

ZÁPADOČESKÁ UNIVERZITA V PLZNI
FAKULTA STROJNÍ

Studijní program: N0715A270011 - Obrábění, aditivní
technologie a zabezpečování kvality

Studijní specializace: Bez specializace

DIPLOMOVÁ PRÁCE

Možnosti uplatnění 3D korekce nástroje a její aplikace

Autor: Bc. Dominik LUST

Vedoucí práce: Ing. Jan HNÁTÍK, Ph.D.

Akademický rok 2023/2024

ZÁPADOČESKÁ UNIVERZITA V PLZNI

Fakulta strojní

Akademický rok: 2023/2024

ZADÁNÍ DIPLOMOVÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: **Bc. Dominik LUST**
Osobní číslo: **S21N0009P**
Studijní program: **N0715A270011 Obrábění, aditivní technologie a zabezpečování kvality**
Téma práce: **Možnosti uplatnění 3D korekce nástroje a její aplikace**
Zadávající katedra: **Katedra technologie obrábění**

Zásady pro vypracování

- Úvod
- Definice cílů
- Rozbor současného stavu
- Realizace úpravy postprocesoru pro vybraný stroj
- Test 3D korekcí na vybraném stroji
- Vyhodnocení testu
- Závěr

Rozsah diplomové práce: **50 – 70 stran**
Rozsah grafických prací: **dle potřeby**
Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam doporučené literatury:

- POLÁŠEK, Jaromír. Číslicově řízené stroje – online. Kopřivnice: VOŠ, SOŠ A SOU Kopřivnice cit. 2022-10-14. Dostupné z: http://moodle2.voskop.eu/download/teu/U31_Cislicove_rizene_stroje.pdf
- MÁDL, J. Teorie obrábění. Praha : ČVUT, 1994.
- HOFMANN, D. Priemyselná meracia technika. Bratislava : ALFA, 1988.
- TALÁCKO, J. Automatizace výrobních zařízení. Praha : skripta ČVUT, 1993.
- JANDEČKA, K. , ČESÁNEK, J. , KOŽMÍN, P. : Programování NC strojů. Plzeň: ZČU, 2000.
- MÁDL, J. Optimalizace řezných podmínek v teorii obrábění. Praha : ČVUT, 1990.
- SANDVIK COROMANT. Příručka obrábění. Praha : Scientia, 1997.
- Manuál řídicího systému Heidenhain iTNC530.
- SCHMIDT, D. a kolektiv: Řízení a regulace pro strojírenství a mechatroniku, Europa – Sobotáles, 2005.
- KORDINA, Michal. Využití 3D korekcí nástroje při obrábění tvarových ploch. Plzeň, 2011. diplomová práce (Ing.). ZÁPADOČESKÁ UNIVERZITA V PLZNI. Fakulta strojní.

Vedoucí diplomové práce: **Ing. Jan Hnátík, Ph.D.**
Katedra technologie obrábění

Konzultant diplomové práce: **Ing. Luboš Kroft, Ph.D.**
Regionální technologický institut

Datum zadání diplomové práce: **16. října 2023**
Termín odevzdání diplomové práce: **24. května 2024**

L.S.

Doc. Ing. Vladimír Duchek, Ph.D.
děkan

Doc. Ing. Miroslav Zetek, Ph.D.
vedoucí katedry

Prohlášení o autorství

Předkládám tímto k posouzení a obhajobě diplomovou práci zpracovanou na závěr studia na Fakultě strojní Západočeské univerzity v Plzni.

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně, s použitím odborné literatury a pramenů uvedených v seznamu, který je součástí této diplomové práce.

V Plzni dne:

.....

podpis autora

Poděkování

Tímto bych chtěl poděkovat vedoucímu mé diplomové práce panu Ing. Janu Hnátíkovi, Ph.D. za ochotu, konzultace a všechny odborné rady při tvorbě této práce.

ANOTAČNÍ LIST DIPLOMOVÉ PRÁCE

AUTOR	Příjmení Lust	Jméno Dominik	
STUDIJNÍ PROGRAM	N0715A270011 - Obrábění, aditivní technologie a zabezpečování kvality		
VEDOUCÍ PRÁCE	Příjmení (včetně titulů) Ing. Hnátík, Ph.D.	Jméno Jan	
PRACOVISŤE	ZČU - FST - KTO		
DRUH PRÁCE	DIPLOMOVÁ	BAKALÁŘSKÁ	Nehodící se škrtněte
NÁZEV PRÁCE	Možnosti uplatnění 3D korekce nástroje a její aplikace		

FAKULTA	strojní	KATEDRA	KTO	ROK ODEVZD.	2024
----------------	---------	----------------	-----	--------------------	------

POČET STRAN (A4 a ekvivalentů A4)

CELKEM	75	TEXTOVÁ ČÁST	52	GRAFICKÁ ČÁST	23
---------------	----	---------------------	----	----------------------	----

STRUČNÝ POPIS (MAX 10 ŘÁDEK) ZAMĚŘENÍ, TÉMA, CÍL POZNATKY A PŘÍNOSY	Diplomová práce se zabývá rozborem 3D korekce nástroje a úpravou postprocesoru. Cílem je upravit postprocesor pro vybraný stroj tak, aby byla zprovozněna 3D korekce nástroje. Je proveden test vygenerovaného NC programu pomocí upraveného postprocesoru. Součástí s 3D korekcí je změřena a je provedeno vyhodnocení testu.
KLÍČOVÁ SLOVA ZPRAVIDLA JEDNOSLOVNÉ POJMY, KTERÉ VYSTIHUJÍ PODSTATU PRÁCE	obrábění, 3D korekce nástroje, postprocesor, normálový vektor povrchu, řídicí systém, kulová fréza, NC program, CAM systém

SUMMARY OF DIPLOMA SHEET

AUTHOR	Surname Lust	Name Dominik		
STUDY PROGRAMME	N0715A270011 - Machining, Additive technology and Quality Assurance			
SUPERVISOR	Surname (Inclusive of Degrees) Ing. Hnátík, Ph.D.	Name Jan		
INSTITUTION	ZČU - FST - KKS			
TYPE OF WORK	DIPLOMA	BACHELOR	Delete when not applicable	
TITLE OF THE WORK	Possibilities of applying 3D tool compensation and their application			

FACULTY	Mechanical Engineering	DEPARTMENT	Machine Design	SUBMITTED IN	2024
----------------	------------------------	-------------------	----------------	---------------------	------

NUMBER OF PAGES (A4 and eq. A4)

TOTALLY	75	TEXT PART	52	GRAPHICAL PART	23
----------------	----	------------------	----	-----------------------	----

BRIEF DESCRIPTION TOPIC, GOAL, RESULTS AND CONTRIBUTIONS	The thesis deals with the analysis of 3D tool correction and postprocessor modification. The aim is to modify the postprocessor for the selected machine in order to make the 3D tool correction work. The test of the generated NC program using the modified postprocessor is performed. The part with 3D correction is measured and the test is evaluated.
KEY WORDS	machining, 3D tool correction, postprocessor, surface normal vector, control system, ball mill, NC program, CAM system

Obsah

Přehled použitých zkratk a symbolů.....	10
Seznam obrázků	11
Seznam tabulek	13
Úvod.....	14
1 Rozbor současného stavu	15
1.1 2D korekce nástroje.....	15
1.1.1 Poloměrová korekce nástroje při soustružení.....	15
1.1.2 Korekce nástroje při frézování	16
1.2 3D korekce nástroje.....	17
1.2.1 Metoda 3D korekce poloměru prostorové frézy pro čelní frézování volných ploch.....	18
1.2.2 Vektor poloměru frézy a vektor směru	19
1.3 Schéma procesu výroby součásti.....	20
1.4 CAM systém podporující 3D korekci nástroje.....	21
1.4.1 HSS.....	21
1.4.2 Souvislé víceosé frézování	21
1.4.3 Postup zapnutí 3D korekce nástroje v SolidCAM 2020	21
1.5 Postprocesor	23
1.5.1 Struktura GPP.....	23
1.5.2 Typy proměnných	24
1.5.3 Podmínka v GPP	25
1.5.4 Způsob zobrazení proměnných	26
1.6 Řídicí systém stroje	27
1.6.1 Trojrozměrná korekce nástroje v Heidenhain TNC 530	27
2 Realizace úpravy postprocesoru pro vybraný stroj	37
2.1 Součást pro výrobu.....	37
2.2 Obrobení součásti v SolidCAM	38
2.3 Postprocesor a SolidCAM.....	40
2.4 SolidCAM pomocník pro GPP.....	41
2.5 Nastavení Visual Studio Code.....	44
2.5.1 Instalace českého jazyka	44
2.5.2 Instalace rozšíření pro GPP	44
2.5.3 Nastavení kódování znaků	47
2.6 Data pro 3D korekci	48

2.7	Úprava postprocesoru.....	49
2.7.1	Pochopení příkazu @line v postprocesoru.....	50
2.7.2	Testování proměnných normálového vektoru.....	54
2.7.3	Úprava @line pro 3D korekce.....	55
2.7.4	Nastavení normálového vektoru na 7 desetinných míst.....	56
2.7.5	Podmínka pro 3D korekci	58
2.7.6	Testování proměnných pro 3D korekci	59
2.7.7	Přidání podmínky do @line	62
3	Test postprocesoru s 3D korekcí	64
3.1	Naměření kulové frézy	65
3.2	Obrábění	66
3.2.1	Hrubování.....	66
3.2.2	Dokončení kulové plochy s 3D korekcí	67
3.2.3	Dokončení kulové plochy bez 3D korekce	67
3.2.4	Dokončení dna kulovou frézou	68
4	Vyhodnocení testu.....	69
4.1	Měření nástroje.....	69
4.2	Měření obrobené součásti sondou	69
4.3	Výsledky měření	70
4.4	Analýza dle naměřených rozměrů.....	71
5	Závěr.....	73
	Seznam použitých zdrojů	74
	PŘÍLOHA č. 1.....	i

Přehled použitých zkratk a symbolů

2D	Dvourozměrný
3D	Trojrozměrný
CNC	Computer numerical control – Číslicové řízení stroj
CC	Bod kontaktu nástroje s obrobkem
CL	Cutter Location - Bod středu nástroje
TCPM	Tool Center Point Management - řízení středu nástroje
GPP	General Post Processor - Soubor postprocesoru
GPPL	General Post Processor Language - Jazyk postprocesoru
NC	Numeric Control - číslicově řízený
TC	Tool Center - proměnná pro korekci na střed
TT	Tool Tip - proměnná pro korekci na špičku
CAD	Computer aided design - Počítačem podporované projektování
CAM	Computer aided manufacturing - Počítačem podporovaná výroba

Seznam obrázků

Obr. 1: Problematika korekce zaoblení špičky nástroje při soustružení [1].....	15
Obr. 2: Znázornění 2D korekce při frézování [3].....	16
Obr. 3: Roviny v kartézském souřadném systému [5]	16
Obr. 4: Nájezd frézy při 2D poloměrové korekci [4].....	17
Obr. 5: Schématické znázornění 3D korekce za použití kulové frézy [3].....	18
Obr. 6: Normálový vektor povrchu v různých místech kontaktu kulové frézy s obrobkem [12]	18
Obr. 7: Postup provedení funkce 3D korekci poloměru nástroje v pětiosém CNC systému [8]	19
Obr. 8: Schématický diagram, který znázorňuje tvorbu NC programu [10].....	20
Obr. 9: Zapnutí 3D korekce v systému SolidCAM [SolidCAM 2020].....	22
Obr. 10: Struktura podmínky v GPPL [15].....	25
Obr. 11: Nastavení proměnné v GPP	26
Obr. 12: Přehled možných fréz pro 3D korekci v Heidenhain iTNC 530 [11]	28
Obr. 13: Schématické znázornění špičky a středu kulové frézy [11].....	29
Obr. 14: Schématické znázornění hodnot R2 a DR2 [11].....	29
Obr. 15: Směr normálového vektoru plochy [6]	30
Obr. 16: Sklon osy nástroje [6]	31
Obr. 17: Odchyłka kulové frézy od ideálního tvaru [13]	34
Obr. 18: Znázornění nepřesnosti nástroje [11].....	34
Obr. 19: Znázornění kulové frézy od 0° do 90° [11]	35
Obr. 20: Interpolace hodnot pro určité úhly	36
Obr. 21: Vymodelovaná součást pro obrábění v SolidCAMu	37
Obr. 22: Dráhy nástroje při operaci 3D HSR technologie HM hrubování v SolidCAM.....	38
Obr. 23: Dráhy nástroje pro kulovou plochu.....	39
Obr. 24: Korekce kulové frézy na špičku a na střed [9].....	40
Obr. 25: Povolení frézy pro 3D nástrojovou korekci v SolidCAMu [18].....	41
Obr. 26: Stránka pro vysvětlení GPP kódu, zobrazena záložka Contents [15].....	41
Obr. 27: Postup pro nápovědu pro GPP v SolidCAM [SolidCAM 2020]	42
Obr. 28: Postup pro nápovědu pro GPP v SolidCAM [SolidCAM 2020]	43
Obr. 29: Instalace českého jazyka ve Visual Studio Code [Visual Studio Code].....	44
Obr. 30: Vyhledání rozšíření pro GPP přímo ve Visual Studio Code [Visual Studio Code]...	45
Obr. 31: Instalace rozšíření pro GPP ve Visual Studio Code [Visual Studio Code].....	45
Obr. 32: Nainstalované rozšíření ve Visual Studio Code [Visual Studio Code].....	46

Obr. 33: Zjednodušené vyhledávání procesů s rozšířením editor toolkit [Visual Studio Code]	46
Obr. 34: Nesprávné zobrazení znaků po otevření souboru GPP ve Visual Studio Code	47
Obr. 35: Postup pro nastavení ve Visual Studio Code [Visual Studio Code]	47
Obr. 36: Postup pro nastavení dekódovače ve Visual Studio Code [Visual Studio Code]	48
Obr. 37: Funkce trace v postprocesoru	48
Obr. 38: Zapnutá funkce trace	48
Obr. 39: Dialogové okno pro nastavení hodnoty trace v SolidCAM 2020 [SolidCAM 2020]	49
Obr. 40: Normálový vektor v NC programu po nastavení trace na hodnotu 5	49
Obr. 41: Výběr stroje [SolidCAM 2020]	50
Obr. 42: Výběr postprocesoru [SolidCAM 2020]	50
Obr. 43: První část procesu @line	51
Obr. 44: Nastavení proměnných s xlpos, ylpos a zlpos	51
Obr. 45: Nastavení proměnných s xpos, ypos a zpos	52
Obr. 46: NC program bez 3D korekce	53
Obr. 47: Třetí část procesu @line	53
Obr. 48: Příkaz pro vypsání normálového vektoru	54
Obr. 49: Výpis normálových vektorů do NC programu	54
Obr. 50: Vytvoření proměnných pro normálový vektor	55
Obr. 51: Upravená druhá část @line	55
Obr. 52: Upravená proměnná S_line	56
Obr. 53: Vygenerovaná část programu s LN a normálovým vektorem	56
Obr. 54: Nastavení x_surface_normal v @unit_post	56
Obr. 55: Chybová hláška po nastavení x_surface_normal v @unit_post	56
Obr. 56: Úprava pro 7 desetinných míst	57
Obr. 57: Vygenerovaný NC program po úpravě na 7 desetinných míst	57
Obr. 58: Upravený postprocesor pro vypsání koncových nul pro normálový vektor	57
Obr. 59: Upravený NC program s koncovou nulou	58
Obr. 60: Struktura podmínky [15]	58
Obr. 61: Přidání proměnných pro špičku a střed nástroje do @init_post	59
Obr. 62: Část programu pro otestování hodnot pro 3D korekci	59
Obr. 63: Hodnoty proměnných při zapnuté 3D korekci na špičku	59
Obr. 64: Zapnutá korekce na střed	59
Obr. 65: Testování podmínky s type_offset_3D	60
Obr. 66: Přidání podmínky do procesu @compensation_3d	60

Obr. 67: Upravený GPP pro otestování podmínky	61
Obr. 68: Upravená podmínka v @compensation_3d	62
Obr. 69: Podmínka pro @line	62
Obr. 70: Upravený kód při korekci na špičku	63
Obr. 71: pětiosé obráběcí centrum DMU eVo linear DMG MORI	64
Obr. 72: Zadání hodnot pro automatické měření do tabulky nástrojů	65
Obr. 73: Naměřená odchylna v tabulce nástrojů	65
Obr. 74: Obrobek po hrubování	66
Obr. 75: Obrobek po dokončení s 3D korekcí	67
Obr. 76: Obrobek po dokončení kulové plochy bez 3D korekce	67
Obr. 77: Kompletně obrobená testovací součást	68
Obr. 78: Změřená kulová fréza a vynesení přímek pro jednotlivé úhly	69
Obr. 79: Výsledky měření laserovým snímáním povrchu	70
Obr. 80: Naměřené body CPM	71

Seznam tabulek

Tab. 1: Typy proměnných v GPP [16]	24
Tab. 2: Vysvětlení funkcí některých důležitých proměnných [15]	25
Tab. 3: Příklad naměřených hodnoty pro určité úhly a interpolace pro 45°	36
Tab. 4: Proměnná tool_tip [15]	59
Tab. 5: Všechny možnosti zapnutí 3D korekce	61
Tab. 6: Parametry stroje DMU eVo linear DMG MORI [19]	64
Tab. 7: Přehled rezných podmínek	66
Tab. 8: Měření sondou vrcholu kulových ploch po obrábění	69
Tab. 9: Naměřené hodnoty na obou polokoulích	71

Úvod

2D korekce nástroje je v obrábění velmi hojně využívána a to primárně k ulehčení programování. Naproti tomu 3D korekce nástroje je stále málo rozšířenou a využívanou funkcí navzdory jejím benefitům a rozvoji víceosého obrábění a jejím zakomponování do některých CAM systémů a řídicích systémů CNC strojů. V současné době i přesto, že některé CAM systémy mají možnost zapnutí 3D korekce nástroje, ve většině případů při jejím zapnutí v CAM systému nemá žádný vliv na výsledný NC program, protože každý CNC stroj má svůj specifický postprocesor a řídicí systém. S rostoucím trendem zvyšujících se požadavků na tvarovou a rozměrovou přesnost výroby je možné, že implementace 3D nástrojové korekce bude do budoucna stále více důležitá.

Tato práce je zaměřena na problematiku využití 3D korekce nástroje. Cílem je upravit postprocesor pro vybraný stroj tak, aby byla zprovozněna 3D korekce nástroje. V první části práce je popsána funkce 2D korekce a následně 3D korekce nástroje, jejich rozdíly a k čemu jsou určeny. Dále je nastíněno schéma procesu výroby a na jeho základě jsou rozebrány všechny části, které jsou podstatné k tomu, aby 3D korekce nástroje mohla fungovat. K těmto částem patří popis zvoleného CAM programu. Následují informace o postprocesoru související s implementací trojrozměrné korekce. Dále je popsán vybraný řídicí systém stroje v souvislosti s 3D korekcí.

Druhá část práce je zaměřená na vymodelování a obrobení součásti v CAD/CAM systému pro otestování funkčnosti programu s 3D korekcí. Dále je zde popsána celá úprava postprocesoru, aby bylo možné vygenerovat NC program, který má zakomponovanou 3D korekcí.

Ve třetí části je popsáno samotné otestování vygenerovaného programu s 3D korekcí vyrobením testovací součásti.

Poslední část práce se zabývá vyhodnocením měření součásti a celkové zhodnocení testu.

1 Rozbor současného stavu

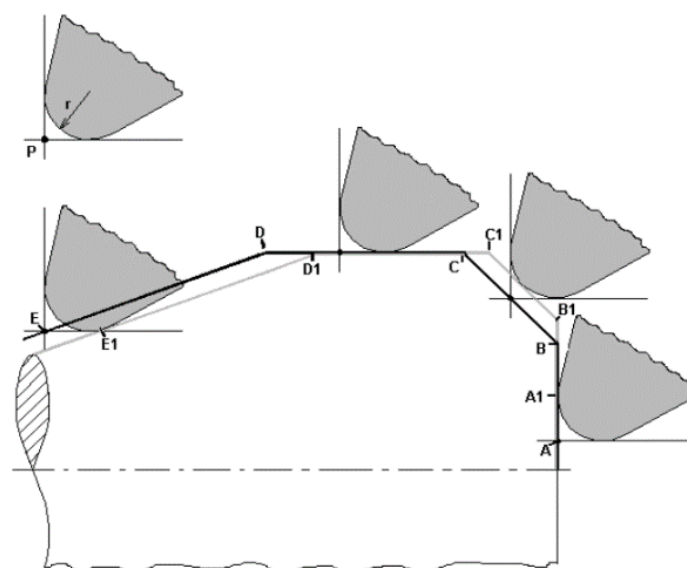
Korekce ve dvourozměrném prostoru se používá jak pro soustružení, tak pro frézování. Pro 2D korekce je volena rovina, ve které má být korekce provedena. Oproti tomu trojrozměrná korekce je výhradně používána pouze pro frézování a zde už nelze pouze vybrat rovinu, na které má být provedena korekce zprava nebo zleva, protože se jedná o 3D prostor, kde je nekonečně mnoho rovin. K tomuto je zásadní normálový vektor plochy v každém bodě, aby byla korekce provedena. Nejprve je důležité pochopit to, jakým způsobem funguje 3D korekce a jaký je rozdíl oproti běžně používané 2D korekci. Nejprve je tedy rozebrána běžně používaná dvojrozměrná korekce nástroje a následně trojrozměrná korekce. Dále je rozebráno to, co všechno je potřeba k tomu, aby mohla být 3D korekce zprovozněna.

1.1 2D korekce nástroje

Korekce ve dvourozměrném prostoru se používá jak pro soustružení, tak pro frézování. Pokud se jedná o soustružení, je korekce realizována v osách X a Z. V případě frézování je situace o něco složitější, protože zde je na výběr ze tří možných rovin G17 (rovina XY), G18 (rovina ZX) nebo G19 (rovina YZ). V následujících kapitolách jsou obě tyto varianty popsány.

1.1.1 Poloměrová korekce nástroje při soustružení

Soustružnický nůž má rádius zaoblání špičky, který je v řádu několika desetin milimetru. Obr. 1 znázorňuje problematiku korekce špičky soustružnického nože. Bod P je označení pro teoretickou špičku nástroje. Nástroj je na tento bod programován. Nástroj má začátek v bodě A a končí v bodě E. Když není aktivována korekce pro soustružnický nůž, nástroj nekopíruje dráhu B až C ani D až E. Místo toho, aby nástroj následoval dráhu od B do C, jeho reálná dráha je od bodu B1 do bodu C1 což znamená, že obrábí nad požadovaným obrysem součásti. Naproti tomu namísto toho, aby kopíroval dráhu od bodu D do bodu E, reálná dráha nástroje je od bodu D1 do bodu E1, tedy soustružnický nůž podřezává požadovaný obrys obrobku. Aby bylo docíleno toho, že nástroj bude obrábět správně, je zařazována korekce nástroje. K tomu je využívána funkce G41 a G42. U těchto funkcí je důležité, že G41 je určena pro levou korekci nástroje a G42 pro pravou korekci nástroje. Korekce se nevypíná sama od sebe, proto je nutné ji vypnout funkcí G40. [1]

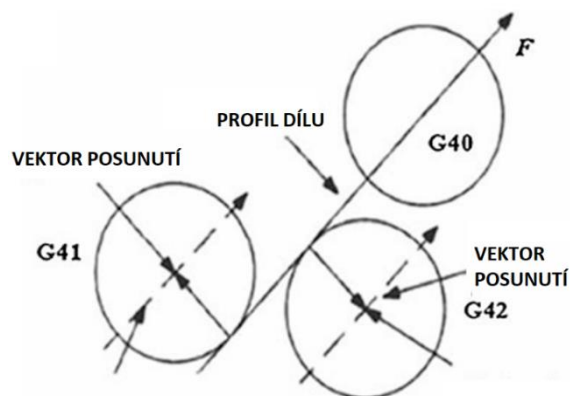


Obr. 1: Problematika korekce zaoblání špičky nástroje při soustružení [1]

1.1.2 Korekce nástroje při frézování

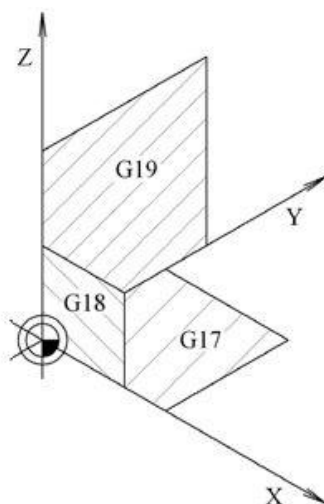
Stejně jako při soustružení, tak při frézování je přímo programován obrys vyráběné součásti. Pro správnou pozici nástroje je využívána korekce nástroje. V případě, že je naprogramován střed nástroje na obrys, korekce posune naprogramovanou dráhu nástroje o velikost jeho poloměru, aby bylo docíleno toho, že bok nástroje bude kopírovat požadovanou dráhu. U dvourozměrné korekce při frézování se jedná o posunutí středu nástroje o hodnotu jeho poloměru, aby bylo ulehčeno programování.

V kapitole 1.1.1 bylo zmíněno, jaké funkce jsou potřeba k tomu, aby byla zapnutá a vypnutá 2D korekce. U frézování pro aktivaci korekce se používají stejné funkce G41 a G42 jako u soustružení, kde G41 je používána pro korekci zleva a G42 pro korekci zprava a G40 pro zrušení korekce. Pokud však funkce pro korekci bude mít zápornou hodnotu dojde k jejich prohození. 2D korekce při frézování je znázorněna na Obr. 2. [3]



Obr. 2: Znázornění 2D korekce při frézování [3]

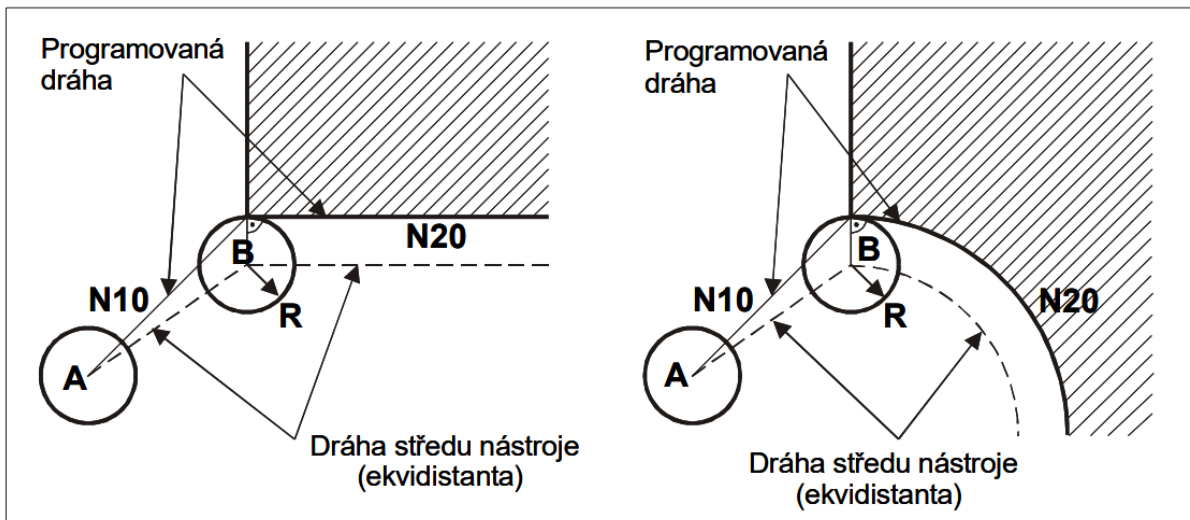
Je zde možné si zvolit ze tří korekčních rovin a to G17, G18 nebo G19. Obr. 3 ilustruje roviny v kartézském souřadném systému.



Obr. 3: Roviny v kartézském souřadném systému [5]

Na Obr. 4 je znázorněno, jak probíhá zařazení dvojrozměrné korekce při frézování. Jedná se o korekci zprava a nástroj má startovací bod A kde není zapnutá korekce. Cílový bod B je kolmý k dráze bloku N20 a zde již je zapnutá korekce. Čárkovaná čára značí dráhu, po které se nástroj pohybuje, když přechází z bodu bez korekce do bodu s korekcí. Levý obrázek

znázorňuje korekci po přímce a pravý obrázek znázorňuje korekci po kružnici. Korekci je možné použít jak pro lineární, tak pro kruhovou interpolaci. [4]



Obr. 4: Nájezd frézy při 2D poloměrové korekci [4]

1.2 3D korekce nástroje

V předchozí kapitole bylo zmíněno, že dvojrozměrná korekce je používána jak pro soustružení, tak pro frézování. Naproti tomu trojrozměrná korekce je využívána pouze při frézování. Při 2D korekci u frézování je zvolena jedna ze tří rovin G17, G18 nebo G19, na které má být korekce provedena, což znamená že je uskutečněna vždy pouze ve dvou osách.

