ZÁPADOČESKÁ UNIVERZITA V PLZNI

FAKULTA STROJNÍ

Studijní program: **B0715P270003 – Strojírenství**

Studijní specializace: **Programování NC strojů**

BAKALÁŘSKÁ PRÁCE

Návrh a optimalizace parametrického programu pro frézování typového výrobku v různých řídících systémech

Autor: Jakub BELŠÁN

Vedoucí práce: Ing. Luboš KROFT, Ph.D.

Akademický rok 2021/2022

ZÁPADOČESKÁ UNIVERZITA V PLZNI
Fakulta strojní
Akademický rok: 2021/2022

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení:	**Jakub BELŠÁN**
Osobní číslo:	**S19B0047P**
Studijní program:	**B0715P270003 Strojírenství**
Specializace:	**Programování NC strojů**
Téma práce:	**Návrh a optimalizace parametrického programu pro frézování typového výrobku v různých řídicích systémech**
Zadávající katedra:	**Katedra technologie obrábění**

Zásady pro vypracování

1. Úvod do problematiky a cíle řešení
2. Analýza současného stavu parametrického programování
3. Porovnání parametrického programování pro zadané řídicí systémy
4. Návrh, tvorba a odlazení parametrického programu pro zadanou součást
5. Technické hodnocení
6. Zhodnocení a závěr

Rozsah bakalářské práce: **30 – 40 stran**
Rozsah grafických prací: **dle potřeby**
Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam doporučené literatury:

- Elektronické informační zdroje dostupné z www.knihovna.zcu.cz.
- CIRP ANNALS – Manufacturing Technology: (http://www.cirp.net/)
- SANDVIK Coromant, Sandviken: Technická příručka obrábění, dostupné z: http://www.sandvik.coromant.com/cs-cz/
- Elektronické informační zdroje

Vedoucí bakalářské práce: **Ing. Luboš Kroft, Ph.D.**
Katedra technologie obrábění

Konzultant bakalářské práce: **Ing. Ondřej Maršálek**
Regionální technologický institut

Datum zadání bakalářské práce: **15. října 2021**
Termín odevzdání bakalářské práce: **27. května 2022**

L.S.

Doc. Ing. Milan Edl, Ph.D.
děkan

Doc. Ing. Jan Řehoř, Ph.D.
vedoucí katedry

Prohlášení o autorství

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

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

V Plzni dne: ……. …….

podpis autora

Poděkování:

Tímto bych rád poděkoval panu Ing. Luboši Kroftovi, Ph.D. za cenné rady, připomínky, aktivní zájem, veškerou pomoc a trpělivost při vypracovávání bakalářské práce. Dále děkuji za spolupráci kolektivu pracovníků z firmy STREICHER, spol. s.r.o. Plzeň, jmenovitě pak Václavu Zvěřinovi za podání podnětu na vypracování bakalářské práce.

ANOTAČNÍ LIST BAKALÁŘSKÉ PRÁCE

AUTOR	**Příjmení** Belšán	**Jméno** Jakub
STUDIJNÍ PROGRAM	B0715P270003 – Strojírenství	
VEDOUCÍ PRÁCE	**Příjmení (včetně titulů)** Ing. Kroft, Ph.D.	**Jméno** Luboš
PRACOVIŠTĚ	ZČU - FST – KTO	

DRUH PRÁCE	~~**DIPLOMOVÁ**~~	**BAKALÁŘSKÁ**	**Nehodící se škrtněte**
NÁZEV PRÁCE	Návrh a optimalizace parametrického programu pro frézování typového výrobku v různých řídicích systémech		

FAKULTA	strojní		**KATEDRA**	KTO	**ROK ODEVZD.**	2022

POČET STRAN (A4 a ekvivalentů A4)

CELKEM	57		**TEXTOVÁ ČÁST**	38	**GRAFICKÁ ČÁST**	12

STRUČNÝ POPIS (MAX 10 ŘÁDEK) **ZAMĚŘENÍ, TÉMA, CÍL POZNATKY A PŘÍNOSY**	Hlavním cílem bakalářské práce byla tvorba parametrického porgramu pro zadanou typovou součást od firmy STREICHER, spol. s r.o. Plzeň. Po navržení různých verzí programu, jsou jednotlivé varianty porovnány a je vybrán nejvýhodnější program pro firmu Streicher. Výstupem práce je optimalizovaný parametrický program vytvořen pro co největší unevarálnost použití.
KLÍČOVÁ SLOVA **ZPRAVIDLA JEDNOSLOVNÉ POJMY, KTERÉ VYSTIHUJÍ PODSTATU PRÁCE**	Parametrické programování, cykly, logické funkce, podprogramy Heidenhain, Sinumerik

SUMMARY OF BACHELOR SHEET

AUTHOR	**Surname** Belšán	**Name** Jakub	
STUDY PROGRAMME	B0715P270003 – Engineering		
SUPERVISOR	**Surname (Inclusive of Degrees)** Ing. Kroft, Ph.D.	**Name** Luboš	
INSTITUTION	ZČU - FST - KTO		
TYPE OF WORK	~~**DIPLOMA**~~	**BACHELOR**	**Delete when not applicable**
TITLE OF THE WORK	Design and optimization of a parametric program for milling a type product in different control systems		

FACULTY	Mechanical Engineering	**DEPARTMENT**	KTO	**SUBMITTED IN**	2022

NUMBER OF PAGES (A4 and eq. A4)

TOTALLY	57	**TEXT PART**	38	**GRAPHICAL PART**	12

BRIEF DESCRIPTION **TOPIC, GOAL, RESULTS AND CONTRIBUTIONS**	The main goal of the bachelor's thesis was to create a parametric program for a given type part from the company STREICHER, spol. s r.o. Pilsen. After designing different versions of the program, the individual variants are compared and the most advantageous program for Streicher is selected. The output of the work is an optimized parametric program created for the greatest possible usability.
KEY WORDS	Parametric programming, cycles, logical functions, subroutines, Heidenhain, Sinumerik

Obsah

Seznam příloh 2
Seznam obrázků 3
Seznam tabulek 3
1. Úvod 5
1.1 Cíl práce 5
2. Analýza současného stavu NC programování 6
2.1 Definice NC programování a výhody parametrického programování 6
2.1.1 Struktura NC programu 6
2.1.2 ISO kód 7
2.1.3 Dílenské programování - dialog 7
2.1.4 CAD/CAM programování 7
2.2 Parametrické programování 8
2.2.1 Výhody parametrizace při programování 8
2.2.2 Rozdělení parametrů 9
2.3 Využití vyššího programovacího jazyka 11
2.3.1 Definování parametrů 11
2.3.2 Funkce a výpočty 11
2.3.3 Logické funkce 11
2.3.4 Tvorba a využívání předdefinovaných cyklů 14
3. Porovnání parametrického programování pro zadané řídicí systémy 15
3.1 Řídicí systém Sinumerik 840 15
3.1.1 Tvorba parametrického programu 15
3.2 Řídicí systém Heidenhain iTNC530 16
3.2.1 Tvorba parametrického programu 16
3.3 Porovnání programování s parametry a bez nich 18
4. Analýza současného stavu ve firmě Streicher 20
4.1 Úvod do problematiky 20
4.2 Materiál součásti 21
4.2.1 Konstrukční ocel 11 523 21
4.2.2 Nerezová austenitická chrom-niklová ocel 1.4571 21
4.3 Návrh technologie frézování součásti 21
4.4 Nástroj pro obrábění 22
4.5 Polotovar 23

4.6 Analýza současného stavu 23
5. Návrh, tvorba a odladění optimalizovaného programu pro zadanou součást 27
5.1 Zjednodušení stávajícího programu 27
5.2 Návrh programu pomocí kruhové interpolace 31
5.3 Tvorba programu s nulovým bodem na horní ploše 34
5.4 Shrnutí 36
5.5 Zhodnocení variant a výběr optimální varianty 36
6. Závěr 37
Seznam použitých zdrojů 38

Seznam příloh

Příloha č.1 Původní program používaný ve firmě STREICHER, spol. s r.o. Plzeň……P1
Příloha č.1 Optimalizovaný parametrický program s nulovým bodem na spodní ploše polotovaru v řídícím systému Sinumerik……...P3
Příloha č.2 Optimalizovaný parametrický program s nulovým bodem na spodní ploše polotovaru v řídícím systému Heidenhain……P5
Příloha č.3 Optimalizovaný parametrický program s nulovým bodem na horní ploše polotovaru v řídícím systému Sinumerik……..……P7
Příloha č.4 Optimalizovaný parametrický program s nulovým bodem na horní ploše polotovaru v řídícím systému Heidenhain……...……..P9
Príloha č.5 Parametrický program pro obrábění po spirále ve dvou osách v řídícím systému Heidenhain……..…P11

Seznam obrázků

Obr. 1 – příruba v sestavě ... 5
Obr. 2 – výkres příruby ... 5
Obr. 3 – syntaxe bloku [6] ... 7
Obr. 4 – příklad parametrizace součásti ... 8
Obr. 5 – příklad – nepodmíněný skok [4] ... 12
Obr. 6 – příklad – podmíněný skok [4] ... 12
Obr. 7 – příklad – nekonečná smyčka [4] ... 13
Obr. 8 – čítací smyčka (opakování od 1 do 15) [4] ... 13
Obr. 9 – příklad – funkce CASE [4] ... 13
Obr. 10 – tabulka parametrů v Sinumeriku ... 15
Obr. 11 – program bez použití parametrů ... 18
Obr. 12 – program s využitím parametrů ... 18
Obr. 13 – příruba s rádiusem ... 20
Obr. 14 – polotovar příruby ... 21
Obr. 15 – ukázka obrábění pomocí spirály ... 22
Obr. 16 – ukázka obrábění drážkováním ... 22
Obr. 17 – hlavička programu ... 24
Obr. 18 – volba nástroje a řezných podmínek ... 24
Obr. 19 – definice parametrů ... 24
Obr. 20 – volání podprogramu a konec hlavního programu ... 24
Obr. 21 – začátek podprogramu a posunutí nulového bodu ... 25
Obr. 22 – okrajové podmínky obrábění ... 25
Obr. 23 – ukázka opakujících se bloků ... 25
Obr. 24 – vnořený podprogram LBL122 ... 25
Obr. 25 – dráhová funkce zadána parametricky a logická funkce tvořící smyčku ... 26
Obr. 26 – náčrt výpočtu souřadnic ... 26
Obr. 27 – zjednodušená hlavička programu ... 27
Obr. 28 – podprogram s cyklem pro frézování kruhové kapsy ... 28
Obr. 29 – posunutí nulového bodu a začátak podprogramu LBL1 ... 28
Obr. 30 – parametrický program s ukázkou drah za simulace ... 30
Obr. 31 – definice parametrů v Sinumeriku ... 30
Obr. 32 – hlavička programu s definicí parametrů ... 31
Obr. 33 – volba nástroje a volání podprogramu ... 31
Obr. 34 – program s kruhovou interpolací v Sinumeriku s ukázkou drah za simulace ... 32

Obr. 35 – podprogram LBL1 a LBL2 32
Obr. 36 – ukázka výpočtů v programu 33
Obr. 37 – podprogram LABEL2 33
Obr. 38 – pozice nulového bodu na spodní ploše 34
Obr. 39 – pozice nulového bodu na horní ploše 34
Obr. 40 – výpočet X-ové a Y-ové souřadnice 34
Obr. 41 – výpočet Z-ové souřadnice 35
Obr. 42 – výpočet parametru Q43 35

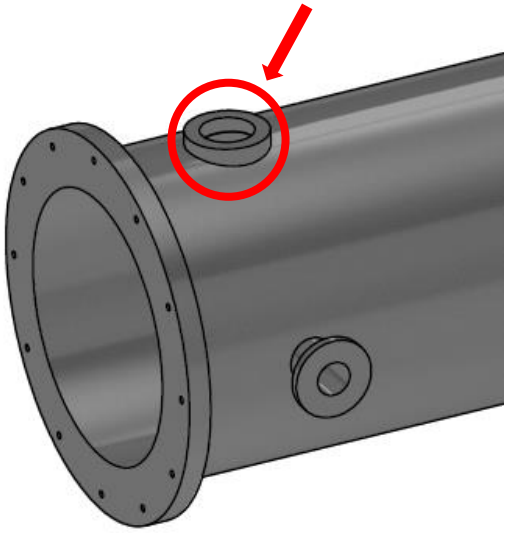
Seznam tabulek

Tab. 1 – základní NC funkce 6
Tab. 3 – Ukázka základních pomocnýc funkcí [2] 7
Tab. 2 – Ukázka základních přípravných funkcí [2] 7
Tab. 4 - Příklad systémových proměnných 9
Tab. 5 – Rozsah Q-parametrů [3] 9
Tab. 6 – Rozsah QS-parametrů [3] 10
Tab. 7 – Přehled datových typů [3] 10
Tab. 8 – Definice řezného nástroje [8] 23
Tab. 9 – Definice řezného nástroje [7] 29

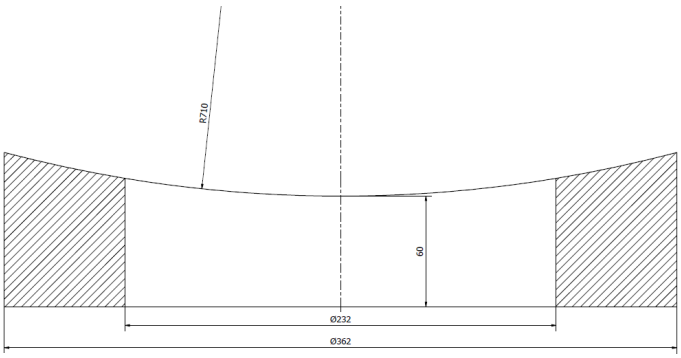
1. Úvod

S celkovým zrychlováním a usnadňováním chodu výrobních podniků je snaha o snížení času potřebného pro tvorbu výrobní dokumentace jako jsou seřizovací listy nástrojů, výrobních postupů a samotné tvorby programů pro obrábění součástí. Ve firmě STREICHER, spol. s r.o. Plzeň, která zadala téma této práce, *„Tvorba parametrického programu pro typovou součást“* je snaha o zapojení obsluhy stroje do úpravy programů přímo na stroji. Vzhledem k tomu, že se jedná většinou o typově podobné výrobky, jsou programy na strojích připraveny pomocí parametrického programování a změny v programech většinou jednoduché s možností úpravy obsluhou bez zásahu programátora. Proto se ve většině programů používaných ve výrobě firmy Streicher pracuje s parametry, aby byl program snadno upravitelný pro podobnou součást s jinými rozměry.

Firma Streicher je rozdělena do dvou divizí, jedna je stavební a druhá je strojní. Strojní divize se zaměřuje na zakázkovou výrobu vakuových komor a zařízení, dále pak výroba tlakových nádob. Nachází se zde svařovací hala, obráběcí hala, kontrolní hala a lakovna s pískovnou. Vzhledem k mému zaměření na programování NC strojů je práce zadána vedoucím obráběcí haly. Jedná se o přírubu, která se přivařuje na plášť tlakové komory. Při sestavování různých typů vakuových komor je potřeba na vnější plášť komory přivařit přírubu pro následné připojení do požadovaného systému. Tyto příruby mají však odlišné rozměry a různé jsou i velikosti tlakových komor. Proto je potřeba navrhnout parametrický program pro zajištění jeho univrzálního použití na různé velikosti přírub. Na Obr. 1 je vyznačeno použití příruby v sestavě a na Obr. 2 je výkres příruby.



Obr. 1 – příruba v sestavě



Obr. 2 – výkres příruby

1.1 Cíl práce

Cílem této práce by mělo být vytvořit parametrický program pro výše zmíněnou přírubu s ohledem na jednoduchost zadávání parametrů přímo z výkresu a srozumitelnost pro obsluhu. Vytvoření programu by mělo proběhnout ve dvou řídících systémech, Sinumerik 840D a Heidenhain iTNC 530.

2. Analýza současného stavu NC programování

2.1 Definice NC programování a výhody parametrického programování

V současné době s rostoucími požadavky na přesnost a vyšší produktivitu výroby se zdokonalují výrobní možnosti. Pro dosažení těchto požadavků bylo zapotřebí vzdát se zastaralých způsobů obrábění a řízení pomocí šablon, vaček a kopírování podle modelu a nahradit je novějšími způsoby, jako jsou CNC obráběcí centra s moderními řídícími systémy. Tyto systémy umožňují více možností programování počínaje ručním programováním pomocí ISO kódu, přes uživatelsky přívětivější dílenské programování až po programování složitých tvarů v CAD/CAM systémech. NC obráběcí stroj zpracovává data technologických vstupů za pomocí řídícího systému z číslicově kódovaných informací, jako jsou informace o geometrii (určení velikosti a směru dráhy, velikosti obrobku), technologii (otáčky, posuvy, nástroje) a organizaci programu (začátek a konec programu, programové skoky).

