene i et program må beskrivelsen av posisjonen og orienteringen til verktøyet være likt i CAM og i maskinstyringen. I enkle maskineringsoppgaver, som 3-aksefresing, er det vanlig at maskinoperatøren definerer maskinens koordinatsystem, ofte kalt nullpunkt, på det samme stedet som nullpunktet ble definert i CAM, relativt til arbeidsstykket. En vanlig metode for å definere nullpunkt i verktøymaskiner er ved hjelp av en måleprobe som monteres i spindelen. Når proben føres mot råemnet registreres aksenes nøyaktige posisjoner og et nullpunkt for maskinens koordinatsystem kan legges inn i maskinstyringen. I slike tilfeller har det ingen betydning for resultatet hvor maskinens koordinatsystem er posisjonert i CAM-programvaren, så lenge nullpunktet i maskinen etableres i den samme posisjonen relativt til arbeidsstykket. Av praktiske årsaker er det vanlig å plassere maskinens koordinatsystem på et hjørnet eller i senter av arbeidsstykket i CAM-programvaren, slik at det er lett å måle opp og sette som nullpunkt i verktøymaskinen.

I avanserte verktøymaskiner kan ikke alltid maskinens koordinatsystem, ofte kalt nullpunkt, velges på samme måte som ved 3-aksemaskinering. Grunnet rotasjonsaksene i slike maskiner må et bestemt nullpunkt som er definert av maskinens kinematikk, og som ble brukt i utviklingen av postprosessen, benyttes når maskinens koordinatsystem defineres i CAM-systemet. Arbeidsstykket spennes da opp i verktøymaskinen først, og dets posisjon og orientering relativt til nullpunktet i maskinen finnes ved hjelp av en måleprobe som er montert i spindelen. De målte verdiene brukes til å posisjonere og orientere arbeidsstykket relativt til nullpunktet i CAM-programvaren slik at det samsvarer med de virkelige forholdene i verktøymaskinen før maskinkoden genereres i postprosessen.

For at postprosessen skal kunne generere maskinkode hvor verktøyets posisjon og orientering er beskrevet med maskinaksenes posisjoner og vinkler, relativt til maskinens nullpunkt, må en modell for invers kinematikk for verktøymaskinen etableres. Løsningen for den inverse kinematikken kan finnes analytisk eller ved numeriske beregninger. Numerisk løsning baseres på en stor mengde numeriske operasjoner. Analytisk løsning er derfor raskere, og bedre egnet dersom postprosesseringen utføres i sanntid på maskinstyringen [2]. I praksis utføres postprosesseringen ofte i sammenheng med genereringen av verktøybanene i CAM-programvaren, men i fremtiden kan postprosessering på maskinstyringen bli mer vanlig. Siden verktøymaskiner med forskjellig geometrisk oppsett vil kreve forskjellig posisjonering av aksene for å oppnå samme verktøyposisjon og verktøyorientering relativt til arbeidsstykket, er kinematikken og postprosessen maskinspesifikk. Tilleggsfunksjonalitet og rutiner som det enkelte verksted har nytte av kan også programmeres inn i postprosessen, som for eksempel å kontrollere at ingen posisjoner i programmet ligger utenfor aksenes begrensninger, eller rutiner for å returnere alle aksene til en gitt posisjon i slutten av programmet. Postprosessen er derfor ofte noe som blir tilpasset både til den enkelte maskin og det enkelte verksted.

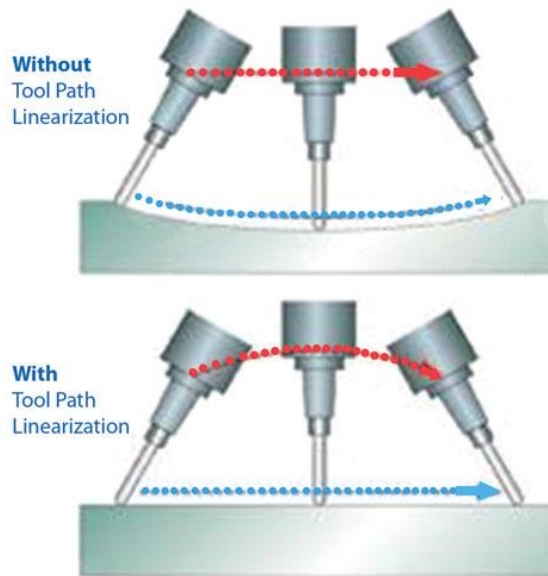
Programmeringsspråket som brukes når postprosessen genererer maskinkode kalles numerical control (NC) programming language, men kalles ofte for G-kode eller bare maskinkode. Programmer som er skrevet i NC-kode leses fra venstre til høyre og linje for linje av maskinstyringen. Hver linje kalles en blokk og inneholder instruksjoner som forteller maskinen hva den skal gjøre og hvordan den skal gjøre det. Alle instruksjonene på hver linje må utføres før

programmet kan fortsette til neste linje. Vanlige adresser som brukes i maskinkode er G og M. Adressen G forteller hvilken type handling som skal utføres. Når adressen kombineres med en tallkode dannes et ord som beskriver en bestemt handling. Det kan for eksempel være ilmatning, sirkelinterpolasjon eller radiuskompensering. Adressen M danner også ord sammen med en tallkode og brukes til andre typer handlinger, som for eksempel start av spindel eller slå på kjølevæske.

I standarden ISO 6983 *Automation systems and integration – Numerical control of machines – Program format and definitions of address words* [8] spesifiseres krav og anbefalinger for et dataformat for posisjonering, bevegelse langs linjer og styresystemer for bevegelse langs konturer i den numeriske styringen av maskiner. Selv om denne standarden eksisterer velger mange leverandører å bruke sin egen dialekt i NC-koden og egne filendinger til programmene. Dette kan gi leverandørene muligheter til å implementere nye funksjoner i maskinstyringen, men det krever også at postprosessorene til maskinene tilpasses til maskinkode-dialekten som maskinstyringen bruker.

2.2.1 Linearisering

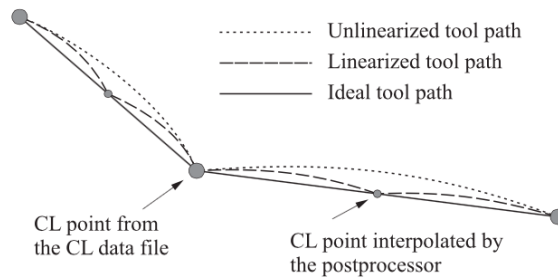
En utfordring ved generering av maskinkode for verktøymaskiner med roterende akser er det ulinære forholdet mellom maskinens koordinatsystem i CAM-programvaren for posisjonering og orientering av verktøyet, og koordinatsystemet i maskinen som brukes for utførelse av bevegelsene. Problemet oppstår når en rettlinjet bevegelse i CAM-programvaren utføres av roterende akser i verktøymaskinen. Selv om verktøyets posisjon og orientering i starten og slutten av bevegelsen er den samme relativt til arbeidsstykket i CAM-programvaren og i verktøymaskinen vil den faktiske banen som verktøyet har tilbakelagt være forskjellig. Grunnet de roterende aksene i verktøymaskinen vil verktøyet følge en buet bane fra punkt A til punkt B, som vist i Figur 2.3, mens bevegelsen fra de samme punktene i CAM-systemet vil være rettlinjet. Avviket fra den genererte verktøybanen i CAM-programvaren og den virkelige verktøybanen kan føre til geometriske forskjeller mellom CAD-modellen og den maskinerte componenten. Avviket vil bli større ved økende avstand mellom punktene. For at den faktiske verktøybanen skal følge den planlagte banen i CAM-programvaren må endringer gjøres i genereringen av maskinkoden, og til det brukes lineariseringsalgoritmer i postprosessoren.



Figur 2.3: Verktøybane med og uten linearisering. [1]

For å ta hånd om avvik mellom den planlagte verktøybanen i CL-data og den faktiske banen i NC-data brukes invers kinematikk og direkte kinematikk i lineariseringsalgoritmen i postprosessoren. I det første trinnet brukes invers kinematikk til å finne hvilke posisjoner maskinaksene må ha i de forskjellige delene av programmet som ble generert i CAM-programvaren. Deretter brukes direkte kinematikk til å sammenligne den genererte banen i NC-data med den planlagte banen i CL-data. Dersom avviket mellom banen i NC-data og CL-data er større enn den bestemte toleransen interpoleres nye datapunkt langs den optimale banen i CL-data og legges til i verktøybanen i NC-data, som vist i Figur 2.4. Dersom avviket fremdeles er for stort interpoleres det nye punkter helt til avviket er innenfor den bestemte toleransen. Ved å benytte linearisering i postprosessoren vil det alltid være avvik mellom den planlagt banen i CL-data og den faktiske banen i NC-data, men avviket kan styres av størrelsen på toleransen

som brukes i postprosessen. Grunnet de ekstra punktene som legges til i lineariseringen vil NC-data ofte bestå av flere linjer enn CL-data.

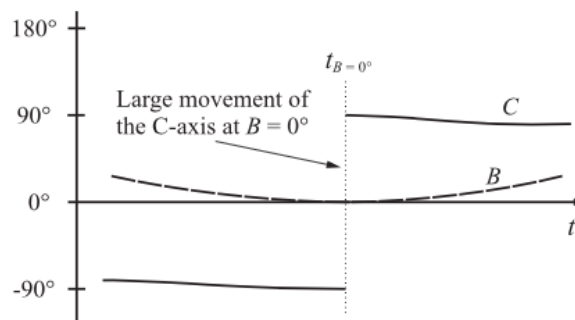


Figur 2.4: CL-data definerer verktøybanen [2]

2.2.2 Singulariteter

En annen utfordring ved kinematikk for avanserte verktøymaskiner med rotasjonsakser er singulariteter. En singularitet oppstår når to eller flere rotasjonsakser i verktøymaskinen er parallelle. I verktøymaskiner inntreffer vanligvis en singularitet når verktøyaksen er parallel med en av rotasjonsaksene. Enhver orientering av rotasjonsaksen vil da resultere i den samme verktøyorienteringen relativt til arbeidsstykket [2]. I nærheten av en singularitet kan lave hastigheter i programmet gi høye og uforutsigbare hastigheter i maskinen. I Deckel Maho DMU 50eVolution oppstår en singularitet når B-aksen er 0° . Rotasjonsaksen C er da parallell med verktøyaksen.

For å unngå verktøykrasj, geometriske feil på arbeidsstykket og uforutsigbare bevegelser og hastigheter i verktøymaskinen må postprosessen inneholde algoritmer som tar seg av singulariteter når NC-data genereres. For Deckel Maho DMU 50eVolution presenterte Sørby [2] en algoritme for å håndtere singulariteter ved kontinuerlig 5-aksemaskinering av et arbeidsstykke utformet som en halv sylinder. Når verktøyet beveger seg over toppen av sylindere må en singularitet passeres og C-aksen må gjøre en rask rotasjon på 180° for å gi riktig verktøybevegelse, som vist i Figur 2.5.



Figur 2.5: B- og C-aksenes posisjon for en verktøybevegelse gjennom singulariteten [2].

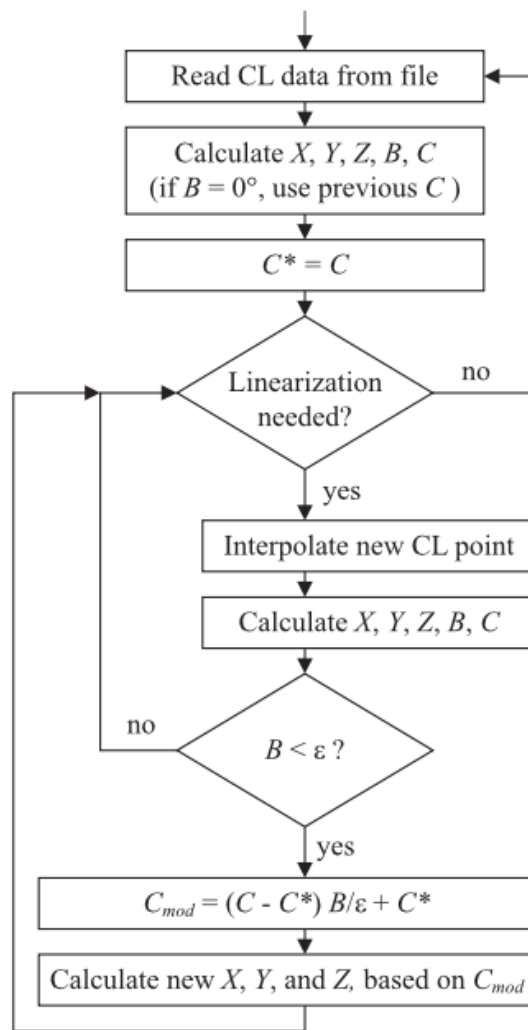
For å håndtere singulariteten utviklet Sørby først en enkel algoritme, vist i Ligning 2.1, som tvinger C-aksen til 0° når B-aksen er mindre enn parameteren ε , som er en liten vinkel typisk

0.01°. Algoritmen fungerte, men førte til unødvendige bevegelser i noen tilfeller. Det ble derfor utviklet en forbedret utgave av algoritmen, vist i Ligning 2.2, som fjernet de unødvendige bevegelsene.

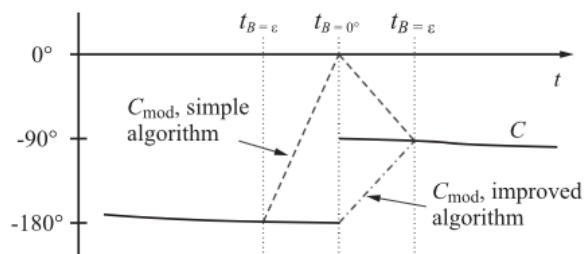
$$\textit{if } B < \varepsilon \textit{ then } C_{mod} = C \frac{B}{\varepsilon} \quad [2]. \quad (2.1)$$

$$\textit{if } B < \varepsilon \textit{ then } C_{mod} = (C - C^*) \frac{B}{\varepsilon} + C^* \quad [2]. \quad (2.2)$$

Variabelen C^* i Ligning 2.2 er C-aksens posisjon som ble kalkulert fra siste CL-data som ble lest før algoritmene for linearitet og singularitet ble brukt. Begge algoritmene løser problemet med singulariteten på bekostning av et lite avvik i verktøyorienteringen. Figur 2.6 viser flyt-diagrammet for den forbedrede algoritmen (Ligning 2.2), og Figur 2.7 viser effekten av den forbedrede algoritmen (Ligning 2.2) i forhold til den enkle algoritmen (Ligning 2.1).



Figur 2.6: Algoritme for postprosessering nær singulariteten, forbedret fremgangsmåte. [2]



Figur 2.7: Effekten av den forbedrede algoritmen. [2]

2.2.3 Kinematikk

Kinematikk kan defineres som *studien av geometrien til bevegelse* [9]. For verktøymaskiner (og roboter) brukes kinematikk til å beskrive sammenhengen mellom maskinaksenes, eller leddenes, posisjoner og verktøyets posisjon og orientering relativt til arbeidsstykket. Maskiner med ulik konfigurasjon av maskinaksene vil ha forskjellige kinematiske ligninger. På generelt grunnlag er kinematikken for verktøymaskiner enklere enn for industriroboter, som ofte har flere rota-sjonsakser og flere frihetsgrader. Samtidig er krav til nøyaktighet ofte større i verktøymaskiner enn for roboter.

Kinematikken kan deles opp i to deler, direkte kinematikk og invers kinematikk. I direkte kinematikk kan verktøyets posisjon og orientering beregnes fra de avleste posisjonsmålingene på maskinaksene. Fra maskinkoden vet maskinstyringen til enhver tid hvilke posisjoner aksene i maskinen skal ha. Den direkte kinematikken er derfor svært anvendelig å bruke i en tilbakekoblingsløkke i maskinstyringen, som kontrollerer om de avleste posisjonene til aksene samsvarer med posisjonene i maskinkoden. Dersom det finnes avvik mellom ønsket og faktisk posisjon legges en korreksjon inn i bevegelsen. I postprosessorer kan direkte kinematikk benyttes til å kontrollere verktøybanens nøyaktighet i lineariseringsalgoritmen i postprosessen [2].

For å beskrive hvilke posisjoner og vinkler maskinaksene må ha for å oppnå ønsket verktøyposisjon og verktøyorientering relativt til arbeidsstykket må invers kinematikk benyttes i postprosessen.

Posisjon

I kinematikken kan et punkt på et stivt element i rommet beskrives fullstending fra dets posisjon og orientering relativt til et referanse-koordinatsystem [3]. Dersom et punkt P , beskrevet i koordinatsystemet $O-xyz$, flyttes langs vektoren \mathbf{d} vil punktets nye posisjon være

$$\mathbf{p}' = \mathbf{p} + \mathbf{d} \quad (2.3)$$

skrevet i i homogene koordinater, hvor

$$\mathbf{p}' = \begin{pmatrix} x' \\ y' \\ z' \\ 1 \end{pmatrix}, \quad \mathbf{p} = \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}, \quad \mathbf{d} = \begin{pmatrix} p \\ q \\ r \\ 0 \end{pmatrix}.$$

Translasjonen fra \mathbf{p} til \mathbf{p}' kan beskrives ved hjelp av transformasjonsmatrisen \mathbf{T} i uttrykket

$$\mathbf{p}' = \mathbf{T}\mathbf{p} \quad (2.4)$$

hvor

$$\mathbf{T} = \begin{pmatrix} 1 & 0 & 0 & p \\ 0 & 1 & 0 & q \\ 0 & 0 & 1 & r \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (2.5)$$

Orientering

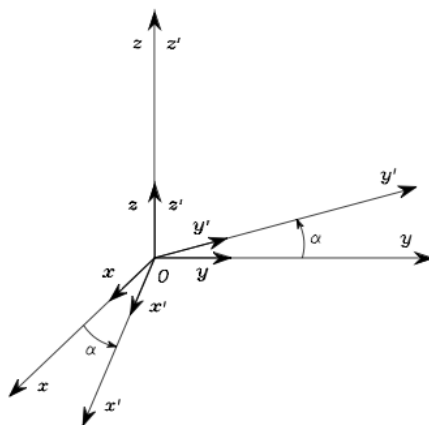
Et stivt elements orientering i rommet kan beskrives som en rotasjon av et kjent referansekoordinatsystem. Elementets enhetsvektorer vil da være beskrevet med sine komponenter i rotasjonsmatrisen

$$\mathbf{R} = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} \quad (2.6)$$

relativt til referansekoordinatsystemet.

Dersom et referansekoordinatsystem $O-xyz$ roteres med vinkel α om z-aksen og at det roterte koordinatsystemet kalles $O'-x'y'z'$, som vist i Figur 2.8, vil de nye enhetsvektorene i $O'-x'y'z'$ være beskrevet i $O-xyz$ som

$$\mathbf{x}' = \begin{pmatrix} \cos\alpha \\ \sin\alpha \\ 0 \end{pmatrix} \quad \mathbf{y}' = \begin{pmatrix} -\sin\alpha \\ \cos\alpha \\ 0 \end{pmatrix} \quad \mathbf{z}' = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}. \quad (2.7)$$



Figur 2.8: Rotasjon av koordinatsystemet $O-xyz$ med vinkelen α om z-aksen [3]

Rotasjonsmatrisen fra $O-xyz$ til $O'-x'y'z'$ blir da

$$\mathbf{R}_z(\alpha) = \begin{pmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{pmatrix}. \quad (2.8)$$

På samme måte gir rotasjon med vinkelen β om y-aksen og rotasjon med vinkelen γ om x-aksen rotasjonsmatrisene

$$\mathbf{R}_y(\gamma) = \begin{pmatrix} \cos\gamma & 0 & \sin\gamma \\ 0 & 1 & 0 \\ -\sin\gamma & 0 & \cos\gamma \end{pmatrix} \quad (2.9)$$

$$\mathbf{R}_x(\beta) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\beta & -\sin\beta \\ 0 & \sin\beta & \cos\beta \end{pmatrix}. \quad (2.10)$$

Samlet kan translasjon og rotasjon beskrives i en 4x4 homogen transformasjonsmatrise T på formen

$$\mathbf{T}_1^0 = \begin{pmatrix} \mathbf{R}_1^0 & \mathbf{p}_1^0 \\ \mathbf{0}^T & 1 \end{pmatrix} \quad (2.11)$$

hvor \mathbf{R}_1^0 og \mathbf{p}_1^0 representerer henholdsvis rotasjon og translasjon fra koordinatsystem 0 til koordinatsystem 1 . Nullvektoren $\mathbf{0}^T$ gjør transformasjonsmatrisen homogen.

I en kjede med flere ledd kan transformasjonsmatrisene for hvert ledd multipliseres med hverandre for å danne transformasjonsmatrisen \mathbf{T} for hele kjeden. Hvert ledd i kjeden er da transponert og / eller rotert med referanse til det forrige koordinatsystemet i kjeden. Transformasjonen i en kinematisk kjede fra arbeidsstykkets koordinatsystem w til verktøyets koordinatsystem t med (n) transformasjoner blir da

$$\mathbf{T}_t^w = \mathbf{T}_1^w \mathbf{T}_2^1 \mathbf{T}_3^2 \dots \mathbf{T}_n^{n-1} \mathbf{T}_t^n. \quad (2.12)$$

I transformasjonsmatrisen \mathbf{T}_t^w i ligning 2.12 er verktøyets posisjon relativt til arbeidsstykkets koordinatsystem gitt i den siste kolonnen, som vist i ligning 2.11. Orienteringen til verktøyets Z-akse er gitt i den tredje kolonnen i ligning 2.12, relativt til arbeidsstykkets koordinatsystem. CL-data for et gitt sett med akseposisjoner til verktøymaskinen kan da finnes med ligningen

$$\begin{pmatrix} i & x \\ j & y \\ k & z \\ 0 & 1 \end{pmatrix} = \mathbf{T}_w^t \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} \quad (2.13)$$

Transformasjonen fra verktøyet til arbeidsstykket kan finnes ved å benytte notasjonen:

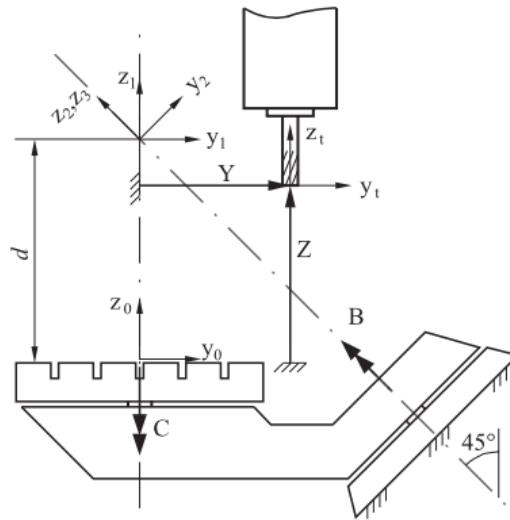
$$\mathbf{T}_w^t = (\mathbf{T}_t^w)^{-1}. \quad (2.14)$$

Med grunnlag i transformasjonene fra verktøyets koordinatsystem til arbeidsstykkets koordinatsystem og motsatt kan ligningene for den direkte- og indirekte kinematikken etableres og brukes i lineariseringsalgoritmer og for generering av NC-data fra CL-data.

Kinematikk for Deckel Maho DMU 50eVolution

I artikkelen *Inverse kinematics of five-axis machines near singular configurations* presenterte Sørby [2] den direkte kinematikken og den analytiske løsningen for den inverse kinematikken for en 5-akse fresemaskin med uortogonale rotasjonsakser, som i Deckel Maho DMU 50eVolution, vist i Figur 2.9. For løsning av de kinematiske ligningene for direkte kinematikk ble følgende koordinatsystem etablert, med referanse til Figur 2.9:

- $x_0y_0z_0$: Base-koordinatsystem, lokalisert i senter av bordet når $B = C = 0^\circ$.
- $x_1y_1z_1$: En translasjon av $x_0y_0z_0$ i en distanse d langs z_0 .
- $x_2y_2z_2$: En rotasjon av $x_1y_1z_1$ i en vinkel $+45^\circ$ om x_1 . Koordinatsystem $x_0y_0z_0$, Koordinatsystem $x_1y_1z_1$ og Koordinatsystem $x_2y_2z_2$ er faste; de beveger seg ikke med maskinaksene.
- $x_3y_3z_3$: En rotasjon av $x_2y_2z_2$ i en vinkel B om z_2 .
- $x_4y_4z_4$: En rotasjon av $x_3y_3z_3$ i en vinkel -45° om x_3 .
- $x_5y_5z_5$: En translasjon av $x_4y_4z_4$ i en distanse $-d$ langs z_4 . Koordinatsystemet $x_5y_5z_5$ er alltid lokalisert i senter av maskinbordet, også etter at B -aksen har blitt rotert.
- $x_wy_wz_w$: Arbeidsstykkets koordinatsystem. Funnet ved å rotere $x_5y_5z_5$ i en vinkel $-C$ om z_5 .
- $x_ty_tz_t$: Et koordinatsystem som er fast tilknyttet freseverktøyet med origo på verktøytuppen.



Figur 2.9: Skisse av en fem-akse maskin med uortogonale rotasjonsakser sett fra siden. Vist med B -akse i posisjonen $B = 0^\circ$ [2].

Med notasjonene $\sin \phi = s_\phi$ og $\cos \phi = c_\phi$ definerte Sørby de homogene transformasjonsmatrisene:

$$\mathbf{T}_0^1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad (2.15)$$

$$\mathbf{T}_1^2 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & c_{45^\circ} & -s_{45^\circ} & 0 \\ 0 & s_{45^\circ} & c_{45^\circ} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad (2.16)$$

$$\mathbf{T}_2^3 = \begin{pmatrix} c_B & -s_B & 0 & 0 \\ s_B & c_B & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad (2.17)$$

$$\mathbf{T}_3^4 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & c_{45^\circ} & s_{45^\circ} & 0 \\ 0 & -s_{45^\circ} & c_{45^\circ} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad (2.18)$$

$$\mathbf{T}_4^5 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -d \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad (2.19)$$

$$\mathbf{T}_5^w = \begin{pmatrix} c_C & s_C & 0 & 0 \\ -s_C & c_C & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad (2.20)$$

$$\mathbf{T}_0^t = \begin{pmatrix} 1 & 0 & 0 & X \\ 0 & 1 & 0 & Y \\ 0 & 0 & 1 & Z \\ 0 & 0 & 0 & 1 \end{pmatrix}. \quad (2.21)$$

Transformasjonen fra freseverktøyets koordinatsystem til arbeidsstykkets koordinatsystem ble definert som:

$$\mathbf{T}_w^t = \mathbf{T}_w^5 \mathbf{T}_5^4 \mathbf{T}_4^3 \mathbf{T}_3^2 \mathbf{T}_2^1 \mathbf{T}_1^0 \mathbf{T}_0^t \quad (2.22)$$

hvor \mathbf{T}_w^t er en funksjon av maskinaksenes variabler, dvs. X, Y, Z, B og C .

Med grunnlag i transformasjonsmatrisene fant Sørby følgende ligninger for den direkte kinematikken:

$$i = \frac{1}{2}(\sqrt{2}c_C s_B - s_C c_B + s_C), \quad (2.23)$$

$$j = \frac{1}{2}(\sqrt{2}s_C s_B + c_C c_B - c_C), \quad (2.24)$$

$$k = \frac{1}{2} + \frac{1}{2}c_B, \quad (2.25)$$

$$x = \frac{\sqrt{2}}{2}[X s_C + (Y + Z - d)c_C]s_B + \frac{1}{2}[-Y + Z - d - (Y + Z - d)c_B]s_C + X c_C c_B, \quad (2.26)$$

$$y = \frac{\sqrt{2}}{2}[(Y + Z - d)s_C - X c_C]s_B + X s_C c_B + \frac{1}{2}[(Y + Z - d)c_C c_B + (Y - Z + d)c_C], \quad (2.27)$$

$$z = \frac{1}{2}[-\sqrt{2}X s_B + (Y + Z - d)c_B - Y + Z + d], \quad (2.28)$$

og følgende ligninger for den indirekte kinematikken:

$$B = \arccos(2k - 1), \quad \text{for } 0^\circ \leq B \leq 180^\circ \quad (2.29)$$

$$C = \arctan \left[(1 - k)i + \sqrt{2(k - k^2)}j, \sqrt{2(k - k^2)}i + (k - 1)j \right], \quad \text{for } -180^\circ \leq B \leq 180^\circ \quad (2.30)$$

$$X = [-y\sqrt{2(k - k^2)} - x + s_x k] \cos C + [x\sqrt{2(k - k^2)} + 2yk - y] \sin C + (d - z)\sqrt{2(k - k^2)}, \quad (2.31)$$

$$Y = [x\sqrt{2(k - k^2)} + yk] \cos C + [y\sqrt{2(k - k^2)} - xk] \sin C - z + d - dk + zk, \quad (2.32)$$

$$Z = [x\sqrt{2(k - k^2)} + yk - y] \cos C + [y\sqrt{2(k - k^2)} - xk + x] \sin C + d - dk + zk. \quad (2.33)$$

Ligningene danner grunnlaget for den frittstående postprosessen som Sørby utviklet for verktøymaskinen, og som kandidaten også har fått tilgang til i forbindelse med prosjektet.

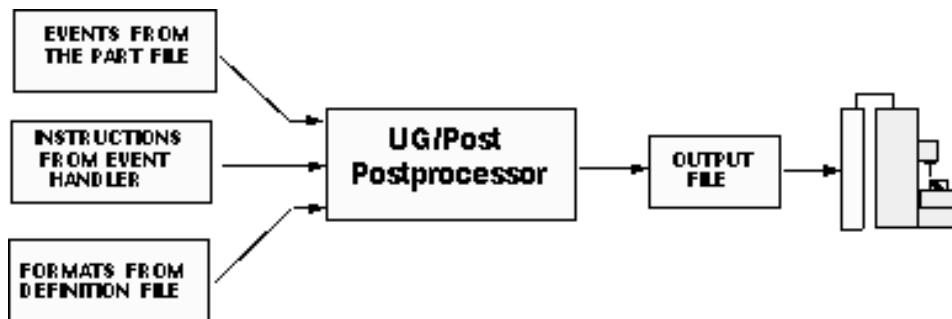
2.2.4 NX Post

NX Post er en integrert postprosessormodul i NX Manufacture. Når maskinkode skal genereres for de planlagte operasjonene i NX Manufacture åpnes NX Post fra en knapp på verktøylinjen. I vinduet for NX Post velges det hvilken maskin det skal postprosesseres for ved å velge riktig postprosessor, og hvor maskinkoden skal lagres. Det kan også spesifiseres en annen filendelse for programfilen som inneholder maskinkoden, enn den som er forhåndsdefinert i postprosessen. Når postprosesseringen utføres blir programfilen opprettet på den bestemte lokasjonen i tillegg til at maskinkoden vises i et eget dialogvindu i NX Manufacture. Dette gjør det raskt og enkelt å kontrollere maskinkoden manuelt.

Kjernen i NX post er Manufacturing Output Manager (MOM). Når NX Post kjøres genererer MOM NC-data fra informasjonen om verktøybanene som er lagret i modellfilen (.prt) fra NX Manufacture. For at MOM skal kunne generere korrekt NC-data for en bestemt maskin kreves to forskjellige filer, i tillegg til modellfilen:

- Event Handler (.tcl)
- Definition File (.def)

Event Handler inneholder et sett av instruksjoner som dikterer hvordan hver type hendelse skal prosesseres av NX Post. Definition File inneholder informasjon som er knyttet til en bestemt kombinasjon av maskin og maskinstyring. Sammen utgjør disse to filene postprosessen til en bestemt maskin for NX. Figur 2.10 viser input til og output fra NX Post.

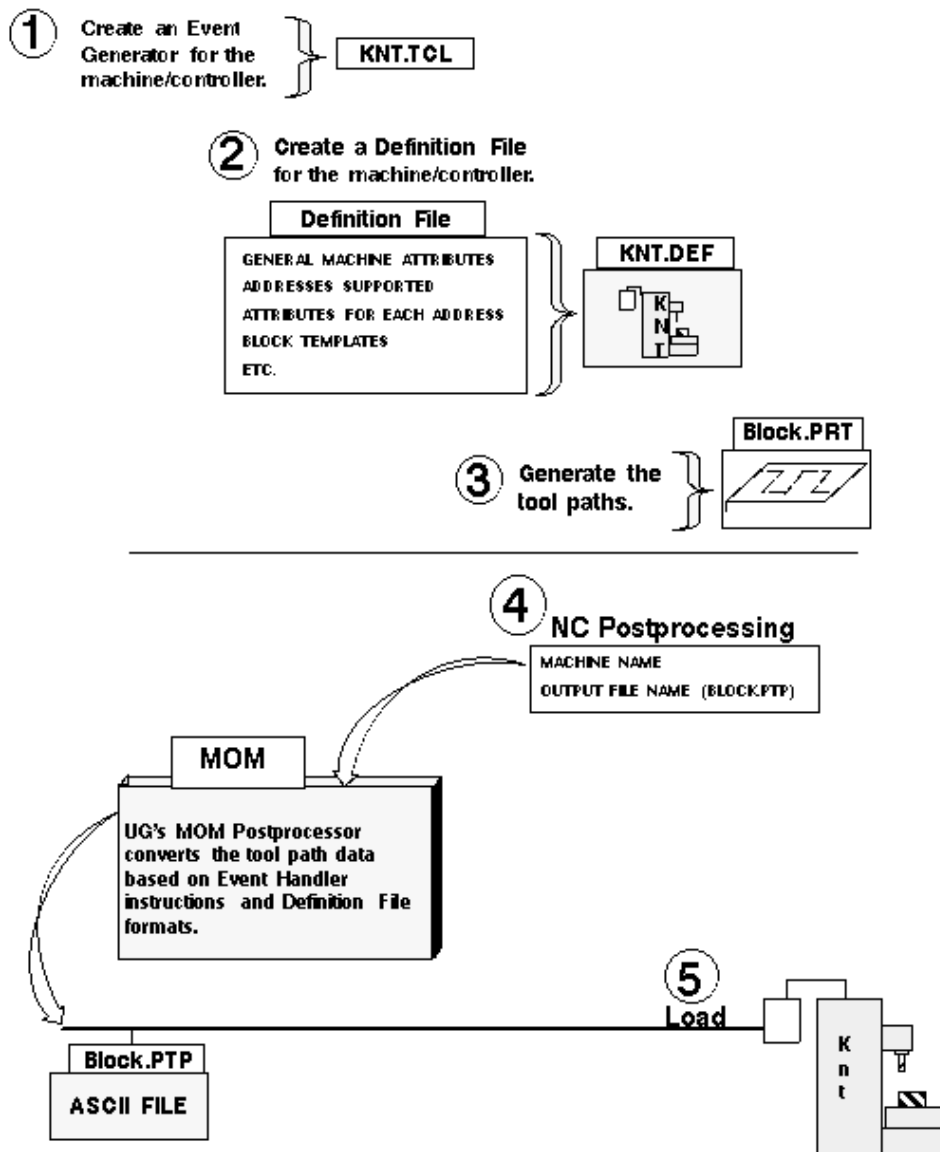


Figur 2.10: Input og output i NX Post [4].