Při 3D korekci je situace rozdílná, protože už se nejedná pouze o jednu ze tří rovin ve 2D prostoru, ale zde může být zvolena jakákoliv rovina ve 3D prostoru. V současné době existují práce na téma zprovoznění 3D nástrojové korekce, přesto je tato funkce velmi málo využívána.

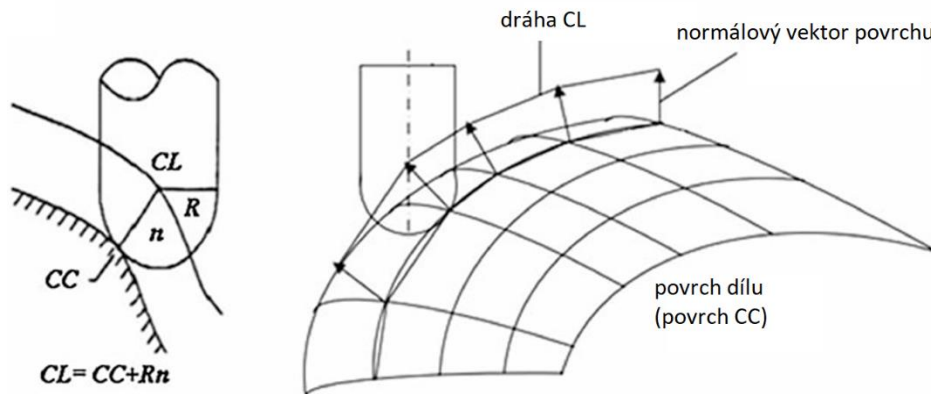
Rozdíl mezi dvojrozměrnou a trojrozměrnou korekcí je v tom, že 2D korekce posouvá střed nástroje o hodnotu rádiusu, aby bylo ulehčeno programování. U 3D korekce nejde pouze o posunutí o rádius, ale je například možné kompenzovat nepřesnosti na celém rádiusu kulové frézy, což při 2D korekci není možné provést. 3D korekce může být použita i pro víceosé obrábění, tedy pro čtyřosé a pětiosé frézování, což znamená při výrobě tvarově složitých součástí.

Pro docílení přesné výroby je velmi zásadní přesné nastavení nástrojů. Při obrábění dochází k opotřebení nástroje a změně rozměrů nástroje. Nejzásadnější vliv na vznik chyby mají řezné podmínky, deformace nástroje a deformace obrobku. Pokud chceme minimalizovat či zamezit negativní vliv opotřebení nástroje na výsledný výrobek, je potřeba nástroj vyměnit za nový nebo chybu, která vznikla opotřebením kompenzovat. Výměna nástroje způsobuje finanční a časovou ztrátu. Pokud by byla potřeba znovu generovat NC program, kvůli tomu, abychom kompenzovali chybu způsobenou například opotřebením nástroje, tak se prodlužuje čas výroby. Ke kompenzaci chyby přímo v bodě dotyku frézy s obrobkem pro tvarově složití plochy je určena 3D korekce. V následující kapitole je rozebrán princip 3D korekce nástroje. [11]

1.2.1 Metoda 3D korekce poloměru prostorové frézy pro čelní frézování volných ploch

Na Obr. 5 ilustrována problematika 3D korekce při obrábění volné plochy kulovou frézou. Na levé straně obrázku jsou zobrazeny parametry kulové frézy. K těmto parametrům patří bod CL, který představuje střed kulové frézy, dále je zde bod CC, pro místo dotyku nástroje s obrobkem. Kromě toho je zde rádius nástroje R a normálový vektor n. Mezi těmito parametry platí vztah (1). Tento vztah říká, že střed frézy CL se získá tím, že se posune místo dotyku frézy směrem normálového vektoru povrchu [3].

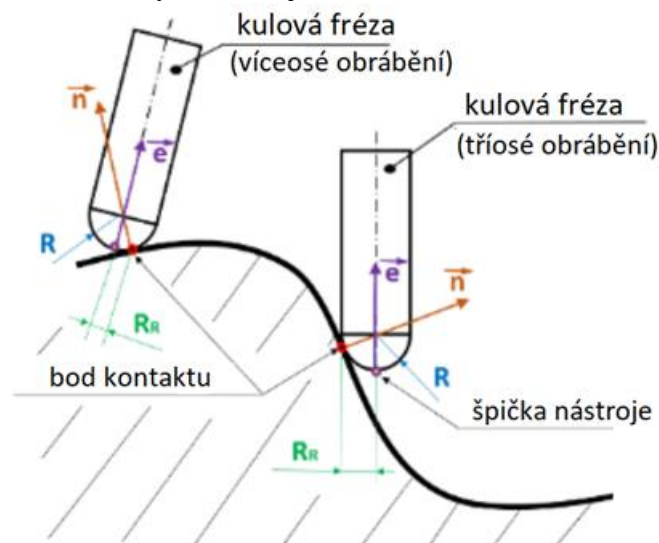
$$CL = CC + Rn \quad (1)$$



Obr. 5: Schématické znázornění 3D korekce za použití kulové frézy [3]

Na pravé části je pohyb nástroje bodem dotyku CC po povrchu dílu. Je zde dráha středu nástroje CL. Na pravé části obrázku je nejdůležitější to, že zde je normálový vektoru povrchu, který je pro korekci ve trojrozměrném prostoru klíčový. Podle normálového vektoru má řídicí systém informace o tom, jakým směrem má provést korekci, v tomto případě se jedná o kulovou frézu. [3]

Obr. 6 ilustruje to, jakým způsobem se mění normálový vektor povrchu s tím, jak je mění poloha nástroje. Jsou zde zobrazeny dvě kulové frézy. Fréza na levé části je při víceosém obrábění, protože je natočená orientace nástroje. Na pravé části se jedná o tříosé obrábění, kde se nemění orientace nástroje. Z tohoto důvodu je normálový vektor povrchu klíčový pro 3D korekci, protože se v každém dotyku nástroje s obrobkem mění.



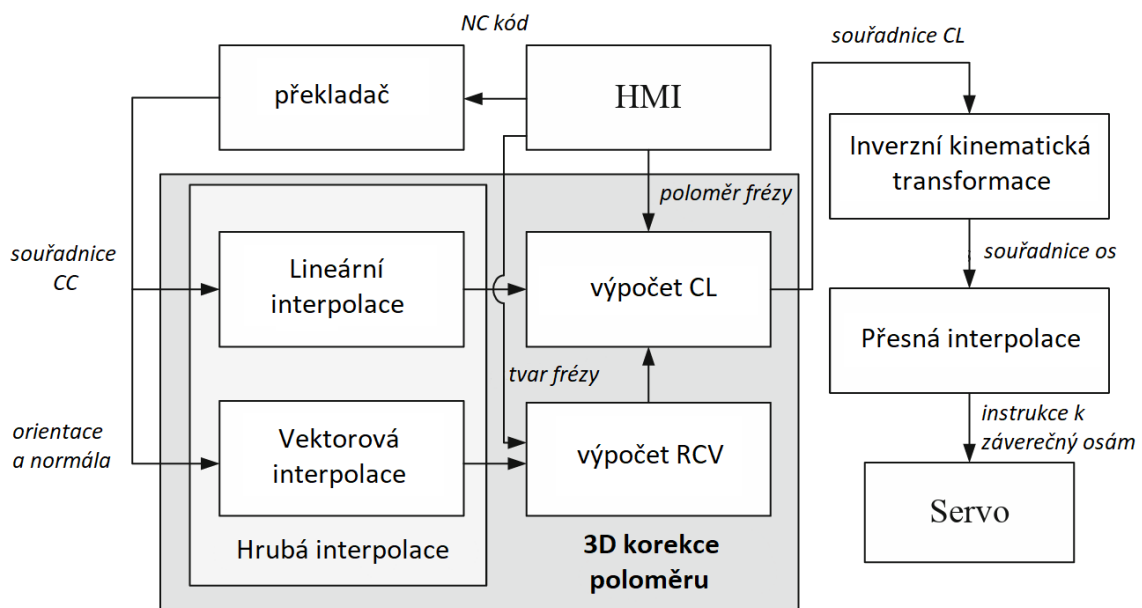
Obr. 6: Normálový vektor povrchu v různých místech kontaktu kulové frézy s obrobkem [12]

1.2.2 Vektor poloměru frézy a vektor směru

Ohledně aplikace a zadávání 3D korekce nástroje je důležité pochopit jaké hodnoty a parametry jsou důležité, aby korekce ve trojrozměrném prostoru mohla fungovat.

Vektor představující poloměr frézy je definován jako vektor, který je nasměřován směrem ke středu frézy během operace obrábění. Zachovává si kolmou orientaci vzhledem k naprogramované dráze s velikostí poloměru frézy. Směrový vektor je naproti tomu jednotkový vektor, který se zarovnává s pohybem samotné frézy. Pokud jde o oblouk, vektor směru oblouku je k němu tečný a mění se v každém bodě podél dráhy oblouku, což zajišťuje, že se jeho orientace neustále přizpůsobuje zakřivení oblouku v každém konkrétním bodě. [2]

Postup provedení 3D korekce je také zásadní. Obr. 7 ukazuje provozní sekvenci 3D korekci poloměru nástroje v reálném čase v rámci pětiosého obrábění. Zpočátku jsou části NC kódu převedeny do strojově čitelných direktiv. Systém provádí lineární interpolaci těchto přeložených instrukcí a současně vypočítává vektor korekce poloměru pro každý interpolovaný bod. Využitím vektoru korekce poloměru frézy se kromě souřadnice CC odvodí souřadnice CL. Zkratka HMI (Human-machine interface) ve schématu znamená rozhraní mezi člověkem a strojem. Dále RCV (Radius compensation vector) znamená vektor korekce poloměru. Po provedení přesné interpolace za účelem zpřesnění těchto pohybů systém vydá definitivní pokyny pro osy do servomechanismu, čímž aktivuje pohyb stroje. [8]

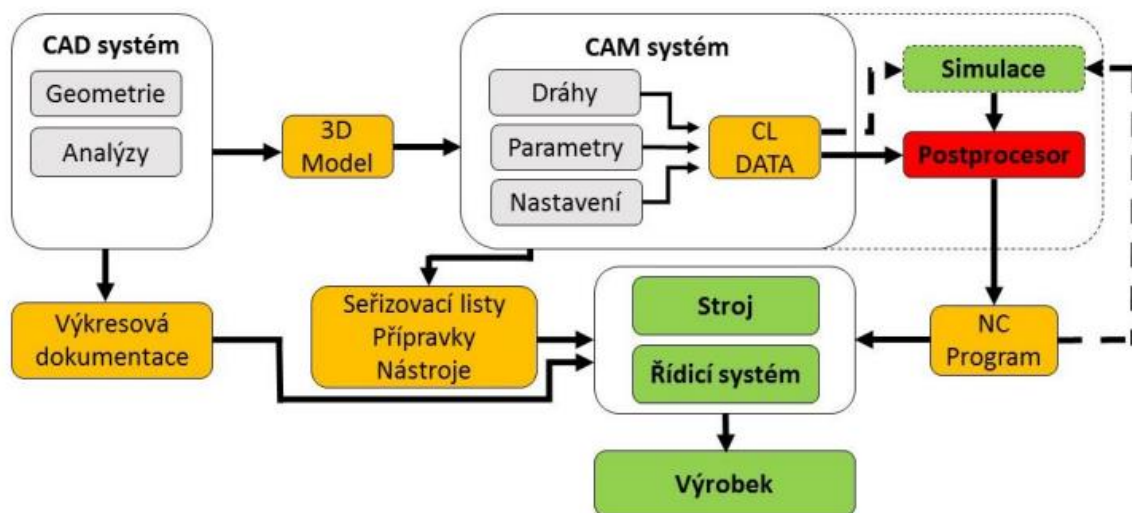


Obr. 7: Postup provedení funkce 3D korekci poloměru nástroje v pětiosém CNC systému [8]

1.3 Schéma procesu výroby součásti

Pro určení a pochopení toho, co vše je potřeba k zprovoznění a aplikaci 3D korekce nástroje, je důležité pochopit celý proces výroby součásti. Pro zprovoznění 3D nástrojové korekce je potřeba vygenerovat NC program pomocí CAM softwaru, kterému bude rozumět řídicí systém stroje. Začátek výroby součásti začíná u výkresu a vymodelování součásti v CAD systému. Poté následuje obrobení součásti v CAM systému, který generuje CL data. K tomu, aby CL data byla převedena do NC programu, je kritickým prvkem postprocessor, který potřebuje řídicí systém stroje, aby mohl vyrobit požadovaný výrobek. NC program se nahraje do řídicího systému stroje a pokračuje se simulací výroby v řídicím systému stroje a následuje výroba součásti.

Na Obr. 8 je schématický diagram, který znázorňuje celý proces výroby. Prvním krokem je zhotovení modelu součásti v CAD softwaru pomocí výkresové dokumentace. Následně se využívá CAM systém, který generuje dráhy a parametry nástroje pro obrábění. CAM systém poskytne tyto informace do CL dat, což jsou údaje o poloze nástroje. V CAM systému je možné provést virtuální simulaci a provést ověření obrábění a zamezení chyb. CL data jsou poslána do postprocesoru, který převede tyto data na NC program pro řídicí systém stroje. [10]



Obr. 8: Schématický diagram, který znázorňuje tvorbu NC programu [10]

Podle schématu bylo možné odvodit kroky potřebné k vytvoření NC programu s 3D korekcí:

1. Volba CAD/CAM systému, který podporuje 3D korekci nástroje.
2. Výběr řídicího systému a stroje, který umožňuje 3D korekci nástroje.
3. Úprava postprocesoru pro vybraný řídicí systém a stroj.

1.4 CAM systém podporující 3D korekci nástroje

Některé CAM systémy mají funkci generování potřebných parametrů pro 3D korekci nástroje. Jak bylo zmíněno klíčovým parametrem, který je zásadní je normálový vektor povrchu, aby řídicí systém stroje věděl, jakým směrem má provést korekci v konkrétním bodě dotyku nástroje s obrobkem. Tento normálový vektor musí CAM systém generovat pro všechny osy X, Y a Z. Současný stav je takový, že pouze některé CAM systémy podporují trojrozměrnou korekci nástroje. Z těchto CAM systémů jenom některé poskytnou takový program s 3D korekcí nástroje, kterému by rozuměl řídicí systém stroje. [14]

V této práci byl jako CAM systém použit program SolidCAM 2020. Tento systém u některých funkcí frézování umožňuje v pokročilých funkcích modifikovat dráhu nástroje a zde přidat 3D korekci na špičku nebo střed nástroje. Pokročilé možnosti, které upravují dráhu nástroje jsou u jednotlivých operací v záložce parametry dráhy nástroje. Pokročilé funkce úpravy dráhy nástroje umožňuje v SolidCAMu operace HSS a operace souvislé víceosé frézování. V následujících kapitolách jsou tyto operace popsány.

1.4.1 HSS

HSS (High Speed Surface) je operace, která jak vyplývá z názvu podporuje obrábění ve vysokých rychlostech, které podporuje podříznutí. Je zde možnost speciální spojení dráhy nástroje. K těmto spojení patří tečné a jemné náběhy a výběhy. Tato operace má také kontrolu kolize nástroje, držáku a trnu. V parametrech dráhy nástroje je možnost zaškrtnout pole pokročilé a tím se zobrazí záložka modifikovat, kde je možné zapnout 3D nástrojovou korekci. [9]

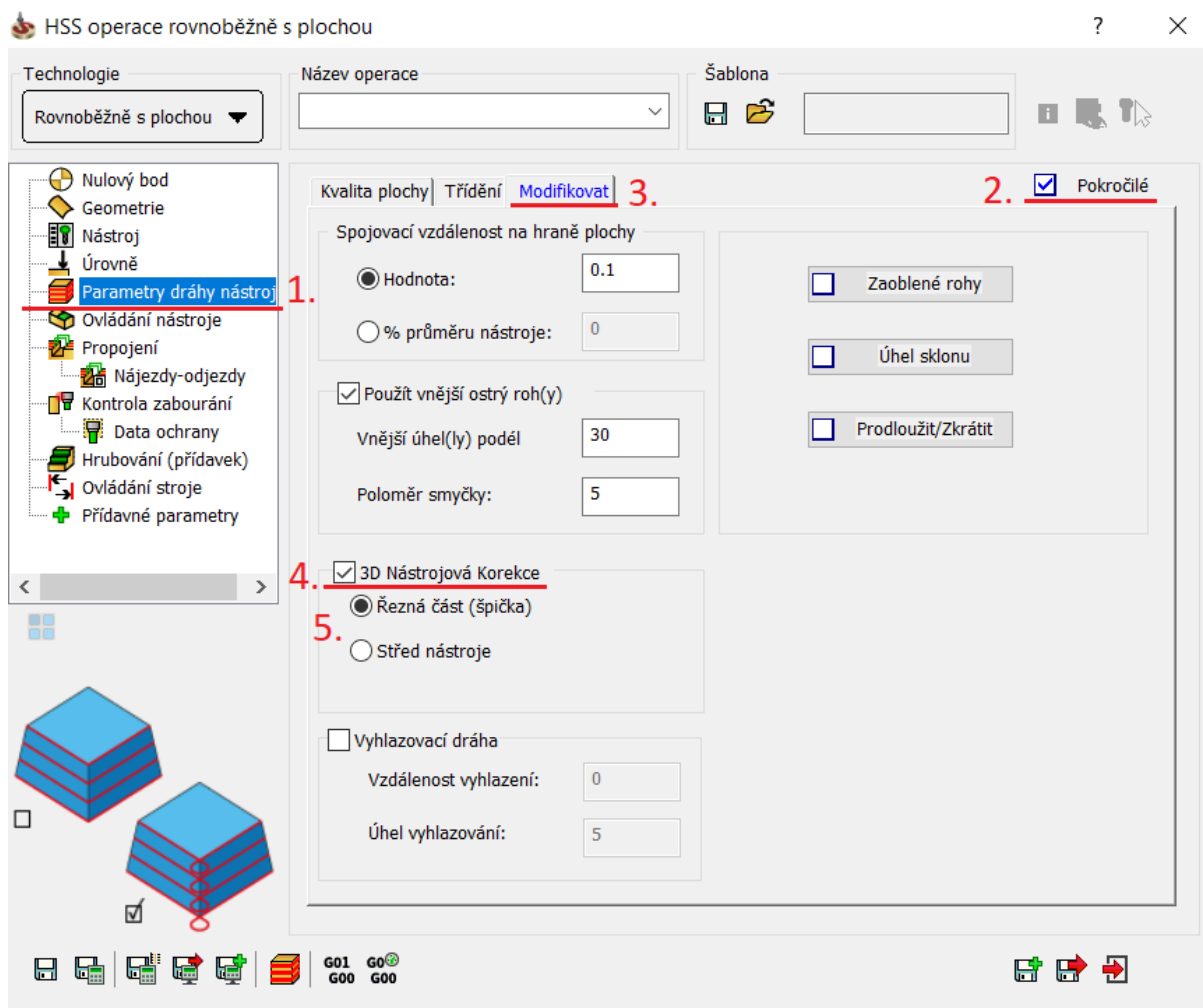
1.4.2 Souvislé víceosé frézování

Tato operace je jedna z nejpokročilejších a nejvýkonnějších v SolidCAMu, protože poskytuje velké spektrum souběžných řezných strategií v pěti osách a má nejvyspělejší kontrolu nad veškerými hledisky ohledně dráhy nástroje a dohled nad kolizí. Stejně jako u HSM v parametrech nástroje je po zaškrtnutí možnosti pokročilé, záložka modifikovat, která umožňuje zapnutí a vypnutí 3D nástrojové korekce. [14]

1.4.3 Postup zapnutí 3D korekce nástroje v SolidCAM 2020

Nejprve je potřeba přidat frézovací operaci, která podporuje možnost 3D korekce nástroje, tedy HSS nebo víceosé souvislé frézování. Postup zapnutí 3D nástrojové korekce je zobrazen na Obr. 9. Vysvětlení jednotlivých kroků:

1. Následně v zobrazeném dialogovém okně na levé straně vybrat záložku parametry dráhy nástroje.
2. V pravém horním rohu zaškrtnout políčko pokročilé, čímž se zobrazí záložka Modifikovat.
3. Zvolit záložku modifikovat.
4. Zaškrtnou políčko u 3D Nástrojová korekce
5. Vybrat možnost Řezná část (špička) nebo Střed nástroje



Obr. 9: Zapnutí 3D korekce v systému SolidCAM [SolidCAM 2020]

Aktuální stav je však takový, že i když je korekce zapnutá, vygenerovaný NC program neobsahuje parametry pro 3D nástrojovou korekci. Je tedy třeba prozkoumat řídicí systém a postprocessor, aby data pro trojrozměrnou korekci byla vypisována do NC programu a rozuměl tomu řídicí systém stroje.

1.5 Postprocessor

Tato kapitola je zaměřená na nejdůležitější aspekt této práce, kterou je postprocessor. Postprocessor je nejzásadnějším faktorem, protože ke zprovoznění 3D korekce je klíčové pochopit to, jakým způsobem postprocessor funguje a pochopit strukturu a syntaxi kódu, aby mohl být upraven tak, že informace, které jsou zásadní pro 3D korekci budou vypsaná do NC programu. Postprocessor převádí CL data, která jsou vygenerována CAM systémem do NC programu pro řídicí systém. Každý řídicí systém a stroj vyžaduje specificky upravený postprocessor.

Postprocesory se běžně skládají ze dvou částí, kterými jsou zvláště dva soubory, což je GPP soubor a VMID soubor.

VMID (z anglického názvu Virtual Machine ID) obsahuje data týkající se stroje, mezi které patří například to, jaký je rozsah pohybových os stroje nebo kinematika stroje, a to jakým způsobem se má postprocessor chovat, což jsou například uživatelské parametry, chlazení atp.

GPP je zkratka pro obecný postprocessor (z anglického názvu General Post-processor) a ve své podstatě se jedná o kód, neboli algoritmus, který má za účel převést CL data do NC programu pro určitý řídicí systém a stroj.

GPP využívá programovací jazyk GPPL (general post processor language). V GPPL je napsán soubor GPP, ve kterém je napsán kód postprocesoru pro konkrétní stroj a řídicí systém. Tento kód je ve své podstatě algoritmus, jehož funkcí je převod všech dat, která jsou poskytnuta CAM systémem. Má za úkol převést data z dokumentu VMID do požadovaného NC programu pro řídicí systém stroje. Součástí této práce je zaměření se na část postprocesoru GPP, protože to je ta část, kterou je možné opravit k tomu, aby bylo možné zprovoznit a otestovat 3D korekci nástroje. Pro úpravu je možno využít běžně dostupné textové editory, například Visual Studio Code, který je široce používán pro programování. [10]

1.5.1 Struktura GPP

Pro pochopení struktury GPP existují základní procesy, které jsou v GPP používány. Každý programovací jazyk má svojí specifickou syntaxi. Pro pochopení jsou důležité pochopit základní principy, jak je strukturován postprocessor, pro převod CL dat do NC programu. Procesy jsou zahájeny pomocí formátu @“název operace“ a končí funkcí „endp“. V průběhu tvorby NC programu dochází k postupnému volání jednotlivých procesů prostřednictvím algoritmu. Proces call @“název procesu“ můžeme procesy volat. To, jak jsou za sebou procesy uspořádán není podstatné. Následuje základní přehled procesů ke správnému pochopení základu postprocesoru. [10]

Proces @unit_post

Zde jsou definovány proměnné, které neobsahuje GPP a jsou využívány v celém programu. To jaký počet desetinných míst je vypisován do NC kódu je u každé proměnné možné definovat. Je zde také funkce trace, která nabývá tří hodnot a to 0, 1 a 5. Při nastavení funkce trace na hodnotu 0 nejsou vypisovány žádné informace navíc, jsou pouze generovány data NC programu pro stroj. Při nastavení na hodnotu 1, jsou do NC programu přidány informace o procedurách, které definují určitý blok. Při nastavení trace na 5 jsou do NC programu vypisovány všechny parametry ze systému CAM. [10]

Proces @rapid_move

Tento proces je určen, pro generování rychloposuvů, které se používají například při nájezdech či odjezdech nástroje z pracovní polohy. Běžně generuje data na začátek a konec operace, proto je vhodný při testování úpravy postprocesoru. [15]

Proces @start_of_file

Zde je možné určit jakým způsobem se bude chovat postprocessor, přidáním proměnné, která určí jednání postprocessoru po celkový čas tvorby NC programu. V této proceduře jsou zahrnuty informace nastavení uživatele, kde se nachází soubory a informace o obrobku. Mezi další parametry, které se v této proceduře definují jsou cykly a chlazení. [10]

Proces @def_tool

Tato procedura slouží k definování nástrojů. Patří sem údaje o posuvu, otáčkách a dalších zásadních datech, které se týkají nástroje, mezi které patří průměr, délka a číslo nástroje. [10]

Proces @change_tool

Tento proces, který má za úkol určit nástroj a provést výměnu nástroje. [10]

Proces @start_of_job

Uživatel určuje to, jaké hodnoty bude mít CAM operace. Tyto přesná data jsou obsažena v tomto procesu. V momentě, když začíná operace, je tento proces spuštěn. Je určena pro procesy, které mají za úkol generovat přídatné funkce nebo komentáře. Používá se také pro operace, které je užitečná k tvorbě dráhy pohybu, což jsou procesy jako @line, @move, @line_5x, @move_5x, @arc. Tyto procesy, které slouží pro generování pohybů nástroje mají za úkol, aby výslednému NC programu rozuměl stroj. [10]

Co se týče víceosého obrábění, pokud jde o víceosý stroj, který má rotační osy, zde je nutné provést transformaci souřadnic. Tato transformace je uskutečněná a počítána v SolidCAM souřadným systémem, který je nastaven v programu VMID, který je součástí SolidCAM. [10]

1.5.2 Typy proměnných

Pro porozumění kódu GPP jsou podstatné typy proměnných, které jsou používány. V jazyce GPP se pracuje s různými typy proměnných. Existují pouze čtyři typy proměnných v GPPL, které jsou vypsány v Tab. 1.

Tab. 1: Typy proměnných v GPP [16]

Typ proměnné	Popis proměnné
celé číslo (integer)	Tato proměnná může být jak kladná, tak záporná. Její rozsah je od -999999999 do +999999999
číslo (numeric)	Tato proměnná má daleko větší rozsah než proměnná integer. Zde je rozsah od -1×10^{300} do $+1 \times 10^{300}$
logická (logical)	Logická proměnná může nabývat pouze dvou hodnot. Tyto hodnoty jsou buďto 1 při pravdě, nebo 0 při nepravdě.
řetězec (string)	Tato proměnná je určena k vypisování jakéhokoliv množství ASCII znaků.

Zde byl jednoduchý přehled typů proměnných. Proměnné, které již jsou v jazyce GPPL nastaveny je velké množství. V Tab. 2 je přehled několika proměnných, kde některé z nich jsou základní proměnné a některé přímo souvisejí s trojrozměrnou korekcí.

Tab. 2: Vysvětlení funkcí některých důležitých proměnných [15]

Proměnná	typ	funkce
xpos, ypos, zpos	číslo	Tyto proměnné jsou pro aktuální pozici na konci pohybu.
xlpos, ylpos, zlpos	číslo	Do této proměnné se ukládá souřadnicový systém posunutý pracovní roviny
xpos_f ypos_f zpos_f	řetězec	Nastavuje formát kterým se zobrazují hodnoty X, Y a Z.
xpos_normal, ypos_normal zpos_normal	číslo	Normálový vektor povrchu
x_surface_point y_surface_point z_surface_point	číslo	Proměnná, která je určena pro bod kontaktu nástroje s obráběným povrchem.
x_surface_normal y_surface_normal z_surface_normal	číslo	Normálový vektor povrchu, který je určený pro kontaktní bod nástroje s obráběným povrchem.

