

# Laborator 7

## Mașina CNC

As. ing. Alexandru Dumitrache  
As. ing. Raluca Tudorie  
[www.scr.cimr.pub.ro](http://www.scr.cimr.pub.ro)

### Introducere

Un echipament CNC (Computer Numerically Controlled) este o mașină de prelucrare comandată de un microprocesor.

Cele mai simple mașini au 2 sau 3 axe carteziene. Mașinile mai avansate pot avea și axe rotative. Din punct de vedere cinematic, mașinile CNC se aseamănă foarte mult cu roboții industriali, în special cu cei cartezieni.

Câteva tipuri de echipamente CNC:

- Mașini de găurit în coordonate (poziționare pe X și Y);
- Strunguri comandate numeric (cu două sau mai multe grade de libertate);
- Mașini de frezare;
- Centre de prelucrare prin așchiere;
- Mașini pentru prelucrarea lemnului (de dimensiuni foarte mari);
- Mașini de prelucrare prin electroeroziune (Wire EDM);
- Mașini de tăiere cu plasmă, cu laser sau cu jet de apă;
- Mașini cu fir incandescent, pentru tăierea polistirenului.

## Mașina EMCO ConceptMill 105

Mașina EMCO ConceptMill 105 este o freză verticală cu 3 axe de translație și una suplimentară de rotație. Interfața cu operatorul emulează un controller industrial de tip Siemens Sinumerik 840D.

Date tehnice și dotări:

- Materiale prelucrate: plastic, metal, lemn;
- Operații efectuate: frezare, găurire, filetare;
- Magazie de scule și schimbător automat (10 poziții SK-30);
- Spindle: 150 - 5000 rpm, 1.1 kW;
- 3 axe carteziane X / Y / Z:
  - Cursa efectivă:  $200 \times 150 \times 150$  mm;
  - Forța de înaintare: 2000 / 2000 / 2400 N;
  - Acuratețe la poziționare: 3 / 3 / 4  $\mu m$ ;
  - Acționare: motoare pas cu pas;
  - Viteza de traversare rapidă: 5000 mm/min;
- Axa rotativă A:
  - Tip constructiv: cap divizor;
  - Cuplu maxim: 42 Nm;
  - Viteza de rotație:  $0 \div 8$  rpm;
  - Rezoluție:  $\pm 100$  arcsecunde;
  - Repetabilitate:  $\pm 15$  arcsecunde;
  - Viteza de traversare rapidă: 5000 mm/min;
  - Fixarea piesei de prelucrat:
    - \* mandrină tip strung  $\Phi 74$  mm cu 3 bacuri reversibile;
    - \* păpușă mobilă;
    - \* piesa se fixează manual;
- Elemente de automatizare:
  - Menghină cu acționare pneumatică - hidraulică;
  - Ușă automată, acționată pneumatic;
  - Interfața cu robotul industrial, prin comunicație DNC și semnale I/O digitale;
- Restricție: Menghina pneumatică și axa rotativă nu pot fi montate simultan pe masa de lucru.

## Realizarea unei piese pe mașina CNC

*It is easier to change the specification to fit the CNC program than viceversa.*

---

Etapele necesare pentru realizarea unei piese:

- Proiectarea piesei într-un program CAD (Computer Aided Design);
- Definirea traiectoriilor de prelucrare într-un program de tip CAM (Computer Aided Manufacturing);
- Simularea programului (G-Code) obținut;
- Testarea programului pe mașină, mai întâi în aer, apoi pe piesă.

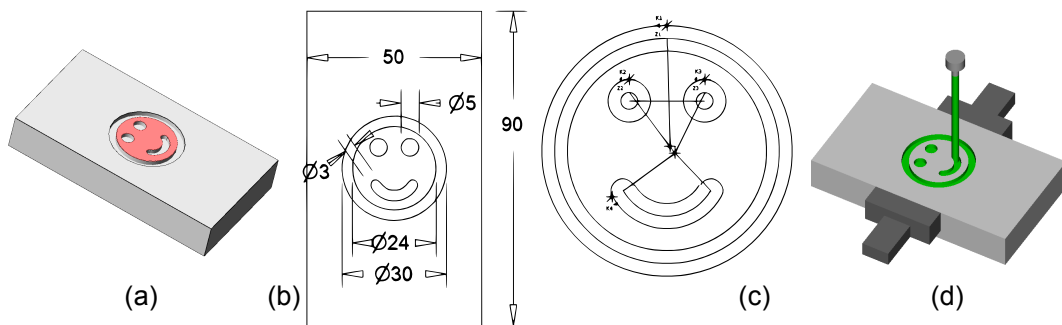


Figura 7.1: Piesa de tip „față zâmbitoare”: (a) Model CAD (Alibre Design); (b) Format DXF; (c) Traiectoriile frezei în 2D (EMCO CamConcept); (d) Simularea traiectoriilor de prelucrare (EMCO 3D View)

Software CAD disponibil în laborator:

- Alibre Design Professional (proiectare parametrică 3D)
- Alibre Design Xpress (versiunea gratuită, are câteva limitări)

Software CAM disponibil în laborator:

- EMCO CamConcept Mill - permite definirea prelucrărilor 2D pornind de la un fișier DXF
- Alibre CAM
- SprutCAM - poate genera și simula traiectorii de prelucrare complexe plecând de la modele 3D (importă IGES)
- g3d - pentru prelucrarea suprafețelor 2.5 D descrise printr-o hartă de profunzime (imagine cu niveluri de gri) pe freza verticală cu 3 axe
- g4d - extensie pentru axa rotativă (experimental)
- POV-Ray - pentru generarea hărților de profunzime

Software pentru simularea programelor G-Code:

- EMCO 3DView
- FlashCut CNC (Demo), NCPlot (free), CNC Simulator (free)

Programul G-Code (DIN 66025) poate fi editat și manual, în Notepad sau în editorul intern al programului EMCO WinNC. Câteva comenzi de bază:

- G0 X... Y... Z... ; deplasare cu viteză maximă, prin aer
- G1 X... Y... Z... F... ; deplasare cu viteză de avans programată (Feed Rate - exprimată în mm/min)
- G2 X... Y... I... J... F... ; arc de cerc în sens orar
- G3 X... Y... I... J... F... ; arc de cerc în sens antiorar
- M3 S... ; pornire freză (Spindle) în sens normal (orar)
- M4 S... ; pornire freză (Spindle) în sens invers
- M5 ; oprire freză
- M6 T... D... ; selectează freza cu nr. T și setul de parametri D
- M0 ; pauză în program (ușa mașinii se deschide)
- M30 ; sfârșit program

Înainte de a rula un program CNC, trebuie învățată originea piesei. Procedeu este similar cu învățarea unui punct robot, cu deosebirea că axele sistemelor de coordonate utilizator sunt paralele cu axele mașinii.

Pentru setarea originii pe o latură a piesei neprelucrate se poate folosi instrumentul excentric Edge Finder, rotit cu 1000 rpm:

- Dacă muchia indicatorului este departe de muchia piesei, în exterior, instrumentul este descentrat;
- Dacă muchia instrumentului se apropie de muchia piesei, însă este încă în exterior, instrumentul se centrează;
- În momentul în care muchia instrumentului a depășit muchia piesei, ajungând în interior, unealta de aliniere se descentrează imediat.

Sistemele de coordonate folosite de mașina EMCO 105:

- Machine Coordinate System (MCS) - selectat cu G500 (similar cu sistemul World al robotului);
- Patru origini (zero offsets) atașate piesei care va fi prelucrată (Work Coordinate System - WCS), selectate cu instrucțiunile:
  - G54 - pentru prelucrări 2.5D de uz general;
  - G55 - pentru prelucrări folosind axa rotativă;
  - G56 - originea pieselor la prelucrarea în regim automat;
  - G57 - pentru poziția de alimentare/descărcare cu ajutorul robotului;

Factorii care determină calitatea unei prelucrări:

- Materia primă folosită;
- Caracteristicile sculei așchietoare:
  - Tipul sculei: freză deget, freză cu cap sferic, freză tip „coadă de rândunică”, freză pentru degroșare, pentru finisare etc.;
  - Materialul din care este făcută scula: oțel carbon, oțel rapid (HSS), carbură, vârfuri de diamant;
  - Diametrul sculei;
  - Modul de ascuțire, numărul de dinți;
- Parametrii tehnologici:
  - Tipul prelucrării: Face milling, Side milling, End milling;
  - Cutting Speed: viteza cu care dintele frezei taie materialul. Valori uzuale pentru freze HSS fără lichid de răcire: oțel -  $20 \div 40$  m/min; aluminiu -  $70 \div 110$  m/min.
  - Feed Rate: viteza de avans a frezei în material [mm/min]. O viteză prea mică poate duce la supraîncălzire, o viteză prea mare poate duce la ruperea frezei.
  - Spindle Speed: viteza de rotație a frezei [rpm]. Se corelează cu viteza de avans pentru a menține constantă dimensiunea așchiilor.
  - Adâncimea de tăiere [mm].
  - Tool Engagement Angle: unghiul de contact dintre freză și material. Este direct proporțional cu efortul resimțit de freză. Dacă valoarea este prea mare, așchiile nu pot fi îndepărtate ușor și astfel apar probleme (supraîncălzire).
  - Sensul de tăiere: *climb milling* (așchiile se formează de la partea mai groasă la partea mai subțire) și *conventional milling* (așchiile se formează de la partea mai subțire la partea mai groasă). De obicei prima variantă dă rezultate mai bune.
- Metoda de răcire utilizată:
  - fără răcire
  - răcire cu aer comprimat (minimal lubrication)
  - răcire cu lichid
- Condiția mașinii (rigiditatea, puterea frezei)

## Realizarea unei piese în regim automat

Robotul va alimenta mașina cu materie primă. Vom folosi piese paralelipipedice cu dimensiuni cunoscute, denumite și *semifabricate* sau *piese brute* (engl. *raw stock*). În urma prelucrării materiei prime pot rezulta fie piese finite, fie alte semifabricate care necesită prelucrări suplimentare.