Når postprosessering utføres av NX Post brukes først en kjernemodul i NX Post, kalt Event Generator, til å lese gjennom verktøybanedataene i modellfilen. Basert på verktøybanedataene og tilhørende variabelinformasjon danner Event Generator hendeleser. En hendelse er en samling av data som til slutt vil føre til at CNC-maskinen utfører en handling. Hendelsene blir deretter sendt videre til MOM, som anvender kinematikk og bruker Event Handler og Definition File for å bestemme hvordan hendelsene skal prosesseres og hvordan formateringen av NC-dataene skal være for den bestemte maskinstyringen. Deretter skriver MOM ut den formaterte maskinkoden til den spesifiserte utfilen [4].

Figur 2.11 viser en illustrasjon over de forskjellige stegene i postprosessering i NX Post. Siden

Event Handler-filen og Definition File er postprocessorfiler som hører til den maskinen som maskinkoden skal kjøres i, er det kun nødvendig å lage disse filene en gang for hver maskin.



Figur 2.11: Steg for postprosessering med NX Post [4].

2.2.5 NX PostBuilder

PostBuilder er Siemens UGS' egen programvare for å utvikle postprocessorfilene (Definition File og Event Handler) som brukes sammen med NX Post. Filene er programmert i TCL

(Tool Command Language), som er et åpent og gratis programmeringsspråk. Siden TCL er et script-språk, trenger det ikke å kompiles og syntaksen som brukes ligner på den som brukes i programmeringsspråket C. For en erfaren programmerer kan postprosessorfilene som brukes sammen med NX Post skrives fra bunnen i TCL, men PostBuilder gjør det enklere og raskere å sette opp postprosessorfilene.

PostBuilder har et grafisk brukergrensesnitt og baserer seg på en blanding av brukervalg og spesialkommandoer som brukeren selv kan programmere i TCL. Når en ny post opprettes i PostBuilder dannes postprosessorfilene, som NX Post bruker, i tillegg til en Post User Interface file (.pui), som brukes av Post Builder til å vise og redigere postprosessorfilene i det grafiske brukergrensesnittet.

Når arbeidet med en ny postprosessor startes i PostBuilder velges først maskinens type og grunnoppsett som et brukervalg, for eksempel 5-akse fresemaskin med to rotasjonsakser i bordet og enheter i [mm]. Når maskintype og oppsett er valgt kommer man inn i hoveddelen av programmet, som består av fem faner:

- Machine Tool
- Program & Tool Path
- N/C Data Definitions
- Output Settings
- Virtual N/C Controller

Under fanen Machine Tool kan maskinens konfigurasjon og parametre spesifiseres. Hjemposisjon, aksenes begrensninger og maksimal matingshastighet kan blant annet defineres her, i tillegg til geometriske størrelser og vektorer som har betydning for kinematikken i PostBuilder, som avstand mellom krysningspunkt for rotasjonsakser og rotasjonsaksenes enhetsvektorer.

Under fanen Program & Tool Path gjøres mesteparten av arbeidet i PostBuilder. Det er i denne delen av programmet ordene som skal brukes (for eksempel G00 for ilmating og M08 for kjølevæske) defineres og sekvensen de skal skrives ut i (IJK skal skrives ut etter XYZ, for eksempel). Det er også i denne delen av PostBuilder at spesialkommandoer (custom commands) kan programmeres. Custom Commands brukes til å programmere funksjonalitet i TCL som postprosessoren i utgangspunktet ikke har støtte for.

I Program & Tool Path er genereringen av maskinkoden delt opp flere i kategorier, som for eksempel "Operation Start Sequence" eller "Tool Path". Hver kategori inneholder ett eller flere nivå, for eksempel "Auto Tool Change" eller "Linear Motion", hvor NC-blokker kan legges til. NC-blokkene kan bestå av posisjoner, ord (G-koder og M-koder), operatørmeldinger og mer. De programmerte spesialkommandoene kan også legges til med NC-blokker som inneholder kallenavnet på kommandoen.

Under fanen Program & Tool Path er det også støtte for å legge til koblede postprosessorer. Dette brukes for eksempel når postprosessorer for flerfunksjonsmaskiner utvikles, da disse

maskinene ofte krever forskjellige sett med kinematikk etter for hvilken funksjon i maskinen som skal brukes. I master-postprozessoren, som inneholder koblingene til de andre postprosessorene, defineres et kallenavn for hver postprocessor. Når maskineringsoperasjoner planlegges i NX Manufacture må det samme kallenavnet som ble definert i master-postprozessoren legges til som tekst i UDE-en "Head". Når programmet skal postprosesseres i NX Manufacture velges kun master-postprozessoren. Når programmet kommer til en operasjon som krever et bytte av postprocessor, fra dreining til fresing for eksempel, henter automatisk master-postprozessoren inn den aktuelle postprozessoren, gitt at kallenavnet for den er definert som en UDE i den aktuelle operasjonen, eller gruppen av operasjoner.

Under fanen N/C Data Definitions defineres formateringen for ordene og kodene med grenseverdier, antall siffer, ledende nulltall og mer. Formatering for sekvensnummerering og tegnsetting for operatørmelding hører også til denne menyen.

Hvilken filendelse som skal brukes som standard for de postprosesserte programmene defineres under fanen "Output Settings". I denne menyen finnes også et verktøy som viser endringene som er gjort i postprozessoren fra da filen ble åpnet.

Menyen Virtual N/C Controller brukes til å generere filer som brukes for å kjøre postprocessorbasert-maskinsimulering i NX Manufacture. Med virtuell N/C kontroller brukes NC-data som postprozessoren genererer som grunnlag for maskinsimuleringen i stedet for CL-data, som normalt brukes til simulering.

2.3 STEP-NC

STEP (Standard for the Exchange of Product model data) er en samling av ISO - standarder som er utviklet for å standardisere måten å beskrive produktdata på. Standarden har ISO - nummer 10303 og den formelle tittelen *Automation systems and integration — Product data representation and exchange* [10]. Til sammenligning med IGES (Initial Graphics Exchange Specification), som primært ble utviklet for deling av geometrisk data mellom CAD (computer aided design) - systemer, omfatter STEP produktdata som dekker hele livsløpet til et produkt [11].

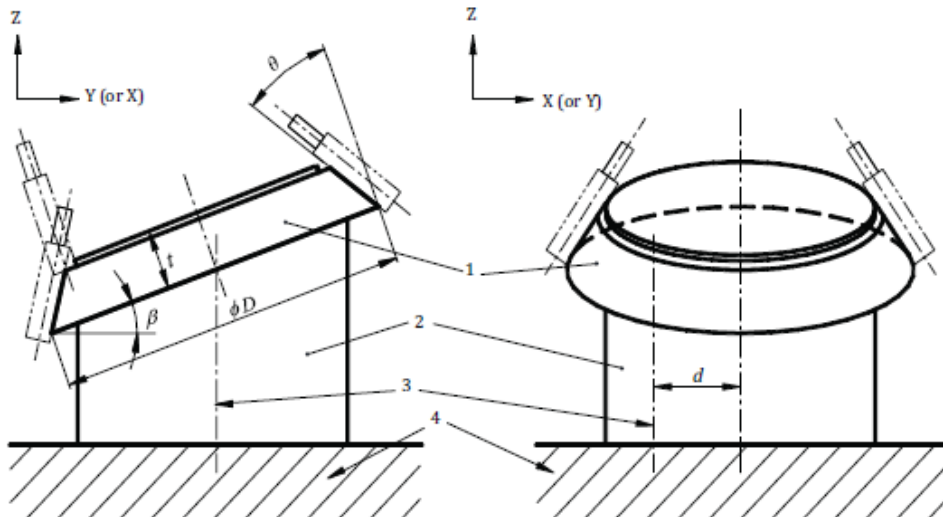
STEP-NC omhandler bruken av STEP for programmering av numerisk styrte verktøymaskiner og har ISO-nummer 14649 [12]. I motsetning til ISO6983 [8], som bruker programmer med G-kode som er postprosessert for en bestemt maskin, legger STEP-NC opp til at verktøymaskinen skal motta nøytral data om geometri, verktøybaner krav til nøyaktighet etc. og prosessere denne informasjonen på maskinstyringen [13]. STEP-NC programmer kan dermed programmeres en gang og kjøres på flere forskjellige typer maskiner så lenge maskinene har støtte for operasjonene som skal gjøres, og dermed eliminere behovet for den typen postprosessorer som brukes i dag mellom CAM og verktøymaskin.

Programmeringsspråket som brukes i STEP-NC er det nøytrale datamodeleringsspråket EXPRESS, definert i ISO 10303-11 [14]. Dette standardiserte og nøytrale språket utgjør et knutepunkt for deling av informasjon mellom forskjellige avdelinger, programvarer og maskiner gjennom hele produktets levetid.

2.4 Teststandarder for testkomponenter

ISO 10791 er en standard som har til hensikt å gi så bred og omfattende informasjon som mulig for tester og kontroller for numerisk styrte maskiner som kan utføres for sammenligning, aksept, vedlikehold eller andre formål. I del 7 av standarden, ISO 10791-7 [5], spesifiseres standard testkomponenter for 3- til 5-akse maskineringsentre med referanse til ISO 230-1 [15]. Test M3 i standarden er spesielt interessant da den tar for seg en testkomponent for simultan 5-aksefresing for alle typer 5-akse maskineringsentre med tre lineære akser og to rotasjonsaker.

Testkomponenten i M3, vist i Figur 2.12, består av en konisk skive som er oppspent med en fast vinkel i en fikstur. Fiksturen monteres med en forskyvning fra senter på arbeidsbordet både langs X-akse og Y-akse. Standarden inneholder også dimensjoner, vinkler og annen relevant informasjon for utføring av testen.



Figur 2.12: Testkomponent for simultan 5-aksemaskinering [5].

Kapittel 3

Utvikling av postprosessorer for NX

I dette avsnittet beskrives utviklingen av postprosessorer til NX Manufactue for Deckel Maho 5-akse fresemaskin og Okuma Genos flerfunksjonsmaskin i NX PostBuilder. Av hensyn til plass og sammenheng i teksten er kommandoer som er programmert i TCL plassert i vedlegg med referanse i teksten.

3.1 Postprosessor for Deckel Maho 5-akse fresemaskin



Figur 3.1: Deckel Maho DMU 50eVolution i verktøymaskinlaboratoriet ved IPK

I arbeidet med utvikling av postprosessor for Deckel Maho (avbildet i Figur 3.1) ble først kinematikken til maskinen definert i NX PostBuilder. I denne maskinen har rotasjonsvektoren til basen, som har adressen B, en vinkel på 45° relativt til arbeidsbordet i maskinen når B-aksen er posisjonert i 0° . I PostBuilder ble denne vinkelen definert med enhetsvektorene $[0 \ -1 \ 1]$. Maskinens nullpunkt ble definert til å ligge langs arbeidsbordets rotasjonsakse, som har adressen C, og på bordets overflate. Den relative avstanden fra nullpunktet til skjæringspunktet mellom B-aksen og C-aksen har stor betydning for kinematikken og ble lagt inn med data fra oppmålinger som ble utført da maskinen ble installert. Målebrevet som inneholder denne informasjonen finnes i Vedlegg B. B-aksens arbeidsområde $[0^\circ - 180^\circ]$ ble lagt inn og aksens rotasjonsretning ble definert til negativ, i henhold til de faktiske forholdene i maskinen. C-aksens arbeidsområde ble definert til $[0^\circ - 360^\circ]$ med positiv rotasjonsretning og til slutt ble spindelaksen definert med enhetsvektoren $[0 \ 0 \ 1]$.

Etter at kinematikken var definert ble de korrekte G-kodene og M-kodene for maskinen definert i samsvar med informasjonen funnet i dokumentasjonen til maskinen [16]. Riktig tegnsetting for operatørmelding og slutt på kodeblokk ble definert og at .i skulle brukes som filendelse for den genererte maskinkoden. Maskinstyringen krever også at symbolet % plasseres i starten og slutten av programmet og dette ble lagt inn som blokker i "Program and Tool Paths" i PostBuilder. For å øke leseligheten til de genererte programmene ble det også programmert en kommando som skriver ut navnet på den aktive operasjonen i starten av hver operasjon i maskinkoden. Denne kommandoene ble lagt til som en blokk i "Program and Tool Paths" i PostBuilder.

Et program som var generert med postprosessoren ble testet på maskinen og ble det oppdaget at uttrykket for sirkelinterpolasjon i postprosessoren var feil. I NX PostBuilder uttrykkes sirkelens senterpunkt relativt til verktøyets posisjon som standard. Maskinstyringen til denne maskinen bruker absolutt posisjonering av sirkelens senterpunkt ved sirkelinterpolering. Utrykket for I J og K ble derfor endret til å gi ut absolutt posisjon til sirkelsenter i PostBuilder. Etter denne endringen ble sirkelinterpoleringen utført korrekt i maskinstyringen uten feilmeldinger.

Det ble også programmert en kommando i PostBuilder som kjører aksene bort fra arbeidsstykket og roterer basen og bordet til 0° . Kommandoen ble lagt til som en blokk i "Program and Tool Paths" og kalles opp i slutten av programmet. Denne kommandoen har ingen innvirkning på resultatet for maskineringen, men sparer maskinoperatøren for å måtte kjøre aksene bort fra arbeidsstykket og rotere tilbake basen og bordet etter maskineringen. Koden for kommandoen finnes i Vedlegg C.1.

3.2 Postprocessor for Okuma Genos flerfunksjonsmaskin



Figur 3.2: Okuma Genos L200E-M i verktøymaskinlaboratoriet ved IPK

Flerfunksjonsmaskinen Okuma Genos L200E-M, avbildet i Figur 3.2, er en relativt ny maskin med mye funksjonalitet. Maskinen kan maskinere i XY-planen (på endeflaten til arbeidsstykket) og YZ-planen (på siden av arbeidsstykket) i tillegg til å kunne dreie i XZ-planen. Hovedspindelen i maskinen, som brukes til dreining, kan også låses eller brukes til posisjonering og har da adressen C. I tillegg har maskinen verktøyspindel langs X-aksen og langs Z-aksen og fysisk Y-akse, i tillegg til flere andre funksjoner. Den store funksjonaliteten som maskinen har krever også en omfattende postprocessor.

I NX PostBuilder kan kun ett sett med kinematikk defineres per postprocessor. Da funksjonaliteten og kinematikken for denne maskinen endres etter hvilken type oppgave som skal utføres ble det utviklet flere postprocessorer for maskinen som kalles opp av en master-postprocessor. Følgende fire postprocessorer ble utviklet for maskinen:

Postprocessor	Plan	Akser	Verktøyspindel	Kallenavn for UDE
Master	-	-	-	-
Dreining	XZ	XZ	-	TURN
Fresing/boring	XY	XZC	Z	MILL
Fresing/boring	YZ	XYZC	X	MILL_X

Tabell 3.1: Postprocessorer for Okuma Genos L200E-M

Master-postprozessoren starter og avslutter programmet, og henter inn de andre postprosessorene i henhold til det som har blitt definert i NX Manufacture. For at master-postprozessoren skal kunne hente inn de andre postprosessorene ble de lagt til som "Linked Postprocessors" i NX PostBuilder. I denne delen av PostBuilder ble det også definert kallenavn for de forskjellige postprosessorene, som må samsvare med oppkallingen fra "User defined events" (UDE) i NX Manufacture. Kallenavnene som ble benyttet for de forskjellige postprosessorene for denne maskinen finnes i den siste kolonnen i Tabell 3.1. Hver enkelt operasjon, eller gruppe av operasjoner, må merkes med UDE-en "Head" og kallenavnet til den postprozessoren som skal brukes i NX Manufacture.

Postprozessoren for dreining er den enkleste av postprosessorene for maskinen, foruten master-postprozessoren. I PostBuilder ble de korrekte G-kodene og M-kodene definert ut fra dokumentasjonen til maskinen [17] og riktig tegnsetting for operatørmelding ble valgt. For å gjøre den genererte maskinkoden mer lettlest ble det lagt til en kommando som skriver ut operasjonsnavnet i starten av hver operasjon sammen med ordet "DREIING", som signaliserer hvilken postprocessor som brukes i den aktuelle operasjonen. Tilsvarende kommandoer ble også programmert for de andre postprosessorene.

I standardoppsettet for postprosessorer i Post Builder beskrives verktøy med adressen "T" og to siffer for verktøynummer. I denne maskinen brukes ordet "TXXXXXX" når verktøy kalles opp, hvor XX er verktøynummeret. De to gjentakelsene av verktøynummeret laster inn tilhørende offset-data fra verktøytabellen. I maskinkoden skal derfor verktøyet som står i verktøyholder nr. 1 kalles opp med ordet "T010101". For å få riktig formatering av ordet for oppkalling av verktøy ble en kommando programmert og lagt til som en blokk i verktøybytte-hendelsen under "Program and Tool Path" i PostBuilder. Kommandoen finnes i Vedlegg C.2.

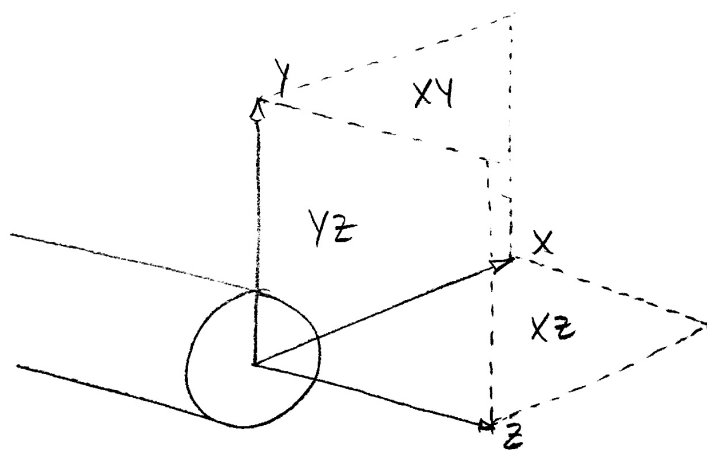
I PostBuilder lagres verktøynummer i en variabel uten ledende nulltall for verdier som er mindre enn 10. For å unngå at formateringen ville bli "TXXX" for verktøynummer som var mindre enn 10, ble det nødvendig å legge inn en if-setning i kommandoen for å legge til et ledende nulltall dersom verktøynummeret var mindre enn 10. Da PostBuilder ikke aksepterer at tall står sammen med variabelnavn ble det også nødvendig å lage en ny variabel som inneholdt verdien "0" i kommandoen. Denne måten som ble brukt for å endre formateringen av verktøynummeret kunne nok ha blitt gjort på en mer elegant måte, men i praksis har koden fungert uten feil. Siden oppkallingen av verktøy er lik uavhengig av hvilken type operasjon som skal gjøres i maskinen ble kommandoen også brukt i de andre postprosessorene for maskinen.

Når verktøymaskinen skal utføre maskineringsoperasjoner i XY-planet og YZ-planet (se Figur 3.3) kreves det at noen av G-kodene og M-kodene som brukes ved dreining byttes. Blant annet brukes andre koder for sirkelinterpolasjon og spindelhastighet for maskinering i XY-planet og YZ-planet, enn ved dreining i XZ-planet. I noen tilfeller er det også forskjeller mellom kodene som brukes for maskinering i XY-planet og YZ-planet. I PostBuilder brukes normalt menyen "Format" under fanen "NC-definitions" til å formatere hvordan de forskjellige kodene skal skrives ut. En "bug" i installasjonen av PostBuilder gjorde det umulig å endre eller definere nye formateringer uten at antall siffer ble definert til "0" eller "9". Interpolasjon i XY-planet og YZ-planet krever 3 siffer i G-koden og det ble derfor nødvendig å legge inn denne formateringen direkte i definisjonsfilen (.def) til postprozessoren. TCL-editoren Crimson Editor [18] ble brukt til redigeringen av filen.

Ved maskinering med de linære aksene i XY-planet og i YZ-planet kreves det av maskinstyringen at C-aksen låses med en bestemt M-kode. PostBuilder har ikke en standardfunksjon for dette, men det finnes en forhåndsprogrammert kommando som kan kalles opp ved hjelp av en UDE. Kommandoen trer i kraft når auto-låsing av C-aksen defineres som en UDE for en operasjon i NX Manufacture. Da det alltid er ønskelig å låse denne aksene når den ikke brukes under maskineringen ble en ny kommando programmert, som definerer at auto-låsing av C-aksen skal være aktiv. Kommandoen ble lagt til som en blokk i starten av operasjonen under fanen "Program and Tool Path" i PostBuilder og finnes i Vedlegg C.3.

Når C-aksen brukes til posisjonering i et program kreves forskjellige M-koder for å beskrive rotasjonsretningen som skal brukes for å oppnå den programmerte vinkelen. Dersom aksene kun skal posisjonere arbeidsstykket før maskinering er ikke rotasjonsretningen av stor betydning, men når C-aksen brukes i maskineringen er det essensielt at aksene roterer riktig retning. I PostBuilder er det ikke støtte for å definere M-koder for beskrivelse av rotasjonsretning, i det grafiske grensesnittet. For å skrive ut de nødvendige M-kodene for beskrivelse av rotasjonsretning, henholdsvis M15 for rotasjon med klokken og M16 for rotasjon mot klokken, ble en ny kommando programmert og lagt til som en blokk i alle bevegelseshendelsene (linear move, rotary move og rapid move) i Postbuilder. Kommandoen finnes i Vedlegg C.4 og er den mest omfattende av kommandoene som ble programmert i TCL.

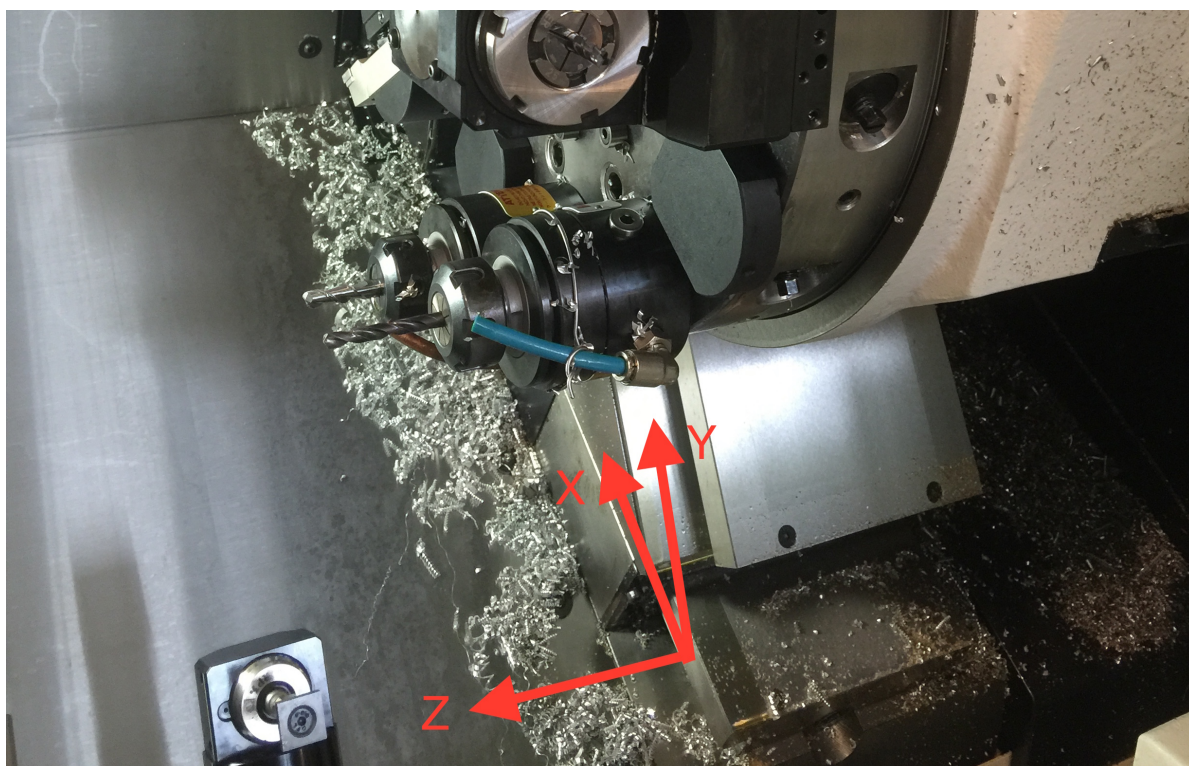
Kommandoen er programmert for å kun kjøres når det er nødvendig, som er ved endring i vinkelen for C-aksen. Dersom det er en endring i vinkelen sammenlignes rotasjonsretningen med forrige rotasjonsretning og det er kun ved endring i rotasjonsretningen ny M-kode blir skrevet ut i maskinkoden. På dette stadiet ble koden testet og et problem ved rotasjon over 0° ble oppdaget. Selv om rotasjonsretningen var den samme ville logikken i koden skrive ut ny, og feil, retning for rotasjonen i den første blokken som inneholdt en vinkel på den andre siden av 0° . Kommandoen ble derfor utvidet med logikk som sjekker om vinkelen krysser 0° og unngår å skrive ut ny retning dersom det er tilfellet. Grenseverdiene som ble brukt for dette var 350° og 10° , men andre verdier kan også benyttes. Etter denne endringen har kommandoen fungert feilfritt i alle tilfeller. Kommandoen ble brukt både i postprosessoren for maskinering i XY-planet og for postprosessoren for maskinering i ZY-planet.



Figur 3.3: Skisse av arbeidsstykke og maskineringsplan i OKUMA GENOS L200 E-M

For maskinering i XY-planet er den fysiske Y-aksen (se Figur 3.4) i maskinen overflødig, da verktøyet kan posisjoneres i hele arbeidsområdet med X- og C-aksen. Ved å benytte interpolasjon med disse aksene kan også verktøyet maskinere i rette linjer. Med hensyn til dette, og at den fysiske Y-aksen har et begrenset arbeidsrom, spesielt når posisjonen i X er negativ, ble postprosessoren for maskinering i XY-planet utviklet uten støtte for Y-akse.

I motsetning til maskinering i XY-planet har den fysiske Y-aksen verdi ved maskinering i YZ-planet. I dette planet kan Y-aksen brukes til "off-center" maskinering, som for eksempel hullboring som ikke går gjennom senter av komponenten. Denne postprosessoren ble derfor utviklet med støtte for fysisk Y-akse. For å benytte den fysiske Y-aksen kreves det egne G-koder for aktivering og deaktivering. Dette er ikke en standardfunksjon og det ble derfor programmert to kommandoer som skriver ut disse G-kodene. Kommandoene lagrer Y-aksens status (aktivert/deaktivert) i en felles variabel og bruker denne variabelen til å avgjøre om ny G-kode skal skrives ut. Kommandoene er lagt til som blokker i starten og slutten av operasjoner under "Program and Tool Paths", for henholdsvis aktivering og deaktivering av den fysiske Y-aksen. Kommandoen for aktivering av fysisk Y-akse finnes i Vedlegg C.5, og kommandoen for deaktivering av fysisk y-akse finnes i Vedlegg C.6.



Figur 3.4: Orientering av de fysiske, lineære aksene i OKUMA GENOS L200 E-M

Kapittel 4

Testmodeller

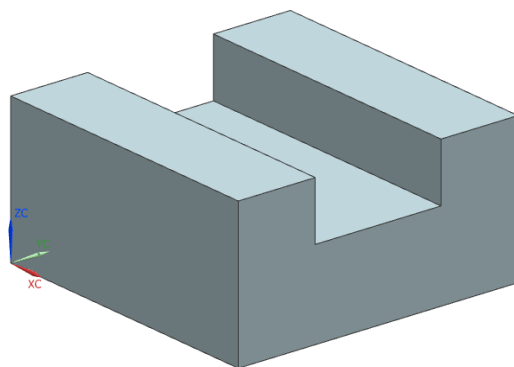
I dette kapitlet presenteres CAD-modellene som ble brukt som grunnlag for arbeidet i CAM og for testing av postprosessorene. Alle CAD-modellene er konstruert i Siemens NX 9.0 av kandidaten, og maskintegningene for konstruksjonene finnes som vedlegg i slutten av oppgaven. Formålet med CAD-modellene var å gi et godt grunnlag for å teste funksjonaliteten i CAM-systemene og for utvikling, testing og feilsøking av postprosessorene.

For videre arbeid i CAM ble konstruksjonene eksportert som part-fil for NX Manufacture og IGES-fil (Initial Graphics Exchange Specification) for GibbsCam.

4.1 5-aksefresing

4.1.1 Testmodell 1

Den første testmodellen, vist i Figur 4.1, bestod av en enkel blokk med et utfrest spor. Hensikten med denne testmodellen var å bli kjent med de elementære funksjonene i CAM-systemene, verifisere at posisjonering for de lineære aksene var korrekt i de genererte maskinkodene, og at formateringen stemte overens med verktøymaskinens dialekt. Det ble planlagt at verktøybanene for testmodellen skulle ligge i to eller flere baner, parallelt med sideveggene i sporet, og at maskineringen skulle gjøres i to eller flere lag.

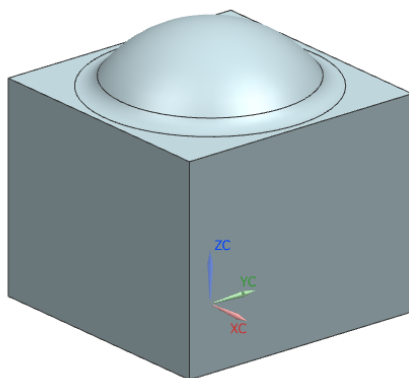


Figur 4.1: Testmodell 1 - Deckel Maho

4.1.2 Testmodell 2

Den andre testmodellen, vist i Figur 4.2, bestod av en kuleformet overflate som skulle maskineres fra et råemne utformet som en blokk. Med denne testkomponenten ble det lagt opp til at maskineringen skulle bestå av to forskjellige operasjoner med et programmert verktøybytte mellom operasjonene. Den først operasjonen skulle være grovfresing med de linære aksene og en pinnefres som det valgte verktøyet. Det ble lagt opp til at verktøybanene skulle inneholde sirkelbevegelser for å teste funksjonaliteten for sirkelinterpolasjon. Den neste maskineringsoperasjonen skulle være konturfresing av kuleflaten med en kulefres. Ved å posisjonere arbeidsstykket ut fra senter på arbeidsbordet i maskinen i X og Y, og spesifisere at verktøyaksen skulle stå vinkelrett på den drivende geometrien, kuleoverflaten, ville det være nødvendig å bruke samtlige av de 5 aksene under maskineringen. Det er også lett å inspisere og se feil og defekter på en kuleformet flate.

Med kulefres kan den samme komponenten maskineres i en 3-akse fresemaskin, men siden hovedformålet med testen var å feilsøke og teste postprosessorer, ble det ikke fokusert på å konstruere geometri med formål om å være umulig eller problematisk å maskinere i maskiner med færre akser.



Figur 4.2: Testmodell 2 - Deckel Maho

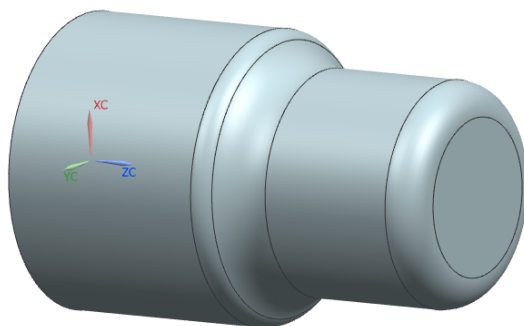
I stedet for, eller i tillegg til, testmodell 2 kunne testkomponent M3 i ISO 10791-7 [5] ha blitt brukt som testmodell, med grunnet behovet for fikstur og at hensikten først og fremst var å teste funksjonalitet fremfor nøyaktighet ble denne testkomponenten ikke valgt i oppgaven.

4.2 Dreining og 4-aksefresing

Siden flerfunksjonsmaskinen kan maskinere i flere forskjellige plan med både linære og roterende akser ble fire forskjellige testmodeller konstruert. Av praktiske årsaker ble alle testmodellene basert på 30mm diameter aluminiumsstang som råemne.

4.2.1 Testmodell 1

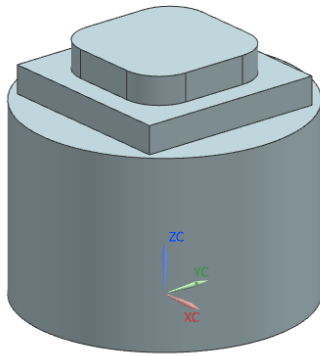
Testmodell 1, vist i Figur 4.3, ble konstruert for å teste funksjonaliteten for planlegging av dreieoperasjoner i CAM-systemene og for å analysere den genererte maskinkoden fra postprosessorene. Det ble lagt opp til at maskineringen skulle bestå av plandreining, grovdreining, verktøybytte og kontur findreining. I konstruksjonen av testmodellen ble det lagt inn buformet geometri/radiuser for å teste funksjonaliteten for sirkelinterpolasjon i postprosessorene.



