

PERANCANGAN SISTEM PENGKODEAN FITUR PRODUK (*CODING*) DENGAN MENGGUNAKAN *SOFTWARE PRO/ENGINEER*

Sunardi Tjandra, Arum Soesanti

Program Studi Teknik Manufaktur, Fakultas Teknik
Universitas Surabaya
Jln. Raya Kalirungkt, Surabaya 60293, Indonesia
shiong_master@yahoo.com

Abstrak

Salah satu tahapan dalam perancangan dan pengembangan produk adalah penggambaran komponen. Proses penggambaran yang dilakukan ada dua jenis, yaitu: pembuatan gambar baru dan pengembangan dari gambar yang sudah ada. Untuk menggambar komponen yang bentuk dan dimensinya hampir sama dengan gambar sebelumnya, desainer harus mencari file gambar tersebut berdasarkan nama gambarnya. Hal ini membutuhkan waktu lama sehingga dapat meningkatkan biaya perancangan. Oleh karena itu, dirancang sebuah program dengan mengaplikasikan sistem coding Group Technology, dimana komponen yang mempunyai kemiripan bentuk dan dimensi, dikelompokkan berdasarkan kesamaan fitur dasarnya. Jenis komponen yang menjadi obyek penelitian ini adalah poros alat/mesin pengolah bahan makanan. Poros dimodelkan dengan menggunakan Pro/Engineer berdasarkan fitur-fitur yang diperlukan, yaitu: bentuk poros (L/D), dengan/tanpa groove, dan jumlah tingkatan poros. Langkah selanjutnya, menentukan digit dari sistem coding berdasarkan penggolongan kemiripan fitur dasar dari masing-masing poros. Kemudian dilakukan pembuatan program dengan menggunakan Pro/Program, sehingga desainer hanya perlu menginput 3 digit kode sesuai dengan poros yang diinginkan. Software akan menampilkan model 3D dan gambar kerja dari poros tersebut, dan selanjutnya desainer dapat melakukan modifikasi lebih lanjut. Dengan program ini, desainer lebih mudah dalam membuat gambar komponen yang mempunyai kemiripan fitur dasar tanpa harus menggambar dari awal sehingga dapat menghemat waktu dan tenaga serta mendukung proses pengembangan produk.

Kata kunci : coding, program, poros, perancangan produk

1. PENDAHULUAN

Dalam melakukan suatu perancangan dan pengembangan produk, banyak sekali tahapan-tahapan yang harus dilakukan, salah satunya adalah pembuatan gambar *part*/komponen. Pada tahapan ini dilakukan penggambaran komponen secara manual (menggunakan kertas dan alat meja gambar) ataupun menggunakan bantuan *software*

CAD, baik berupa gambar 3D model maupun gambar kerja (2D). Jenis proses penggambaran yang dilakukan ada dua jenis, yaitu: pembuatan gambar baru dan pengembangan dari gambar lama, yang sebelumnya sudah dalam database desainer. Beberapa komponen standar memiliki varian-varian berdasarkan kemiripan bentuk dan ukurannya. Selama ini, apabila ingin menggambar komponen baru yang bentuk

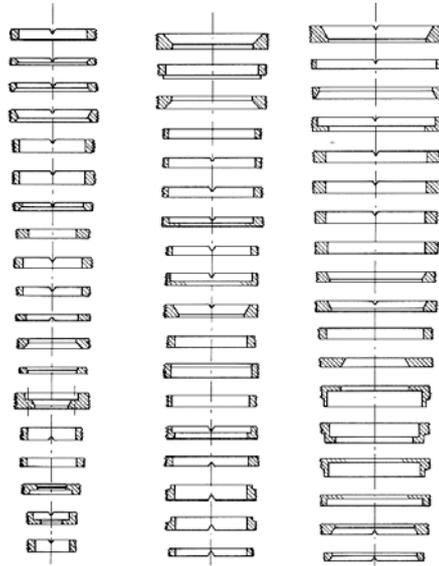
dan dimensinya hampir sama dengan komponen yang sudah digambar sebelumnya, desainer harus mengingat apakah sudah pernah melakukan penggambaran untuk komponen tersebut. Disamping itu, desainer juga harus mencari *file* gambar tersebut berdasarkan nama gambar yang sudah diberikan sebelumnya. Setelah menemukan *file* gambar yang diinginkan, barulah desainer dapat melakukan penggambaran ataupun pengembangan dari file gambar tersebut. Proses mengingat, mencari *file* gambar berdasarkan nama komponen, membutuhkan waktu yang cukup lama sehingga akan meningkatkan biaya perancangan dan pengembangan produk. Apalagi jika komponen yang akan dirancang berjumlah banyak dan mempunyai kemiripan bentuk dasar dan ukuran.