1.5.3 Podmínka v GPP

V jazyce GPP je struktura podmínky prakticky stejná, jako v jiných programovacích jazycích. Na Obr. 10 je zobrazena struktura podmínky v GPPL:

```

if (podmínka 1)                ;nastavení podmínky 1
    ;Proved' operaci 1
elseif (podmínka 2) then      ;nastavení podmínky 2
    ;Proved' operaci 2
else                            ;V případě že není splněna podmínka 1 ani podmínka 2
    ;proved' operaci 3
Endif                          ;ukončení podmínky

```

Obr. 10: Struktura podmínky v GPPL [15]

Toto je jednoduchý příklad toho, jakým je zadávána podmínka v GPPL. Podmínky jsou velmi často využívané a pro vypisování hodnot normálového vektoru, bude podmínka využita. Středníky jsou používány pro komentář v kódu, který nemá žádný vliv na algoritmus postprocesoru. Slouží k tomu, aby se programátor mohl v kódu lépe orientovat. [15]

1.5.4 Způsob zobrazení proměnných

V postprocesoru lze nastavit jakým způsobem jsou vypisovány proměnné do NC programu, což je důležitá část, protože řídicí systém stroje potřebuje data pro 3D korekci mít zadaná přesně daným způsobem. Formát je takový, že se v procesu @unit_post nastavují hodnoty tímto způsobem:

```
xpos_f = '+6.1'
```

Obr. 11: Nastavení proměnné v GPP

Na Obr. 11 je proměnná xpos_f a za rovnítkem je nastavení zobrazení proměnné. Proměnná může mít například hodnotu 21.715. '+6.1' nastavuje proměnnou následovně:

- Číslo před desetinnou čárkou v uvozovkách nastavuje proměnné maximální počet míst, který jsou vypisována před desetinnou čárkou, v tomto případě 6.
- Číslo za desetinnou čárkou v uvozovkách nastavuje proměnné maximální počet míst, která jsou vypisována za desetinnou čárkou, v tomto případě 1.
- Znaménko + nastavuje to, aby byly vypisovány kladná nebo záporná znaménka hodnota proměnné.

To znamená, že pokud by proměnná měla hodnotu 21,715 a byla by nastavená na '6.1' do NC programu by bylo vypsáno +21,7. [15]

1.6 Řídicí systém stroje

Řídicí systém posílá data z NC programu stroji, který na jejich základě vyrábí součást. Ne všechny řídicí systémy podporují korekci nástroje ve třech osách. K tomu, aby trojrozměrná korekce fungovala, musí mít řídicí systém nastavené jakým způsobem bude korekce zadávána, aby byl schopen korekci porozumět. Z toho důvodu je zásadní porozumět tomu, jak daný řídicí systém potřebuje zadat data pro trojrozměrnou korekce nástroje a jaké jsou podmínky zadávání.

Pro tuto práci byl vybrán řídicí systém Heidenhain TNC 530. Tento řídicí systém je jeden z nejpoužívanějších řídicích systémů a už do předchozích verzí byla přidána funkce korekce nástroje v trojrozměrném prostoru.

1.6.1 Trojrozměrná korekce nástroje v Heidenhain TNC 530

Každý řídicí systém zadává 3D korekci jiným způsobem. Heidenhain TNC 530 má konkrétní syntaxi, neboli pravidla bloku kódu, která musí být dodržena, aby mohla být 3D korekce skutečně provedena. Bez nedodržení této syntaxe a pravidel je patrné, že se trojrozměrná korekce nezprovozní.

Bylo tedy potřeba zjistit, jak přesně má vypadat NC program pro Heidenhain, aby 3D korekce byla uskutečněna. Heidenhain TNC 530 dokáže pro přímkové části kódu uskutečnit 3D korekci. U těchto částí je nutné, aby zahrnovaly souřadnice normálového vektoru povrchu NX, NY a NZ. Pro polohu nástroje při trojrozměrné korekci rádiusu ve víceosém obrábění je nutné, aby součástí kódu byl normálový vektor pro natočení, který se zadává pomocí souřadnic TX, TY a TZ. Zde přichází na řadu SolidCAM, protože souřadnice normálového vektoru je potřeba nechat vygenerovat CAM systémem, který dokáže vygenerovat normálový vektor povrchu. V momentě, kdy nastane případ, že by byla požadována trojrozměrná korekce ve víceosém obrábění, použije se CAM systém pro vygenerování dat pro orientaci nástroje. [11]

Důležité jsou specifické možnosti 3D korekce v řídicím systému Heidenhain pro různé typy nástrojů a metody frézování. Zde je přehled frézovacích operací, kde lze tuto funkci využít:

- Využití 3D korekce je možné při čelním frézování, kde dochází k obrábění v trojrozměrném prostoru ve třech osách.
- Při obvodovém frézování pro korekci rádiusu, který je orientován kolmo k trajektorii nástroje a také kolmo ke směru pohybu frézy. Tato korekce zahrnuje se využívá při víceosých operacích, kvůli specifické orientaci nástroje.
- V situacích, kdy dochází k rozporu mezi rozměry vypočítanými systémem CAM a skutečnými rozměry nástroje. [11]

V následujících kapitolách je popsáno detailně, jaké jsou možnosti a omezení pro 3D korekci v Heidenhainu. Dále jsou zde zmíněna pravidla pro zadávání a jaké jsou rozdíly korekce bez a s orientací nástroje.

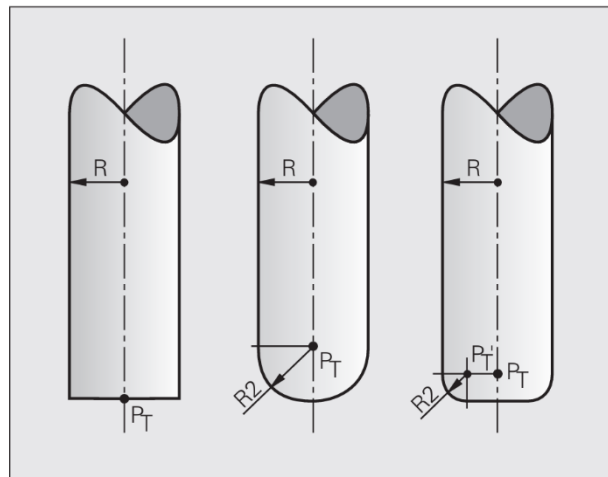
1.6.1.1 Definice normálové vektoru

K tomu, aby mohla 3D korekce pomocí řídicího systému fungovat, je třeba si dát pozor na to, aby byly informace, které řídicí systém potřebuje správně vypsány do NC programu. Definice normálového vektoru zní, že jeho hodnota se rovná jedné a jeho vlastností je libovolný směr. Řádek NC programu s normálovým vektorem má začátku řádky LN místo L, čímž Heidenhain pozná, že se jedná o lineární blok s trojrozměrnou korekcí frézy. Podle typu obrábění, tedy jestli se jedná o tříosé nebo víceosé, je zapotřebí definovat jeden či více normálových vektorů. Pro tříosé obrábění je dostačující jediná normála k povrchu, zatímco pro víceosé obrábění je nezbytné specifikovat dodatečný vektor, který určuje natočení frézy.

Orientace normály k povrchu je určena pomocí souřadnic NX, NY a NZ, jež se odvíjí od typu frézovacího nástroje. Existují tři hlavní kategorie fréz, které jsou vhodné pro 3D korekci:

- stopková fréza
- rádiusová fréza
- fréza která má zaoblené rohy

Tyto nástroje a jejich parametry jsou zobrazeny na Obr. 12. U stopkové a rádiusové frézy se situace neliší, což znamená, že normální vektor je orientovaný kolmo od povrchu vyráběného dílu směrem k bodu P_T frézy. U fréz se zaoblenými hranami směřuje tento vektor kolmo vůči povrchu obráběné součásti směrem k referenčnímu bodu frézy P_T , případně k P_T . Co se týče určení orientace frézy, zde slouží vektory TX, TY a TZ. [11]

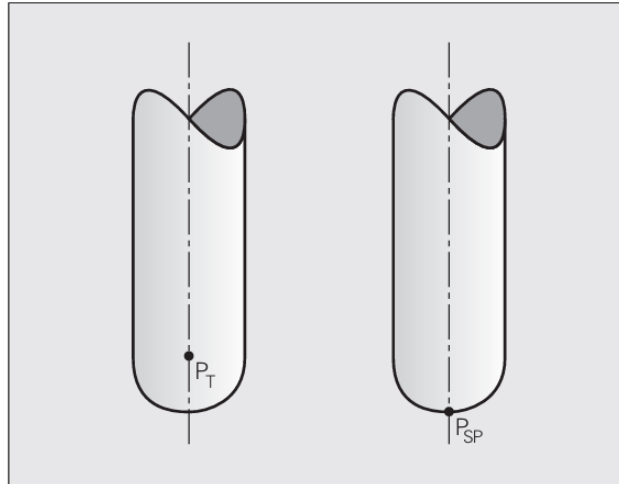


Obr. 12: Přehled možných fréz pro 3D korekci v Heidenhain iTNC 530 [11]

1.6.1.2 Pravidla pro zadávání 3D korekce

Jak již bylo zmíněno, každý řídicí systém má svoje pravidla pro zadávání korekce. Pokud by se tyto pravidla nedodržela, korekce by nemusela fungovat. Zde jsou vypsána tři zásadní pravidla, která se v Heidenhainu při 3D korekci musí dodržet.

- Do bloku s trojrozměrnou korekcí je důležité zadat všechny souřadnice a normálový vektor. Toto platí i v případě, jestliže nebyla upravena hodnota souřadnice od předchozího bloku.
- Jediná správná možnost je zadávat normálový vektor povrchu na 7 desetinných míst.
- Je zde důležitý parametr stroje 7680, který má vliv na to, jakým způsobem bude provedena korekce nástroje, jestli na jeho špičku P_{SP} nebo střed P_T . Tyto body jsou znázorněny na Obr. 13. [11]



Obr. 13: Schématické znázornění špičky a středu kulové frézy [11]

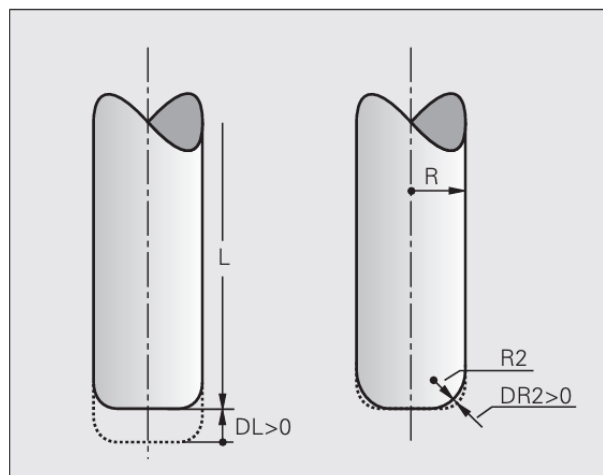
Jak nastavit korekci na špičku nástroje nebo na střed v prostředí SolidCAMu bylo popsáno v kapitole 1.4.3.

1.6.1.3 Dovolené tvary nástroje

Kromě dovolených typů fréz je zde také požadavek na tvar nástroje. Pro určení správného tvaru jsou zde zásadní některé parametry, kterými jsou:

- Rádus nástroje R
- Rádus nástroje $R2$

Toto jsou parametry pro tvar nástroje, které jsou zobrazené na Obr. 14. [11]



Obr. 14: Schématické znázornění hodnot $R2$ a $DR2$ [11]

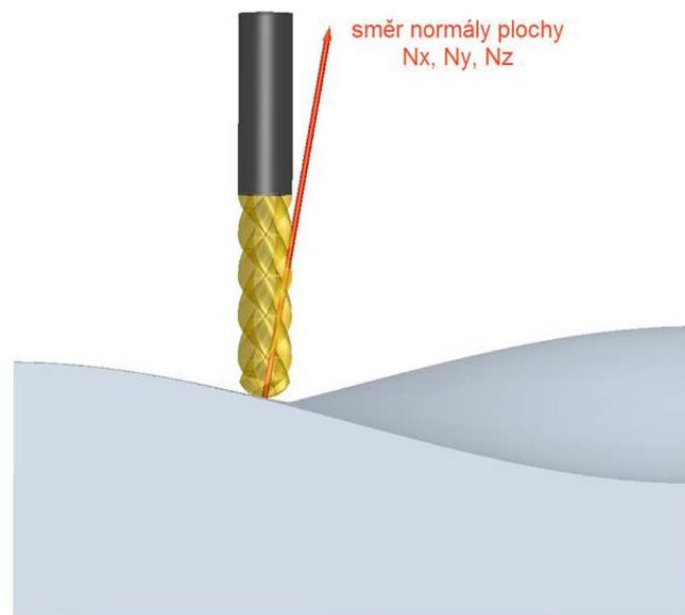
To v jakém poměru R a $R2$ vůči sobě určuje tvar nástroje. Pro různé nástroje platí:

- Stopková fréza: $R2 = 0$
- Rádusová fréza: $R2 = R$
- Fréza s rohovým rádiusem: $0 < R2 < R$

1.6.1.4 3D korekce bez definice orientace nástroje

3D korekce, při které není použita orientace nástroje v praxi znamená, že při této metodě nenastane žádné natočení nástroje vůči obrobku, tedy slon zůstává neměnný. Tento přístup je běžný u strojů, které pracují ve třech osách X, Y a Z.

Tento druh korekce využívá kromě souřadnic X, Y a Z hodnoty, které určují směr normálového vektoru plochy. Tento vektor má souřadnice N_x , N_y a N_z . Zásadou normálového vektoru plochy, může být řídicím systémem poloha nástroje upravena ve všech třech osách. Korekce je prováděna ve směru normály povrchu. Pro získání vektoru normály plochy je nutné použít CAM systém, který podporuje výpočet těchto souřadnic, aby bylo možné použít 3D korekci. Směr vektoru je naprogramován tak, aby směřoval od bodu kontaktu nástroje a programované dráhy ke středu nástroje, jak je znázorněno na Obr. 15. Tímto způsobem je řídicí systém schopný určit směr posunutí vzhledem k obráběné ploše a vypočítat posunutí nástroje v každé ose. [6]

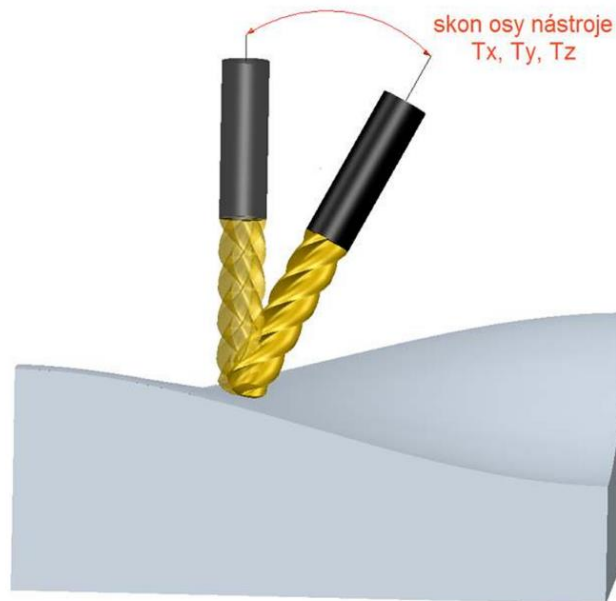


Obr. 15: Směr normálového vektoru plochy [6]

1.6.1.5 3D korekce s definicí orientace nástroje

Při 3D korekci nástroje s určením orientace nástroje znamená, že je aktivováno natočení nástroje vzhledem k obrobku, což vede ke změně sklonu nástroje. Tato funkce je využívána u víceosého frézování, tedy v případě že k osám X, Y, Z jsou přidány rotační osy A, B nebo C.

Zde je odlišnost oproti předchozí variantě v tom, že jsou navíc přidány souřadnice normálového vektoru natočení T_x , T_y a T_z , které umožňují přesné naklopení nástroje během obrábění. Normálový vektor je rovněž jednotkový a jeho počátek má polohu střed nástroje. Stejně jako u přechodí varianty je normálový vektor vypočítán pomocí CAM systému. Sklonění nástroje je znázorněno na Obr. 16. [6]



Obr. 16: Sklon osy nástroje [6]

1.6.1.6 Trojrozměrná korekce při čelním frézování bez a s orientací nástroje

V této kapitole je popsáno, jak konkrétně vypadá blok NC kódu pro Heidenhain s 3D korekcí. Jsou zde značné rozdíly při obrábění bez orientace nástroje a s orientací nástroje. V programu může být nastavena různá poloha pro naklápěcí osu. Naklápěcí osa slouží k naklonění nástroje různými směry. V momentě, když nastane změna v naklonění osy, tak TNC musí přepočítat, jakým způsobem se má nástroj pohybovat v lineárních osách X, Y a Z, aby bylo dosaženo požadované orientace. [11]

1.6.1.6.1 Funkce M128 a TCPM

Tato funkce je určena pro řízení středu nástroje a jeho orientace. Funkce M128 a TCPM jsou součástí volitelného softwaru 2 a je určena výhradně pro víceosé obrábění. V případě změny polohy osy naklápění je změněna pozice frézy vůči vyráběné součásti v momentě, když probíhá naklápění. Část NC kódu se začíná LN, která neobsahuje souřadnice T samostatně nastaví nástroj takovým způsobem, aby byl kolmý ke kontuře vyráběné součásti. K tomuto musí být aktivní funkce M128. TNC samostatně nastaví osy stroje pro rotaci takovým způsobem, aby bylo dosaženo požadovaného natočení v případě, že součástí kódu, který obsahuje LN je zapnutá souřadnice T a zároveň funkce M128 nebo FUNCTION TCPM. V případě, že funkce M128 nebo TCPM je vypnutá, nastává situace, kdy TNC nebere vůbec ohled na vektor směru T, tedy je přehlížen, a to i v případě že je v části LN určen. Hrozí zde riziko srážky hlavy s obrobkem nebo upínadly u strojů, u kterých osy natočení umožňují limitovanou velikost pojezdu. Funkce M128 se používá na začátku bloku programu. Pro její zrušení M128 je používána funkce M129. V následujících kapitolách jsou uvedeny příklady vzorové části NC programu při čelním frézování. [11]

1.6.1.6.2 LN blok s vektorem normály povrchu, když není zapnutá orientace nástroje

LN X+45,753 Y+95,524 Z+21,822 NX+0,8761259 NY+0,0158974 NZ-0,4268479 F1000 M128

Tento příklad ilustruje to, jak má správně vypadat NC program. V normálním případě by na začátku řádky bylo pouze "L" pro lineární přímkou. Zde řádek začíná "LN" což je přímkou s 3D korekcí, podle které Heidenhain pozná, že se jedná o část programu ve kterém jsou data normálového vektoru a že se jedná o 3D korekci. Po "LN" následují souřadnice lineárních os X, Y a Z. Poté jsou zde normálové vektory povrchu pro NX, NY a NZ, které byly vygenerovány pomocí programu CAM. Následuje F je nastavení rychlosti posuvu a přídavná funkce M128. [11]

1.6.1.6.3 LN blok s normálovým vektorem plochy, když je zapnutá orientace nástroje

LN X+51,854 Y+58,954 Z+12,459 NX+0,1496587 NY+0,0025694 NZ-0,75236856 TX+0,0847569 TY-0,6958742 TZ+0,3658749 F1000 M128

V tomto případě se zde opakuje LN přímkou s trojrozměrnou korekcí nástroje, souřadnice pro lineární osy X, Y a Z, dále NX, NY a NZ jsou normálové vektory povrchu, které byly vygenerovány pomocí CAM, F posuv a M128 pro automatické naklopení nástroje. Po normálách povrchu jsou zde přidány normálové vektory TX, TY a TZ k orientaci nástroje. [11]

1.6.1.7 Trojrozměrná korekce při obvodovém frézování s orientací nástroje

Důležité je také zmínit, jakým způsobem má být zadávána 3D korekce při obvodovém frézování oproti čelnímu frézování.

Zde je poloha nástroje pomocí TNC upravována tak, že nástroj má být vůči směru pohybu kolmý. Směr korekce je v tomto případě určen korekcí RL nebo RR. Orientace nástroje je zapnutá funkcí M128 nebo TCPM. Takto funkce zajistí to, že jsou osy stroje pomocí TNC natočeny tak, aby bylo dosaženo požadované orientace s korekcí nástroje.

Zde jsou dvě podmínky, aby 3D korekce pro obvodové frézování s orientací nástroje mohla být zapnuta.

- Pouze pro stroje, u kterých v nastavení naklápěcích os je možné definovat prostorové úhly.
- Je zde nebezpečí kolize, jako u čelního frézování.

Na následujících řádcích jsou uvedeny příklady, jak má vypadat blok NC programu při obvodovém frézování:

LN blok pro obvodové frézování, když je zapnutá orientace nástroje

LN X+46,496 Y+11,758 Z+36,258 TX+0,0031487 TY-0,7596321 TZ+0,1478931 RR F1000 M128

Při 3D korekci obvodovém frézování, jde o to, že je naklopen nástroj a následně je provedena pravá korekce nebo levá korekce. Důvod, proč se jedná o trojrozměrnou korekci je, že korekce není provedena na jedné z rovin G17, G18 a G19, ale závisí to na naklopení osy ve 3D prostoru, čímž vzniká ohromné množství variant. K tomu, aby tento způsob korekce mohl být uskutečněn jsou potřeba souřadnice normálového vektoru naklopení T. Vzorový řádek kódu má určité rozdíly pro obvodové frézování oproti čelnímu frézování. Začátek řádky začíná stejně LN přímkou pro 3D korekci a lineárními souřadnicemi X, Y, Z. Zásadním rozdílem je absence normálového vektoru povrchu a následují pouze normály vektoru pro natočení vygenerované CAM systémem. Navíc je zde RR nebo RL pro korekci zprava nebo zleva.

1.6.1.8 Volitelný software v rámci TNC

Pro vysvětlení speciální funkce pro 3D korekci nástroje je potřeba nejprve zmínit, že Heidenhain disponuje tzv. volitelným softwarem. Jedná se o jednotlivé opční programy, které je možné aktivovat výrobcem stroje. Každý volitelný software umožňuje funkce navíc. Pro funkci 3D-ToolComp existuje volitelný software 2 a volitelný software 92 (také pod názvem opce).

1.6.1.9 3D-ToolComp

Pokud je vytvořen blok programu pro 3D korekci, korekce je prováděna vždy pro jeden konkrétní bod dotyku nástroje s obrobkem. 3D-ToolComp je volitelná funkce v Heidenhainu, která je přímo určená pro 3D korekci s normálovým vektorem povrchu, který je závislý na úhlu záběru. To znamená, že 3D-ToolComp dokáže zohledňovat odchylky nepřesnosti nástroje přímo na konkrétním úhlu záběru, což vede k podstatnému zlepšení přesnosti obrobku, zejména co se týče volných tvarových ploch. Je možné jej použít ve volitelném softwaru 92 a volitelném softwaru 2. Volitelném softwaru 92 lze použít jenom v případě aktivního volitelného softwaru 2.

Funkce 3D-ToolComp má jistá pravidla a omezení, které je zásadní dodržet, aby mohla být použita. Jedním z pravidel je, že může být použita pouze v blocích, které obsahují LN. V této kapitole popsány podmínky pro to, aby NC program fungoval s funkcí 3D-ToolComp, která je součástí Heidenhainu. Hlavním parametrem pro, který je potřeba dodržet u této funkce je zajistit to, aby NC kód zahrnoval složku normály povrchu N.

Je důležité dodržet následující pravidla:

- Výsledná korekce může být počítána buďto na střed frézy nebo na jeho špičku. V případě, že je zvolen střed frézy, potom je důležité zadat velikost R2 rádiusové frézy do TOOL.T., což je tabulka, která slouží k zadávání nástrojů.
- V případě, že je NC program spočítaný na špičku frézy, poté je v tabulce TOOL.T důležité nastavit velikost R2 pro rádiusovou frézu a také velikost R2 nastavit ve sloupci, který je nazýván DL tak, aby byla záporná velikost Delta.

Příklad bloku programu, který je ve třech osách

FUNCTION TCPM OFF (znamená že funkce TCPM pro orientaci nástroje je vypnutá)

LN X+21,456 Y+11,741 Z+21,254 NX+0,1268749 NY+0,0074687 NZ-0,1576914 F1000

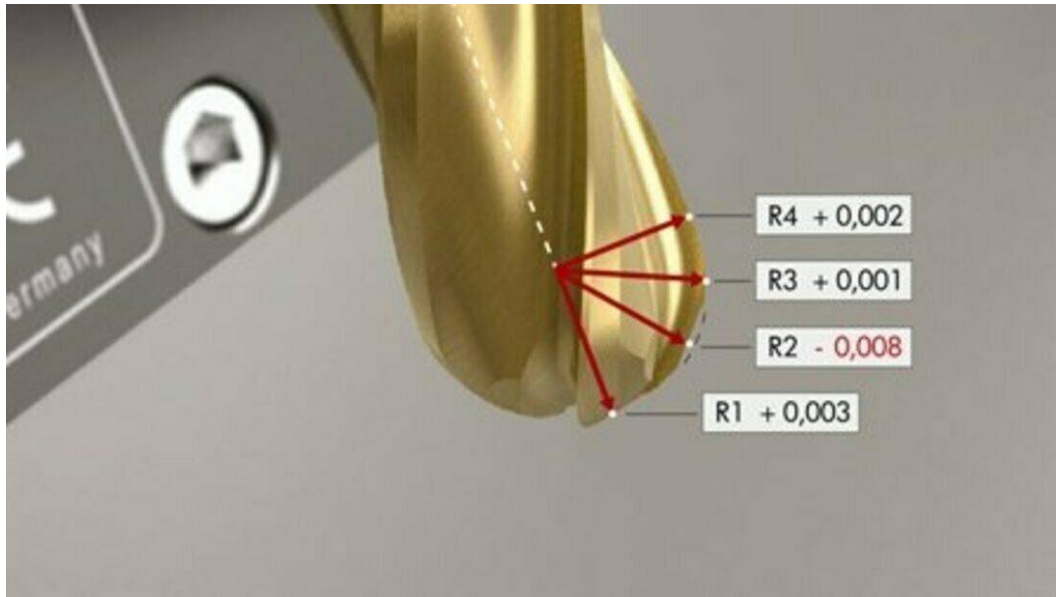
Toto je v zásadě stejný příklad řádku programu, jako při čelním frézování bez naklopení nástroje. Části kódu s LN, kterých je součástí normálový vektor povrchu mají takovou funkci, že určí to, jakou polohu má mít nástroj a bod kontaktu nástroje s obrobkem.

Chyby tvaru frézy se určují pomocí cyklu sondy 444. Je zásadní zmínit, že funkce 3D-ToolComp je příplatková funkce, která není součástí základní verze Heidenhain TNC 530. [7] [11]

1.6.1.10 Měření nástroje

Každý nástroj má odchylku od dokonalého tvaru. Avšak NC program bývá programován na ideální tvar nástroje a tato odchylka způsobuje nepřesnost obrábění. Podle toho, jaký je typ frézy jsou parametry, u kterých je odchylka zjišťována. U čelní stopkové frézy je jedná o délku nástroje a její rádius. Jiná situace je u tvarových nástrojů, například u kulové frézy, pro kterou je 3D korekce využívána. Zde už počet míst, které je potřeba změřit, aby byla zjištěna nepřesnost oproti požadovanému ideálnímu tvaru je podstatně větší. Příklad, jaké body jsou

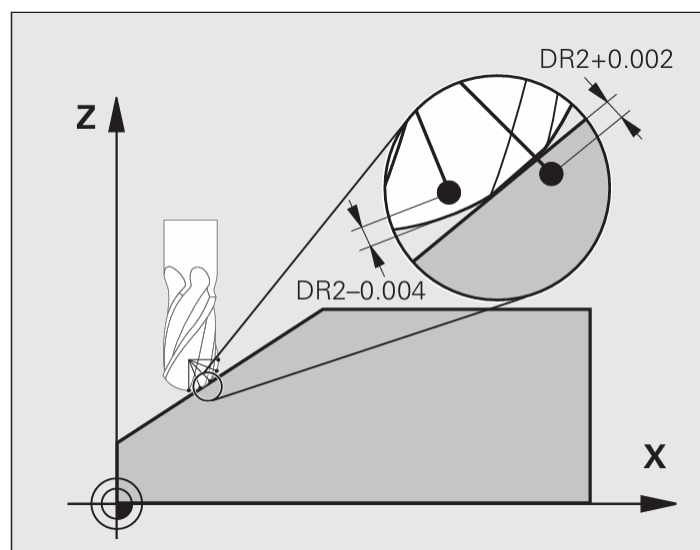
měřeny na kulové fréze ilustruje Obr. 17. Heidenhain k měření kulových frézy využívá laserový systém.