Figur 4.3: Testmodell 1 - Okuma

4.2.2 Testmodell 2

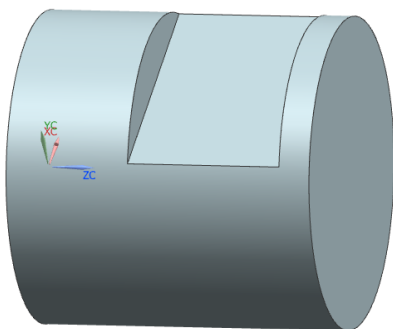
Testmodell 2, vist i Figur 4.4, ble konstruert for å teste funksjonaliteten i postprosessorene for generering av maskinkode for fresing i XY-planet, med verktøyspindel langs Z-aksen. Ved konstruksjonen av testmodellen ble det lagt opp til at de genererte verktøybanene skulle bestå av både rettlinjede og buformede linjestykker for å teste sirkelinterpolasjon. Det ble også planlagt at maskineringen kunne utføres med X-, Y- og Z-akse og låst C-akse, eller med X- og Z-akse og aktiv C-akse.



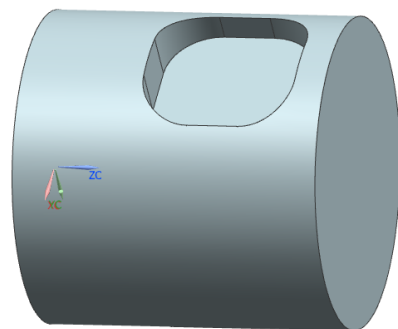
Figur 4.4: Testmodell 2 - Okuma

4.2.3 Testmodell 3

Testmodell 3, vist i Figur 4.5, ble konstruert for å teste generering av maskinkode for fresing i YZ-planet, på emnets side, med verktøyspindel langs X-aksen. I konstruksjonen av testmodellen ble det lagt opp til å teste posisjonering og låsing av C-akse før maskinering med alle de linære aksene. Hensikten med det utfreste sporet (Figur 4.5a) på siden av testmodellen var å teste generering av rettlinjede verktøybaner, mens hensikten med de buede hjørnene i den utfreste lommen (Figur 4.5b) var å teste postprosessorenes støtte for sirkelinterpolasjon for fresing i YZ-planet.



(a) Spor

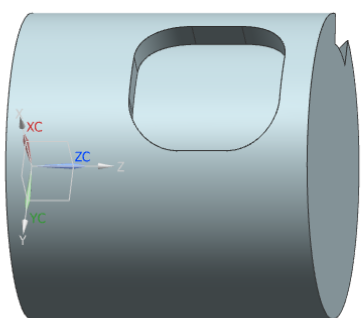


(b) Lomme

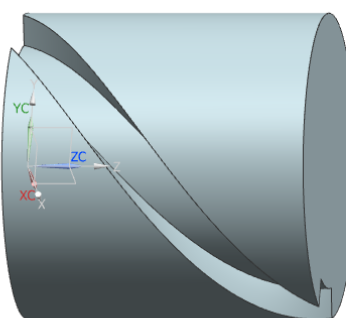
Figur 4.5: Testmodell 3 - Okuma

4.2.4 Testmodell 4

Testmodell 4, vist i Figur 4.6, ble konstruert for å teste funksjonaliteten for radiell lommefresing og radiell konturfresing i CAM-systemene og postprosessorene. Både den buede bunnflaten i den utfreste lommen (Figur 4.6a) og det radielle sporet (Figur 4.6b) ville kreve aktiv bruk av C-aksen under maskineringen. Begge maskineringsoperasjonene skulle utføres i YZ-planet med spindel langs X-aksen.



(a) Lomme



(b) Spor

Figur 4.6: Testmodell 4 - Okuma

Kapittel 5

Planlegging av maskineringsoperasjoner i CAM

I dette kapitlet beskrives planleggingen av maskineringsoperasjonene i CAM-systemene Gibbs-Cam og NX Manufacture for testmodellene som ble presentert i Kapittel 4. Resultatene fra arbeidet i CAM-systemene presenteres i form av figurer og simuleringer som viser verktøybanenes utforming. I tillegg blir de genererte maskinkodene fra testene kommentert, og feil og mangler presentert.

5.1 5-aksemaskinering

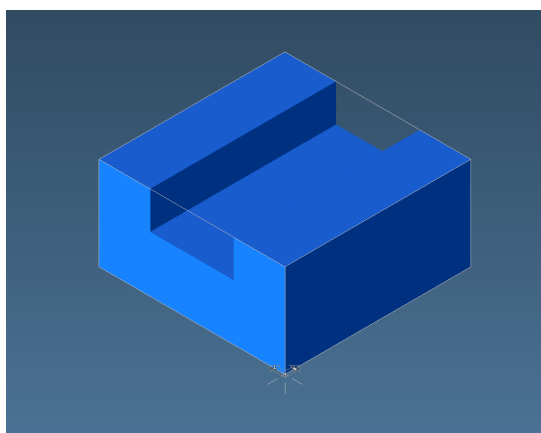
5-aksemaskinering er ikke noe nytt for verktøymaskinlaboratoriet ved IPK, og laboratoriet har hatt utstyr for denne typen bearbeiding i lang tid. På tross av dette er kompetansen for planlegging av 5-aksemaskinering i CAM begrenset blant de ansatte i laboratoriet. Noe av årsaken til dette skyldes sannsynligvis at det lenge har vært uvisst om postprosessoren har vært helt fungerende, og eventuelt hvilke mangler den har. I denne delen av oppgaven blir funksjonaliteten til CAM-systemene og postprosessorene for 5-aksemaskinering testet og eventuelle mangler dokumentert.

5.1.1 GibbsCam

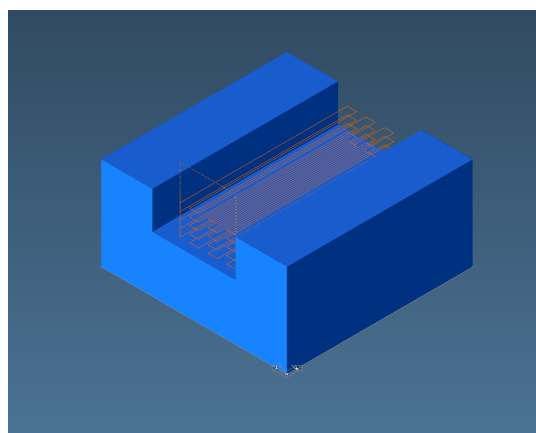
Test 1

Maskineringen av testmodell 1 ble i GibbsCam lagt inn som en grovbearbeidingsprosess ("Roughing") uten gjenstående materiale med en 20 mm diameter pinnefres. Sporet som skulle freses ut hadde en dybde fra toppen av råemnet på 40 mm, som med en kuttdybde på 8mm gav 5 lag. I prosessvinduet ble sikk-sakk mønster for maskineringen valgt og bunnflaten i sporet som drivende geometri. Denne profilen gav et standard inngrep på halvparten av verktøyets diameter, i dette tilfellet 10mm. Som det kommer frem av verktøybanene og simuleringene vist i Figur 5.1a - 5.1c, ville det være gjenstående materiale etter maskineringen. For å strekke verktøybanene lengre ut fra sporet, slik at materialet på endene også ville bli fjernet ble en ny geometri for maskineringen tegnet i GibbsCam. Verktøybanene som ble generert med den nye geometrien gav gode resultater, som det kommer frem i Figur 5.1d - 5.1f.

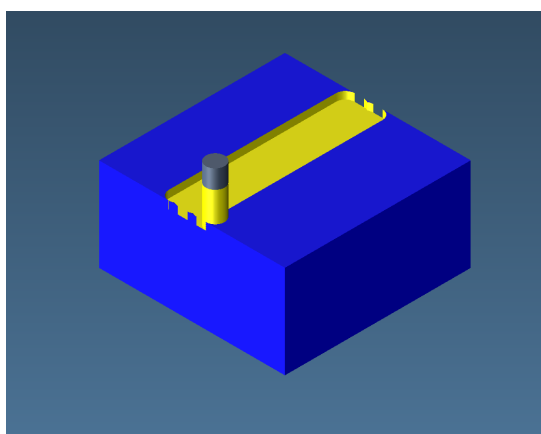
Operasjonene ble postprosessert, og det ble ikke funnet noen feil ved gjennomlesning. Da denne testen var forholdsvis enkel ble ikke programmet kjørt på maskinen.



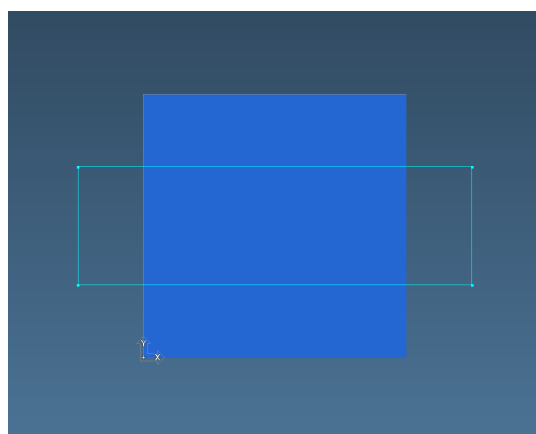
(a) Geometri og råemne



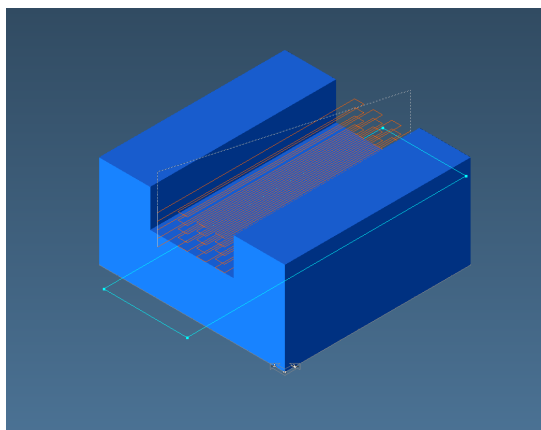
(b) Verktøybaner



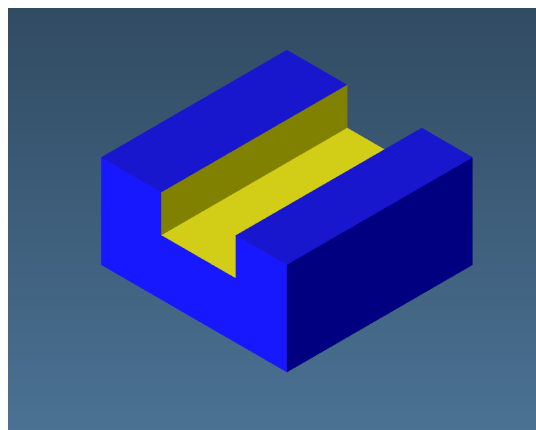
(c) Simulering av maskinering



(d) Inntegnet geometri og verktøybaner



(e) Nye verktøybaner



(f) Simulering av sluttprodukt

Figur 5.1: Verktøybaner og simulering for test 1 i GibbsCam.

Test 2

Maskineringen av testmodell 2 ble planlagt med to forskjellige operasjoner i GibbsCam:

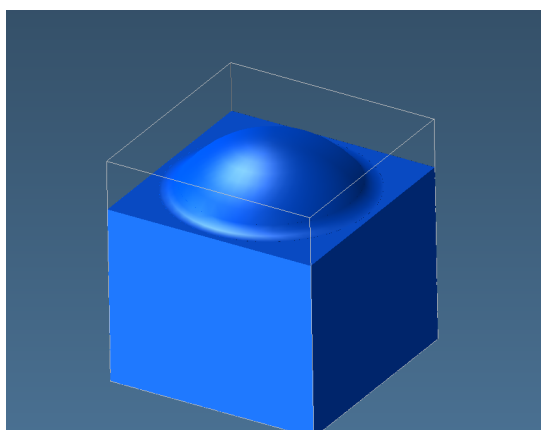
- 3-akse grovfresing (Figur 5.2)
- 5-akse konturfresing (Figur 5.3)

I grovbearbeidingsprosessen ("Roughing") maskineres det meste av materialet bort i fire lag med en 16mm pinnefres som verktøy. I prosessvinduet ble mønsteret "Offset With Cleanup" valgt og de genererte verktøybanene bestod av en blanding av rettlinjede- og sirkelformede linjestykker. Den totale, aksielle dybden som skulle maskineres bort var 22mm, som med en kuttddybde på 5.75mm gav 4 lag. Inngrepsmålet ble definert til 8mm. For å fjerne nok materiale i det siste laget ble det tillatt et verktøyoverheng på 13mm relativt til arbeidsstykket i maskineringen.

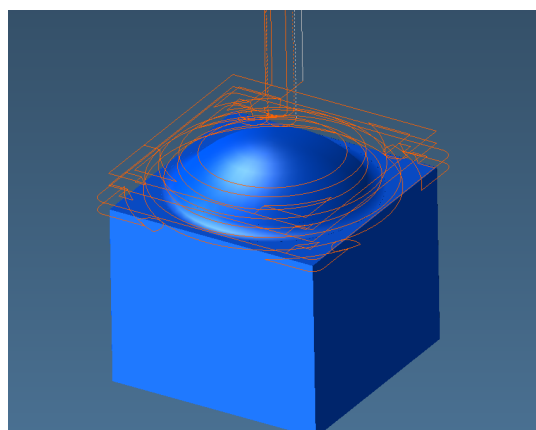
Den andre og siste operasjonen bestod av 5-akse konturfresing med prosessen "5-Axis" og en 20mm kulefres som verktøy. I prosessvinduet ble profilen "General" valgt, og at verktøybanene skulle kalkuleres fra overflater. Mønsteret som ble valgt var "Parallel cuts" og "Zig zag" som metode. Den kuleformede overflaten ble valgt som drivende geometri og verktøyaksen ble definert til å stå normalt på overflaten under bearbeidningen. Bevegelsene før, etter og mellom kuttene ble bestemt i prosessvinduet under fanen "Link" og et sikkerhetsplan ("Clearance plane") ble definert sammen med råemnets dimensjoner, i vinduet navngitt "Documents".

De genererte verktøybanene og simuleringene for operasjonene vises i Figur 5.2 og 5.3.

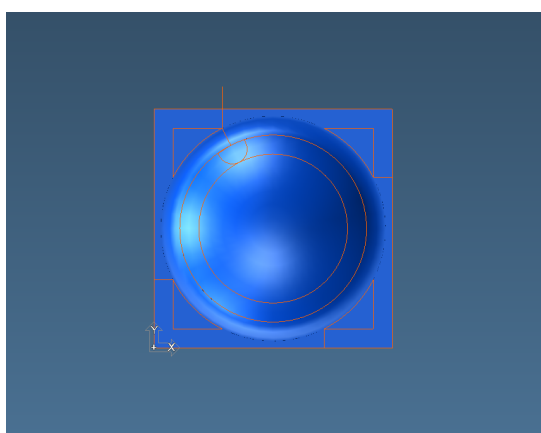
Operasjonene ble deretter postprosessert for å analysere maskinkoden. I den genererte maskinkoden så posisjonene og vinklene korrekte ut. Programmet ble derfor overført og testet på maskinstyringen med tørrkjøring der hele programmet ble kjørt uten arbeidsstykke og en større avstand over bordet. I forbindelse med denne testen gav maskinstyringen en feilmelding på M-koden M92, sammen med koordinatet Z0. I dokumentasjonen til maskinstyringen [16] ble M92 beskrevet som en kode for å kjøre til koordinater i et referanse-koordinatsystem som kunne defineres av maskinbyggeren ("Tool Builder"). Linjene med denne koden i programmet ble brukt til å trekke opp verktøyet i forbindelse med verktøyveksling. Da M91 er en den M-koden som brukes til vanlig for å kjøre denne maskinen med posisjoner i referanse-koordinatsystemet, ble Z0 M92 byttet ut med Z-1 M91. Programmet kjørte da uten feil.



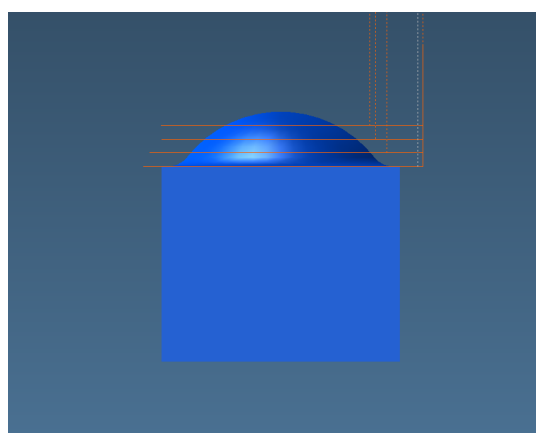
(a) Geometri og råemne



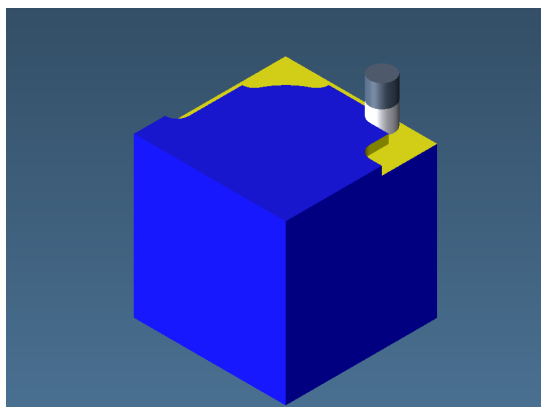
(b) Verktøybaner



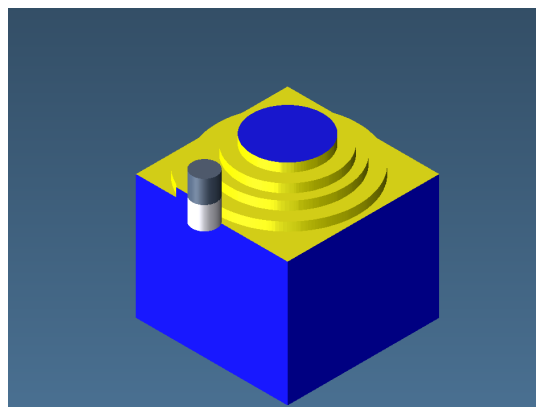
(c) Verktøybaner for det første laget



(d) Verktøybaner sett fra siden

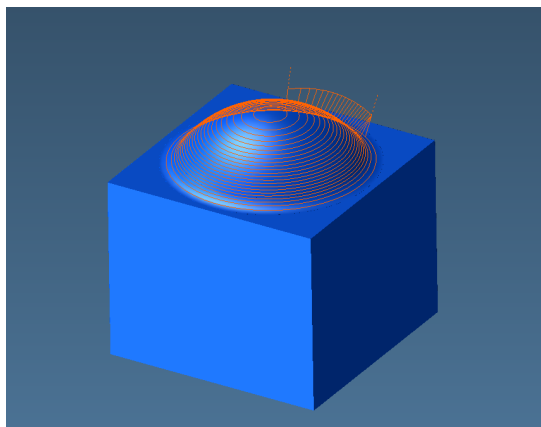


(e) Simulering med materialavvirkning

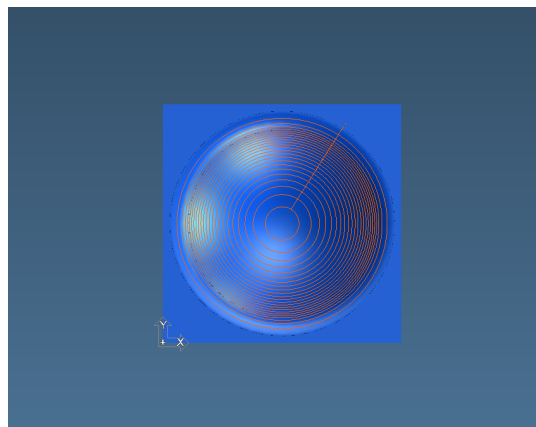


(f) Simulering med materialavvirkning

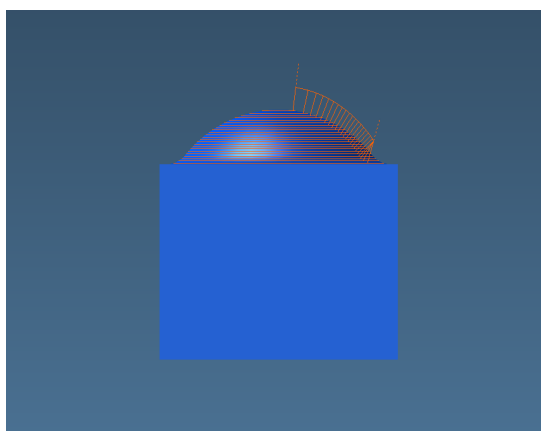
Figur 5.2: Verktøybaner og simulering for operasjon 1 i test 2 i GibbsCam.



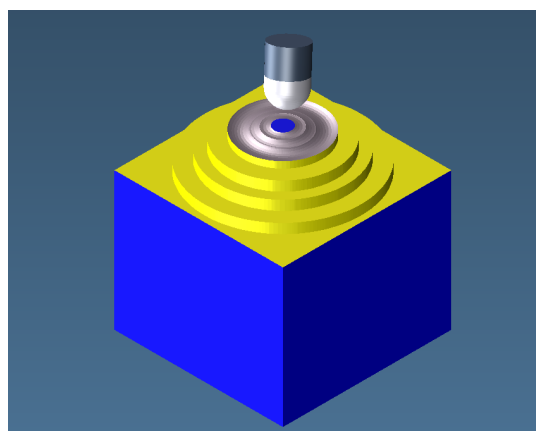
(a) Verktøybaner



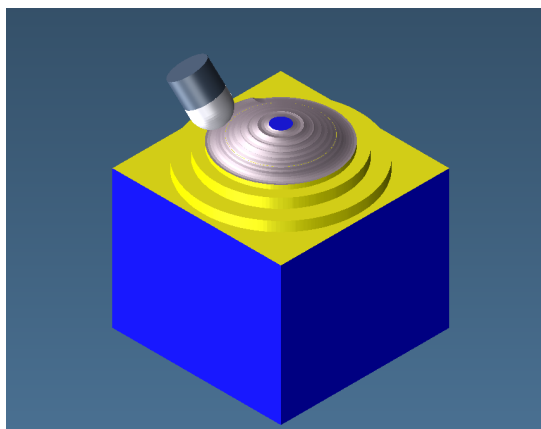
(b) Verktøybaner sett fra ovenfra



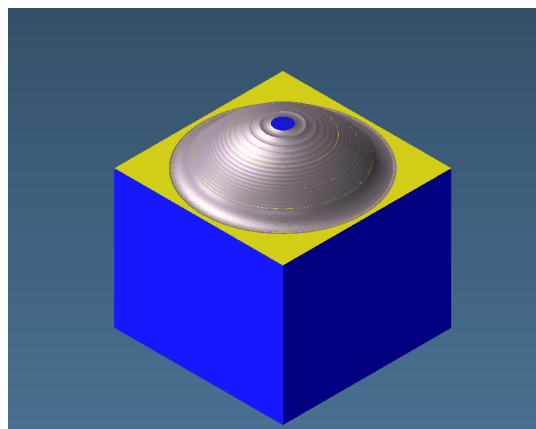
(c) Verktøybaner sett fra siden



(d) Simulering med materialavvirkning



(e) Simulering med materialavvirkning



(f) Simulering av sluttprodukt

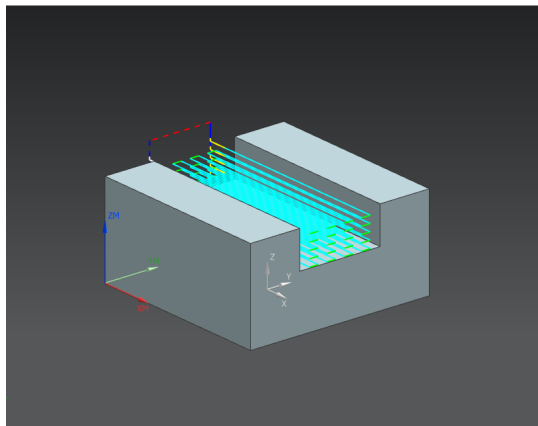
Figur 5.3: Verktøybaner og simuleringer for operasjon 2 i test 2 i GibbsCam.

5.1.2 NX Manufacture

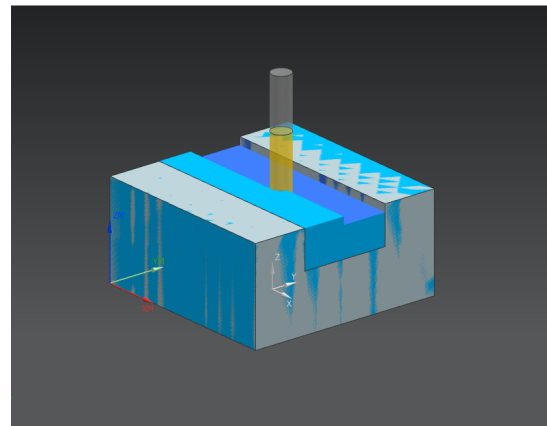
Test 1

Maskineringen av test 1 ble i NX Manufacture lagt inn som en lommefresingsoperasjon ("Cavity Mill") med grovbearbeiding som metode. Bearbeidingen ble planlagt til å utføres med en 20mm pinnefres i en enkelt operasjon, uten gjenstående materiale. Mønsteret som verktøybanene skulle genereres ut fra ble definert til å være sikk-sakk med inngrepsmål på 50% av verktøyets diameter og 8mm kuttdybde. Bunnflaten i sporet ble deretter valgt som drivende geometri. De genererte verktøybanene og simuleringene for denne testen vises i Figur 5.4.

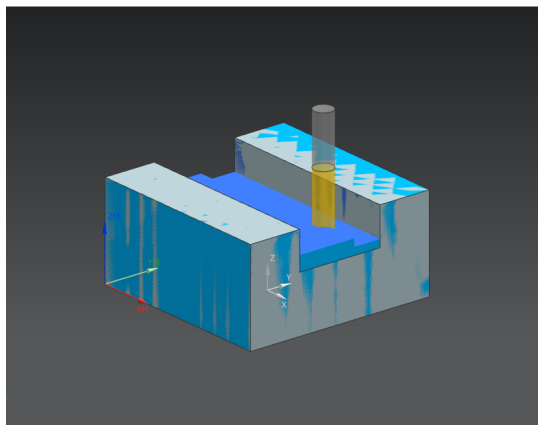
Den planlagte operasjonen ble deretter postprosessert med den utviklede postprosessoren for maskinen, og det ble ikke funnet noen feil. Maskinkoden ble også kontrollert mot maskinkode generert av Sørbys frittstående postprosessor for den samme operasjonen (eksportert fra NX Manufacture i CL-data og importert i postprosessoren).



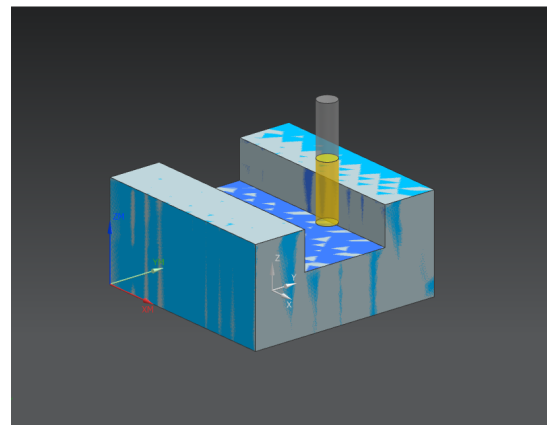
(a) Verktøybaner



(b) Simulering av maskinering



(c) Simulering av maskinering



(d) Simulering av sluttprodukt

Figur 5.4: Verktøybaner og simulering for test 1 i NX Manufacture.

Test 2

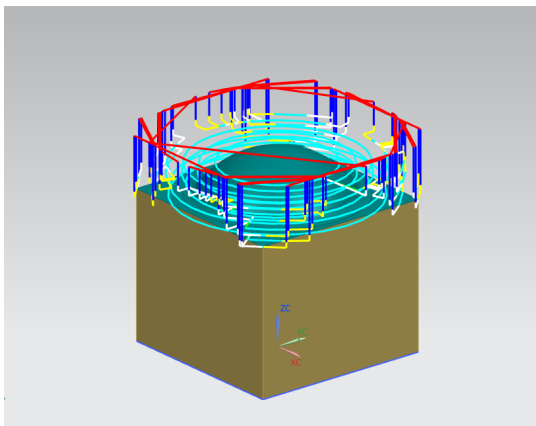
Maskineringen av test 2 ble planlagt med to forskjellige operasjoner i NX Manufacture:

- 3-akse grovfresing (Figur 5.5a)
- 5-akse konturfresing (Figur 5.5b)

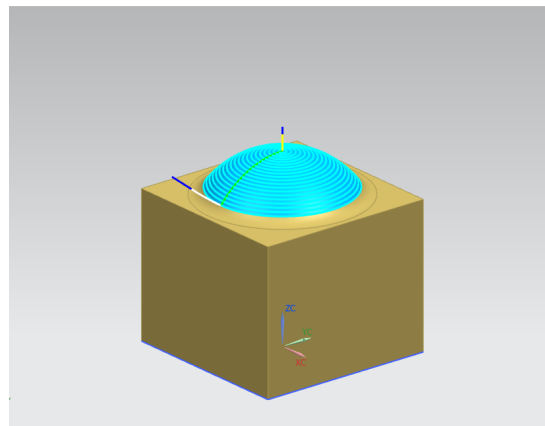
Til grovbearbeidingen ble operasjonstypen "Cavity Mill" valgt, og det ble definert at materialet skulle maskineres bort i fire lag. Både kuleoverflaten og det flate topparealet på modellen ble valgt som drivende geometri og det ble bestemt at verktøybanene skulle følge komponentens overflate. Maksimal kuttdybde og inngrepsmål ble definert til henholdsvis 6mm og 50% av verktøyets diameter.

Den andre og siste operasjonen av maskineringen bestod av 5-akse konturfresing med operasjonstypen "Variable Contour". Kuleoverflaten og radiusen ned mot det flate området på konstruksjonen ble definert som drivende geometri, og sikk-sakk mønster ble valgt som grunnlag for utforming av verktøybanene. En 20mm kulefres ble lagt til som verktøy og det ble definert at verktøyaksen alltid skulle stå normalt på den drivende geometrien under maskineringen. I tillegg til simuleringer av maskineringen ble det utført maskinsimulering for denne testen, som vist i Figur 5.6e og 5.6f. Maskinen som ble brukt i maskinsimuleringen har den samme konfigurasjonen som Deckel Maho 5-aksefresen i laboratoriet, men B-aksen til maskinen har forskjellig rotasjonsretning.

Operasjonene ble postprosessert med den utviklede postprosessoren for maskinen og maskinkoden ble sammenlignet med maskinkode generert med Sørbys postprossessor for de samme operasjonene. Maskinkodene var svært like, og det ble ikke funnet noen feil. Det ble derfor gjort klart for en fysisk maskineringstest.

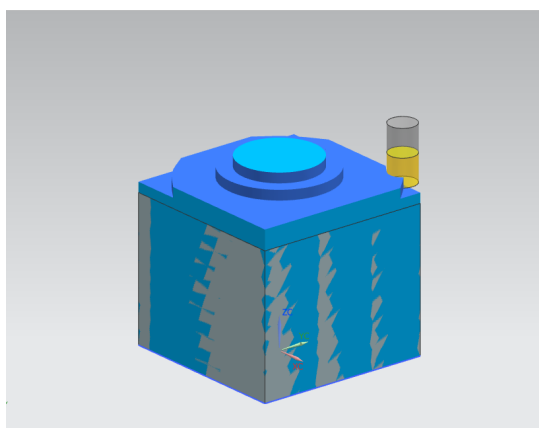


(a) Verktøybaner for 3-akse grovmaskinering

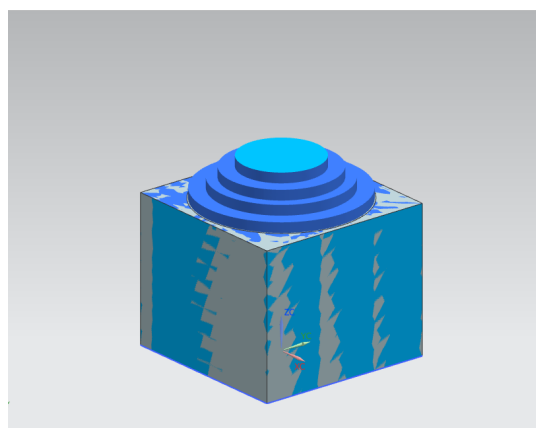


(b) Verktøybaner for 5-akse konturmaskinering

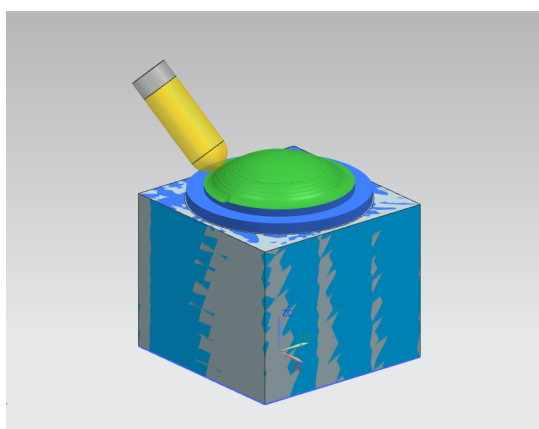
Figur 5.5: Verktøybaner for test 2 i NX Manufacture.