Untuk mengatasi permasalahan tersebut, maka dirancang sebuah program dengan mengaplikasikan sistem *coding* dari *Group Technology* (GT). Untuk merancang program ini digunakan bantuan *software Pro/Engineer 2001* dan *Pro/Program*.

2. KAJIAN LITERATUR

Sistem *coding* dalam *Group Technology* sangat membantu dalam mengetahui golongan suatu komponen, sehingga desainer dapat dengan mudah mengidentifikasi komponen yang sudah ada

maupun yang baru. Hal ini dikarenakan dalam sistem tersebut, semua komponen yang mempunyai kemiripan bentuk dan dimensi akan digolongkan. Gambar 1 menunjukkan suatu golongan komponen.



Gambar 1. *A Design Family*
[source: Chang, Wang and Wysk, CAM, pub.: prentice-hall]

Pada dasarnya *Group Technology* adalah pemahaman bahwa beberapa masalah adalah sama, masalah yang sama tersebut dikelompokkan lalu sebuah pemecahan tunggal dibuat untuk mengatasi hal tersebut sehingga menghemat waktu dan tenaga. Dalam aplikasinya *Group Technology* bermanfaat untuk mengoptimasi perencanaan dalam proses manufaktur karena bentuk yang sama mengacu pada proses manufaktur sama untuk material yang sama. Selain itu dapat dimanfaatkan pada bidang sistem manufaktur yang lain yaitu : *engineering design, layout planning, production control, quality control*.

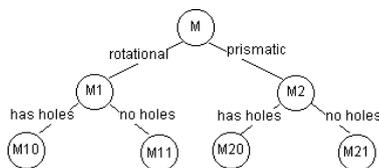
Manfaat pada *engineering design* antara lain :

- reduksi dalam perancangan komponen yang baru
- reduksi jumlah gambar drawing melalui standarisasi
- mereduksi usaha untuk menggambar ulang produk yang sudah ada
- mereduksi jumlah part sama karena memiliki kemudahan untuk diingat kembali

Semua manfaat tersebut tidak terlepas karena adanya *coding*. *Coding* adalah pemberian sekumpulan angka, huruf atau simbol untuk mewakili suatu informasi. Sistem *coding* pada *Group Technology* dibagi menjadi 3 macam yaitu :

- *Monocode*

Sistem pengkodean secara hirarki dimana digit pertama menunjukkan perwakilan dari sebuah kelompok, digit kedua mewakili sub-kelompok tersebut, demikian seterusnya (gambar 2).



Gambar 2. Sistem *Monocode*

- *Polycode*

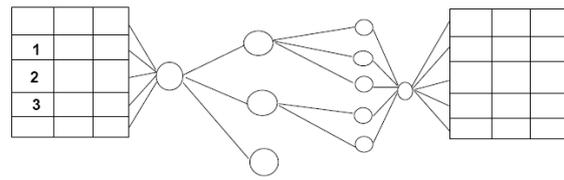
Sistem pengkodean dimana satu angka mewakili satu macam informasi, dan digit yang satu dengan digit yang lain tidak berhubungan (gambar 3).

		digit			
		1	2	3	
value	1				
	2				
	3				

Gambar 3. Sistem *Polycode*

- *Mixed-mode code*

Sistem pengkodean dimana keuntungan dari sisten *monocode* dan *polycode* dipertahankan dan digabungkan. Hal tersebut menyebabkan banyak sistem *coding* yang menggunakan jenis ini (gambar 4).