Obr. 17: Odchylka kulové frézy od ideálního tvaru [13]

Laserový systém umožňuje velmi přesně měřit nástroj a díky tomu kompenzovat nepřesnosti nástroje. Heidenhain v normálním stavu předpokládá perfektní kruhovitost nástroje. Po měření nástroje se však zjišťuje, že zde není perfektní a jsou zde odchylky. Zde je klíčové vysvětlit pojem delta hodnoty. Delta hodnoty jsou udávající odchylku od ideálního tvaru a rozměru nástroje. V Heidenhainu je možné přímo do tabulky pro korekce zadat tyto delta hodnoty, které jsou pro daný úhel kulové frézy. [13]

Kulová fréza, pro kterou je využívána 3D korekce, respektive její rádius, disponuje také odlišností od ideálního tvaru, která je udávána výrobcem frézy. Tato odlišnost od dokonalého tvaru se obvykle pohybuje od 0,005 do 0,01 mm. TNC disponuje laserovým zařízením a cykly, které slouží k rozpoznání odchylky tvaru kulové frézy. Změřená data zahrnují data úhlů, které zahrnují odlišnosti od dokonalého rádiusu. Tato data se zadávají do tabulky a funkce 3D-ToolComp dokáže provést korekci této nepřesnosti. [11]



Obr. 18: Znázornění nepřesnosti nástroje [11]

Parametr R2 je určen pro velikost rádiusu frézy speciálně pouze pro 3D korekci. Pro fungování korekce je zde nutná podmínka, že NC program musí obsahovat normálový vektor plochy. Laserový proměřovací cyklus 598 vytváří tabulku korekčních hodnot automaticky. Dále hodnota DR2 je odchylka od přesného rozměru frézy R2. Odchyly DR2 jsou zobrazeny na Obr. 18. Tabulka pro korekční hodnoty při 3D korekci nástroje má zkratku DR2TABLE. [11]

Je potřeba generovat NC program s normálou plochy v každém bodě do LN bloků, pro zjištění kontaktního bodu nástroje s obrobkem. LN bloky určují přesnou polohu nástroje, místo dotyku s obrobkem a to, jakým způsobem má být orientovaný nástroj. [7]

1.6.1.11 Tabulka pro hodnoty odchylek nástroje

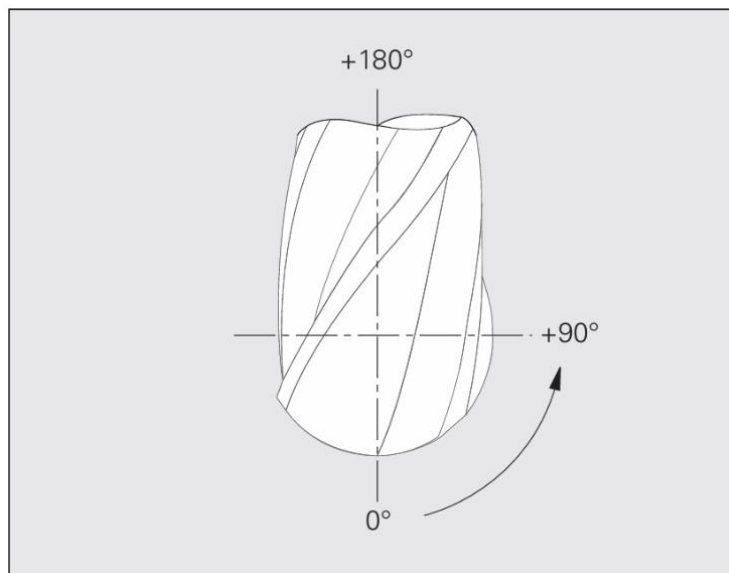
Údaje, které TNC potřebuje pro to, aby mohl provést trojrozměrnou korekci, jsou zadávány do specifické tabulky. Pokud je využit laserový systém pro měření, tak zásluhou cyklu 598 je tabulka vytvořena automaticky. Je však možné tuto tabulku vytvořit ručně a údaje do ní také samostatně zadat, postup je následující:

1. V prostředí Heidenhainu zvolit možnost PGM MGT
2. Zde je možné si jakkoliv tabulku nazvat a možností ENT je dokončen název tabulky
3. Posledním krokem je zvolení možnosti 3DTOOLCOM.TAB pomocí tlačítka ENT a zde je možné ručně vyplnit hodnoty.

Tato tabulka má tři sloupceky pro zadávání údajů:

- V prvním sloupci je hodnota úhlu, který je v případě kulové frézy možný měřit o 0° do 90°.
- V druhém sloupci je hodnota s názvem NOM-R2, což je údaj o ideální hodnotě.
- V posledním sloupci je NOM-DR2, což je údaj o kolik se rádius odchyloje od ideálního tvaru. Hodnota může být jak kladná, tak záporná [11]

Na Obr. 19 je zobrazena kulová fréza a oblast měření odchylek od ideálního tvaru a rozměru.



Obr. 19: Znázornění kulové frézy od 0° do 90° [11]

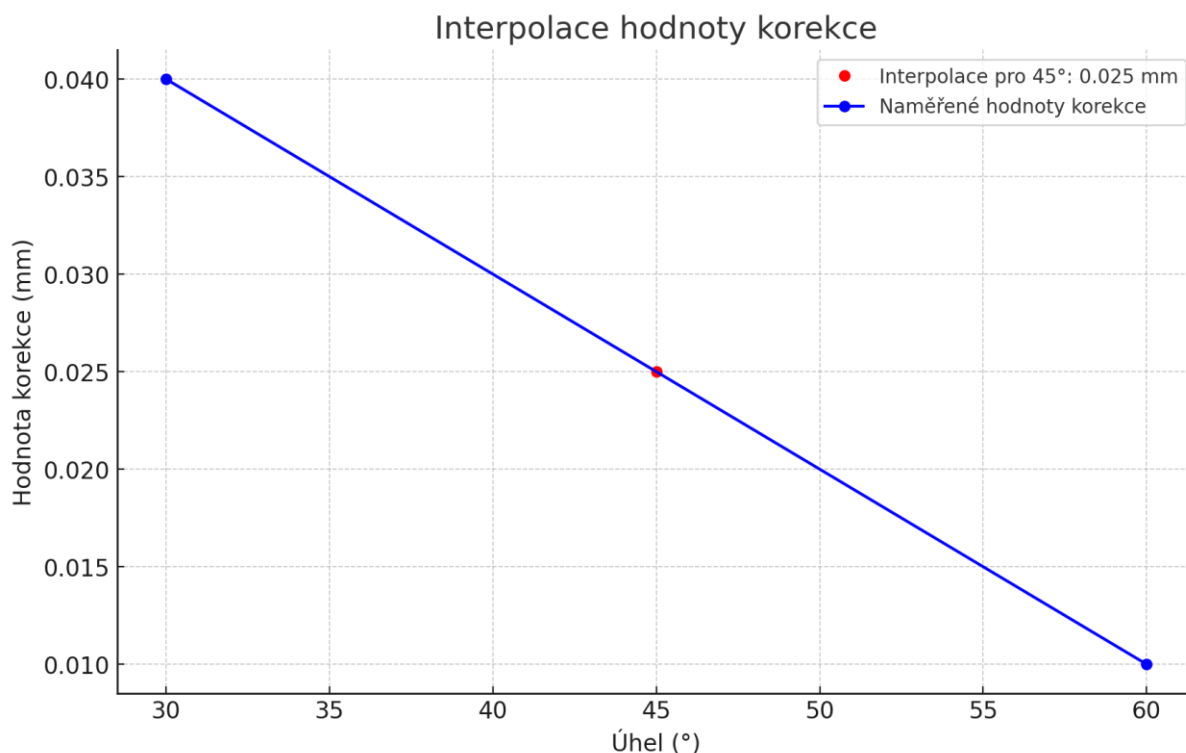
Korekční hodnoty jsou zadávány do sloupce DR2TABLE. Tyto hodnoty mají přednost před zadaným údajem DR2. Dotyk frézy s obrobkem je vždy definován pod určitým úhlem. Když nastane situace, že jsou naměřené dva body pro korekci a dotyk frézy s obrobkem náleží mezi nimi, poté TNC provede lineární interpolaci pro body, které jsou nejbližší bodu dotyku. [11]

V Tab. 3 je příklad naměřených hodnot při určitém úhlu. Úhly měření je při 30° a 60°. Bod dotyku je 45°.

Tab. 3: Příklad naměřených hodnoty pro určité úhly a interpolace pro 45°

Úhel	Hodnota korekce
30°	+0,04 mm (naměřeno)
60°	+0,01 mm (naměřeno)
45° (bod dotyku)	+0,025 mm (interpolováno)

Pro příklad byla provedena lineární interpolace pro tyto hodnoty korekce. Graf na Obr. 20 zobrazuje naměřené hodnoty korekce pro úhly 30° a 60°. Červeným bodem je označena interpolace pro úhel 45° a výsledná hodnota korekce je +0,025 mm.



Obr. 20: Interpolace hodnot pro určité úhly

2 Realizace úpravy postprocesoru pro vybraný stroj

Před úpravou postprocesoru bylo potřeba nejprve navrhnout součást pro výrobu k tomu, aby mohla být rovnou úprava postprocesoru testována na vymodelované součásti, která bude vyráběna.

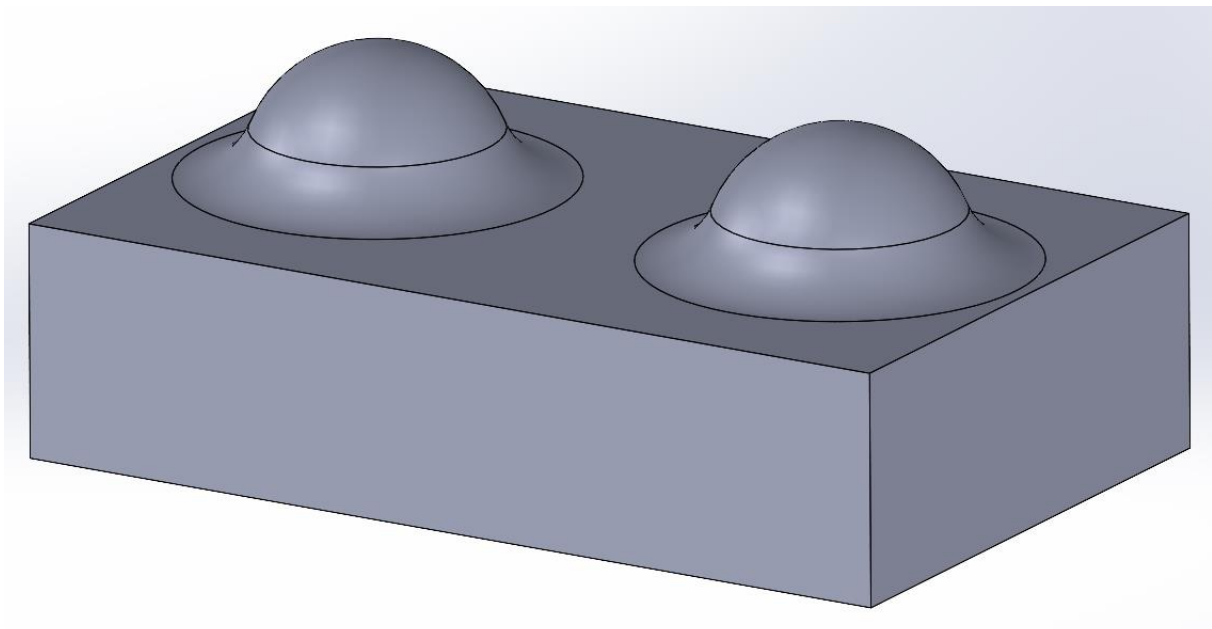
K úpravě postprocesoru bylo potřeba udělat několik kroků, kterými jsou:

- Pochopení kódu postprocesoru.
- Zjištění, jaká data souvisejí v SolidCAMu pro 3D korekci
- Jaké příkazy a funkce pro 3D korekci jsou v postprocesoru pro SolidCAM důležité.
- Úprava postprocesoru a otestování funkčnosti generováním NC programu v SolidCAM

Cesta k postprocesoru GPP je následující: Disk C – Users – veřejné – veřejné dokumenty – SolidCAM – SolidCAM2020 - Gpptool.

2.1 Součást pro výrobu

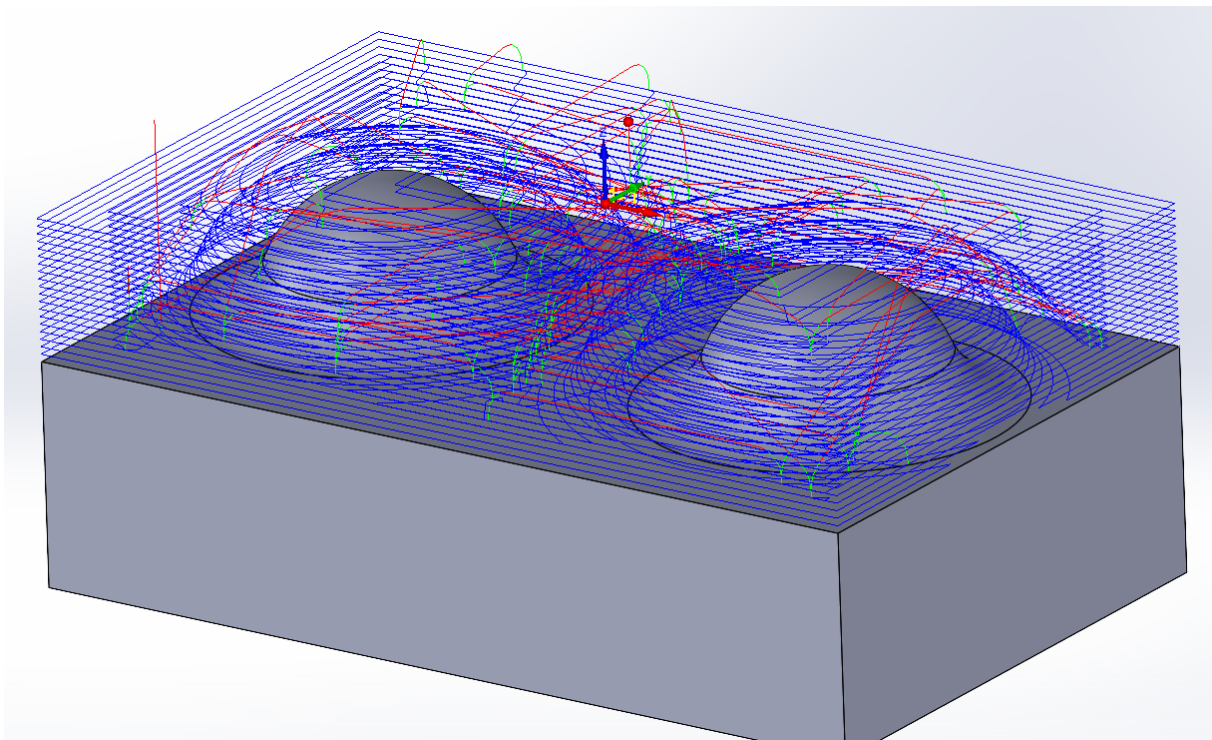
Pro průběžné ověřování funkčnosti postprocesoru a samotné otestování, byla navržena součást, která je zobrazena na Obr. 21: Vymodelovaná součást pro obrábění v SolidCAMu. Polotovár má tvar kvádrů o rozměrech 120 x 75 x 50 mm. Základna pro polokoule má výšku 25 mm. Na součásti byly vymodelovány dvě polokoule o poloměru 17 mm. Střed polokoulí je ve výšce 25 mm stejně základna. Dále bylo přidáno zaoblení o poloměru 10 mm. Vrchol polokoulí je ve výšce -3 mm od horní plochy polotovaru. Tvar polokoule byl zvolen, protože 3D korekce za použití kulové frézy má největší efekt pro tvarové plochy. Pravá polokoule je určena pro obrobení s 3D korekcí a levá pro obrobení bez aktivní 3D korekce. Materiál součásti byl použit hliník.



Obr. 21: Vymodelovaná součást pro obrábění v SolidCAMu

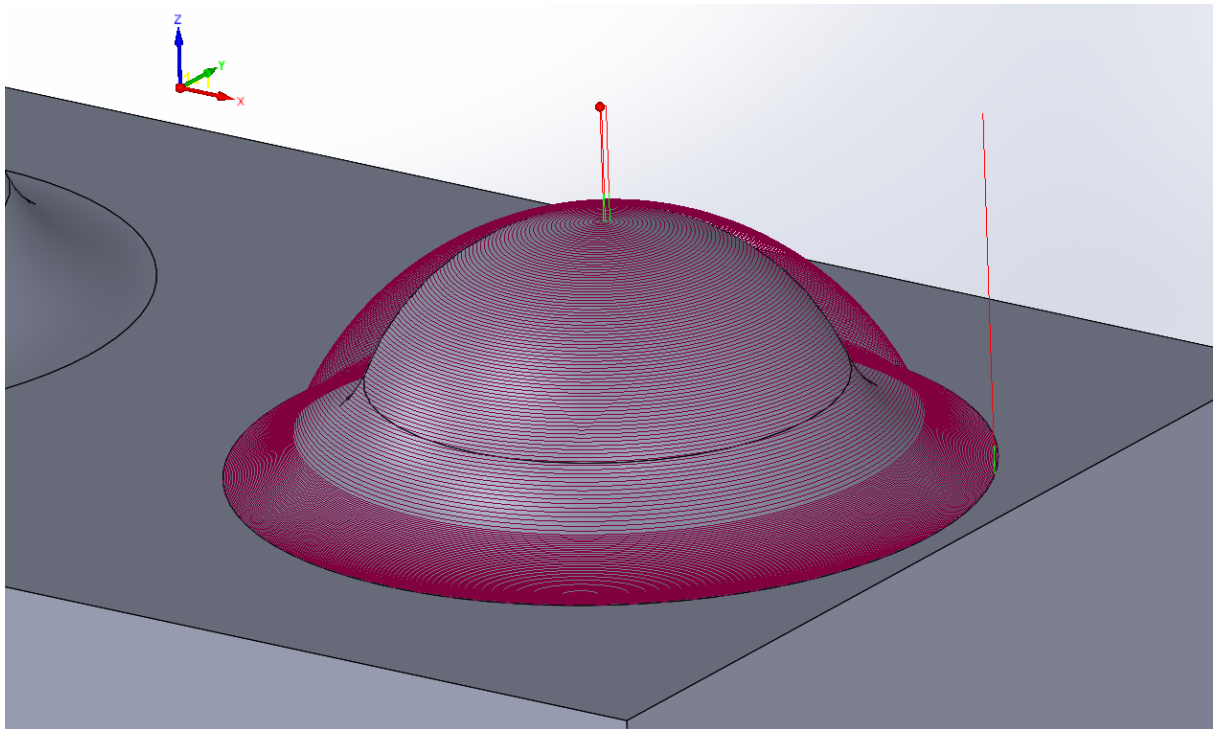
2.2 Obrobení součásti v SolidCAM

Pro vygenerování NC programu bylo potřeba obrobit vymodelovanou součást v SolidCAMu. Nejprve byla vytvořena sestava z vymodelované součásti do, které byl přidán polotovár přidáním nového dílu, který měl rozměry polotovaru a následně byl nastaven na průhledný a změněna barva, aby byl dobře rozeznatelný od obrobku. Poté byl vytvořen SolidCAM projekt pro frézování, kde byl nastaven obrobek, polotovár a nulový bod. Následně bylo provedeno obrábění v SolidCAMu. Pro hrubování byla použita operace 3D HSR a technologie HM hrubování. Zde byl použit nástroj toroidní fréza o průměru 16 mm. Krok dolů, tedy hloubka záběru byla zvolena 1 mm a přídavek na dno 0,5 mm. Dráhy hrubování jsou zobrazeny na Obr. 22. Pro dokončení dna byla použita také operace 3D HSR a technologie zbytkové hrubování. Po vygenerování drah nástroje byla vždy provedena kontrola v simulaci Solid Verify a v simulaci stroje.



Obr. 22: Dráhy nástroje při operaci 3D HSR technologie HM hrubování v SolidCAM

Jak již bylo zmíněno, pro zapnutí 3D nástrojové korekce v SolidCAMu jsou k dispozici pouze dvě operace, kterými jsou HSS a souvislé víceosé frézování. Úprava postprocesoru pro 3D korekci byla provedena pro tříosé obrábění, proto byla použita operace HSS pro 3D korekci. Jako první byla obrobena pravá polokoule, pro kterou byla zapnutá 3D korekce na špičku nástroje. V operaci HSS byla zvolena technologie rovnoběžně s plochou. Zde byla vybrána rovina pracovní plocha, což je plocha, která má být obrobena a jako řídicí plocha byla vybrána plocha dna. Zde byla použita kulová fréza o průměru 10 mm. Maximální boční krok byl nastaven na 0,5 mm. Jako metoda obrábění byla zvolena spirála, což je pro tento tvar nejvhodnější varianta, protože dráha frézy kopíruje kouli a nemá zde žádné zbytečné odjezdy a přejezdy. Dráhy nástroje jsou zobrazeny na Obr. 23.



Obr. 23: Dráhy nástroje pro kulovou plochu

Nejprve bylo nastaven boční krok na 0,5 mm, při kterém dolní zaoblení bylo dle simulace vyrobeno přesně, avšak na horní kulová části se zobrazovali nepřesnosti o velikosti mezi 0.1 mm a 0.2 mm. Z tohoto důvodu byl upraven boční krok na 0,2 mm. Po této úpravě simulace proběhla v pořádku.

2.3 Postprocessor a SolidCAM

V kapitole 1.4.3 je popsán postup, jakým způsobem je 3D nástrojová korekce v SolidCAMu zapínána. Avšak situace je taková, že i když je zaškrtnutá možnost 3D nástrojová korekce v parametrech dráhy nástroje v záložce modifikovat, NC program neobsahuje potřebné parametry, pro 3D korekci, tedy normálový vektor povrchu. Z toho důvod bylo nutné prozkoumat, jaké instrukce potřebuje SolidCAM, aby tyto data do NC programu zahrnul.

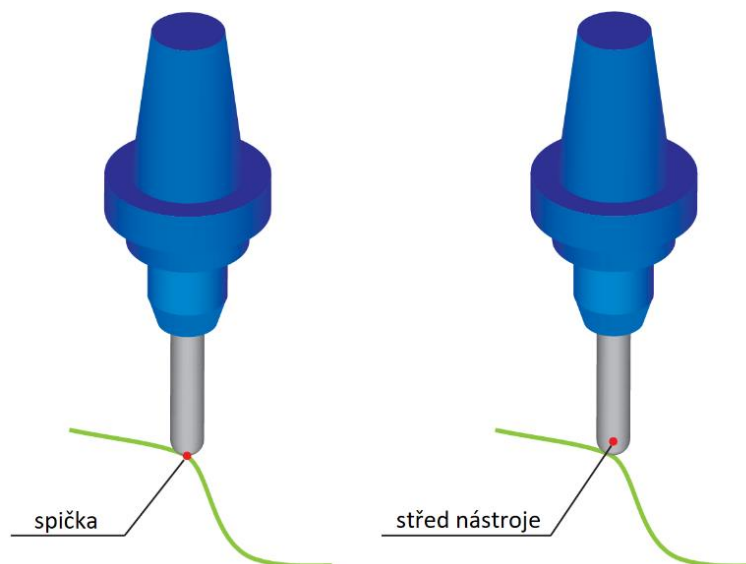
Rovnice (1) je zásadní pro 3D korekci, protože určuje vztah pro výpočet 3D korekce.

$$C = T + R \cdot N \quad (1)$$

Vysvětlení jednotlivých proměnných:

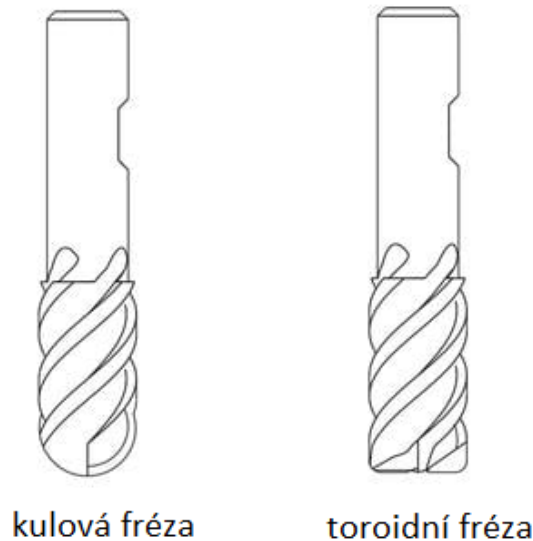
- C jsou nově vypočítané souřadnice středu nástroje
- T je poloha špičky nástroje
- R je poloměr nástroje
- N je normálový vektor

Nejvýznamnější informace pro zprovoznění 3D korekce nástroje v SolidCAMu bylo, jaké jsou nastavené proměnné v GPP pro 3D korekci. Rozdíl mezi nastavením na špičku nástroje a střed nástroje je znázorněn na Obr. 24.



Obr. 24: Korekce kulové frézy na špičku a na střed [9]

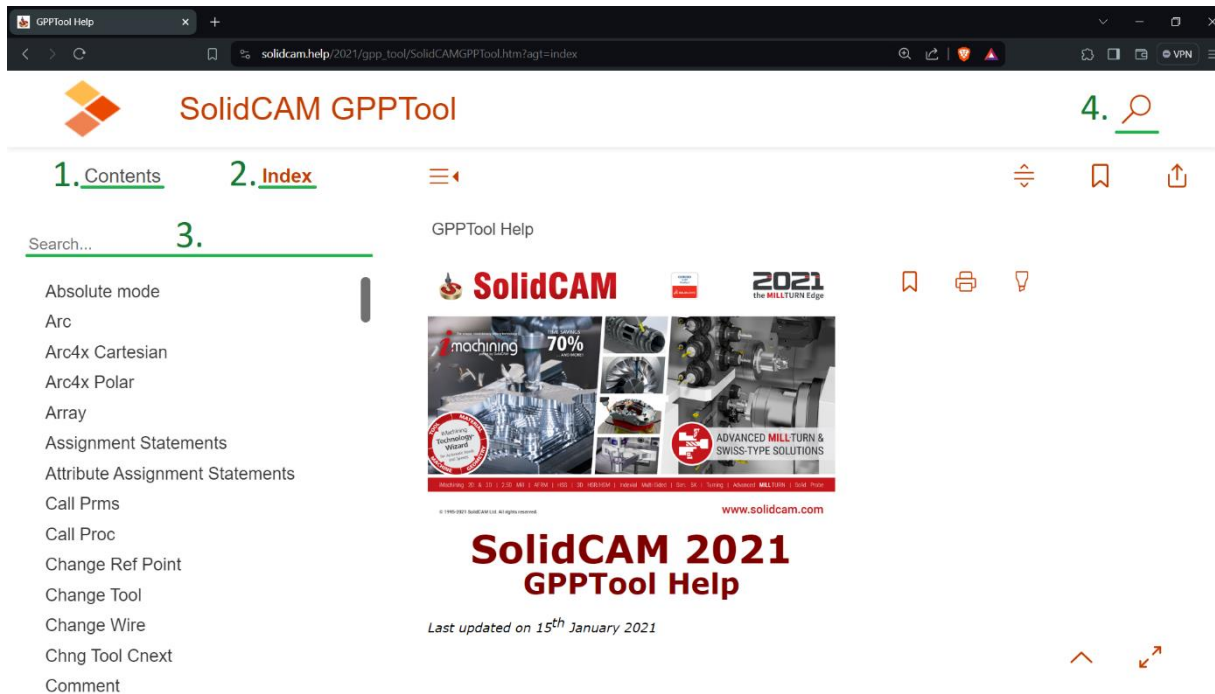
Pokud je zvolena možnost Řezná část (špička), dráha nástroje je vypočítávána na špičku nástroje a příkaz `type_offset_3D:tool_tip` je vypsán pod příkazem `@compensation_3d` v postprocesoru. V případě, že je zvolena možnost střed nástroje, dráha nástroje je vypočítávána na střed nástroje. Příkaz `type_offset_3D:tool_center` vypíše tuto korekci do G-kódu pod příkazem `@compensation_3d` v postprocesoru. Tuto korekci je možné použít pouze pro kulovou frézu a toroidní fréza, které jsou zobrazeny na Obr. 25. [9]



Obr. 25: Povolení frézy pro 3D nástrojovou korekci v SolidCAMu [18]

2.4 SolidCAM pomocník pro GPP

K úpravě postprocesoru bylo zásadní detailnější porozumění GPPL. Existuje přímo pomocná příručka, která obsahuje vysvětlení všech částí GPP kódu pro SolidCAM. Tato příručka existuje ve dvou variantách. První variantou je internetová stránka solidcam.help. Na této stránce se nachází záložka SolidCAM GPP Tool. Když je tato záložka rozkliknuta, zobrazí se úvodní stránka online příručky GPP pro SolidCAM. Tato úvodní stránka je vidět na Obr. 26.



