

BAB III

KONVERSI FILE STEP-NC KE G & M CODES

3.1 PROSES *MAPPING* DAN KONVERSI.

Proses *mapping* adalah proses memetakan hubungan (korelasi) antara suatu standard dengan standard lain yang berbeda sehingga didapatkan suatu persamaan diantara keduanya, proses *mapping* yang dilakukan adalah antara standard ISO 14649 (STEP) dengan ISO 6983 (G-code) yang berhubungan dengan fitur *round_hole*, sedangkan proses konversi adalah proses pencetakan G-code berdasarkan *rule* hasil *mapping* dengan bantuan program yang dibuat, sekaligus dapat mengidentifikasi file STEP-NC.

Hal pertama kali dilakukan adalah mencari informasi yang dibutuhkan untuk fitur *round_hole* yang ada didalam masing-masing standard baik ISO 6983 maupun ISO 14649, kemudian dicari korelasinya.

3.1.1 Informasi Data ISO 6983 (G-code).

Untuk fitur *round hole* terdapat beberapa operasi yang dapat dilakukan seperti *drilling*, *multistep drilling*, *countersinking*, *reaming*, *centerdrilling*. Berikut ini contoh program untuk proses *drilling*:

```
%  
N01 G90 S636,94 T1 M03      ⇒ parameter mesin  
N02 G54  
N03 G00 X50 Y50 Z3         ⇒ lokasi drilling  
N04 G81 Y3 Z-20 B10 F63,69 ⇒ data drilling  
N05 G00 Z50
```

Dari contoh program diatas terdapat 3 jenis data informasi

1. Data parameter pemesinan seperti *spindle*, *tool*, arah pergerakan *spindle*.
2. Data lokasi dari *drilling* (posisi dalam koordinat *catersian point*).

3. Data proses *drilling* seperti *feeding*, *depth of cut*, *retract*.

3.1.2 Informasi Data ISO 14649 (STEP)

Berdasarkan bagian ISO 14649 maka File STEP-NC terdiri dari bagian-bagian sebagai berikut :

1. *Workpiece definition*
2. *Manufacturing features*
3. *Operations*
4. *Project*
5. *Function/Technology*
6. *Strategies*
7. *Placement/Length*
8. *Tools*

```
ISO-10303-21;
HEADER;
FILE_DESCRIPTION('EXAMPLE OF NC PROGRAMME FOR TURNING; COMPLEX DESIGN');
FILE_NAME('EXAMPLE1.STP');
FILE_SCHEMA('MACHINING_SCHEMA');
ENDSEC;
DATA
(* **** Workpiece definition **** *)
#1=WORKPIECE(SIMPLE WORKPIECE);
#2=MATERIAL(ST-50 STEEL);
#3=PROPERTY_PARAMETER(T=20000NMCZ);
(* **** Manufacturing features **** *)
#12=GENERAL_REVOLUTION(GENERAL_REVOLUTION 1);
#13=ROUND_HOLE(HOLE FLAT BOTTOM);
#14=OUTER_DIAMETER(OUTER_DIAMETER 1);
(* **** Turning operations **** *)
#26=DRILLING(DRILL HOLE);
#27=REAMING(REAM HOLE);
#29=CONTOURING_ROUGH(ROUGH OUTER DIAMETER);
(* **** Project **** *)
#34=PROJECT(TURNING EXAMPLE);
#35=WORKPLAN(MAIN WORKPLAN);
#36=WORKPLAN(WORKPLAN FOR SETUP1);
#37=WORKPLAN(WORKPLAN FOR SETUP2);
#38=MACHINING_WORKINGSTEP(WS ROUGH CIRCULAR_FACE2);
(* **** Functions / Technology **** *)
#60=TURNING_MACHINE_FUNCTIONS(T,F,F,F);
#61=TURNING_TECHNOLOGY(TCP);
#62=CONST_SPINDLE_SPEED(500);
(* **** Strategies **** *)
#130=PLUNGE_RAMP;
#131=DIAGONAL_UNIDIRECTIONAL(T);
#132=DIAGONAL_UNIDIRECTIONAL(F);
(* **** Placements / Lengths **** *)
#103=AXIS2_PLACEMENT_3D(SETUP1);
#104=CARTESIAN_POINT(SETUP1_LOCATION);
#105=DIRECTION_AXIS(1);
#106=DIRECTION_REF_DIRECTION(0);
(* **** Tools **** *)
#280=TURNING_MACHINE_TOOL(T);
#281=GENERAL_TURNING_TOOL(LEFT);
#282=TOOL_DIMENSION(25.5, 7.3, 5.0, 5);
#283=CUTTING_COMPONENT(0);
```

Gambar 3.1 Pembagian grup format ISO 14649 [5]

Didalam proses *mapping* ini urutan struktur atau aliran data disusun sesuai dengan format dan standar dari ISO 10303. Untuk mencari informasi dari fitur *round hole*, terlebih dahulu harus diketahui posisi fitur tersebut didalam kerangka data standar ini. Metode *mapping* yang dilakukan ialah dengan cara mengurutkan secara sequensial standar ISO 14649 ini dari kelas yang tertinggi sampai kelas terendah, beserta atributnya. Kelas tertinggi yakni *project* memuat data-data *workplan*, *machining workingstep*, kelas ini menjadi acuan untuk proses *breakdown* atau proses pencarian dan penelusuran data dari kelas tertinggi sampai kelas terendah yang terkait dalam fitur *round hole*.