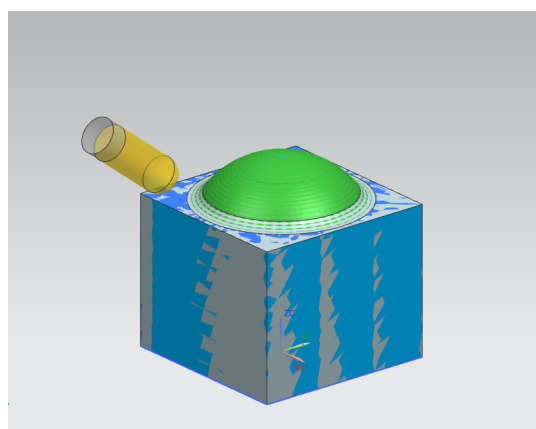
(a) Simulering av grovmaskinering



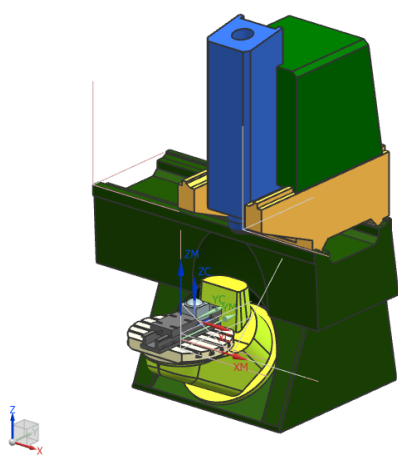
(b) Simulering av grovmaskinering



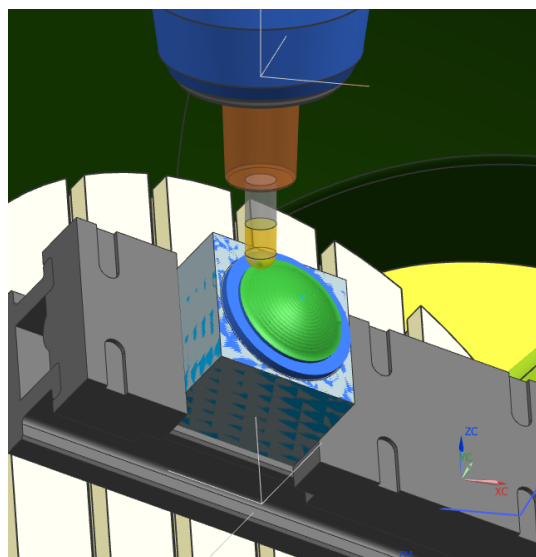
(c) Simulering av konturmaskinering



(d) Simulering av sluttprodukt



(e) Simulering av oppspent arbeidstykke i en lignende 5-akse verktøymaskin



(f) Simulering av 5-akse konturmaskinering i en lignende verktøymaskin

Figur 5.6: Simulering av maskinering for test 2 i NX Manufacture.

5.2 Dreining og 4-aksefresing

5.2.1 GibbsCam

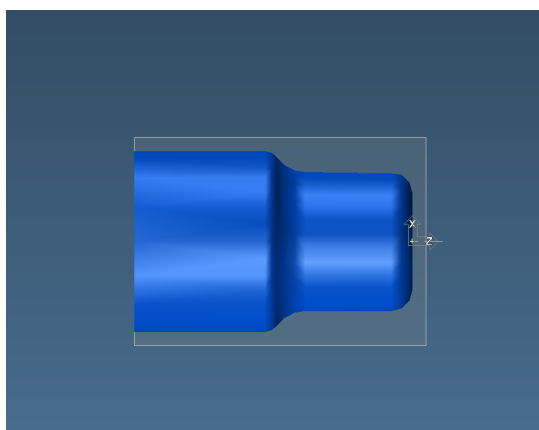
Ved oppgavens start eksisterte det en postprosessor for GibbsCam for denne maskinen. Navnet på postprosessoren bar typebetegnelsen til en annen flerfunksjonsmaskin fra Okuma og postprosessoren hadde aldri blitt testet eller brukt. De ansatte i laboratoriet hadde ingen tidligere erfaring med programmering av flerfunksjonsmaskiner i CAM, og det var derfor ønsket at funksjonaliteten til CAM og den eksisterende postprosessoren skulle testes og eventuelle mangler dokumenteres.

Test 1

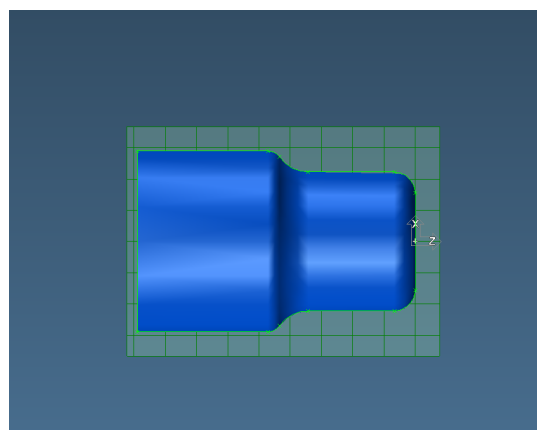
I GibbsCam brukes geometri i form av linjer og punkt til som grunnlag for maskineringsoperasjonene. Grunnet rotasjonssymmetrien i dreining brukes kun geometri i halvparten av snittet, fra senter i modellen og ut i positiv X-retning. Ved hjelp av funksjonen "Extract Geometry" i GibbsCam kan geometrien fra et snitt av modellen hentes ut i form av linjer, men disse linjene vil ligge langs omkretsen til hele snittet, og ikke kun fra senter og ut i modellen. For at verktøybanene for operasjonene på endeflaten av denne testen skulle bli riktige ble det derfor nødvendig å modifisere den ekstraherte geometrien fra modellen før operasjonene ble planlagt. Etter at de nødvendige endringene var utført ble maskineringen planlagt med følgende fire operasjoner i GibbsCam:

- grovdreining av endeflaten (Figur 5.7c)
- grovdreining av sylinderkroppen (Figur 5.7d)
- findreining av endeflaten (Figur 5.7e)
- findreining av sylinderkroppen (Figur 5.7f)

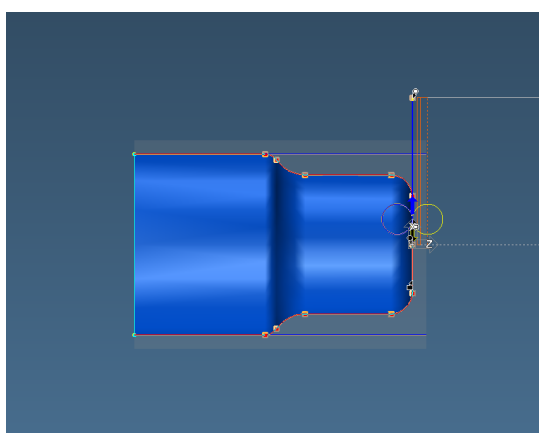
I utgangspunktet ble det forsøkt å bruke én grovdreingsoperasjon for hele komponenten. Dette førte til støtvide verktøybevegelser i X-retning for planplandreiningen, i stedet for langsgående kutt. Begge operasjonene ble derfor splittet til bearbeiding på komponentens endeflate og bearbeiding langs sylinderkroppen. I grovdreingsoperasjonene ble det definert en maksimal kuttdybde på 1mm og 0.5mm gjenstående materiale til finbearbeidingen. I Figur 5.7 og 5.8 vises verktøybanene og simuleringene for operasjonene.



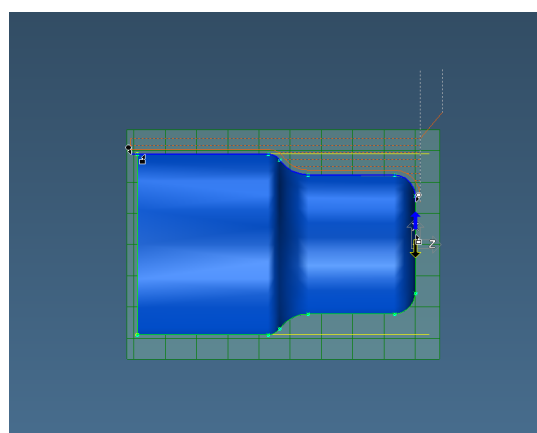
(a) Geometri og råemne



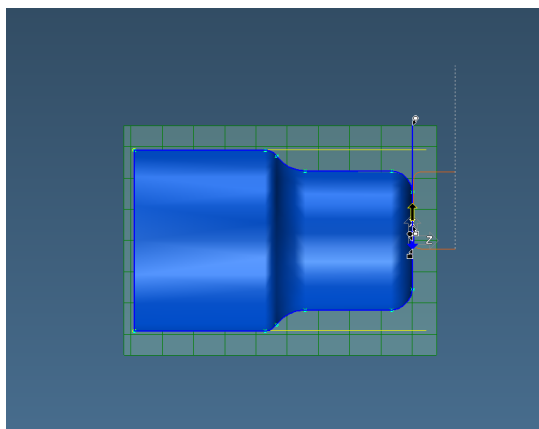
(b) Ekstraherte kurver fra modellen



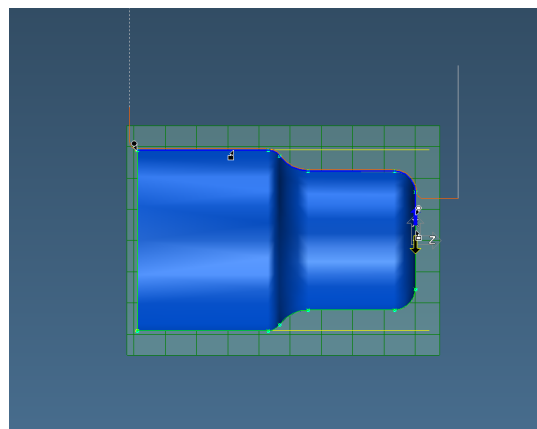
(c) Verktøybaner for plandreiging



(d) Verktøybaner for grovdreiging

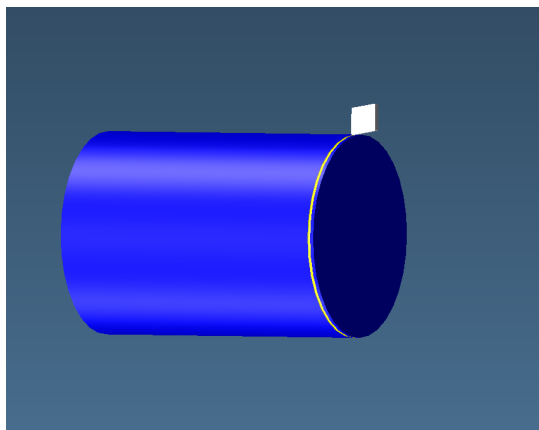


(e) Verktøybaner for findreiging av endeflate

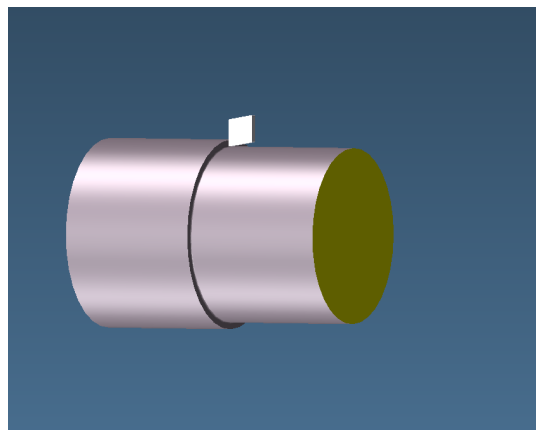


(f) Verktøybaner for findreiging

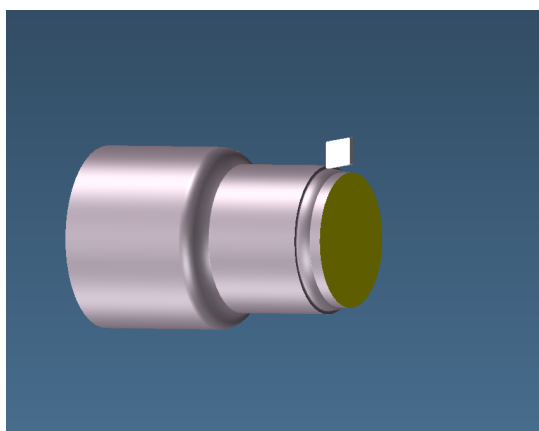
Figur 5.7: Verktøybaner for testmodell 1 i GibbsCam.



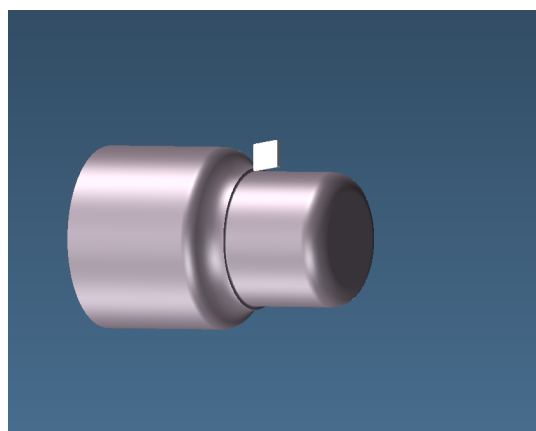
(a) Simulering av plandreieing



(b) Simulering av grovdreieing



(c) Simulering av grovdreieing



(d) Simulering av findreieing

Figur 5.8: Simulering for testmodell 1 i GibbsCam.

Testen ble deretter postprosessert for å analysere maskinkoden for operasjonene. Feilene og manglene som ble funnet, og de tilhørende løsningene, er presentert i Tabell 5.1:

Problem	Løsning
Manglende kommando for definerings av nullpunkt	Redigering av maskinkode
Feil posisjon for verktøyveksling	Redigering av maskinkode eller MDD
Bevegelse før det første verktøyet kalles på	Redigering av maskinkode
Ikke støtte for radiuskompensering	Unngå bruk av radiuskompensering i CAM
Bruk av inkompatibel M-kode (M42)	Redigering av maskinkode

Tabell 5.1: Mangler funnet ved postprocessor til GibbsCam for Okuma Genos etter test 1

Manglende kommando for setting av nullpunkt

Når de ansatte i laboratoriet programmerer flerfunksjonsmaskinen på maskinstyringen brukes en makro for å definere nullpunktet med bakkens dybde og emnets utstikk som input-variabler. Det anses ikke som en feil at denne makroen ikke var inkludert i maskinkoden som ble generert av postprosessen, men ideelt sett skulle den ha vært inkludert. For å definere riktig nullpunkt bør derfor kallenavnet til makroen og de tilhørende variablene legges inn manuelt i starten av programmene som genereres.

Feil posisjon for verktøyvekslerposisjon

Dersom verktøyvekslerposisjonen ikke defineres i menyen "Documents" i GibbsCam vil posisjonen som er definert i MDD-filen brukes. Da ingen MDD-fil fulgte med postprosessen for denne maskinen ble MDD-filen "C-Axis Horizontal Lathe - Generic Shank", som var tilgjengelig i GibbsCam, brukt som utgangspunkt. For å få riktig verktøyvekslerposisjon må hjemposisjonen i MDD-filen (vist i Figur 5.9) endres til den posisjonen som brukes til verktøyveksling i maskinstyringen. En alternativ løsning til å definere verktøyvekslerposisjon i GibbsCam eller endre MDD-filen, er å benytte en makro som vanligvis brukes når maskinen programmeres på maskinstyringen. Når makroen kalles opp i programmet kjører maskinen automatisk til riktig verktøyvekslerposisjon. For å bruke makroen må den legges inn manuelt i maskinkoden.

Bevegelse før det første verktøyet kalles på

I starten av den genererte maskinkoden ble en bevegelse kommandert før det første verktøyet ble kalt på. Dette ble ikke akseptert av maskinstyringen da programmet ble forsøkt simulert. Det er mulig at en bevegelse i denne delen av programmet vil bli akseptert dersom variabelen i MDD-filen som har innvirkning på denne kommandoen endres til verktøyvekslerposisjonen. En enklere løsning er å erstatte denne linjen i programmet med makroen for verktøyvekslerposisjon.

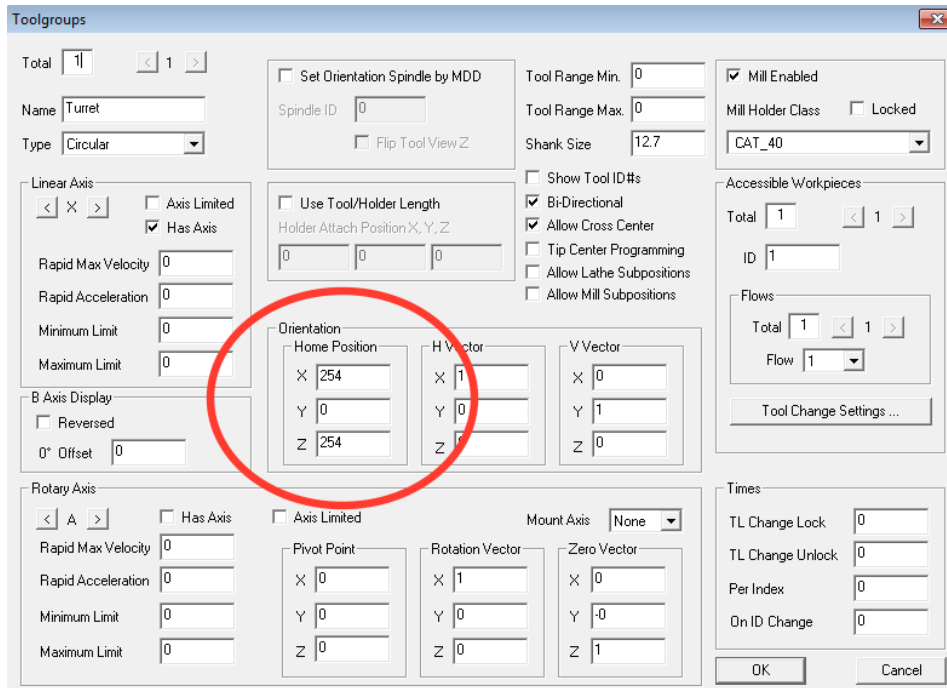
Ikke støtte for radiuskompensering

Maskinkoden for operasjonene i denne testen ble postprosessert både med og uten radiuskompensering/verktøykompensering for å teste funksjonaliteten for dette i postprosessen. I maskinkoden med radiuskompensering ble de riktige G-kodene skrevet ut (G41/G42) men posisjonene og radiusene i maskinkoden forble uendret sammenlignet med maskinkoden som ble generert uten radiuskompensering. Ut fra disse resultatene ser det ut til at denne postprosessen ikke har støtte for bruk av radiuskompensering, og at radiuskompensering derfor bør deaktiveres i GibbsCam når denne postprosessen skal benyttes. Figur 5.10 viser maskinkoden for den siste operasjonen i test 1 med og uten radiuskompensering aktivert i GibbsCam.

Bruk av inkompatibel M-kode (M42)

I maskinkoden brukes M-koden M42 for å definere hvilket gir maskinen skal bruke. Denne verktøymaskinen bruker ikke gir og aksepterer derfor ikke denne M-koden. Foruten å få leverandøren til å endre postprosessen er den eneste løsningen på dette problemet å fjerne M-koden manuelt fra maskinkoden.

Til slutt ble de nevnte feilene og manglene rettet manuelt i maskinkoden og en vellykket simulering ble gjennomført på maskinstyringen.



Figur 5.9: Hjem-posisjon i MDD-fil

N105 G96 S100 M3	N105 G96 S100 M3
N106 G0 G90 G95 T010101 M42 M8	N106 G0 G90 G95 T010101 M42 M8
N107 G0 Z6.	N107 G0 Z6.
N108 X50.	N108 X50.
N109 G18	N109 G18
N110 G1 X11.6 F.2	N110 G1 X11.6 F.2
N111 G42 Z1.	N111 Z1.
N112 G2 X13.6 Z0. L1.	N112 G2 X13.6 Z0. L1.
N113 G3 X20. Z-3.2 L3.2	N113 G3 X20. Z-3.2 L3.2
N114 G1 Z-15.617	N114 G1 Z-15.617
N115 G2 X24.114 Z-19.557 L4.8	N115 G2 X24.114 Z-19.557 L4.8
N116 G3 X26. Z-21.362 L2.2	N116 G3 X26. Z-21.362 L2.2
N117 G1 Z-40.345	N117 G1 Z-40.345
N118 G2 X28. Z-41.345 L1.	N118 G2 X28. Z-41.345 L1.
N119 G40 G1 X38.	N119 G1 X38.
N120 G0 X50.	N120 G0 X50.
N121 M5	N121 M5
N122 M9	N122 M9
N123 G0 X0. Z0.	N123 G0 X0. Z0.
N124 M2	N124 M2

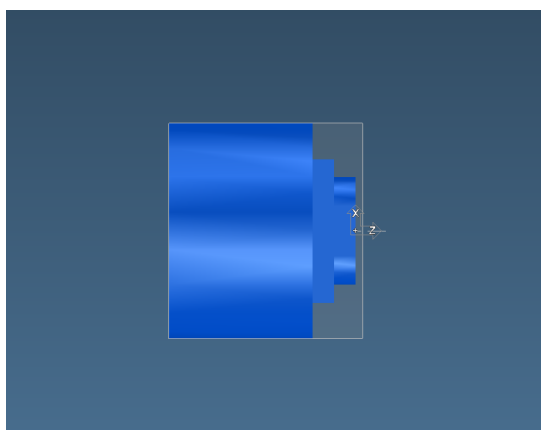
Figur 5.10: Utsnitt av maskinkode for dreining med (t.v.) og uten (t.h.) radiuskompensering.

Test 2

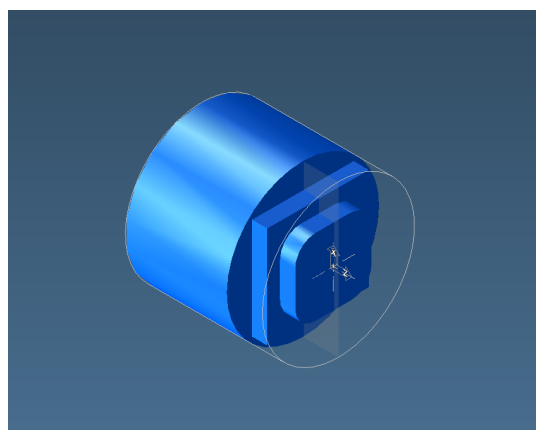
Maskineringen av testmodell 2 i XY-planet ble planlagt med to forskjellige prosesser og to forskjellige verktøy som utgjorde følgende operasjoner i GibbsCam:

- plandreiing av endeflate (Figur 5.11c)
- konturfresing av endeflate(Figur 5.11f)

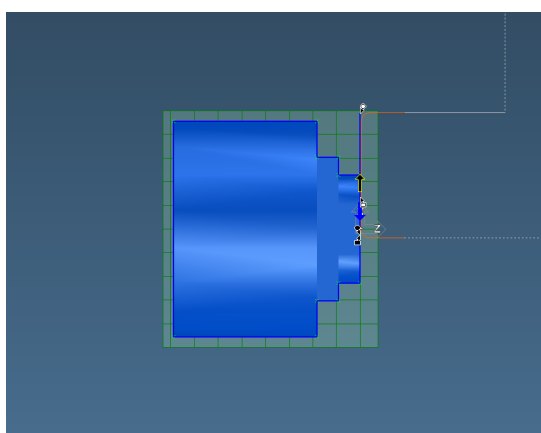
Den første operasjonen bestod av plandreiing av endeflaten i ett enkelt kutt med prosessen "Contour". Til den neste operasjonen ble det lagt inn et verktøybytte fra dreieverktøyet i den første operasjonen til en 8mm pinnefres for den neste operasjonen. Prosessen "Contour" for fresing ble definert og flatene som skulle maskineres på enden av modellen ble merket som drivende geometri. En kuttdybde på 3mm ble definert, som ville gi to lag med verktøybaner i dybden. For å dele maskineringen opp i to baner med en radiell forskyvning, ble det lagt inn én "Offset" med 2mm "Stepover". I Figur 5.11 vises de genererte verktøybanene for operasjonene og i Figur 5.12 vises simuleringene.



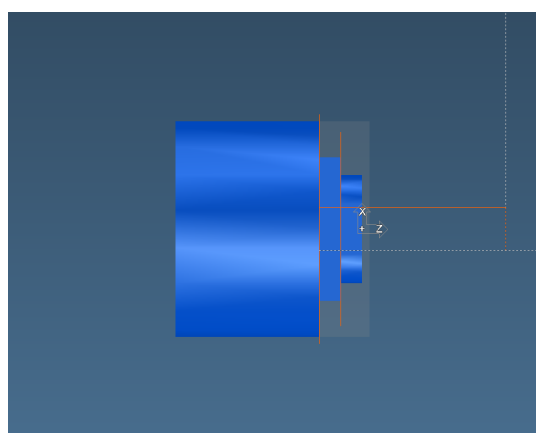
(a) Geometri og råemne



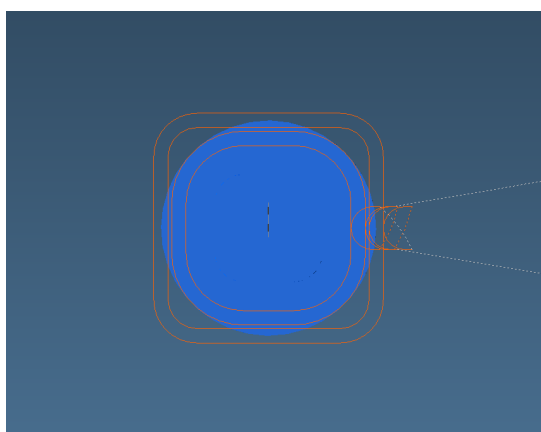
(b) Geometri og råemne



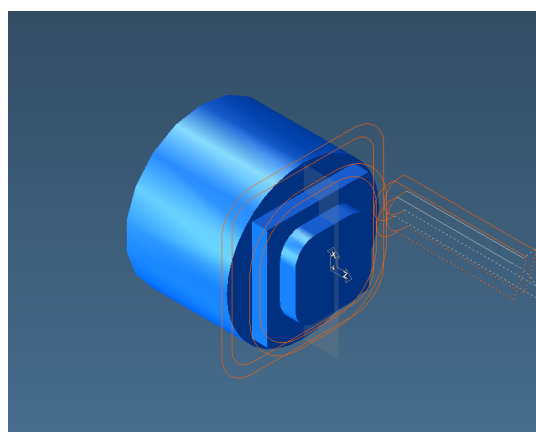
(c) Verktøybaner for plandreing



(d) Verktøybaner for fresing i XY-planet sett fra siden

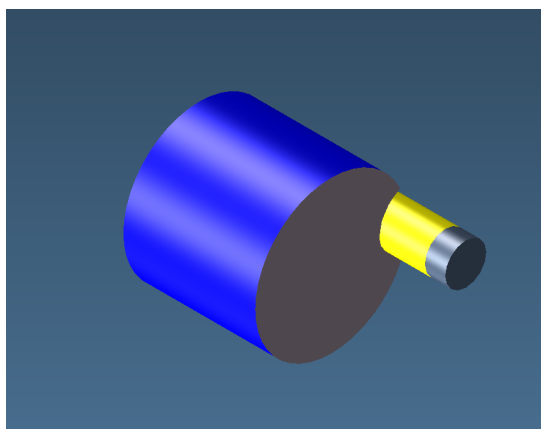


(e) Verktøybaner for fresing i XY-planet sett forfra

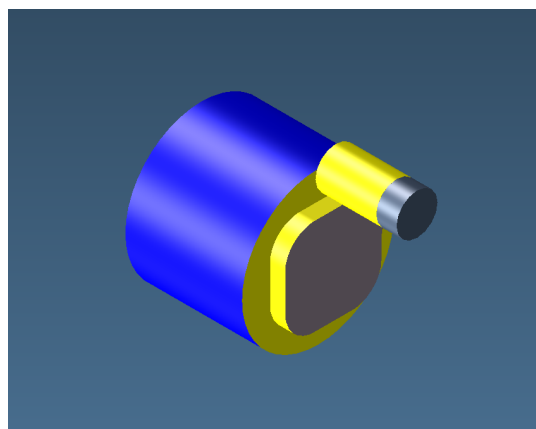


(f) Verktøybaner for fresing i XY-planet

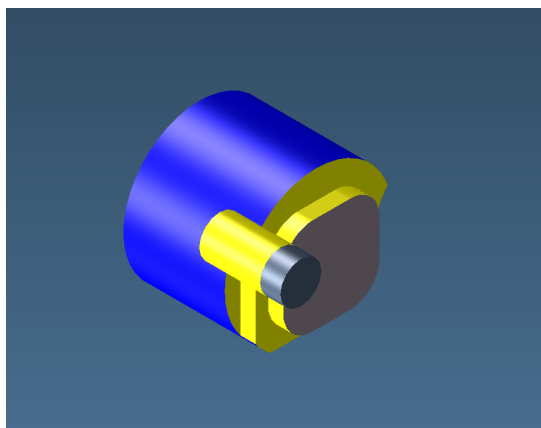
Figur 5.11: Verktøybaner for test 2 i GibbsCam.



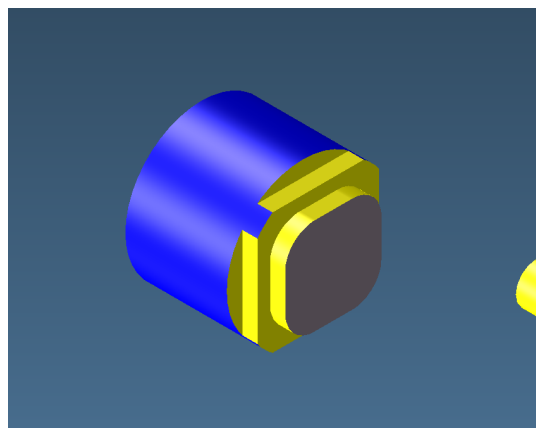
(a) Simulering av freseoperasjon i XY-planet



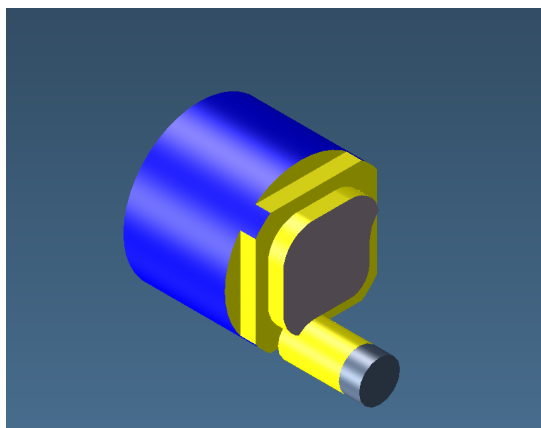
(b) Simulering av freseoperasjon i XY-planet



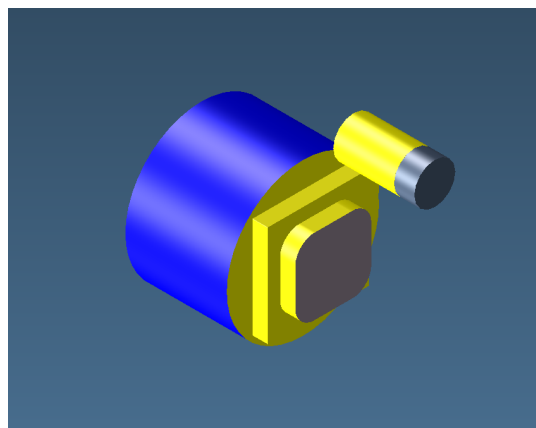
(c) Simulering av freseoperasjon i XY-planet



(d) Simulering av freseoperasjon i XY-planet



(e) Simulering av freseoperasjon i XY-planet



(f) Simulering av freseoperasjon i XY-planet

Figur 5.12: Simulering for test 2 i GibbsCam.

De planlagte operasjonene for test 2 ble deretter postprosessert. Feil og mangler som ble funnet ved postprosessoren fra denne testen er oppsummert i Tabell 5.2:

Problem	Løsning
Posisjoner utenfor X-aksens arbeidsområde	Bruke C-akse i stedet for Y-akse
Ikke støtte for radiuskompensering	Unngå bruk av radiuskompensering i CAM

Tabell 5.2: Mangler funnet ved postprossessor til GibbsCam for Okuma Genos etter test 2

Posisjoner utenfor X-aksens arbeidsområde

Da programmet ble testet med tørrkjøring / luftkjøring i verktøymaskinen oppstod det en feilmelding om at den ene posisjonen som verktøyet skulle gå til var utenfor X-aksens arbeidsområde. Kodelinjen som gav feilmeldingen i programmet inneholdt koordinatet Y-16.5, og på dette stedet i programmet var den gjeldende posisjonen i X 6.

Mest sannsynlig er årsaken til denne feilen at posisjonering av den fysiske Y-aksen til dette koordinatet ville kreve en større bevegelse av X-aksen enn det som var mulig innenfor X-aksens arbeidsområde, grunnet den lille vinkelen mellom X- og Y-aksen i maskinens konfigurasjon. En robust løsning for dette problemet er å bruke C-aksen i stedet for å bruke den fysiske Y-aksen. Med aktiv C-akse vil X-aksen utelukkende kunne jobbe fra nullpunktet og i positiv retning, der aksen har et stort arbeidsrom. Ved hjelp av interpolering vil alle bevegelsene som X- og Y-akse kan utføre også være mulig å utføre med X- og C-akse.

For å aktivere C-aksen til maskinering i XY-planet velges "Polar & Cylindrical Milling" i prosessvinduet i GibbsCam under fanen "Rotate".

Ikke støtte for radiuskompensering

Maskinkoden for operasjonene i denne testen ble postprosessert både med og uten radiuskompensering/verktøykompensering for å teste funksjonaliteten for dette i XY-planet i postprosessoren. I maskinkoden med radiuskompensering ble de riktige G-kodene skrevet ut (G41/G42) men posisjonene og radiusene i maskinkoden forble uendret sammenlignet med maskinkoden som ble generert uten radiuskompensering. Ut fra disse resultatene ser det ut til at denne postprosessoren ikke har støtte for bruk av radiuskompensering i XY-planet, i likhet med den manglende støtten for radiuskompensering i XZ-planet funnet i test 1. Radiuskompensering bør derfor også deaktiveres i GibbsCam når denne postprosessoren skal benyttes for operasjoner i XY-planet. Figur 5.13 viser et utsnitt av maskinkoden for freseoperasjonen med og uten radiuskompensering aktivert i GibbsCam.

N34 G1 Z-3. F100.	N34 G1 Z-3. F100.
N35 G41 Y-16.5	N35 Y-16.5
N36 G3 X0. Y-13.5 L3.	N36 G3 X0. Y-13.5 L3.
N37 G1 X-3.5 F203.	N37 G1 X-3.5 F203.
N38 G2 X-13.5 Y-3.5 L10.	N38 G2 X-13.5 Y-3.5 L10.
N39 G1 Y3.5	N39 G1 Y3.5
N40 G2 X-3.5 Y13.5 L10.	N40 G2 X-3.5 Y13.5 L10.
N41 G1 X3.5	N41 G1 X3.5
N42 G2 X13.5 Y3.5 L10.	N42 G2 X13.5 Y3.5 L10.
N43 G1 Y-3.5	N43 G1 Y-3.5
N44 G2 X3.5 Y-13.5 L10.	N44 G2 X3.5 Y-13.5 L10.
N45 G1 X0.	N45 G1 X0.
N46 G3 X-3. Y-16.5 L3.	N46 G3 X-3. Y-16.5 L3.
N47 G40 G1 Y-17.5	N47 G1 Y-17.5
N48 G0 Z20.	N48 G0 Z20.
N49 X3. Y-20.	N49 X3. Y-20.
N50 G1 Z-6. F100.	N50 G1 Z-6. F100.
N51 G41 Y-19.	N51 Y-19.