Obr. 26: Stránka pro vysvětlení GPP kódu, zobrazena záložka Contents [15]

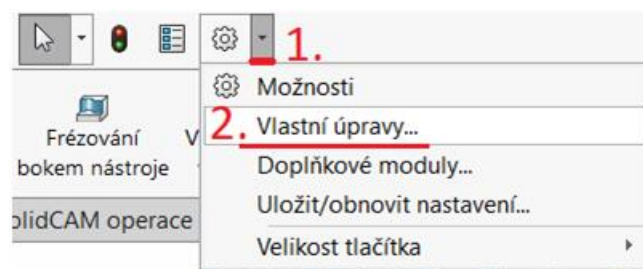
Na této stránce se nachází dvě hlavní záložky. Je zde záložka (1.) s názvem contents, ve které se nachází celý obsah kódu GPP přímo pro SolidCAM. Tyto informace jsou rozděleny přehledně do kategorií. Ve druhé záložce (2.) s názvem index jsou vypsány všechny dostupné procesy. Vzhledem k tomu, že GPP je celkem obsáhlý kód, pro nové uživatele, kteří ještě neví,

pod jakou záložkou se konkrétní informace nachází je toto velmi užitečná možnost vyhledávání. Možnost hledání jakékoliv informace je k dispozici po záložkou index (3.) a v pravém horním rohu (4.). [15]

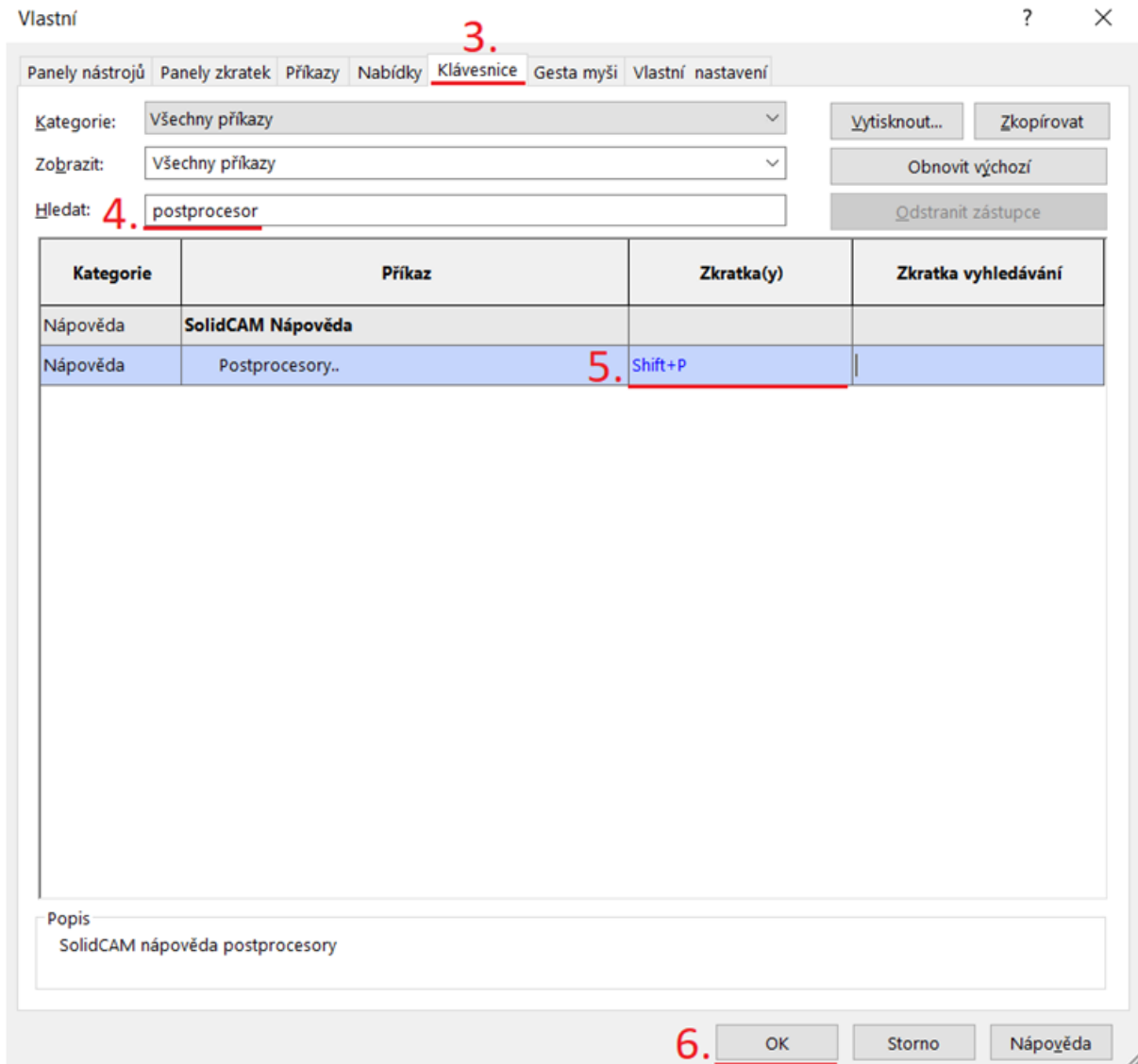
Druhou možností, jak zjistit informace pro GPP je přímo v SolidCAMu. Nejprve je potřeba otevřít CAM projekt. Postup nalezení informací pro GPP v SolidCAM je následující:

1. Na horní liště rozkliknout šipku vedle ikonky ozubeného kolečka.
2. Zvolit možnost vlastní úpravy.
3. Vybrat záložku klávesnice.
4. V poli hledat zadat postprocessor.
5. Nastavit klávesovou zkratku, pro zobrazení nápovědy pro postprocessor.
6. Potvrdit volbou ok.

Všechny kroky jsou znázorněny na Obr. 27 a Obr. 28.



Obr. 27: Postup pro nápovědu pro GPP v SolidCAM [SolidCAM 2020]



Obr. 28: Postup pro nápovědu pro GPP v SolidCAM [SolidCAM 2020]

2.5 Nastavení Visual Studio Code

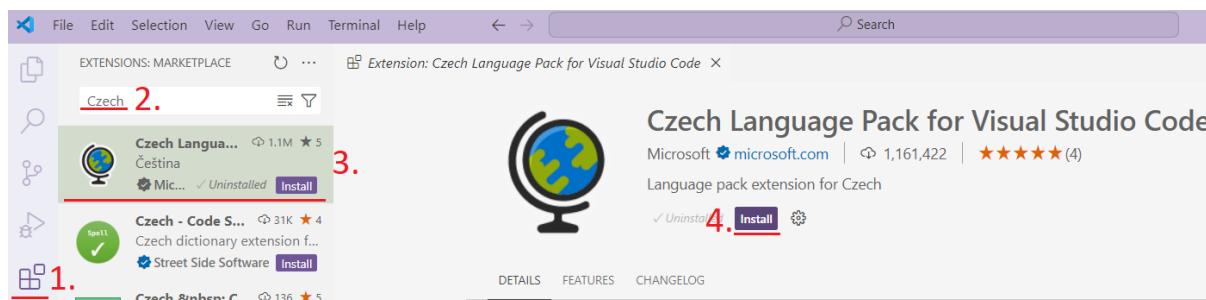
K realizaci úpravě postprocesoru byl důležitý výběr softwaru, pro efektivní programování. Byl zvolen software Visual Studio Code, který je široce využíván k programování. Tento software se také využívá pro programování v jazyce GPPL. Pro to aby byla úprava kódu postprocesoru co nejefektivnější, byla přidána rozšíření.

K tomu, aby mohl být soubor postprocesor efektivně upravován, bylo potřeba nastavení češtiny ve Visual Studio Code, přidat rozšíření pro GPP a upravit nastavení pro správné zobrazování kódu GPP.

2.5.1 Instalace českého jazyka

Postup pro nastavení českého jazyka, který je zobrazen na Obr. 29 je následující.

1. Na levé straně kliknout na ikonku extensions (možnost rozšíření).
2. Do vyhledávače napsat „Czech“.
3. Zvolit první možnost Czech Language Pack for Visual Studio Code .
4. Zvolit možnost Install.
5. Restartovat Visual Studio Code.



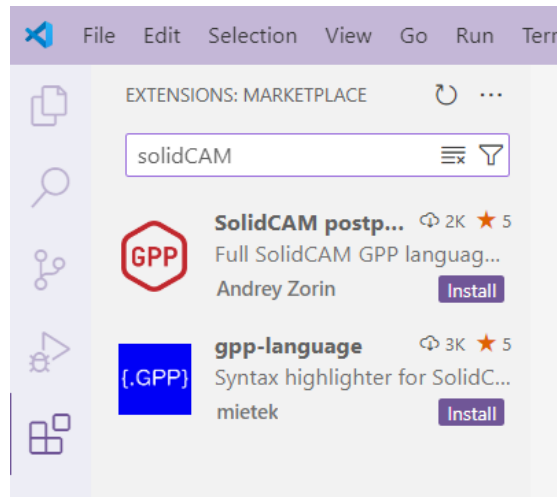
Obr. 29: Instalace českého jazyka ve Visual Studio Code [Visual Studio Code]

2.5.2 Instalace rozšíření pro GPP

Dalším rozšířením bylo pro jazyk GPP, které má funkce zvýraznění syntaxe GPP jazyka a poskytuje návrhy na dokončení při psaní kódu, které umožňuje efektivní kódování.

Existují dvě varianty, jakým způsobem přidat rozšíření určené pro jazyk postprocesoru GPP. Tyto varianty se rozdělují podle toho, kde bylo rozšíření získáno. Lze jej získat buďto přímo ve Visual Studio Code, nebo na internetových stránkách. Byly vyzkoušeny obě varianty. Následuje popsání jejich instalace a porovnání.

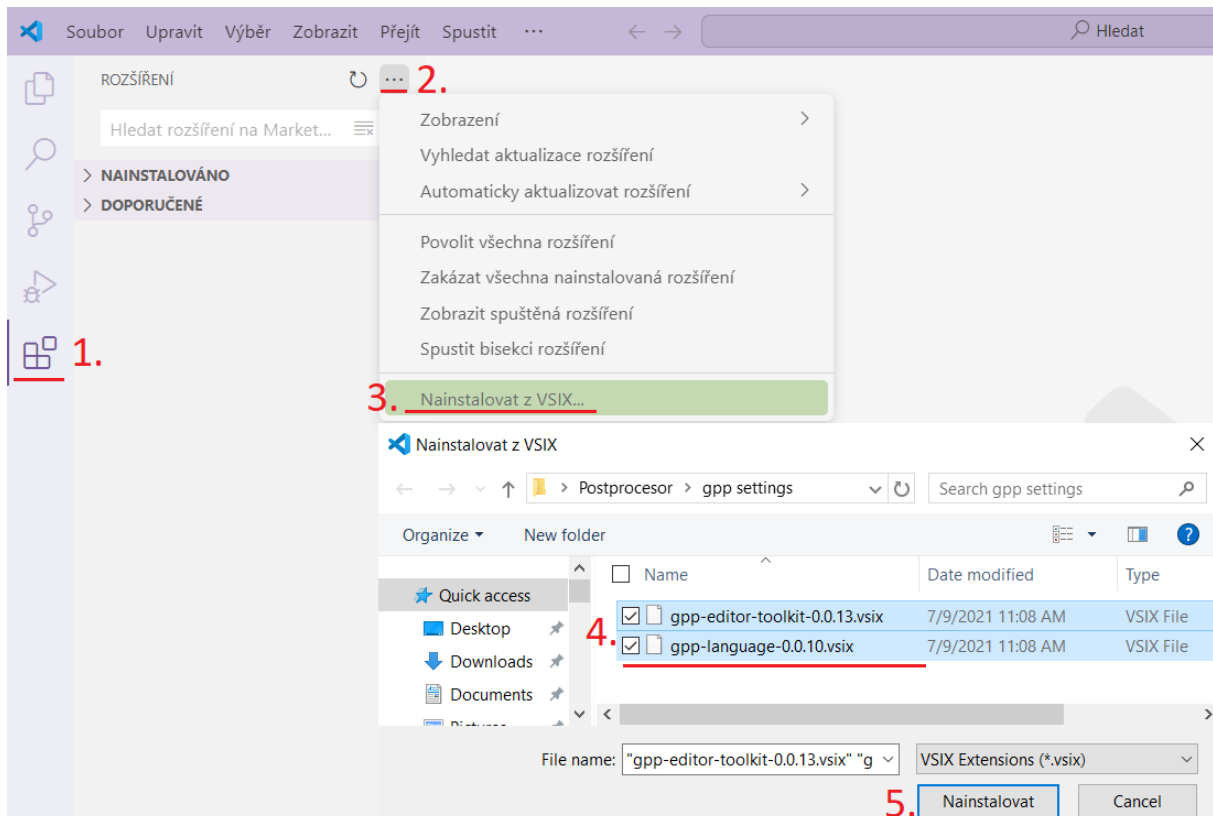
Jednou z možností, jak do Visual Studio Code je možné přidat rozšíření pro GPP jazyk je přímo ve Visual Studio Code v záložce rozšíření, kde se nachází řádek vyhledávání, do kterého se zadá „SolidCAM“ a je možné jej nainstalovat. Rozdíl mezi verzí ve Visual Studio Code a verzí z internetu je ve rozdílném zbarvení kódu. Výběru verze pro GPPL závisí na uživateli. Vyhledávání rozšíření ve Visual Studio Code je zobrazeno na Obr. 30.



Obr. 30: Vyhledání rozšíření pro GPP přímo ve Visual Studio Code [Visual Studio Code]

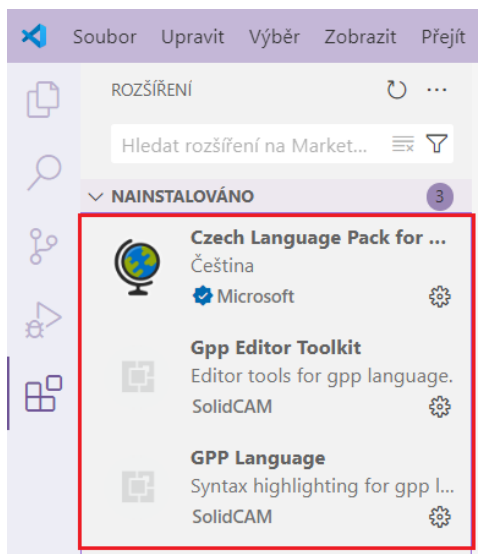
Druhou možností pro přidání rozšíření pro GPPL do Visual Studio Code, která byla zvolena pro tuto práci, bylo stažení souborů pro jazyk GPP a nástroje pro GPP ze stránky <https://solidcam.app.box.com>. Kroky postupu jsou znázorněny na Obr. 31. Dále pokračuje postup v prostředí Visual Studio Code. [17]

1. Zvolit volbu rozšíření
2. Zvolit tři tečky pro více možností
3. Nainstalovat z VSIX
4. Vybrat instalační soubory
5. Zvolit nainstalovat
6. Restartovat Visual Studio Code



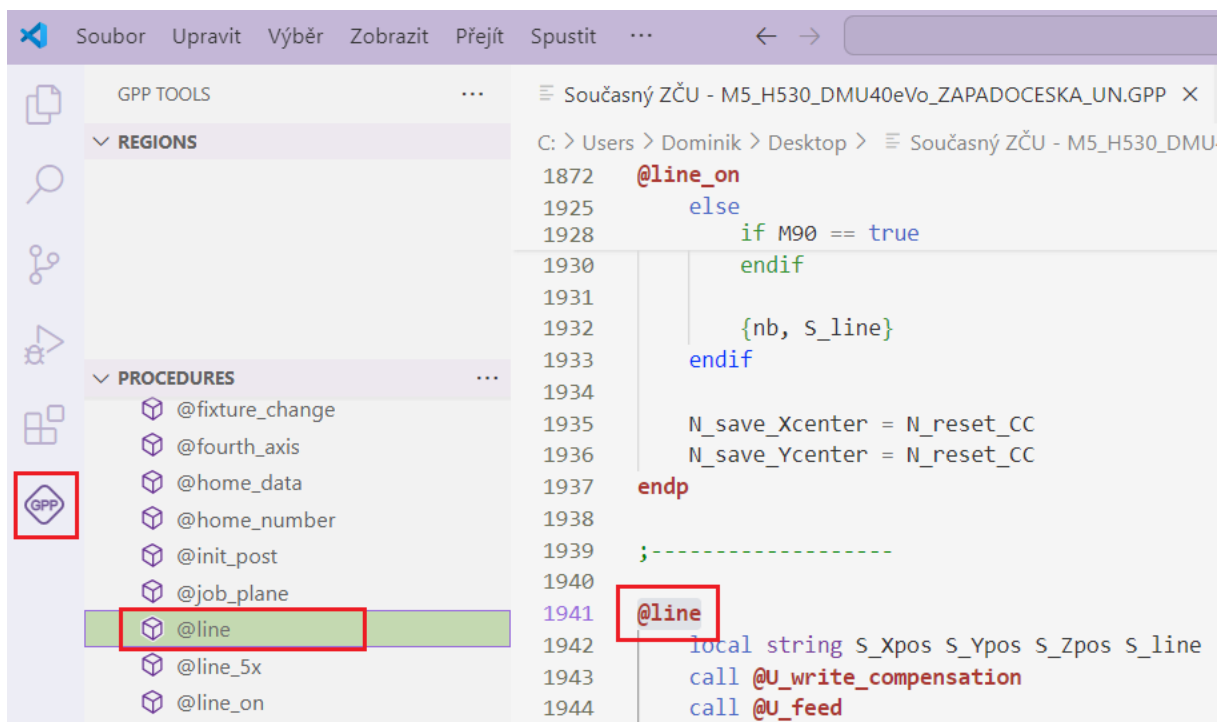
Obr. 31: Instalace rozšíření pro GPP ve Visual Studio Code [Visual Studio Code]

Nainstalovaná rozšíření jsou zobrazeny v záložce rozšíření, jak je vidět na Obr. 32.



Obr. 32: Nainstalované rozšíření ve Visual Studio Code [Visual Studio Code]

Obrázek Obr. 33 zobrazuje rozšíření editor toolkit. Toto rozšíření do Visual Studio Code přidalo seznam všech procedur, které byly nalezeny v otevřeném souboru GPP. Rozšíření editor toolkit urychluje vyhledávání jednotlivých procedur a rychlé procházení kódu. Stačí pouze kliknout v sloupci procedury, který se nachází na levé straně, na požadovanou proceduru a ta je ihned nalezena v GPP kódu.



Obr. 33: Zjednodušené vyhledávání procesů s rozšířením editor toolkit [Visual Studio Code]

2.5.3 Nastavení kódování znaků

Po nainstalování rozšíření, při otevření postprocesoru, písmena s diakritikou byly zobrazovány jako otazníky, jak je zobrazeno na Obr. 34. Bylo potřeba upravit nastavení, protože pokud by byl uložen soubor GPP s těmito otazníky, nemusel by správně fungovat.

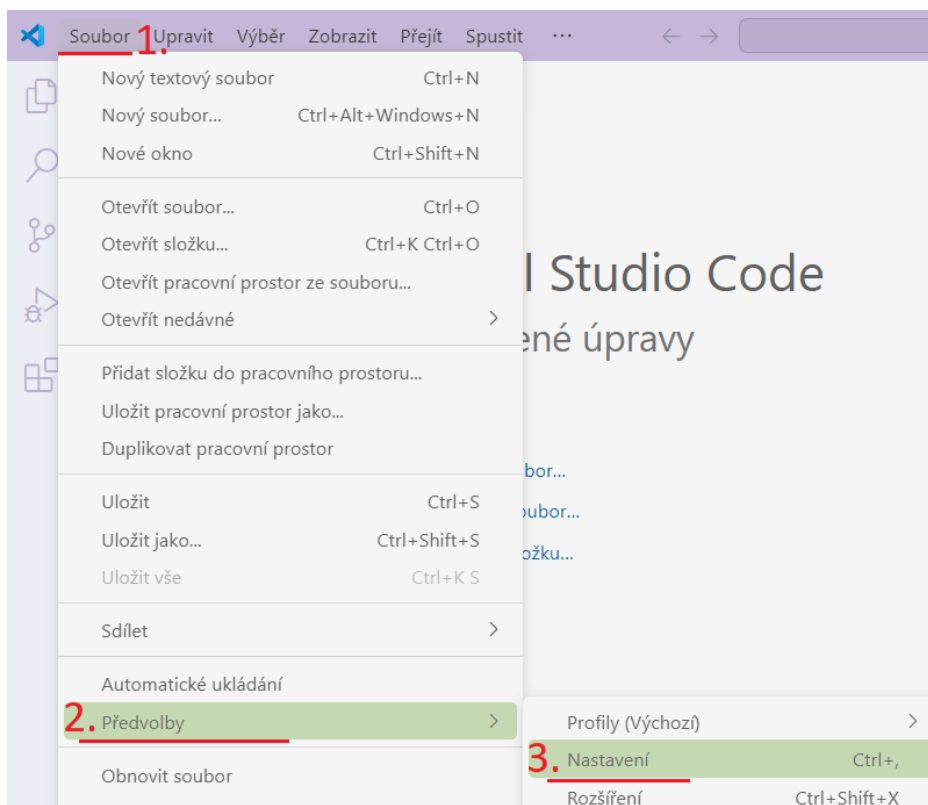
```
; ***** UPRAVY *****  
; radky 3478 - 3483 - nastaveni tlaku vnitřního chlazení  
; radek 1035 - shoulder_length opraveno na tool_length  
; radky 654, 655, 1019 - maximální hodnota síla nástroje pepsna z 1000 na 5000  
; radek 1563 - poidn odjezdu na konec programu  
; radek 1240 - oprava posuvu navratu z 2000 na 99999  
; 11.10.2022 - voln nástroje pomoc ID
```

Obr. 34: Nesprávné zobrazení znaků po otevření souboru GPP ve Visual Studio Code

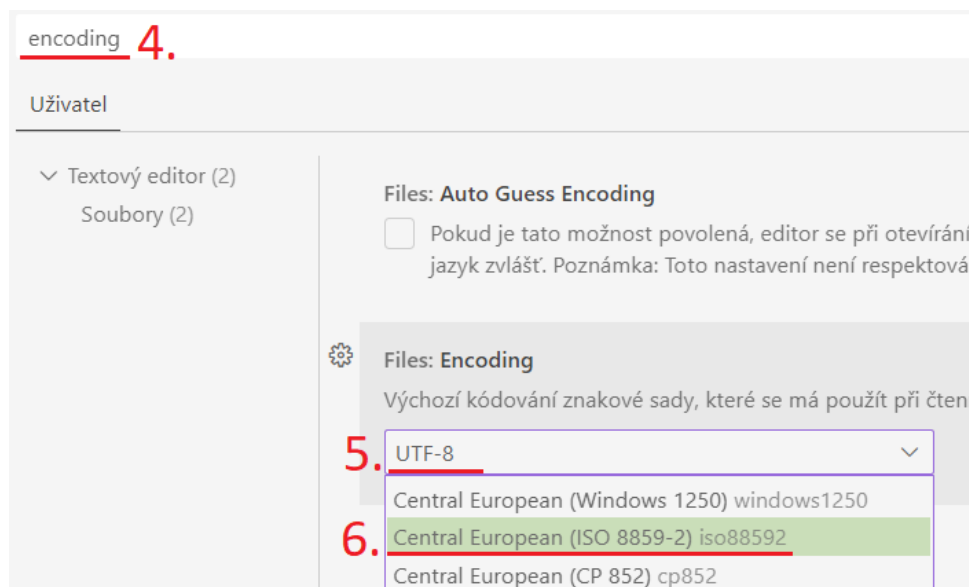
Studio Code používá pro otevření souboru encoder neboli dekódovač, podle kterého zobrazuje kód. Výchozím dekódovačem ve Visual Studio Code je UTF-8. Pro nastavení dekódovače pro český jazyk bylo potřeba provést následující kroky ve Visual Studio Code:

1. Zvolení možnosti soubor v levém horním rohu.
2. Vybrat možnost předvolby.
3. Zvolení možnosti nastavení.
4. Do vyhledávače v nastavení zadat slovo „encoding“.
5. Kliknout na UTF-8.
6. Vybrat jednu z možností, u které je napsáno „Central European“. Zde byla vybrána možnost ISO 8859-2.

Všechny kroky jsou zobrazeny na Obr. 35 a Obr. 36. Po této úpravě nastavení již byla diakritika zobrazena správně. [20] [21]



Obr. 35: Postup pro nastavení ve Visual Studio Code [Visual Studio Code]



Obr. 36: Postup pro nastavení dekódovače ve Visual Studio Code [Visual Studio Code]

2.6 Data pro 3D korekci

Dalším krokem bylo zjistit, jaké parametry SolidCAM již poskytuje pro 3D korekci do NC programu. V kapitole 1.5.1 je zmíněno, že v postprocesoru je funkce trace, která určuje jaké množství parametrů bude vypisováno do NC programu. V případě, že trace je nastaveno na hodnotu 5 jsou vypisována úplně všechny parametry.

V postprocesoru bylo nalezeno nastavení funkce trace, jak je zobrazeno na Obr. 37. Jak je patrné z obrázku, toto nastavení bylo nastavené pouze na komentář, tedy nemělo na postprocesor žádný vliv.

```
;;; global integer trace_num  
;;; Input "Trace --> 0 - 5" trace_num  
;;; trace "all":trace_num
```

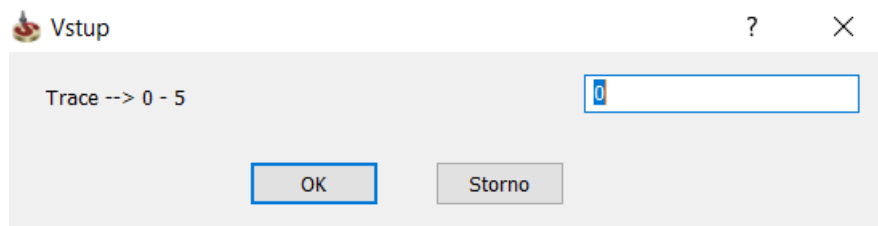
Obr. 37: Funkce trace v postprocesoru

Smazáním středníků byla funkce trace aktivována, jak je zobrazeno na Obr. 38. První řádka znamená nastavení proměnné trace_num na číselnou proměnnou. Funkce Input je určena k tomu, aby uživatel před vygenerováním mohl nastavit hodnotu funkce trace. Na poslední řádce je přiřazení zvolené hodnoty do funkce trace, která byla nastavena uživatelem.

```
global integer trace_num  
Input "Trace --> 0 - 5" trace_num  
trace "all":trace_num
```

Obr. 38: Zapnutá funkce trace

Touto úpravou postprocesoru se po zvolení volby generování NC programu v SolidCAM zobrazilo dialogové okno, které je vidět na Obr. 39.