2.1.1 Struktura NC programu

Při programování NC strojů musíme dodržovat určitou syntaxi, která je ale různá u každého řídícího systému jiná. NC program se skládá z bloků, což jsou jednotlivé řádky programu. Každý blok může obsahovat geometrické, technologické, pomocné a ostatní informace (Obr.3). Podle kontextu zařazení bloků v programu (začátek/konec programu, začátek podprogramu, cyklus) se bloky skládají z posloupnosti adres a číselných hodnot. Číselné hodnoty jsou navíc doplněny o znaménko, zda se jedná o kladnou nebo zápornou hodnotu. Jednotlivé bloky mohou tvořit skupiny bloků, které mají pevnou stavbu podle účelu použití (začátek/konec programu, výměna nástroje atd.).[5]

Pro zachování přehlednosti struktury bloku, je třeba dodržet posloupnost jednotlivých příkazů. Posloupnost je následující: N…, G…, X…, Y…, Z…, F…, S…, T…, D…, M…, H…, s tím že některé funkce mohou být v rámci jednoho bloku použity víckrát, viz. G… a M…

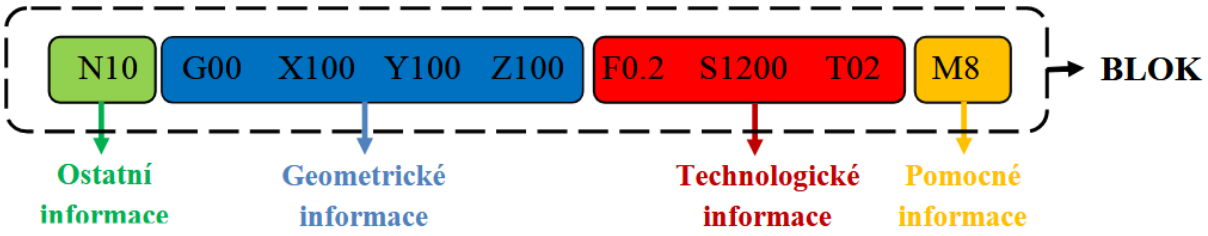
Délku určitého bloku lze rozdělit podle délky na formát s konstantní délkou a formát s proměnnou délkou. U bloku s konstantní délkou musíme dodržet každou polohu adresy bloku a nesmíme žádnou vynechat, a to i případě, že se adresa opakuje z předchozího řádku. (Příklad 1). Ve druhém případě bloku s proměnnou délkou, lze vynechat adresy které se již objevili v předchozím řádku (Příklad 2). Značení adres, co jaká znamená je dáno normou, proto se používá označení ISO programování (Tab. 1).[5]

Příklad 1: *N40 G00 X25 Y50 Z100 F0,8*
N50 G00 X30 Y50 Z100 F0,8

Příklad 2: *N40 G00 X25 Y50 Z100 F0,8*
N50 X30

Písmeno adresy	Funkce
A, B, C	Rotační pohyb kolem os X, Y, Z
F	Posuv
G	Přípravná funkce
I, J, K	Polární souřadnice X, Y, Z
M	Pomocné funkce
Q, R	Volba parametrů programu
S	Otáčky
T	Volba nástroje
X, Y, Z	Osy

Tab. 1 – základní NC funkce



Obr. 3 – syntaxe bloku [6]

2.1.2 ISO kód

Programování v ISO kódu vyžaduje hlubší znalost programovacího jazyka. Je potřeba znát základní přípravné (Tab. 2) a pomocné (Tab. 3) funkce. Pomocí přípravných funkcí lze vykonat pohybové a přípravné instrukce pro obráběcí stroj a jsou součástí řídícího systému. Pro více informací lze nahlédnout do normy DIN 66025. K ovládání periferií stroje jako jsou otáčky, chlazení a například i ovládání automatických upínačů slouží pomocné funkce a ty jsou součástí řízení stroje. [2]

G0	lineární posuv rychloposuvem
G1	lineární posuv pracovním posuvem
G2	kruhová interpolace po směru hod. ručiček
G3	kruhová interpolace proti směru hod. ručiček
G4	časová prodleva
G17,18,19	volba obráběcí roviny X-Y, Z-X, Y-Z
G40	zrušení korekce
G41,42	korekce z leva, z prava
G54-59	posunutí nulového bodu obrobku
G90	absolutní programování
G91	přírůstkové programování

Tab. 3 – Ukázka základních přípravných funkcí [2]

M0	programový stop
M1	podmíněný programový stop
M2	konec programu
M3	spuštění otáček po směru hod. ručiček
M4	spuštění otáček proti směru hod. ručiček
M5	zastavení otáček vřetena
M6	výměna nástroje
M8	spuštění vnějšího kapalinového chlazení
M9	zastavení chlazení

Tab. 2 – Ukázka základních pomocnýc funkcí [2]

2.1.3 Dílenské programování - dialog

Při dílenském programování se využívá nadstavby jednotlivých řídících systémů, do nichž uživatel nezadává ISO kód, ale pouze data do předem připravených buněk. Ve většině případů je programování v dialogu (dílenské programování) doprovázeno grafickým znázorněním a uživatel je veden pokyny softwaru. Dílenské programování lze najít u většiny řídících systémů, u Heidenhainu je to SmartTNC, u Sinumeriku je pro frézování ShopMill a pro soustružení ShopTurn a u dalších systémů je to podobné. [5]

2.1.4 CAD/CAM programování

Pomocí programování pomocí CAD/CAM lze vytvořit program na externím počítači a poté nahrání hotového programu do CNC stroje. Tvorba programu spočívá v naimportování nebo vlastní tvorbě modelu/geometrií, nastavení CAM projektu, tvorbě drah a definování řezných podmínek, grafické simulaci a konečném generování programu pomocí postprocesoru, což je vlastně převedení dat z PC do programovacího jazyka obráběcího stroje.

2.2 Parametrické programování

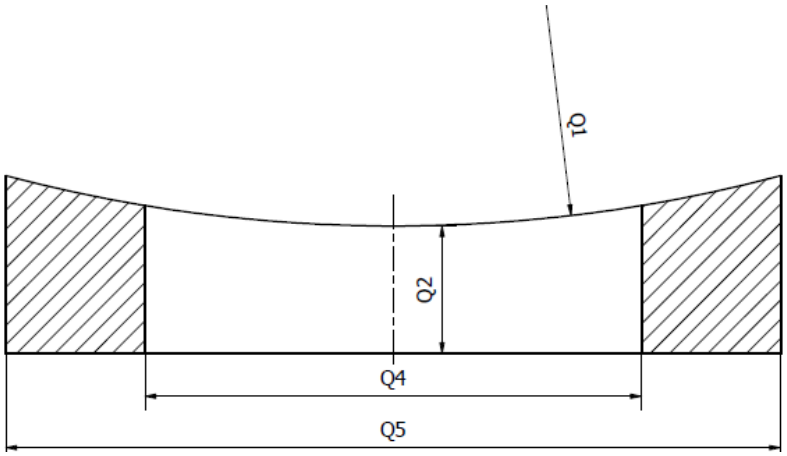
Při vytváření NC programu se řídíme výkresovou dokumentací s rozměry součásti a program je poté tvořen jednotlivými bloky s číselnými hodnotami polohy nástroje a řezných podmínek (např. N25 G01 X50 Y50 F100). U takto vytvořeného programu lze přiřadit jeden NC program pouze jednomu typu součásti. Tuto nevýhodu odstraňuje parametrické programování.[9]

Parametrické programování je založeno na definici parametrů do tabulky parametrů nebo přímo do hlavního NC programu, které lze následně využít v programu. Hodnoty parametrů si spuštěný program načítá a rovnou je aplikuje pro funkce definované těmito parametry nebo je přepočítává pomocí rovnic a následně používá pro pohyb nástroje. Při využití parametrického programování lze zvýšit flexibilitu jednoho programu pro geometricky podobné součásti, vytvoření tzv. rozměrové řady součásti.[9]

Další nespornou výhodou parametrického programování je zjednodušení a universálnost NC programů. V některých podnicích, kde do programu zasahuje obsluha stroje a částečně si stroj programuje sama, je snaha hlavního programátora o zjednodušení programu pomocí parametrického programování. Programátor napíše parametrický program a definuje parametry, které bude obsluha stroje měnit, aniž by musela měnit strukturu celého NC programu a dělat jakékoliv úpravy NC programu.[9]

2.2.1 Výhody parametrizace při programování

Pokud máme více tvarově podobných dílů, ale mění se u nich příslušné parametry a chtěli bychom si ušetřit čas při přípravě programů, je výhodné použít parametrizaci NC programu. Typickým příkladem takové parametrizace jsou například šrouby, matice, a další normalizované řady součástí, které lze najít ve strojnických tabulkách. U těchto součástí se mění dané rozměry, tedy parametry. Jednoduchá parametrizace součásti je vidět na Obr. 4, kde lze jednoduše měnit rozměry součásti pouze pomocí parametrů. Pomocí parametrizace si můžeme vytvořit vlastní rozměrovou řadu jakékoliv součásti, kde máme typovou součást, u které se mění parametry podle požadavků zákazníka. Programátor má dvě možnosti, napsat pokaždé nový program na specifickou součást, nebo použít parametry a napsat parametrický program pro řadu podobných typových výrobků, což mu ušetří čas a práci do budoucna.



Obr. 4 – příklad parametrizace součásti

2.2.2 Rozdělení parametrů

Parametry používané při programování lze rozdělit podle možnosti definování a podle způsobu jejich použití. Tyto parametry lze využívat společně s matematickými funkcemi a řídícími strukturami, pomocí kterých lze vytvářet sofistikované programy a cykly. Použití parametrů se liší mezi jednotlivými řídícími systémy. Sinumerik nabízí uživateli tři různé typy proměných, jedná se o systémové proměnné, předdefinované uživatelské proměnné a uživatelem definované proměnné. Systémové proměnné jsou dány výrobcem stroje a mají určité podmínky zápisu. Pomocí těchto proměných lze definovat parametry stroje jako jsou otáčky, posuvy, polohy os a korekce nástrojů.[3]

Příkaz	Význam
$AA_IW[MZ1]	Systémová proměnná
$TC_DP[N]	Typ nástroje
$P_TOOLNO	Číslo aktivního nástroje
$P_TOOL	Aktivní břit nástroje

Tab. 4 - Příklad systémových proměnných

Dalším typem proměných jsou předdefinované uživatelské proměnné. Do této kategorie patří početní parametry R, které jsou definovány datovým typem REAL. Tyto parametry jsou definovány uživatelem a v systému jsou používány až při zpracování programu.[3]

Sinumerik nabízí možnost definice vlastních proměnných. Pomocí příkazu DEF lze zadefinovat vlastní proměnnou a přiředit jí hodnotu. Uživatelské proměnné lze rozdělit do několika skupin: **lokální uživatelské proměnné (LUD)** – tyto parametry jsou definovány v programu, který ale není výrobní program a jsou při konci programu nebo při resetu NC programu vymazány. Dále jsou zde **programově globální uživatelské proměnné (PUD)** – jsou parametry definovány přímo ve výrobním programu a jsou vymazány s koncem NC programu. Poslední jsou **globální uživatelské proměnné (GUD)** – jsou globálními proměnnými NC systému, lze je využít ve více programech a zůstávají zadefinované i po vypnutí NC systému. K zadefinování použijeme syntaxi:[3]

DEF (typ proměnné) (název) [hodnota pole až 3 dimenze] = výchozí hodnota

Upozornění:

- Maximálně 31 znaků
- První dva znaky musí být písmena a/nebo znak podtržení.
- Znak "$" je vyhrazen pro systémové proměnné a nesmí se používat.

Řídící systém Heidenhain využívá parametry Q označeny od 0 do 1999. Parametry mohou mít různé účinky, viz následující tabulka:

Rozsah	Význam
Q0 – Q99	Volně použitelné parametry účinné globálně pro všechny programy v TNC
Q100 - 199	Parametry pro speciální funkce TNC
Q200 – Q1199	Parametry používané především pro cykly
Q1200 – Q1399	Parametry používané především pro cykly od výrobce stroje
Q1400 – Q1499	Parametry používané především pro cykly od výrobce Call-aktivní
Q1500 – Q1599	Parametry používané především pro cykly od výrobce Def-aktivní
Q1600 – Q1999	Parametry pro uživatele

Tab. 5 – Rozsah Q-parametrů [3]

Dále lze definovat parametry QL, které působí pouze místně v rámci jednoho NC programu. Mohou být označeny od 0 do 499 a všechny jsou pro uživatelské účely. Jedním s posledních možností jak definovat parametry v Heidenhainu jsou parametry QR. Tyto parametry působí trvale na všechny NC programy a i po výpadku NC systému zůstavají v paměti TNC. Mohou být označeny od 0 do 499 a všechny jsou pro uživatelské účely.[3]

Poslední možností jak definovat parametry v Heidenhainu jsou QS-parametry do nichž jsou zadávány data typu STRING, tedy jde s nimi zpracovávat texty. Maximální dovolená dálka přiřazení je 254 znaků. Význam jednotlivých rozsahů parametrů je v následující tabulce:[3]

Rozsah	Význam
Q0 – Q99	Parametry pro uživatele
Q100 - 199	Parametry pro speciální funkce TNC
Q200 – Q1199	Parametry používané především pro cykly
Q1200 – Q1399	Parametry používané především pro cykly od výrobce
Q1400 – Q1599	Parametry používané především pro cykly od výrobce
Q1600 – Q1999	Parametry pro uživatele

Tab. 6 – Rozsah QS-parametrů [3]

Vzhledem k tomu že jsou výše používány zkratky datových typů jako REAL, INT, STRING, atd., jsou tyto pojmy vysvětleny v následující tabulce:

Datový typ	Význam
INT	Anglicky integer, umožňuje zapisovat pouze hodnoty celých čísel doplněných o znaménko
REAL	Reálné číslo umožňuje zapisovat všechna čísla z množiny čísel R, tedy od - ∞ do + ∞ včetně desetinných čísel
BOOL	Booleovká proměnná nabývá pouze hodnot TRUE (1) nebo FALSE (2)
CHAR	Anglicky character, v programování se označuje jako řetězcový datový typ
STRING	Pomocí datového typu STRING lze nadefinovat libovolně dlouhý řetězec znaků, které jsou brány jako text
AXIS	Indikátor osy/vřeten
FRAME	Geometrické údaje pro statickou transformaci souřadného systému

Tab. 7 – Přehled datových typů [3]

2.3 Využití vyššího programovacího jazyka

Parametry využívané v programování NC strojů slouží pro napsání parametrické programu, pro využití funkcí a výpočtů, pro používání a tvorbu cyklů a pro možnost využívání programových skoků.

2.3.1 Definování parametrů

Nejjednodušší způsob parametrického programování spočívá pouze v nadefinování parametrů přímo v programu nebo do tabulky. Následně se v programu místo číselných hodnot dosazují nadefinované parametry. Definování parametrů se liší podle typu řídicího systému, viz další kapitola.[9]

2.3.2 Funkce a výpočty

Pokud chceme naprogramovat složitější tvar součásti a už nám nestačí klasické programování ani programování v dialogu je potřeba využít možnosti zadávání parametrů pomocí funkcí a výpočtů. V těchto případech se všechny tyto výpočty řídí pomocí již známých pravidel matematiky.[9]

2.3.3 Logické funkce

Při programování můžeme využívat tzv. pokročilých metod programování, kam lze zařadit programové skoky a smyčky. Tyto funkce lze zapsat pomocí ISO kódu v editorech, v některých případech i v dílenském programování. Díky programovým skokům máme možnost se řízeně pohybovat a v programu a pomocí podmínek popsaných níže můžeme výrazně zvýšit možnosti ručního programování. Pomocí programových skoků a smyček lze vytvořit adaptivní řízení nebo automaticky cyklující program. Využívají se zde základní logické operace z klasického IT programování jako IF, ELSE, GOTO, WHILE, LOOP, REPEAT.[4]

- **Programové skoky**

Nepodmíněné skoky lze použít tam, kde je potřeba změnit posloupnost bloků v NC programu. Pokud bychom chtěli přeskočit určité bloky, použijeme funkci GOTOF VPRED, kde VPRED určuje místo, kam program přeskočí. Bloky mezi funkcí GOTOF a VPRED nebudou přečteny a vykonány. Pro zpětné přeskočení k začátku programu sloučí funkce GOTOB NAVRAT, kde NAVRAT určuje polohu bloku, kam program přeskočí a poté vykoná všechny následující bloky. Cíl kam program skočí může mít jakékoliv jméno, musí však být ukončeno dvojtečkou, jinak program vyhodí chybové hlášení (slovo bez dvojtečky je branné jako volání podprogramu a absence takového názvu aktivuje chybové hlášení). Těchto skoků lze využít u podobných součástí, kde u jedné budeme například vrtat díry a u druhé ne, tak můžeme přeskočit bloky vrtání u druhé součásti. Příklad nepodmíněného skoku pomocí funkce GOTOF lze vidět na Obr. 5.[4]

Obr. 5 – příklad – nepodmíněný skok [4]

V případě podmíněných programových skoků se využívá příkazu IF, která definuje podmínku programového skoku. Pokud je podmínka splněna, lze provádět skoky pomocí funkcí GOTOF VPRED, GOTOB NAVRAT a GOTO CÍL. Pokud by však podmínka splněna nebyla, podmíněný skok se přeskočí a program pokračuje dál. Blok s podmíněným skokem může vypadat takto: N100 IF Q1<Q2 GOTO LBL10, kde Q1<Q2 je podmínka a lze ji stanovit pomocí matematických operátorů. Dále lze společně s funcí IF využít funci ELSE, kde je potřeba realizovat jednu volbu ze dvou variant. V programu se definuje následovně:[4]

IF<podmínka>

<podmínka>=true GOTO LBL1

ELSE

<podmínka>=false GOTO LBL2

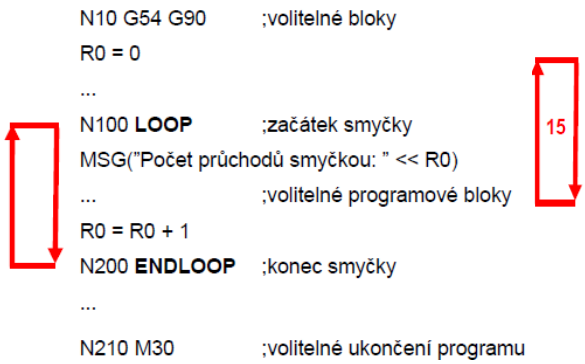
ENDIF

Posloupnost bloků závisí jestli je podmínka splněna či nikoliv. Pokud je podmínka splněna skočí program na LBL1, v opačném případě program skočí na LBL2.