Gambar 4. Sistem *Mixed Code*

Sistem *Opitz* merupakan salah satu sistem *coding* yang didasarkan pada sistem *Mixed Code*. Sistem ini dikembangkan oleh H. Opitz dari *Technical University of Aachen* didukung oleh *German Machine Tool Association*. Kode dari sistem *Opitz* terdiri dari kode geometri (bentuk) dan kode tambahan (suplemen). Geometri kode mewakili komponen dalam variasi : *rotational, flat, long, cubic*. Selain itu yang juga tergolong dalam kode geometri adalah perbandingan dimensi dan tambahan bentuk seperti lubang. Sedangkan kode tambahan berisikan kode yang mewakili diameter

mayor dari benda, tipe material, bentuk bahan baku dan ketelitian. Sistem *Opitz* jelas dan mudah untuk digunakan sehingga digunakan oleh beberapa perusahaan untuk mengkode sub-sistem mereka.

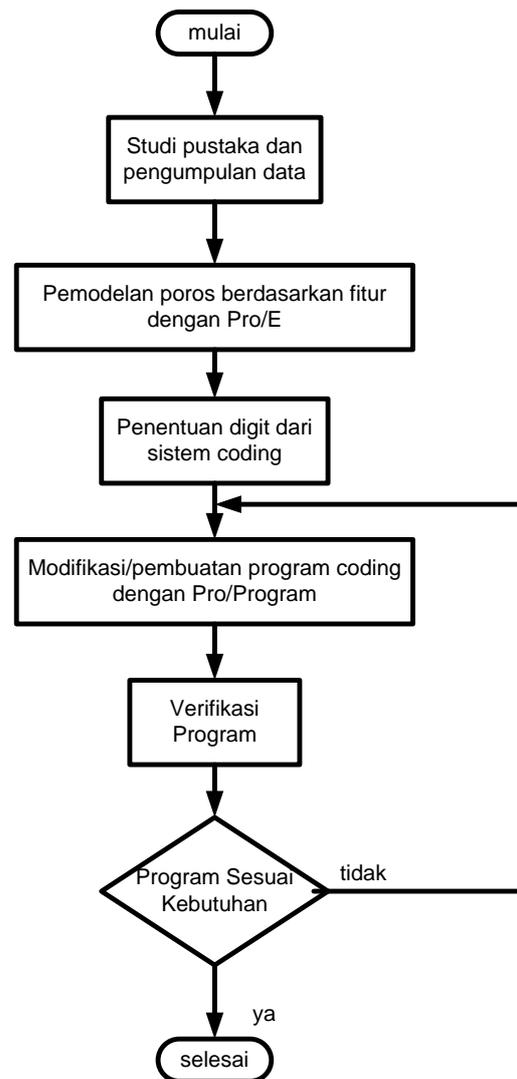
3. METODOLOGI

Jenis komponen yang menjadi obyek dalam penelitian ini adalah poros standar untuk alat/mesin pengolah bahan makanan (*food processing*). Pada tahap awal, dilakukan pemodelan poros dengan menggunakan *software Pro/Engineer 2001* berdasarkan fitur-fitur yang diperlukan. Fitur-fitur tersebut adalah: bentuk poros, dengan/tanpa *groove*, dan jumlah tingkatan poros. Fitur bentuk poros dibuat berdasarkan perbandingan antara panjang poros (L) terhadap diameter poros (D). Dalam fitur ini, dibagi menjadi 3 jenis yaitu: poros kecil ($L/D \leq 10$), poros sedang ($10 < L/D \leq 25$), dan poros besar ($L/D > 25$). Dalam aplikasi sehari-hari, poros kecil banyak digunakan pada alat/mesin dengan skala rumah tangga, seperti mesin pamarut kelapa mini. Poros sedang banyak digunakan pada alat/mesin dengan skala *home industri*, seperti mesin pamarut kelapa yang banyak digunakan di pasar. Sedangkan poros besar sering digunakan pada mesin-mesin skala industri, seperti mesin pemipil jagung dan sejenisnya. Tahap selanjutnya adalah penentuan digit dari sistem pengkodean (*coding*) berdasarkan

penggolongan dari kesamaan fitur dasar dari masing-masing poros.