File STEP NC diuraikan dan diurutkan berdasarkan no atributnya, dibawah ini adalah gambaran dari proses mengurutkan data:

```

ISO-10303-21;
HEADER;
FILE_DESCRIPTION(...);
FILE_NAME('EXAMPLE1.STP',...);
ENDSEC;
DATA;
(* ***** Workpiece definition ***** *)
#1=WORKPIECE('SIMPLE WORKPIECE',#2,...);
#2=MATERIAL('ST-50','STEEL',(#3));
#3=PROPERTY_PARAMETER(...);
(* ***** Project ***** *)
#20=PROJECT('TURNING EXAMPLE 1',#21,(#1)...);
#21=WORKPLAN('WORK PLAN FOR SETUP1',(#30,#31),...);
#30=MACHINING_WORKINGSTEP('WS DRILLING',...,#40,...);
#31=MACHINING_WORKINGSTEP('WS REAMING',...,#40,...);
(* ***** Manufacturing features ***** *)
#40=ROUND_HOLE('HOLE1 ',#1,(#51),.....);
(* ***** Turning operations ***** *)
#51= DRILLING(...,'DRILL HOLE1',...,#60,#70,#80,... ,#90);
(* ***** Functions / Technology ***** *)
#70= MILLING_TECHNOLOGY(0.030,TCP,.$,16.000,$,F.,F.,F.,$);
#80= MILLING_MACHINE_FUNCTIONS(.T.,$,F.,$,(),.T.,$,,$());
(* ***** Strategies ***** *)
#90= DRILLING_TYPE_STRATEGY(75.000,50.000,5.000,50.000,75.000,40.000);
(* ***** Tools ***** *)
#60= MILLING_CUTTING_TOOL('SPIRAL_DRILL_15MM',#61,...);
#61= TWIST_DRILL(#62,2.,RIGHT.,.F.,0.840);
#62= MILLING_TOOL_DIMENSION(15.000,31.000,0.100,45.000,2.000,5.000,8.000);

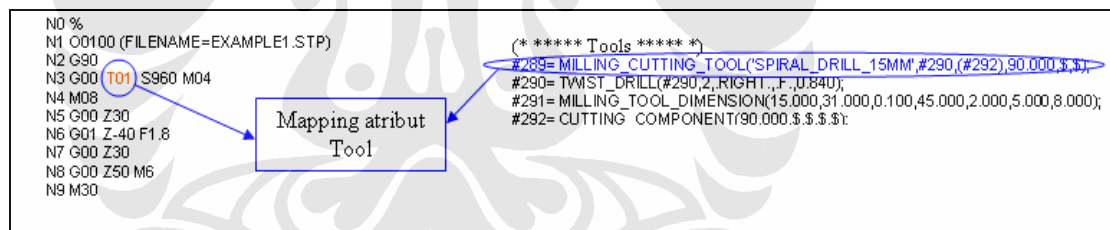
```

Gambar 3.2 Proses mengurutkan data [5]

Setelah diuraikan sampai entitas atau kelas terendah, akan didapatkan aliran data untuk fitur *round hole* dari kelas tertinggi sampai kelas terendah. Kemudian dicari arti dan maksud dari masing-masing atribut yang terdapat pada entitas, sehingga dapat diketahui persamaannya dengan standar G-Code. Tidak semua atribut didalam entitas digunakan hanya yang mempunyai persamaan atau informasi yang sama dengan standar G-Code yang digunakan.

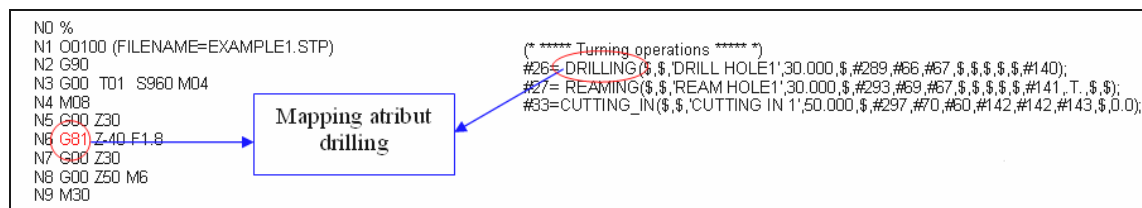
Setelah mencari informasi yang dibutuhkan pada standar ISO 6983 dan ISO 14649, kemudian dicari korelasi antara kedua standar tersebut. Dibawah ini adalah contoh dari korelasi pada fitur *round hole* pada proses *drilling*.

Pada gambar 3.3 menunjukkan korelasi atribut *tool* pada kedua standar, didalam standar G-code direpresentasikan dengan T01, T02 dan seterusnya. Di Step-nc terdapat pada bagian tool, pada proses *drilling* entitas yang digunakan yaitu *milling cutting tool*, informasi yang terdapat pada atribut ini yaitu *tool* yang digunakan, diameter dari *tool*, dimensi *tool* dan lain-lain.



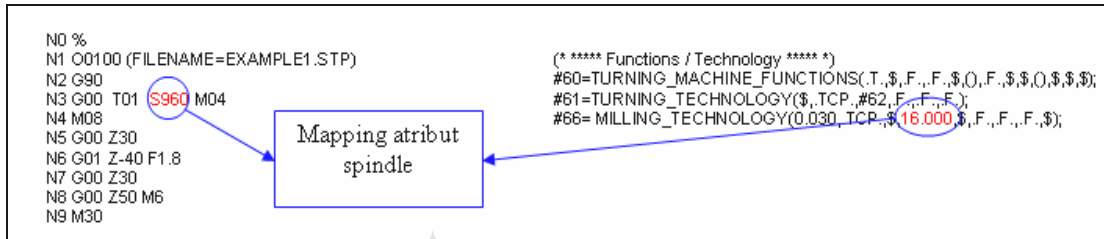
Gambar 3.3 Proses *mapping* atribut *tool*

Untuk korelasi atribut *drilling* pada standar G-code menggunakan fungsi G81, sedangkan pada Step-nc terdapat pada bagian *turning operation* pada entitas *drilling*



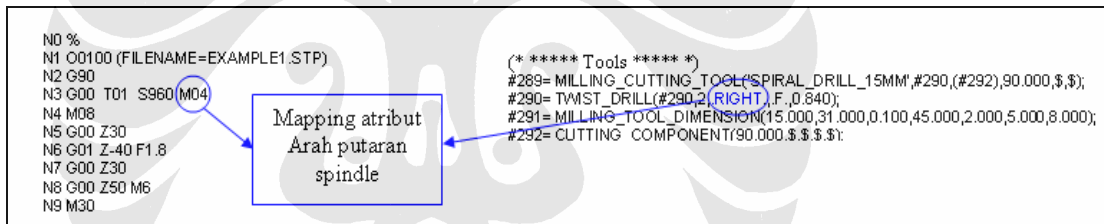
Gambar 3.4 Proses *mapping* atribut *drilling*

Dibawah ini adalah korelasi atribut *spindle* dari standar G-code direpresentasikan dengan fungsi S, sedangkan pada Step-nc entitas yang digunakan adalah *milling technology* dengan atribut *spindle*.