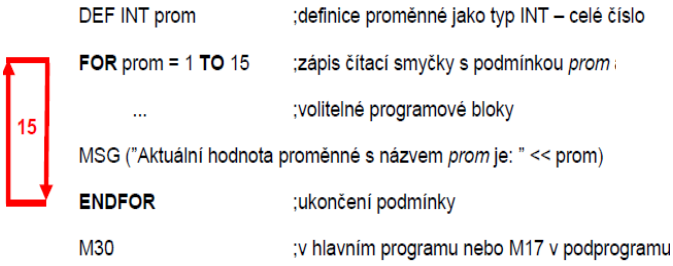
Obr. 6 – příklad – podmíněný skok [4]

- **Nekonečná a čítací smyčka**

Mezi programové skoky lze zařadit nekonečnou smyčku a čítací smyčku. Nekonečná smyčka slouží k tvorbě tzv. nekonečných programů. Pokud se smyčka dostane na konec, dojde ke skoku na začátek smyčky. Lze je kombinovat s parametry nebo textovou poznámkou, například sledování počtu opakování dané smyčky. Pomocí funkce MSG lze na obrazovce stroje vidět počet průchodů smyčkou. Čítací smyčka má na rozdíl od nekonečné dán přesný počet opakování. Hodnoty jsou zadávány typem INT (celočíselná hodnota) a počítá od nejnižší hodnoty po nejvyšší.[4]



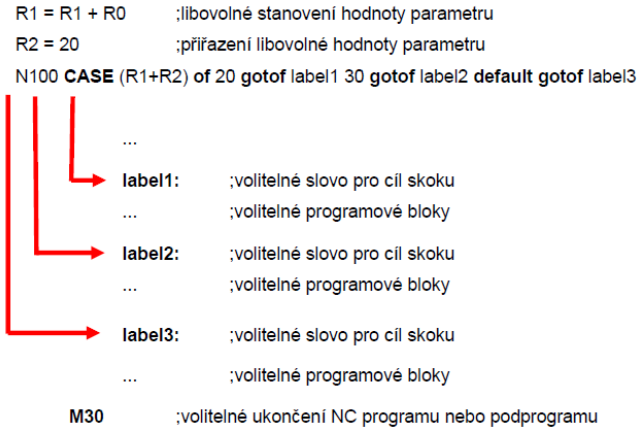
Obr. 7 – příklad – nekonečná smyčka [4]



Obr. 8 – čítací smyčka (opakování od 1 do 15) [4]

- **Funkce CASE a REPEAT**

Dalšími funkcemi, které patří mezi programové skoky jsou REPEAT a CASE. REPEAT slouží k opakování části programu. Funkce CASE umožnuje porovnávání různých hodnot a následně podle hodnoty výsledku skočit v programu na určitý cíl. Skok může být proved směrem k začátku programu nebo ke konci programu.[4]



Obr. 9 – příklad – funkce CASE [4]

Programové skoky je potřeba kombinovat společně s parametrickým programováním pro vytváření složitějších tvarů obrobků. Takto vytvořený NC program přepočítává hodnotu parametru. Parametr je vložen do rovnice, jejíž výsledná hodnota je dosazena do bloku s lineární interpolací. Poté je kontrolována podmínka, zda vypočtená hodnota parametru souhlasí s podmínkou. Pokud podmínka není splněna, program skočí zpět před výpočet, připočte provede znovu výpočet a opakuje bloky s lineární interpolací, dokud není podmínka splněna. Tento typ programování lze využít, pokud má obrobek tvar matematicky definované křivky, které program zjednodušuje na elementární lineární kroky. Příkladem můžou být např.: elipsa, hyperbola, sinusovka nebo Archimedova spirála.[4]

2.3.4 Tvorba a využívání předdefinovaných cyklů

Pokud bychom potřebovali určitý cyklus, který se ovšem v daném řídícím systému nenachází např. frézování obecné kapsy, vrtání děr, hrubování obecných ploch atd. je možné vytvořit si vlastní cyklus. Výhodou vytvoření cyklu pomocí parametrů je velká škála použití cyklu v jakémkoliv programu a měnit se budou pouze definované parametry, kterými je cyklus definován. Některé řídící systémy již disponují některými cykly pro základní obráběcí operace, a proto je můžeme rovnou využívat a nemusíme si je sami definovat.[9]

3. Porovnání parametrického programování pro zadané řídicí systémy

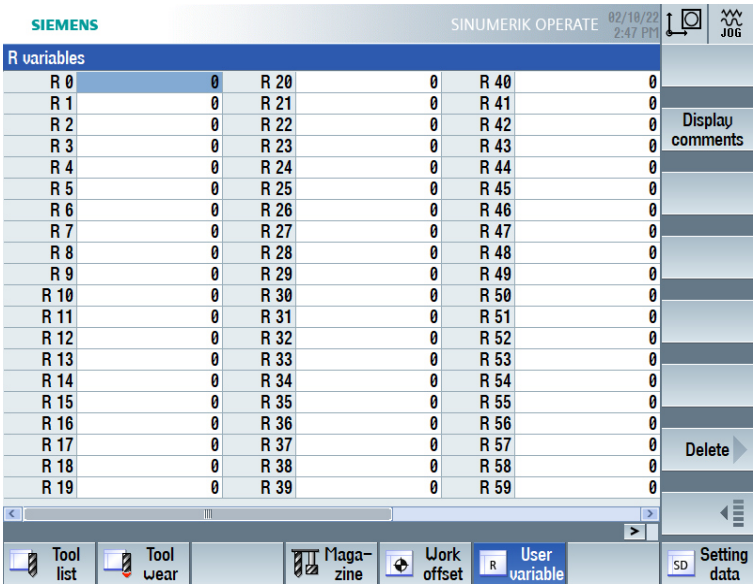
V rámci této bakalářské práce se budu zabývat řídícími systémy od firmy Siemens a Heidenhain. Od Siemensu nás bude zajímat verze Sinumerik 840 pro horizontky FCW 140 a FCW 150 od firmy Škoda MachineTool. Dále pak horizontky SLP8000 a FL12000 od firmy Solaruce. Z řídících systémů nabízených firmou Heidenhain nás bude zajímat verze iTNC530 a iTNC620. Každý z těchto řídících systémů podporuje jak programování v ISO kódu, tak mají i vestavěný modul pro dílenské programování, předdefinované cykly a umožnují využívání parametrického programování. Výhodou je, že všechny výše zmíněné systémy mají verzi, kterou lze naistalovat na stolní počítač programátora. Ten má tak možnost programovat, simulovat a ladit program pomocí grafického zobrazení, který je součástí programovacího systému.

3.1 Řídící systém Sinumerik 840

S pomocí řídícího systému Sinumerik 840 lze programovat ručně pomocí ISO kódu, lze využít nadstavbu dílenského programování pro frézování (ShopMill) a pro soustružení (ShopMill), ale v této práci se budeme zaobírat parametrickým programováním, kterým lze vypracovat univerzální NC program, což je náplní této bakalářské práce.

3.1.1 Tvorba parametrického programu

Tvorbou parametrického programu lze vytvořit flexibilní NC program pro typovou součást, kde jsou zadané rozměry součásti, které se mění, tak je lze parametrizovat. Řídící systém Sinumerik využívá parametry R, které lze definovat několika způsoby. Jedním z nich je definování v předem připravené tabulce, kterou systém obsahuje, nebo lze parametry definovat přímo v hlavním programu. Výhodou definování parametrů přímo v hlavním programu spočívá v tom, že si ke každému můžeme napsat poznámku, co který parametr mění, toto u tabulkového zadávání parametrů neplatí. Další variantou je kombinace dvou předchozích možností parametrizace, část parametrů je v tabulce a zbytek se nadefinuje přímo v programu. Pokud máme parametry za definované, můžeme začít psát samotný NC program. Tvorba je v podstatě stejná jako bez parametrů, akorát místo číselných hodnot zadáváme parametry.[9]



Obr. 10 – tabulka parametrů v Sinumeriku

Příkladem může být výroba šroubu se závitem a osazením, kde se u jednotlivých zakázek mění pouze délka závitu a poloha osazení. Napíše se program, kde budou tyto 2 hodnoty zparametrizovány a po vyrobení jednoho typu nemusíme zdlouhavě přepisovat nebo editovat program, ale můžeme pouze změnit tyto 2 parametry.[9]

V systému Sinumerik lze využít 0 – 999 výpočetních parametrů, a však tento počet je závislý na používané verzi řídícího systému. Těmto parametrům mohou být přiřazeny hodnoty v rozsahu od ± 0.000 000 1 ÷ 99 999 999. K dispozici je tedy prostor pro 8 desetinných míst, znaménko a desetinnou čárku, kdy u kladného čísla můžeme vypustit znaménko a u celého čísla můžeme vypustit desetinnou čárku. V případě nedostačujícího rozsahu, lze čísla zapisovat pomocí exponenciálního zápisu. Více ohledně využívání vyššího programovacího jazyka v Sinumeriku viz. kapitola 2.3.3.[9]

3.2 Řídící systém Heidenhain iTNC530

Řídící systém Heidenhain slouží k programování NC strojů, zejména pak frézek, vrtaček a obráběcích center. Programovat lze přímo na stroji za pomoci nadstavby dílenského programování nebo pomocí ručního programování v ISO kódu. Další možností, jak programovat v tomto řídícím systému je programovací stanice připojená k externímu počítači programátora. Programátor napíše program na počítači, uloží ho na sítový disk stroje, kde si ho obsluha nahraje na stroj a po odladění může obrábět. Dílenské programování obsahuje grafickou podporu drah, grafickou simulaci operací ve 2D i 3D, má nadefinované určité cykly (vrtací, frézovací, závitovací cykly) a pro složitější tvary nebo špatně okótované výkresy lze naprogramovat pomocí FK (free contour-programming) technologií, která dokáže dopočítat automaticky chybějící body obrysu.

3.2.1 Tvorba parametrického programu

Při tvorbě parametrického programu dosadíme místo číselných hodnot Q-parametry, které lze použít pro hodnoty souřadnic, posuvy, otáčky nebo data cyklů. Dále lze využít parametry k programování obrysů popsaných matematickými funkcemi nebo k řízení bloků pomocí logických funkcí. Řídící systém Heidenhain iTNC530 má k dispozici parametry Q0 až Q1999, které jsou však rozděleny podle účinku. Pokud chceme v programu vyvolat Q-parametr, musíme stisknout klávesu Q, která nám nabídne na displeji soft tlačítka s možnostmi práce s parametry:[1]

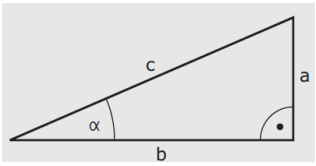
- ***Základní matematické funkce***

Mezi základní funkce patří samotné přiřazení hodnoty k parametru, sčítaní a odčítání, násobení a dělení, mocnina a odmocnina a musí se řídit pravidly matemetiky. Každá zadaná hodnota i každý parametr může být opatřen znaménkem, zda se jená o kladnou nebo zápornou hodnotu. Nevýhodou těchto funkcí může být, že nelze definovat více jak číselné hodnoty, respektive parametry. To znamená, že nelze například sčítat 3 čísla pomocí funce FN1. K eliminování této nevýhody se musí provést jeden parametrický výpočet 2 čísel a v následujícím parametru přičteme třetí hodnotu k prvnímu parametru.[1]

- ***Úhlové funkce***

Pomocí těchto funkcí lze definovat sinus, kosinus a tangens v pravoúhlém trojúhelníku pomocí trigonometrie a využívá vlastností Pythagorovy věty. Veškeré úhlové hodnoty jsou zadávány ve stupních.[1]

$\sin \alpha = {}^{a}/_{c}$ … sinus
$\cos \alpha = {}^{b}/_{c}$ … kosinus
$\tan \alpha = {}^{b}/_{c}$ … tangens



- ***Skok – logické operace***

Programové skoky v programování NC strojů lze využít pro větší flexibilitu programování složitějších tvarů, možnosti cyklování programu i bez zásahu obsluhy. Výhodou je, že tyto bloky řídící systém přečte pouze v případě, že je podmínka v bloku splněna. Skoky v programu mohou být podmíněné nebo nepodmíněné a mohou být prováděny dopředu v programu nebo ke konci programu. Lze je využít k cílenému cyklení programu až po splnění předdefinované podmínky.[1]

- ***Postup – přímé zadávání vzorců***

Při používání parametrů lze zadávat parametry i pomocí matematických vzorců. Pomocí přímého zadávání vzorců se program stává přehlednější a jednodušší pro obsluhu. Vzorce lze zadávat pomocí různých kombinací matematických operaci, které se ovšem musí řídit matematickými pravidly. Zásadní rozdíl u přímého zadávání vzorců a základní funkcí FN1 je že postupovým zadáváním lze sečíst více členů než pouze dva.[1]

- ***Zvláštní funkce definované firmou Heidenhain***

Do této kategorie patří možnosti přerušení programu a zobrazení chybového hlášení, které je předepsané firmou Heidenhain. Další funkcí patřící do této skupiny je čtení systémových dat, které umožňuje načíst systémová data do parametru Q. Lze sem zařadit čtení dat z PLC nebo funkci na čekání mezi NC a PLC, kdy se zastaví obrábění, dokud není splněna předem definovaná podmínka.[1]

- ***Přídavné funkce definované firmou Heidenhain***

Tyto funkce slouží k zapisování parametrů do tabulky. Definovatelnou tabulku lze pomocí funkce TABOPEN otevřít pro zápis a čtení zapsaných hodnot. V programu může být vždy otevřena pouze jedna tabulka. Pomocí funkce TABWRITE lze v již otevřené tabulce zapisovat a měnit data. Pomocí funkce TABREAD lze přečíst data z již předem otevřené tabulky.[1]

3.3 Porovnání programování s parametry a bez nich

V rámci této kapitoly porovnám způsob programování bez parametrů, programování pomocí využívání parametrů dříve, a nakonec jaké jsou všechny možnosti využívání parametrů v dnešní době. Jde o to ukázat vývoj parametrického programování a rozdíl ve zjednodušení programování při využívání výhod parametrického programování vůči programu bez parametrů.

S postupným vývojem obráběcích strojů a možností řízení těchto strojů se v minulosti museli vyvíjet současně se stroji i řídící systémy. Mezi první možnosti číslicově řízených strojů patří děrné štítky nebo magnetické pásky, které umožnovali pouze pravoúhlé řízení. Vše se programovalo ručně, díly byly omezeny na jednoduchou geometrii a jeden program vždy znamenal jednu určitou součást.

```
0 BEGIN PGM 1001 MM
1 BLK FORM 0.1 Z X-181 Y-181 Z-85
2 BLK FORM 0.2 X+181 Y+181 Z+0
10 L Z+0 R0 FMAX M91
13 TOOL CALL 1 Z S390
14 L Z+0 R0 FMAX M91
15 M3
16 L X-1.979 Y-0.454 R0 FMAX
17 L Z+15 R0 FMAX
58114 L Z+0 R0 FMAX M91
25 L Z-22.897 FMAX
26 L Z-22.928 FQ53
27 L X-1.972 Y-0.452 Z-23.104
28 L X-1.949 Y-0.447 Z-23.279
29 L X-1.912 Y-0.439 Z-23.451
30 L X-1.859 Y-0.427 Z-23.619
31 L X-1.793 Y-0.411 Z-23.781
32 L X-1.713 Y-0.393 Z-23.937
.
.
.
58101 L X-83.183 Y-160.776 Z-18.285
58102 L X-83.076 Y-160.833 Z-18.156
58103 L X-82.98 Y-160.885 Z-18.016
58104 L X-82.894 Y-160.93 Z-17.867
58105 L X-82.821 Y-160.969 Z-17.711
58106 L X-82.76 Y-161.001 Z-17.548
58107 L X-82.713 Y-161.027 Z-17.379
58108 L X-82.678 Y-161.045 Z-17.206
58109 L X-82.658 Y-161.056 Z-17.03
58110 L X-82.651 Y-161.06 Z-16.853
58111 L Z+15 FMAX
M30
```

Obr. 12 – program bez použití parametrů

```
0 BEGIN PGM 1 MM
1 BLK FORM 0.1 Z  X-150  Y-150  Z+0
2 BLK FORM 0.2  X+150  Y+150  Z+70
3 FN 0: Q1 =+300
4 FN 0: Q2 =+40
5 FN 0: Q3 =+1
6 FN 0: Q4 =+0Y
7 FN 0: Q5 =+300
8 FN 0: Q6 =+1
9 FN 0: Q10 =+70
10 LBL 3
11 Q7 = Q5 / 2
12 Q8 = Q4 / 2
13 FN 0: Q9 = 0
14 Q12 = Q2 + 7
15 Q40 = Q1 + Q12
16 CYCL DEF 7.0 NULOVY BOD
17 CYCL DEF 7.1  Z+Q40
18 L  X+200  Y+200  Z+200 R0 FMAX
19 TOOL CALL 52 Z S1200 F250
20 L  X+0  Y-7.5  Z+100 R0 FMAX
21 FN 12: IF +Q4 LT +50 GOTO LBL 2
22 CALL LBL 1
23 CYCL DEF 7.0 NULOVY BOD
24 CYCL DEF 7.1  Z-Q40
25 L  Z+200 R0 FMAX M30
26 LBL 1
27 Q8 = Q8 + ( Q3 * Q6 ) / 360
28 Q9 = Q9 - 1
29 Q41 = Q8 * COS Q9
30 Q42 = Q8 * SIN Q9
31 Q43 = - ( SQRT ( SQ Q1 - SQ Q41 ) )
32 L  X+Q41  Y+Q42  Z+Q43 RL FMAX M3
33 FN 12: IF +Q8 LT +Q7 GOTO LBL 1
34 LBL 0
46 END PGM 1 MM
```

Obr. 11 – program s využitím parametrů

Při příchodu počítačů a CNC strojů lze stále programovat ručně, ale některé systémy již obsahují určité nadstavby jako např. dílenské programování, předem definované cykly a umožnují využívat parametry a vyšší programovací jazyk. Začněme u ručního programování bez použití parametrů a funkcí s tím spojených. Při ručním programování se spoléháme pouze na svoje znalosti programovacího jazyka, na znalost syntaxe jednotlivých bloků a jednoduchou grafickou podporu, kterou některé systémy podporují. Programy jsou pouze pro jeden typ součásti, pro jinou součást je třeba program celý přepsat. Nelze používat cykly, programové skoky nebo parametry.