Sistem *coding* yang sudah jadi digunakan sebagai acuan dalam pembuatan program. Program ini dibuat dengan menggunakan bantuan *software Pro/Program*. Setelah program selesai, dilakukan verifikasi terhadap program tersebut, untuk mencoba apakah program yang dibuat sesuai dengan kebutuhan.

Metodologi yang dilakukan dalam penelitian ini dapat dilihat pada gambar 5 berikut ini.

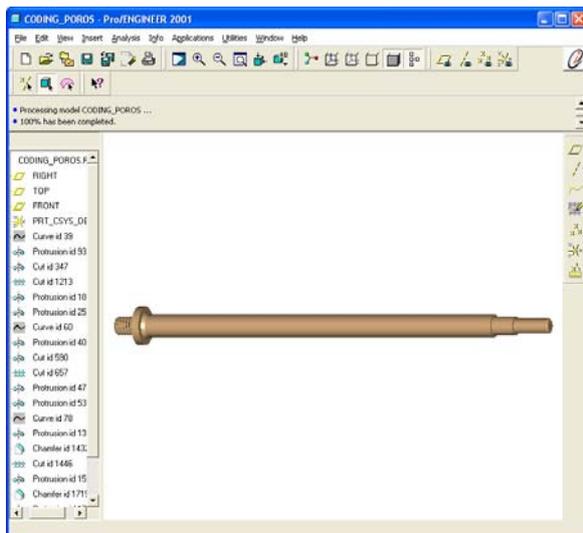


Gambar 5. Metodologi Penelitian

4. HASIL

Sesuai dengan pembagiannya yaitu poros kecil, poros sedang, dan poros besar, maka diperlukan 3 buah poros standar yang dapat mewakili ketiga jenis poros yang akan dijadikan acuan. Poros tersebut antara lain: poros mesin pembuat tepung (poros kecil), poros WP (poros sedang), dan poros mesin *sawmill* (poros besar).

Ketiga jenis poros tersebut dimodelkan dalam satu *file* dengan menggunakan *software Pro/Engineer 2001*. Pemodelan dilakukan berdasarkan fitur-fitur yang nantinya digunakan sebagai acuan dalam sistem *coding*. Masing-masing model poros dibuat berdasarkan referensi yang independen. Hasil pemodelan dari poros dapat dilihat pada gambar 2 berikut ini.



Gambar 6. Pemodelan 3D Poros

Setelah dilakukan pemodelan 3D, langkah berikutnya adalah membuat sistem *coding* yang terdiri dari 3 digit angka,

berdasarkan fitur dari poros. Fitur yang dijadikan digit pertama dari sistem *coding* ini adalah bentuk silinder. Urutan pengkodean dari fitur ini menggunakan angka dengan penjelasan sebagai berikut: nomor 0 (poros kecil), 1 (poros sedang), dan 2 (poros besar). Digit kedua dari sistem pengkodean ini berdasarkan ada atau tidaknya *groove* pada poros tersebut, yaitu tidak ada *groove* (0) dan ada *groove* (1). Khusus untuk digit ini tidak berlaku untuk poros besar. Digit ketiga dari sistem pengkodean ini adalah jumlah tingkatan poros, yaitu poros satu tingkat (1) dan poros dua tingkat (2). Untuk lebih jelasnya mengenai sistem *coding* yang dibuat, dapat dilihat pada Tabel 1 berikut ini.