În acest laborator vom folosi piese din plexiglas transparent, cu dimensiunile  $90 \times 50 \times 15$  mm.

Programul robot va rula în trei etape:

- alimentarea mașinii cu piesa de prelucrat;
- prelucrarea piesei;
- descărcarea mașinii (extragerea piesei prelucrate).

```
.PROGRAM main()
  CALL alimentare()
  CALL prelucrare()
  CALL descarcare()
.END
```

## Interfața cu robotul

Robotul poate controla mașina CNC comunicând cu aceasta prin protocolul DNC, folosind o legătură TCP/IP (Ethernet). Operațiile elementare de interfață sunt implementate în biblioteca de funcții `cnc.monitor`.

Se pot folosi următoarele rutine:

- `cnc.qtest()` ; Quick Test: verifică dacă task-ul de comunicație `cnc.monitor` este activ; în caz contrar, programul robot este oprit forțat. Această rutină se apelează la începutul oricărui program care folosește funcțiile `cnc.*`.
- `cnc.init()` ; Porneste programul `cnc.monitor` pe un task separat (implicit task-ul 4) și inițializează comunicația cu mașina CNC.
- `cnc.ref()` ; Execută referențierea mașinii (procedură similară cu calibrarea roboților).

- `cnc.dopen()` ; Deschide ușa mașinii (Door Open).
- `cnc.dclose()` ; Închide ușa (Door Close).
- `cnc.clamp()` ; Fixează piesa în menghină.
- `cnc.unclamp()` ; Deschide menghina (eliberează piesa).
- `cnc.load($nume.prog)`  
; Încarcă un program în memoria CNC-ului.
- `cnc.exec()` ; Lansează în execuție programul CNC selectat.
- `cnc.refresh()` ; Citește starea mașinii și actualizează semnalele software.
- `cnc.en_aux()` ; Activează Auxiliary Drives <sup>1</sup>.

Toate rutinele sunt blocante, adică returnează controlul după terminarea acțiunii comandate, cu excepția programului `cnc.exec()`, care returnează imediat după ce mașina a început prelucrarea.

Starea mașinii poate fi citită din program folosind semnalele software:

- `cnc.in.busy` ; ON dacă mașina este ocupată (execută un program)  
; OFF dacă mașina este liberă (poate primi comenzi).
- `cnc.in.clamped` ; ON dacă menghina este acționată  
; OFF dacă este deschisă.
- `cnc.in.dopened` ; ON dacă ușa este deschisă complet.
- `cnc.in.dclosed` ; ON dacă ușa este închisă complet.
- `cnc.in.aux` ; ON dacă sunt pornite Auxiliary Drives.
- `cnc.in.ref` ; ON dacă mașina este referențiată  
; (originea mașinii este cunoscută).

Ușa mașinii are trei stări: deschisă, închisă sau într-o poziție intermediară.

Menghina mașinii EMCO 105 se închide și se deschide foarte repede, adică din punct de vedere al robotului are doar două stări: închisă (piesa fixată) și deschisă (piesa eliberată).

---

<sup>1</sup>Oarecum similar cu Enable Power. Auxiliary Drives sunt elementele de acționare pentru ușa și menghina pneumatică, precum și pentru mecanismul de ungere; fără acestea, mașina nu funcționează.

## Alimentarea mașinii CNC

Robotul ia o piesă brută din depozit, o așează în CNC și o fixează în menghină. Alimentarea va fi realizată de un program principal, numit `alimentare()`, care apelează mai multe subrutine pentru operațiile elementare. Subrutinele vor fi gândite astfel încât să poată fi apelate și manual, de la consolă, și de aceea la începutul lor vor fi făcute unele verificări suplimentare.

Restricții:

- Mașina nu se poate mișca decât cu ușa închisă;
- Robotul poate intra în CNC doar atunci când ușa acestuia este deschisă;

Secvența de alimentare:

- La început, robotul verifică starea mașinii:
  - Este programul de comunicație pornit?
  - Este mașina inițializată (referențiată)?
  - Este mașina liberă?
- Robotul extrage piesa brută (semifabricat) din depozit.
- CNC-ul va aduce menghina în poziția de alimentare.
- Robotul deschide ușa mașinii și introduce piesa în CNC.
- Piesa va fi așezată și fixată în menghină:
  - Robotul trimite comanda „deschide menghina” către mașina CNC;
  - Piesa este introdusă în menghină;
  - Se trimite comanda „închide menghina”;
  - Se deschide gripper-ul;
  - Robotul se poate retrage.
- Se execută mișcarea de ieșire din CNC.
- După ce robotul a ieșit, se poate comanda închiderea ușii.
- La sfârșit, robotul se va duce într-o poziție de așteptare (`#safe`).

```
.PROGRAM alimentare()
CALL machine.check()      ; Verificarea si initializarea masinii.
CALL part.pick()          ; Robotul ia piesa bruta,
CALL depl.menghina()      ; CNC-ul pozitioneaza menghina,
CALL intrare.in.cnc()     ; robotul intra in CNC,
CALL part.clamp()         ; fixeaza piesa in menghina,
CALL iesire.din.cnc()     ; iese din CNC și închide ușa.
MOVE #safe
.END
```