Proto se pro zjednodušení práce a zefektivnění výroby začínají v programech vyskytovat parametry, které zjednoduší práci s programem jeho uživateli, flexibilitu programu pro součásti, které mají stejný tvar, ale rozdílné rozměry a možnost provádění základních matematických operací (sčítání, odčítání, násobení, dělení).

S postupným rozvojem technologie a vylepšováním řídících systému se v rámci parametrického programování a používání parametrů v programu bavíme o využívání předdefinovaných cyklů, tvorbě vlastních cyklů, využívání programových skoků a kombinaci těchto funkcí. S novodobou verzí parametrického programování lze programovat velké škály tvarových ploch i bez použití CAD/CAM sytému, dále lze vytvořit program pro celé rozměrové řady výrobků, pro které bude vždy jen jeden program s nadefinovanými parametry, které se budou měnit podle potřebných požadavků.

4. Analýza současného stavu ve firmě Streicher

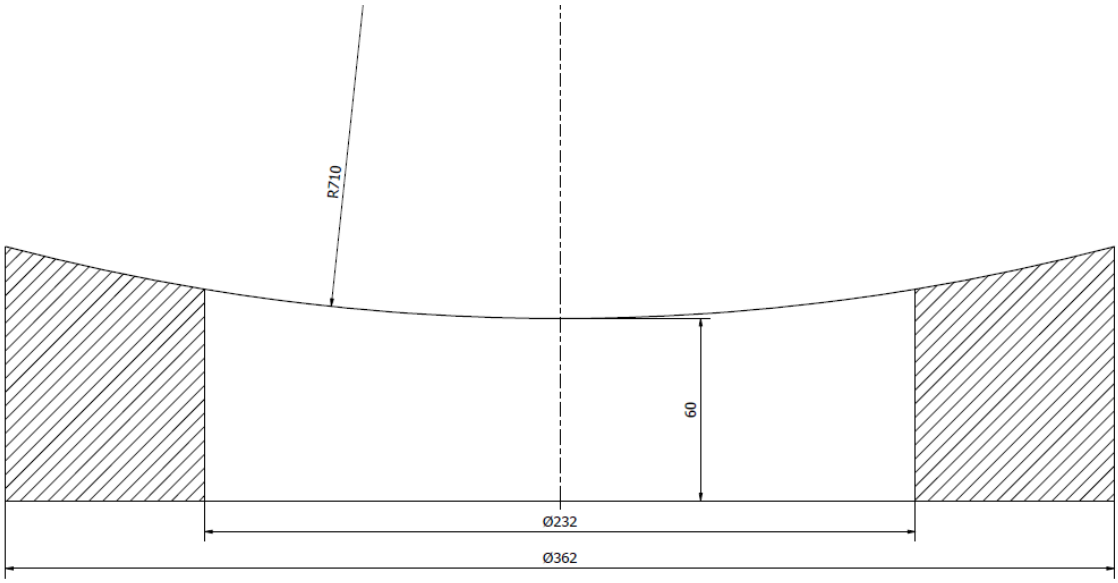
4.1 Úvod do problematiky

Praktická část bakalářské práce se zaměřuje na návrh a optimalizaci programu na součást zadanou firmou Streicher. Cílem této práce by mělo být prozkoumání problematiky, optimalizace programu, návrh optimální řešení a následné odladění programu na stroji. Jedná se o přírubu s vyfrézovaným rádiusem pro následné broušení a navaření na část tlakové komory. Jde o typový díl, který firma Streicher používá ve více velikostních variantách. Proto se při tvorbě programu zaměřím na parametrické programování pro zjednodušení práce programátora i obsluhy stroje.

Požadavky firmy Streicher jsou, aby se vytvořil odladěný program, který bude v paměti každého stroje a pouze se budou měnit parametry podle výkresové dokumentace. Tuto činnost bude provádět obsluha a tím se usnadní práce pro programátora. Díl může být vyráběn z různých materiálů např. S235JRG1, S235J2G3, 1.4301 nebo 1.4571. Jako nástroj pro obrábění se používá válcová fréza s VBD, která jede plnou hloubku s relativně malým bočním úběrem.

Vzhledem ke složitosti obráběné plochy, která vyžaduje souvislé obrábění ve 3 osách je samotné ruční programování v ISO kódu velmi složité. Pro obrobení takového tvaru lze využít CAM systémy, ale ty mají nevýhodu, že je nelze parametrizovat a pro každý rozměr součásti by se musel generovat nový program. Proto se v této bakalářské práci zaměřím na využití parametrického programování.

Ve firmě Streicher, kde jsem dostal zadanou tuto součást, již tento díl několik let obrábí. Technologii, kterou k tomu používají, je frézování po spirále s maximální hloubkou třísky a malým bočním krokem cca 1mm nástrojem průměr 32 mm s vyměnitelnými břitovými destičkami. Snaha této práce bude spočívat ve zlepšení technologie obrábění a zjednodušení programu pro obsluhu.



Obr. 13 – příruba s rádiusem

4.2 Materiál součásti

Když se podíváme na funkci a použití této příruby zjistíme, že se bude obrábět a následně svařovat. Proto bychom měli volit materiál, který má dobrou obrobitelnost i svařitelnost. V přírubě se nejdříve vyfrézuje rádius o velikosti průměru válcové plochy, na kterou se příruba následně přivaří. Ve Streicheru používají k výrobě této příruby několik různých materiálů. Od nelegovaných ocelí (11 373), přes konstrukční oceli (11 523) až po příruby z nerezových ocelí (1.4301, 1.4571), vše závisí na materiálech použitých v daných sestavách, kde bude příruba použita.

4.2.1 Konstrukční ocel 11 523

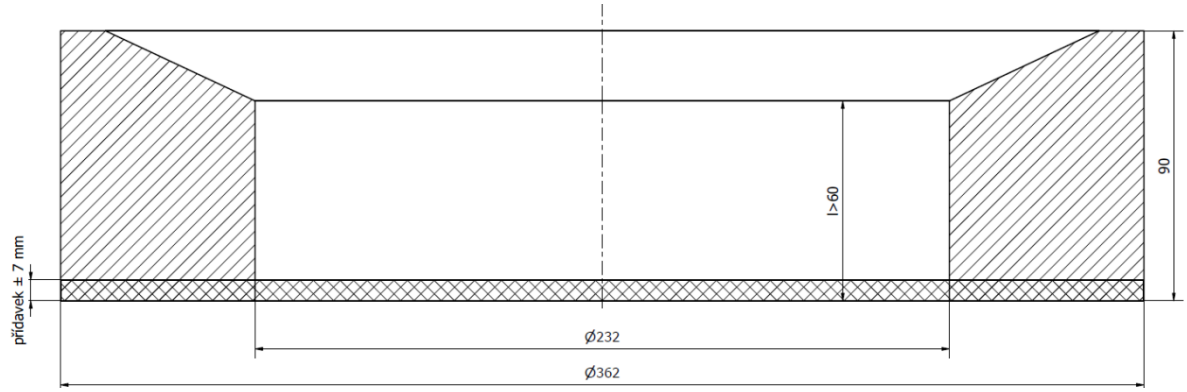
Tuto ocel lze použít k mostním a jiným svařovaným konstrukcím, vyrábí se z ní ohýbané profily, svařované části automobilů a součásti tlakových nádob. Je vhodná ke svařování a má dobrou obrobitelnost (14 b). Vzhledem k dobré obrobitelnosti lze využít základní řezné nástroje pro oceli a doporučené řezné podmínky od výrobce nástroje.

4.2.2 Nerezová austenitická chrom-niklová ocel 1.4571

Výhodou této oceli je její vysoká odolnost proti korozi a zároveň velmi dobrá svařitelnost. Hlavní využití je v oblastech s vyšším korozním zatížením, a to zejména potravinářský průmysl, ropný průmysl, loďařství, chemický průmysl a zdravotní technika. Obrobitelnost tohoto materiálu je horší, proto se musí optimalizovat řezné podmínky a musí se zvolit vhodné řezné nástroje pro dosažení požadovaných vlastností obrobeného povrchu.

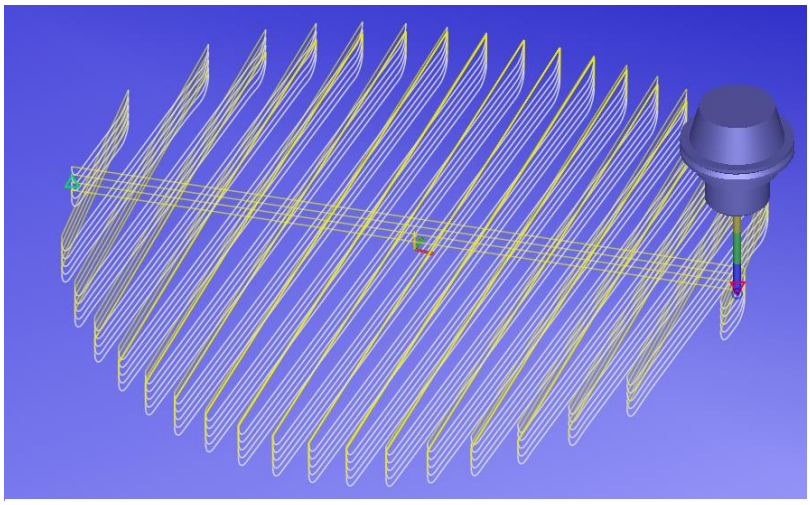
4.3 Návrh technologie frézování součásti

Pro navržení vhodného nástroje pro obrábění zadané součásti je potřeba navrhnout technologii, se kterou budeme součást obrábět. Abychom byli schopni navrhnout technologii, je důležité znát polotovar, který v tomto případě má podobný tvar jako hotový obrobek, pouze má přídavky na obrobení potřebného tvaru. Polotovar má i přídavky na spodní části, obráběním těchto přídavků se však parametrický program nezaobírá. Rozměry a tvar polotovaru lze vidět na Obr.14.



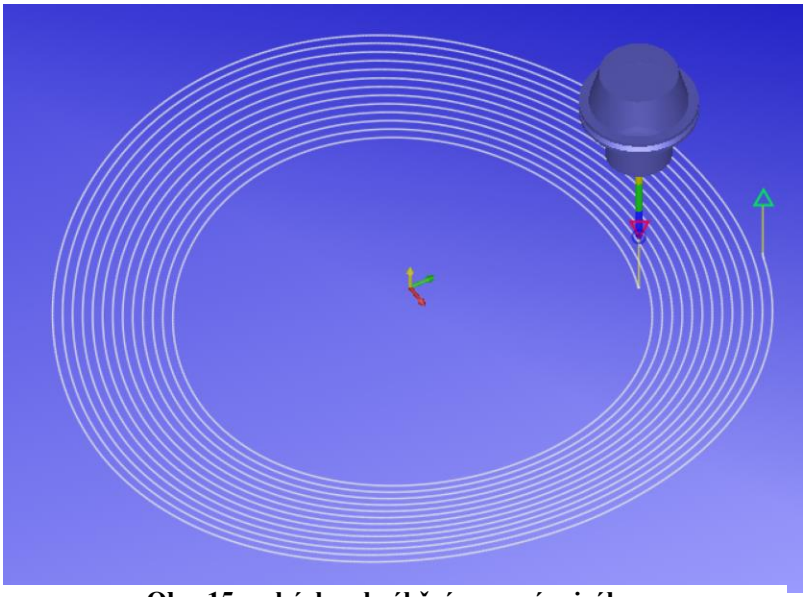
Obr. 14 – polotovar příruby

Pro zadanou součást lze navrhnout několik různých variant tegnologie obrábění. Jednou z nich je podélné řádkování čelní frézou. U této technologie využíváme větší boční třísku, cca 60% – 70% průměru nástroje, ale menší krok do hlouky. Nevýhodou takto zvolené technologie je čas přejezdů, který je zde navíc, jedná se o nájezdy a odjezdy při každém řezu.



Obr. 16 – ukázka obrábění drážkováním

Další technologie je frézování po spirále. Vzhledem ktomu jaký nástroj bychom měli k dispozici, bychom volili potřebnou hloubku řezu. Pokud bychom měli nástroj s malou hloubkou řezu museli bychom zvolit frézování na více třísek do hloubky. Pokud by však byla hloubka nástroje dostatečná a nástroj by umožňoval obrábění na plnou hloubku s malým bočním krokem, lze obrábět pouze na jednu hloubku.



Obr. 15 – ukázka obrábění pomocí spirály

Alternativou může být frézování bokem nástroje s otočením vřetena s nástrojem o 90°. Bohužel má tato technologii nevýhodu v nutnosti volby dlouhého nástroje, což by mohlo vést k rozvibrobání nástoje a následnému horšímu obrobenému povrchu.

4.4 Nástroj pro obrábění

Při volbě nástroje se zaměříme na obráběný materiál a na zvolenou technologii obrábění, která spočívá v obrábění drážkováním v jedné ose nebo v souvislém obrábění ve 3 osách po spirále. Dále musíme přihlédnout k nástrojovému vybavení firmy Streicher a možnostem strojů, na kterých se součást bude obrábět. V rámci požadavků firmy jde o minimalizaci počtu nástrojů na jeden, maximálně dva s ohledem na tvar polotovaru a na počet obráběcích operací, viz následující kapitoly. Cílem pro firmu Streicher je, aby měli sadu nástrojů vždy připravenou pro obrábění tohoto typu výrobku.

Firma Streicher využívá k obrábění těchto součástí válcovou frézu s vyměnitelnými břitovými destičkami ve šroubovici ø32 - R217.69-1632.RE-034-10.4A. Výhodou je větší množství obvodových břitů, které přispívají pro stabilnější řez při hrubování. Maximální hloubka řezu je 34 mm, která je více než dostačující.[8]

Pokud budeme uvažovat plný polotovar, je třeba obrobit nebo vrtat díru pro následnou technologii frézování rádiusu. V tomto případě přichází v úvahu předvrtání nebo zafrézování po šroubovici. U případného předvrtání bychom potřebovali vrták větší než ø32 mm nebo menší vrták a menší frézu pro rozfrézování díry na dostatečný průměr pro frézu ø32 mm, kterou se následně obrábí celý rádius. Nevýhoda tohoto postupu je horší vyrobitelnost velké díry vrtákem a již větší množství nástrojů. Proto se spíše zaměřím na rozfrézování díry menší frézou po šroubovici.

	Maximální hloubka řezu[]	34 mm
	Obráběcí průměr	32 mm
	Počet břitů	4
	Celkový počet břitů	16
	Typ upnutí	Combimaster
	Řezná rychlost	117–154 m/min
	Posuv	0,0635-0,126 mm/zub
	Typ destiček	XOMX10T308TR-M09 F40M

Tab. 8 – Definice řezného nástroje [8]

4.5 Polotovar

Při výrobě zadané typové součásti využívá firma Streicher výhradně 2 typy polotovarů. Tvar polotovaru závisí na výsledném tvaru a funkci obrobené příruby. Jeden z typů, který se vyskytuje častěji, je již dutá příruba s přídavky pro následné obrábění. U tohoto polotovaru je menší úběr materiálu právě z důvodu přídavků na obrábění. U druhého typu polotovaru se jedná o plnou přírubu, proto je zde větší úběr materiálu.

4.6 Analýza současného stavu

Po domluvě firma poskytla aktuálně používaný parametrický program, který se v současné době využívá k obrábění příruby. V současnosti Streicher využívá pro obrábění pouze 2 stroje s řídícím systémem Heidenhain, proto jsem dostal pouze program pro tento řídící systém. Ovšem požadavek firmy není pouze optimalizace stávajícího programu, ale také vytvoření nového programu pro řídící systém Sinumerik 840D.