Tabel 1. Tabel Sistem Coding Poros

Kode	Digit 1 (L/D)	Digit 2 Groove	Digit 3 Juml. Tingkat
0	$L/D \leq 10$	no	n.a
1	$10 < L/D \leq 25$	yes	1
2	$L/D > 25$	n.a.	2

Untuk mendapatkan poros sesuai dengan keinginan, kita harus mengetahui sistem *coding* seperti pada tabel 1. Misalkan untuk poros kecil dengan *groove*, dan jumlah tingkatan poros adalah 2, maka kode yang harus diberikan adalah "012". Sistem *coding* yang sudah jadi ini akan dibuat dalam bentuk program dengan menggunakan *software Pro/Program*. *Software* ini sudah terintegrasi dalam *software Pro/Engineer 2001*, sehingga

lebih mudah dalam pembuatan dan modifikasinya.

Logika dasar yang digunakan dalam pembuatan program ini adalah dengan mengatur fitur-fitur pembentuk poros yang disesuaikan dengan sistem *coding* yang telah disusun. Setelah kita melakukan pemodelan 3D dengan menggunakan *Pro/Engineer*, secara otomatis *software* akan menuliskan program dasar berupa program fitur-fitur pembentuk poros. Yang dilakukan pada penelitian ini adalah memodifikasi program agar dapat diaplikasikan sesuai dengan sistem *coding*. Hasil modifikasi program adalah sebagai berikut:

```

VERSION
REVNUM 1708
LISTING FOR PART CODING_POROS

INPUT
DIGIT1 NUMBER
"INPUTKAN DIGIT PERTAMA (L/D) [0:KECIL 1:SEDANG
2:BESAR]:"
DIGIT2 NUMBER
"INPUTKAN DIGIT KEDUA (GROOVE - n.a for poros besar)
[0:NO 1:YES]:"
DIGIT3 NUMBER
"INPUTKAN DIGIT KETIGA (JUML. TINGKAT) [1:SATU
2:DUA]:"
END INPUT

RELATIONS
END RELATIONS

IF DIGIT1==0

    ADD FEATURE (initial number 6)
    INTERNAL FEATURE ID 93
    PARENTS = 3(#2) 39(#5) 1(#1) 5(#3)

PROTRUSION: Revolve

NO. ELEMENT NAME  INFO
STATUS
1 Attributes  One Side          Defined
2 Section    Sk. plane - Surface FRONT Defined
3 Direction
4 Angle      Angle = 360 degrees  Defined

SECTION NAME = S2D0006

FEATURE IS IN LAYER(S) :
02__PRT_ALL_AXES - OPERATION = SHOWN

END ADD

```

```

ADD FEATURE (initial number 8)
INTERNAL FEATURE ID 1213
PARENTS = 3(#2) 39(#5) 93(#6) 1(#1) 5(#3)

```

CUT: Helical Sweep

```

NO. ELEMENT NAME  INFO
STATUS
1 Attributes  Constant Pitch      Defined
2 Swp Profile  Sk. Plane                 Defined
3 Pitch       Pitch = 2.25                Defined
4 Section
5 MaterialSide Inside section  Defined

```

```

SECTION NAME = S2D0001
SECTION NAME = S2D0002

```

```

FEATURE'S DIMENSIONS:
d95 = 18
d96 = 22
d97 = 2.25PITCH
d100 = 2
d101 = 60
d102 = 60
END ADD

```

```

IF DIGIT2==1
    ADD FEATURE (initial number 9)
    INTERNAL FEATURE ID 347
    PARENTS = 3(#2) 93(#6) 1(#1) 5(#3)

```

CUT: Revolve

```

NO. ELEMENT NAME  INFO
STATUS
1 Attributes  One Side          Defined
2 Section    Sk. plane - Surface FRONT Defined
3 MaterialSide Inside section  Defined
4 Direction
5 Angle      Angle = 360 degrees  Defined

