



UNIVERSITAS KATOLIK PARAHYANGAN

LEMBAGA PENELITIAN

(RESEARCH INSTITUTE AT PARAHYANGAN CATHOLIC UNIVERSITY)

Jalan Ciumbuleuit 94 – Phone: (022) 2030918, 2030919, 2030920 Pes. 142, 148

Bandung 40141 – Indonesia

LAPORAN HASIL PENELITIAN

Judul Penelitian:

PENGEMBANGAN ALGORITMA SETTING DAN SUB-ROUTINE PEMROGRAMAN MESIN CNC (MILLING)

Disusun oleh:

Bagus Arthaya

**Jurusan Teknik Industri
Fakultas Teknologi Industri
Universitas Katolik Parahnyangan
2012**

Abstrak

Pemrograman mesin perkakas CNC merupakan keahlian khusus yang harus dimiliki oleh pengguna mesin CNC dimana hal ini bukan merupakan hal yang mudah untuk dikuasai. Fasilitas pemrograman yang tersedia saat ini umumnya merupakan satu paket yang terintegrasi dengan jenis mesin perkakas yang digunakan dan mahal harganya. Untuk membantu industri kecil menyiapkan tenaga pemrogram CNC, maka perlu dirancang suatu media/fasilitas yang dapat mendukung kegiatan pemrograman CNC tersebut. Beberapa penelitian sebelumnya telah membahas kajian awal tentang uji dan ekstraksi suatu program NC yang dibuat oleh seorang programmer. Disamping itu formulasi lintasan yang dihasilkan program NC tersebut juga telah dijabarkan, walaupun belum seluruh aspek lintasan secara rinci dibahas.

Pelatihan ini merupakan lanjutan dari penelitian tersebut, dimana fungsi-fungsi khusus seperti penentuan setting benda kerja dan pahat terhadap titik referensi mesin belum diperhitungkan dalam rangka mensimulasi gerakan mesin saat beroperasi. Disamping itu, berbagai fungsi khusus pada mesin Milling, seperti fungsi koordinasi untuk perubahan bidang interpolasi, fungsi copying yang berperan penting dalam pembuatan fitur pemesinan sejenis, formulasi lintasan yang berpotongan antara berbagai macam gerak seperti lingkaran dengan lingkaran, garis lurus sembarang dengan lingkaran sembarang dan lain-lain. Penelitian ini bertujuan untuk menghasilkan suatu langkah-langkah atau algoritma dasar yang sangat diperlukan pada saat perangkat lunak simulasi gerak akan dibuat.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Mesin perkakas sangat diperlukan dalam industri manufaktur dewasa ini, lebih-lebih mesin perkakas moderen yang memerlukan penanganan yang khusus. Mesin CNC merupakan salah satu mesin perkakas yang dimiliki oleh Lab. Proses Produksi Jur. TI Unpar, akan tetapi fasilitas pemrograman secara *off-line* pada suatu komputer pribadi belum tersedia. Untuk dapat memprogram mesin ini, diperlukan seorang *programmer* yang memiliki kemampuan khusus yakni pengetahuan di bidang mesin perkakas, teknik produksi dan juga pengetahuan dasar komputer. Tidak semua pihak dapat memiliki tenaga ahli seperti ini, sementara industri kecil sudah mulai melakukan proses produksinya menggunakan mesin perkakas CNC. Maka dari itu dipandang perlu untuk membuat suatu sistem/alat yang dapat membantu industri kecil agar dapat lebih mampu memanfaatkan mesin-mesin perkakas modern tersebut.