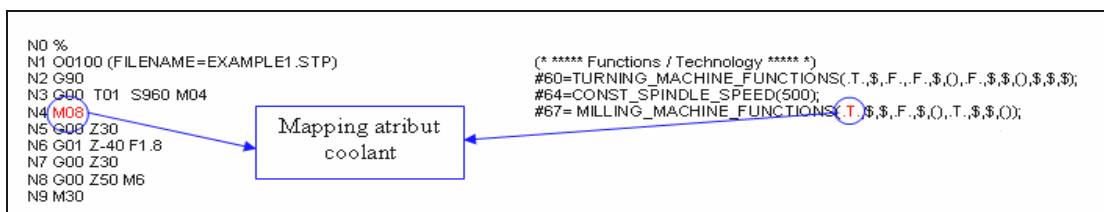
Gambar 3.5 Proses *mapping* atribut *spindle*

Untuk korelasi arah putaran *spindle* pada standar G-code menggunakan fungsi M04/M03 (*Counterclockwise/Clockwise*), sedangkan pada Step-nc terdapat pada entitas *twins drill* atribut *hand of cut* pada atribut ini memberikan informasi *right* atau *left*.



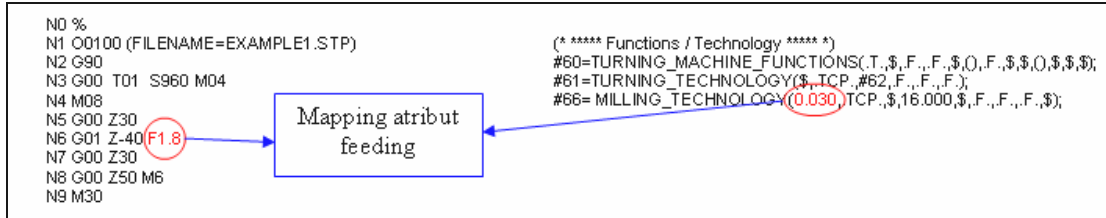
Gambar 3.6 Proses *mapping* atribut arah putaran *spindle*

Untuk korelasi atribut *coolant* pada standar G-code menggunakan fungsi M08/M09 (*Coolant ON/ Coolant OFF*), sedangkan pada Step-nc terdapat pada entitas *milling machine functions* yang atributnya *coolant*, atribut ini memberikan informasi *true* atau *false*.



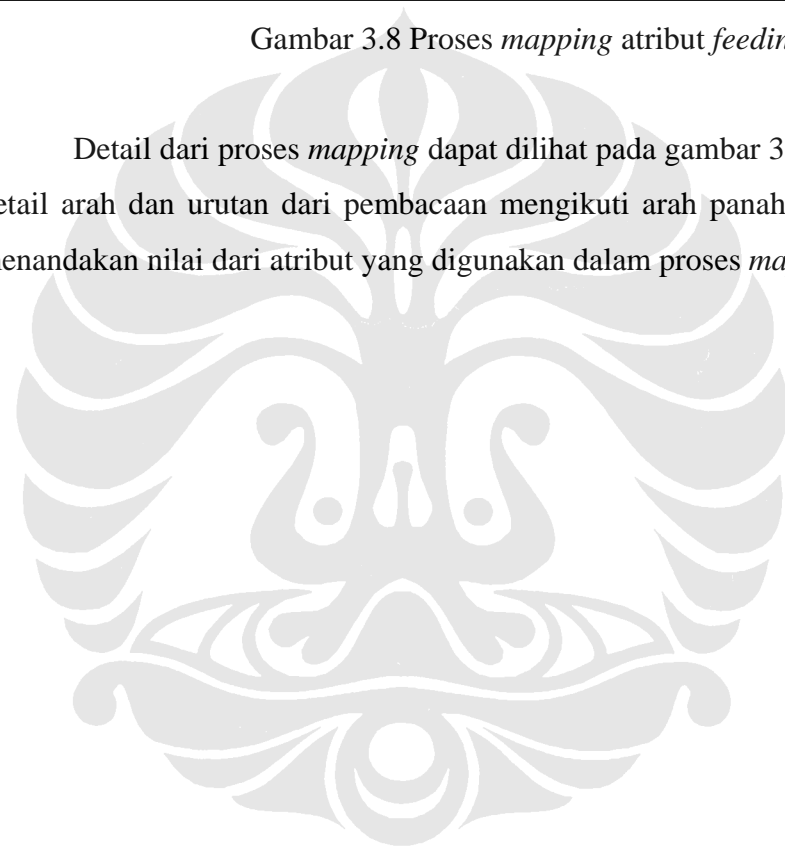
Gambar 3.7 Proses *mapping* atribut *coolant*