```

```
SECTION NAME = S2D0007
```

```

FEATURE'S DIMENSIONS:
d57 = 2
d58 = 6
d59 = 1
d84 = 1
END ADD
END IF

```

IF DIGIT3==1

```

ADD FEATURE
INTERNAL FEATURE ID 186
PARENTS = 3(#2) 39(#5) 1(#1) 5(#3)

```

PROTRUSION: Revolve

```

NO. ELEMENT NAME  INFO
STATUS
1 Attributes  One Side          Defined
2 Section    Sk. plane - Surface Defined
3 Direction
4 Angle      Angle = 360 degrees  Defined

```

```
SECTION NAME = S2D0005
```

```

END ADD
END IF

```

```

IF DIGIT3==2
    ADD FEATURE (initial number 10)
    INTERNAL FEATURE ID 253
    PARENTS = 3(#2) 39(#5) 1(#1) 5(#3)

```

```

PROTRUSION: Revolve
4 Direction Defined
5 Angle Angle = 360 degrees Defined

NO. ELEMENT NAME INFO
STATUS
1 Attributes One Side Defined
2 Section Sk. plane - Surface FRONT Defined
3 Direction Defined
4 Angle Angle = 360 degrees Defined

SECTION NAME = S2D0001

END ADD
END IF
END IF

IF DIGIT1==1
ADD FEATURE
INTERNAL FEATURE ID 400
PARENTS = 3(#2) 60(#11) 1(#1) 5(#3)

PROTRUSION: Revolve

NO. ELEMENT NAME INFO
STATUS
1 Attributes One Side Defined
2 Section Sk. plane - Surface FRONT Defined
3 Direction Defined
4 Angle Angle = 360 degrees Defined

SECTION NAME = S2D0002

END ADD
END IF

IF DIGIT3==1
ADD FEATURE
INTERNAL FEATURE ID 475
PARENTS = 3(#2) 60(#11) 1(#1) 5(#3)

PROTRUSION: Revolve

NO. ELEMENT NAME INFO
STATUS
1 Attributes One Side Defined
2 Section Sk. plane - Surface FRONT Defined
3 Direction Defined
4 Angle Angle = 360 degrees Defined

SECTION NAME = S2D0002

END ADD
END IF

IF DIGIT3==2
ADD FEATURE
INTERNAL FEATURE ID 536
PARENTS = 3(#2) 60(#11) 1(#1) 5(#3)

PROTRUSION: Revolve

NO. ELEMENT NAME INFO
STATUS
1 Attributes One Side Defined
2 Section Sk. plane - Surface FRONT Defined
3 Direction Defined
4 Angle Angle = 360 degrees Defined

SECTION NAME = S2D0001
END ADD
END IF

IF DIGIT1==2
ADD FEATURE (initial number 18)
INTERNAL FEATURE ID 1349
PARENTS = 3(#2) 78(#17) 1(#1) 5(#3)

PROTRUSION: Revolve

NO. ELEMENT NAME INFO
STATUS
1 Attributes One Side Defined
2 Section Sk. plane - Surface FRONT Defined
3 Direction Defined
4 Angle Angle = 360 degrees Defined

SECTION NAME = S2D0001

FEATURE IS IN LAYER(S) :
02__PRT_ALL_AXES - OPERATION = SHOWN

END ADD

ADD FEATURE (initial number 20)
INTERNAL FEATURE ID 1446
PARENTS = 3(#2) 78(#17) 1(#1) 5(#3)

CUT: Helical Sweep

```

```

NO. ELEMENT NAME INFO
STATUS
1 Attributes Constant Pitch Defined
2 Swp Profile Sk. plane - Surface Defined
3 Pitch Pitch = 4.25 Defined
4 Section Defined
5 MaterialSide Inside section Defined

```

```

SECTION NAME = S2D0002
SECTION NAME = S2D0003

```

```

FEATURE'S DIMENSIONS:
d118 = 41.19 (weak)
d119 = 4.25PITCH
d122 = 4
d123 = 60
d124 = 60
END ADD