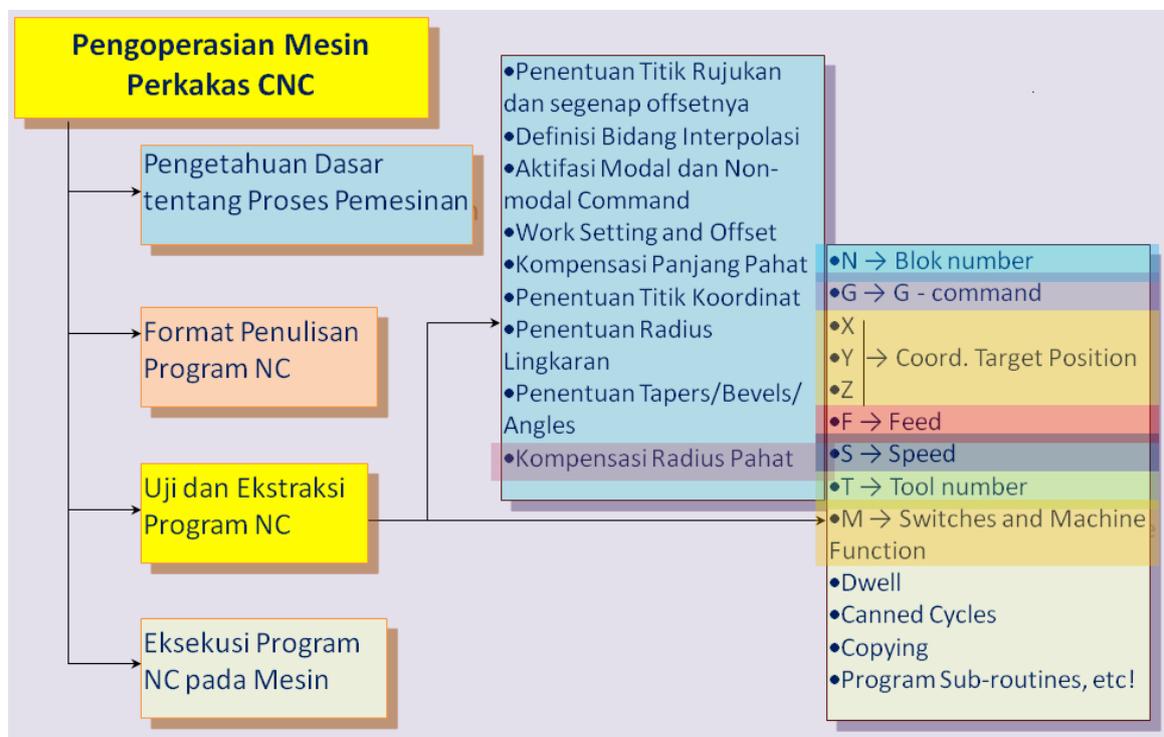
Memprogram mesin perkakas CNC memerlukan keahlian khusus yang bukan merupakan hal yang mudah untuk dikuasai. Fasilitas pemrograman yang tersedia saat ini umumnya merupakan satu paket yang terintegrasi dengan jenis mesin perkakas yang digunakan dan mahal harganya. Untuk membantu industri kecil menyiapkan tenaga pemrogram CNC, maka perlu dirancang suatu media/fasilitas yang dapat mendukung kegiatan pemrograman CNC tersebut. Penelitian sebelumnya (Arthaya, 2011), telah membuka kajian awal tentang uji dan ekstraksi suatu program NC yang dibuat oleh seorang *programmer*. Disamping itu formulasi lintasan yang dihasilkan program NC tersebut juga telah dijabarkan, walaupun belum seluruh aspek lintasan dibahas.

Fungsi-fungsi khusus untuk pemrograman mesin CNC sangat banyak macamnya. Untuk dapat membuat media media/fasilitas yang dapat mendukung kegiatan pemrograman CNC secara *off-line* diperlukan tenaga dan waktu yang sangat banyak. Dalam penelitian ini, beberapa fungsi khusus dalam pemrograman mesin CNC secara bertahap akan dikembangkan, tertama yang terkait dengan langkah-langkah dasar proses pemesinan pada suatu mesin perkakas, khususnya mesin Milling. Dengan adanya fungsi yang lebih lengkap, maka media/fasilitas pemrograman ini akan memberi gambaran yang lebih rinci, lebih lengkap dan lebih mudah divisualisasikan pada layar komputer.

Pengembangan lebih lanjut terhadap penelitian awal tersebut di atas, bertujuan agar penerapan fungsi-fungsi khusus yang diperlukan dalam operasi pemesinan Milling dapat dilakukan dengan lebih sempurna. Pembatasan masalah yang masih berlaku pada penelitian sebelumnya akan dibuka dan akan menuntut penanganan yang lebih mendetail. Kelengkapan fungsi-fungsi baru ini akan menambah kelengkapan fungsi operasi pada mesin Milling secara umum. Pembahasan lanjutan dari penelitian terdahulu tentang perancangan algoritma pengujian dan ekstraksi G-Code untuk persiapan perintah gerak mesin CNC, dimana secara umum algoritma ini dapat diimplementasikan