Obr. 39: Dialogové okno pro nastavení hodnoty trace v SolidCAM 2020 [SolidCAM 2020]

Zde byla nastavena hodnota 5 a potvrzení možností ok, čímž byla vygenerována všechna data v programu SolidCAM do NC programu. Zde bylo zásadní najít data, která jsou potřeba pro 3D korekci. 3D nástrojová korekce potřebuje ke své funkci normálový vektor povrchu. V kapitole 1.5.2 jsou zmíněné dvě proměnné v GPP pro normálový vektor povrchu. Jsou to proměnné `xpos_normal`, `ypos_normal`, `zpos_normal` a `x_surface_normal`, `y_surface_normal`, `z_surface_normal`. Z tohoto důvodu byly vyhledávány tyto proměnné. V NC programu u procesu `@line` byly vždy zobrazeny proměnné `surface_normal`, jak je zobrazeno na Obr. 40.

```
==> xpos:107.986T ypos:45.177T zpos:-4.014F feed:2546.479F feed_teeth:0.003F spin:6366.198F
.> xhpos:107.986T yhpos:45.177T zhpos:-4.014F
.> xmpos:357.414T ympos:294.958T zmpos:-505.764F
.> xopos:107.986T yopos:45.177T zopos:-4.014F
.> xlpes:107.986T ylpes:45.177T zlpes:-4.014F
.> xtpos:357.414T ytpes:294.958T ztpes:-376.464F
.> xtopos:107.986T ytopos:45.177T ztopos:125.286F
.> xrpos:107.986T yrpos:45.177T zrpos:-134.114F
.> xrtpos:107.986T yrtpos:45.177T zrtpos:-4.814F
.> x_surface_point:109.012T y_surface_point:46.273T z_surface_point:-3.785T
.> x_surface_normal:-0.2051967T y_surface_normal:-0.2192467T z_surface_normal:0.9538476T
.> next_direction:137.193 feed_type:'feed_rate'
```

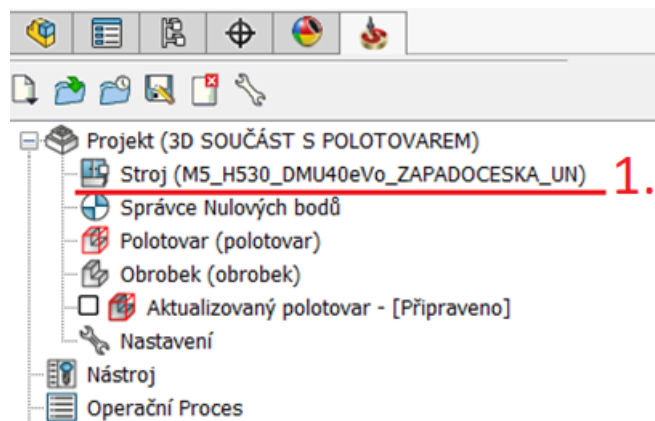
Obr. 40: Normálový vektor v NC programu po nastavení trace na hodnotu 5

2.7 Úprava postprocesoru

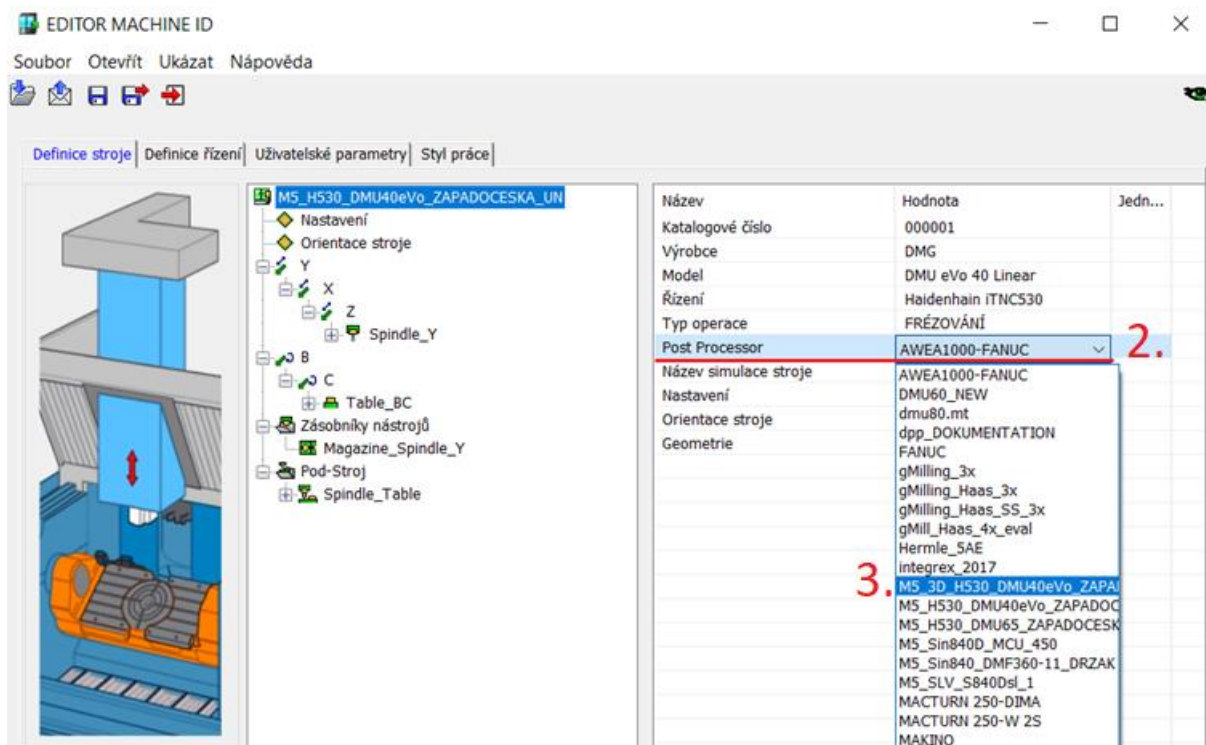
Dalším krokem bylo pochopit, jakým způsobem je napsaný postprocesor a na tomto základě jej upravit tak, aby normálový vektor povrchu byl vypsán do NC programu. Pro výrobu součásti s trojrozměrnou korekcí byl použit stroj Dmu 40 eVo – DMG MORI. Tento stroj využívá řídicí systém Heidenhain TNC 530 a postprocesor učený pro tento stroj. Pro testovací účely byla vytvořena kopie postprocesoru. Tato kopie byla použita pro úpravy a testování. Přímo v prostředí SolidCAMu, bylo možné měnit aktuální postprocesor, který je používán. Postup pro zvolení postprocesoru v SolidCAMu je následující:

1. V levé části přímo pod projektem rozkliknout aktuální stroj a zobrazí se VMID nastavení.
2. V nastavení VMID rozkliknout záložku u Post Procesoru
3. Vybrat požadovaný postprocesor

Tyto body jsou zobrazeny na Obr. 41 a Obr. 42. Testování úpravy GPP probíhalo takovým způsobem, že byl vždy upraven postprocesor a následně vygenerován NC program.



Obr. 41: Výběr stroje [SolidCAM 2020]



Obr. 42: Výběr postprocesoru [SolidCAM 2020]

2.7.1 Pochopení příkazu @line v postprocesoru

K úpravě postprocesoru bylo potřeba pochopit strukturu a funkci stávajícího postprocesoru bez 3D korekce pro stroj DMU 40 eVo. Hlavní část, která byla upravována je v procesu @line. Tento proces má 3 části, které jsou popsány v následujících kapitolách. Při úpravě postprocesoru byly prováděny úpravy pro souřadnice X, Y a Z, kde často byly úpravy pro každou z těchto souřadnic stejné. Z tohoto důvodu, jsou některé změny popsány pouze pro souřadnici X.

2.7.1.1 1. část procesu @line

Na Obr. 43 je zobrazena první část procesu @line. Jak již bylo zmíněno v kapitole 1.5.2 jazyk GPP podporuje čtyři druhy operátorů, kterými jsou:

- Celé číslo (integer)
- Číslo (numeric)
- Logické (logical)
- Řetězec (string)

Na první řádce této části kódu se nachází lokální řetězcové proměnné S_Xpos S_Ypos S_Zpos a S_line. Lokální znamená, že nastavuje proměnnou na místní typ, který může být použit pouze v určité proceduře, v tomto případě v @line.

```
@line
local string S_Xpos S_Ypos S_Zpos S_line
call @U_write_compensation
call @U_feed
```

Obr. 43: První část procesu @line

“Call“ je příkaz pro volání procedury. Proces @U_write_compensation upravuje 2D korekci. Proces @U_feed má funkci zaokrouhlení posuvu.

2.7.1.2 2. část procesu @line

Tato část procesu @line je určena pro nastavení hodnot proměnných S_Xpos, S_Ypos a S_Zpos. Do těchto proměnných jsou ukládány aktuální souřadnice X, Y a Z. Pro tento účel jsou zde dvě varianty. První varianta je na obrázku Obr. 44 a druhá varianta je na obrázku Obr. 45.

```
; rezim GPP pos_to_machine = Y
if L_pos_to_machine_Y == true and L_cycle_19 == false
  if change(xlpos) == true
    S_Xpos = " " + S_I + "X" + tostr(xlpos:xpos_f)
  else
    S_Xpos = ""
  endif

  if change(ylpos) == true
    S_Ypos = " " + S_I + "Y" + tostr(ylpos:ypos_f)
  else
    S_Ypos = ""
  endif

  if change(zlpos) == true
    S_Zpos = " " + S_I + "Z" + tostr(zlpos:zpos_f)
    N_save_Z = zlpos
  else
    S_Zpos = ""
  endif
```

Obr. 44: Nastavení proměnných s xlpos, ylpos a zlpos

```
else ; rezim GPP pos_to_machine = N

    if change(xpos) == true
        S_Xpos = " " + S_I + "X" + tostr(xpos:xpos_f)
    else
        S_Xpos = ""
    endif

    if change(ypos) == true
        S_Ypos = " " + S_I + "Y" + tostr(ypos:ypos_f)
    else
        S_Ypos = ""
    endif

    if change(zpos) == true
        S_Zpos = " " + S_I + "Z" + tostr(zpos:zpos_f)
        N_save_Z = zpos
    else
        S_Zpos = ""
    endif
```

Obr. 45: Natavení proměnných s xpos, ypos a zpos

Zde je zásadním rozdílem, že v první variantě je proměnná xlpos, ylpos, zlpos a v druhé variantě jsou proměnné xpos, ypos, zpos. Nejprve bylo potřeba pochopit jednotlivé části kódu a následně to, jakým způsobem funguje celá část kódu. V následujících bodech jsou vysvětleny jednotlivé části.

- V první variantě (Obr. 44) jsou důležité proměnné xlpos, ylpos a zlpos. Jedná se o číselné proměnné, do kterých je ukládán souřadnicový systém posunuté pracovní roviny, jak bylo popsáno v kapitole 1.5.2.
- V druhé variantě (Obr. 45) jsou oproti první variantě proměnné xpos, ypos a zpos. Jedná se o číselné proměnné, které jsou určeny pro aktuální pozici nástroje na konci pohybu, jak bylo popsáno v kapitole 1.5.2.
- Dalšími proměnnými jsou xpos_f, ypos_f a zpos_f, což jsou řetězcové proměnné, které zobrazují hodnoty X, Y a Z.
- Funkce change, která porovnává to, jestli nastala změna u proměnné, která je v závorce.
- Dále je zde funkce **tostr**. Tato funkce převádí číselné výrazy na řetězce podle specifického formátu, který je definován jako volitelný argument. Účelem této funkce je, že převádí číslo na řetězec. Tato funkce se zapisuje ve tvaru „tostr (number [:format])“. Za „number“ je dosazována číselná hodnota a za „format“ to, jak má být proměnná vypisována, což bylo popsáno v kapitole 1.5.4. Pokud je proměnná xpos_f definovaná v kódu, její hodnota bude určovat to, jak se xlpos převede na řetězec. Příklad výstupu bude záviset na aktuální hodnotě xpos_f. Pokud je xpos_f například "0.00", pak výsledek funkce tostr by mohl být řetězec, který zobrazuje xlpos s dvěma desetinnými místy. Úprava funkce tostr je popsána v kapitolách 2.7.3 a 2.7.4. [15]

V této části, jsou tři podmínky, jak je zobrazeno na obrázcích Obr. 44 a Obr. 45. Každá z těchto podmínek funguje tak, že pokud nastane změna v souřadnicích X, Y nebo Z, pak je do

proměnných S_Xpos, S_Ypos, a S_Zpos načtena příslušná hodnota souřadnice Y, Y nebo Z. Cílem bylo upravit tento kód takovým způsobem, aby kromě běžných souřadnic X, Y a Z, byl vypisován do NC programu i normálový vektor povrchu, který je nutný pro zprovoznění 3D korekce. Při této verzi postprocesoru i v případě, že byla v SolidCAMu zapnutá 3D nástrojová korekce, data pro 3D korekci nebyla vypisována do NC programu. Část NC programu, kdy nebyly data pro 3D korekci vypisovány je zobrazena na Obr. 46.

```
36 L X+112.799 Y+50.387
37 L X+112.84 Y+50.353
38 L X+112.874 Y+50.315
39 L X+112.912 Y+50.263
40 L X+112.942 Y+50.211
41 L X+112.954 Y+50.185
42 L X+112.971 Y+50.13
```

Obr. 46: NC program bez 3D korekce

2.7.1.3 3. část procesu @line

Třetí část procesu @line je zobrazena na Obr. 47. V této části jsou dva zásadní řádky, které mají souvislost s úpravou kódu. Prvním řádkem je řádek s proměnnou S_line, který je určen pro načtení všech informací, které mají být vypsány do řádku NC programu v lineárním bloku kódu. Druhý podstatný řádek obsahuje "{nb, S_line}". "nb" je funkce určená pro vytvoření nových bloků NC programu. "{nb, S_line}" je tedy určeno pro vytvoření nového bloku kódu, do kterého jsou vypsána data z proměnné S_line. Kromě lineárních proměnných, je zde proměnná S_comp, která je určena pro 2D korekci a proměnná S_feed, která je určena pro informace o posuvu. S_line byla upravována pro vypsání normálového vektoru.

```
if L_first_rapid_move
  call @U_start_of_job
  L_first_rapid_move = false
else
  S_line = "L" + S_Xpos + S_Ypos + S_Zpos + S_comp + S_feed

  if S_comp == " RL" or S_comp == " RR"
    S_line = S_line + " M110"
  endif

  if M90 == true
    S_line = S_line + " M90"
  endif

  {nb, S_line}
endif
```

Obr. 47: Třetí část procesu @line

2.7.2 Testování proměnných normálového vektoru

První krokem bylo provést testování, jak jsou vypisovány všechny typy proměnných pro normálový vektor, kterými jsou:

- x_surface_normal, y_surface_normal a z_surface_normal.
- xpos_normal, ypos_normal a zpos_normal.

Do procesu @line byl přidán příkaz pro vypsání těchto proměnných do NC programu, jak je zobrazeno na Obr. 48.

```
{n1, 'x_surface_normal = 'x_surface_normal} ;vypíše hodnotu x_surface_normal
{n1, 'y_surface_normal = 'y_surface_normal} ;vypíše hodnotu y_surface_normal
{n1, 'z_surface_normal = 'z_surface_normal} ;vypíše hodnotu z_surface_normal

{n1, 'xpos_normal = 'xpos_normal}           ;vypíše hodnotu xpos_normal
{n1, 'ypos_normal = 'ypos_normal}           ;vypíše hodnotu ypos_normal
{n1, 'zpos_normal = 'zpos_normal}           ;vypíše hodnotu zpos_normal
```

Obr. 48: Příkaz pro vypsání normálového vektoru

Po této úpravě, byly do NC programu vypsány souřadnice normálového vektoru, jak je zobrazeno na Obr. 49.

```
x_surface_normal = 0.0031
y_surface_normal = -0.0011
z_surface_normal = 1
xpos_normal = 0
ypos_normal = 0
zpos_normal = 0
38 L X+112.552 Y+49.95 F+Q1
x_surface_normal = 0.0024
y_surface_normal = -0.0023
z_surface_normal = 1
xpos_normal = 0
ypos_normal = 0
zpos_normal = 0
39 L X+112.514 Y+49.93
```

Obr. 49: Výpis normálových vektorů do NC programu

Zde bylo zjištěno, že proměnné xpos poskytovaly nulové hodnoty. Naproti tomu proměnné surface_normal hodnoty poskytovaly. Z tohoto vyplývá, že bylo potřeba do postprocesoru přidat proměnné x_surface_normal, y_surface_normal a z_surface_normal.

2.7.3 Úprava @line pro 3D korekce

V @line byly nastavené proměnné S_Xpos, S_Ypos a S_Zpos, které vypisují lineární souřadnice X, Y a Z. V první části procesu @line bylo potřeba vytvořit proměnné pro normálový vektor, které budou zadané do proměnné S_line a byly nazvány Xsurface_normal, Ysurface_normal a Zsurface_normal. Tyto nové proměnné byly přidány ke stávajícím proměnným do lokálního řetězce (local string), jak je zobrazeno na Obr. 50.

```
@line
local string S_Xpos S_Ypos S_Zpos Xsurface_normal Ysurface_normal Zsurface_normal S_line
call @U_write_compensation
call @U_feed
```

Obr. 50: Vytvoření proměnných pro normálový vektor

Dále bylo zjištěno, jakým způsobem upravit druhou část, která přidá hodnotu těmto nově vytvořeným proměnným. Bylo důležité zjistit, jakou proměnou zadat do závorky k funkci change a funkci tostr. Pro lineární osy X, Y a Z jsou k funkcím change a tostr přiřazeny číselné proměnné xlpos, ylpos, zlpos nebo xpos, ypos, zpos.

Ve funkci tostr je xpos_f, ypos_f a zpos_f což jsou řetězcové proměnné pro určení formátu číselné proměnné. Proměnné x_surface_normal, y_surface_normal a z_surface_normal jsou číselnými proměnnými. Tyto proměnné byly přidány k funkci change a tostr jak je zobrazeno na Obr. 51. Dále zde byly nahrazeny proměnné S_Xpos, S_Ypos a S_Zpos, vytvořenými proměnnými Xsurface_normal, Ysurface_normal a Zsurface_normal pro normálový vektor.

Název souřadnic byl změněn z původních X, Y a Z na NX, NY a NZ. Celá úprava je zobrazena a popsána na Obr. 51.

```
if change(x_surface_normal) == true ;změna proměnné change
    Xsurface_normal= " " + S_I + "NX" + tostr(x_surface_normal)
    ;změna z S_Xpos na Xsurface_normal, upraveno z "X" na "NX" a změna proměnné tostr
else
    Xsurface_normal = "" ;změna z S_Xpos na Xsurface_normal
endif

if change(y_surface_normal) == true ;změna proměnné change
    Ysurface_normal = " " + S_I + "NY" + tostr(y_surface_normal)
    ;změna z S_Ypos na Ysurface_normal, upraveno z "Y" na "YX" a změna proměnné tostr
else
    Ysurface_normal = "" ;změna z S_Ypos na Ysurface_normal
endif

if change(z_surface_normal) == true ;změna proměnné change
    Zsurface_normal= " " + S_I + "NZ" + tostr(z_surface_normal)
    ;změna z S_Zpos na Ysurface_normal, upraveno z "Z" na "ZX" a změna proměnné tostr
else
    Zsurface_normal = "" ;změna z S_Zpos na Zsurface_normal
endif
```

Obr. 51: Upravená druhá část @line

Do proměnné S_line byly přiřazeny nově vytvořené proměnné Xsurface_normal, Ysurface_normal a Zsurface_normal. Tato úprava je zobrazena na Obr. 52.

```
S_line = "LN" + S_Xpos + S_Ypos + S_Zpos + Xsurface_normal + Ysurface_normal + Zsurface_normal + S_comp + S_feed
```

Obr. 52: Upravená proměnná S_line

Následně byl vygenerován NC program. Část NC programu je zobrazena na Obr. 53. kde je patrné, že hodnoty pro normálový vektor byly vygenerovány. Tyto hodnoty byly vypsané i v případě, že 3D korekce byla v SolidCAM vypnutá. Tento problém je řešen v kapitole 2.7.5. Dalším krokem bylo, nastavit vypisování normálového vektoru na sedm desetinných míst, jak požaduje Heidenhain TNC 530.

```
36 LN X+112.799 Y+50.387 TX0.014 TY0.018 TZ1.  
37 LN X+112.84 Y+50.353 TX0.015 TY0.016 TZ1.  
38 LN X+112.874 Y+50.315 TX0.017 TY0.014 TZ1.  
39 LN X+112.912 Y+50.263 TX0.019 TY0.012 TZ1.  
40 LN X+112.942 Y+50.211 TX0.02 TY0.01 TZ1.  
41 LN X+112.954 Y+50.185 TX0.021 TY0.008 TZ1.  
42 LN X+112.971 Y+50.13 TX0.021 TY0.006 TZ1.
```

Obr. 53: Vygenerovaná část programu s LN a normálovým vektorem

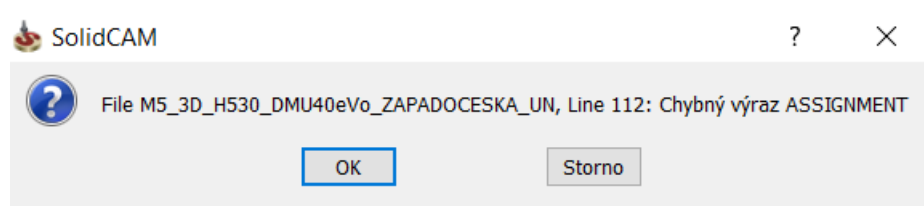
2.7.4 Nastavení normálového vektoru na 7 desetinných míst

Nastavení proměnných je běžně provedeno v procesu @init_post. První pokus nastavení desetinných míst byl uskutečněn v této proměnné. Toto nastavení normálového vektoru povrchu je zobrazeno na Obr. 54.

```
; GPPL proměnné  
x_surface_normal = '5.7'  
numeric_def_f = '5.4/0(P)'  
integer_def_f = '5.0(P)'  
gcode_f = '2/2.0(P)'
```

Obr. 54: Nastavení x_surface_normal v @unit_post

Po spuštění generování programu se zobrazila chybová hláška, která je na Obr. 55.



Obr. 55: Chybová hláška po nastavení x_surface_normal v @unit_post

Kvůli této chybové hlášce bylo nutné hledat jiné řešení. Dalším způsobem, jak nastavit počet desetinných míst, je přímo u proměnné v @line ve funkci tostr. Nastavení funkce tostr je: tostr (number [:format]), kde proměnná format určuje to, v jakém formátu bude vypisovaná proměnná. Za pole number bylo třeba dosadit číselnou hodnotu, v tomto případě normálový vektor povrchu a do pole format nastavit to, jak má být normálový vektor vypisován do NC programu. Na Obr. 56 je zobrazena tato úprava postprocesoru.


```
if change(x_surface_normal) == true
    Xsurface_normal = " " + S_I + "NX" + tostr(x_surface_normal:'3.7')
    ;přidáno '3.7' pro vypsání na 7 desetinných míst
else
    Xsurface_normal = ""
endif
```

Obr. 56: Úprava pro 7 desetinných míst

Poté byl vygenerován NC program. Jeho část je zobrazena na Obr. 57.

```
36 LN X+112.799 Y+50.387 TX0.0136045 TY0.0175706 TZ0.9997531
37 LN X+112.84 Y+50.353 TX0.0154751 TY0.0160412 TZ0.9997516
38 LN X+112.874 Y+50.315 TX0.0170224 TY0.0143183 TZ0.9997526
39 LN X+112.912 Y+50.263 TX0.0187188 TY0.0119486 TZ0.9997534
40 LN X+112.942 Y+50.211 TX0.0200906 TY0.0095987 TZ0.9997521
41 LN X+112.954 Y+50.185 TX0.0206396 TY0.0084308 TZ0.9997514
42 LN X+112.971 Y+50.13 TX0.0214226 TY0.0059103 TZ0.999753
```

Obr. 57: Vygenerovaný NC program po úpravě na 7 desetinných míst

Zde je důležitá maličkost, která není správně a kvůli které by program nemusel správně fungovat. Tento problém je zobrazen na Obr. 57 na řádce 42. Zde je hodnota normálového vektoru povrchu $TZ = 0,999753$, což znamená, že je vypsán pouze na 6 desetinných. Správně by měl být vypisován na sedm desetinných míst. To znamená, že jsou vypisovány čísla od jedné do devíti a nula na konci není vypisována. Z tohoto důvodu byl prozkoumán GPP, aby byly vypisovány i koncové nuly. Formát GPP pro vypisování nul je:

$$\{-21.4:'4.4z'\} \implies -21.4000$$

Z toho plyne, že na konec formátu pro normálový vektor bylo přidáno „z“, jak je zobrazeno na Obr. 58. [15]

```
if change(x_surface_normal) == true
    Xsurface_normal = " " + S_I + "NX" + tostr(x_surface_normal:'3.7z')
    ;přidáno z pro vypisování 0
else
    Xsurface_normal = ""
endif
```

Obr. 58: Upravený postprocesor pro vypsání koncových nul pro normálový vektor

Po této úpravě je vidět NC program na Obr. 59. Zde již byl vygenerován program s 3D korekcí jak požaduje Heidenhain TNC 530.

```
36 LN X+112.799 Y+50.387 TX0.0136045 TY0.0175706 TZ0.9997531
37 LN X+112.84 Y+50.353 TX0.0154751 TY0.0160412 TZ0.9997516
38 LN X+112.874 Y+50.315 TX0.0170224 TY0.0143183 TZ0.9997526
39 LN X+112.912 Y+50.263 TX0.0187188 TY0.0119486 TZ0.9997534
40 LN X+112.942 Y+50.211 TX0.0200906 TY0.0095987 TZ0.9997521
41 LN X+112.954 Y+50.185 TX0.0206396 TY0.0084308 TZ0.9997514
42 LN X+112.971 Y+50.13 TX0.0214226 TY0.0059103 TZ0.9997530
```

Obr. 59: Upravený NC program s koncovou nulou

2.7.5 Podmínka pro 3D korekci

Upravený GPP ještě nefungoval správně, protože jak bylo zmíněno, normálový vektor byl vypisován i v případě, že v SolidCAMu byla 3D korekce vypnutá. Z tohoto důvodu byla vytvořena podmínka, která zaručila, aby 3D korekce byla do NC programu zahrnuta pouze v případě, když je zapnutá. Formát podmínky v GPP pro SolidCAM je zobrazen na Obr. 60.

```
if <podmínka 1> ;pokud je splněna podmínka 1 provede požadavek 1
    požadavek 1
elseif <podmínka 2> then ;jinak pokud je splněna podmínka 2 provede požadavek 2
    požadavek 2
else ;jinak provede požadavek 3
    požadavek 3
endif ;konec podmínky
```

Obr. 60: Struktura podmínky [15]

V tento moment bylo potřeba přijít na to, jaká je podmínka pro situaci, když je zapnutá korekce. V kapitole 1.4.3 je zmíněno, že 3D korekci je v SolidCAMu možné nastavit buď na špičku nebo na střed nástroje. Naskytla se otázka, co zadat do postprocesoru, aby rozeznal kdy je zapnutá korekce na špičku nástroje nebo korekce na střed nástroje. Příkazy, které SolidCAM používá pro 3D korekci jsou **type_offset_3D:tool_tip**, **type_offset_3D:tool_center** a proces **@compensation_3d**.

V SolidCAM pomocníku pro GPP se nevyskytují žádné informace k procesu **@compensation_3d**. Jsou zde informace pouze pro proces **@compensation**, který je pro 2D korekci, což v tomto případě nebylo užitečné. V případě hledání informací k příkazu **type_offset_3D** se zde také nenacházely žádné informace. Byly však nalezeny informace k proměnné **tool_tip**, které jsou v Tab. 4. [15]

Tab. 4: Proměnná tool_tip [15]

Proměnná	Typ	Popis
tool_tip_x tool_tip_y tool_tip_z	Číslo (numeric)	Vzdálenost mezi špičkou nástroje a: 1. Poloha stanice 2. upevňovací bod nástroje

2.7.6 Testování proměnných pro 3D korekci

Ke zprovoznění podmínky byly zjišťované hodnoty proměnných tool_tip, tool_center a type_offset_3D. Při prvním pokusu o generování NC kódu s těmito proměnnými se objevila chybová hláška, že tool_tip a tool_center není deklarována. Tento problém byl vyřešen tak, že na začátku postprocesoru v procesu @init_post, byly přidány proměnné tool_tip a tool_center do global numeric. Přidání proměnných je zobrazeno na Obr. 61.

```
global numeric tool_tip tool_center
```

Obr. 61: Přidání proměnných pro špičku a střed nástroje do @init_post

Na konec procesu @start_program byla přidána sekvence, která je vidět na Obr. 62.

```
{nl, 'hodnota tool_tip = 'tool_tip} ;vypíše hodnotu tool_tip  
{nl, 'hodnota tool_center = 'tool_center} ;vypíše hodnotu tool_center  
{nl, 'hodnota type_offset_3D = 'type_offset_3D} ;vypíše hodnotu type_offset_3D
```

Obr. 62: Část programu pro otestování hodnot pro 3D korekci

Tato sekvence měla za účel vypsát do NC kódu hodnoty tool_tip, tool_center a type_offset_3D. Avšak při tomto nastavení byla stále hodnota všech tří proměnných rovna nule, i přestože byla zkoušena v SolidCAMu možnost jak s vypnutou, tak se zapnutou 3D korekcí. Příčinou bylo, že hodnoty byly vypisovány ještě před tím, než začala operace, pro kterou byla 3D korekce zapínána. Sekvence, která byla přidána do @start_program byla smazána a přidána na začátek procesu @rapid_move, ze kterého byla již generována data do operace, ve které byla zapínána 3D korekce. Při aktivování 3D korekce na špičku v SolidCAMu hodnoty všech tří proměnných byly rovny nule, jak ukazuje Obr. 63. Stejně hodnoty byly i v případě, když byla 3D nástrojová korekce vypnutá.

```
tool_tip = 0  
tool_center = 0  
type_offset_3D = 0
```

Obr. 63: Hodnoty proměnných při zapnuté 3D korekci na špičku

Následně byla aktivována 3D korekce na střed nástroje v SolidCAMu. Zde byla změněna jediná hodnota, což byla type_offset_3D je roven 1, jak je vidět na Obr. 64.

```
tool_tip = 0  
tool_center = 0  
type_offset_3D = 1
```

Obr. 64: Zapnutá korekce na střed

Zde bylo potřeba přijít na to, jakým způsobem tyto proměnné zakomponovat do podmínky tak, aby normálový vektor byl vypisován do NC programu. Při zkoušení proměnných byla jediná měněná proměnná `type_offset_3D`. Prvním pokusem tedy bylo do podmínky zakomponovat proměnnou `type_offset_3D` a podle toho nechat vypisovat normálový vektor. První test této podmínky je zobrazen na Obr. 65. V kódu GPP "eq" znamená rovná se. [15]

```
if type_offset_3D eq 0
  {nl, 'Korekce je zapnutá na špičku'}
elseif type_offset_3D eq 1
  {nl, 'Korekce je zapnutá na střed'}
else
  {nl, 'Vypnuto'}
endif
```

Obr. 65: Testování podmínky s `type_offset_3D`

Následně byl generován NC program, pro otestování této podmínky. Při zapnutí korekce na špičku nástroje bylo do NC programu správně vypsáno "Korekce zapnutá na špičku". Když byla nastavená korekce na špičku, bylo do NC programu správně vypsáno "Korekce zapnutá na střed". Avšak v případě, když byla 3D korekce vypnutá, bylo do NC programu vypsáno "Korekce zapnutá na špičku", místo "Vypnuto". Tato podmínka fungovala tímto způsobem, protože proměnná `type_offset_3D` nabývá stejné hodnoty (v tomto případě nula) jak při vypnuté korekci, tak při zapnuté korekci na špičku.