```

```

IF DIGIT3==1
ADD FEATURE (initial number 21)
INTERNAL FEATURE ID 1569
PARENTS = 3(#2) 78(#17) 1(#1) 5(#3)

```

```
PROTRUSION: Revolve
```

```

NO. ELEMENT NAME INFO
STATUS
1 Attributes One Side Defined
2 Section Sk. plane - Surface FRONT Defined
3 Direction Defined
4 Angle Angle = 360 degrees Defined

```

```
SECTION NAME = S2D0006
```

```

FEATURE IS IN LAYER(S) :
02__PRT_ALL_AXES - OPERATION = SHOWN

```

```

END ADD
ENDIF

```

```

IF DIGIT3==2
ADD FEATURE (initial number 23)
INTERNAL FEATURE ID 1733
PARENTS = 3(#2) 78(#17) 1(#1) 5(#3)

```

```
PROTRUSION: Revolve
```

```

NO. ELEMENT NAME INFO
STATUS
1 Attributes One Side Defined
2 Section Sk. plane - Surface Defined
3 Direction Defined
4 Angle Angle = 360 degrees Defined

```

```
SECTION NAME = S2D0006
```

```
END ADD
```

```

ADD FEATURE (initial number 24)
INTERNAL FEATURE ID 1811
PARENTS = 1733(#23)

```

```
CHAMFER: Edge
```

```

NO. ELEMENT NAME INFO
STATUS
1 Scheme 45 x d Defined
2 Edge Refs 1 curve or edge(s) Defined

```

```

FEATURE'S DIMENSIONS:
d142 = 2 X 45 Deg
END ADD
ENDIF
ENDIF

```

```

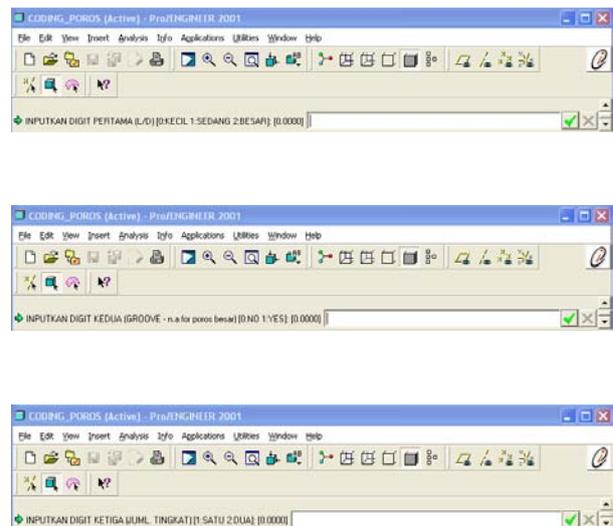
MASSPROP
END MASSPROP

```

Program diatas disimpan dalam bentuk *text file*, yang selanjutnya diintegrasikan ulang dalam *file* poros.

5. DISKUSI

Setelah dilakukan integrasi ulang, program tersebut diuji coba untuk mengetahui apakah program bisa beroperasi dengan benar. Dalam program ini, kita memasukkan 3 digit kode secara bergantian dan berurutan. Perintah input kode dibuat sederhana dan se jelas mungkin sehingga kita tidak perlu menghafalkan definisi masing-masing kode. Perintah input kode yang sudah diintegrasikan ulang dapat dilihat pada gambar 7.



Gambar 7. Perintah Input Digit 1,2,dan 3.