**Verificarea inițială a mașinii**

Înainte de a efectua alimentarea mașinii cu o piesă neprelucrată, robotul execută următoarele verificări:

- Se testează starea programului de comunicație cu masina: `cnc.qtest()`;
- Se execută o interogare a stării mașinii;
- Se verifică dacă semnalul `cnc.in.busy` este OFF;
- Se verifică starea motoarelor auxiliare; dacă este necesar, se activează;
- Se verifică dacă mașina este referențiată (calibrată); dacă nu este, se execută procedura Reference Machine și se verifică din nou;
- Dacă oricare din teste eșuează, programul robot este oprit forțat.

```
.PROGRAM machine.check()
  CALL cnc.qtest()
  CALL cnc.refresh();

  IF SIG(cnc.in.busy) THEN
    TYPE "Machine is busy."
    ABORT
  END

  IF NOT SIG(cnc.in.aux) THEN
    CALL cnc.en_aux()
  END

  IF NOT SIG(cnc.in.ref) THEN
    CALL cnc.dclose()
    CALL cnc.ref()
    CALL cnc.refresh()
    IF NOT SIG(cnc.in.ref) THEN
      TYPE "Machine could not be referenced."
      ABORT
    END
  END
END
.END
```

### Mișcarea robotului de la paletă la CNC

Robotul va fi deplasat de la paletă la CNC și invers în poziția „ghemuit”; în caz contrar, acesta ar lovi mașina.

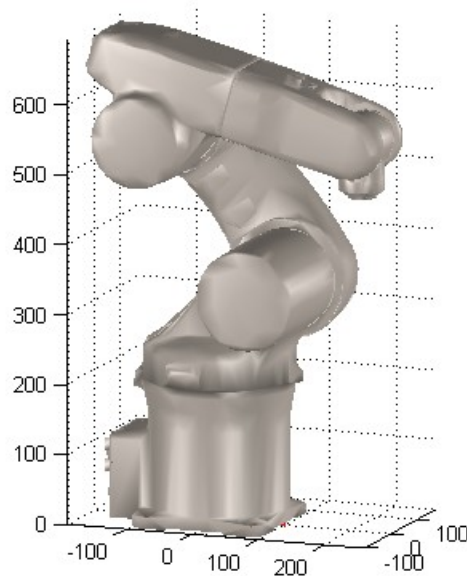


Figura 7.2: *Poziția „ghemuit” a robotului*

Vom învăța trei puncte de precizie:

- `#loc.ghemuit.pal` lângă paletă;
- `#loc.ghemuit.cnc` lângă mașina numerică;
- `#loc.ghemuit.mid` la jumătate (Fig. 7.2).

Mișcarea spre și dinspre `#safe` se va face fie prin `#loc.ghemuit.pal`, fie prin punctul intermediar `#loc.ghemuit.mid`. Dacă se face prin `loc.ghemuit.cnc`, robotul se va apropia foarte mult de mașină, existând riscul unei coliziuni.

Configurația robotului în pozițiile „ghemuit” este `LEFTY / ABOVE / NOFLIP`.

### Extragerea piesei semifabricate din depozit

În aplicația din laborator, piesa va fi prinsă cu gripper-ul robotului în poziție orizontală. În această situație, instrucțiunile APPRO/S și DEPART/S vor lucra pe orizontală (în sensul negativ al axei  $Z_{tool}$ ). Dacă dorim să ridicăm piesa vertical, vom folosi SHIFT(piesa.bruta BY 0, 0, dz) în locul lui DEPARTS.

Punctul de prindere al piesei este piesa.bruta, având  $pitch = 90^\circ$  (Fig. 7.3).