Figur 5.13: Utsnitt av maskinkode for fresing med (t.v.) og uten (t.h.) radiuskompensering.

Test 3

Maskineringen av testmodell 3 i YZ-planet ble planlagt med følgende to operasjoner i Gibbs-Cam:

- fresing av spor (Figur 5.14b, 5.14c)
- fresing av lomme (Figur 5.14d, 5.14e)

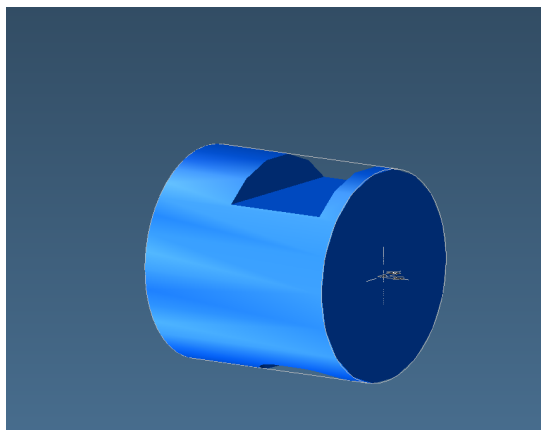
Den første operasjonen i denne testen skulle frese sporet på den ene siden av arbeidsstykket. Til denne operasjonen ble freseprosessen "Surfacing" og metoden "Lace Cut" valgt, med bunnflaten i sporet som den drivende geometrien. For at verktøybanene skulle gå på tvers av cylinderen i stedet for på langs ble kuttvinkelen ("Cut Angle") definert til 90° . Kuttddybden ble definert til 5mm, som gav to lag med verktøybaner i dybden.

Den andre og siste operasjonen som ble planlagt var lommefresing med prosessen "Roughing" og "Zig Zag" som mønster for verktøybanene. Det ble definert at maskineringen skulle utføres i to lag med kuttddybde 2.5mm og at inngrepsmålet skulle være 3mm. De genererte verktøybanene og simuleringene for operasjonene vises i Figur 5.14 og Figur 5.15.

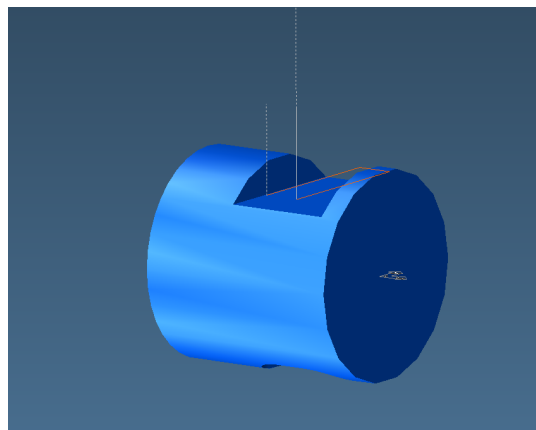
I GibbsCam ble det valgt at den fysiske Y-aksen skulle brukes i begge operasjonene. Når maskinering utføres i YZ-planet jobber X-aksen nesten uten unntak fra nullpunktet og ut i positiv X-retning. I dette området har X-aksen et stort arbeidsrom, som gir gode forhold for å bruke Y-aksen. I motsetning til maskinering i XY-planet er ikke C-aksen en fullgod erstatning for den fysiske Y-aksen når maskinering utføres i YZ-planet.

De planlagte operasjonene ble postprosessert og det ble ikke funnet noen nye feil ved postprosessoren fra denne testen. Uredigert og redigert maskinkode for denne testen finnes i Vedlegg E som et eksempel på hvordan maskinkode generert i denne postprosessoren ser ut med og uten de nødvendige endringene.

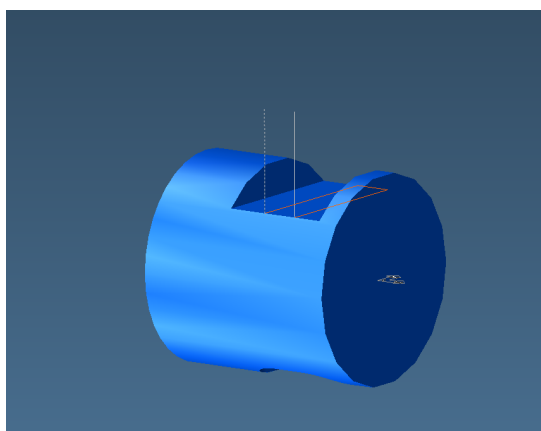
Den genererte maskinkoden ble kjørt i simuleringsmodus på maskinstyringen uten feil, etter at de kjente feilene som ble funnet tidligere i test 1 og test 2 ble rettet.



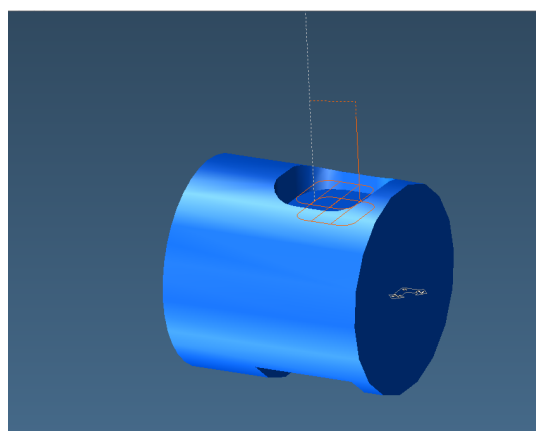
(a) Geometri og r emne



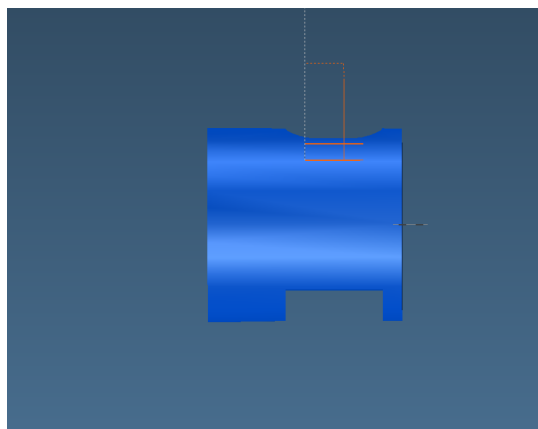
(b) Verkt ybaner for fresing av spor



(c) Verkt ybaner for fresing av spor

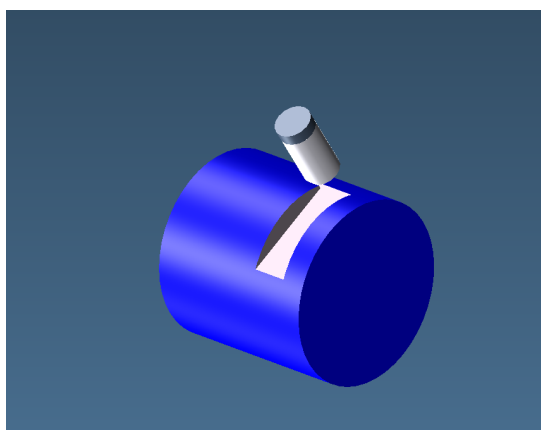


(d) Verkt ybaner for fresing av lomme

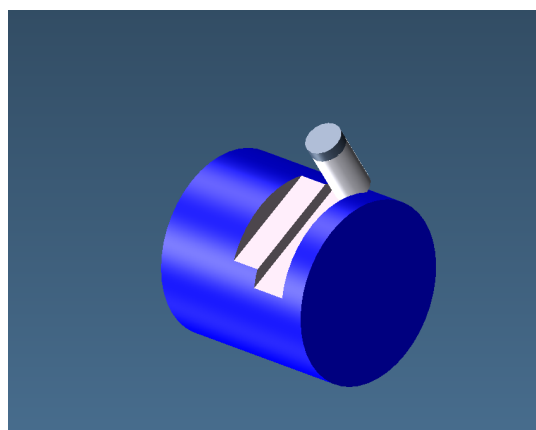


(e) Verkt ybaner for fresing av lomme

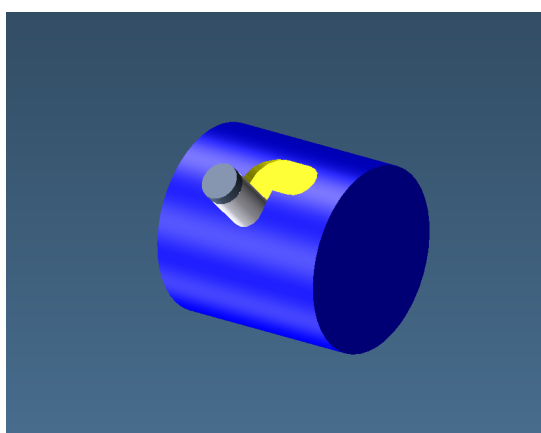
Figur 5.14: Verkt ybaner for test 3 i GibbsCam.



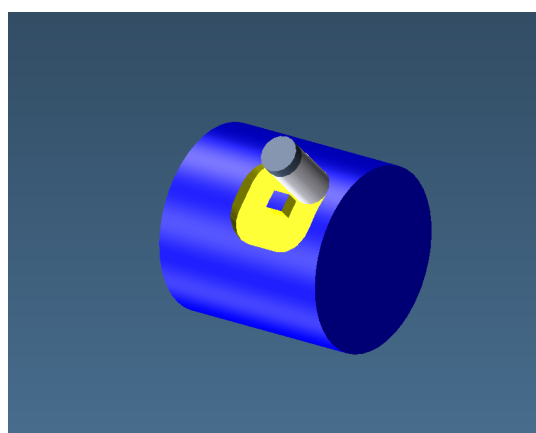
(a) Simulering av fresing av spor



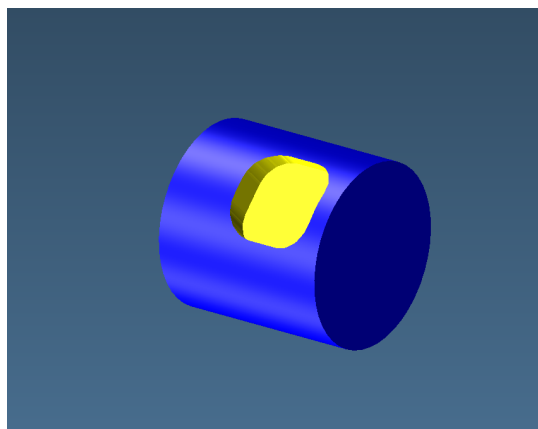
(b) Simulering av fresing av spor



(c) Simulering av fresing av lomme



(d) Simulering av fresing av lomme



(e) Simulering av sluttprodukt

Figur 5.15: Simulering for test 3 i GibbsCam.

Test 4

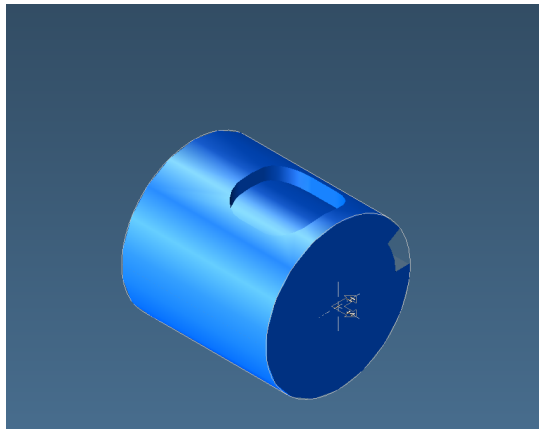
Maskineringen av testmodell 4 ble planlagt med følgende to operasjoner i GibbsCam:

- radiell lommefresing (Figur 5.16b)
- radiell konturfresing (Figur 5.16e)

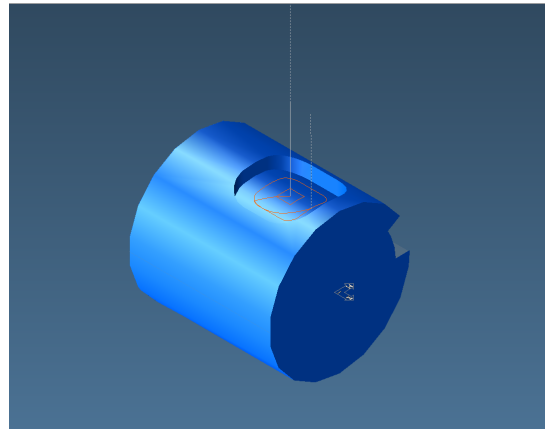
Den første operasjonen i test 4 var radiell lommefresing og bestod av prosessen "Roughing" med "Offset With Cleanup" som mønster for maskineringen. Bunnflaten i modellen ble valgt som drivende geometri og det ble definert at maskineringen skulle utføres i ett lag med 2.5mm kutt-dybde og 3mm inngrepsmål. Siden bunnflaten i testmodellen var buet ble "Polar & Cylindrical Milling" valgt under fanen "Rotate" i prosessvinduet for å bruke C-aksen under maskineringen.

Den andre og siste operasjonen i test 4 var radiell konturfresing langs arbeidsstykkets side. For å kunne definere denne operasjonen i GibbsCam måtte den planlegges som en 5-akse maskineringsprosess med den femte akselen (som ikke eksisterer i denne maskinen) låst i posisjonen 0°. I prosessvinduet ble det definert at mønsteret til verktøybanene skulle styres av sideveggene i sporet og at det skulle gjøres ett enkelt kutt. Det ble også definert at verktøyaksen alltid skulle stå vinkelrett på overflaten som ble maskinert. Verktøybanene og simuleringene for operasjonene vises i Figur 5.16 og Figur 5.17.

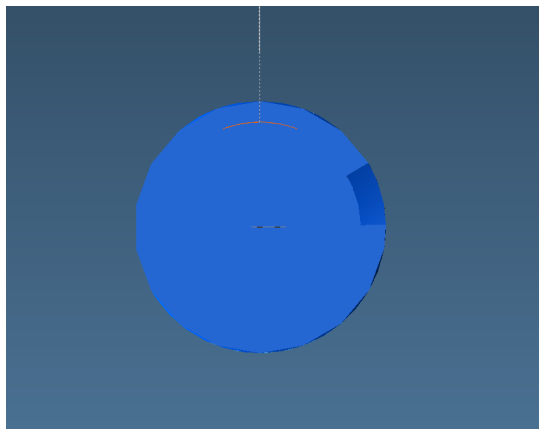
Etter postprosessering ble den genererte maskinkoden kontrollert uten at det ble funnet andre feil enn de som ble funnet i de tidligere testene. Maskinkoden ble rettet for disse feilene og simulert med vellykket resultat. Da dette var den mest avanserte testen, og den eneste som inneholdt simultan 4-akse maskinering, ble denne testen klargjort for en fysisk maskineringstest.



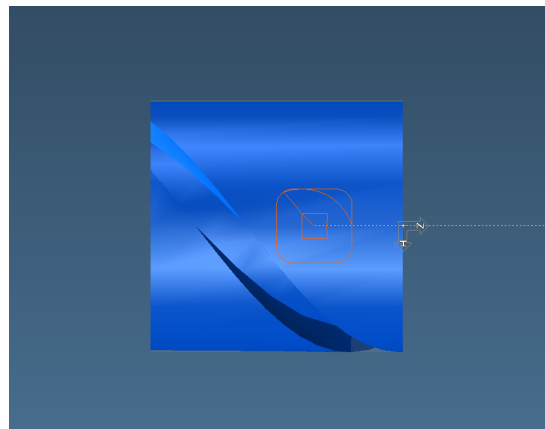
(a) Geometri og råemne



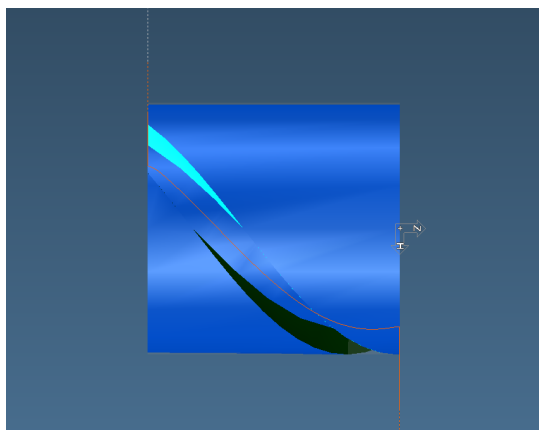
(b) Verktøybaner for fresing av lomme



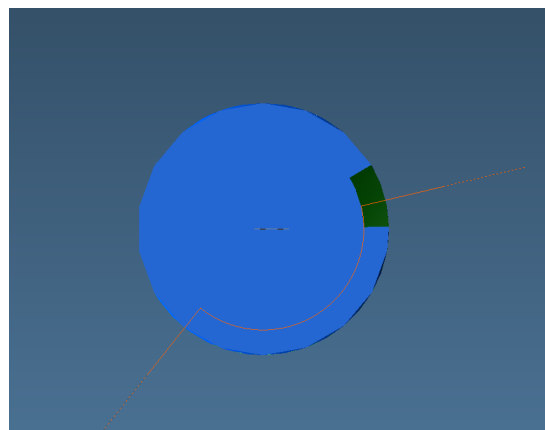
(c) Verktøybaner for fresing av lomme sett fra siden



(d) Verktøybaner for fresing av lomme sett ovenfra

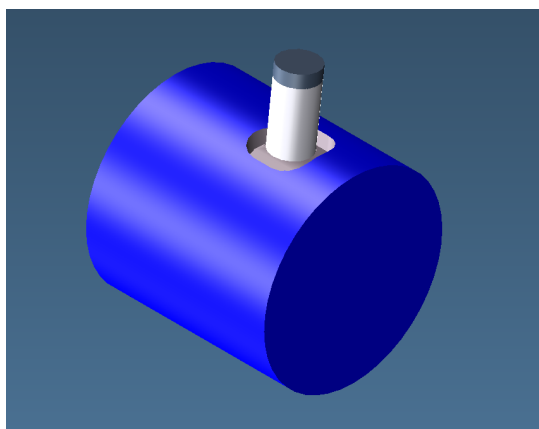


(e) Verktøybaner for fresing av spor

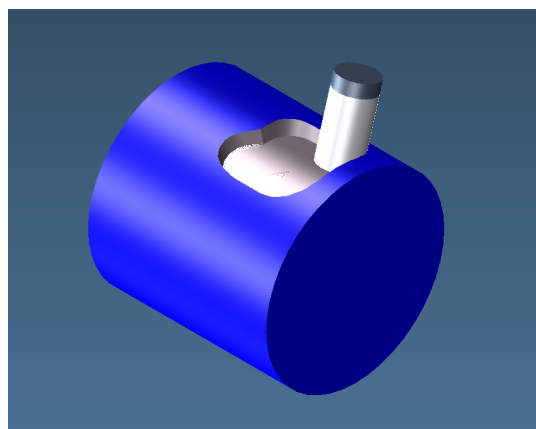


(f) Verktøybaner for fresing av spor sett fra siden

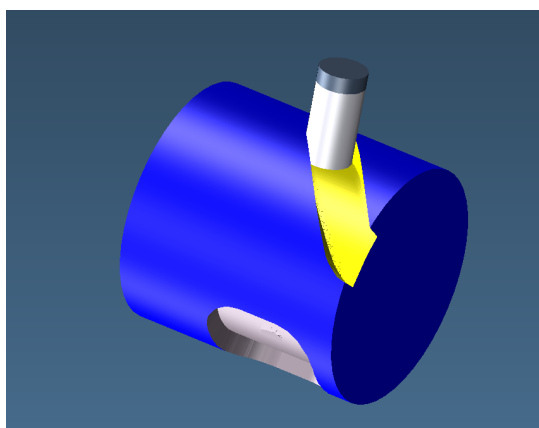
Figur 5.16: Verktøybaner for test 4 i GibbsCam.



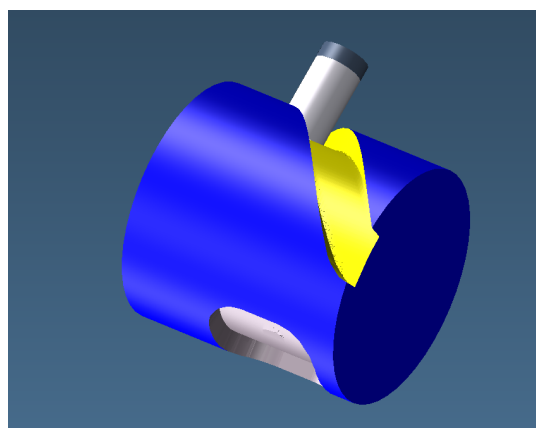
(a) Simulering av lommesfresing



(b) Simulering av lommesfresing



(c) Simulering av radiell konturfresing



(d) Simulering av radiell konturfresing

Figur 5.17: Simulering for test 4 i GibbsCam.

5.2.2 NX Manufacture

Test 1

Maskineringen av testmodell 1 ble planlagt med følgende tre operasjoner i NX:

- grovdreining av endeflaten (Figur 5.18b)
- grovdreining av sylinderkroppen (Figur 5.19a)
- findreining av hele komponenten (Figur 5.19b)

Den første operasjonen bestod av grovdreining "Facing" av endeflaten på arbeidsstykket. Råemnet ble definert til å være 2mm lengre enn modellen og grovdreiningen ble definert til å la 0.5mm av materialet stå igjen til finbearbeidingen. Maksimal kuttdybde ble bestemt til 1mm, som gav 2 kutt i denne operasjonen.

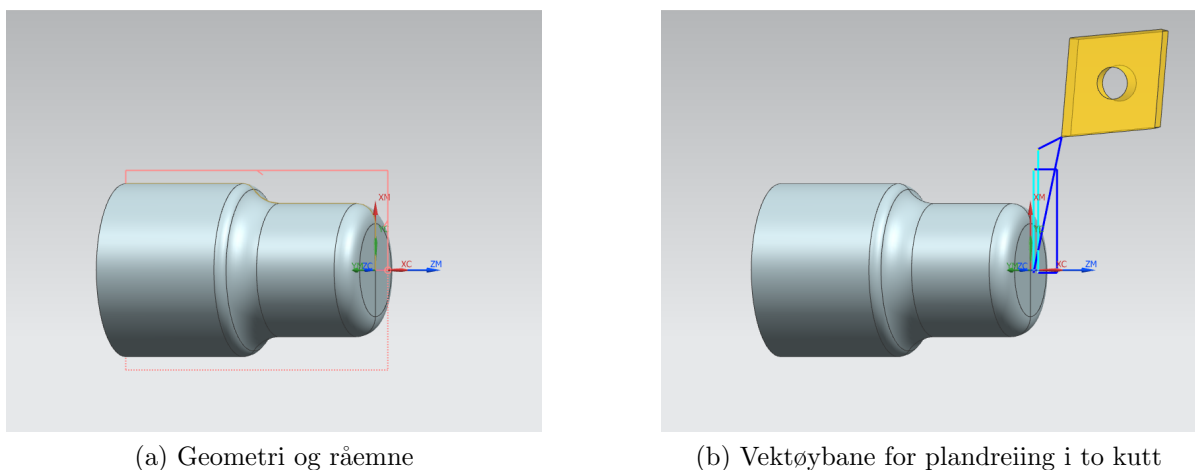
I den andre operasjonen ble grovdreining av sylinderkroppen med operasjonstypen "Rough Turn OD" valgt med det samme verktøy som i den første operasjonen. En kuttdybde på 1mm ble definert, og 0.5mm som gjenstående materiale.

Den siste operasjonen bestod av findreining "Finish Turn OD" av hele komponenten i ett kutt. Et nytt verktøy med mindre neseradius ble valgt for denne operasjonen for å teste funksjonaliteten for verktøybytte i postprosessen.

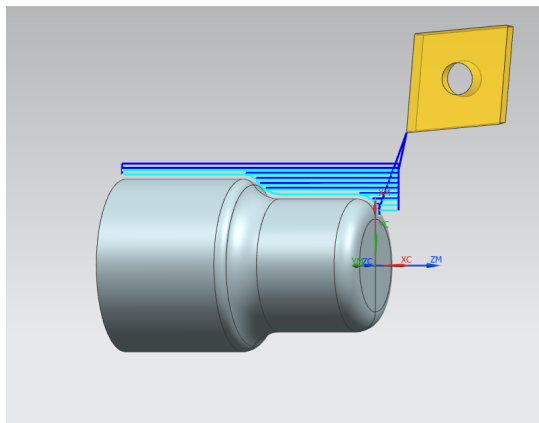
De genererte verktøybanene og simuleringene for operasjonene vises i Figur 5.18 og Figur 5.19.

Operasjonene ble deretter merket med teksten "TURN" i UDE-en "HEAD" i NX Manufacture, i henhold til det forhåndsdefinerte kallenavnet for dreining gitt i Tabell 3.1 i Kapittel 3.

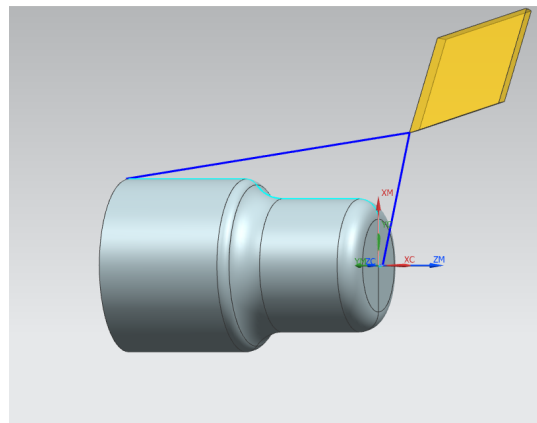
Den genererte maskinkoden inneholdt ingen feil, og postprosessen for dreining ble hentet inn på korrekt vis av master-postprosessen. Maskinkoden ble også simulert på maskinstyringen uten feil.



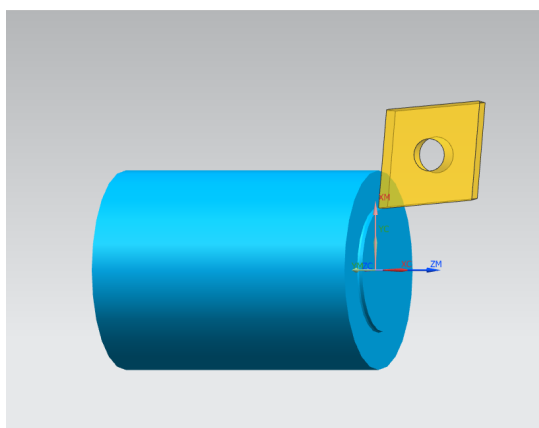
Figur 5.18: Verktøybaner for test 1 i NX Manufacture



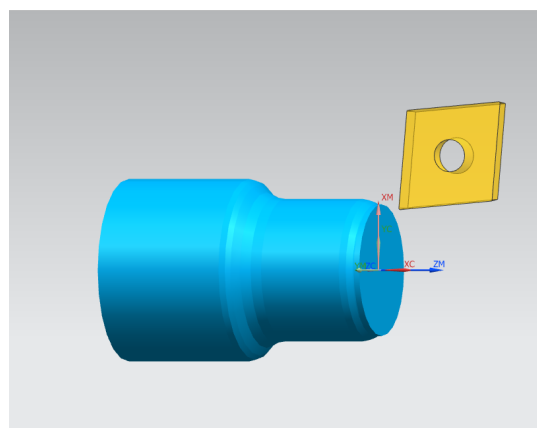
(a) Verktøybane for grovdreining av sylinderkropp



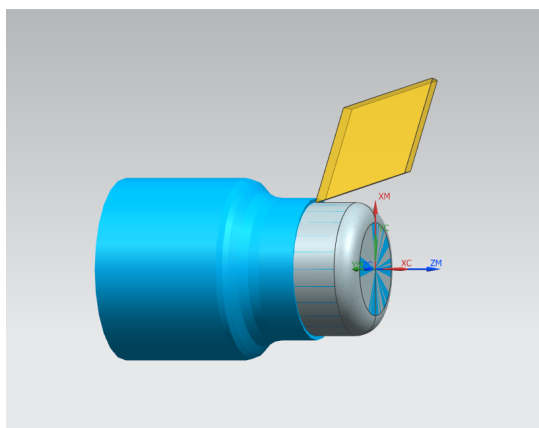
(b) Findreining



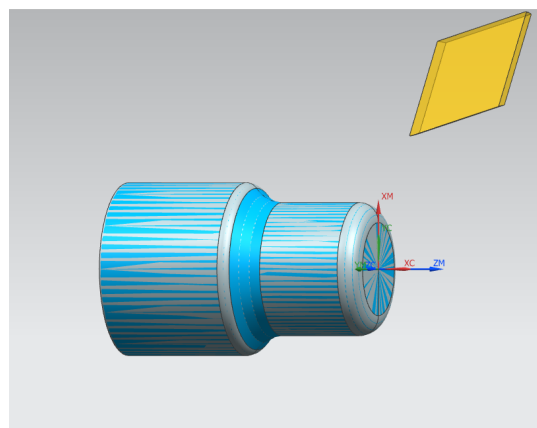
(c) Simulering av plandreining



(d) Simulering av grovdreining



(e) Simulering av findreining



(f) Simulering av sluttprodukt

Figur 5.19: Verktøybaner og simulering for test 1 i NX Manufacture.

Test 2

Maskineringen av testmodell 2 ble planlagt til å utføres med følgende tre operasjoner i NX Manufacture:

- plandreiing av endeflaten (Figur 5.20a)
- fresing av firkantgeometri med avrundede hjørner (Figur 5.20b)
- fresing av firkantgeometri med rette hjørner (Figur 5.20c)

Den første operasjonen var plandreiing av endeflaten med operasjonen "Facing". Råemnet var definert med 1mm ekstra materiale på endeflaten, som plandreiingen ville ta bort i ett kutt.

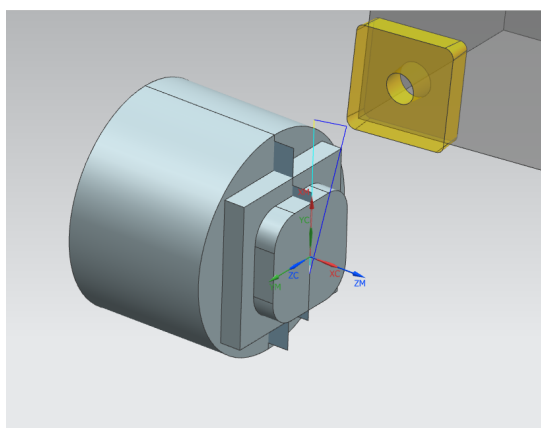
Den andre operasjonen bestod av fresing i XY-planet av den firkantede geometrien med avrundede hjørner, og det ble lagt inn et verktøybytte fra dreieverktøyet i den første operasjonen til en 8mm pinnefres. Operasjonstypen "Floor and Wall" ble valgt og kuttdybde på 3mm og et maksimalt inngrepsmål på 50% av verktøyets diameter ble definert. Mønsteret som verktøybanene skulle genereres etter ble bestemt til "Follow part" og toppflaten til den firkantede geometrien med rette hjørner ble definert som drivende geometri.

Den siste operasjonen bestod av fresing i XY-planet av den firkantede geometrien med rette hjørner. Det samme verktøyet og de samme maskinparametrene ble valgt som i den første freseoperasjonen, men med toppflaten av sylindere som drivende geometri. Mønster for utforming av verktøybaner ble definert til "Zig Zag".

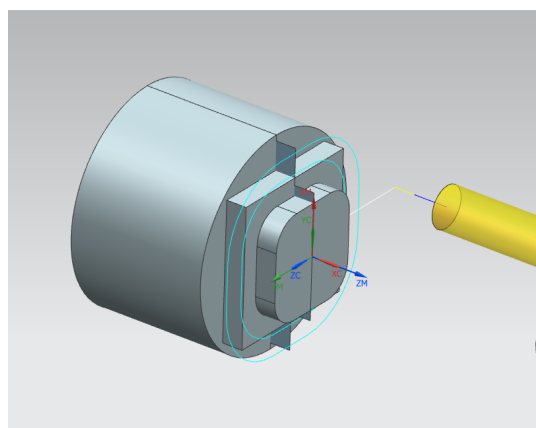
Verktøybanene og simuleringene for operasjonene vises i Figur 5.20.

Til slutt ble dreieoperasjonen og de to freseoperasjonene merket på samme måte som i test 1 med de riktige kallenavnene for dreieing og maskinering i XY-planet fra Tabell 3.1 i Kapittel 3.

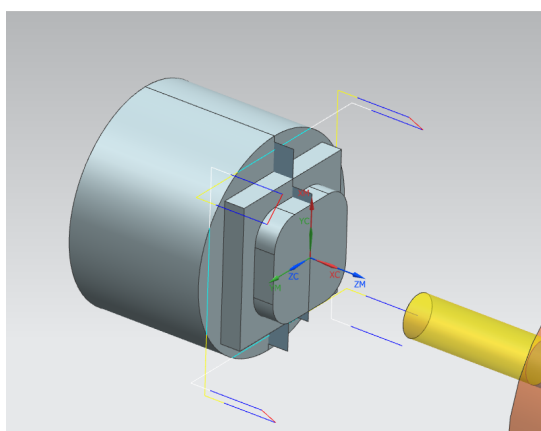
Den genererte maskinkoden for operasjonene inneholdt ingen feil, og de korrekte postprosessorne ble hentet inn til de forskjellige operasjonene. Maskinkoden ble også kjørt i simuleringmodus på maskinstyringen uten feil.