Firma disponuje dvěma stroji s řídícím systémem Heidenhain, které mají ale odlišné verze, jeden stroj pracuje s iTNC 530 a druhý s iTNC 620. V tomto případě se však nejedná o rozdíly v používání parametrů a logických funkcí, takže program pro řídící systém Heidenhain bude pouze jeden pro oba stroje stejný.

Program PRSUD-SPIRALOU.H (Příloha 1) je obrábění válcovou frézou po spirále nedefinovanou pomocí parametrického programování. Jako první je začátek programu pro dodržení syntaxe, dále je nadefinován polotovar a hlavička programu firmy Streicher. V hlavičce najdeme polohu nulového bodu pro informai obsluhy, kam má nulový bod nastavit, poté jsou zde vidět nástorje, které bude potřebovat pro obrábění daného dílu. Všechny položky jsou doplněny komentářem pro lepší srozumitelnost programu.

```
0 BEGIN PGM PRSUD-SPIRALOU MM
1 BLK FORM 0.1 Z X-180 Y-180 Z+0
2 BLK FORM 0.2  X+180  Y+180  Z+83.1
3 ; NAZEV : FREZOVANI RADIUSU SUDU DO PRIRUBY SPIRALOU
4 ; --------------------------
5 * - NULOVY BOD
6 ; X0 - OSA PRIRUBY
7 ; Y0 - OSA PRIRUBY
8 ; Z0 - SPODNI PLOCHA PRIRUBY
9 ; --------------------------
10 * - DEFINICE NASTROJU
11 LBL 30
12 ;
13 FN 0: Q51 =+15 ;JEZEK NEBO KULOVKA
14 ;
15 FN 0: Q52 =+2 ;
16 ;
17 LBL 0
18 ; --------------------------
```

Definice polotovaru

Poloha nulového bodu

Definice nástrojů

Obr. 17 – hlavička programu

```
19 * - NACISTO TVAR
20 ;JEZEK D32
21 CALL LBL 30
22 TOOL CALL Q51 Z S1930 F1100
23 L  Z+200 R0 FMAX M3 M18
```

Obr. 18 – volba nástroje a řezných podmínek

Po nadefinování nástroje a jeho korekcí se nástroj zavolá, nadefinují se řezné podmínky, najede do výchozího bodu a zapnou se otáčky popřípadě chlazení nástroje.

V další části programu jsou nadefinovány všechny vstupní parametry. Tyto parametry se odvíjí z výkresu součásti a jsou nadefinovány tak, aby bylo možné je jednoduše měnit v závislosti na dodaném polotovaru a požadovaném rozměru kusu.

```
24 FN 0: Q1 =+710 ; FREZOVANY RADIUS SUDU
25 FN 0: Q2 =+60 ; ROZMER KUSU OD SPODNI HRANY K RADIUSU SUDU
26 FN 0: Q3 =+1 ; UHLOVY KROK VE ST
27 FN 0: Q4 =+232 ; OBRABENY VNITRNI PRUMER PRIRUBY
28 FN 0: Q5 =+362 ; OBRABENY VNEJSI PRUMER PRIRUBY
29 FN 0: Q6 =+1 ; KROK NA RADIUSU V MM (SPONA DO BOKU)
```

Obr. 19 – definice parametrů

Poté následuje vyvolání podprogramu, který součást obrobí po spirále a konec hlavního programu, po kterém budou následovat podprogramy.

```
30 CALL LBL 1
31 ; --------------------------
32 M2
```

Obr. 20 – volání podprogramu a konec hlavního programu

Podprogram LBL 1 se skládá z posunutí nulového bodu, výpočtů parametrů pro snadnější počítání v rovnicích a podprogram v podprogramu LBL 122. Součástí LBL 122 jsou početní funkce, které pomocí programového skoku dopočítávají v každém kroku koncové souřadnice lineární interpolace až po splnění podmínky.

Posunutí středu radiusu v ose Z je zde použito z důvodu snadnějšího programování a dosazování hodnot přímo z výkresu. Je to výhoda pro programátora, který může zadávat hodnoty přímo z výkresu a nemusí nic sám dopočítávat. Otáčením si srovnáme nulový bod.

```
37 LBL 1
38 Q40 = Q1 + Q2 ; POSUNUTI STREDU RADIUSU
39 CYCL DEF 7.0 NULOVY BOD
40 CYCL DEF 7.1 IZ+Q40
41 CYCL DEF 10.0 OTACENI
42 CYCL DEF 10.1  ROT+0
```

Obr. 21 – začátek podprogramu a posunutí nulového bodu

Výpočet Q-parametrů pro zjištění potřebných hodnot pro následné dosazení do rovnic pro výpočet drah nástroje. Tímto výpočtem se zjednodušší následný výpočet a rovnice tak budou jednodušší. V tomto případě by však stačilo tyto výpočty nechat v hlavním programu, což by vedlo i k jednodušší kontrole a zjednodušení celého podprogramu.

```
44 Q8 = Q4 / 2 - Q6 ; OBRABENY RADIUS
45 Q7 = Q5 / 2 + ( Q6 * 2 ) ; KONCOVY RADIUS
46 Q8 = Q8 + Q6 ; OBRABENY RADIUS
47 FN 0: Q9 =+0 ; VYCHOZI UHEL
```

Obr. 22 – okrajové podmínky obrábění

Jednou z dalších věcí, které lze zoptimalizovat, je odstranění opakujících se bloků, které jsou ovšem zbytečně používané vícekrát. Po odstranění bloků 47-49 zbydou pouze bloky, které jsou v podprogramu, který se bude cyklyt, tedy bloky 58-60.

Vzhledem k tomu, že se jedná o obrábění po krátkých lineárních interpolacích definovaných v podprogramu LBL 122, je zbytečné definovat počátek polárních souřadnic a přímku LP, definovanou pomocí polárních souřadnic v tomto místě. Na průběh programu to nemá žádný vliv.

```
47 Q10 = Q8 * SIN ( Q9 - 90 )
48 Q11 = SQRT ( SQ Q1 - SQ Q10 )
49 Q19 = - Q11
50 L  X+0  Y+0 R0 FMAX
51 L  Z+Q19 R0 FMAX
52 CC  X+0  Y+0
53 LP  PR+Q8  PA+Q9 RL F AUTO
58 Q10 = Q8 * SIN ( Q9 - 90 )
59 Q11 = SQRT ( SQ Q1 - SQ Q10 )
60 Q19 = - Q11
```

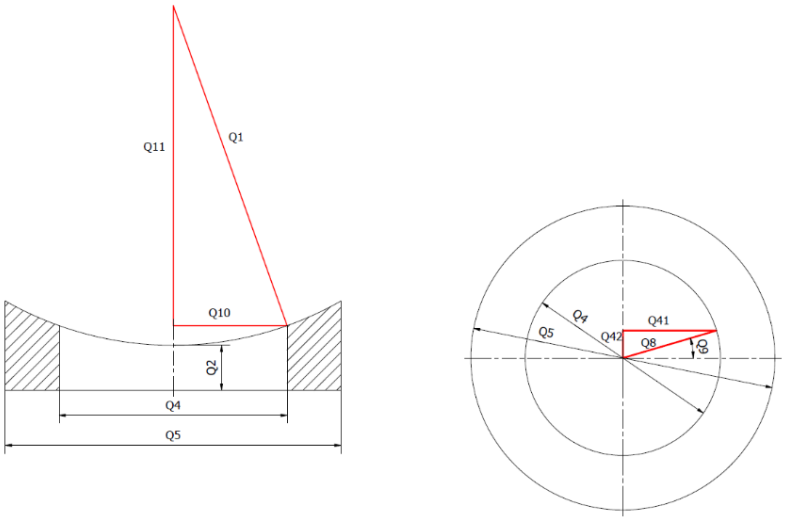
Obr. 23 – ukázka opakujících se bloků

LBL 122 obsahuje výpočty přímo pro dráhy nástroje. Tyto hodnoty se mění pokaždé, co se podprogram zacyklí přes programový skok. Výpočty jsou vytvořeny srozumitelně a jednoduše, pouze by bylo dobré sjednotit Q10 a Q41 vzhledem k tomu, že jedná o stejný rozměr. Zjednodušení by se dalo dosáhnout i přesunutím znaménka mínus z bloku 60 již do bloku 59.

```
54 LBL 122
55 Q8 = Q8 + ( Q6 / 360 * Q3 ) ; OBRABENY RADIUS
56 Q9 = Q9 + Q3
57 Q10 = Q8 * SIN ( Q9 - 90 )
58 Q11 = SQRT ( SQ Q1 - SQ Q10 )
59 Q19 = - Q11
60 Q41 = COS Q9 * Q8 ; X-OVA SOURADNICE
61 Q42 = SIN Q9 * Q8 ; Y-OVA SOURADNICE
```

Obr. 24 – vnořený podprogram LBL122

V současném stavu je program napsán pomocí lineární interpolace a mění se po krátkých vzdálenostech, které jsou rozpočteny po kruhu v rozteči po 1° (parametr Q3), tedy 360 drah za jedno otočení kolem celé osy. Dále se každou otočku zvětšuje obráběný rádius o boční krok nástroje, který je nadefinovaný pomocí parametru Q6. Pro výpočet linárních souřadnic jsou využívány základní goniometrické funkce. Na obrázku níže je vidět podle kterých funkcí a náčrtů jsou výpočty odvozovány.



Obr. 26 – náčrt výpočtu souřadnic

```
62 L  X+Q41  Y+Q42  Z+Q19 RL F AUTO
63 FN 12: IF +Q8 LT +Q7 GOTO LBL 122
64 L  X+0  Y+0 R0 FMAX
65 CYCL DEF 10.0 OTACENI
66 CYCL DEF 10.1  ROT+0
67 CYCL DEF 7.0 NULOVY BOD
68 CYCL DEF 7.1 IZ-Q40
69 L  Z+200 R0 FMAX
70 LBL 0
71 ; ------------------------
72 END PGM PRSUD-SPIRALOU MM
```

Obr. 25 – dráhová funkce zadána parametricky a logická funkce tvořící smyčku

Po zadefinování potřebných parametrů jsou parametry dráhy nástroje vloženy do funkce lineární interpolace, kde je zvolena levá korekce nástroje a automatický posuv. Další funkcí je podmíněný programový skok, který určuje do jaké hodnoty se bude program cyklovat. Nakonec je odjezd na definiované místo, posunutí Z-ové souřadnice zpět na nulu a ukončení podprogramu a celého programu.

Po prostudování programu vyšlo najevo, že jsou zde místa pro zlepšení, jak z hlediska technologického, například změna typu dráhy z lineární na kruhovou interpolaci, nebo volba vhodnější funkce pro výpočet drah nástroje, tak z hlediska přehlednosti a zjednodušení programu, kde se opakují a vyskytují bloky, které by zde vůbec nemusely být.

Při zkoumání současného stavu této problematiky ve firmě Streicher se zhodnotil současný program pro obrábění na strojích s řídícím systémem Heidenhain, ovšem náplní práce je tvorba programu i pro řídící systém Sinumerik, který ovšem firma nemá. Tudíž se nebude jednat o optimalizeci stávajícího programu, ale o tvorbu a odladění nového.

5. Návrh, tvorba a odladění optimalizovaného programu pro zadanou součást

Při návrhu nového programu se zaměřím na zakomponování vrtacího nebo frézovacího cyklu pro požadavek na plný polotovar. Vzhledem k tomu, že tento cyklus v současném program úplně chybí, ale i tento typ součástí se obrábí, musela se vždy přidat do programu operace pro předfrézování/vrtání, což už komplikovalo práci obsluze. Ovšem požadavek operátorů stroje byl ve sjednocení programu a možnosti využití přípravného cyklu, který bude stálou součástí programu, pouze se bude volat pomocí nadefinováním parametrů.

Dalším návrhem bude asi změna z lineárních pohybů na dráhy kruhové pro lepší optimalizaci dráhy nástroje. Záměna lineárních drah by mohla být i za jednu funkci popsanou parametry, ovšem není jisté, zda se podaří tuto funkci nadefinovat.

5.1 Zjednodušení stávajícího programu

Když vezmeme stávající program a pouze ho zjednodušíme, tzn. upravíme nebo odstraníme nevyhovující bloky, dostaneme optimalizovanou verzi používaného programu. Poté přidáme frézovací cyklus s podmínkou, zda se cyklus vyskytuje či nikoliv, a to v závislosti na tom, zda je polotovar dutý či nikoliv, viz parametr Q4, popř. R4 a odladíme program v simulaci na počítači. Tato varianta stále počítá s lineární interpolací dráhy nástroje pomocí výpočtů souřadnic viz. *Obr. 26.: Náčrt výpočetních souřadnic*. Přidaly se další parametry potřebné pro správnou funkčnost programu s předfrézovacím cyklem a nadefinoval se cyklus frézování kruhové kapsy, který proběhne před frézováním po spirále v případě plného polotovaru.

```
0 BEGIN PGM 1 MM
1 BLK FORM 0.1 Z  X-150  Y-150  Z+0
2 BLK FORM 0.2  X+150  Y+150  Z+70
3 FN 0: Q1 =+300 ; RADIUS FREZOVANEHO SUDU
4 FN 0: Q2 =+40 ; ROZMER PRIRUBY OD SPODNI HRANY
5 FN 0: Q3 =+1  ; UHLOVY KROK VE ST
6 FN 0: Q4 =+0  ; VNITRNI PRUMER PRIRUBY
7 FN 0: Q5 =+300 ; VNEJSI PRUMER PRIRUBY
8 FN 0: Q6 =+1  ; RADIALNI KROK
9 FN 0: Q10 =+70 ; CELKOVA VYSKA KUSU
10 LBL 3
11 Q7 = Q5 / 2
12 Q8 = Q4 / 2
13 FN 0: Q9 = 0 ; VYCHOZI UHEL
14 Q12 = Q2 + 7 ; PRIPOCITANI PRIDAVKU SPODNI PLOCHY 7MM
15 Q40 = Q1 + Q12 ; POSUNUTI STREDU RADIUSU
```

Obr. 27 – zjednodušená hlavička programu

Všechny parametry a potřebné výpočty, které mění obsluha, ale v průběhu programu se nemění se seřadily na začátek programu pro lepší přehlednost, přidal se parametr celkové výšky polotovaru a zohlednil se ve výpočtu přídavek na spodní plochu, se kterým musela dříva obsluha počítat už při zadávání parametrů z výkresu. Velikost tohoto přídavku je 7 mm. Cíl skoku LBL 3 odkazuje na návrat z podprogramu s frézovacím cyklem a program pokračuje frézováním rádiusu.

Posunutí nulového bodu v ose Z zde zůstalo právě pro zjednodušení práce s výpočty, které využívají přímo hodnoty z výkresu bez nutnosti dopočítávání dalších hodnot. Probíhá zde výměna nástroje, nastavení řezných podmínek a vyvolání podprogramu pro frézování rádiusu. Nakonec je tady zpětné posunutí nulového bodu a konec hlavního programu s odjezdem na definované místo pro možnost odebrání hotového výrobku.

```
16 CYCL DEF 7.0 NULOVY BOD
17 CYCL DEF 7.1  Z+Q40
18 L  X+200  Y+200  Z+200 R0 FMAX
19 TOOL CALL 52 Z S1200 F250
20 L  X+0  Y-7.5  Z+100 R0 FMAX
21 FN 12: IF +Q4 LT +50 GOTO LBL 2
22 CALL LBL 1
23 CYCL DEF 7.0 NULOVY BOD
24 CYCL DEF 7.1  Z-Q40
25 L  Z+200 R0 FMAX M30
26 LBL 1
27 Q8 = Q8 + ( Q3 * Q6 ) / 360
28 Q9 = Q9 - 1
29 Q41 = Q8 * COS Q9 ; X-OVA SOURADNICE
30 Q42 = Q8 * SIN Q9 ; Y-OVA SOURADNICE
31 Q43 = - ( SQRT ( SQ Q1 - SQ Q41 ) ) ; Z-OVA SOURADNICE
32 L  X+Q41  Y+Q42  Z+Q43 RL FMAX M3
33 FN 12: IF +Q8 LT +Q7 GOTO LBL 1
34 LBL 0
```

Obr. 29 – posunutí nulového bodu a začátak podprogramu LBL1

V samotném podprogramu jsou výpočty měnící svoji hodnotu pokaždé, co se program zacyklí pomocí programového skoku IF. Objevují se tam parametry poloměru (Q8), úhlu natočení (Q9) a výpočty souřadnic X, Y, Z. Další částí je lineární interpolace, podle které se nástroj pohybuje.

```
36 LBL 2
37 L  Z+200 R0 FMAX
38 TOOL CALL 10 Z S1200 F2500
39 Q60 = Q40 - Q10
40 CYCL DEF 252 KRUHOVA KAPSA ~
    Q215=+0     ;ZPUSOB OBRABENI ~
    Q223=+50    ;PRUMER KRUHU ~
    Q368=+0     ;PRIDAVEK PRO STRANU ~
    Q207=+500   ;FREZOVACI POSUV ~
    Q351=+1     ;ZPUSOB FREZOVANI ~
    Q201=-Q11   ;HLOUBKA ~
    Q202=+5     ;HLOUBKA PRISUVU ~
    Q369=+0     ;PRIDAVEK PRO DNO ~
    Q206=+150   ;POSUV NA HLOUBKU ~
    Q338=+0     ;PRISUV NA CISTO ~
    Q200=+2     ;BEZPECNOSTNI VZDAL. ~
    Q203=-Q60   ;SOURADNICE POVRCHU ~
    Q204=+50    ;2. BEZPEC.VZDALENOST ~
    Q370=+1     ;PREKRYTI DRAHY NAST. ~
    Q366=+1     ;ZANOROVANI ~
    Q385=+500   ;POSUV NACISTO
41 L  X+0  Y+0  Z+0 R0 FMAX M3
42 CYCL CALL
43 Q4 = Q223
44 CALL LBL 3
45 LBL 0
46 END PGM 1 MM
```

Obr. 28 – podprogram s cyklem pro frézování kruhové kapsy

Podprogram LBL 2 obsahuje frézovací cyklus pro kruhovou kapsu sloužící pro přípravu plného polotovaru pro náslené zavolání cyklu LBL 1. Tento cyklus je nadefinován pevnými hodnotymi kromě hloubky kapsy a souřadnic povrchu, které jsou nadefinovány pomocí parametrů pro dosažení potřebné hloubky. Průměr předfrézované kapsy Q223 je nasteven na 50 mm, což je dostačující pro frézu průměr 32 mm, která je zvolena pro obrábění tvaru. Po vyfrézování program pokračuje v místě skoku LBL 3 a je tutíž potřeba aktualizovat parametr vnitřního průměru příruby Q4, který již není roven 0, ale rovná se průměru kruhové kapsy, tedy 50 mm. Skok LBL 3 musí být v programu nadefinován na vhoném místě pro správnou funkčnost zbylé části programu.