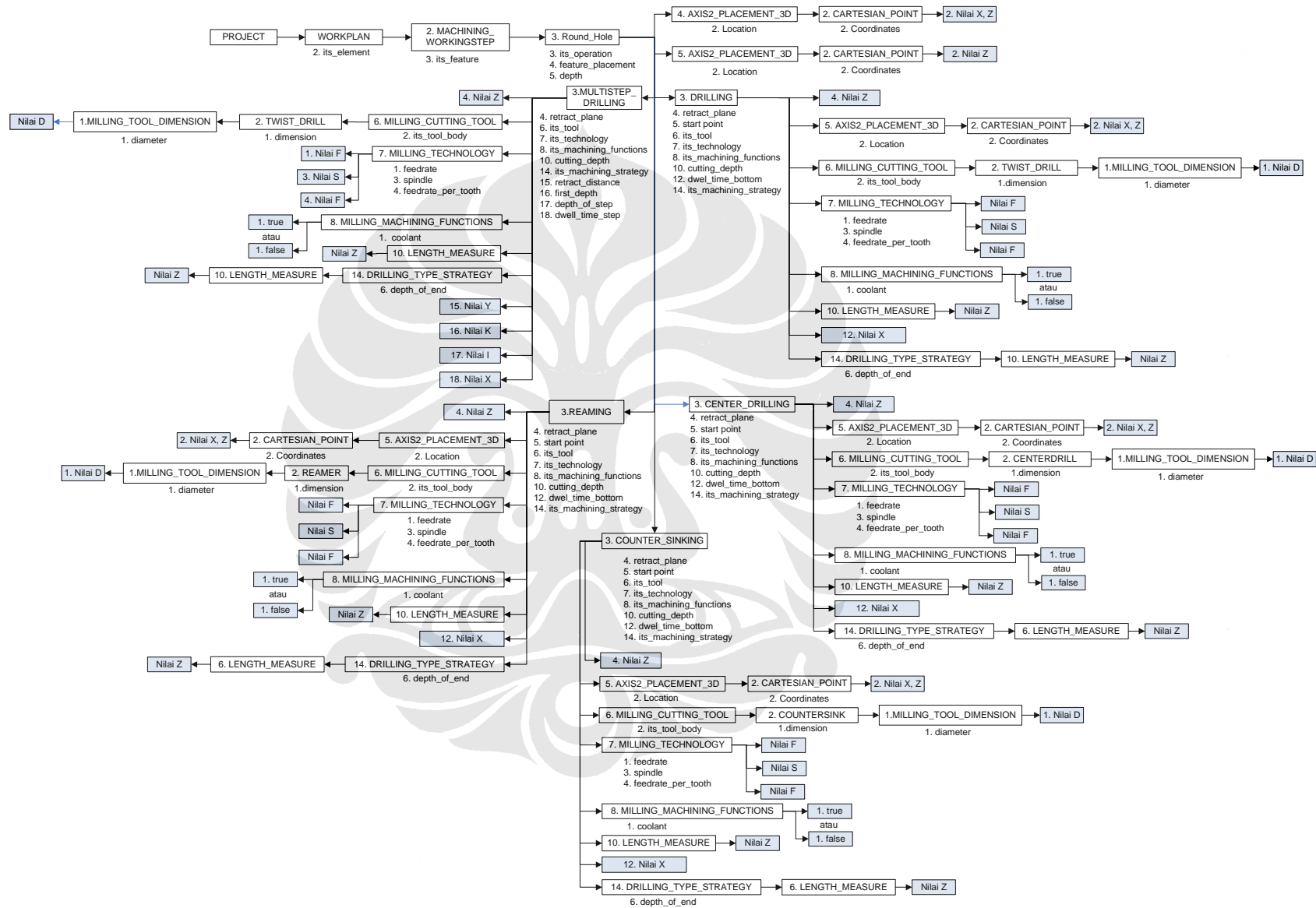
Untuk korelasi atribut *feeding* pada standar G-code menggunakan fungsi F, sedangkan pada Step-nc terdapat pada entitas *milling technology* yang atributnya *feedrate*.



Gambar 3.8 Proses *mapping* atribut *feeding*

Detail dari proses *mapping* dapat dilihat pada gambar 3.9, digambarkan secara detail arah dan urutan dari pembacaan mengikuti arah panah. Kotak berwarna biru menandakan nilai dari atribut yang digunakan dalam proses *mapping*.





Gambar 3.9 Diagram proses *mapping*

3.2 DATABASE

Dalam sistem informasi berbasis komputer *database* merupakan kumpulan data yang distrukturkan sehingga memungkinkan kemudahan untuk menyimpan, dan data yang disimpan tersebut dapat diambil atau digunakan. Sistem manajemen *database* DBMS (*Database Management System*) merupakan *software* yang digunakan untuk mengelola *database*, dari pendefinisian *database*, pengelolaan data, *maintenance* data, sampai dengan pengaturan siapa saja yang dapat mengakses dan menggunakan *database* tersebut.

Sistem manajemen *database* relasional RDBMS (*Relational Management Database System*), merupakan salah satu model dari suatu system manajemen *database*. Beberapa RDBMS yang terkenal saat ini : MySQL, Oracle, Sybase, MS SQL Server, Interbase, dan MS access. Sistem *database* yang disebutkan tadi menggunakan bahasa SQL untuk mengoperasikannya. Pembuatan *database* dalam tulisan ini menggunakan MySQL dengan bahasa SQL sebagai pengoperasiannya.

Database yang dibuat adalah *database* yang terdiri dari tabel-tabel, dimana setiap tabel memuat data atribut dari suatu entitas. Data atribut disimpan dalam baris-baris dari suatu kolom. Sehingga nama tabel adalah nama entitas yang atributnya akan dimasukkan ke dalam tabel dan jumlah kolom akan sama dengan jumlah atribut dari entitas tersebut. Berikut adalah contoh dari tabel yang dibuat berdasarkan suatu entitas.

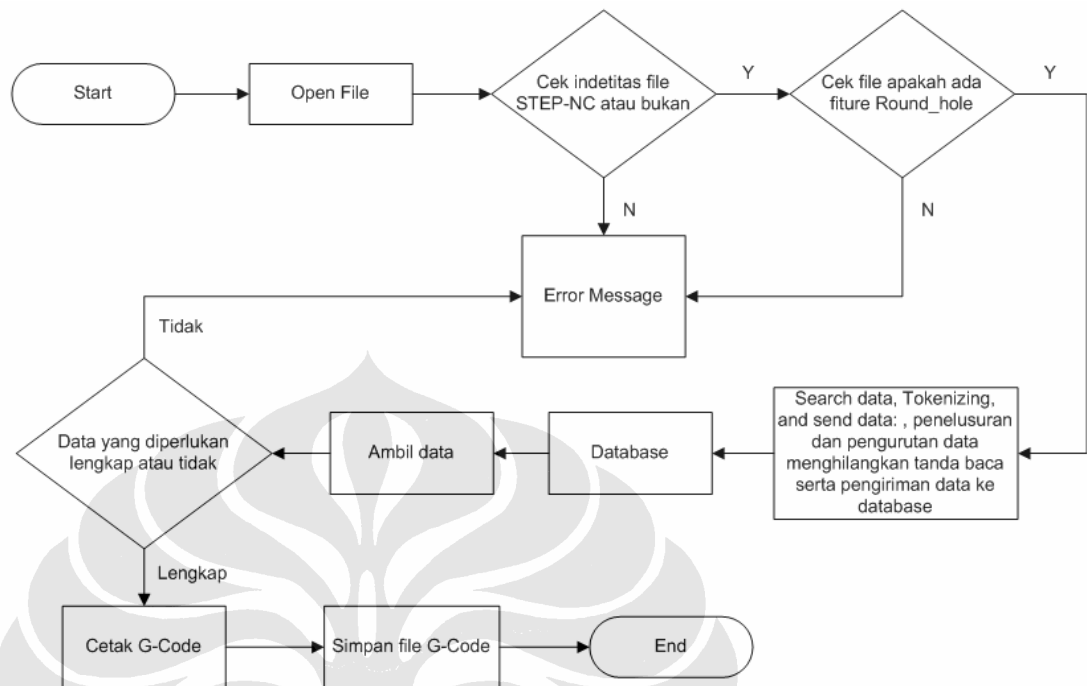
	its_id	dimension	number_of_teeth	hand_of_cut	cooland_throught_tool	pilot_length
	SPIRAL_DRILL_15MM	MILLING_TOOL_DIMENSION 2		.RIGHT.	.F.	0.840
*		(NULL)	(NULL)	(NULL)	(NULL)	(NULL)