Byl přidán proces `@compensation_3d`, který je určen pouze pro 3D nástrojovou korekci. Do procesu `@compensation_3d` byla vytvořena upravená podmínka se zakomponovanými proměnnými `tool_tip` a `tool_center`. Tato podmínka je zobrazena na Obr. 66.

```
@compensation_3d

if type_offset_3D eq 0 ;pokud je type_offset_3D roven nule
  tool_tip = 1 ;je přiřazena tool_tip hodnota 1
elseif type_offset_3D eq 1 then ;pokud je type_offset_3D roven 1
  tool_center = 2 ;je přiřazena tool_center hodnota 2
endif

endp
```

Obr. 66: Přidání podmínky do procesu `@compensation_3d`

Následovalo další testování podmínky. Do procesu `@rapid_move` byla přidána podmínka, která je zobrazena na Obr. 67 pro otestování nastavených hodnot podmínkou v procesu `@compensation_3d` (Obr. 66).

```

@rapid_move
  local string S_Xpos S_Ypos S_Zpos S_line

  if tool_tip eq 1                                ;pokud je tool_tip roven 1
    {nl, 'Korekce zapnutá na špičku'}           ;vypíše se do programu: Korekce zapnutá na špičku
  elseif tool_center eq 2                          ;pokud je tool_tip roven 2
    {nl, 'Korekce zapnutá na střed'}            ;vypíše se do programu: Korekce zapnutá na střed
  else                                             ;v jiném případě
    {nl, '3D korekce je vypnutá'}              ;vypíše se do programu: 3D korekce je vypnutá
  endif

```

Obr. 67: Upravený GPP pro otestování podmínky

Při této podmínce jak při zapnutých obou možnostech 3D nástroje korekce, tak při jejím vypnutí byly komentáře vypisovány správně. Tato podmínka tedy fungovala pro možnosti, když je korekce zapnutá nebo vypnutá.

Poté se však objevil ještě jeden problém. V případě, že první operace byla bez 3D korekce a druhá s 3D korekcí bylo vše v pořádku. Avšak v případě, že první operace byla s 3D korekcí a druhá bez 3D korekce, tak u druhé operace, která měla korekci vypnutou byl generován NC program, jako kdyby byla 3D korekce zapnutá. Z tohoto důvodu bylo potřeba udělat testování hodnot proměnných při všech variantách. Všechny možnosti jsou vypsány v Tab. 5.

Tab. 5: Všechny možnosti zapnutí 3D korekce

Stav korekce	Hodnota type_offset_3D	Hodnota Tool_tip (TT) a Tool_center (TC)
Zapnutá korekce na špičku	0	TT = 1 TC = 0
Vypnutá korekce	2	TT = 1 TC = 0
Zapnutá korekce na střed	1	TT = 0 TC = 2
Vypnutá korekce	2	TT = 1 TC = 2
Vypnutá korekce	0	TT = 0 TC = 0
Zapnutá korekce na špičku	0	TT = 1 TC = 0
Vypnutá korekce	0	TT = 0 TC = 0
Zapnutá korekce na střed	1	TT = 0 TC = 2

Pro úpravě podmínky bylo zjištěno, při jakých možnostech nastává problémová situace. Nastávala pouze v případě, když nejprve byla zapnutá 3D korekce a poté vypnutá. U vypnuté korekce proměnná type_offset_3D nabývala hodnoty 2. Problémem bylo, že TT a TC nebylo nastavené na nulu, ale bylo kvůli předešlé zapnuté korekci změněno. Proto byla upravená podmínka pro hodnotu type_offset_3D rovné 2 tak, aby TC a TT byla nastavena na 0, protože podmínka byla uvažována pro situaci pokud TT a TC je nula, tak je korekce vypnutá. Upravená podmínka je zobrazena na Obr. 68.

```
@compensation_3d

if type_offset_3D eq 0           ;pokud je type_offset_3D roven nule
    tool_tip = 1                 ;je přiřazena tool_tip hodnota 1
elseif type_offset_3D eq 1 then ;pokud je type_offset_3D roven 1
    tool_center = 2             ;je přiřazena tool_center hodnota 2
elseif type_offset_3D eq 2 then ;pokud je type_offset_3D roven 2
    tool_tip = 0                 ;nastavení tool_tip na 0
    tool_center = 0              ;nastavení tool_center na 0
endif

endp
```

Obr. 68: Upravená podmínka v @compensation_3d

2.7.7 Přidání podmínky do @line

Posledním krokem bylo zakomponování podmínky (**Error! Reference source not found.**) do procesu @line. Nejprve bylo popsáno, co bude do podmínky zadáváno (Obr. 69).

```
if tool_tip eq 1                 ;pokud je zapnutá 3D korekce na špičku
    ;úpravná část pro 3D korekci na špičku nástroje
elseif tool_center eq 2 then     ;pokud je zapnutá 3D korekce na střed
    ;úpravná část pro 3D korekci na střed nástroje
else                             ;pokud není 3D korekce zapnutá
    ;původní kód bez 3D korekce
endif
```

Obr. 69: Podmínka pro @line

Jako první byla přidána část kódu do podmínky pro případ, když je tool_tip roven 1, tedy je zapnutá 3D korekce na špičku nástroje. Na Obr. 70 je zobrazena finální verze upraveného kódu s normálovým vektorem povrchu při 3D korekci na špičku pro pos_to_machine = Y. Pro pos_to_machine = N je kód skoro stejný. Jediným rozdílem je to, že místo proměnné xlpos, ylpos a zlpos je zde proměnná xpos, ypos a zpos. Dále pokračovala podmínka druhou částí, pro podmínku “elseif tool_center eq 2 then“ což je pro možnost, při které je zapnutá 3D korekce na střed nástroje. Pod touto podmínkou byl vložen stejný kód, jako pod podmínku v případě, že je

zapnutá korekce na špičku nástroje. Do poslední části podmínky pod else (případ, když korekce není zapnutá) byla vložena původní neupravená verze kódu. K tomu, aby kód na Obr. 70 mohl fungovat je nutné, aby součástí kódu byla podmínka přidána v procesu @compensation_3d (Obr. 68).

```
If tool_tip eq 1 ;pokud je zapnuta korekce na spicku
; rezim GPP pos_to_machine = Y
if L_pos_to_machine_Y == true and L_cycle_19 == false
if change(xlpos) == true
| S_Xpos = " " + S_I + "X" + tostr(xlpos:xpos_f) ;souřadnice X
else
| S_Xpos = ""
endif

if change(ylpos) == true
| S_Ypos = " " + S_I + "Y" + tostr(ylpos:ypos_f) ;souřadnice Y
else
| S_Ypos = ""
endif

if change(zlpos) == true
| S_Zpos = " " + S_I + "Z" + tostr(zlpos:zpos_f) ;souřadnice Z
| N_save_Z = zlpos
else
| S_Zpos = ""
endif

;přidaná část pro 3D korekci
if change(x_surface_normal) == true
| Xsurface_normal = " " + S_I + "NX" + tostr(x_surface_normal:'+3.7z') ;souřadnice NX
else
| Xsurface_normal = ""
endif

if change(y_surface_normal) == true
| Ysurface_normal = " " + S_I + "NY" + tostr(y_surface_normal:'+3.7z') ;souřadnice NY
else
| Ysurface_normal = ""
endif

if change(z_surface_normal) == true
| Zsurface_normal = " " + S_I + "NZ" + tostr(z_surface_normal:'+3.7z') ;souřadnice NZ
else
| Zsurface_normal = ""
endif
```

Obr. 70: Upravený kód při korekci na špičku

3 Test postprocesoru s 3D korekcí

Pro otestování 3D korekce bylo zvoleno pětiosé obráběcí centrum DMU 40 eVo linear DMG MORI, které je zobrazeno na Obr. 71, z důvodu přesného obrábění. V Tab. 6 jsou vypsány základní parametry stroje.



Obr. 71: pětiosé obráběcí centrum DMU eVo linear DMG MORI

Tab. 6: Parametry stroje DMU eVo linear DMG MORI [19]

Parametr	Hodnota
Maximální pojezd v ose X	400 mm
Maximální pojezd v ose Y	400 mm
Maximální pojezd v ose Z	375 mm
Maximální průměr obrobku	450 mm
Maximální výška obrobku	460 mm
Maximální hmotnost obrobku	250 kg
Maximální otáčky vřetene	20 000 ot/min
Systémy řízení a software	SIEMENS, HEIDENHAIN

3.1 Naměření kulové frézy

Před obráběním bylo potřeba naměřit kulovou frézou, kvůli 3D korekci. Nejlepším řešením bylo použití funkce 3D-ToolComp a naměření více bodů na nástroji v závislosti na úhlu. Toto však nebylo možné provést, protože stroj neměl k dispozici volitelný software 92, který obsahuje funkci 3D-ToolComp, protože není součástí základní výbavy Heidenhainu.

Do tabulky nástrojů byla zadána data pro automatické měření nástrojů. Bylo potřeba zadat hodnotu délky a hodnotu rádiusu. Pro měření délky je v tabulce určen sloupec TT:R-OFFS a pro měření rádiusu je sloupec TT:L-OFFS. Vzhledem k tomu, že byla měřena špička nástroje, byly zadány k hodnotě rádiusu nástroje TT:L-OFFS rozměr 5 mm, protože to je ideální velikost rádiusu frézy. Dále k hodnotě délky nástroje TT:R-OFFS byla zadána hodnota 0, protože měření bylo požadováno na špičce. Zadání do tabulky je zobrazeno na Obr. 72.

Soubor: TOOL.T		MM					
T	CUT.	LTOL	RTOL	DIRECT.	PLC	TT:L-OFFS	TT:R-OFFS
198	0	0	0	-	%00000000	+0	R
199	0	0	0	-	%00000010	+0	+0
200	0	0	0	-	%00000010	+0.5	+0.5
201	4	0	0	-	%00000000	+0	+0
202	0	0	0	-	%00000000	+0	R
203	0	0	0	-	%00000000	+0	+0
204	0	0	0	-	%00000000	+0	+0
205	0	0	0	-	%00000000	+0	+0
206	0	0	0	-	%00000000	+0	R
207	0	0	0	-	%00000000	+0	+0
208	0	0	0	-	%00000000	+0	+0
209	0	0	0	-	%00000000	+0	+0
210	0	0	0	-	%00000000	+5	+0
211	4	0	0	-	%00000000	+0	+0
212	4	0	0	-	%00000000	+0	+0
213	0	0	0	-	%00000000	+0	R
214	0	0	0	-	%00000000	+6	+0

Obr. 72: Zadání hodnot pro automatické měření do tabulky nástrojů

Poté bylo provedeno automatické měření nástroje laserovým systémem stroje. V tabulce nástrojů přibyla neměřená hodnota odchylky DR -0,0285 mm, jak je vidět na Obr. 73.

Soubor: TOOL.T		MM					
T	NAME	L	R	R2	DL	DR	
209	KUL_D8_Z4	+130.7611	+4	+4	+0	+0	
210	KUL_D10_Z4	+137.5773	+5	+5	+0	-0.0285	
211	KUL_D10_Z4_LONG	+152.34	+5	+5	+0	+0	
212	KUL_D8_Z4_LONG	+151.6351	+4	+4	+0	+0	
213		+0	+0	+0	+0	+0	
214	KUL_D12_Z4	+212.4282	+6	+6	+0	+0	
215		+0	+0	+0	+0	+0	
216		+0	+0	+0	+0	+0	
217	KUL_D16_Z2_ISCAR	+155.1543	+8	+8	+0	+0	
218		+0	+0	+0	+0	+0	

Obr. 73: Naměřená odchylka v tabulce nástrojů

3.2 Obrábění

Po úpravě postprocesoru pro 3D korekci a naměření kulové frézy, následovalo obrobení součásti pro otestování funkčnosti programu. Jako materiál obrobku byl zvolen hliník. V Tab. 7 je přehled zvolených řezných podmínek pro hrubování a dokončení.

Tab. 7: Přehled řezných podmínek

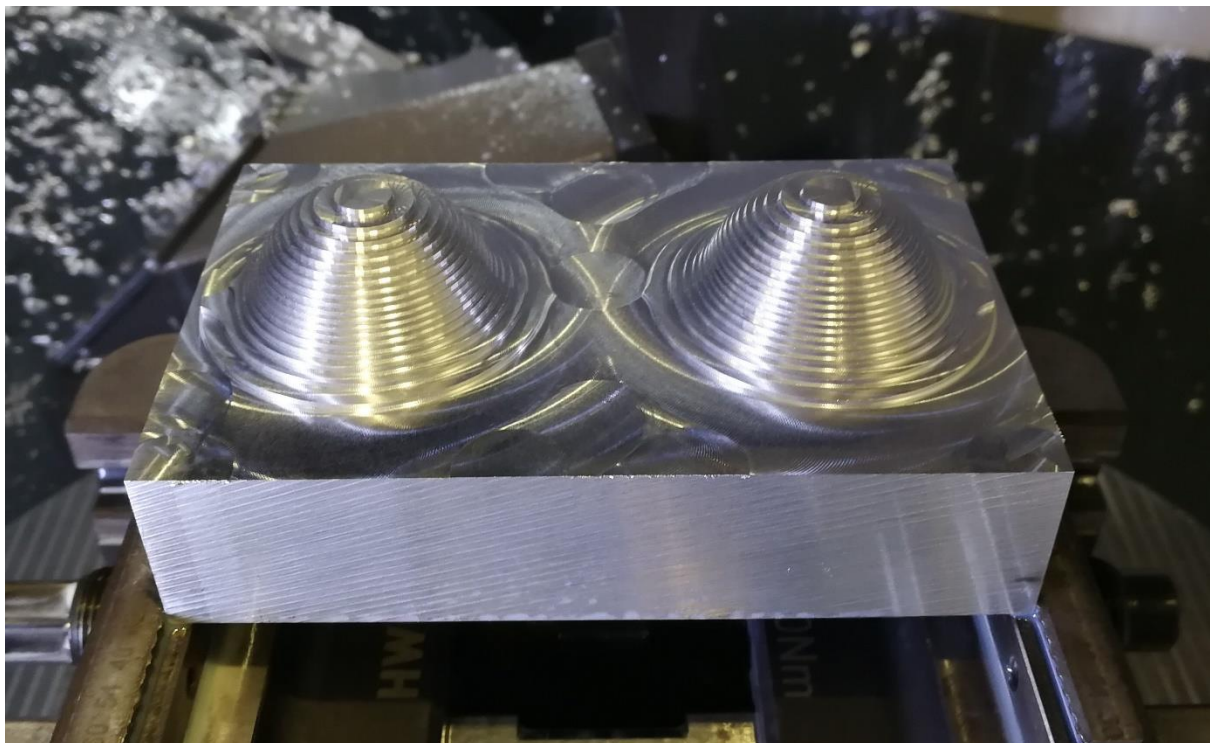
Typ operace	Otáčky [ot/min]	Řzná rychlost [m/min]	Nástroj	Průměr nástroje [mm]	Počet zubů	Hloubka řezu [mm]	Boční krok [mm]
Hrubování	14 000	700	Toroidní fréza	16	4	1	-
Dokončení	18 000	700	Kulová fréza	10	4	-	0,2

Program v SolidCAMu se skládá prakticky ze tří částí:

1. Hrubování
2. Dokončení pravé kulové plochy s aktivní 3D nástrojovou korekcí
3. Dokončení levé kulové plochy bez aktivní 3D nástrojové korekce

3.2.1 Hrubování

Jako první byla do stroje nahrána část programu pro hrubování. Simulace v řídicím systému Heidenhain proběhla v pořádku, takže bylo spuštěno hrubování. Hrubování proběhlo bez problémů. Obrobek po hrubování je zobrazen na Obr. 74.



Obr. 74: Obrobek po hrubování

3.2.2 Dokončení kulové plochy s 3D korekcí

Po hrubování následovalo dokončení kulových ploch. Levá kulová plocha měla být obrobena bez 3D korekce a pravá s 3D korekcí. Nejprve byl vygenerován program pro plochu s 3D korekcí na špičku nástroje. Program byl vygenerován v SolidCAMu a nahrán do stroje. Posuvy byly sníženy na minimální rychlost, aby se mohl stroj zastavit v případě, že by nastala kolize. Když nástroj najel na hodnotu 0 souřadnice Z, stroj byl zastaven a bylo zkontrolováno, jestli je nástroj ve správné výšce u horního bodu kulové plochy. Vzhledem k tomu, že byl ve správné výšce bylo spuštěno pokračování obrábění. Pohyby nástroje byly správné a obrábění proběhlo bez komplikací. Obrobená pravá kulová plocha se zapnutou 3D korekcí je zobrazena na Obr. 75.



Obr. 75: Obrobek po dokončení s 3D korekcí

3.2.3 Dokončení kulové plochy bez 3D korekce

Poté byl vygenerován program bez 3D korekce, nahrán do stroje a spuštěn. Obrábění proběhlo v pořádku stejně jako u programu s 3D korekcí. Výsledek po obrábění je vidět na Obr. 76.



Obr. 76: Obrobek po dokončení kulové plochy bez 3D korekce

3.2.4 Dokončení dna kulovou frézou

Po dokončení kulových ploch byl malý schod mezi dnem a koncem kulových ploch, z důvodu použití různých nástrojů na obrobení těchto ploch. Proto byla dokončeno dno kulovou frézou, (nyní bez 3D korekce), která byla použita pro dokončení kulových ploch. Obrobek s dokončeným dnem je zobrazen na Obr. 77.



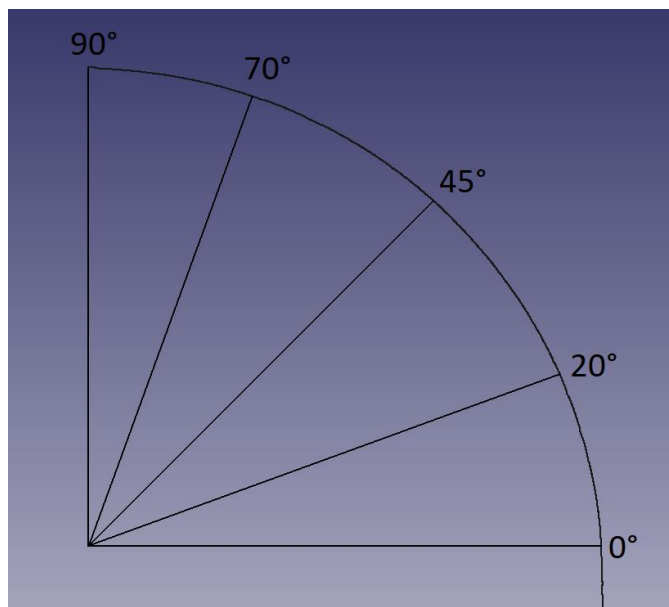
Obr. 77: Kompletně obrobená testovací součást

Po obrobení součásti následovalo měření a vyhodnocení testu, které je popsáno v závěrečné kapitole.

4 Vyhodnocení testu

4.1 Měření nástroje

Pro kontrolu výsledků byla kulová fréza naměřena na celém profilu. Toto měření proběhlo na optickém zařízení Zoller Genius 3. Výsledkem tohoto měření byla skutečná obálka měřeného nástroje. Pro vytvoření představy o tvaru nástroje byly zjištěny hodnoty odchylek pro jednotlivé úhly vynesemím přímek od středu kulové frézy, jak je znázorněno na Obr. 78. Toto měření však slouží pouze pro účely vyhodnocení testu, kvůli absenci doplňkového softwaru 3D-ToolComp není možné tato data pro 3D korekci na stroji použít.



Obr. 78: Změřená kulová fréza a vynesení přímek pro jednotlivé úhly

4.2 Měření obrobené součásti sondou

Po obrobení byl změřen vrchol obou polokoulí měřící sondou. Vrchol kulové plochy měl být obroben ve výšce -3 mm od nulového bodu. Výsledky měření jsou v Tab. 8.

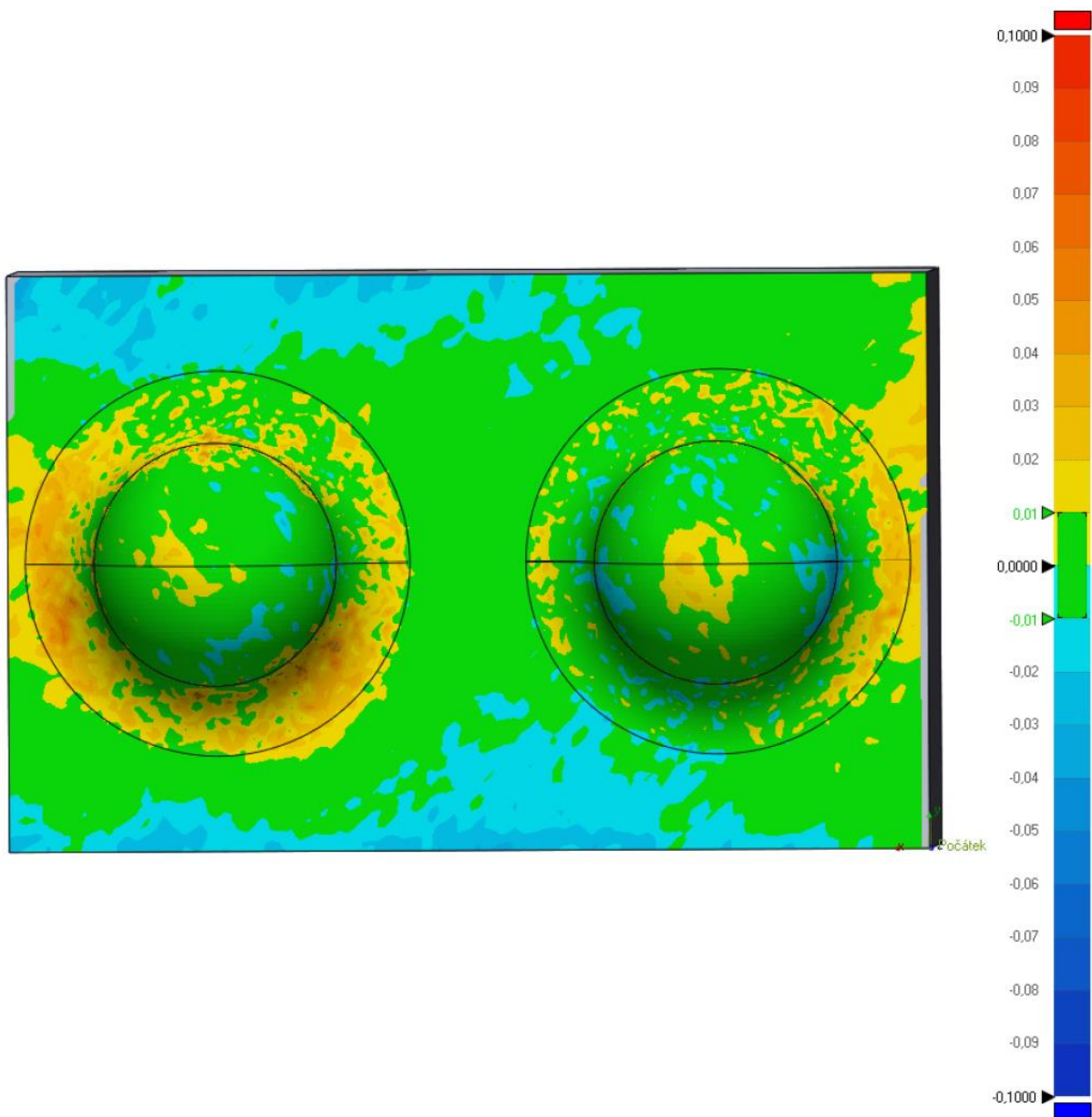
Tab. 8: Měření sondou vrcholu kulových ploch po obrábění

Stav	Výška vrcholu kulové plochy od nulového bodu roviny Z (mm)
Dle výkresu	-3.0000
Po obrobení s 3D korekcí	-3.0218
Po obrobení bez 3D korekce	-3.0097

Po obrobení kulové plochy bez 3D korekce byla výška vrcholu kulové plochy -3.0097 mm. Po obrobení kulové plochy s 3D korekcí byla naměřená výška vrcholu -3.0218 mm. Z toho plyne, že u vrcholu při korekci došlo k větší odchylce, než u vrcholu bez korekce. Avšak 3D korekce není určena pouze pro špičku nástroje, ale pro každý bod dotyku nástroje s obrobkem. Z tohoto důvodu nelze kompletně posoudit výsledek pouze na změřeném vrcholu. Bylo provedeno komplexnější měření, které je popsáno v následující kapitole.

4.3 Výsledky měření

Měření celé součásti bylo provedeno laserovým snímáním povrchu. Odchyšky od požadovaného rozměru byly zobrazeny dle barevné stupnice. Výsledek měření je zobrazen na Obr. 79. Podstatné je, že kulové plochy jsou u měření obráceně než při výrobě, tedy pravá polokoule představuje část bez 3D korekce a levá polokoule s 3D korekcí. Zelená barva představuje naměřené hodnoty pro toleranci $\pm 0,01$ mm. Modré odstíny představují vyšší záporné hodnoty než 0,01 mm od požadovaného rozměru. Barvy od žluté po červenou představují vyšší kladné hodnoty než 0,01 mm od požadovaného rozměru.



Obr. 79: Výsledky měření laserovým snímáním povrchu

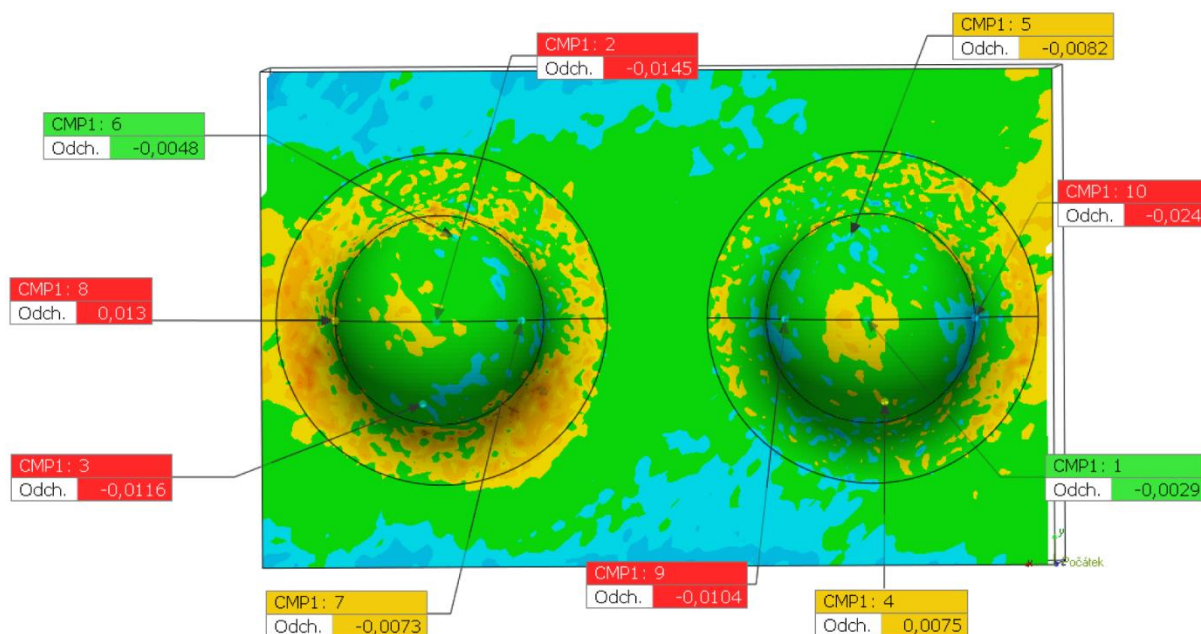
Na levé polokouli s 3D korekcí je vidět úbytek modrých podříznutých částí. Dále žlutá oblast u vrcholu polokoule se podstatně zmenšila. V dolní části, kde polokoule končí, se objevují místa zbarvená do oranžova, tedy místa větší než požadovaný rozměr. Na pravé polokouli, kde nebyla aplikována 3D korekce je patrné větší množství odchylek. Při celkovém porovnání obou polokoulí lze říci, že výsledky na levé polokouli s 3D korekcí jsou více konzistentní s celkově menším obsahem odchylek.