pada mesin milling 2-axis, 3-axis dan juga pada mesin bubut. Algoritma ekstraksi ini masih memiliki beberapa keterbatasan yakni beberapa fungsi dasar belum semuanya diterapkan. Penambahan beberapa fungsi yang akan dikaji dalam penelitian ini selanjutnya akan digunakan untuk meningkatkan kelengkapan fungsi-fungsi simulasi dalam perangkat lunak pengolah G-Code yang mendapat input dari suatu program pengujian dan simulasi program NC untuk mesin milling CNC 3-sumbu ataupun mesin bubut CNC.

Secara garis besar dapat dijelaskan bahwa untuk mengoperasikan mesin perkakas CNC maka diperlukan penguasaan beberapa fungsi dasar seperti yang disajikan pada Gambar 1.1. Pada penelitian sebelumnya, fokus pekerjaan ditekankan pada Blok Uji dan Ekstraksi Program NC, dimana didalamnya terdapat banyak fungsi-fungsi pendukungnya. Dalam penelitian inipun belum semua fungsi pendukung dapat dibahas karena luasnya topik bahasan tentang pengoperasian mesin CNC.

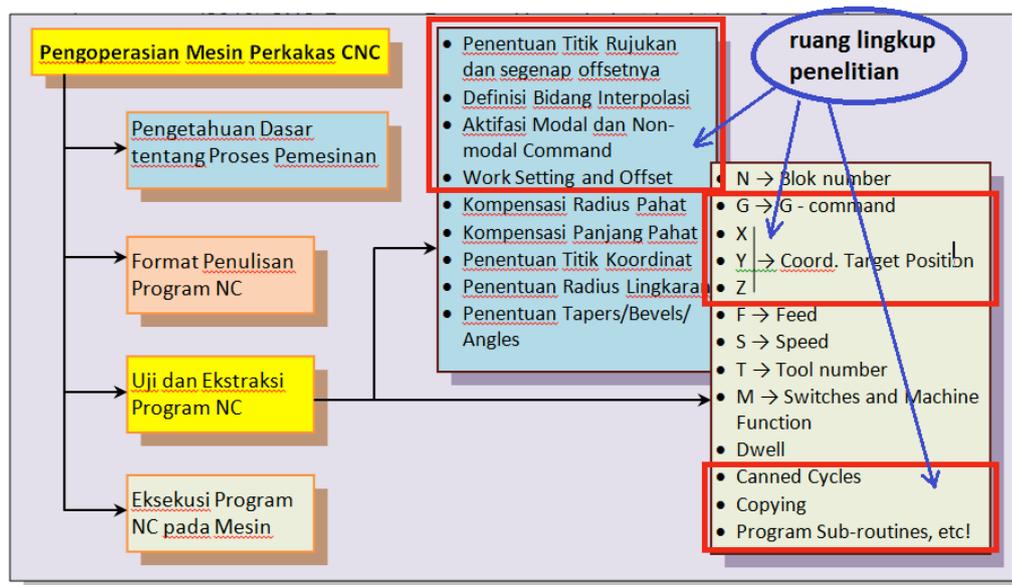


Gambar 1.1 Ruang lingkup bahasan dalam pemrograman mesin CNC

Dalam Gambar 1.1 terlihat beberapa fungsi dasar yang sudah dikembangkan pada penelitian yang lalu, antara lain adalah:

- a. Kompensasi radius pahat
- b. Block number
- c. G-code (dasar, belum seluruhnya)
- d. Koordinat dari posisi target
- e. Speed
- f. Feed
- g. Tool number
- h. M-function (belum seluruhnya)

Dalam penelitian kali ini, beberapa fungsi baru akan ditambahkan dan dibahas bagaimana langkah yang diperlukan untuk mengolah informasi yang ada bila jenis fungsi baru tersebut muncul dalam suatu blok informasi dalam G-code. Beberapa fungsi tersebut menjadi bahasan utama dalam ruang lingkup penelitian ini seperti yang disajikan dalam Gambar 1.2.