Gambar 3.10 Tabel *Twist drill*

	its_id	reduced_cut_at_start	reduced_feed_at_start	depth_of_start	reduced_cut_at_end	reduced_feed_at_end	depth_of_end
	DRILL HOLE1	75.000	50.000	5.000	50.000	75.000	40.000
	REAM HOLE1	¢	¢	¢	¢	¢	¢
*		(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)

Gambar 3.11 Tabel *Drilling type strategy*

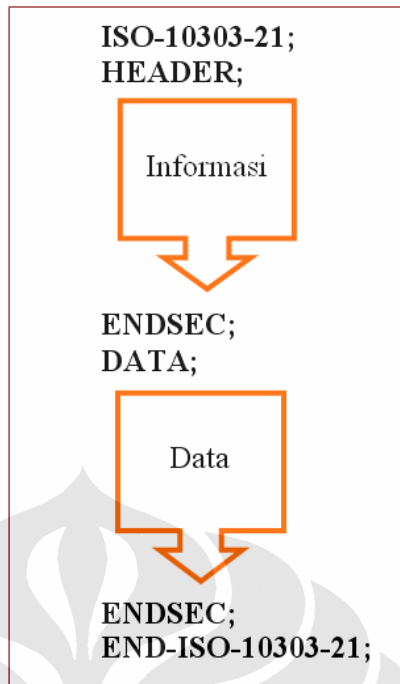
3.3 SOFTWARE KONVERSI



Gambar 3.12 Algoritma pemrograman *software* konversi

Setelah membangun *database*, langkah selanjutnya adalah membuat *software konversi*, urutan proses konversi mengacu kepada algoritma pemrograman, *software konversi* yang telah dibuat melalui tahapan-tahapan sebagai berikut:

1. Sebelum melakukan proses konversi file tersebut diidentifikasi apakah file merupakan file STEP-NC atau bukan, file STEP-NC memiliki bagian yang dapat digunakan sebagai identitas file yaitu bagian *header* dan data:
 - File selalu diawali dengan ISO-10303-21; dan diakhiri dengan END-ISO-10303-21;
 - Bagian *Header* diawali dengan: *HEADER*; Diakhiri dengan: *ENDSEC*;
 - Bagian data diawali: *DATA*; Diakhiri dengan: *ENDSEC*;



Bila file ada identitas tersebut maka program akan dilanjutkan ketahap selanjutnya jika tidak proses berhenti dan memberikan tanda bahwa file bukan file STEP-NC. Setelah itu program akan mengecek apakah didalam file tersebut terdapat fitur *round hole*, jika ada maka program akan melanjutkan ketahap selanjutnya, jika tidak ada proses akan berhenti dan memberikan tanda file tidak terdapat fitur *round hole*.

2. Program melakukan proses *Tokenizing*, yaitu menghilangkan tanda baca yang tidak diperlukan, seperti '#', ';', '(', '=', sehingga data dalam file tidak memiliki tanda baca yang telah *ditoken*. setelah melalui proses *tokenizing* kemudian data dikirim ke *database*.
3. *Search and send data*, pola ini diadopsi dari penelitian sebelumnya [8]. Pada tahapan ini proses penelusuran entitas dan atributnya dari kelas tertinggi sampai kelas terendah. Setelah melalui proses ini kemudian data dikirim ke *database*.



Gambar 3.13 Urutan proses *search and send data*

4. *Take and processing data*, pola ini diadopsi dari penelitian sebelumnya^[8]. Pada tahap ini adalah proses pengambilan data atribut dari *database* yang diperlukan untuk mencetak G-code. Pada proses ini mengacu pada *rule take and processing data*, yaitu sebagai berikut:

- *Rule Main_program* (lihat rincian rule pada lampiran)
- *Rule Operation* (lihat rincian rule pada lampiran)

3. (1). *Rule main program*

Suatu program G-code adalah serangkaian blok-blok yang terdiri dari :

✓ *Blok start*

- %
- *nama_file* : O0100 (nama file)
- *absolute_value* atau *incremental_value* : G 90 / G 91
- *tool* yang dipakai : T01
- *Spindle speed* : S “value”
- arah putaran spindel : M03 / M04
- status *coolant* : M08 / M09

✓ *Blok command*

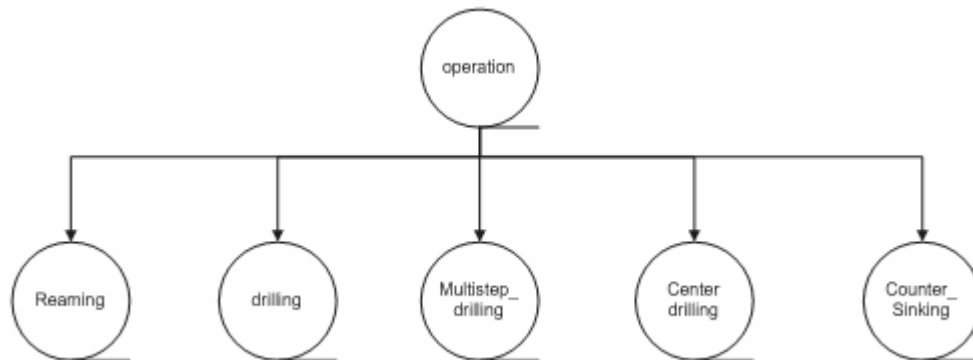
Operation (blok pergerakan *tool*)

✓ *Blok end*

- *Program end* : M30
- %

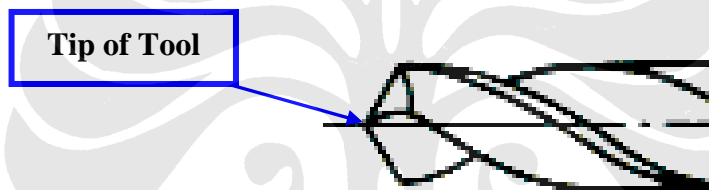
3. (2). *Rule Operation*

Rule operation adalah *rule* yang memuat pergerakan *tool* proses pemakanan. Pada *rule* ini menentukan operasi yang akan dilakukan pada fitur *round hole* seperti *drilling*, *multistep_drilling*, *center_drilling*, *reaming* dan *countersinking*.