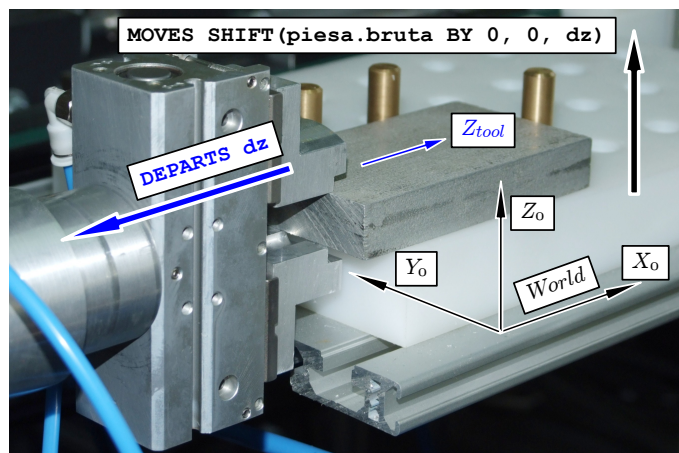


Figura 7.3: Poziția de prindere a piesei neprelucrate

Mișcarea începe și se termină în #loc.ghemuit.pal.

```
.PROGRAM part.pick()
  AUTO x.pick, z.pick
  x.pick = 50
  z.pick = 200
  PARAMETER HAND.TIME = 0.2

  OPEN
  MOVE #loc.ghemuit.pal
  MOVES SHIFT(piesa.bruta BY -x.pick, 0, z.pick)
  MOVES SHIFT(piesa.bruta BY -x.pick, 0, 0)
                                     ; sau APPROX piesa.bruta, z.pick
  MOVES piesa.bruta
  CLOSEI
  MOVES SHIFT(piesa.bruta BY 0, 0, z.pick)
  MOVES #loc.ghemuit.pal
.END
```

### Aducerea menghinei în poziția de fixare a piesei

Pentru ca robotul să poată așeza piesa în mașina CNC, menghina trebuie să se afle într-o poziție cunoscută (cea în care se învață punctul `loc.clamp`).

La sfârșitul fiecărui program de prelucrare ar putea fi inclusă o secvență care aduce mașina în punctul în care robotul va fixa piesa. La prima vedere pare o soluție bună, însă ea garantează poziționarea corectă a menghinei numai după ce se execută un program NC. La pornire, sau după operarea manuală a mașinii, sau după acționarea opririi de urgență, menghina nu mai este poziționată corect.

Soluția preferată este folosirea unui program CNC separat, care are rolul de a deplasa menghina în poziția de alimentare / descărcare. De fiecare dată când robotul va dori să așeze sau să extragă o piesă, va încărca și va rula acest program. Există un mic dezavantaj: se pierde câteva secunde la schimbarea programelor din memoria mașinii, dar se câștigă robustețe.

```
.PROGRAM depl.menghina()
  CALL cnc.qtest()
  CALL cnc.dclose()      ; Usa masinii trebuie sa fie inchisa.
  CALL cnc.load("0000") ; Incarc si execut programul
  CALL cnc.exec()       ; pentru pozitionarea menghinei

; La sfarsitul executiei unui program CNC,
; usa masinii se va deschide automat.

WHILE NOT SIG(cnc.in.dopened) DO
  WAIT          ; asteptare eficienta d.p.d.v. al incarcarii
END            ; procesorului (mai ruleaza si alte taskuri)

.END
```

La sfârșitul programului de prelucrare (G-Code) se va insera un apel către subrutina de aducere a menghinei în poziția de descărcare (identică cu poziția de alimentare). Subrutina se numește `GOTO_CLAMPING_POS`, iar apelul se face imediat înainte de `M30`:

```
; program NC generat de software-ul CAM
...
GOTO_CLAMPING_POS P1
M30
```

Poziția de alimentare este învățată în originea sistemului de coordonate `G57`.

## Intrarea și ieșirea în/din CNC

*Better safe than sorry...*

---

Intrarea în mașina numerică se face plecând din punctul `#loc.ghemuit.cnc`, prin punctul intermediar `#loc.intrare.cnc`. Se poate folosi o mișcare procedurală interpolată pe articulații.

Înainte de a intra în CNC, robotul se va asigura că ușa mașinii este deschisă. Testarea semnalelor într-un mod pasiv, cu instrucțiuni de forma `IF SIG(...)` sau `WAIT SIG(...)`, este riscantă. Comunicația cu mașina CNC se face printr-un program care rulează pe un alt task; dacă apare o eroare, și task-ul respectiv se oprește, semnalele rămân „agățate” pe ultima valoare. La o verificare pasivă a semnalelor, programul robot ar putea vedea că acestea au valoarea așteptată (aici, ușa deschisă), chiar dacă starea semnalelor nu ar mai corespunde cu situația reală.

Rezultatul ar fi coliziunea între robot și CNC.

De aceea, verificarea ușii se va face printr-o fie printr-o interogare *activă* a stării ușii mașinii, fie prin trimiterea unei cereri suplimentare de deschidere a ușii. Astfel, vom apela funcția `cnc.dopen()` chiar dacă ușa este probabil deschisă. Funcția va trimite o cerere mașinii (*deschide ușa!*), iar aceasta va răspunde cu mesajul *ușa este deschisă*.

```
.PROGRAM intrare.in.cnc()
  CALL cnc.qtest()
  CALL cnc.dopen()

  MOVE #loc.ghemuit.cnc
  MOVE #loc.intrare.cnc
.END
```

Ieșirea din CNC se face prin același punct intermediar `#loc.intrare.cnc`. După ce robotul a ieșit complet, se emite comanda de închidere a ușii în timp ce robotul se deplasează din `#loc.ghemuit.cnc` spre `#loc.ghemuit.mid`.

```
.PROGRAM iesire.din.cnc()
  CALL cnc.qtest()
  MOVE #loc.intrare.cnc
  MOVE #loc.ghemuit.cnc
  MOVE #loc.ghemuit.mid
  CALL cnc.dclose()
.END
```

### Fixarea piesei în menghină

Menghina are două bacuri: unul fix (stânga) și unul mobil (dreapta).