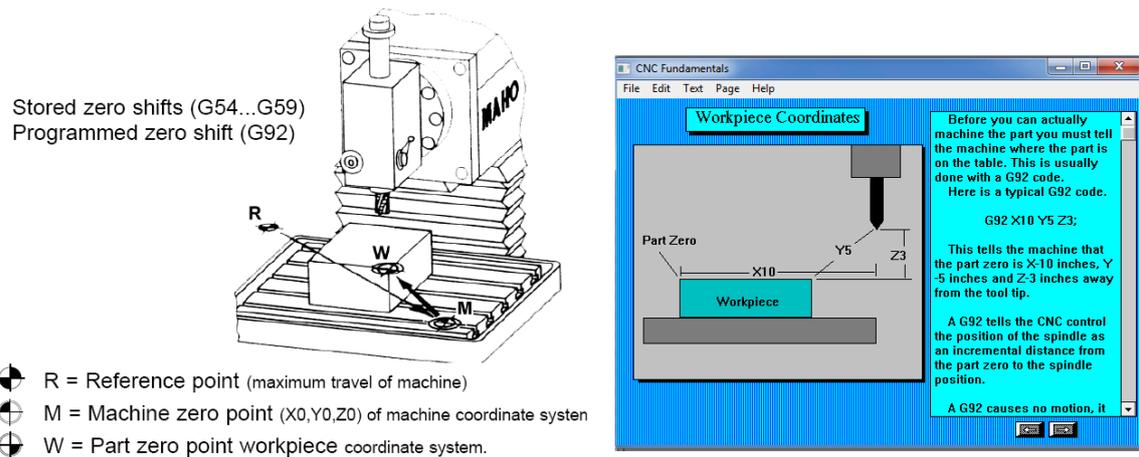
Gambar 1.2 Ruang lingkup penelitian yang akan dibahas dalam penelitian

Dengan memperhatikan tingkat urgensitas fungsi yang harus disiapkan, maka topik yang akan dibahas memiliki urutan sebagai berikut:

1) Penentuan Titik Referensi Mesin/Pahat Relatif Terhadap Benda Kerja;

Suatu titik rujukan harus diketahui terlebih dahulu untuk memastikan bahwa posisi benda kerja di atas meja mesin relatif terhadap meja mesin atau titik referensi mesin diketahui dengan pasti oleh pengendali mesin CNC. Bila nilai ini telah dipastikan, maka operator tidak perlu khawatir lagi bagaimana program akan dilaksanakan pada benda kerja. Disamping posisi, orientasi benda kerja juga perlu disesuaikan dengan salib sumbu meja mesin. Beberapa G-Code sudah disiapkan untuk menangani masalah ini antara lain: G54.... G59 dan juga G92.

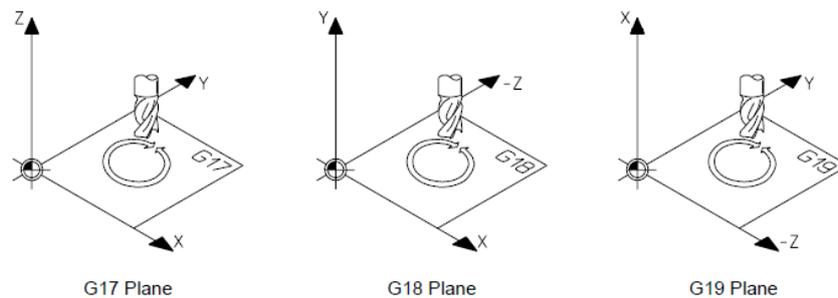
Seperti telah disebutkan di atas posisi benda kerja di atas meja mesin dapat didefinisikan menggunakan G92. Apabila posisi titik origin benda kerja telah diketahui terhadap posisi mata pahat, maka suatu blok informasi misalnya: G92 X10 Y5 Z3; seperti pada Gambar 1.3 maka kendali mesin akan menetapkan posisi mata pahat adalah pasti relatif terhadap titik koordinat yang dimaksud dalam program NC.



Gambar 1.3 Penentuan posisi benda kerja di atas meja mesin dengan titik-titik rujukan yang ada (Krar,1999; Stenerson,1997)

2) Pendefinisian Bidang Interpolasi;

Pada kasus-kasus pengerjaan bidang yang kompleks, seringkali terjadi perubahan bidang interpolasi gerakan pahat sehingga fungsi khusus diperlukan untuk dapat melakukan gerakan interpolasi di bidang yang baru. Kondisi ini sering terjadi pada mesin milling horizontal karena sumbu spindel dapat mengarah ke bidang x-z maupun y-z. Kode-kode yang perlu diaktifkan adalah G17/G18/G19 untuk mengakomodasi 3 bidang interpolasi yang berbeda yakni pada bidang x-y, bidang x-z dan bidang y-z seperti yang disajikan pada Gambar 1.4.