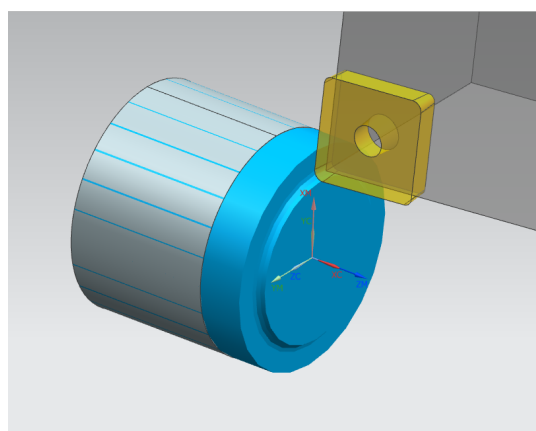
(a) Verktøybaner for plandreiring



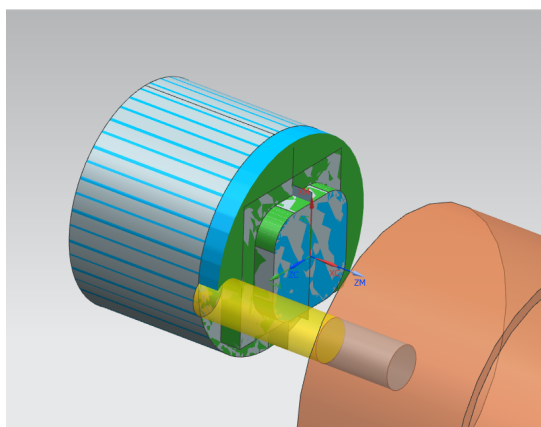
(b) Verktøybaner for freseoperasjon 1



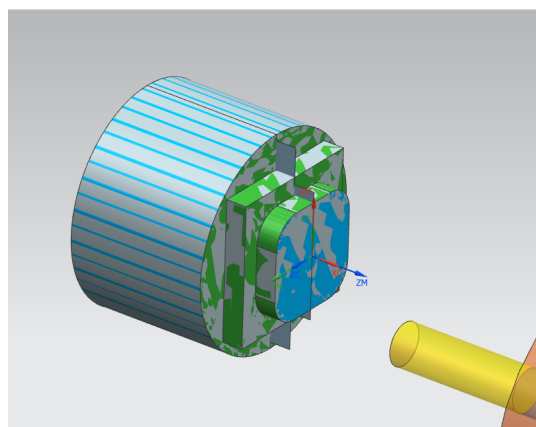
(c) Verktøybaner for freseoperasjon 2



(d) Simulering av plandreiring



(e) Simulering av freseoperasjon 1



(f) Simulering av freseoperasjon 2

Figur 5.20: Verktøybaner og simulering for test 2 i NX Manufacture.

Test 3

Maskineringen av testmodell 3 ble planlagt med følgende to freseoperasjoner i XZ-planet i NX Manufacture:

- fresing av spor (Figur 5.21a)
- fresing av lomme (Figur 5.21b)

I den første operasjonen skulle sporet på siden av testmodellen freses ut, og operasjonstypen "Floor & Wall" ble valgt i NX Manufacture. Bunnflaten i sporet ble valgt som drivende geometri og mønsteret "Zig" ble valgt for utforming av verktøybanene. Det ble definert en kuttdybde på 3mm, som ville maskinere bort alt materialet i dybden i ett lag, og et inngrepsmål på maksimalt 50% av verktøyets diameter. En 6mm pinnefres ble valgt som verktøy og med de definerte maskinparametrene gav dette fire parallelle kutt.

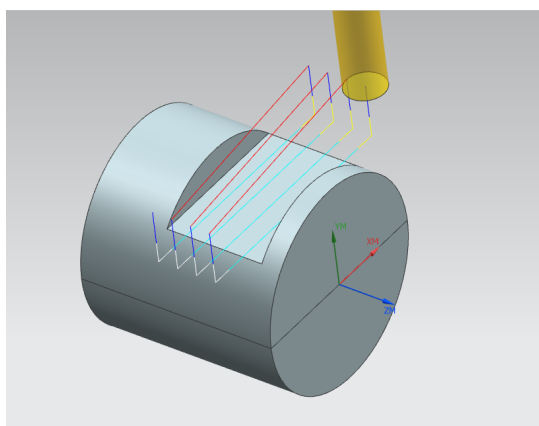
Den andre operasjonene som ble planlagt var lommefresing på den andre siden av arbeidsstykket. Den samme operasjonstypen og de samme maskinparametrene ble valgt for denne operasjonen, men med bunnen i lommen som drivende geometri og "Follow Part" som mønster for utforming av verktøybanene.

Verktøybanene og simuleringene for test 3 vises i Figur 5.21.

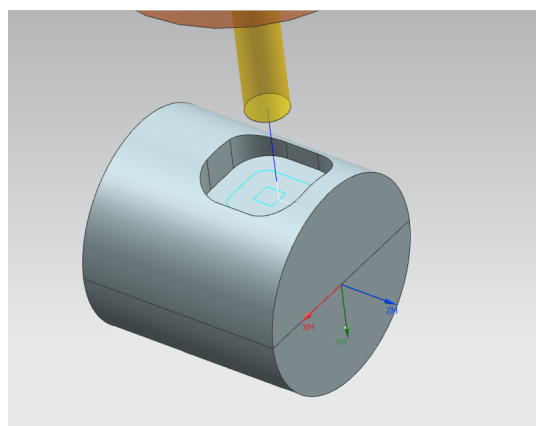
Til slutt ble operasjonene merket på samme måte som i test 1 og test 2, med det riktige kallenavnet for maskinering i ZY-planet fra Tabell 3.1 i Kapittel 3.

Den genererte maskinkoden for test 3 inneholdt ingen feil, og den korrekte postprosessoren ble hentet inn av master-postprosessoren for operasjonene. Maskinkoden for denne testen finnes i Vedlegg E, som et eksempel på hvordan maskinkode generert i denne postprosessoren ser ut.

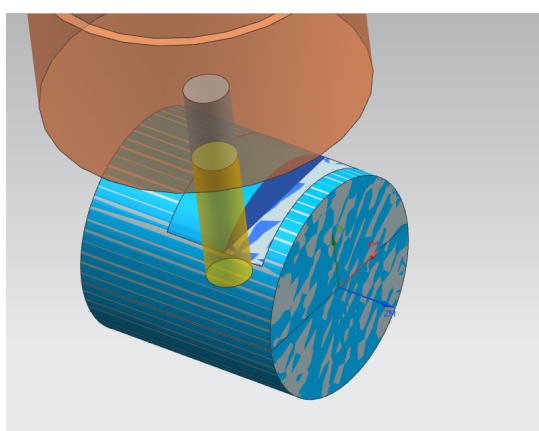
Maskinkoden ble også kjørt i simuleringsmodus på maskinstyringen uten feil.



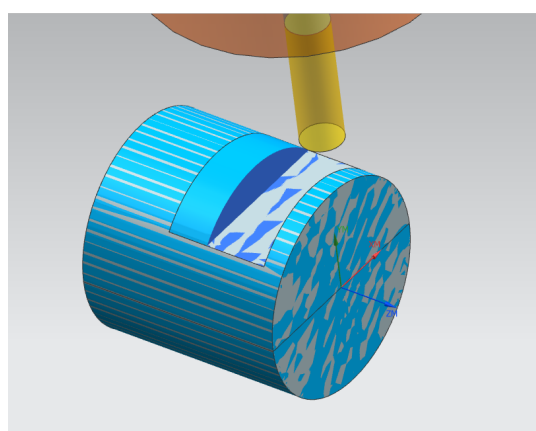
(a) Verktøybaner for freseoperasjon 1



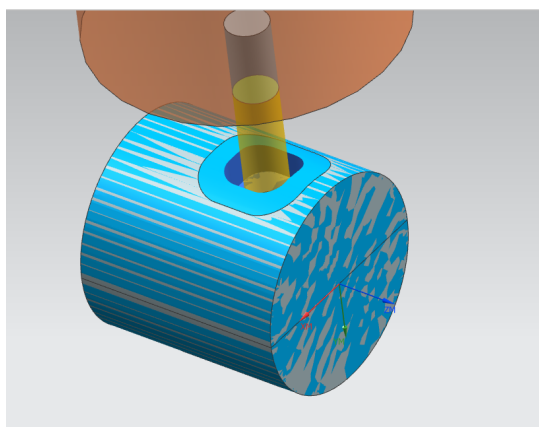
(b) Verktøybaner for freseoperasjon 2



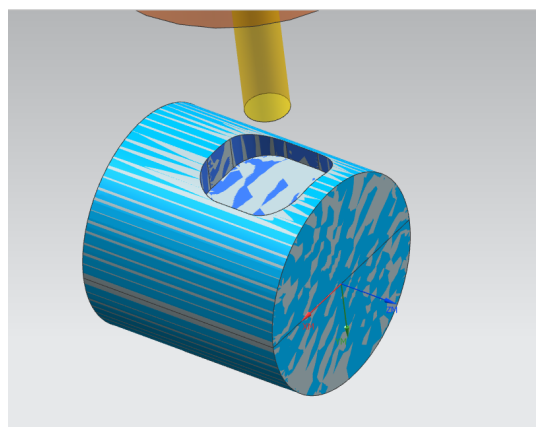
(c) Simulering av freseoperasjon 1



(d) Simulering av freseoperasjon 1



(e) Simulering av freseoperasjon 2



(f) Simulering av sluttprodukt

Figur 5.21: Verktøybaner og simulering for test 3 i NX Manufacture.

Test 4

For å planlegge maskineringen av testmodell 4 i NX Manufacture måtte noen endringer gjøres med modellen. Den opprinnelige modellen som ble brukt i GibbsCam hadde en svakhet i designet som gjorde det problematisk å planlegge maskineringen i NX Manufacture. I testmodellen var veggene i lommen som skulle freses ut rette, samtidig som bunnflaten var buet. For å få modellen til å samsvare med den opprinnelige planen om hvordan komponenten skulle maskineres ble veggene i lommen vinklet slik at de ville samsvare med de faktiske forholdene det ville være i maskinen. Maskineringen av modellen ble deretter planlagt med følgende tre operasjoner:

- radiell fresing av bunnflate i lomme (Figur 5.22a)
- konturfresing av vegger i lomme (Figur 5.22b)
- konturfresing av radielt spor (Figur 5.22c)

Den første operasjonen bestod av fresing av bunnflaten i lommen. I denne operasjonen ble "Variable contour" fra kategorien "Multi axis" valgt som operasjonstype og bunnflaten i lommen ble valgt som drivende geometri. Mønsteret for utformingen av verktøybanene ble valgt til "Zig Zag" og 3mm kuttdybde og et maksimalt inngrepsmål på 50% ble definert. Det ble også valgt at verktøyaksen skulle stå normalt på den drivende geometrien.

I den neste operasjonen skulle det resterende materialet langs veggene i lommen maskineres bort, og den samme operasjonstypen som i den første operasjonen, "Variable Contour", ble valgt. I denne operasjonen ble veggene definert som drivende geometri for operasjonen sammen med mønsteret "Follow part". "Swarf Drive" ble valgt som metode for verktøyaksen og "Swarf Ruling" for projeksjonsvektoren.

Den siste operasjonen bestod av radiell konturfresing av sporet på siden av arbeidsstykket. Til denne operasjonen ble operasjonstypen "Variable contour" og metoden "Curve/Point" valgt. Helix-kurven som ble brukt til å lage sporet i testmodellen ble valgt som drivende geometri for verktøybanen og det ble definert at hele sporet skulle maskineres bort i ett kutt.

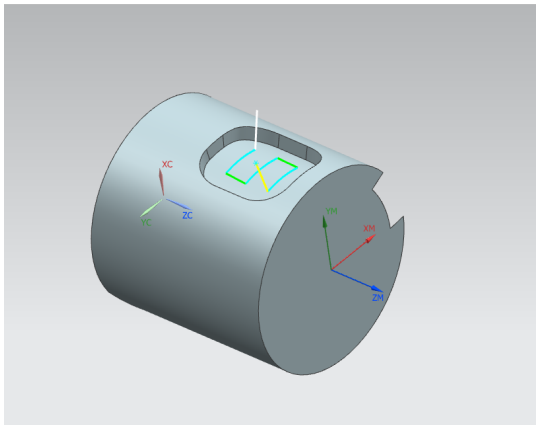
Den siste operasjonen kunne også ha blitt planlagt med metoden "Surface area" og bunnflaten som drivende geometri, eller en av veggene som drivende geometri med metoden "Swarf Drive" for verktøyaksen. Fordelen ved å benytte kurven, og ikke flatene, fra modellen som drivende geometri var at utformingen av verktøybanene i forlengelsen av sporet ville bli riktig. Ved å velge en av sideflatene eller bunnflaten som drivende geometri ville verktøybanen slutte før hele verktøyet var ute av sporet, eller starte med verktøyet i inngrep. En forlengelse kunne ha blitt lagt inn for å sikre at alt materialet ble fjernet i kuttet, men forlengelsen av verktøybanen ville ikke ha fulgt den samme helix-kurven som sporet.

De genererte verktøybanene og simuleringene for test 4 vises i Figur 5.22 og Figur 5.23.

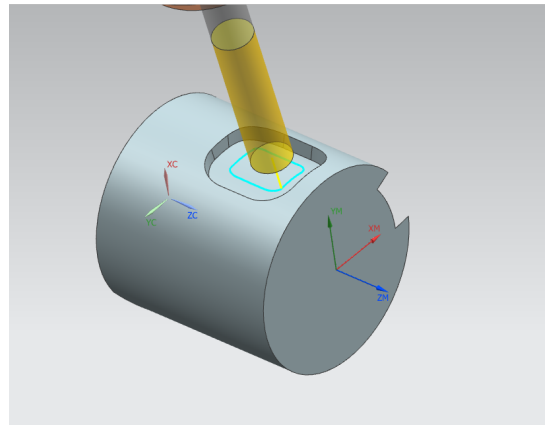
Til slutt ble operasjonene merket på samme måte som i test 1, 2 og 3, med det riktige kallenavnet for maskinering i ZY-planet fra Tabell 3.1 i Kapittel 3.

Den genererte maskinkoden for lommefresingen i de to første operasjonene i testen hadde ingen feil, og den korrekte postprosessen ble hentet inn av master-postprosessen for operasjonene. Maskinkoden for disse operasjonene ble kjørt i simuleringsmodus på maskinstyringen uten feil.

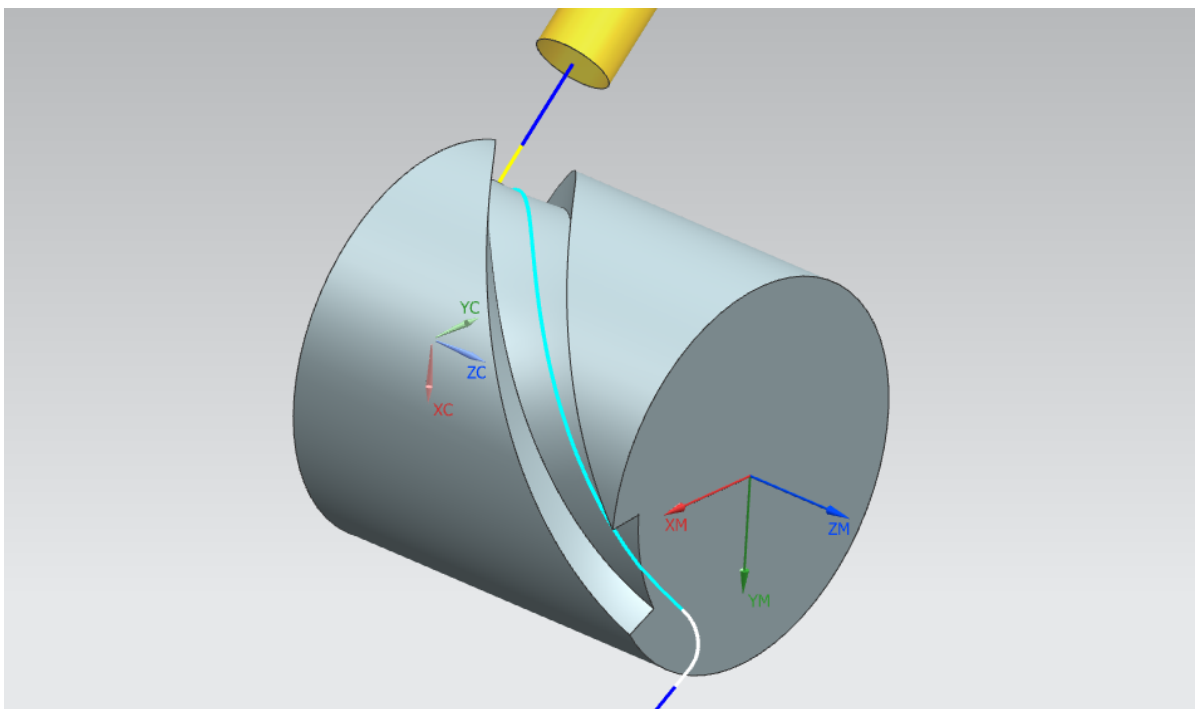
Den genererte maskinkoden for det radielle sporet i den siste operasjonen ble ikke generert som planlagt i denne testen. Av ukjent grunn ble ikke C-aksen rotert langs hele maskineringen. Operasjonen ble forsøkt postprosessert med en eksempel-postprosessor for en annen flerfunksjonsmaskin som fulgte med installasjonen av NX Manufacture. Denne maskinkoden hadde den samme feilen, med at C-aksen ikke roterte kontinuerlig gjennom hele maskineringen av sporet. Grunnet resultatet fra denne postprosessoren, og at den utviklede postprosessoren har fungert feilfritt for både posisjonering og radiell fresing i andre operasjoner, ser kandidaten det som mest sannsynlig at feilen ligger i modellen eller i den planlagte operasjonen i NX Manufacture, og ikke i postprosessoren.



(a) Verktøybaner for freseoperasjon 1

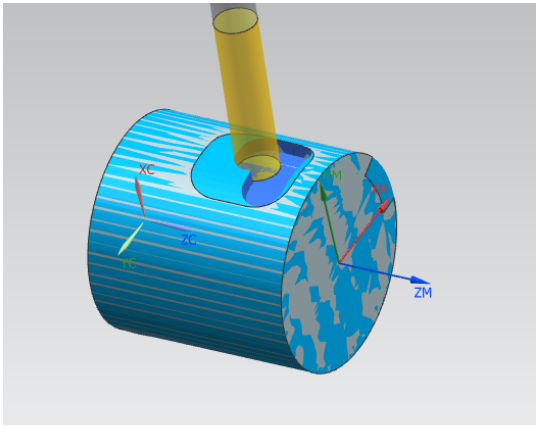


(b) Verktøybaner for freseoperasjon 2

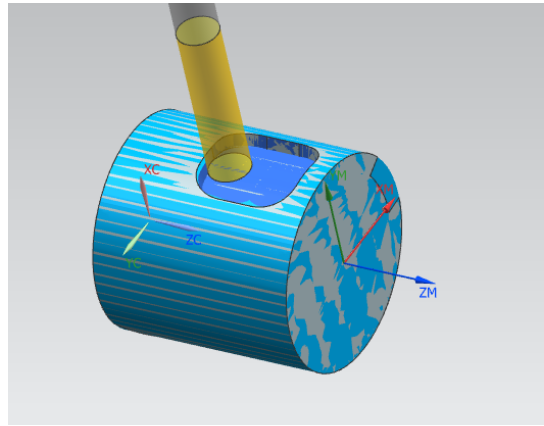


(c) Verktøybaner for freseoperasjon 3

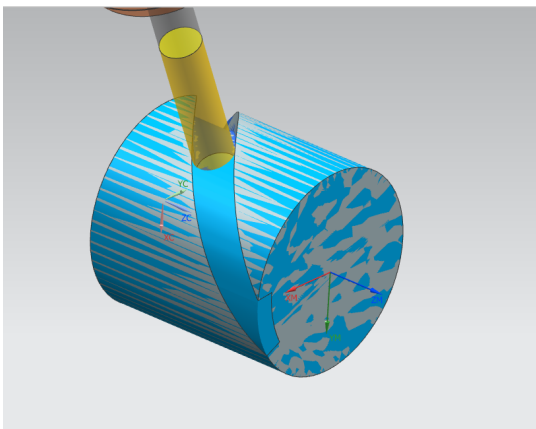
Figur 5.22: Verktøybaner for test 4 i NX Manufacture.



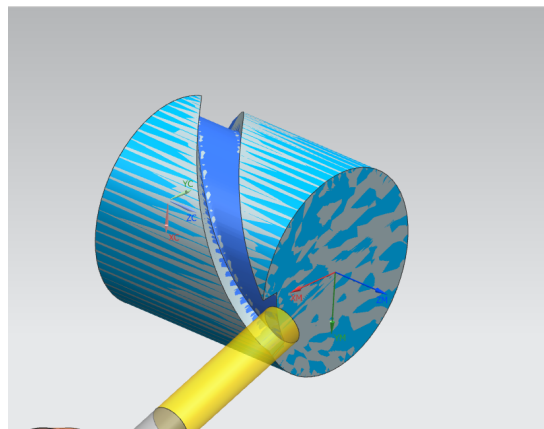
(a) Simulering av freseoperasjon 1



(b) Simulering av freseoperasjon 2



(c) Simulering av freseoperasjon 3



(d) Simulering av sluttprodukt

Figur 5.23: Simuleringer for test 4 i NX Manufacture.

5.3 Oppsummering

- Begge de brukte CAM-systemene har god støtte for programmering av operasjoner for 5-aksemaskinering og for flerfunksjonsmaskin. Sammenlignet har GibbsCam lavere brukerterskel enn NX Manufacture, men også mindre funksjonalitet.
- Den eksisterende postprosessoren til GibbsCam for Deckel Maho DMU 50eVolution 5-aksefres fungerer godt, men har en systematisk feil i genereringen av maskinkoden. I forbindelse med verktøyveksling i den genererte maskinkoden må *Z0 M92* endres til *Z-1 M91*.
- Den eksisterende postprosessoren til GibbsCam for Okuma Genos L200E-M flerfunksjonsmaskin ble grundig testet med operasjoner i XY-, XZ- og YZ-planet. Flere mangler ved postprosessoren ble funnet og fungerende løsninger for samtlige av manglene ble utviklet. Manglene og de anbefalte løsningene er oppsummert i Tabell 5.3.
- Den utviklede postprosessoren til NX Manufacture for Deckel Maho DMU 50eVolution 5-aksefres fungerte godt og genererte tilnærmet identisk maskinkode som Sørbys frittstående postprosessor for de samme operasjonene. Det ble ikke funnet noen feil ved postprosessoren.
- Den utviklede postprosessoren til NX Manufacture for Okuma Genos L200E-M flerfunksjonsmaskin fungerte godt og genererte oversiktlig maskinkode uten feil for operasjonene i testene. For radiell konturfresing i test 4 genererte postprosessoren ikke den ønskede maskinkoden, men feilen skyldes sannsynligvis geometrisk feil i CAD-modellen eller feil programmering av operasjonen i NX Manufacture.

Problem	Løsning
Manglende kommando for definering av nullpkt.	Makro og variabler legges til manuelt i maskinkode
Feil posisjon for verktøyveksling	Makro legges til manuelt i maskinkode
Bevegelse før det første verktøyet kalles på	Bevegelse erstattes med makro for verktøyvekslerpos.
Ikke støtte for radiuskompensering	Unngå bruk av radiuskompensering i CAM
Bruk av inkompatibel M-kode (M42)	Fjernes manuelt i maskinkode
Begrenset arbeidsrom for Y-akse i XY-plan	C-akse brukes i stedet for Y-akse. Defineres i CAM

Tabell 5.3: Oversikt over mangler med tilhørende løsninger for postprosessor til GibbsCam for Okuma Genos L200E-M

Kapittel 6

Prosedyre for definering av nullpunkt for 5-aksemaskinering i Deckel Maho DMU 50eVolution

I dette kapittelet beskrives en prosedyre for å definere riktig nullpunkt i maskinstyringen til Deckel Maho DMU 50eVolution for 5-aksemaskinering. For beskrivelse av hva som menes med nullpunkt refereres det til Kapittel 2.2.

6.1 Prosedyre for manuell definering av nullpunkt

For å kjøre 5-aksemaskinering med Deckel Maho DMU50eVolution må nullpunktet plasseres riktig før programmet startes. Den riktige posisjonen for nullpunktet for denne maskinen er på bordets overflate, kolinært med C-aksen. Når 5-aksemaskinering skal utføres må avstanden fra dette punktet til det oppspente arbeidsstykket måles opp og defineres i CAM. Avstanden fra maskinens koordinatsystem til arbeidsstykket i CAM-systemet skal være den samme som avstanden fra nullpunktet i maskinen til det samme punktet på arbeidsstykket i virkeligheten.

For å definere nullpunktet til den riktige posisjonen i Deckel Maho kan et referansekoordinatsystem som er plassert øverst til høyre i maskinen brukes. Dette koordinatsystemet er fast og kan ikke endres. For å beskrive en posisjon relativt til referansekoordinatsystemet i et program brukes M-koden M91 i den samme linjen som posisjonen det skal kjøres til. M91 gjelder kun for den blokken / linjen som koden er skrevet i. Posisjonen som de forskjellige aksene har relativt til dette referansekoordinatsystemet kan også leses av på maskinstyringen.

I målebrevet til maskinen (se Vedlegg B) er avstanden fra referansekoordinatsystemet i maskinen til den posisjonen som nullpunktet for 5-aksemaskinering skal ha oppgitt. For å kjøre til denne posisjonen ble et nytt program laget på maskinstyringen med posisjonene fra målebrevet, oppgitt i referansekoordinatsystemet. For å unngå kollisjon med en skrustikke og et emne som var oppspent ble det programmert at Z-aksen skulle gå til 500mm over den posisjonen som var oppgitt i målebrevet. I denne posisjonen ble det nye nullpunktet lagret for X og Y.

I maskinstyringen kompenseres det automatisk for verktøylengden, slik at Z-koordinatet alltid vil være gitt av tuppen på det aktive verktøyet, og ikke av spindelen eller verktøyholderen. For å få riktig nullpunkt i Z ble derfor verktøylengden for det aktive verktøyet satt til 0mm i verktøytabellen før nullpunktet i Z ble definert til å være 500mm under posisjonen til aksen. Verktøylengden ble deretter justert tilbake til korrekt verdi, og defineringen av nullpunktet var fullstendig.

Denne prosedyren for å sette riktig nullpunkt for 5-aksemaskinering er tidkrevende, og krever at de korrekte koordinatene fra målebrevet brukes. For å gjøre innstillingen av nullpunkt for 5-aksemaskinering raskere, enklere og med mindre sannsynlighet for brukerfeil ble to nye programmer utviklet:

- program for automatisk nullpunktsforskyvning
- hjelpeprogram for definering av nullpunkt

6.2 Program for automatisk nullpunktsforskyvning

Programmet for automatisk nullpunktsforskyvning, vist i Figur 6.1, flytter det aktive nullpunktet, som kan være vilkårlig plassert i arbeidsrommet til maskinen, til den korrekte posisjonen lokalt i programmet. Da det nye nullpunktet kun er definert lokalt i programmet, må programkoden kopieres inn i begynnelsen av det programmet som skal kjøres.

Når programmet kjøres nullstilles en eventuell aktiv forskyvning av nullpunktet først. Deretter posisjoneres aksene nært referanse-kordinatsystemets nullpunkt, øverst til høyre i maskinen og rotasjonsaksene roteres til nullposisjon. For å finne ut hvor det aktive nullpunktet i Z er i maskinens referanse-kordinatsystem kjøres aksene til det aktive nullpunktet, som kan ligge hvor som helst langs Z-aksen. I denne delen av programmet er det en liten fare for kollisjon med oppspenningsverktøy eller arbeidsstykke som er satt fast på arbeidsbordet i maskinen. For å minimere denne risikoen blir X-aksen og Y-aksen trukket så langt bort fra bordet som mulig tidligere i programmet, før nullpunktet i Z defineres. I Z-aksens nullpunkt leses posisjonen i referanse-kordinatsystemet ut fra maskinstyringen og lagres i variabelen Q13. Avstanden fra det aktive nullpunktet i Z til det ønskede nullpunktet regnes ut og lagres i variabelen Q16 etter at verktøylengden er trukket fra med variabelen Q114, som er en forhåndsdefinert variabel i maskinstyringen. I de neste linjene i programmet beregnes avstanden fra det aktive nullpunktet til det ønskede nullpunktet for X og Y, på samme måte som for Z.

Når avstanden til det ønskede nullpunktet er funnet for X,Y og Z-aksen forskyves det aktive nullpunktet til posisjonen til det ønskede nullpunktet med syklusen "DATUM SHIFT". Til slutt kjøres verktøyet til 400mm over det nye nullpunktet for å bekrefte for maskinoperatøren at det nye nullpunktet er posisjonert riktig.

Programmet ble testet i maskinen med suksessfullt resultat, hvor et tilfeldig plassert nullpunkt ble flyttet til den ønskede posisjonen for 5-aksemaskinering.

Språket som programmet ble skrevet i er *Heidenhein conversational language*, som er det primære språket til maskinstyringen.

```

0 BEGIN PGM NULLPKT MM
1 CYCL DEF 7.0 DATUM SHIFT
2 CYCL DEF 7.1 X+0 ;
3 CYCL DEF 7.2 Y+0 ;
4 CYCL DEF 7.3 Z+0 ;
5 L Z-1 F1000 M91
6 L X-1 F1000 M91
7 L Y-1 M91
8 L C+0 B+0 M91
9 L Z+0 F1000
10 FN 18: SYSREAD Q10 = ID20 NR1 ; LES AKTIVT VERKTOEY
11 FN 18: SYSREAD Q13 = ID240 NR1 IDX3 ; LES AKTIVT DATUM Z POS I M91
12 Q16 = -542.974 - Q13 + Q114 ; DIST FRA AKTIVT DATUM TIL NULLPUNKT I Z
13 L Z-42.974 F1000 M91
14 L X+0 F1000
15 L Y+0 F1000
16 FN 18: SYSREAD Q11 = ID240 NR1 IDX1 ; LES AKTIVT DATUM X POS I M91
17 FN 18: SYSREAD Q12 = ID240 NR1 IDX2 ; LES AKTIVT DATUM Y POS I M91
18 Q14 = -250.557 - Q11 ; DIST FRA AKTIVT DATUM TIL NULLPUNKT I X
19 Q15 = -155.073 - Q12 ; DIST FRA AKTIVT DATUM TIL NULLPUNKT I Y
20 CYCL DEF 7.0 DATUM SHIFT
21 CYCL DEF 7.1 X+Q14 ;
22 CYCL DEF 7.2 Y+Q15 ;
23 CYCL DEF 7.3 Z+Q16 ;
24 L X+0 Y+0 F1000
25 L Z+400 F1000
26 END PGM NULLPKT MM

```

Figur 6.1: Program for automatisk nullpunktsforskyvning i Deckel Maho

6.3 Hjelpeprogram for definering av nullpunkt

I tillegg til programmet for automatisk nullpunktsforskyvning ble det utviklet et hjelpeprogram, vist i Figur 6.2, for å definere et globalt nullpunkt som lagres i maskinstyringen. Programmet kan derfor kjøres selvstendig for å sette nullpunkt for 5-aksemaskinering som kan brukes i alle andre programmer som kjøres etterpå. Programmet krever noe innsats fra maskinoperatøren, men det er en enklere og raskere metode for å sette et globalt nullpunkt enn den manuelle prosedyren som ble beskrevet i seksjon 6.1.

I likhet med programmet for nullpunktsforskyvning brukes avstandene fra målebrevet (Vedlegg B) som grunnlag for å definere nullpunkt i dette programmet også. Når programmet startes kjøres aksene automatisk til de programmerte posisjonene relativt til maskinens referansekoordinatsystem. Verktøylengden leses inn fra verktøytabelen i maskinstyringen og lagres i variabelen Q10. Deretter kompenseres det for verktøylengden og programmet stopper. I dette

punktet veiledes maskinoperatøren til å sette nullpunktet i maskinstyringen med verdiene som er oppgitt i programmet.

Hjelpeprogrammet for definering av nullpunkt ble testet i maskinen med suksessfullt resultat, hvor et tilfeldig plassert nullpunkt ble flyttet til den ønskede posisjonen for 5-aksemaskinering og lagret i maskinstyringen. Den forklarende teksten i programmet ble lagt til i etterkant.

Programmet er skrevet i språket *Heidenhein conversational language*.

```
0 BEGIN PGM NULLPKT MM
1 ;
2 ; DETTE PROGRAMMET BRUKES FOR AA SETTE NULLPUNKT FOR
3 ; BRUK VED 5-AKSEMASKINERING. NULLPUNKTET PLASSERES I
4 ; C-AKSENS ROTASJONSSENTER PAA BORDETS OVERFLATE. DET
5 ; ER LIKGYLDIG HVILKET VERKTOEY SOM STAAR I HOLDEREN NAAR
6 ; PROGRAMMET KJOERES
7 ;
8 ; VED PROGRAMSTOPP SETTES DATUM / NULLPUNKT PAA
9 ; MASKINSTYRINGEN SOM BESKREVET I PROGRAMBLOKKEN
10 ;
11 L Z-1 F1000 M91
12 L X-250.557 Y-155.073 M91
13 FN 18: SYSREAD Q10 = ID20 NR1 ; LES AKTIVT VERKTOEY
14 Q1 = -242.974+Q114
15 L Z+Q1 M91
16 ;
17 M01 ; SETT DATUM / NULLPUNKT MED VERDIENE X = 0, Y = 0, Z = 300
18 ;
19 END PGM NULLPKT MM
```

Figur 6.2: Hjelpeprogram for posisjonering av nullpunkt for 5-aksemaskinering

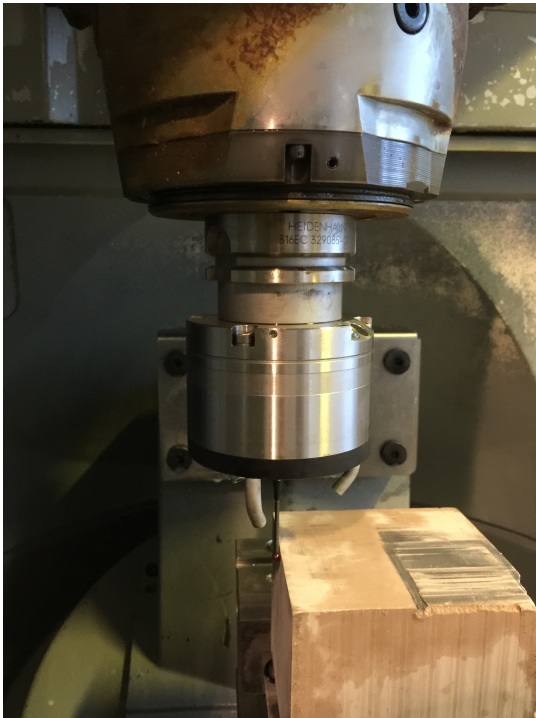
Kapittel 7

Maskineringstester

7.1 5-aksemaskinering av testmodell 2 i Deckel Maho

For å teste funksjonaliteten til postprosessoren og eventuelt avdekke flere feil, ble en maskineringstest for testmodell 2 satt opp. Med hensyn til økonomi og skadebegrensning ved en eventuell kollisjon mellom verktøy og arbeidsstykke ble et syntetisk materiale valgt som råemne. Emnet, utformet som en blokk, ble spent opp i en skrustikke ut fra senter i bordet for X, Y og Z. Det aktive nullpunktet i maskinen var ikke plassert i senter på bordets overflate og måtte derfor flyttes. Den manuelle prosedyren som ble beskrevet i seksjon 6.1 ble brukt for å sette et nytt nullpunkt. I etterkant av dette ble programmet for nullpunktsforskyvning og hjelpeprogrammet for definering av nullpunkt, beskrevet i Kapittel 6, utviklet for å gjøre arbeidet med å sette riktig nullpunkt i maskinstyringen raskere, enklere og med mindre sannsynlighet for feil.