Poziția de așezare a semifabricatului *loc.clamp* (Fig. 7.5) este învățată cu piesa atingând bacul fix al menghinei. Dacă piesa ar fi introdusă cu secvența de la Pick and Place (APPRO urmat de MOVES), ar exista riscul ca muchia piesei să lovească menghina, deoarece dimensiunile pieselor brute pot varia puțin, și în plus, repetabilitatea sistemului robot / CNC nu este foarte bună.

Din acest motiv, piesa va fi introdusă în menghină pe o traiectorie similară celei din Fig. 7.4, menținând o distanță *d.clamp* față de bacul fix.

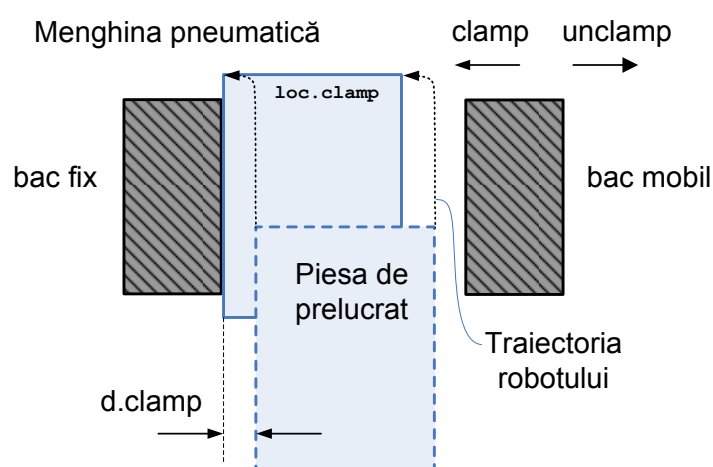


Figura 7.4: Introducerea piesei în menghină

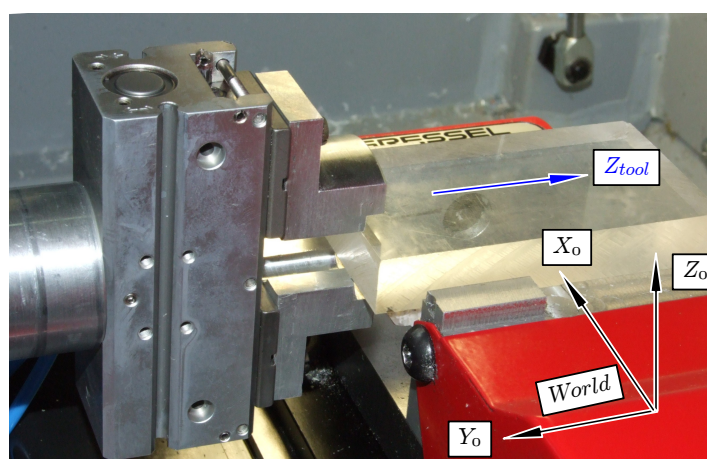


Figura 7.5: Poziția *loc.clamp*

Este important ca piesa să nu fie lăsată liberă. În momentul transferului piesei, ea va fi prinsă simultan și în gripper și în menghină.

```
.PROGRAM part.clamp()
  AUTO d.appro, d.clamp
  d.appro = 150 ; distanta (pozitiva) pt. APPRO / DEPARTS
  d.clamp = -1 ; distanta (cu semn) intre piesa si bacul fix

  CALL cnc.qtest()
  CALL cnc.unclamp()
  APPRO SHIFT(loc.clamp BY d.clamp, 0, 0), d.appro
  MOVES SHIFT(loc.clamp BY d.clamp, 0, 0)
  MOVES loc.clamp
  CALL cnc.clamp()
  OPENI
  DEPARTS d.appro
.END
```

## Încărcarea și execuția programului CNC

Programele care pot fi încărcate pe mașină au numele (`$nume.prog`) format din 4 cifre. Această limitare este prezentă din cauză că mașina CNC folosește pentru comunicație protocolul DNC standard. Protocolul DNC extins permite încărcarea programelor fără restricții privind numele acestora.

Programele care pot fi încărcate pe mașina CNC în acest laborator:

- aduce menghina în poziția de alimentare / descărcare:  
.exec cnc.load("0000")
- pornește freza pentru 5 secunde, în aer, apoi o oprește:  
.exec cnc.load("9999")
- înlătură sigla EMCO de pe piesele de plexiglas transparente  $90 \times 50 \times 15$ :  
.exec cnc.load("1024")
- desenează o față zâmbitoare pe o piesă de plexiglas  $90 \times 50 \times 15$ :  
.exec cnc.load("1025")

Un program nu poate fi rulat pe CNC decât cu ușa închisă. Mai mult, dacă programul CNC rotește freza, programul poate rula doar cu menghina închisă.