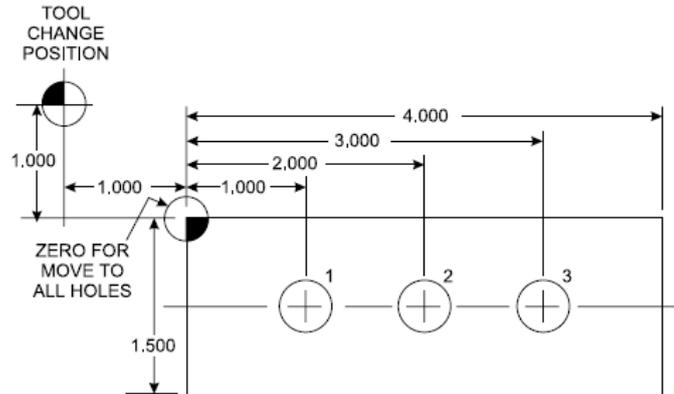


Gambar 1.4 Tiga macam bidang interpolasi gerakan pahat yang berbeda [MTS, 1995]

3) Work Setting and Offset;

Semua mesin CNC memerlukan semacam langkah penentuan work setting, tool setting, dan offsets (kompensasi) untuk mendapatkan hubungan/relasi yang tepat antara pahat potong dan benda kerja. Adanya kompensasi memungkinkan operator untuk membuat berbagai pada kondisi perkakas dan setup yang tidak diharapkan.

Dalam sistem pemosisian absolut, koordinat benda kerja biasanya di-set (ditentukan) pada sisi atau pojok tertentu pada benda and semua langkah pemrograman biasanya diambil/diacu dari titik rujukan ini. Dalam Gambar 1.5, posisi nol (rujukan) benda kerja digunakan untuk penentuan posisi dari semua posisi lubang 1, 2, dan 3.



Gambar 1.5 Pada pemrograman absolute, semua dimensi harus mengacu pada titik nol di ujung kiri atas benda kerja [Krar, 1999].

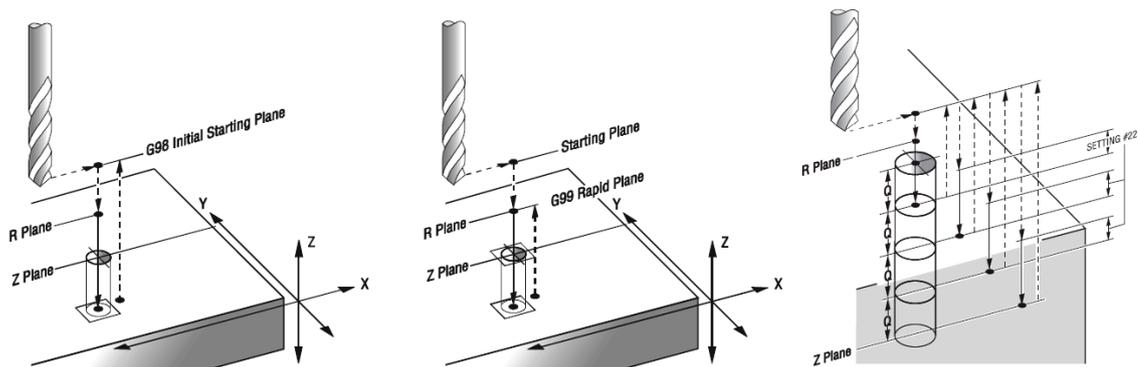
4) Canned Cycles untuk Penentuan Fungsi Dwell;

Fasilitas canned cycle digunakan untuk menyederhanakan pemrograman suatu benda kerja. Canned cycles didefinisikan sebagai operasi repetitif yang paling umum dilakukan pada arah sumbu-Z seperti proses drilling, tapping, dan boring. Dari macamnya, terdapat 13 jenis canned cycles yang bisa dipilih dan sekali dipilih cycle ini akan tetap aktif sampai dia dibatalkan dengan kode G80. Ketika aktif, cycle ini bekerja untuk setiap nilai sumbu-X dan/atau sumbu-Y yang diprogram. Gerakan pada sumbu X/Y tersebut dilakukan dengan mode perintah RAPID dan untuk gerakan Z dilaksanakan dengan perintah FEEDING, dan canned cycle ini dilakukan lagi untuk setiap koordinat X-Y yang mengikutinya. Ada lima macam operasi dalam suatu canned cycle, yaitu [Haas]:

- a. Positioning pada sumbu- X dan sumbu-Y.
- b. Gerakan cepat ke bidang R (referensi).
- c. Lakukan drilling, tapping, dan boring bagi canned cycle yang sedang dijalankan.
- b. Operasi pada bagian bawah dalam lubang.
- c. Kembali ke bidang rujukan R (G99) atau titik start awal (G98).

Penempatan suatu canned cycle pada sumbu- X dan/atau -Y dapat dilakukan dalam mode absolut (G90) ataupun incremental (G91). Gerakan incremental (G91) dalam suatu canned cycle sering berguna saat digabungkan dengan loop (Lnn) count yang akan mengulangi operasi canned cycle tersebut beberapa kali untuk setiap gerakan incremental pada sumbu-X atau -Y move bagi canned cycle yang bersangkutan.

Pada proses pembuatan lubang, terutama yang kedalamannya cukup besar, diperlukan fungsi pendukung seperti Dwell. Fungsi ini memaksa pahat bertahan pada posisi tertentu untuk beberapa saat hingga kedalaman yang diinginkan tercapai. Urutan menekan kedalam bagi pahat saat membuat lubang, dapat ditentukan untuk keseragam kerja dan hal ini dilakukan dalam Canned cycle. Contoh pembuatan lubang secara bertahap dapat dilihat pada Gambar 1.6.

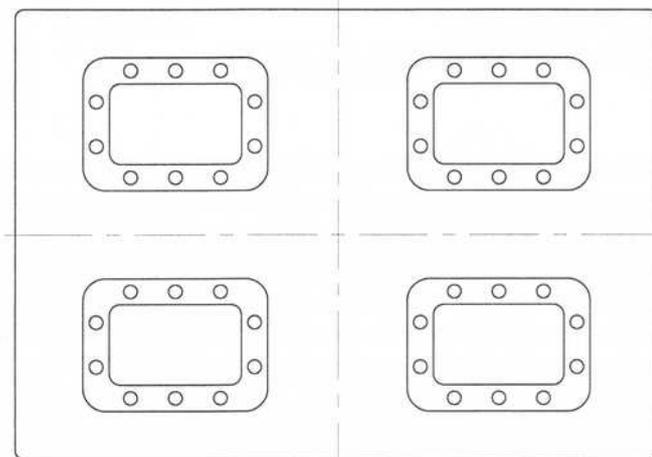


Gambar 1.6 Posisi Dwell untuk membuat lubang dengan kedalaman bertahap [Haas Automation, 2006].

5) **Copying;**

Sering kali dalam proses pemesinan menggunakan mesin milling, diperlukan suatu fitur pemesinan yang berulang-ulang kali dikerjakan. Antara satu fitur dengan fitur lainnya terdapat kesamaan ukuran tetapi berbeda pada titik rujukan atau posisinya. Untuk melakukan operasi copying ini dapat dilakukan dengan istilah repetitive machining motion.

Pada Gambar 1.7 terlihat suatu contoh fitur pemesinan yang memiliki empat pola identik tetapi terletak pada posisi yang berbeda. Cara yang paling efektif untuk mendapatkan pola ini adalah dengan membuat repetisi proses pemesinan untuk pola dasar dan dilakukan sebanyak empat kali pada posisi lain yang diinginkan.

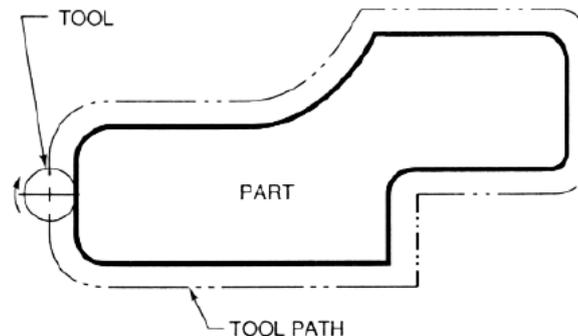


Gambar 1.7 Bentuk pola/fitur pemesinan yang berulang dapat diselesaikan dengan repetitive machining motion [Smid, 2003].