Etter at nullpunktet var plassert korrekt ble avstanden fra nullpunktet til komponenten målt opp med en måleprobe som ble installert i spindelen. Avstanden ble tatt høyde for ved å flytte maskinens koordinatsystem i NX Manufacture og maskinkoden ble postprosessert på nytt. Programmet ble deretter kjørt i verktøymaskinen, hvor verktøyet fulgte de samme verktøybanene som var programmert i NX Manufacture. Figur 7.1 viser bilder fra oppmålingen og maskineringen i testen, og Figur 7.2 viser sluttproduktet.



(a) Oppmåling av råemnets posisjon



(b) 3-akse grovmaskinering



(c) 5-akse maskinering



(d) Sluttprodukt

Figur 7.1: Maskinering av testmodell 2 i Deckel Maho DMU 50eVolution

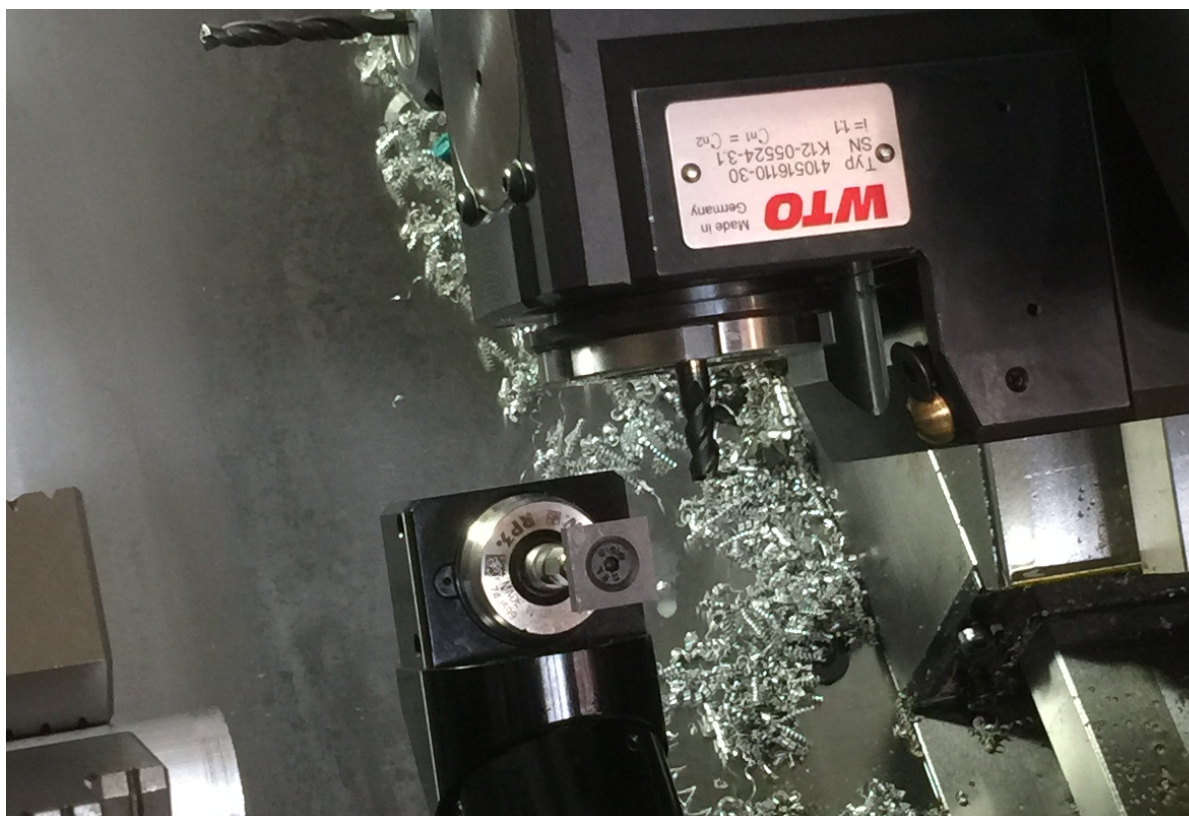


Figur 7.2: Sluttprodukt fra maskinering av testmodell 2 i Deckel Maho DMU 50eVolution

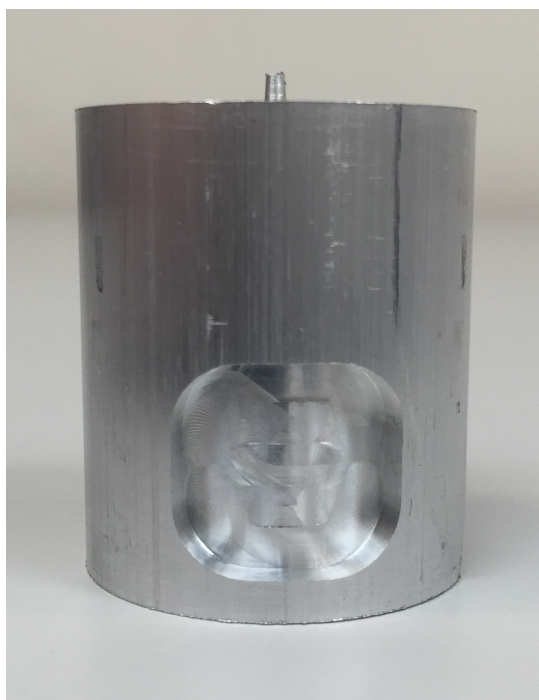
I Figur 7.2 er noe restmateriale synlig på toppen av kuleflaten. Dette skyldes at råemnet som ble brukt i testen ikke hadde en jevn toppflate. Da komponentens posisjon ble oppmålt med måleproben ble Z-koordinatet tatt fra et vilkårlig sted på overflaten av råemnet. Da det ikke var planlagt å fjerne noe materiale over toppen av kuleflaten i CAM kom denne ujevnheten i overflaten til syne på sluttproduktet. Hakket som synes på baksiden av kuleoverflaten i Figur 7.2 stammer ikke fra denne maskineringstesten, men fra en tidligere feil i defineringen av nullpunktet.

7.2 Maskinering av testmodell 4 i Okuma Genos

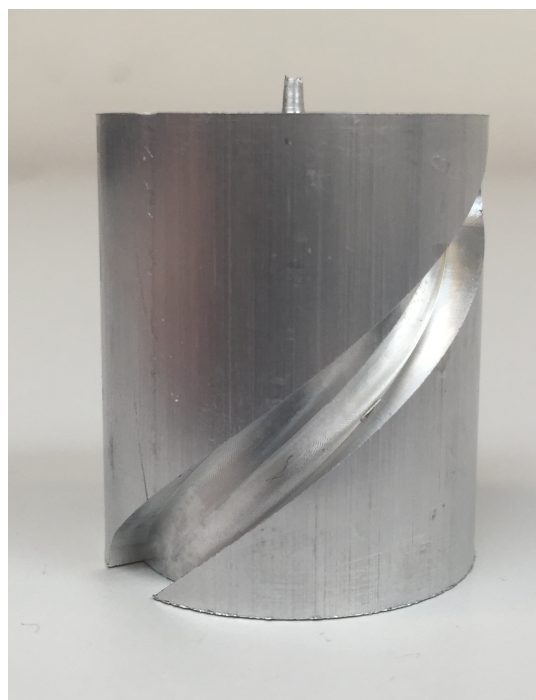
Til den fysiske maskineringstesten i Okuma Genos ble testmodell 4 utvalgt. Denne testen ble ansett som den mest avanserte blant testmodellene, og den eneste som inneholdt 4-akse simultanmaskinering. I forbindelse med gjennomføringen av testen ble plandreiing og avstikking lagt til i programmet av praktiske hensyn. Materialet som ble brukt i testen var aluminium og resultatet fra maskineringen vises i Figur 7.3. Verktøyet fulgte de genererte verktøybanene slik som de ble planlagt i GibbsCam, og som det sees fra bildene i Figur 7.3 endte det maskinerte emnet opp med den samme geometrien som planlagt. Et lite unntak for dette var at overflaten i bunnflaten av den utfreste lommen på arbeidsstykket hadde noen kanter som ikke samsvarte med CAD-modellen. Avviket skyldes sannsynligvis at spissen på freseverktøyet som ble brukt i testen var trekt noe inn fra endeflaten. Dette utgjør ingen forskjell ved fresing på et plant underlag, men ved maskinering på en sylinderflate vil en slik geometri i utformingen av verktøyet utgjøre en forskjell. Det er også sannsynlig at verktøybanene med fordel kunne hatt en annen utforming, for eksempel sikk-sakk mønster. Siden formålet med oppgaven var å utvikle og teste postprosessorer for CAM ble det ikke brukt mer tid på å optimalisere valg av verktøybaner så lenge de planlagte verktøybanene samsvarte mellom CAM og verktøymaskin. Det radielle sporet ble frest uten feil, foruten en liten rygg/forhøyning midt i sporet som skyldes den tidligere nevnte geometrien til freseverktøyet. Figur 7.3 viser resultatet fra maskineringen.



(a) Oppmåling av verktøy



(b) Frest lomme



(c) Frest spor

Figur 7.3: Maskinering av testmodell 4 i Okuma Genos L200E-M

Kapittel 8

Konklusjon og videre arbeid

8.1 Konklusjon

I prosjektet har funksjonaliteten til CAM-systemene GibbsCam og NX manufacture blitt utforsket for 5-aksemaskinering og maskinering i flerfunksjonsmaskin. De viktigste elementene i planleggingen av maskineringsoperasjoner for komponenter med forskjellige geometrier har blitt gjennomgått og de genererte verktøybanene og simuleringene har blitt vist. Resultatene fra testene viser at begge CAM-systemene har god støtte for planlegging av denne typen maskineringsoperasjoner.

I prosjektet har også en eksisterende postprosessor til GibbsCam for Deckel Maho DMU 50eVolution har blitt testet, og en systematisk feil i genereringen av maskinkoden ble funnet. Det ble også funnet en god løsning for å håndtere feilen ved hjelp av manuell redigering av maskinkoden, der linjer med *Z0 M92* som skrives ut i forbindelse med verktøyveksling ertsattes med *Z-1 M91*.

En eksisterende postprosessor til GibbsCam for Okuma Genos L200E-M ble testet grundig for operasjoner med dreining i XZ-planet og maskinering i XY-planet og YZ-planet. Flere mangler ved postprosessoren ble avdekket i testene og fungerende løsninger for manglene ble funnet. For oppsummering av manglene og de anbefalte løsningene refereres det til Tabell 5.3.

Det ble også utviklet postprosessorer til NX Manufacture for begge de nevnte maskinene. Begge postprosessorene er tilpasset rutine som brukes i verkstedet og genererer oversiktlig maskinkode. Kommandoene som ble programmert i postprosessorene er godt beskrevet med forklarende kommentarer i TCL-koden for å legge godt tilrette for eventuell videreutvikling. Postprosessoren til flerfunksjonsmaskinen er programmert til å bruke de samme makroene som de ansatte bruker til vanlig når maskinen programmeres manuelt, og en mal for å definere emne til simulering er inkludert som standard i begynnelsen av alle program som genereres.

De utviklede postprosessorene ble testet for flere typer maskineringsoperasjoner i forskjellige plan og med forskjellige typer verktøy. Foruten radiell konturfresing i test 4 har den genererte maskinkoden for samtlige tester blitt simulert eller maskinert med gode resultater. For den radielle konturfresingen i test 4 er det usikkert om den ugunstige maskinkoden skyldtes feil i modellen, programmeringen i CAM eller i postprosessoren.

For definering av riktig nullpunkt for 5-aksemaksinering ble det utviklet en prosedyre og to forskjellige programmer for å sette lokalt nullpunkt i programmet som kjøres, og globalt nullpunkt

som lagres på maskinstyringen.

Til slutt ble det utført fysiske maskineringstester i verktøymaskinene med 5-aksefresing og 4-akse radiell lommefresing og konturfresing.

8.2 Videre arbeid

Videre arbeid innenfor området er i hovedsak knyttet til kartlegging av form- og dimensjonsnøyaktighet fra maskinering hvor de samme CAM-systemene, postprosessorene og verktøymaskinene brukes. I prosjektet ble utvikling og testing av funksjonaliteten til postprosessorene prioritert fremfor analyse av nøyaktighet, og kun enkle målinger med skyvelære ble utført på de maskinerte komponentene.

I oppgaven ble oppmålingsdata fra installasjonen av Deckel Maho DMU 50eVolution brukt for å definere nullpunkt til 5-aksemaskinering. Hvorvidt disse posisjonene fremdeles er korrekte bør undersøkes sammen med nøyaktigheten til måleutstyret for oppmåling av verktøydimensjoner og posisjoner ved hjelp av måleprober i maskinene.

Postprosessorene som ble utviklet og eller testet i prosjektet kan inneholde feil eller mangle funksjonalitet som ikke ble avdekket i prosjektet. Maskinkoden som postprosessorene genererer bør derfor leses og simuleres før den kjøres fysisk på maskinene. Postprosessorene som ble utviklet i prosjektet for NX Manufacture kan og bør oppdateres løpende dersom manglende funksjonalitet og eller feil avdekkes under bruk.

For å benytte mulighetene som CAM gir sammen med avanserte verktøymaskiner kreves det også at de ansatte i verkstedet får mer opplæring og erfaring fra dagens nivå i bruk av CAM og postprosessering for dreining, 3-, 4- og 5-aksemaskinering.

Bibliografi

- [1] Making the move to a 5-axis vmc. <http://www.moldmakingtechnology.com/articles/making-the-move-to-a-5-axis-vmc>.
- [2] Knut Sørby. Inverse kinematics of five-axis machines near singular configurations. *International Journal of Machine Tools & Manufacture*, 47:299–306, 2007.
- [3] Siciliano B. *Robotics*. Springer-Verlag London, 2009.
- [4] Siemens. *Siemens NX Help Documentation - NX 9.0*.
- [5] International Organization for Standardization. ISO 10791-7:2014 Test conditions for machining centres – Part 7: Accuracy of finished test pieces. 2014.
- [6] Feng Wang. Design and implementation of five-axis transformation function in {CNC} system. *Chinese Journal of Aeronautics*, 27(2):425 – 437, 2014.
- [7] Tsong Der Lin Alan C. Lin, Tzu-Kuan Lin. Deriving generic transformation matrices for multi-axis milling machine. *International Journal of Mechanical, Aerospace, Industrial and Mechatronics Engineering*, 8(7):1223–1227, 2014.
- [8] International Organization for Standardization. ISO 6983-1:2009 Automation systems and integration – Numerical control of machines –Program format and definitions of address words –Part 1: Data format for positioning, line motion and countouring control systems.
- [9] Joseph Stiles Beggs. *Kinematics*. Hemisphere Publishing Corporation, 1983.
- [10] International Organization for Standardization. ISO 10303-1:1994 Industrial automation systems and integration – Product data representation and exchange – Part 1: Overview and fundamental principles.
- [11] Michael J. Pratt. Introduction to ISO 10303 - the STEP Standard for Product Data Exchange. *Journal of Computing and Information Science in Engineering*, 1:102–103, January 2001. <http://www.mel.nist.gov/msidlibrary/doc/jcise1.pdf> (sist sjekket 18.02.2015).
- [12] International Organization for Standardization. ISO 14649-1:2003 Industrial automation systems and integration – Physical device control – Data model for computerized numerical controllers – Part 1: Overview and fundamental principles.
- [13] W.Z. Yang, X. Xu, and S.Q. Xie. Development of step-nc compliant machine tool data model. In Srichand Hinduja and Kuang-Chao Fan, editors, *Proceedings of the 35th International MATADOR Conference*, pages 35–40. Springer London, 2007.

- [14] International Organization for Standardization. ISO 10303-11:2004 Industrial automation systems and integration – Product data representation and exchange – Part 11: Description methods: The EXPRESS language reference manual.
- [15] International Organization for Standardization. ISO 230-1:2012 Test code for machine tools – Part 1: Geometric accuracy of machines operating under no-load or quasi-static conditions.
- [16] Heidenhein. *User's Manual HEIDENHEIN Conversational Programming TNC 426 B TNC 430*.
- [17] Okuma. *OSP-P200L/P200LA/P20L, OSP P200L-R/P200LA-R/P20L-R, Programming Manual*. Pub. No 5238-E-R6 (LE33-013-R7), 7th edition, July 2011.
- [18] Crimson editor. <http://www.crimsoneditor.com/english/>.

Tillegg A

Liste over elektroniske vedlegg

Postprocessor til NX for Deckel Maho DMU 50eVolution
Postprocessor til NX for Okuma Genos L200E-M

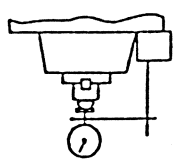
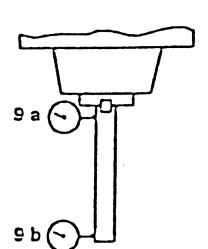
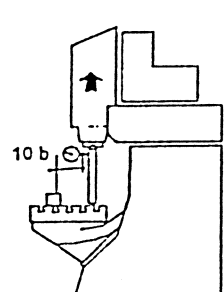
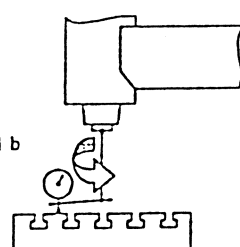
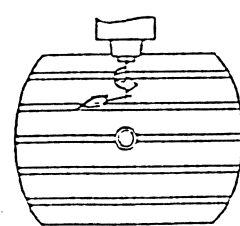
Tillegg B

Målebrev for Deckel Maho DMU 50eVolution

ABNAHMEPROTOKOLL

Geometrische Abnahme an Universal- Fräs- und Bohrmaschine- Vertikal Mit NC-Rundtisch mit Schwenkachse Messungen 1 - 19. Messung 20 nur bei Version L. Blatt 4 nur bei Maschine mit Gegehhalter.					Ident/Zeich-Nr.	
Typ DMU 50V-Neu		Masch. Nr. 11045515374	Projekt-Nr. 101365		O. Nr. SM000079	
Kunde			Abnehmer/Datum <i>Weyland</i>		Seite 1	von 4
Nr.	Gegenstand der Prüfung	Bild	Prüfmittel	Prüfanleitung	Abweichung	
					zulässig	gemessen
1	1 a Planlauf der Aufspanfläche r 350 mm		Meßuhr	Meßuhr auf Tisch anstellen. Tischplatte 350 Grad drehen.	0,01 mm 350 mm	1 a 0,005
	1 b Rundlauf der Zentrierbuchse		Fühlhebelmeßgerät	Fühlhebelmeßgerät in Zentrierbuchse anstellen. Tischplatte drehen.	0,01 mm	1 b 0,006
2	Parallelität der Längsbewegung zur Tischnut 2 a Aufspanfläche waagrecht		Fühlhebelmeßgerät	2 a Aufspanfläche waagrecht. Fühlhebelmeßgerät in T-Nute anstellen. Referenznut parallel zur Längsbewegung ausrichten.	2 a 0,005 mm, Meßlänge 300 mm	2 a 0,005
	2 b Aufspanfläche senkrecht			2 b Aufspanfläche senkrecht. Fühlhebelmeßgerät in gleiche T-Nute, auf gleicher Fläche (2a) anstellen. Meßlänge längs verfahren.	2 b 0,016 mm bis Meßlänge 400 mm	2 b 0,006
3	Parallelität der Längsbewegung zur Aufspanfläche 3 a Aufspanfläche waagrecht		Meßuhr, Meßschiene	Meßschiene auflegen, befestigen. Meßuhr anstellen. Meßlänge längs verfahren.	0,015 mm bis Meßlänge 400 mm	3 a 0,005
	3 b Aufspanfläche senkrecht			Tisch um 180 Grad schwenken. Meßuhr auf gleiche Fläche (3a) anstellen. Meßlänge längs verfahren.		3 b 0,010
4	4 a Parallelität der Querbewegung zur Aufspanfläche		Meßuhr, Meßschiene	Meßschiene auflegen. Meßuhr anstellen. Meßlänge quer verfahren.	0,02 mm bis Meßlänge 300 mm	4 a 0,008
	4 b Parallelität der Senkrechtbewegung zur Aufspanfläche			Tisch um 180 Grad schwenken. Meßuhr auf gleiche Fläche (4a) anstellen. Meßlänge senkrecht verfahren.		4 b 0,006
5	Rechtwinkligkeit der Querbewegung zur Aufspanfläche		Meßuhr, Meßwinkel	Meßwinkel auf Tischmitte befestigen. Meßuhr anstellen. Meßlänge quer verfahren.	0,02 mm, Meßlänge 300 mm	5 0,006
6	Rechtwinkligkeit der Längsbewegung zur Querbewegung		Meßuhr, Meßwinkel	Meßwinkelunterseite parallel zur Längsbewegung ausrichten. Meßuhr an zylindrischen Teil des Meßwinkels anstellen. Meßlänge quer verfahren.	0,02 mm, Meßlänge 300 mm	6 0,007
7	Rechtwinkligkeit der senkrechten Bewegung zur Aufspanfläche 7 a in Längsrichtung		Meßuhr, Meßwinkel	Meßwinkel auf Tischmitte stellen. Längsachse mittig positionieren. Meßuhr an Meßwinkel anstellen. Meßlänge senkrecht verfahren.	0,02 mm, Meßlänge 300 mm	7 a 0,008
	7 b in Querrichtung			Zu Messung 7b Meßwinkel um 90 Grad drehen.		7 b 0,005

ABNAHMEPROTOKOLL

Geometrische Abnahme an Universal- Fräs- und Bohrmaschine- Vertikal Mit NC-Rundtisch mit Schwenkachse Messungen 1 - 19. Messung 20 nur bei Version L, Blatt 4 nur bei Maschine mit Gegenhalter.					Ident/Zeich.-Nr.		
Typ DMU 50V-Neu		Masch. Nr.		Projekt-Nr.		O. Nr. SM000079	
Kunde				Abnehmer/Datum		Seite von 2 4	
Nr.	Gegenstand der Prüfung	Bild	Prüfmittel	Prüfanleitung	Abweichung		
					zulässig	gemessen	
8	Axialruhe der Arbeitsspindel		Meßuhr, Prüfdorn	Prüfdorn in Arbeitsspindel einsetzen. Meßuhr mittig anstellen. Arbeitsspindel mit niedrigster Drehzahl laufen lassen.	0,01 mm	8 0,003	
9	Rundlauf des Innenkegels der Arbeitsspindel 9 a nahe der Spindelnahe		Meßuhr, Prüfdorn 300 mm	Prüfdorn in Arbeitsspindel einsetzen. Meßuhr wie unter 9 a bzw. 9 b anstellen. Arbeitsspindel drehen.	0,01 mm	9 a 0,006	
	9 b im Abstand von 300 mm zur Spindelnahe				0,02 mm	9 b 0,014	
10	Parallellität der Arbeitsspindel zur senkrechten Bewegung 10 a in Längsrichtung 10 b in Querrichtung		Meßuhr, Prüfdorn 300 mm	Prüfdorn in Arbeitsspindel einsetzen. Rundlauffehler ausmitteln. Meßuhr wie unter 10 a bzw. 10 b anstellen. Meßlänge senkrecht verfahren.	0,02 mm, Meßlänge 300 mm	10 a 0,005	
						10 b 0,006	
11	Umschlagmessung mit der Arbeitsspindel 11 a in Längsrichtung		Meßuhr, Umschlagarm 150 mm	Arbeitsspindel mittig zum Tisch positionieren. Umschlagarm mit Meßuhr in Arbeitsspindel einsetzen. Meßuhr auf Tischoberfläche nullen. Umschlagarm 180 Grad drehen. Umschlag längs (11 a) und quer (11 b) messen.	0,02 mm, Meßlänge 300 mm	11 a 0,010	
	11 b in Querrichtung				11 b 0,005		
12	Umschlagmessung mit der Arbeitsspindel 12 a in Längsrichtung		Fühlhebelmeßgerät, Umschlagarm 150 mm	Tischoberfläche waagrecht. Tischnut zur Längsachse ausrichten. Tischoberfläche in senkrechte Stellung schwenken. Arbeitsspindel mittig zum Tisch positionieren. Umschlagarm mit Fühlhebelmeßgerät in Arbeitsspindel einsetzen. Fühlhebelmeßgerät in gleiche Tischnut, auf gleicher Fläche anstellen, nullen. Umschlagarm 180 Grad drehen.	0,02 mm, Meßlänge 300 mm	12 a 0,005	
	12 b in Querrichtung					12 b 0,005	

ABNAHMEPROTOKOLL

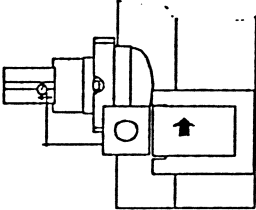

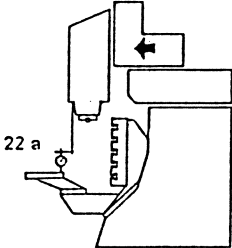


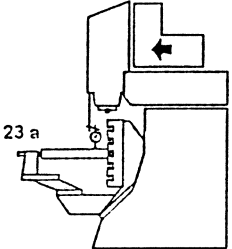

Maße / Programmierhilfe für Universal- Fräs- und Bohrmaschine- Vertikal Mit NC-Rundtisch mit Schwenkachse Messungen 1 - 19 Messung 20 nur bei Version L. Blatt 4 nur bei Maschine mit Gegenhalter			Ident/Zeich.-Nr.
Typ DMU 50V-Neu	Masch. Nr. 11045515374	Projekt-Nr. 101365	Q. Nr. SM000079
Kunde		Acnehmer/Datum	Seite von 3 4
Nr.	Gegenstand der Messung	Bild	Vorraussetzungen
13	Abstand der Frässpindel vom Referenzpunkt der Längsachse. Aufspannfläche in Pos. 0- Grad. Schwenkachse in Pos. 0- Grad. Bezug: Zentrierbuchse.		Referenzpunktverschiebung und Tischkompensation nicht aktiv. - 250,683 - 250,557
14	Abstand der Frässpindel vom Referenzpunkt der Querachse. Aufspannfläche in Pos. 0- Grad. Schwenkachse in Pos. 0- Grad. Bezug: Zentrierbuchse.		Referenzpunktverschiebung nicht aktiv. - 155,029 - 155,073
15	Abstand des Frässpindelkonus vom Referenzpunkt der senkrechten Achse. Aufspannfläche in Pos. 0- Grad. Schwenkachse in Pos. 0- Grad. Bezug: Aufspannfläche nahe der Zentrierbuchse.		Referenzpunktverschiebung nicht aktiv. - 542,974
16	Abstand der Frässpindel vom Referenzpunkt Längsachse. Aufspannfläche in Pos. 0- Grad. Schwenkachse in Pos. 180- Grad. Bezug: Zentrierbuchse.		Referenzpunktverschiebung und Tischkompensation nicht aktiv. - 250,673
17	Abstand der Frässpindel vom Referenzpunkt der Querachse. Aufspannfläche in Pos. 0- Grad. Schwenkachse in Pos. 180- Grad. Bezug: Aufspannfläche nahe der Zentrierbuchse.		Referenzpunktverschiebung nicht aktiv. - 0,033
18	Abstand des Frässpindelkonus vom Referenzpunkt der senkrechten Achse. Aufspannfläche in Pos. 0- Grad. Schwenkachse in Pos. 180- Grad. Bezug: Zentrierbuchse.		Referenzpunktverschiebung nicht aktiv. - 387,978
19	19 a: Abstand des Schnittpunktes Drehachse- Schwenkachse von der Aufspannfläche, in waagrechter Richtung.		Referenzpunktverschiebung nicht aktiv. 19 a berechnen: Abstand 14 minus Abstand 17 = 154,996
	19 b: Abstand des Schnittpunktes Drehachse- Schwenkachse von der Aufspannfläche, in senkrechter Richtung.		Referenzpunktverschiebung nicht aktiv. 19 b berechnen: Abstand 15 minus Abstand 18 = 154,996
20	Abstand des Frässpindelkonus vom Referenzpunkt der senkrechten Achse. Aufspannfläche in Pos. 0- Grad. Schwenkachse in Pos. 0- Grad. Bezug: Tisch in Kabine rechts. Aufspannfläche nahe der Zentrierbuchse.		Referenzpunktverschiebung nicht aktiv. 20

Hinweis: Die Achsposition der Längsachse werden nach dem Schwenken des Tisches um 180 Grad (Tisch senkrecht, Tisch waagrecht) von der Maschine gleich angefahren. Dies ist durch die werkseitige Eingabe von Kompensationswerten in die entsprechenden Maschinenkonstanten gewährleistet. Die Kompensation erfolgt automatisch beim Schwenken des Tisches.

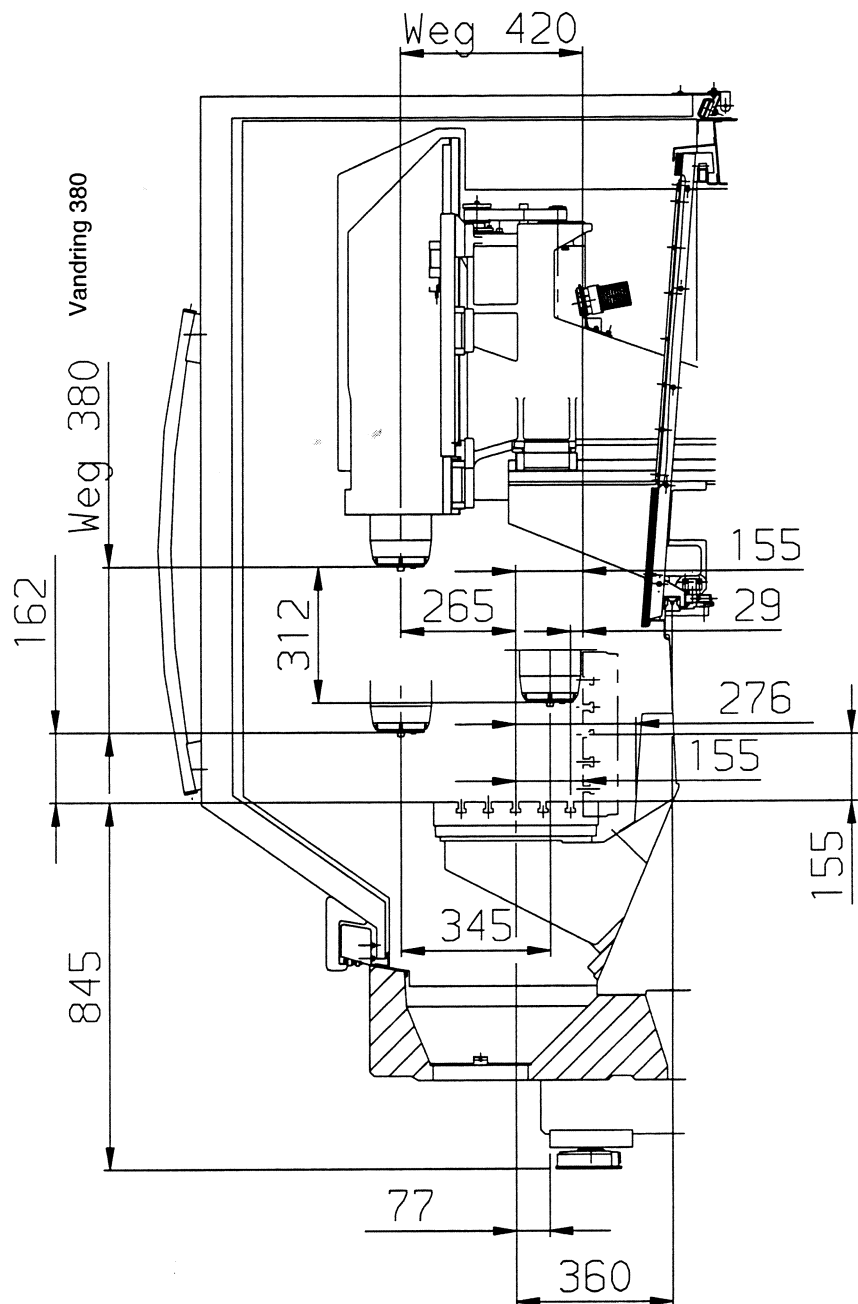
Achtung: Bitte beachten Sie die Lage der Referenzpunkte und Maschinennullpunkte an Ihrer Maschine. Sie kann von den symbolischen Darstellungen abweichen (z.B. nach Tausch der Maschinenachsen).