Programul 0000 poate rula fără probleme cu menghina deschisă.

Cât timp mașina lucrează, semnalul `cnc.in.busy` este **ON**. Putem spune că mașina a terminat atunci când semnalul devine **OFF**.

După terminarea operației, mașina deschide automat ușa. O metodă alternativă este să așteptăm ca semnalul `cnc.in.dopened` să devină **ON**.

Deoarece mașina ține ușa închisă atunci când lucrează, verificarea ambelor semnale este redundantă. Totuși, dacă programul va verifica ambele semnale, robustețea programului va fi îmbunătățită (puțin).

```
.PROGRAM prelucrare()
  CALL cnc.qtest()

  CALL cnc.dclose()
  CALL cnc.load("1025")
  CALL cnc.exec()

  WHILE NOT SIG(cnc.in.dopened) DO
    WAIT
  END

  IF SIG(cnc.in.busy) THEN
    TYPE "Eroare: masina este ocupata, iar usa este deschisa."
    ABORT
  END
.END
```



Figura 7.6: *Istorie* – (stânga) prima piesă realizată pe mașina EMCO F1 în 2006, folosind funcțiile CAD/CAM din MS Paint ☺; (dreapta) prima piesă decupată.



## Descărcarea mașinii CNC

*Nu forța nimic, adu un ciocan mai mare!*

---

Legea lui Anthony [din Murphy]

După ce mașina numerică a executat programul, robotul va lua piesa prelucrată din CNC și o va așeza într-un depozit.

Secvența de descărcare:

- După ce mașina a terminat, ușa acesteia se deschide automat <sup>2</sup>. Robotul poate intra în CNC să ia piesa.
- Robotul extrage piesa din menghină:
  - se prinde piesa în gripper;
  - se trimite comanda „deschide menghina”;
  - robotul se retrage (DEPARTS) cu piesa prinsă în gripper.
- În acest moment, robotul poate ieși din mașina CNC.
- Piesa prelucrată este așezată la locul ei, iar robotul se retrage în #safe.

### Programul pentru descărcare

```
.PROGRAM descarcare()  
  MOVE #safe  
  MOVE #loc.ghemuit.mid  
  
  CALL intrare.in.cnc()  
  CALL part.unclamp()  
  CALL iesire.din.cnc()  
  
  CALL part.place()  
  MOVE #safe  
.END
```

---

<sup>2</sup>Valabil doar pentru EMCO ConceptMill 105. Mașina EMCO F1 nu are ușa automată, iar la alte mașini poate fi necesară o comandă separată pentru deschiderea ușii.

**Extragerea piesei din menghină**

Această operație se poate realiza pe o traiectorie de tip „pick“ clasică. Putem folosi același punct ca la alimentare (`loc.clamp`) atâta timp cât piesa nu a fost prelucrată în regiunea de prindere.

```
.PROGRAM part.unclamp()
  AUTO d.appro
  d.appro = 150 ; distanta (pozitiva) pt. APPRO / DEPARTS

  CALL cnc.qtest()
  OPEN
  APPRO loc.clamp, d.appro
  MOVES loc.clamp
  CLOSEI
  CALL cnc.unclamp()
  DEPARTS d.appro
.END
```

**Așezarea piesei prelucrate în depozit**

În acest laborator, piesa prelucrată va fi așezată în aceeași poziție cu piesa finită, printr-o operație de tip *place*. Robotul va așeza piesa printr-o mișcare verticală și se va retrage printr-o mișcare orizontală.

```
.PROGRAM part.place()
  AUTO x.pick, z.pick
  x.pick = 50
  z.pick = 200
  PARAMETER HAND.TIME = 0.1

  MOVE #loc.ghemuit.pal

  MOVES SHIFT(piesa.bruta BY 0, 0, z.pick)
  MOVES piesa.bruta
  OPENI
  DEPARTS x.pick
  MOVES SHIFT(piesa.bruta BY -x.pick, 0, z.pick)

  MOVES #loc.ghemuit.pal
.END
```

## Optimizare

Observăm că programele de interfațare cu mașina pot fi îmbunătățite:

- De fiecare dată când dorim să punem o piesă, încercăm și executăm programul de ajustare a poziției menghinei. De cele mai multe ori, aceasta implică o închidere și o deschidere a ușii în plus.

Soluție: putem apela `depl.menghina()` doar atunci când este necesar (doar dacă menghina nu se află acolo unde trebuie). Deoarece la sfârșitul unui program CNC apelăm și subrutina de poziționare a menghinei, aceasta se va afla în alt loc decât cel corect doar în situații speciale:

- înainte primului ciclu de prelucrare;
- după o oprire forțată (Emergency Stop);
- dacă operatorul a deplasat manual axele mașinii între două cicluri;
- dacă se execută mai multe programe de prelucrare, cu poziții de alimentare/descărcare diferite.

Putem interoga mașina și verifica poziția axelor fără a închide ușa.