Dále se změnil nástroj, kterým lze zavrtávat do plného materiálu. Nástroj nadefinovaný pro obrábění bokem nástroje po spirále na tuto operaci není vhodný, protože nemá žádnou řeznou hranu přes střed nástroje. Zvolen byl nástroj ⌀20 - R217.69-2020.3-025-10.1A, který umožnuje zavrtání po šroubovici do plného materiálu.

Maximální hloubka řezu	25 mm
Obráběcí průměr	20 mm
Počet břitů	2
Celkový počet břitů	4
Typ upnutí	Combimaster
Řezná rychlost	156–212 m/min
Posuv	0,0573-0,114 mm/zub
Typ destiček	XOMX10T308TR-M09 F40M

Tab. 9 – Definice řezného nástroje [7]

Po zjednodušení programu je vidět, že se program zkrátil i přesto, že se doplnily další požadované operace a přitom technologie obrábění rádiusu zůstala podobná. Přidán byl cyklus frézování kruhové kapsy pro případné obrábění plného polotovaru a byl odladěn vůči tvaru součásti a průběhu programu. Neměnily se parametry definované obluhou stroje, pouze se některé další přidaly, např. celková výška polotovaru a zakomponování přídavku spodní plochy do výpočtů. Skoky ani výpočetní funkce se také neměnily, pouze se program naformátoval do přehlednější formy. V další části práce se budu zaobírat návrhem programu v řídícím systému Sinumerik a změnou na dráhy s kruhovou interpolací.

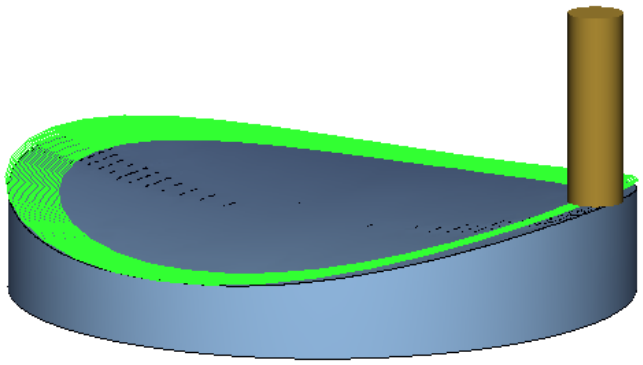
Vzhledem k tomu, že se tato součást bude vyrábět na více strojích s jiným řídícím systémem, je třeba vytvořit tento parametrický program v Sinumeriku 840D, jelikož na ostatních strojích ve firmě Streicher disponují tímto řídícím systémem. Program pro řídící systém Sinumerik je velmi podobný tomu v Heidenhainu, ovšem pouze obsahově. Parametry v Simuneriku jsou používány jinak, Q parametry používané Heidenhainem jsou zde nahrazeny R parametry. Sinumerik využívá také jinak definované podprogramy a cykly.

```
WORKPIECE(,"",,"CYLINDER",64,0,70,-80,360)
G54 G90 G17
R1=525 ; RADIUS FREZOVANEHO SUDU
R2=40 ; ROZMER PRIRUBY OD SPODNI HRANY
R3=1 ; UHLOVY KROK VE ST
R4=0 ; VNITRNI PRUMER PRIRUBY
R5=360 ; VNEJSI PRUMER PRIRUBY
R6=10 ; RADIALNI KROK
R10=70 ; CELKOVA VYSKA SUDU
```

Obr. 31 – definice parametrů v Sinumeriku

Definice parametrů zůstala stejná, akorát se změnily Q parametry na R paramtry. Vzhledem k tomu, že se programovalo v ISO kódu, je potřeba nadefinovat začátek programu pomocí základních G funkcí, jako je v tomto případě nastavení nulového bodu G54, nastavení absolutního programování a výběr roviny, ve které chceme obrábět.

```
LABEL3:
G00 X200 Y200 Z200
T="JEŽEK 32" M6
G94 S1500 F5000
R7=R5/2+32
R8=R4/2-16
R9=0 ; VYCHOZI UHEL
R11=R10-R2
R40=R1+R2 ; POSUNUTI NULOVEHO BODU
TRANS Z=R40
IF R4<50 GOTOF LABEL2
G00 X0 Y0 Z=-R1+10
LABEL1:
R8=R8+(R6/360*R3)
R9=R9+R3
R10=R8*SIN(R9-90)
R11=-SQRT(R1*R1-R10*R10)
R41=COS(R9)*R8
R42=SIN(R9)*R8
G01 X=R41 Y=R42 Z=R11
IF R8<R7 GOTOB LABEL1
LABEL0:
M30
LABEL2:
G00 X200 Y200 Z200
T="FRÉZA D20" M6
G94 S1500 F5000
R60=R40-R10
R61=50
G00 X0 Y0 Z0 M3
POCKET4(100,-R60,1,-R11,R61,0,0,2.5,0,0,200,0.1,0,1011,0.5,9,15,
0,2,0,1,2,10100,111,101)
R4=R61
GOTOB LABEL3
LABEL0:
```



Obr. 30 – parametrický program s ukázkou drah za simulace

Zbytek programu je stejný jako u řídícího systému Heidenhain kromě používaných funkcí, které má každý systém jiné. Jediný problém byl u kruhové kapsy POCKET4, kde nejde použít nadefinovaný parametr průměru díry dále v programu, proto se musel vytvořit nový parametr R61, aby se dal aktualizovat vnitřní průměr příruby R4.

5.2 Návrh programu pomocí kruhové interpolace

Druhou variantou návrhu programu je změna dráhových pohybů z linární na kruhovou interpolaci. Vzhledem k tomu, že se jedná o obrábění kruhové součásti po spirálové dráze, je výhodnější obrábět pomocí kruhové interpolace nežli pomocí lineární interpolace. Dále je snaha o zvětšení úhlového kroku, kde v současném stavu je dráha rozpočítána po 1° po celé délce křivky. Podle symetrie součásti by bylo výhodné dostat úhlový krok např. na 90°.

Varianta kruhové interpolace v Heidenhainu spočívá v zadefinování středu kruhu pomocí funkce CC, který je [0;0]. Následně se vytvoří tečný radius pomocí polárních parametrů CTP, který nastaví výchozí bod spirály, tedy počáteční průměr a úhel natočení. Poté se nastaví další rádius, tentokrát už ne tečný, ale také pomocí polárních souřadnic CP a nastaví se přes něj úhlový krok a směr otáčení spirály, tedy směr obrábění. Poté se podprogram pouze zacykluje programovým skokem do nastavené meze.

```
0  BEGIN PGM 3 MM
1  BLK FORM 0.1 Z  X-150  Y-150  Z+0
2  BLK FORM 0.2  X+150  Y+150  Z+70
3  FN 0: Q3 =+1 ; UHLOVY KROK VE ST
4  FN 0: Q4 =+80 ; VNITRNI PRUMER PRIRUBY
5  FN 0: Q5 =+300 ; VNEJSI PRUMER PRIRUBY
6  FN 0: Q6 =+1 ; RADIALNI KROK
7  LBL 3
8  Q7 = Q5 / 2
9  Q8 = Q4 / 2 - 5
10 FN 0: Q9 =+0 ; VYCHOZI UHEL
```

Obr. 32 – hlavička programu s definicí parametrů

Definice parametrů součásti zůstává stejná, pouze se počet paramentrů zmenšil kvůli jejich nevyužitelnosti v rámci této technologie. Lze si všimnout, že se zachovaly pouze parametry, které souvisí pouze s 2D průmětem spirály. Jsou to vnitřní a vnější průměr, radiální a úhlový krok. To znamená, že tato technologie se dá použít pouze pro obrábění ve dvou osách, tedy ji nelze využít na zadanou součást.

```
11 L  Z+200 R0 FMAX
12 TOOL CALL 16 Z S1200 F2500
13 FN 12: IF +Q4 LT +50 GOTO LBL 2
14 L  X+0  Y+0  Z+65 RL FMAX
15 CALL LBL 1
16 L  Z+200 R0 FMAX M30
```

Obr. 33 – volba nástroje a volání podprogramu

Konec hlavního programu je stejný jako u předchozí operace, tudíž nemá smysl ho tady znovu rozebírat (viz. začátek strany 26).

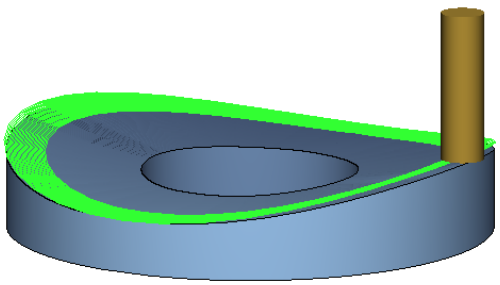
V podprogramu LBL 1 lze vidět nastavení 2D spirály pomocí dvou kruhových interpolací s polárním zadáváním souřadnic a zadefinovaným středem, který je u této funkce potřeba pro zjištění roviny obrábění. Tyto funkce jsou však pouze pro řídící systém Heidenhain, proto se u návrhu programu pro řídící systém Sinumerik bude potřeba zaměřit na jinou metodu. Podprogram LBL 2 je stejný jako u předchozího typu obrábění, pouze jsou hloubka a souřadnice povrchu nadefinovány přímo číselnou hodnotou a ne pomocí parametrů. Závěr k této technologii je takový, že se jedná o typ obrábění dílů kruhového průřezu po spirále ovšem pouze ve dvou osách.

```
17 LBL 1
18 Q9 = Q9 + 1
19 Q8 = Q8 + 5
20 CC  X+0  Y+0
21 CTP  PR+Q8  PA+0
22 CP IPA+90 DR+
23 FN 12: IF +Q8 LT +Q7 GOTO LBL 1
24 LBL 0
25 LBL 2
26 L  Z+200 R0 FMAX
27 TOOL CALL 10 Z S1200 F2500
28 CYCL DEF 252 KRUHOVA KAPSA ~
    Q215=+0     ;ZPUSOB OBRABENI ~
    Q223=+80    ;PRUMER KRUHU ~
    Q368=+0     ;PRIDAVEK PRO STRANU ~
    Q207=+500   ;FREZOVACI POSUV ~
    Q351=+1     ;ZPUSOB FREZOVANI ~
    Q201=-5     ;HLOUBKA ~
    Q202=+5     ;HLOUBKA PRISUVU ~
    Q369=+0     ;PRIDAVEK PRO DNO ~
    Q206=+150   ;POSUV NA HLOUBKU ~
    Q338=+0     ;PRISUV NA CISTO ~
    Q200=+2     ;BEZPECNOSTNI VZDAL. ~
    Q203=+70    ;SOURADNICE POVRCHU ~
    Q204=+50    ;2. BEZPEC.VZDALENOST ~
    Q370=+1     ;PREKRYTI DRAHY NAST. ~
    Q366=+1     ;ZANOROVANI ~
    Q385=+500   ;POSUV NACISTO
29 L  X+0  Y+0 R0 FMAX M3
30 CYCL CALL
31 Q4 = Q223
32 CALL LBL 3
33 LBL 0
34 END PGM 3 MM
```

Obr. 35 – podprogram LBL1 a LBL2

Při návrhu programu s kruhovou interpolací v řídícím systému Sinumerik se jednalo o nalezení potřebné funkce, podle které by bylo možné součást obrobit, vzhledem k tomu, že se zde nenachází kruhové interpolace CTP a CP. Jedním řešením je tvorba spirály pomocí nadstavby dílenského programování, a to frézování po kontuře. V kontuře se vytvoří spirála a poté se vyvolá fukce frézování po kontuře, která spirálu obrobí. To by ovšem nefungovalo pro případ zadané příruby, protože nevíme kolik otoček spirála bude mít a bylo by na dlouho takovou konturu nakreslit. Navíc by nebylo možné takto vytvořený program zparametrizovat.

```
WORKPIECE(,"",,"PIPE",64,0,70,-80,360)
G54 G90 G17
R1=525 ; RADIUS FREZOVANEHO SUDU
R2=40 ; ROZMER PRIRUBY OD SPODNI HRANY
R3=10 ; UHLOVY KROK VE ST
R4=0 ; VNITRNI PRUMER PRIRUBY
R5=360 ; VNEJSI PRUMER PRIRUBY
R6=1 ; RADIALNI KROK
LABEL3:
G00 X200 Y200 Z200
T="JEŽEK 32" M6
G94 S1500 F5000
R7=R5/2+32
R8=R4/2-16
R9=0 ; VYCHOZI UHEL
R10=70 ; CELKOVA VYSKA SUDU
R11=R10-R2
R40=R1+R2 ; POSUNUTI NULOVEHO BODU
TRANS Z=R40
IF R4<50 GOTOF LABEL2
G00 X0 Y0 Z=-R1+10
```



Obr. 34 – program s kruhovou interpolací v Sinumeriku s ukázkou drah za simulace

Z tohoto důvodu jsem se zaměřil na kruhovou interpolaci G2 a G3, kterou lze definovat pomocí souřadnic X, Y a Z společně s interpolačními parametry I, J a K nebo pomocí rádiusu R. V tomto případě je výhoda, že výpočty pro koncové body kruhu jsou stejné jako u lineární interpolace, tutíž pouze stačí zjistit parametry I, J, K nebo rádius R. Ovšem všechny hodnoty musí být snadno spočítatelné a zparametrizovatelné. Na jedné webové stránce jsem dohledal, že lze využít kruhovou interpolaci G2 nebo G3 společně se souřadnicemi koncových bodů [X;Y] a s využitím CR, kde výpočet vypadal následovně: [10]

R41=COS(R9)*R8

R42=SIN(R9)*R8

R43=SQRT(R41*R41+R42*R42)

G02 X=R41 Y=R42 CR=R43

```
LABEL1:
R8=R8+(R6/360*R3)
R9=R9+R3
R10=R8*SIN(R9-90)
R11=-SQRT(R1*R1-R10*R10)
R41=COS(R9)*R8
R42=SIN(R9)*R8
R43=SQRT(R41*R41+R42*R42+R11*R11)
G03 X=R41 Y=R42 Z=R11 CR=R43
IF R8<R7 GOTOB LABEL1
LABEL0:
```

Obr. 36 – ukázka výpočtů v programu

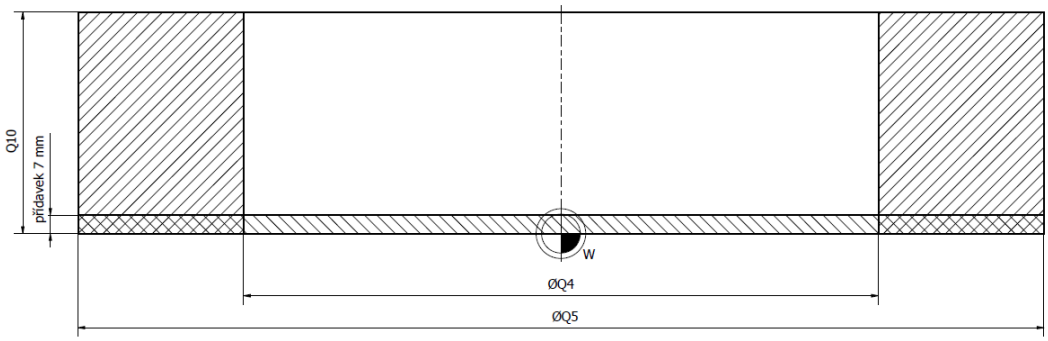
Podprogram LABEL1 obsahuje obrábění rádiusu do polotovaru po spirále, dráhami s kruhovou interpolací G03. V červeném rámečku jsou zvýrazněny potřebné výpočty pro jednotlivé osy a také velikost rádiusu CR, po kterém se nástroj v rámci spirály pohybuje. Dále je zde podmínka určující do jakého průměru má program dojet.