4.4 Analýza dle naměřených rozměrů

Na každé polokouli bylo změřeno pět bodů. Měřené kontrolní body na polokoulích jsou označeny jako CPM1 až CPM10. Tyto naměřené body CPM jsou vypsány v Tab. 9 a zároveň jsou zobrazeny na Obr. 80. V levé části tabulky jsou vypsány naměřené odchylky polokoule bez 3D korekce a v pravé části jsou vypsány odchylky s 3D korekcí. Každá řádka tabulky obsahuje naměřené body ve stejné poloze na polokouli. Rozmezí naměřených hodnot odchylek bez 3D korekce je od +0,0075 mm do -0,0240 mm. Rozmezí naměřených hodnot odchylek s 3D korekcí je od +0,0130 mm do -0,0145 mm. Největší změna rozměru je u vrcholu polokoule, který byl kompenzován z -0,0029 mm na -0,0145 mm. Při porovnání jednotlivých naměřených hodnot je patrné, že vrchol kulové plochy byl korekcí podříznut. Avšak při porovnání naměřených bodů na různých částí polokoulí bylo zjištěno, že při aktivní 3D korekci došlo ke zlepšení odchylek od ideálního rozměru, než bez aktivní 3D korekce. Z toho vyplývá, že zásluhou 3D korekce byl zachován tvar polokoule, kdežto u polokoule bez 3D korekce celkový tvar se méně blíží tvaru koule, než s aktivní 3D korekcí. Z testu tedy vyplývá, že 3D korekce není užitečná jenom pro kompenzování nepřesnosti nástroje, ale je užitečná pro zachování tvarové přesnosti.

Tab. 9: Naměřené hodnoty na obou polokoulích

CMP1 bod bez 3D korekce	Odchylka bez 3D korekce [mm]	CMP1 bod s 3D korekcí	Odchylka s 3D korekcí [mm]
1	-0,0029	2	-0,0145
4	+0,0075	3	-0,0116
5	-0,0082	6	-0,0048
9	-0,0104	7	-0,0073
10	-0,0240	8	+0,0130



Obr. 80: Naměřené body CPM

Z měření je patrné, že je zde rozdíl mezi polokoulí bez 3D korekce a s 3D korekcí. Tímto bylo prokázáno, že systém na základě toho, kde má normálový vektor povrchu posune nástroj v tomto směru o hodnotu korekce. Při 3D korekci se tedy nejedná pouze o posunutí poloměru

nástroje, ale jedná se o korekci odchylky celého nástroje, která kompenzuje chyby ve 3D prostoru, což by za použití 2D korekce nebylo možné uskutečnit. Díky normálovému vektoru povrchu má stroj informaci o tom, jakým směrem má korekci v daném bodě dotyku provést.

Dále bylo zjištěno, že plocha polokoule s 3D korekcí byla posunuta, avšak naměřené odchylky jsou rovnoměrnější. 3D korekce vedla ke zlepšení odchylek o ideálního rozměru a kulovitosti. Z porovnání lze odvodit, že 3D korekce nástroje měla pozitivní efekt na výrobu. Tato analýza potvrzuje, že i přes určitá omezení je 3D korekce způsobem ke zlepšení přesnosti a kvality obrábění.

5 Závěr

Tato diplomová práce se zabývá problematikou zprovoznění 3D korekce nástroje. Cílem této diplomové práce bylo upravit postprocesor pro vybraný stroj tak, aby byla zprovozněna 3D korekce nástroje.

První část práce je zaměřena na současný stav 3D korekce nástroje. Nejprve bylo popsána to, jakým způsobem funguje a může být použita klasická 2D korekce a následně byla popsána 3D korekce nástroje. Důležitou informací je, že k tomu, aby 3D korekce mohla fungovat už není potřeba pouze informace o tom, jestli má být provedena zprava nebo zleva. Je zde zásadní normálový vektor pro každý bod dotyku nástroje s obrobkem, bez kterého 3D korekce nemůže fungovat, protože stroj potřebuje vědět, jakým směrem má 3D korekci provést v daném bodě. Na základě schématu procesu výroby součásti na CNC stroji, bylo zjištěno, jaké kroky bylo potřeba udělat k aktivaci 3D korekce. Jsou zde popsány všechny důležité oblasti související s aplikováním 3D korekce. Nejprve je popsán systém CAM, konkrétně zaměření na SolidCAM a které procesy jsou pro tento systém z hlediska 3D korekce podstatné. Dále je popsán postup zapnutí 3D korekce v SolidCAMu. Následuje popsání základních procesů postprocesoru a poté konkrétní proměnné, které jsou stěžejní pro úpravu postprocesoru. Následovalo detailní popsání zadávání trojrozměrné korekce v řídicím systému Heidenhainu TNC 530.

Druhá část práce byla zaměřena na úpravu postprocesoru pro vybraný stroj. Nejprve byl navrhnout a vymodelován model součásti pro otestování 3D korekce a následně obrobení této součásti s SolidCAMu. Před samotnou úpravou postprocesoru, bylo důležité si připravit Visual Studio Code pro efektní úpravu kódu. Byl analyzován kód postprocesoru pro vybraný stroj DMU 40 eVo linear DMG MORI. Kód byl mnohokrát upravován a testován pro docílení generování NC programu s normálovým vektorem povrchu. V práci je detailně krok za krokem popsáno, co vše muselo být provedeno pro úpravu postprocesoru tak, aby generoval NC program s 3D korekcí, přesně tak, jak požaduje řídicí systém Heidenhain.

Třetí část práce byla zaměřena na samotný test upraveného postprocesoru a vytvořeného programu pro výrobu. Pro výrobu bylo použito pětiosé obráběcí centrum DMU 40 eVo linear DMG MORI. Obrábění proběhlo bez komplikací včetně obrobení polokoule s LN bloky kódu, které obsahovaly normálový vektor pro 3D korekci.

Poslední část práce je zaměřena na vyhodnocení změřené součásti po obrobení. Z měření je patrné, že aktivováním 3D korekce nástroje nastala změna, oproti obrobení bez 3D korekce, čímž bylo prokázáno, že přidání normálové vektoru mělo vliv na výrobu. Výsledky na polokouli s 3D korekcí jsou více konzistentní s celkově menším obsahem odchylek. Vrchol s 3D korekcí byl podříznut více než bez aktivní 3D korekce, avšak při porovnání naměřených bodů na různých částí polokoulí, bylo zjištěno, že při aktivní 3D korekci došlo ke zlepšení odchylek od ideálního rozměru. Po analýze bylo zjištěno, že došlo k celkovému posunutí polokoule s 3D korekcí. Dále u polokoule s 3D korekcí došlo ke zlepšení odchylek, které byly celkově rovnoměrnější, což znamená že byla zachována kulovitost. Z testu vyplývá, že 3D korekce není užitečná jenom pro kompenzování nepřesností nástroje, ale je užitečná pro zachování tvarové přesnosti. Při 3D korekci se tedy nejedná pouze o posunutí poloměru nástroje, ale jedná se o korekci odchylky celého nástroje, která kompenzuje chyby ve 3D prostoru, což by za použití 2D korekce nebylo možné uskutečnit. 3D korekce nástroje tedy měla pozitivní vliv na obrábění.

Seznam použitých zdrojů

- [1] CZÁN, PHD, prof. Ing. Andrej, doc. Ing. Robert ČEP, PH.D, Ing. Jiří KRATOCHVÍL, PH.D., Ing. et Ing. Mgr. Jana PETRŮ, PH.D, doc. Ing. Marek SADÍLEK, PH.D., doc. Ing. Dana STANČEKOVÁ, PHD a Ing. Tomáš ZLÁMAL. *Progresivní technologie v obrábění a NC programování obráběcích strojů* [pdf]. VŠB-TU Ostrava, 2014.
- [2] XU, Ai fen a Tai yong WANG. *Study on the Technology of 3D Cutter Radius Compensation Based on Direction Vector*. Xi'an, 2010, 5 s. Vědecký článek. International Conference on Mechatronics and Automation.
- [3] HAN, Li, Xiao-Shan GAO a Hongbo LI. *Space cutter radius compensation method for free form surface end milling* [online]. Londýn, 2012 [cit. 2024-04-10]. Dostupné z: <https://link.springer.com/article/10.1007/s00170-012-4674-2>
- [4] *Poloměrové korekce, optimalizace ekvidistanty pro pálicí stroje: 7. KOREKCE NÁSTROJE* [pdf]. MEFI, 2006. Dostupné také z: https://www.mefi.cz/files/dokumentace/cnc8x9/prilohynavodu/vyber/kor_palici.pdf
- [5] Akademie CNC obrábění (8). *Technickytydenik.cz* [online]. 2009 [cit. 2024-04-10]. Dostupné z: https://www.technickytydenik.cz/rubriky/serialy/akademie-cnc/akademie-cnc-obrabeni-8_8543.html
- [6] KORDINA, Michal. *Využití 3D korekcí nástroje při obrábění tvarových ploch*. Plzeň, 2011. diplomová práce (Ing.). ZÁPADOČESKÁ UNIVERZITA V PLZNI. Fakulta strojní
- [7] 3-D ToolComp: work with the actual cutter radius. *Klartext Portal* [online]. [cit. 2024-04-16]. Dostupné z: <https://www.klartext-portal.com/en/tips/functions/3d-toolcomp/>
- [8] WANG, Feng, Hu LIN, Liaomo ZHENG, Feng LIU a Lei YANG. A control algorithm for rapid movements near the radius compensation singularity in five-axis end milling. *Journal of Systems Science and Complexity*. 2013, 2013(26), 17. ISSN 26: 718–734.
- [9] *SolidCAM 2021 HSS Machining User Guide* [online]. SolidCAM. 2021, 180 s. [cit. 2024-03-21].
- [10] MACHALA, Zdeněk. *Postprocesory v SolidCAM*. Praha, 2017. Bakalářská práce. České vysoké učení technické v Praze.
- [11] *Příručka uživatele Popisný dialog HEIDENHAIN iTNC 530* [online]. 2. Německo: Heidenhain, 2013, 735 s. [cit. 2024-03-31]. 737759-C2.
- [12] MUDRA, Josef. *Posouzení vlivů na jakost a produktivitu při obrábění na CNC stroji*. Praha, 2021. Bakalářská práce. České vysoké učení technické v Praze fakulta strojní.
- [13] Geometrie kulových fréz dokonale pod kontrolou. *MM Průmyslové spektrum* [online]. 2012 [cit. 2024-04-11]. Dostupné z: <https://www.mmspektrum.com/clanek/geometrie-kulovych-frez-dokonale-pod-kontrolou>
- [14] *SolidCAM 2021 Simultaneous 5-Axis Machining User Guide* [online]. SolidCAM, 2021, 493 s. [cit. 2024-03-30]. SC5AUG0121SP1.
- [15] SolidCAM GPPTool. *SolidCAM Help* [online]. 2021 [cit. 2024-03-07]. Dostupné z: https://solidcam.help/2021/gpp_tool/SolidCAMGPPTool.htm
- [16] SolidCAM GPPTool User Guide ©1995-2009 SolidCAM LTD. All Rights reserved
- [17] Release Version. *Box* [online]. 2019 [cit. 2024-04-15]. Dostupné z: <https://solidcam.app.box.com/s/18vex20fzh7fu8eomvzgmhvx80dmiri>

[18] MILTON, Joe. Product Design Portfolio. *Blogs brighton* [online]. [cit. 2024-04-01]. Dostupné z: <https://blogs.brighton.ac.uk/xe404joemilton/week-8-computer-controlled-milling/>

[19] DMU 40 (FD) eVo. *DMG MORI* [online]. 2024 [cit. 2024-04-30]. Dostupné z: <https://cz.dmgmori.com/produkty/stroje/frezovani/petiose-frezovani/evo/dmu-40-fd-evo>

[20] My text file is riddled with question marks. How can I make it readable? *Superuser* [online]. 2022 [cit. 2024-05-21]. Dostupné z: <https://superuser.com/questions/1744157/my-text-file-is-riddled-with-question-marks-how-can-i-make-it-readable>

[21] How to setup Visual Studio Code to detect and set the correct encoding on file open. *Stackoverflow* [online]. 2016. Dostupné také z: <https://stackoverflow.com/questions/38528384/how-to-setup-visual-studio-code-to-detect-and-set-the-correct-encoding-on-file-o>

PŘÍLOHA č. 1

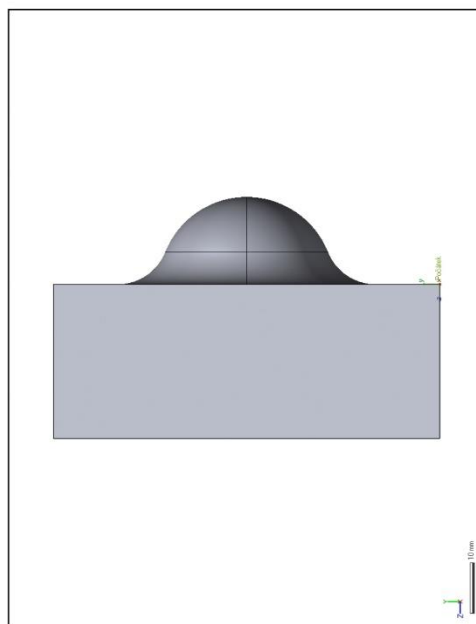
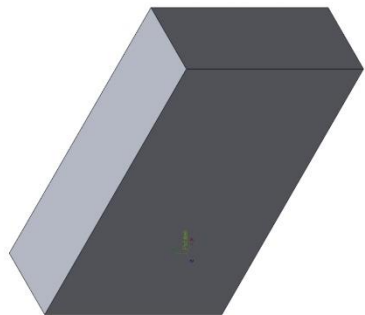
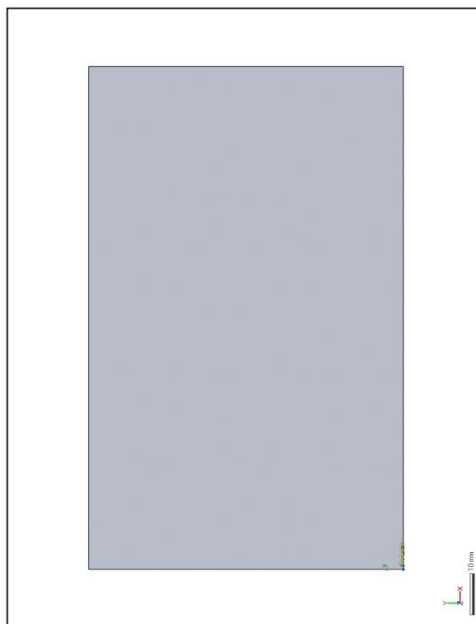
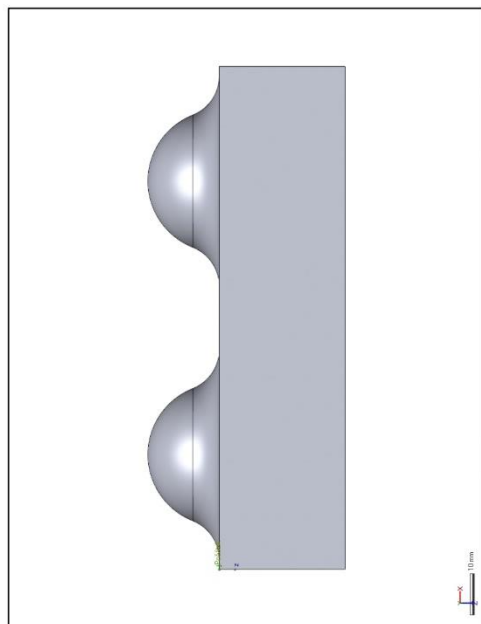
Měření součásti laserovým snímáním povrchu



Product Name	bábovka
Part Name	bábovka
Part Number	díl 1
Department	LDM
Inspector	Dana Kubátová
Date	May 16, 2024
Unit	mm

Disclaimer
The results of this analysis and forecasts are believed to be reliable but are not to be construed as providing a warranty, including any warranty of merchantability or fitness for purpose, or representation for which 3D Systems, Inc. assumes legal responsibility. Users should undertake sufficient verification and iterative testing to determine the suitability of any information presented. Nothing herein is to be taken as permission, inducement or recommendation by 3D Systems, Inc. to practice any patented invention without a license or to in any way infringe upon the intellectual property rights of any other party.

Výsledná data - 1 : Referenční Data - Zrcadlit1

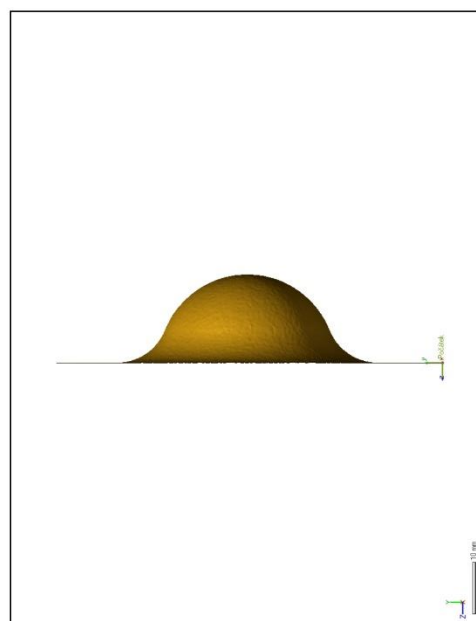
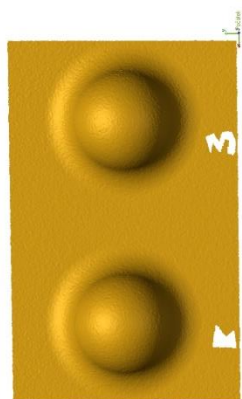
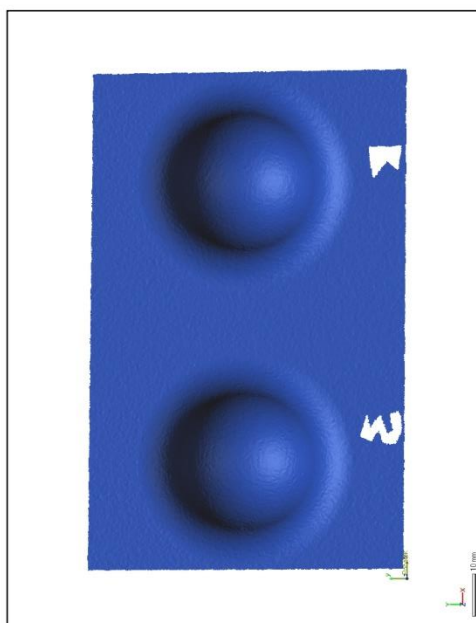
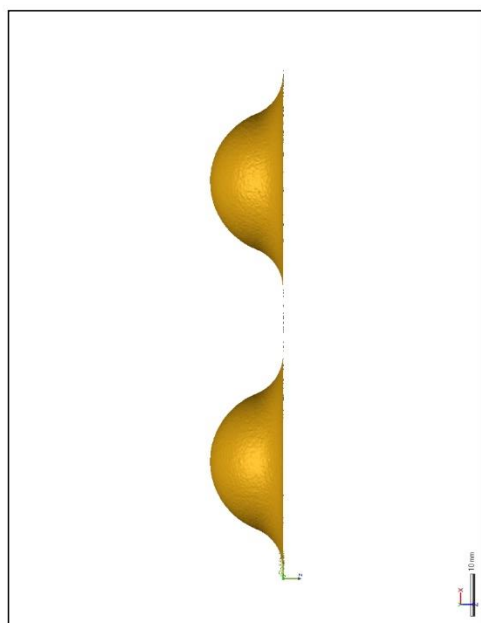


Product Name	bábovka
Part Name	bábovka

Department	LDM
Inspector	Dana Kubátová

Date	May 16, 2024
Unit	mm

Výsledná data - 1 : Naměřená data - Scan 2

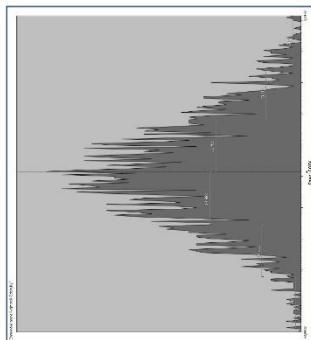
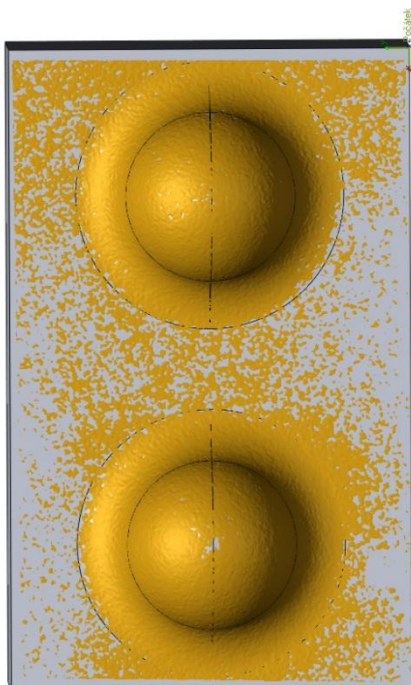


Product Name	bábovka
Part Name	bábovka

Department	LDM
Inspector	Dana Kúbátová

Date	May 16, 2024
Unit	mm

Výsledná data - 1 : Zarovnání podle Nejmenší Odchylky1



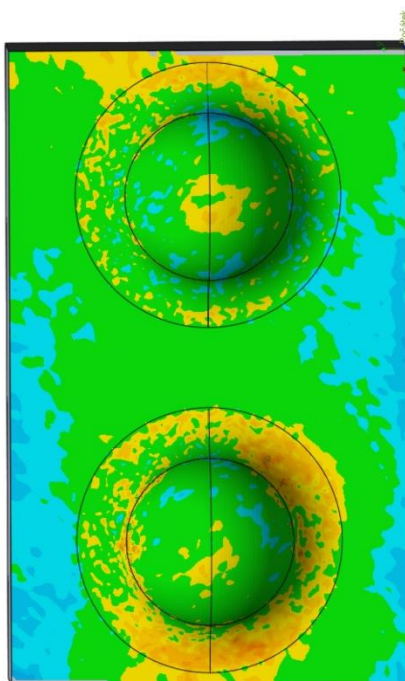
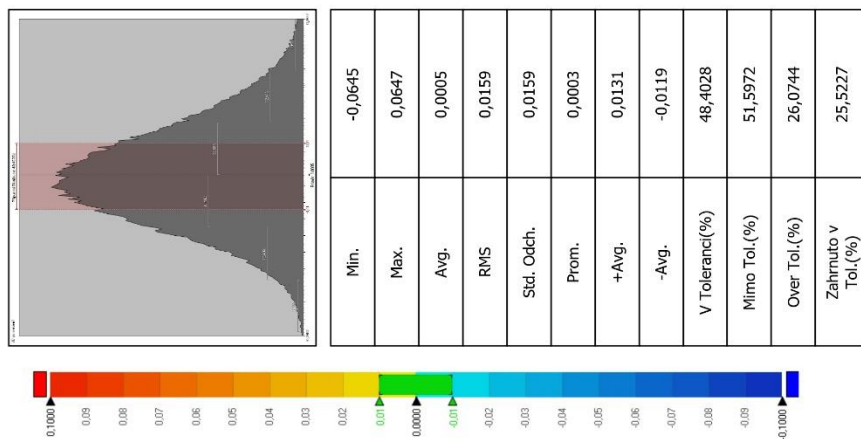
Min.	-0,0665
Max.	0,0565
Avg.	0,0004
RMS	0,0163
Std. Odch.	0,0163
Prom.	0,0003
+Avg.	0,0133
-Avg.	-0,0125

Date	May 16, 2024
Unit	mm

Department	LDM
Inspector	Dana Kubátová

Product Name	bábovka
Part Name	bábovka

Výsledná data - 1 : 3D porovnání1



Date	May 16, 2024
Unit	mm

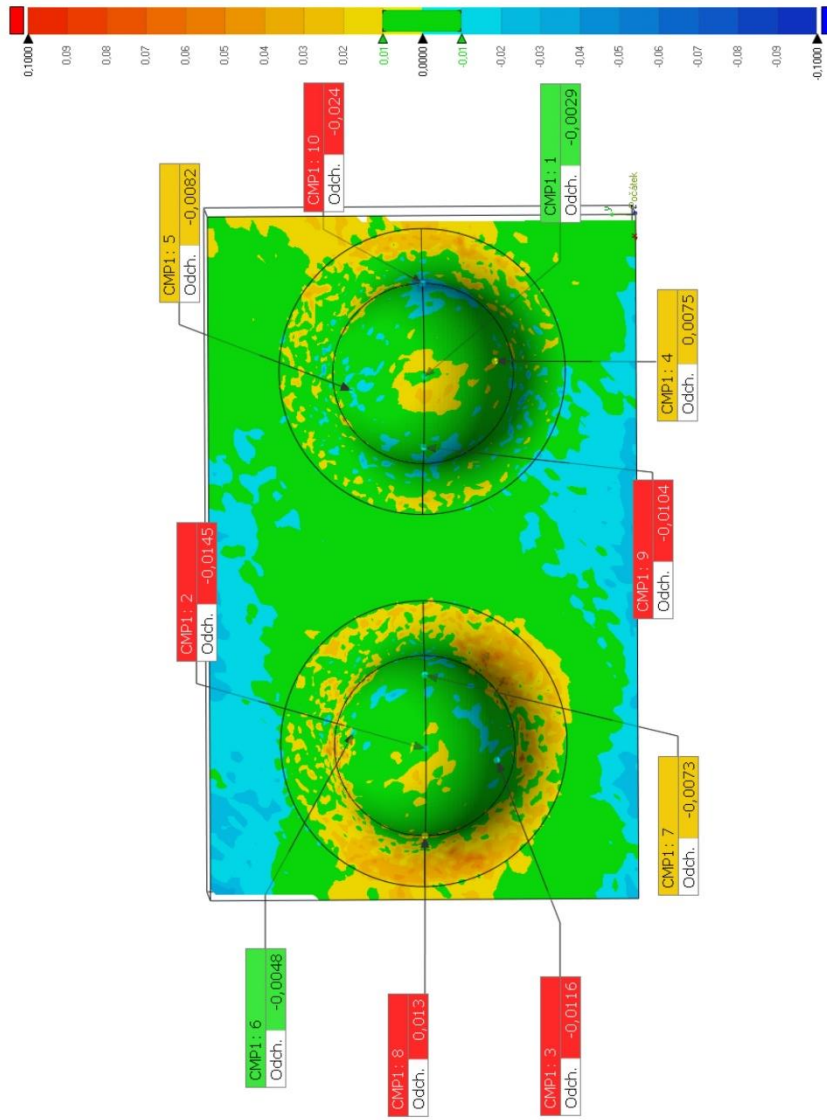
Department	LDM
Inspector	Dana Kubátová

Product Name	bábovka
Part Name	bábovka

Název	Min.	Max.	Avg.	RMS	Std. Odch.	Prom.	+Avg.	-Avg.
3D porovnání1	-0,0645	0,0647	0,0005	0,0159	0,0159	0,0003	0,0131	-0,0119

Product Name	bábovka	Department	LDM	Date	May 16, 2024
Part Name	bábovka	Inspector	Dana Kubátová	Unit	mm

Výsledná data - 1 : CMP1



Date	May 16, 2024
Unit	mm

Department	LDM
Inspector	Dana Kubátová

Product Name	bábovka
Part Name	bábovka

Název	Min.	Max.	Avg.	RMS	Std. Odch.	Prom.	+Avg.	-Avg.
CMP1	-0,1583	0,0404	-0,0063	0,0207	0,0197	0,0004	0,0095	-0,0156
Název	Název výsledku	Tol.	Odch.	Referenční poz.			Měř. Poz.	
				X	Y	Z	X	Y
CMP1: 1	Výsledná data - 1	±0,01	-0,0029	27,3564	37,5	-17	27,3565	37,4999
CMP1: 2	Výsledná data - 1	±0,01	-0,0145	92,6436	37,5	-17	92,6437	37,5003
CMP1: 3	Výsledná data - 1	±0,01	-0,0116	94,9994	25,0038	-11,2749	94,9979	25,0124
CMP1: 4	Výsledná data - 1	±0,01	0,0075	25,0006	25,0038	-11,2749	24,9997	24,9982
CMP1: 5	Výsledná data - 1	±0,01	-0,0082	29,9991	49,996	-11,2109	29,9978	49,9899
CMP1: 6	Výsledná data - 1	±0,01	-0,0048	90,0009	49,996	-11,2109	90,0016	49,9924
CMP1: 7	Výsledná data - 1	±0,01	-0,0073	80,0014	37,5	-11,3627	80,0068	37,5002
CMP1: 8	Výsledná data - 1	±0,01	0,013	108,4346	37,5	-6,2963	108,4467	37,4997
CMP1: 9	Výsledná data - 1	±0,01	-0,0104	39,9986	37,5	-11,3627	39,991	37,4998
CMP1: 10	Výsledná data - 1	±0,01	-0,024	11,5654	37,5	-6,2963	11,5876	37,4999

Date	May 16, 2024
Unit	mm

LDM	Dana Kúbatová
Department	Inspector

Product Name	bábovka
Part Name	bábovka