- Deschiderea și închiderea ușii durează destul de mult (aprox.  $5 \div 10$  secunde). Putem emite comanda de deschidere a ușii încă din momentul în care robotul știe că urmează să pună o piesă. Pentru aceasta vom activa semnalul software `cnc.do.open`<sup>3</sup>. Astfel, ușa mașinii se va deschide în paralel cu mișcarea `pick` a robotului.

Imediat înainte de a intra în CNC, robotul va verifica încă o dată starea ușii (prin interogare/cerere activă, nu doar prin citirea unui semnal).

```
; Cer task-ului de comunicatie sa deschida usa masinii.
SIGNAL cnc.do.dopen
```

```
; Intre timp, task-ul care controleaza robotul
; (acest program) poate face si altceva.
; De exemplu: robotul poate lua piesa din
; depozit si se poate apropia de CNC.
```

```
; Cand robotul a ajuns la intrarea in CNC (dar este
; inca afara), se va asigura ca usa s-a deschis complet:
CALL cnc.dopen()
```

```
; In acest moment, usa este deschisa.
; Robotul poate sa intre în CNC.
```

---

<sup>3</sup>Toate rutinele `cnc.*` au asociat un semnal `cnc.do.*` prin care programul robot poate cere începerea unei acțiuni în mod asincron.

## Comunicație TCP/IP

Deschiderea conexiunii (controller-ul robot este clientul):

```
ATTACH (tcplun, 4) "TCP"
FOPEN (tcplun, 0) "pc_cnc /remote_port 5557 /buffer_size 1024"

IF IOSTAT(tcplun,0) < 0 THEN
    TYPE "Could not connect: ", $ERROR(IOSTAT(tcplun,0))
    RETURN
END
```

Trimiterea unui mesaj ASCII urmat de <CR><LF>:

```
WRITE (tcplun) $msg
```

Trimiterea unui mesaj binar se face apelând WRITE cu parametrul /S, care înseamnă că nu se mai adaugă <CR><LF> la mesaj:

```
WRITE (tcplun) /S, $data
FEMPTY (tcplun)
```

Citirea unui caracter (apel blocant):

```
ch = GETC(tcplun)
```

Citirea unui caracter (apel non-blocant):

```
ch = GETC(tcplun, 1)
IF ch > 0 THEN
    ; caracter valid, il adaug la variabila mesaj $msg
    $msg = $ENCODE($msg, $CHR(ch))
ELSE
    IF ch == -526 THEN
        ; Err. -526: No data received
        ; Mai astept (WAIT) si apelez inca o data GETC
    ELSE
        TYPE "Eroare de comunicatie: ", $ERROR(ch)
    END
END
```

Închiderea conexiunii:

```
FCLOSE(tcplun)
DETACH(tcplun)
```

## Desfășurarea laboratorului

Pornirea mașinii CNC:

- Se deschide valva de aer comprimat;
- Se pornește mașina CNC;
- Se așteaptă aprox. 20 secunde și se deschide programul EMCO WinNC Sinumerik 840D. Se așteaptă zgomotul produs de cuplarea releelor pentru Auxiliary Drives.
- În acest moment, LED-urile de pe panoul de control încep să clipească, iar mașina trebuie inițializată.

Pe robot se execută programul `cnc.init`, care pornește task-ul de comunicație cu mașina numerică.

În acest moment se poate rula programul `main`, care face automat inițializările necesare pentru mașina numerică. Se recomandă să nu se execute direct acest program, ci să se ruleze mai întâi operațiile elementare ale mașinii.

Pentru inițializarea manuală se execută programul `machine.check`.

Se testează operațiile de bază cu mașina CNC de la consola  $V^+$ . Starea mașinii se poate vedea fie în fereastra `CNC.Monitor`, fie citind semnalele software `cnc.in.*`.

```
.exec cnc.clamp          ; inchide menghina
.exec cnc.unclamp       ; deschide menghina
.exec cnc.dopen         ; deschide usa
.exec cnc.dclose        ; inchide usa
.exec cnc.load("9999") ; incarca prog. 9999 care porneste
                       ; freza ptr 5 sec.

.exec cnc.dclose        ; programul merge doar cu usa inchisa
.exec cnc.clamp         ; si menghina actionata,
.exec cnc.exec          ; deoarece prog. 9999 porneste freza.
```

Operațiile de alimentare, prelucrare și descărcare pot fi rulate fie separat, fie prin executarea programului `main`.

Pentru operarea manuală a mașinii este necesară închiderea conexiunii cu robotul (clic în interiorul ferestrei `CNC` din `AdeptWindows` și eventual `ENTER` în consola `Monitor`).

Oprirea mașinii CNC:

- Se oprește comunicația între robot și CNC (clic în fereastra CNC);
- Se opresc motoarele auxiliare (auxiliary drives);
- Se închide aplicația WinNC;
- Se oprește alimentarea mașinii CNC;
- Se închide valva de aer comprimat.

Revenirea din Emergency Stop:

- Se anulează condiția de eroare folosind butonul Reset;
- Se rulează programul `machine.check`.

*Vacanță plăcută! ☺*