Další snaha je tedy pokusit se takto definovanou 2D spirálu dostat do 3D, tak aby se při pohybu nástroje měnily všechny tři osy najednou. Vzhledem k tomu, že kruhová interpolace G2 i G3 umožňuje pohyb ve všech třech osách, tak je o problém méně. Souřadnice všech těchto bodů jsou stejné jako u předchozích operací, tudíž je stačí dosadit. U výpočtu rádiusu se metodicky postupuje stejně jako u 2D operace, tedy pod odmocninu přidáme druhou mocninu Z-ové souřadnice, viz. ohraničená část programu výše. Tuto technologii lze tedy využít pro obrábění typově podobných dílů jako je zadaná součást, a to jak ve dvou, tak i ve třech osách. Po převedení programu ze systému Sinumerik do systému Heidenhain lze obrábět na více strojích s rozdílnými řídícími systémy.

```
M30
LABEL2:
G00 X200 Y200 Z200
T="FRÉZA D20" M6
G94 S1500 F5000
R60=R40-R10
R61=50
G00 X0 Y0 Z0 M3
POCKET4(100,-R60,1,-R11,R61,0,0,2.5,0,0,200,0.1,0,1011,0.5,9,15,
0,2,0,1,2,10100,111,101)
R4=R61
GOTOB LABEL3
LABEL0:
```

Obr. 37 – podprogram LABEL2

Za funkcí M30, určující konec programu, se nachází podprogram pro předfrézování kruhové kapsy při dodání plného polotovaru. Je zvolena menší fréza, která umožňuje zavrtání po šroubovici do plného materiálu a znovu je aktualizován parametr vnitřního průměru z 0 na 50 mm po dokončení kapsování.

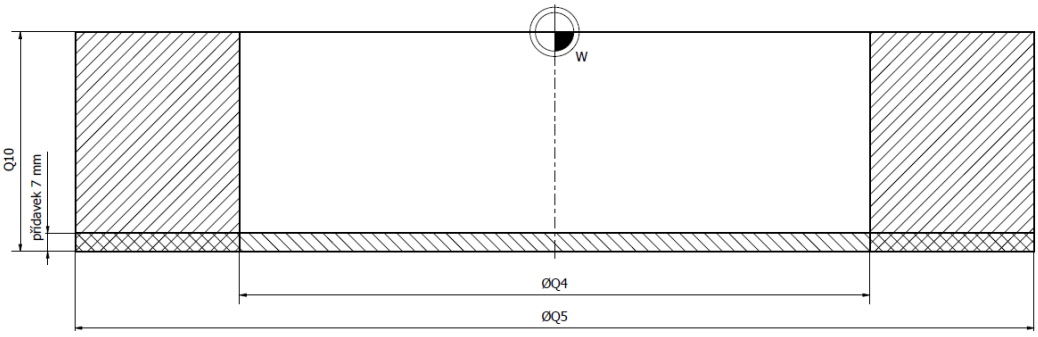


Obr. 38 – pozice nulového bodu na spodní ploše

Ve všech výše zmíněných variantách programu je nastaven nulový bod obrobku na spodní hraně dílu, tedy na pracovním stole obráběcího stroje. V tomto případě je ale potřeba počítat s přídavkem spodní plochy a posunout nulový bod pro výpočet i o tento přídavek. Po domluvě s vedoucím výroby ve firmě Streicher je snaha o přesunutí nulového bodu na horní stranu polotovaru a tím se vyvarovat nepřesnosti obrobené plochy při špatném započítání přídavku nebo pokud by byl přídavek jiný.

5.3 Tvorba programu s nulovým bodem na horní ploše

Přesunutí nulového bodu na horní plochu polotovaru má výhodu v možnosti zanedbání přídavku spodní plochy pro výpočty, vzhledem k tomu, že veškeré výpočty a dráhy nástroje se budou vztahovat k nulovému bodu, který je na horní ploše. Tím, že přesuneme nulový bod, se ovšem změní výpočet souřadnic pro osu Z. Tyto výpočty byly odvozeny přímo od rádiusu sudu a to díky použitému posunutí nulového bodu. V tomto případě je snaha o dosažení výpočtu souřadnic pouze pomocí funkce, tzn. bez použití posunutí nulového bodu.



Obr. 39 – pozice nulového bodu na horní ploše

Při změně nulového bodu a výpočtů potřebných pro pohyb nástroje je jasné, že není potřeba měnit rovnice pro výpočet souřadnic pro osy X a Y, v těch se totiž změna v ose Z neobjevuje.

```
23 Q41 = Q8 * COS Q9 ; X-OVA SOURADNICE
24 Q42 = Q8 * SIN Q9 ; Y-OVA SOUZEDNICE
```

Obr. 40 – výpočet X-ové a Y-ové souřadnice

Díky tomu bude potřeba se zamyslet pouze nad rovnicí pro výpočet drah v ose Z. Při odvozování vztahu pro výpočet Z-oté osy se vyšlo z předchozího výpočtu a ten se upravil pro dosažení finálního vzorce. Pro získání parametru Q43, který je dosazen za osu Z v dráhové funkci je potřeba využít výpočtu pro parametr Q11 z Pythagorovy věty a od něj odečíst hodnotu vzdálednosti středu radiusu od horní plochy polotovaru c, viz. Obr. 42. Výsledný vzorec má tedy tvar:

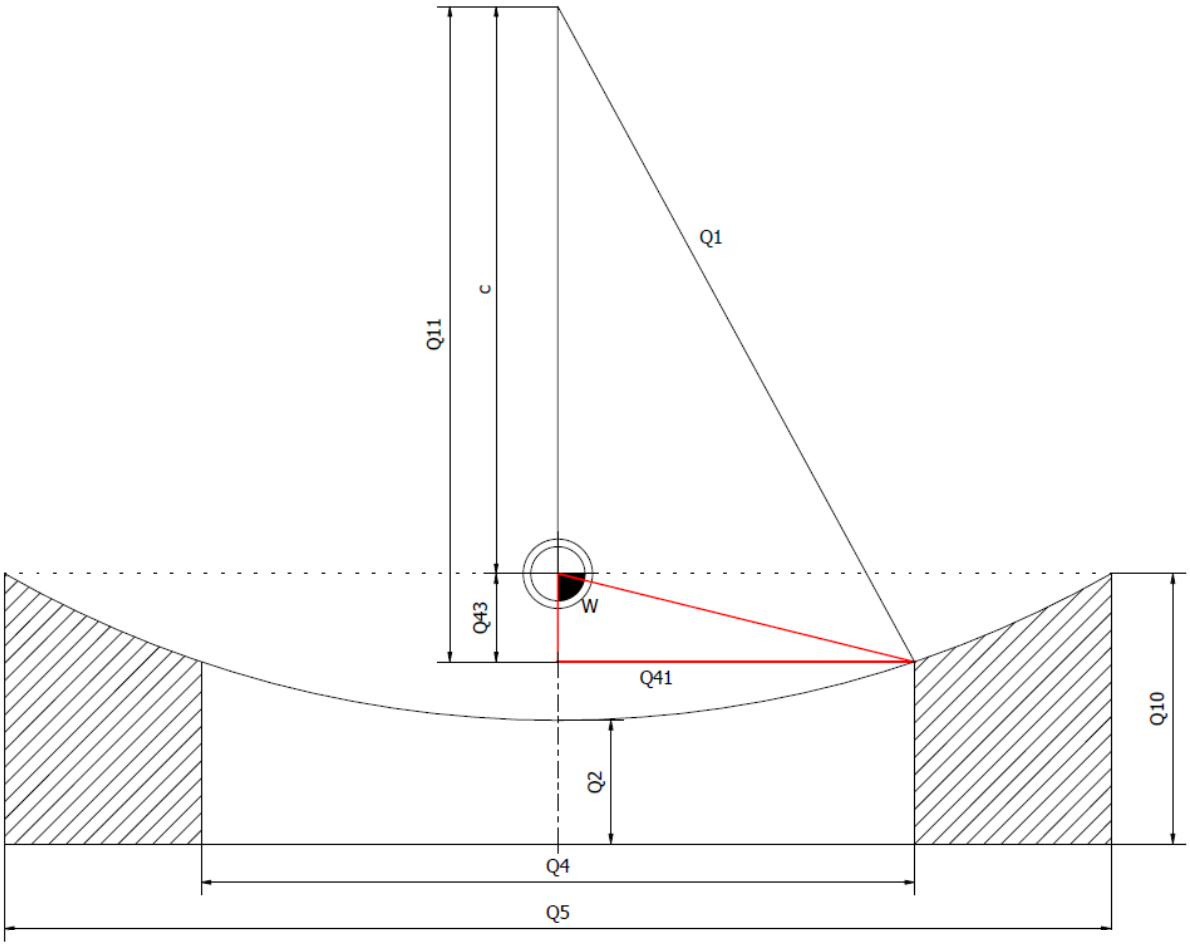
$$Q43 = Q11 - c = Q11 - (Q1 - (Q10 - Q2))$$

$$Q43 = \sqrt{Q1^2 - Q41^2} - (Q1 - (Q10 - Q2))$$

```
25 Q11 = SQRT ( SQ Q1 - SQ Q41 )
26 Q43 = - ( Q11 - ( Q1 - ( Q10 - Q2 ) ) ) ; Z-OVA SOURADNICE
```

Obr. 41 – výpočet Z-ové souřadnice

Pro lepší přehlednost v programu jsou tyto výpočty rozděleny na 2 samostatné řádky.



Obr. 42 – výpočet parametru Q43

Vzhledem k tomu, že je souřadnice počítána rovnou od nulového bodu obrobku, již se v programu nenachází posunutí nulového bodu. Další změnou je definice kruhové kapsy, kde souřadnice povrchu kapsy je 0, vzhledem k nastavení nulového bodu na této ploše a změnil se výpočet pro hloubku kapsy, který vychází z celkové výšky kusu a vzdálenosti rádiusu od spodní hrany.

5.4 Shrnutí

V rámci praktické části této práce jsem navrhoval a optimalizoval parametrický program pro zadanou typovou součást, kterou je příruba používaná do sestavy tlakových nádob. Podnět pro vypracování podala firma Streicher. Proběhl návrh několika variant, které v následující kapitole porovnáme.

5.5 Zhodnocení variant a výběr optimální varianty

Vzhledem k tomu že jsem v rámci své bakalářské práce navrhl několik variant parametrického programu pro zadanou součást od firmy Streicher, je třeba tyto varianty porovnat a vybrat tu nejvýhodnější variantu. První verze programu je s použitím CAM systému a generování kódu v postprocesoru. Druhou variantou je parametrický program používaný ve firmě v současnosti. Další variantou je upravený program s přidaným cyklem pro kruhovou kapsu a nulovým bodem na spodní hraně polotovaru. Poslední varianta je podobná té třetí, pouze je přesunut nulový bod na horní plochu polotovaru. Porovnání je znázorněno níže, kde Varianta A představuje program v CAM systému, Varianta B představuje současný stav, Varianta C představuje s nulovým bodem dole a Varianta D představuje přesunutí nulového bodu na horní plochu polotovaru.

- Varianta A – CAM systém:
  - Je zapotřebí vlastnit licenci na CAM systém
  - Není možné pro obsluhu vytvořit a měnit program v provozu
  - Program není flexibilní, tutíž je pouze pro jednu součást
- Varianta B – současný stav:
  - Složitější programování než u CAM sastému, je potřeba znát možnost použití parametrů a práce s nimi
  - Zjednodušení práce pro programátora, který napíše jeden program, kde pouze obsluha mění nadefinované parametry
  - Obsluha zadává parametry přímo z výkresu
  - Není dostatečně flexibilní, chybí zde cyklus pro frézování kapsy pro plný polotovar
- Varianta C – spodní nulový bod:
  - Oproti variantě B je zde přidán cyklus pro frézování kruhové kapsy, pro případ plného polotovaru
  - Je zde počítáno s přídavkem spodní plochy, který na součásti zůstává i po obrobení požadovaného rádiusu, ovšem pouze pro jednu velikost přídavku, při jeho změně dochází k výrobě špatných kusů
- Varianta D – horní nulový bod:
  - Přesunutím nulového bodu zamezíme případné výrobě špatných kusů při změně přídavku spodní plochy
  - Jednodušší zadefinování cyklu pro kruhovou kapsu

Z porovnání variant lze vyčíst, že varianta A je pro tento typ výroby nejméně vhodná pro její dlouhý čas přípravy jednotlivých programů na každou součást zvlášť. U varianty B už si program podle výkresové dokumentace upraví obsluha, musí však nějaké hodnoty dopočítávat a není zde cyklus pro plný polotovar. Varianta C už je použitelná, ovšem obsluha musí kontrolovat, zda sedí výška polotovaru a velikost přídavku. Tohoto problému se zbavujeme při použití varianty D, která má nulový bod na horní ploše polotovaru, tutíž jsou všechny souřadnice počítány od horní plochy a nezáleží tak na velikosti přídavku.

6. Závěr

Cílem této práce bylo vytvořit a optimalizovat parametrický program pro typovou součást, přírubu, která se přivařujue na plášť tlakové komory. Podnět na toto téma podala firma STREICHER, spol. s.r.o., Plzeň. Program byl navrhován tak, aby ho mohla bez problémů měnit obsluha na stroji bez potřeby zásahu programátora.

V první částí práce je seznámení s teoretickou částí programování NC strojů od ručního programování po zavedení parametrického programování. Jsou zde ukázány možnosti použití parametrů v programu. Dále jsou porovnávány rozdíly mezi různými řídícími systémy a rozdíl v používání parametrů v jednotlivých systémech.

V praktické části proběhla analýza současného stavu firmy Streicher, náhled do současného programu a jeho popis. V dalším kroku bylo třeba zamyslet se nad tím, jak program optimalizovat, nebo ho popřípadě celý přepsat. Po zanesení podmínek od firmy Streicher, které spočívaly v obrábění pouze jedním nástrojem a zajištění co nejpružnějšího a optimalizovaného programu, byla zvolena technologie frezování po rádiusu.

Původní program nebyl dokonale odladěný, ani nebyl připraven pro obsluhu, která by pouze zadávala parametry z výkresu. Navíc se objevovaly i plné polotovary, s čímž program nebyl schopný pracovat. Proto bylo třeba přidat operaci pro tuto technogii předfrézování. První varianta využívala podobné výpočty a pohyby nástroje jako původní, ovšem s přidáním cyklu kruhové kapsy pro plný polotovar a přepsání programu do přehlednější podoby. V další variantě byla snaha o zvětšení úhlového kroku změnou lineárních drah na kruhové. To ovšem fungovalo pouze v dvou osách, pro tuto práci tedy nepoužitelné. Poslední změnou bylo přesunutí nulového bodu na horní plochu polotovaru, které vedlo ke změně výpočtů, ale zjednodušení práce pro obsluhu při zadávání parametrů.

Všechny varianty programu byly odzkoušeny a odladěny v simulaci na virtuálním stroji a budou předloženy programátorům ve firmě Streicher pro získání zpětné vazby, která verze programu jim nejvíce vyhovuje.

Stanovené cíle této bakalářské práce byly naplněny, ovšem stále by šel program dále zjednodušit a optimalizovala by se dráha nástroje využím komplexnějších výpočtů a pokročilejší práce s parametry. Toto téma by mohlo být podkladem pro navazující diplomovou práci.

Seznam použitých zdrojů

[1] HEIDENHAIN: Příručka uživatele, Popisný dialog firmy Heidenhain iTNC 530. NC-software 606-42x-02, 01/2012

[2] *K čemu slouží G kódy a M kódy?* [online]. [cit. 2022-05-08]. Dostupné z: https://www.macmatic.cz/component/content/article/40-technicke-clanky/66-k-cemu-slouzi-g-kody-a-m-kody

[3] SIEMENS: Programovací příručka, SINUMERIK 840D sl/828D, Pro pokročilé. 6FC5398-2BP40-3UA1, 03/2013

[4] POLZER, Aleš. *Parametrické programování a skoky v NC programech* [online]. [cit. 2022-03-06]. Dostupné z: https://web.archive.org/web/20110910070538/http://www.techtydenik.cz/cnc_priklady/priklad_19.pdf

[5] JANDEČKA, Karel, Pavel KOŽMÍN a Jiří ČESÁNEK. *Programování NC strojů*. Plzeň: Západočeská univerzita, 2000. ISBN 80-708-2692-4.