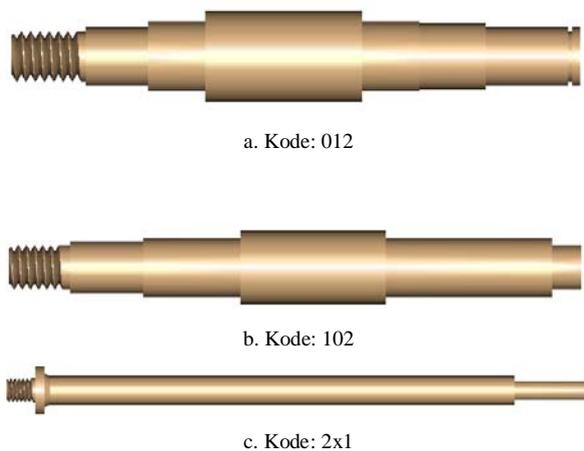
Dengan menginputkan kode yang benar, maka akan diperoleh model poros sesuai

dengan kebutuhan. Kode yang digunakan terdiri dari 3 digit kode, dan jenis kode dalam masing-masing digit ada 3 buah (tabel 1), sehingga dalam sistem *coding* ini bisa dibuat 4 buah varian poros untuk masing-masing jenis. Akan tetapi, karena digit kedua (*groove*) tidak berlaku untuk poros besar, maka untuk poros besar hanya ada 2 varian, yaitu poros besar satu tingkat dan poros besar dua tingkat. Total varian yang dapat dibentuk dalam program ini sebanyak 10 buah varian (tabel 2).

Tabel 2. Varian Kode dalam Program

Poros kecil	Poros sedang	Poros besar
001	101	
011	111	2x1
002	102	2x2
012	112	

Gambar 8 berikut ini merupakan salah satu contoh varian dari masing-masing jenis poros.



Gambar 8. Contoh Varian dari Tiap Poros

Dengan program ini, apabila desainer ingin mendesain poros dengan fitur-fitur

yang terdapat pada sistem pengkodean tadi, desainer hanya perlu melakukan input kode ke dalam program sesuai dengan fitur poros yang diinginkan, dan secara otomatis, program akan menampilkan model 3 dimensi dan gambar kerja dari poros tersebut. Masing-masing varian yang dihasilkan dari input kode dapat dijadikan sebagai suatu *instance* atau komponen baru, sehingga kita dapat melakukan modifikasi lebih lanjut atau menambahkan fitur baru pada komponen tersebut, misalkan menambahkan *key slot*, *chamfer*, dan sebagainya.

Semua *instance* beserta modifikasi yang kita lakukan, dapat langsung dituangkan dalam *drawing model* (gambar kerja) pada *Pro/Engineer* tanpa harus menggambar dan memodifikasi ulang.

6. KESIMPULAN

Dengan menggunakan program ini, desainer lebih mudah dalam membuat gambar komponen yang mempunyai kemiripan fitur-fitur dasar dengan gambar komponen yang sudah ada sebelumnya, tanpa harus menggambar dari awal. Proses mengingat dan mencari *file* gambar sebelumnya berdasarkan nama *file* tidak perlu dilakukan lagi. Dalam satu *file* komponen poros yang menggunakan program sistem *coding*, dapat memuat 12 varian komponen. Hanya dengan menginputkan 3 digit kode, maka kita dapat

memperoleh model 3D dan gambar kerja dari poros yang dikehendaki. Hal ini dapat menghemat waktu dan tenaga serta sangat mendukung proses perancangan dan pengembangan produk.

Jumlah varian dari sebuah komponen dapat diperbanyak, tergantung dari jenis fitur yang diperlukan. Selain itu, jumlah digit kode juga dapat diperbanyak, terutama untuk komponen-komponen yang memiliki tingkat kompleksitas cukup tinggi.

7. DAFTAR PUSTAKA

- [1] Henault Mark., Sean Sevrence, Mike Walraven, *Automating Design in Pro/Engineer with Pro/Program*, 2530 Camino Entrada, Santa Fe: OnWord Press, 1997.
- [2] Singh Nanua., *System Approach to Computer-Integrated Design and Manufacturing*, Canada : John Wisley & Sons, Inc., 1996.
- [3] Chang Tien-Chien., Richard A. Wysk., Hsu-Pin Wang., *Computer-Aided Manufacturing*, Englewood Cliffs, New Jersey 07632 : Prentice Hall, 1991
- [4] <http://cobweb.ecn.purdue.edu/~tchang/doc/download/camnotes/chap12gt.pdf>