Gambar 3.14 Operation

Hal yang perlu diingat posisi tool dalam operasi untuk *turning* adalah bahwa titik/posisi tool (x dan z) berada pada titik ujung tool.

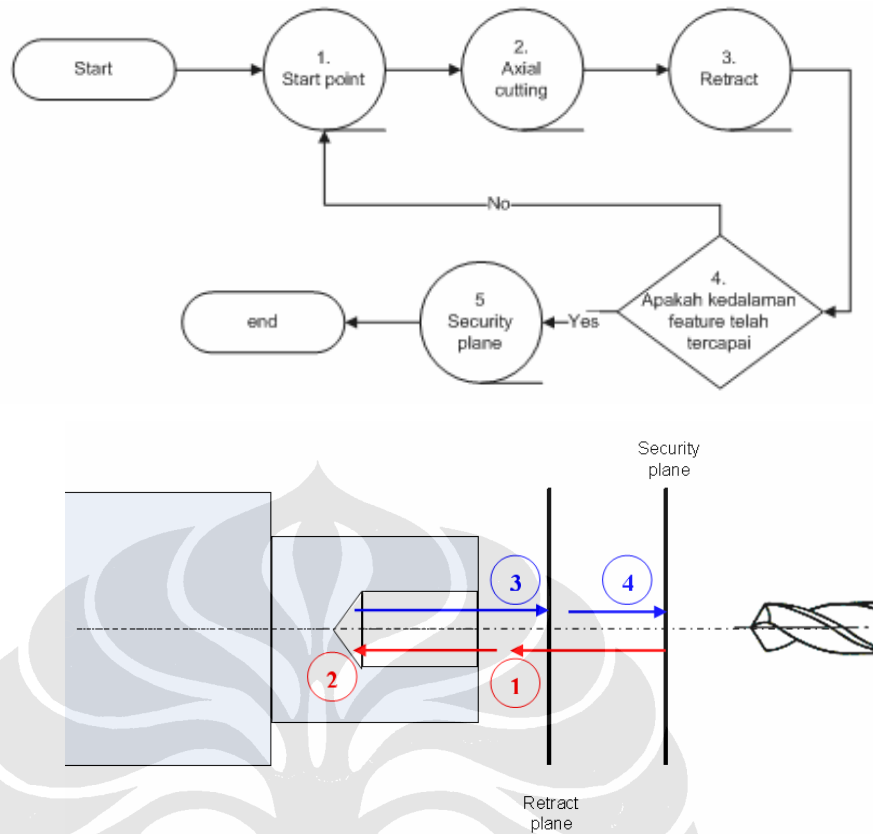


Gambar 3.15 Titik posisi ujung tool

Proses pemesinan yang akan dilakukan pada fitur *round hole* memiliki rule operasi sebagai berikut:

- Rule Drilling (lihat rincian pada lampiran)
- Rule Multistep_drilling (lihat rincian pada lampiran)
- Rule Center_drilling (lihat rincian pada lampiran)
- Rule Reaming (lihat rincian pada lampiran)
- Rule Counter_sinking. (lihat rincian pada lampiran)

Tahapan-tahapan pergerakan tool proses *center_drilling*, *reaming*, dan *counter_sinking* terdapat kesamaan dengan tahapan pergerakan tool pada proses *drilling*, tahapan pergerakan tool proses *drilling* dapat dilihat pada gambar 3.16.