[6] *Obecný úvod do problematiky NC programování* [online]. In: . s. 20-51 [cit. 2022-03-06]. Dostupné z: https://www.sosbites.cz/images/stories/Pro-studenty/studijni-materialy/VUKOV_TEXT_-_1.ST.pdf

[7] *SECOtools* [online]. [cit. 2022-05-18]. Dostupné z: https://www.secotools.com/article/p_02827017

[8] *SECOtools* [online]. [cit. 2022-05-18]. Dostupné z: https://www.secotools.com/article/p_02827021?language=cs

[9] POLZER, Aleš. *Parametrické programování a skoky v NC programech* [online]. [cit. 2022-03-06]. Dostupné z: https://web.archive.org/web/20110910070538/http://www.techtydenik.cz/cnc_priklady/priklad_4.pdf

[10] Sinumerik 840D easy screen - spiral outside- in. *YOUTUBE* [online]. 2018 [cit. 2022-05-18]. Dostupné z: https://www.youtube.com/watch?v=nNbVnBiU9MY

PŘÍLOHA č. 1

Původní program používaný ve firmě STREICHER, spol. s r.o. Plzeň

```
0 BEGIN PGM PRSUD-SPIRALOU MM
1 BLK FORM 0.1 Z X-180 Y-180 Z+0
2 BLK FORM 0.2  X+180  Y+180  Z+83.1
3 ; NAZEV : FREZOVANI RADIUSU SUDU DO PRIRUBY SPIRALOU
4 ; ----------------------------
5 * - NULOVY BOD
6 ; X0 - OSA PRIRUBY
7 ; Y0 - OSA PRIRUBY
8 ; Z0 - SPODNI PLOCHA PRIRUBY
9 ; ----------------------------
10 * - DEFINICE NASTROJU
11 LBL 30
12 ;
13 FN 0: Q51 =+15 ;JEZEK NEBO KULOVKA
14 ;
15 FN 0: Q52 =+2 ;
16 ;
17 LBL 0
18 ; ----------------------------
19 * - NACISTO TVAR
20 ;JEZEK D32
21 CALL LBL 30
22 TOOL CALL Q51 Z S1930 F1100
23 L  Z+200 R0 FMAX M3 M18
24 FN 0: Q1 =+710 ; FREZOVANY RADIUS SUDU
25 FN 0: Q2 =+60 ; ROZMER KUSU OD SPODNI HRANY K RADIUSU SUDU
26 FN 0: Q3 =+1 ; UHLOVY KROK VE ST
27 FN 0: Q4 =+232 ; OBRABENY VNITRNI PRUMER PRIRUBY
28 FN 0: Q5 =+362 ; OBRABENY VNEJSI PRUMER PRIRUBY
29 FN 0: Q6 =+1 ; KROK NA RADIUSU V MM (SPONA DO BOKU)
30 CALL LBL 1
31 ; ----------------------------
32 M2
33 ; ****************************
34 * - *** PODPROGRAMY ***
35 ; ****************************
36 * - LBL 1 - TVAR
37 LBL 1
38 Q40 = Q1 + Q2 ; POSUNUTI STREDU RADIUSU
39 CYCL DEF 7.0 NULOVY BOD
40 CYCL DEF 7.1 IZ+Q40
41 CYCL DEF 10.0 OTACENI
42 CYCL DEF 10.1  ROT+0
43 Q8 = Q4 / 2 - Q6 ; OBRABENY RADIUS
44 Q7 = Q5 / 2 + ( Q6 * 2 ) ; KONCOVY RADIUS
45 Q8 = Q8 + Q6 ; OBRABENY RADIUS
46 FN 0: Q9 =+0 ; VYCHOZI UHEL
47 Q10 = Q8 * SIN ( Q9 - 90 )
48 Q11 = SQRT ( SQ Q1 - SQ Q10 )
49 Q19 = - Q11
50 L  X+0  Y+0 R0 FMAX
51 L  Z+Q19 R0 FMAX
52 CC  X+0  Y+0
53 LP  PR+Q8  PA+Q9 RL F AUTO
54 LBL 122
55 Q8 = Q8 + ( Q6 / 360 * Q3 ) ; OBRABENY RADIUS
56 Q9 = Q9 + Q3
57 Q10 = Q8 * SIN ( Q9 - 90 )
58 Q11 = SQRT ( SQ Q1 - SQ Q10 )
59 Q19 = - Q11
60 Q41 = COS Q9 * Q8 ; X-OVA SOURADNICE
61 Q42 = SIN Q9 * Q8 ; Y-OVA SOURADNICE
62 L  X+Q41  Y+Q42  Z+Q19 RL F AUTO
63 FN 12: IF +Q8 LT +Q7 GOTO LBL 122
64 L  X+0  Y+0 R0 FMAX
65 CYCL DEF 10.0 OTACENI
66 CYCL DEF 10.1  ROT+0
67 CYCL DEF 7.0 NULOVY BOD
68 CYCL DEF 7.1 IZ-Q40
69 L  Z+200 R0 FMAX
70 LBL 0
71 ; ----------------------------
72 END PGM PRSUD-SPIRALOU MM
```

PŘÍLOHA č. 2

Optimalizovaný parametrický program s nulovým bodem na spodní ploše polotovaru v řídícím systému Sinumerik

```
WORKPIECE(,"",,"CYLINDER",64,0,70,-80,360)
G54 G90 G17
R1=525 ; RADIUS FREZOVANEHO SUDU
R2=40 ; ROZMER PRIRUBY OD SPODNI HRANY
R3=1 ; UHLOVY KROK VE ST
R4=0 ; VNITRNI PRUMER PRIRUBY
R5=360 ; VNEJSI PRUMER PRIRUBY
R6=10 ; RADIALNI KROK
R10=70 ; CELKOVA VYSKA SUDU
LABEL3:
G00 X200 Y200 Z200
T="JEŽEK 32" M6
G94 S1500 F5000
R7=R5/2+32
R8=R4/2-16
R9=0 ; VYCHOZI UHEL
R11=R10-R2
R40=R1+R2 ; POSUNUTI NULOVEHO BODU
TRANS Z=R40
IF R4<50 GOTOF LABEL2
G00 X0 Y0 Z=-R1+10
LABEL1:
R8=R8+(R6/360*R3)
R9=R9+R3
R10=R8*SIN(R9-90)
R11=-SQRT(R1*R1-R10*R10)
R41=COS(R9)*R8
R42=SIN(R9)*R8
G01 X=R41 Y=R42 Z=R11
IF R8<R7 GOTOB LABEL1
LABEL0:
M30
LABEL2:
G00 X200 Y200 Z200
T="FRÉZA D20" M6
G94 S1500 F5000
R60=R40-R10
R61=50
G00 X0 Y0 Z0 M3
POCKET4(100,-R60,1,-R11,R61,0,0,2.5,0,0,200,0.1,0,1011,0.5,9,15,0,2,0,1,2,10100,111,101)
R4=R61
GOTOB LABEL3
LABEL0:
```

PŘÍLOHA č. 3

Optimalizovaný parametrický program s nulovým bodem na spodní ploše polotovaru v řídícím systému Heidenhain

```
0 BEGIN PGM 1 MM
1 BLK FORM 0.1 Z  X-150  Y-150  Z+0
2 BLK FORM 0.2  X+150  Y+150  Z+70
3 FN 0: Q1 =+300 ; RADIUS FREZOVANEHO SUDU
4 FN 0: Q2 =+40 ; ROZMER PRIRUBY OD SPODNI HRANY
5 FN 0: Q3 =+1      ; UHLOVY KROK VE ST
6 FN 0: Q4 =+0      ; VNITRNI PRUMER PRIRUBY
7 FN 0: Q5 =+300 ; VNEJSI PRUMER PRIRUBY
8 FN 0: Q6 =+1      ; RADIALNI KROK
9 FN 0: Q10 =+70 ; CELKOVA VYSKA KUSU
10 LBL 3
11 Q7 = Q5 / 2
12 Q8 = Q4 / 2
13 FN 0: Q9 = 0 ; VYCHOZI UHEL
14 Q12 = Q2 + 7 ; PRIPOCITANI PRIDAVKU SPODNI PLOCHY 7MM
15 Q40 = Q1 + Q12 ; POSUNUTI STREDU RADIUSU
16 CYCL DEF 7.0 NULOVY BOD
17 CYCL DEF 7.1  Z+Q40
18 L  X+200  Y+200  Z+200 R0 FMAX
19 TOOL CALL 52 Z S1200 F250
20 L  X+0  Y-7.5  Z+100 R0 FMAX
21 FN 12: IF +Q4 LT +50 GOTO LBL 2
22 CALL LBL 1
23 CYCL DEF 7.0 NULOVY BOD
24 CYCL DEF 7.1  Z-Q40
25 L  Z+200 R0 FMAX M30
26 LBL 1
27 Q8 = Q8 + ( Q3 * Q6 ) / 360
28 Q9 = Q9 - 1
29 Q41 = Q8 * COS Q9 ; X-OVA SOURADNICE
30 Q42 = Q8 * SIN Q9 ; Y-OVA SOURADNICE
31 Q43 = - ( SQRT ( SQ Q1 - SQ Q41 ) ) ; Z-OVA SOURADNICE
32 L  X+Q41  Y+Q42  Z+Q43 RL FMAX M3
33 FN 12: IF +Q8 LT +Q7 GOTO LBL 1
34 LBL 0
35 LBL 2
36 L  Z+200 R0 FMAX
37 TOOL CALL 10 Z S1200 F2500
38 Q11 = Q10 - Q12
39 Q60 = Q40 - Q10
40 CYCL DEF 252 KRUHOVA KAPSA ~
   Q215=+0    ;ZPUSOB OBRABENI ~
   Q223=+50   ;PRUMER KRUHU ~
   Q368=+0    ;PRIDAVEK PRO STRANU ~
   Q207=+500  ;FREZOVACI POSUV ~
   Q351=+1    ;ZPUSOB FREZOVANI ~
   Q201=-Q11  ;HLOUBKA ~
   Q202=+5    ;HLOUBKA PRISUVU ~
   Q369=+0    ;PRIDAVEK PRO DNO ~
   Q206=+150  ;POSUV NA HLOUBKU ~
   Q338=+0    ;PRISUV NA CISTO ~
   Q200=+2    ;BEZPECNOSTNI VZDAL. ~
   Q203=-Q60  ;SOURADNICE POVRCHU ~
   Q204=+50   ;2. BEZPEC.VZDALENOST ~
   Q370=+1    ;PREKRYTI DRAHY NAST. ~
   Q366=+1    ;ZANOROVANI ~
   Q385=+500  ;POSUV NACISTO
41 L  X+0  Y+0  Z+0 R0 FMAX M3
42 CYCL CALL
43 Q4 = Q223
44 CALL LBL 3
45 LBL 0
46 END PGM 1 MM
```

PŘÍLOHA č. 4

Optimalizovaný parametrický program s nulovým bodem na horní ploše polotovaru v řídícím systému Sinumerik

```
WORKPIECE(,"",,"CYLINDER",64,0,-70,-80,360)
G54 G90 G17
R1=525 ; RADIUS FREZOVANEHO SUDU
R2=40 ; ROZMER PRIRUBY OD SPODNI HRANY
R3=10 ; UHLOVY KROK VE ST
R4=0 ; VNITRNI PRUMER PRIRUBY
R5=360 ; VNEJSI PRUMER PRIRUBY
R6=1 ; RADIALNI KROK
R10=70 ; CELKOVA VYSKA SUDU
LABEL3:
G00 X200 Y200 Z200
T="JEŽEK 32" M6
G94 S1500 F5000
R7=R5/2+32
R8=R4/2-16
R9=0 ; VYCHOZI UHEL
IF R4<50 GOTOF LABEL2
G00 X0 Y0 Z=-R60+5
LABEL1:
R8=R8+(R6/360*R3)
R9=R9+R3
R41=COS(R9)*R8
R42=SIN(R9)*R8
R11=SQRT(R1*R1-R41*R41)
R43=-(R11-(R1-(R10-R2)))
G01 X=R41 Y=R42 Z=R43
IF R8<R7 GOTOB LABEL1
LABEL0:
M30
LABEL2:
G00 X200 Y200 Z200
T="FRÉZA D20" M6
G94 S1500 F5000
R60=R10-R2
R61=50
G00 X0 Y0 Z0 M3
POCKET4(100,0,1,-R60,R61,0,0,2.5,0,0,200,0.1,0,1011,0.5,9,15,0,2,0,1,2,10100,111,101)
R4=R61
GOTOB LABEL3
LABEL0:
```

PŘÍLOHA č. 5

Optimalizovaný parametrický program s nulovým bodem na horní ploše polotovaru v řídícím systému Heidenhain

```
0 BEGIN PGM 6 MM
1 BLK FORM 0.1 Z  X-150  Y-150  Z-70
2 BLK FORM 0.2  X+150  Y+150  Z+0
3 FN 0: Q1 =+300 ; RADIUS FREZOVANEHO SUDU
4 FN 0: Q2 =+40 ; ROZMER PRIRUBY OD SPODNI HRANY
5 FN 0: Q3 =+1   ; UHLOVY KROK VE ST
6 FN 0: Q4 =+0   ; VNITRNI PRUMER PRIRUBY
7 FN 0: Q5 =+300 ; VNEJSI PRUMER PRIRUBY
8 FN 0: Q6 =+1   ; RADIALNI KROK
9 FN 0: Q10 =+70 ; CELKOVA VYSKA KUSU
10 LBL 3
11 Q7 = Q5 / 2
12 Q8 = Q4 / 2
13 FN 0: Q9 =-90 ; VYCHOZI UHEL
14 L  X+200  Y+200  Z+200 R0 FMAX
15 TOOL CALL 52 Z S1200 F250
16 L  X+0  Y-7.5  Z+100 R0 FMAX
17 FN 12: IF +Q4 LT +50 GOTO LBL 2
18 CALL LBL 1
19 L  Z+200 R0 FMAX M30
20 LBL 1
21 Q8 = Q8 + ( Q3 * Q6 ) / 360
22 Q9 = Q9 - 1
23 Q41 = Q8 * COS Q9 ; X-OVA SOURADNICE
24 Q42 = Q8 * SIN Q9 ; Y-OVA SOUZEDNICE
25 Q11 = SQRT ( SQ Q1 - SQ Q41 )
26 Q43 = - ( Q11 - ( Q1 - ( Q10 - Q2 ) ) ) ; Z-OVA SOURADNICE
27 L  X+Q41  Y+Q42  Z+Q43 R0 FMAX M3
28 FN 12: IF +Q8 LT +Q7 GOTO LBL 1
29 LBL 0
30 LBL 2
31 Q60 = Q10 - Q2
32 CYCL DEF 252 KRUHOVA KAPSA ~
   Q215=+0    ;ZPUSOB OBRABENI ~
   Q223=+50   ;PRUMER KRUHU ~
   Q368=+0    ;PRIDAVEK PRO STRANU ~
   Q207=+500  ;FREZOVACI POSUV ~
   Q351=+1    ;ZPUSOB FREZOVANI ~
   Q201=-Q60  ;HLOUBKA ~
   Q202=+5    ;HLOUBKA PRISUVU ~
   Q369=+0    ;PRIDAVEK PRO DNO ~
   Q206=+150  ;POSUV NA HLOUBKU ~
   Q338=+0    ;PRISUV NA CISTO ~
   Q200=+2    ;BEZPECNOSTNI VZDAL. ~
   Q203=+0    ;SOURADNICE POVRCHU ~
   Q204=+50   ;2. BEZPEC.VZDALENOST ~
   Q370=+1    ;PREKRYTI DRAHY NAST. ~
   Q366=+1    ;ZANOROVANI ~
   Q385=+500  ;POSUV NACISTO
33 L  X+0  Y+0  Z+0 R0 FMAX M3
34 CYCL CALL
35 Q4 = Q223
36 CALL LBL 3
37 LBL 0
38 END PGM 6 MM
```

PŘÍLOHA č. 6

Parametrický program pro obrábění po spirále ve dvou osách v řídícím systému Heidenhain

```
0 BEGIN PGM 3 MM
1 BLK FORM 0.1 Z  X-150  Y-150  Z+0
2 BLK FORM 0.2  X+150  Y+150  Z+70
3 FN 0: Q3 =+1 ; UHLOVY KROK VE ST
4 FN 0: Q4 =+80 ; VNITRNI PRUMER PRIRUBY
5 FN 0: Q5 =+300 ; VNEJSI PRUMER PRIRUBY
6 FN 0: Q6 =+1 ; RADIALNI KROK
7 LBL 3
8 Q7 = Q5 / 2
9 Q8 = Q4 / 2 - 5
10 FN 0: Q9 =+0 ; VYCHOZI UHEL
11 L  Z+200 R0 FMAX
12 TOOL CALL 16 Z S1200 F2500
13 FN 12: IF +Q4 LT +50 GOTO LBL 2
14 L  X+0  Y+0  Z+65 RL FMAX
15 CALL LBL 1
16 L  Z+200 R0 FMAX M30
17 LBL 1
18 Q9 = Q9 + 1
19 Q8 = Q8 + 5
20 CC  X+0  Y+0
21 CTP  PR+Q8  PA+0
22 CP IPA+90 DR+
23 FN 12: IF +Q8 LT +Q7 GOTO LBL 1
24 LBL 0
25 LBL 2
26 L  Z+200 R0 FMAX
27 TOOL CALL 10 Z S1200 F2500
28 CYCL DEF 252 KRUHOVA KAPSA ~
   Q215=+0    ;ZPUSOB OBRABENI ~
   Q223=+80   ;PRUMER KRUHU ~
   Q368=+0    ;PRIDAVEK PRO STRANU ~
   Q207=+500  ;FREZOVACI POSUV ~
   Q351=+1    ;ZPUSOB FREZOVANI ~
   Q201=-5    ;HLOUBKA ~
   Q202=+5    ;HLOUBKA PRISUVU ~
   Q369=+0    ;PRIDAVEK PRO DNO ~
   Q206=+150  ;POSUV NA HLOUBKU ~
   Q338=+0    ;PRISUV NA CISTO ~
   Q200=+2    ;BEZPECNOSTNI VZDAL. ~
   Q203=+70   ;SOURADNICE POVRCHU ~
   Q204=+50   ;2. BEZPEC.VZDALENOST ~
   Q370=+1    ;PREKRYTI DRAHY NAST. ~
   Q366=+1    ;ZANOROVANI ~
   Q385=+500  ;POSUV NACISTO
29 L  X+0  Y+0 R0 FMAX M3
30 CYCL CALL
31 Q4 = Q223
32 CALL LBL 3
33 LBL 0
34 END PGM 3 MM
```