Achse	Längsachse	Querachse	Senkrechte Achse
Bild			

ABNAHMEPROTOKOLL

Geometrische Abnahme an Universal- Fräs- und Bohrmaschine- Vertikal Mit NC-Rundtisch mit Schwenkachse Messungen 1 - 19, Messung 20 nur bei Version L, Blatt 4 nur bei Maschine mit Gegenhalter.					Ident/Zeich.-Nr.	
Typ DMU 50V-Neu		Masch. Nr.		Projekt-Nr.		Q Nr. SM000079
Kunde				Abnehmer/Datum		Seite 4 von 4
Nr.	Gegenstand der Prüfung	Bild	Prüfmittel	Prüfanleitung	Abweichung	
					zulässig	gemessen
21	Parallelität der Längsbewegung zur Gegenhalterfläche		Meßuhr	Meßuhr an der Gegenhalterfläche anstellen und nullen. Meßlänge längs verfahren.	0,01 mm, Meßlänge 90 mm	21 0,005
22	Parallelität der Querbewegung zum Gegenhalter 22 a Fläche 		Meßuhr, Fühlhebelmeßgerät	Meßuhr auf Gegenhalterfläche anstellen und nullen. Meßlänge quer verfahren.	0,015 mm, Meßlänge 200 mm	22 a 0,007
	22 b Nut 					Fühlhebelmeßgerät in T-Nute anstellen und nullen. Meßlänge quer verfahren.
23	Parallelität der Querbewegung zu Gegenhalter mit Reitstock 23 a oben 		Meßuhr, Prüfdorn	Prüfdorn in Zentrierbuchse geben, Reitstock mit Zentrierspitze anstellen. Meßuhr an Prüfdorn anstellen und nullen. Meßlänge quer verfahren.	0,02 mm, Meßlänge 300 mm	23 a 0,006
	23 b seitlich 					23 b 0,009

Med 2 axels NC-rundbord



Tillegg C

Programmerte kommandoer

C.1 Kommando for kjøring til sikker posisjon - Deckel Maho

```
proc PB_CMD_sikkerpos { } {  
  # Denne kommandoen kan legges til som en event for å sende X,Y og Z-aksene  
  # til en sikker posisjon før B og C-aksene roteres til 0.  
  # Posisjonen som X,Y og Z-aksene sendes til før rotasjon av B og C-aksene er øverst til venstre i maskinen.  
  
  MOM_output_literal "G01 Z-1 F5000 M91"  
  MOM_output_literal "G01 Y-420 M91"  
  MOM_output_literal "G01 X-500 M91"  
  MOM_output_literal "G01 Z-1 M91"  
  MOM_output_literal "G01 B0 C0 M91"  
  
}
```

C.2 Kommando for verktøybytte - Okuma

```
proc PB_CMD_verktoeybytte {} {
```

```
#Denne kommandoen brukes i forbindelse med verktoeybytte og kan legges til som en "event" i "Program & Tool Path"  
#Kommandoen sender først verktoeyholderen til X-limit ved hjelp av en macro i maskinstyringen  
#verktoeynummer skives deretter ut i formatet TXXXXXXX, hvor X-er verkoeynummeret med to siffer.  
#Kommandoen legger ogsaa til M8, for kjoeleveaeske.
```

```
global mom_tool_number
```

```
MOM_output_literal "CALL OVVP"
```

```
if { $mom_tool_number>=10 } {  
    MOM_output_literal "T$mom_tool_number$mom_tool_number M08"  
} else {  
    set nulltall 0  
    MOM_output_literal "T0$mom_tool_number$nulltall$mom_tool_number$mom_tool_number M08"  
}
```

```
}
```

C.3 Kommando for aktivering av auto-låsning av C-aksen - Okuma

```
proc PB_CMD_M146/M147 {} {  
  
    # Denne kommandoen aktiverer auto-clamp av C-aksen  
  
    global mom_sys_auto_clamp  
  
    set mom_sys_auto_clamp "ON"  
  
}
```


C.4 Kommando for beskrivelse av rotasjonsretning C-akse - Okuma

```
proc PB_CMD_M15/M16 {} {  
#Denne kommandoen brukes til aa skrive ut M15 og M16 for posisjonering av C- akse  
#med rotasjonsretning med (M15)og mot (M16) klokken.  
#Kommandoen kontrollerer om det er endring i vinkel, hvilken retningskode som er aktiv  
#og skriver kun ut ny retning dersom det er endring i rotasjonsretningen.  
#For aa unngaa feilaktig utskrift av rotasjonsretning naar vinkelen krysser 0, brukes vinkelen  
#350 grader og 10 grader til aa sjekke om vinkelendringen har krysset 0 men roterer i samme retning.  
#Kommandoen kalles opp i linear move, rotary move og rapid move.
```

```

global mom_user_aktiv_rotasjonsretning mom_user_forrige_vinkel
global mom_pos mom_out_angle_pos mom_pos

if { ![info exists mom_user_aktiv_rotasjonsretning] } {

    set mom_user_aktiv_rotasjonsretning CLW
}
if { ![info exists mom_user_forrige_vinkel] } {
    set mom_user_forrige_vinkel $mom_out_angle_pos(0)
}
if { $mom_out_angle_pos(0) != $mom_user_forrige_vinkel } {

    if { $mom_out_angle_pos(0) > $mom_user_forrige_vinkel && \
        [string match "CCLW" $mom_user_aktiv_rotasjonsretning] } {

        if { $mom_out_angle_pos(0) > 350 && $mom_user_forrige_vinkel < 10 } {

            } else {

                set mom_user_aktiv_rotasjonsretning CLW MDM_output_literal "M15"

            }

        if { $mom_out_angle_pos(0) < $mom_user_forrige_vinkel && \
            [string match "CLW" $mom_user_aktiv_rotasjonsretning] } {

            if { $mom_out_angle_pos(0) < 10 && $mom_user_forrige_vinkel > 350 } {

                } else {

                    set mom_user_aktiv_rotasjonsretning CCLW
                    MDM_output_literal "M16"

                }

            }

        }

    }
}

```

C.5 Kommando for aktivering av fysisk y-akse - Okuma

```
proc PB_CMD_G138 {} {  
  
#Denne kommandoen skriver ut G138 (aktiver fysisk Y-akse) dersom den ikke er  
#skrevet ut tidligere, eller dersom G136 er aktiv. Ny status legges i variabelen  
#mom_user_fysisk_Y_status.  
  
    global mom_user_fysisk_Y_status  
  
    if { ![info exists mom_user_fysisk_Y_status] ||\  
        [string match "AV" $mom_user_fysisk_Y_status] } {  
  
        set mom_user_fysisk_Y_status PAA  
        MOM_output_literal "G138"  
  
    }  
  
}
```

C.6 Kommando for deaktivering av fysisk Y-akse - Okuma

```
proc PB_CMD_G136 {} {
#Denne kommandoen skriver ut G136 (deaktiver fysisk Y-akse) dersom G138 er skrevet
#ut tidligere og oppdaterer ny status i en variabelen mom_user_fysisk_Y_status.

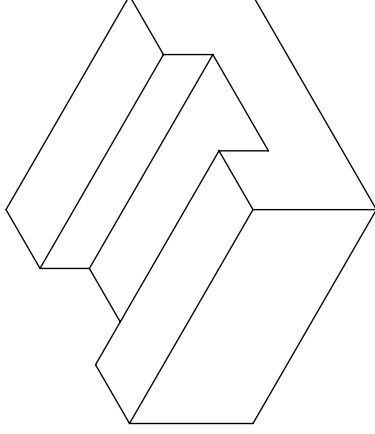
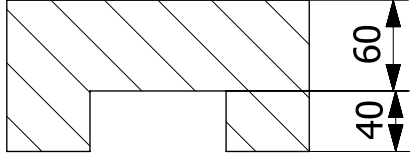
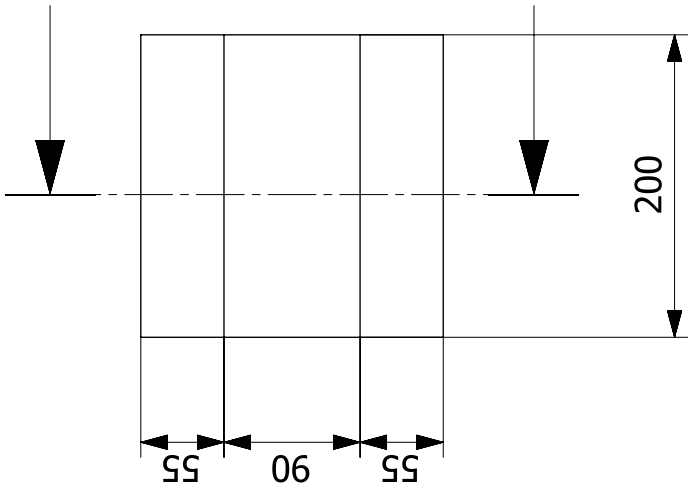
    global mom_user_fysisk_Y_status

    if { [info exists mom_user_fysisk_Y_status] &&\
        [string match "PAA" $mom_user_fysisk_Y_status] } {


        set mom_user_fysisk_Y_status AV
        MOM_output_literal "G136"
    }
}
```

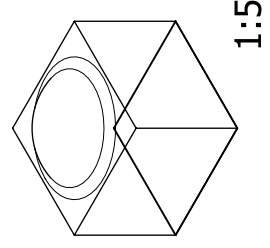
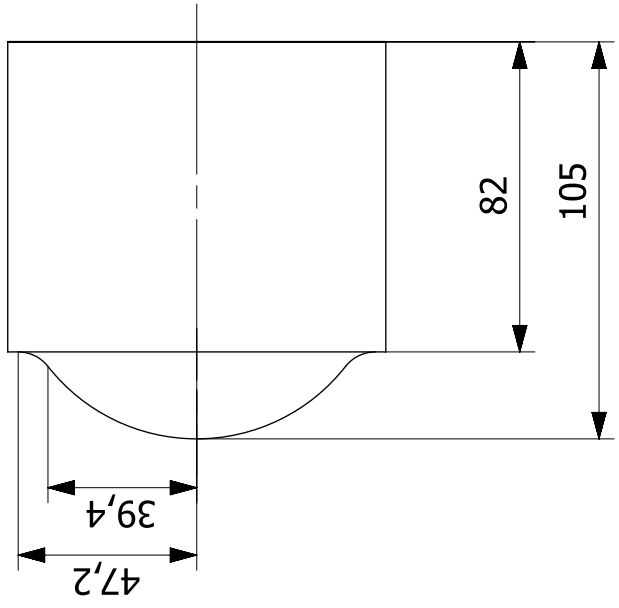
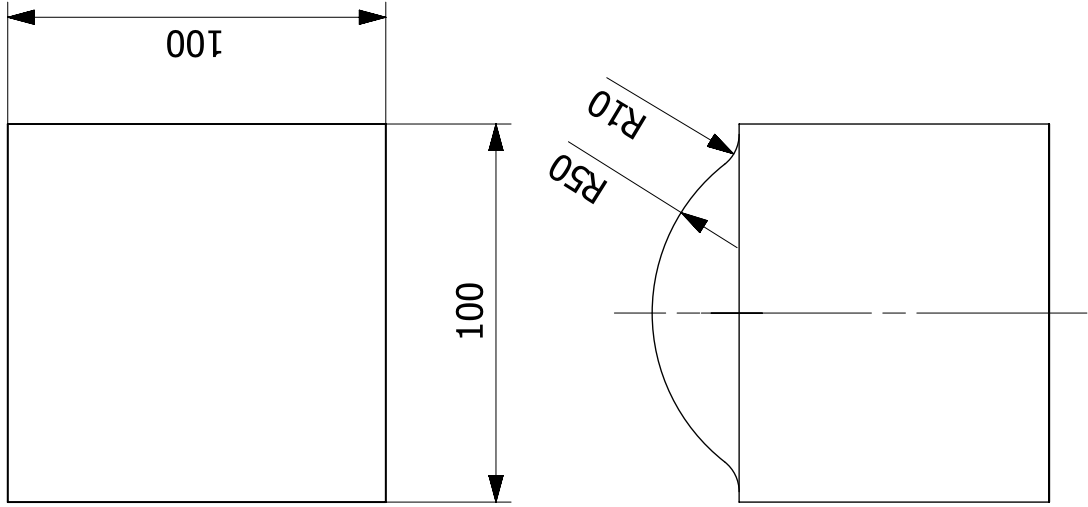
Tillegg D

Maskintegninger av testmodeller

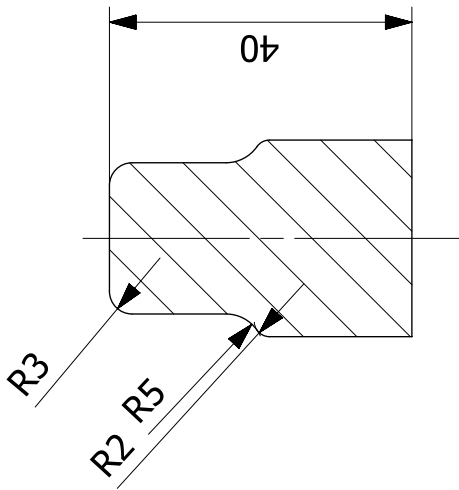
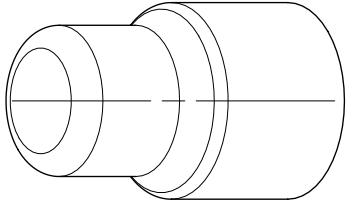
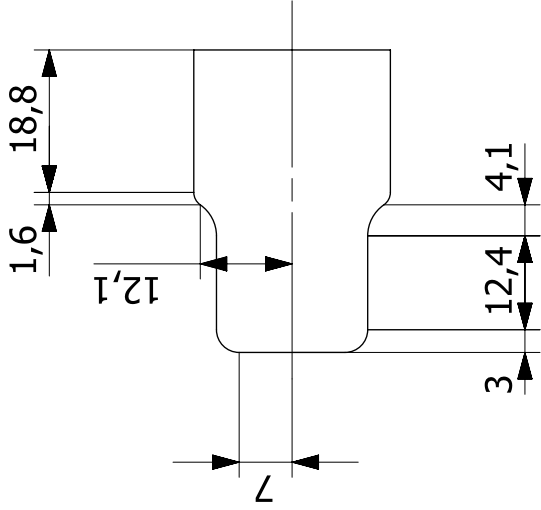
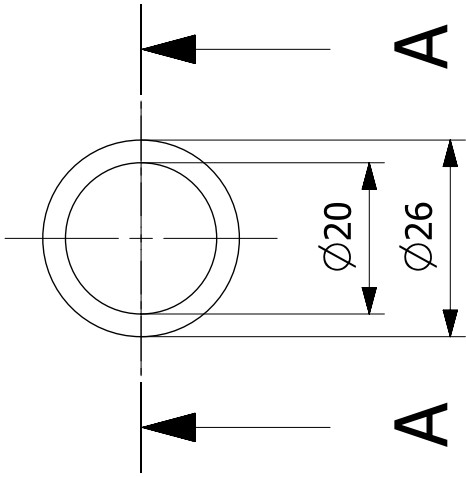


SECTION A-A


Antall	Pos. nr.	Navn, type, dimensjon		Tegning produktid	Materiale	kg pr. stk.
Dato	Konstr./Tegnet	Prosjeksjonsmetode 	Målestokk 1:5	Firmanavn NTNU		
Tittel:		TEST 1 - Deckel Maho				
Henvisning		Beregning		Erstatning for:	Erstatter av:	
				Tegn nr		

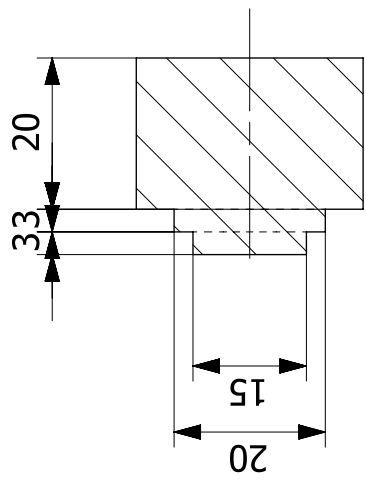
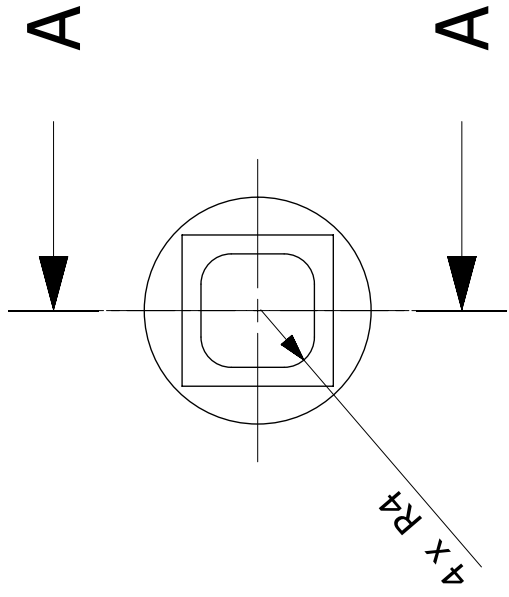


Antall	Pos nr.	Navn, type, dimensjon	Tegning produktstid	Materiale	kg pr. stk
Dato	Konstr./Tegnet	Prosjeksjonsmetode	Målestokk	Firmanavn	NTNU
			1:2		
Tittel			Erstatning for: Erstattet av:		
TEST2 - Deckel Maho			Tegn nr		
Henvisning		Beregning			

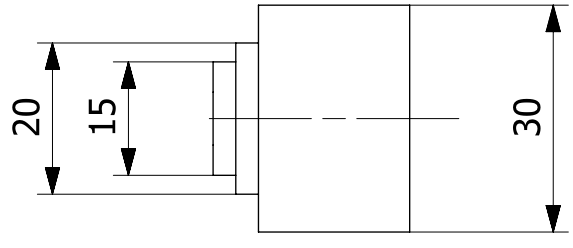
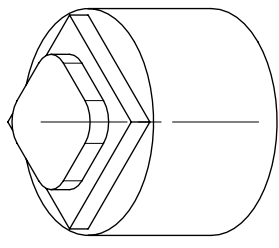


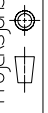
SNITT A-A

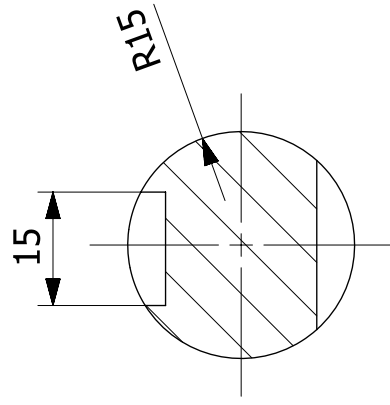
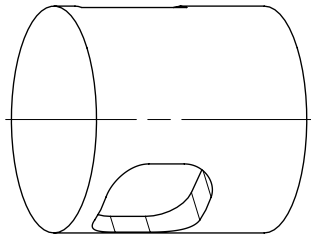
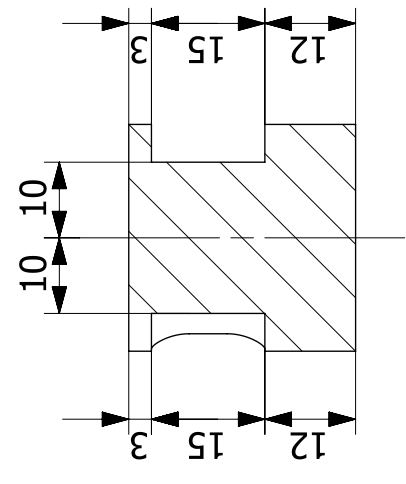
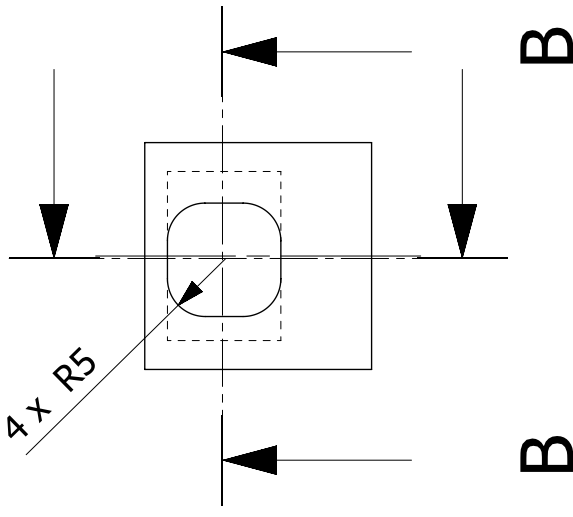
Antall	Pos. nr.	Navn, type, dimensjon	Tegning produktstid	Materiale	AI	kg pr. stk.
Dato	Konstr./Tegnet	Prosjeksjonsmetode 	Målestokk	Firmanavn	NTNU	
			1:1			
Tittel:		TEST1 - Okuma		Erstatning for:		Erstattet av:
				Tegn nr		
Henvisning			Beregning			



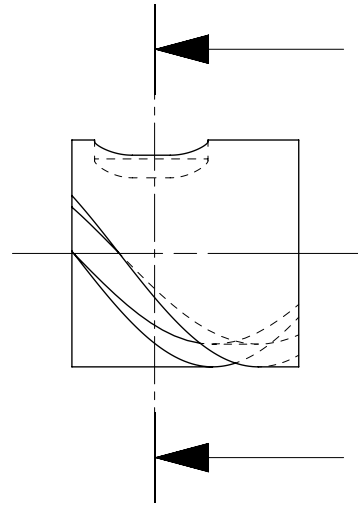
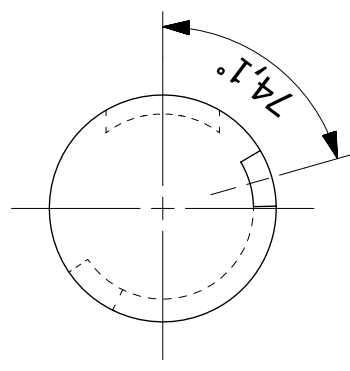
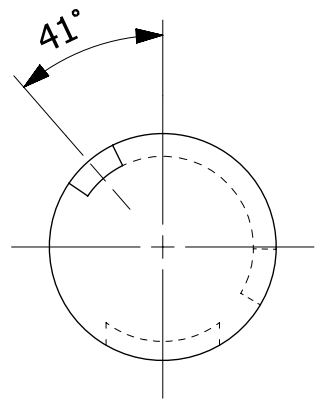
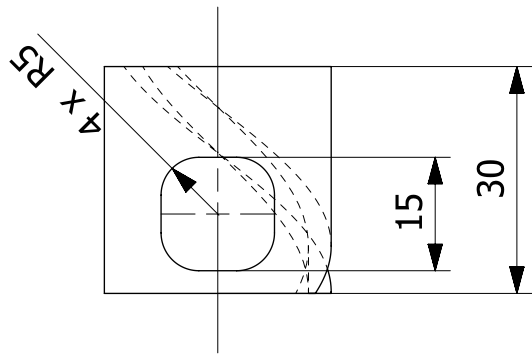
SNITT A-A



Antall	Pos nr.	Navn, type, dimensjon	Tegning produktstid	Materiale	AI	kg pr. stk
Dato	Konstr/Tegnet	Prosjeksjonsmetode 	Målestokk 1:1	Firmanavn NTNU		
Tittel		TEST2 - Okuma		Erstatning for:		Erstattet av:
Henvisning		Beregning		Tegn nr		

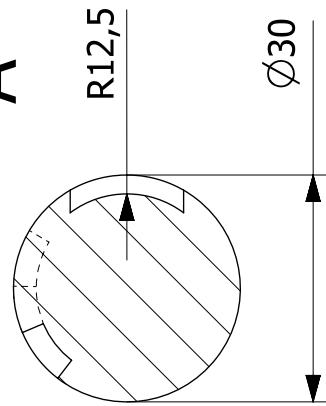


Anfall	Pos nr.	Navn, type, dimensjon	Tegning produktstid	Materiale	AI	kg pr. stk
Dato	Konstr/Tegnet	Projeksjonsmetode -1- ⊕	Målestokk	Firmanavn NTNU		
			1:1	Erstatning for: Erstattet av:		
Tittel		TEST3 - Okuma		Tegn nr		
Henvisning		Beregning				

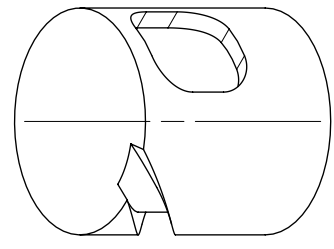


A

A



SECTION A-A



Antall	Pos nr.	Navn, type, dimensjon	Tegning produktstid	Materiale	kg pr. stk
		Konstr/Tegnet	Målestokk	AI	
Dato		Prosjeksjonsmetode	1:1	Firmanavn	NTNU
				Erstatning for:	Erstattet av:
Tittel		TEST4 - Okuma			
Henvisning		Beregning			

Tillegg E

Maskinkode

Vedlagt er maskinkode som ble generert i postprosessoren for NX Manufacture og postprosessoren for GibbsCam i test 3 for Okuma Genos.

Maskinkoden som ble generert i postprosessoren til GibbsCam er vedlagt i to eksemplarer, i uredigert form og i redigert form. Av plasshensyn ble maskinkodene som ble generert av postprosessoren til GibbsCam kortet ned, illustrert med ... i maskinkodene. Det ble fokusert på at minst mulig verdifull informasjon skulle bli fjernet i nedkortingene.

E.1 Maskinkode generert av postprossessor til NX Manufacture

```
( -----EMNE GRAFIKK----- )
N0010 DEF WORK
N0020 PS LC, [-60,0], [62,30]
N0030 END
N0040 CLEAR
N0050 DRAW
( - )
( -----NULLPUNKT----- )
N0060 EMNE=100.0 BAKE=53.0
N0070 CALL ONUL
( - )
( -----VARIABLER----- )
N0080 XVVP=999 ZVVP=222 ZSIK=10
N0090 CALL OVVP
( - )
N0100 G90
(-----FLOOR_WALL-----)
(-----FRESE_XY FRESE_XY FRESE_XY-----)
N0110 CALL OVVP
N0120 T020202 M08
N0130 M110
N0140 G119
N0150 M146
N0160 G138
N0170 M15
```

N0180 G0 G90 Y-17.18 Z-6.003 C90. SB=2000 M13
N0190 X18.
N0200 M147
N0210 X13.
N0220 G1 X10. F250. M08
N0230 Y-14.18
N0240 Y14.18
N0250 Y17.18
N0260 X13.
N0270 G0 X18.
N0280 Y-17.18 Z-9.002
N0290 X13.
N0300 G1 X10.
N0310 Y-14.18
N0320 Y14.18
N0330 Y17.18
N0340 X13.
N0350 G0 X18.
N0360 Y-17.18 Z-12.001
N0370 X13.
N0380 G1 X10.
N0390 Y-14.18
N0400 Y14.18
N0410 Y17.18
N0420 X13.
N0430 G0 X18.
N0440 Y-17.18 Z-15.
N0450 X13.
N0460 G1 X10.
N0470 Y-14.18
N0480 Y14.18
N0490 Y17.18
N0500 X13.
N0510 G0 X18.
N0520 Y0.0
N0530 G136
(-----FLOOR_WALL_IPW-----)
(-----FRESE_XY FRESE_XY FRESE_XY-----)
N0540 M146
N0550 G138
N0560 G0 Y-.5 Z-9. C270. SB=2653 M13
N0570 X25.
N0580 M147
N0590 X17.997
N0600 G1 X10. F100.
N0610 Y-2.

N0620 Z-12.
N0630 Y1.
N0640 Z-9.
N0650 Y-.5
N0660 Z-6.
N0670 Y-3.
N0680 Y-3.738 Z-6.141
N0690 Y-4.393 Z-6.565
N0700 Y-4.836 Z-7.207
N0710 Y-5. Z-8.
N0720 Z-13.
N0730 Y-4.875 Z-13.777
N0740 Y-4.435 Z-14.435
N0750 Y-3.777 Z-14.875
N0760 Y-3. Z-15.
N0770 Y2.005
N0780 Y2.741 Z-14.86
N0790 Y3.394 Z-14.437
N0800 Y3.837 Z-13.796
N0810 Y4. Z-13.005
N0820 Z-7.953
N0830 Y3.879 Z-7.194
N0840 Y3.449 Z-6.551
N0850 Y2.806 Z-6.121
N0860 Y2.047 Z-6.
N0870 Y-.5
N0880 Z-9.
N0890 X13.
N0900 G0 X25.
N0910 Y0.0
N0920 G136
N0930 M146
N0940 CALL OVVP
N0950 M02

E.2 Uredigert maskinkode generert av postprosessor til GibbsCam

```
( TOOLS LIST )
( T01 - )

( TEGNINGSNUMMER : )
( PROGRAMMERT : )
( ORDRE :XXXXX XXXXXXXXXXX XX XX.XX.XX XXXX )
( )
N1(PROGRAM RUNTIME = 0. HRS 1. MINS 36. SECS)
N2 M1
N1
N3 G50 S2000
N4 GO X999. Z999. M216
NT01
N5 GO G90 G94 T010101 M8
N6 M110
N7 SB=2000 M13
N8 GO C0.
N9 M15
N10 GO Z-7.158 C90.
N11 GO X50.
N12 G138
N13 Y11.178
N14 G19
N15 M147
N16 X25.
N17 G1 X12. F100.
N18 Y-11.178 F200.
N19 Z-12.238
N20 Y11.178
N21 GO X25.
N22 M146
N23 M1
N2
NT01
N24 GO G90 G94 T010101 M8
N25 M110
N26 SB=2000 M13
N27 F200.
N28 M15
N29 GO C90.
N30 G138
N31 G19
N32 GO G94 Y11.178 Z-7.158
N33 M147
```

N34 G1 X10. F100.
N35 Y-11.178 F200.
N36 Z-12.238
N37 Y11.178
N38 G0 X25.
N39 M146
N40 M1
N3
NT01
N41 G0 G90 G94 T010101 M8
N42 M110
N43 SB=2000 M13
N44 F200.
N45 M15
N46 G0 C270.
N47 G138
N48 G19
N49 G0 G94 Y-4.987 Z-8.987
N50 M147
N51 G1 X12.5 F100.
N52 Z-12.778 F200.
N53 Y-4.986 Z-13.046
N54 Y-4.962 Z-13.314
N55 Y-4.905 Z-13.567
N56 Y-4.9 Z-13.583
N57 Y-4.796 Z-13.851
N58 Y-4.642 Z-14.119
N59 Y-4.636 Z-14.128
N60 Y-4.412 Z-14.387
N61 Y-4.368 Z-14.435
N62 Y-4.099 Z-14.656
N63 Y-3.832 Z-14.805
N64 Y-3.563 Z-14.906
N65 Y-3.497 Z-14.924
N66 Y-3.295 Z-14.964
N67 Y-3.027 Z-14.987
N68 Y-2.759
N69 Y1.802
N70 Y2.07 Z-14.986
N71 Y2.338 Z-14.959
N72 Y2.501 Z-14.924
N73 Y2.607 Z-14.893
N74 Y2.875 Z-14.786
N75 Y3.101 Z-14.656
N76 Y3.143 Z-14.628
N77 Y3.411 Z-14.393

.
.
.
N235 Y-3.563 Z-14.906
N236 Y-3.497 Z-14.924
N237 Y-3.295 Z-14.964
N238 Y-3.027 Z-14.987
N239 Y-2.759
N240 G0 X25.
N241 Y0.
N242 M146
N243 G136
N244 G0 G18
N245 M9
N246 G0 X999. Z999.
N247 M12
N248 M109
N249 M2
\%

E.3 Redigert maskinkode generert av postprosessor til GibbsCam

```
( TOOLS LIST )
( T01 - )

( TEGNINGSNUMMER : )
( PROGRAMMERT : )
( ORDRE :XXXXX XXXXXXXXXX XX XX.XX.XX XXXX )
( )
( -----EMNE GRAFIKK----- )
N0010 DEF WORK
N0020 PS LC, [-60,0], [62,30]
N0030 END
N0040 CLEAR
N0050 DRAW
( - )
( -----NULLPUNKT----- )
N000 EMNE=100.0 BAKE=53.0
N000 CALL ONUL
( - )
N000 XVVP=999 ZVVP=222 ZSIK=10
N000 CALL OVVP
( )
N1(PROGRAM RUNTIME = 0. HRS 1. MINS 36. SECS)
N2 M1
N1
N3 G50 S2000
NT01
N5 GO G90 G94 T010101
N6 M110
N7 SB=2000 M13
N8 GO C0.
N9 M15
N10 GO Z-7.158 C90.
N11 GO X50.
N12 G138
N13 Y11.178
N14 G19
N15 M147
N16 X25.
N17 G1 X12. F100.
N18 Y-11.178 F200.
N19 Z-12.238
N20 Y11.178
N21 GO X25.
```

N22 M146
N23 M1
N2
NT01
N24 G0 G90 G94 T010101
N25 M110
N26 SB=2000 M13
N27 F200.
N28 M15
N29 G0 C90.
N30 G138
N31 G19
N32 G0 G94 Y11.178 Z-7.158
N33 M147
N34 G1 X10. F100.
N35 Y-11.178 F200.
N36 Z-12.238
N37 Y11.178
N38 G0 X25.
N39 M146
N40 M1
N3
NT01
N41 G0 G90 G94 T010101
N42 M110
N43 SB=2000 M13
N44 F200.
N45 M15
N46 G0 C270.
N47 G138
N48 G19
N49 G0 G94 Y-4.987 Z-8.987
N50 M147
N51 G1 X12.5 F100.
N52 Z-12.778 F200.
N53 Y-4.986 Z-13.046
N54 Y-4.962 Z-13.314
N55 Y-4.905 Z-13.567
N56 Y-4.9 Z-13.583
N57 Y-4.796 Z-13.851
N58 Y-4.642 Z-14.119
N59 Y-4.636 Z-14.128
N60 Y-4.412 Z-14.387
N61 Y-4.368 Z-14.435
N62 Y-4.099 Z-14.656
N63 Y-3.832 Z-14.805

N64 Y-3.563 Z-14.906
N65 Y-3.497 Z-14.924
N66 Y-3.295 Z-14.964
N67 Y-3.027 Z-14.987
N68 Y-2.759
N69 Y1.802
N70 Y2.07 Z-14.986
N71 Y2.338 Z-14.959
N72 Y2.501 Z-14.924
N73 Y2.607 Z-14.893
N74 Y2.875 Z-14.786
N75 Y3.101 Z-14.656
N76 Y3.143 Z-14.628
N77 Y3.411 Z-14.393
.
.
.
N235 Y-3.563 Z-14.906
N236 Y-3.497 Z-14.924
N237 Y-3.295 Z-14.964
N238 Y-3.027 Z-14.987
N239 Y-2.759
N240 G0 X25.
N241 Y0.
N242 M146
N243 G136
N244 G0 G18
N245 M9
N000 CALL OVVP
N247 M12
N248 M109
N249 M2
\%