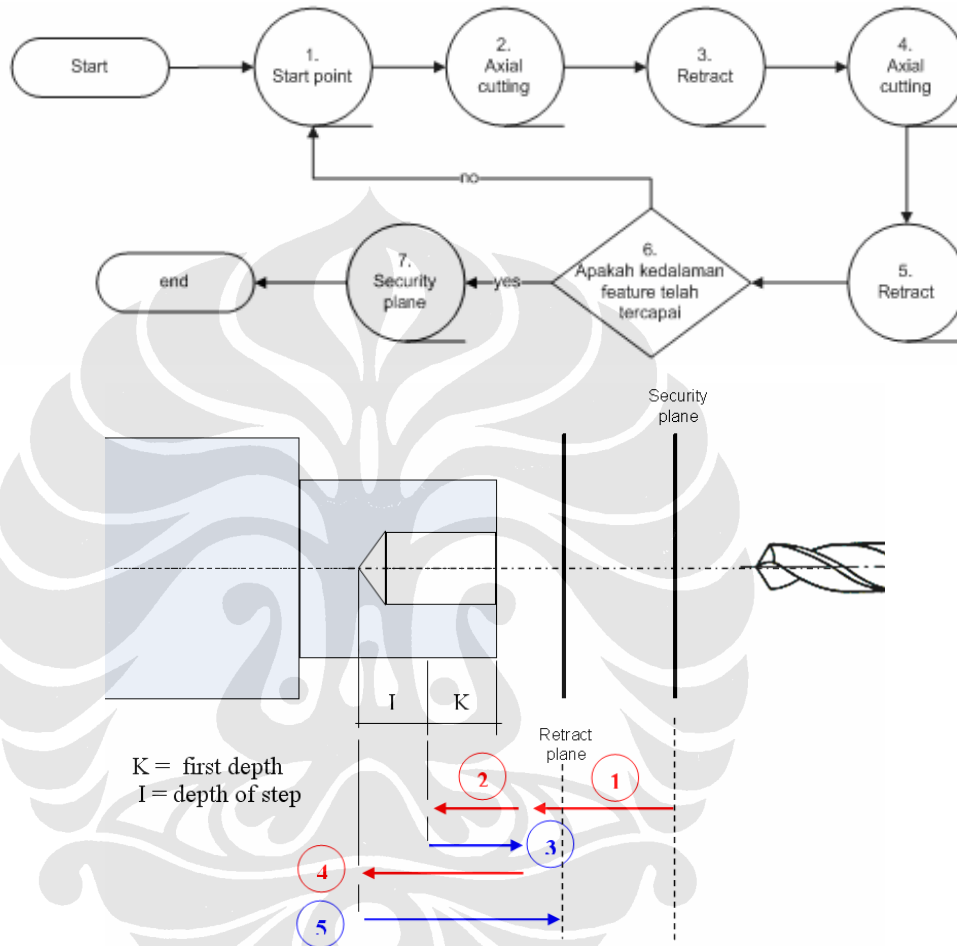


Gambar 3.16 Skema pergerakan *tool* proses *drilling*

1. Posisi awal *tool* berada pada bidang aman (*security plane*), kemudian *tool* bergerak searah sumbu z menuju *start_point* untuk mendekati benda kerja. Apabila data *start_point* bernilai *optional* (\$) maka data untuk mendekati benda kerja diambil dari data *retract plane*.
2. Tahap selanjutnya yaitu proses pemakanan *axial* (*axial cutting*) adalah pemakanan searah dengan sumbu benda kerja atau sumbu z, besarnya nilai *axial cutting* diambil dari data *cutting depth*. Apabila data *cutting depth* bernilai *optional* (\$) maka data diambil dari data *depth of end*.
3. Setelah melakukan proses pemakanan *axial* maka akan diakhiri oleh proses *retract*, yakni proses penarikan *tool* menuju bidang *retract plane*. Nilai *retract* diambil dari *entity drilling* dengan mengambil data dari atribut *retract plane*.
4. Tahap ini memperhitungkan apakah kedalaman dari proses *drilling* telah tercapai atau tidak, apabila kedalaman *drilling* belum tercapai maka *tool* akan kembali ketahap 1 dan terus-menerus melakukan *looping* sampai kedalaman proses *drilling* tercapai.

5. Posisi *tool* selanjutnya akan bergerak menuju bidang aman (*security plane*).

Sedangkan tahapan-tahapan pergerakan *tool* pada proses *multistep drilling* dapat dilihat pada gambar dibawah ini.



Gambar 3.17 Skema pergerakan *tool* proses *multistep drilling*

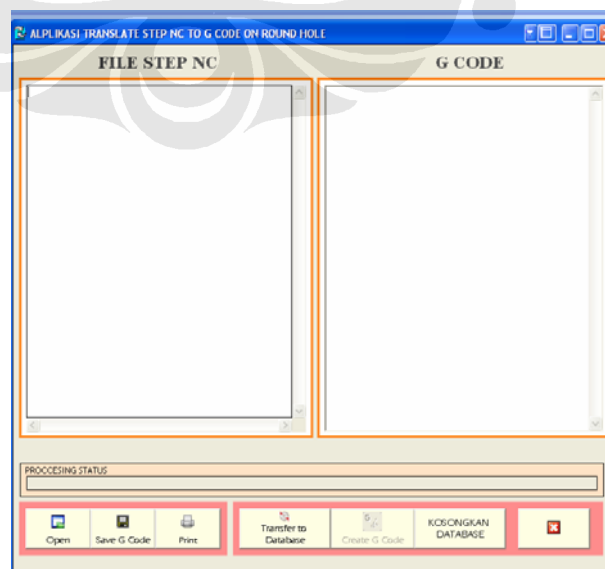
1. Posisi awal *tool* berada pada bidang aman (*security plane*), kemudian *tool* bergerak searah sumbu z menuju *start_point* untuk mendekati benda kerja. Apabila data *start_point* bernilai optional (\$) maka data untuk mendekati benda kerja diambil dari data *retract plane*.
2. Tahap selanjutnya proses pemakanan axial (*axial cutting*) yaitu pemakanan awal (*first depth*) adalah pemakanan searah dengan sumbu benda kerja atau sumbu z, besarnya nilai *axial cutting* diambil dari entitas *multistep_drilling* dengan mengambil data dari atribut *first depth*.

3. Setelah melakukan proses pemakanan *axial* atau *first depth* maka *tool* akan bergerak menuju *retract distance*. Nilai *retract* diambil dari entitas *mutistep drilling* dengan mengambil data dari atribut *retract distance*.
4. Selanjutnya *tool* akan melakukan proses pemakanan *axial* yaitu pemakanan dengan kedalaman *depth of step*, pemakanan dengan kedalaman *depth of step* ini dilakukan secara bertahap sampai kedalaman lubang yang diinginkan tercapai.
5. Setelah melakukan proses pemakanan *axial* maka akan diakhiri oleh proses *retract*, yakni proses penarikan *tool* menuju bidang *retract plane*. Nilai *retract* diambil dari entitas *drilling* dengan mengambil data dari atribut *retract plane*.
6. Tahap ini memperhitungkan apakah kedalaman dari proses *drilling* telah tercapai atau tidak, apabila kedalaman *drilling* belum tercapai maka *tool* akan kembali ketahap 1 dan terus-menerus melakukan *looping* sampai kedalaman proses *drilling* tercapai.
7. Posisi *tool* selanjutnya akan bergerak menuju bidang aman (*security plane*).

3.4. PENGOPERASIAN PROGRAM KONVERSI.

Dibawah ini adalah tampilan program dan cara mengoprasikannya


1. Setelah aplikasi terbuka maka akan muncul jendela program berikut :



Gambar 3.18 Tampilan program konversi

Pada jendela program tersebut terdiri dari :

- ✓ *Textbox* FILE STEP-NC : adalah *textbox* yang akan berisi file STEP-NC
- ✓ *Textbox* G-Code : adalah *textbox* yang akan berisi file G-code hasil konversi

✓  *Button open* file STEP-NC.

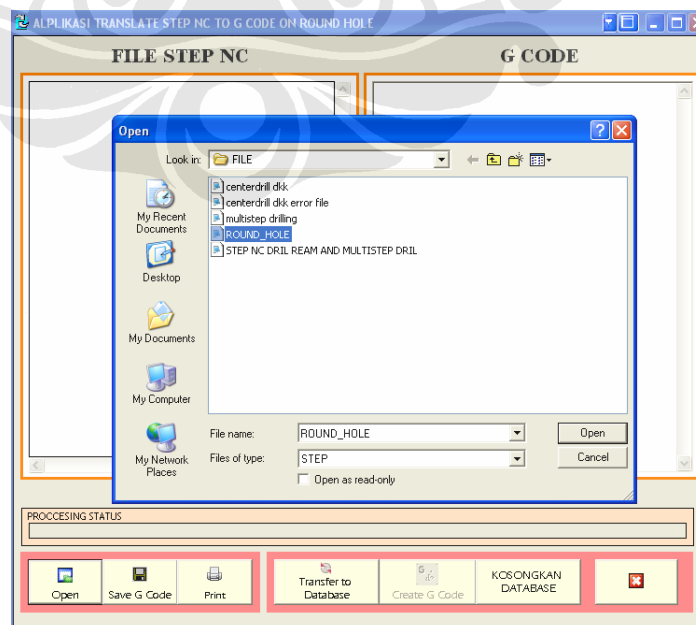
✓  *Button save* G-Code.

✓  *Botton* kosongkan database.

✓  *Button create* G-Code.

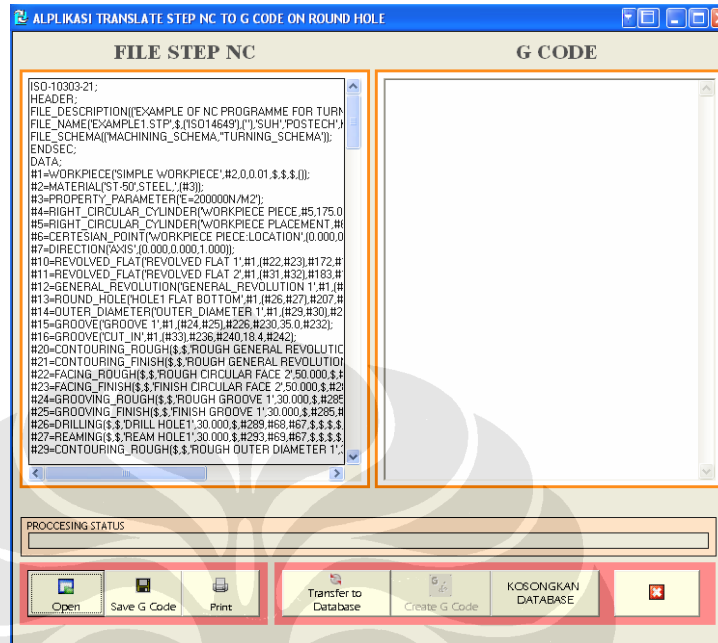
✓  *Button transfer to* database.

2. Langkah selanjutnya adalah menekan *button open file* untuk membuka *file* Step-NC, maka akan muncul jendela dialog sebagai berikut :



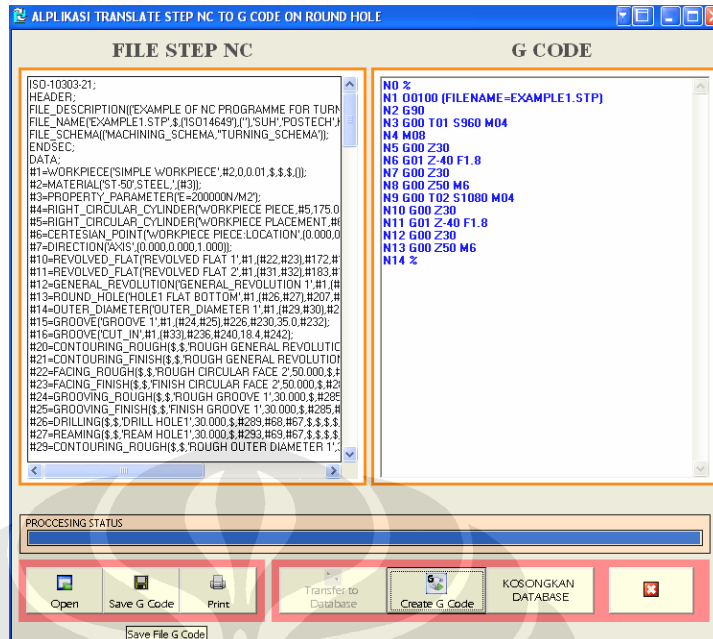
Gambar 3.19 Tampilan apabila mengklik ikon *open file*

- Setelah memilih file lalu tekan *open*, maka *textbox* STEP-NC akan berisi file STEP-NC, seperti pada gambar berikut:



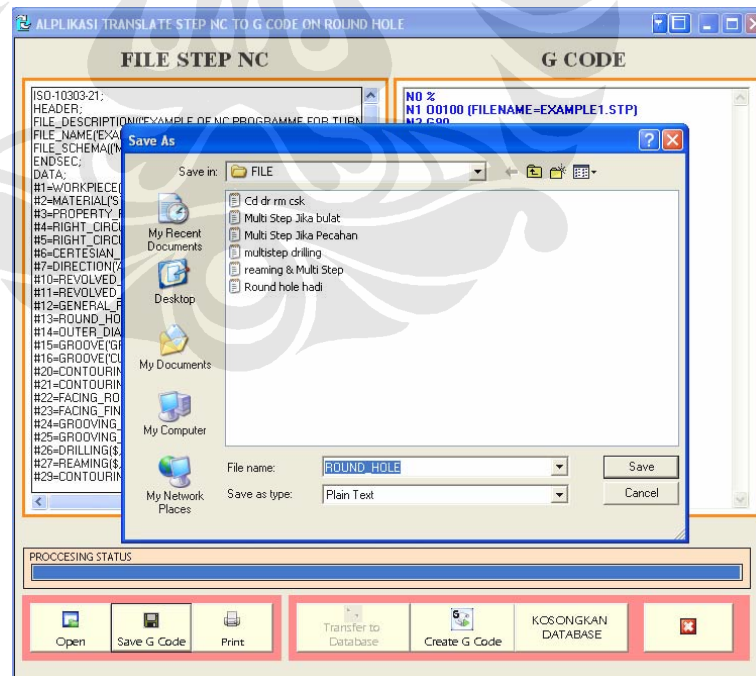
Gambar 3 .20 Tampilan setelah mengklik *open file*

- Langkah selanjutnya adalah melakukan transfer data ke *database*, dengan mengklik tombol “*TRANSFER DATA TO DATABASE*”, status proses sedang berlangsung dengan tanda dari *progress bar*, jika proses telah selesai maka akan muncul *message box report* yang menerangkan bahwa proses telah selesai, tekan ok pada *message box*.
- Langkah selanjutnya adalah mencetak G-code, dengan mengklik tombol “*MAKE G-CODE*”, status proses sedang berlangsung dengan tanda dari *progress bar*, jika proses telah selesai maka pada *textbox* G-code akan terisi dengan hasil G-code.



Gambar 3.21 Tampilan cetak G-code

- Untuk menyimpan file G-code, dengan mengklik *icon save*, maka akan muncul jendela dialog *save*, berikan nama file dan tentukan letak file kemudian klik *save*.



Gambar 3.22 Tampilan proses menyimpan file G-code

- Langkah terakhir adalah keluar dari program, dengan mengklik tombol. 