6) **Formulasi detail untuk G41 dan G42;**

Gerakan interpolasi pada mesin perkakas sangatlah banyak ragamnya. Diantara sekian banyak gerakan, menentukan titik potong dan titik tujuan pahat agar perpotongan antar dua garis potong dapat terealisasi merupakan masalah yang cukup pelik. Perpotongan dua garis lurus akan memberikan output yang berbeda dengan kasus perpotongan antara garis lurus dengan juring lingkaran di sembarang titik potong seperti ditunjukkan pada Gambar 1.8. Fungsi-fungsi seperti ini akan

menjadi sangat penting dalam rangka mensimulasikan gerakan pada media/fasilitas pemrograman NC. Untuk tujuan itu harus dibangun fungsi-fungsi yang dapat memodifikasi titik tujuan pahat potong pada perpotongan dua garis lurus dengan sudut yang lancip ataupun tumpul, maupun memodifikasi titik tujuan pahat potong pada kasus perpotongan garis lurus dengan lingkaran yang memiliki sudut yang lancip dan berpotongan pada titik potong di sembarang tempat.



Gambar 1.8 Lintasan pahat pada berbagai jenis perpotongan garis [Krar, 1999].

Penelitian ini bertujuan untuk membahas konsep dasar bagaimana suatu program NC untuk gerak-gerak dasar tersebut diatas dapat direalisasi menjadi perintah gerak bagi mesin yang bersangkutan. Suatu algoritma dikembangkan untuk menganalisis pola gerakan fisik mesin yang harus dihasilkan. Pola gerakan ini sangat terkait dengan jenis sehingga unit penggerak dapat dengan tepat mengirimkan perintah gerak ke masing-masing pengendali sumbu gerak mesin.

Semakin lengkapnya fungsi-fungsi khusus yang digunakan/diperlukan dalam pemrograman mesin perkakas CNC akan membuat media/fasilitas pemrograman NC semakin mendekati proses nyata bagaimana mesin berkerja. Kelengkapan inipun akan membantu programmer untuk lebih cepat dapat mengidentifikasi status dari suatu program NC yang dihasilkan karena keabsahan program dan kebenaran gerakan pahat yang dihasilkan lebih cepat terdeteksi.

1.2 Rumusan Masalah

Dari penjelasan singkat pada pendahuluan di atas, maka masalah yang akan diteliti dalam penelitian ini dapat dijabarkan dalam beberapa pertanyaan berikut:

1. Bagaimanakah cara penentuan titik referensi pahat terhadap benda kerja?
2. Bagaimanakah langkah penentuan setting titik awal benda kerja?
3. Bagaimanakah langkah-langkah pengembangan fungsi-fungsi khusus seperti canned cycle untuk drilling, copying dan lain-lain?
4. Bagaimanakah cara menentukan modifikasi titik tujuan pahat pada saat penggunaan fungsi kompensasi radius pahat

1.3 Kerangka Pemikiran

1. Suatu algoritma yang dapat mendukung pengembangan perangkat lunak untuk pengujian dan pengecitraan G-Code sangat diperlukan untuk dapat memilah-milah perintah gerak bagi tiap-tiap sumbu gerak pada mesin perkakas.
2. Proses pemilahan G-Code secara logika dicirikan berdasarkan kode persiapan, kode dimensional dan kode pendukung dalam blok perintah.
3. G-code yang memiliki aturan penulisan yang terbalik atau tidak lengkap berdasarkan aturan harus bisa segera mendapat penanganan agar pola gerakan yang diinginkan dapat dilaksanakan dengan aman.
4. Hasil uji dan ekstraksi G-Code ini sebaiknya bersifat umum/generik sehingga bila dikembangkan untuk berbagai jenis mesin perkakas dengan mudah dapat dimodifikasi.

1.4 Batasan dan Asumsi Penelitian

Berikut adalah hal-hal yang menjadi pembatasan dan asumsi penelitian:

1. Perintah gerak G-Code yang diuji dan diekstrak diasumsikan dapat dibuat oleh sembarang pihak yang belum teruji kebenarannya.
2. Perintah gerak yang akan diekstrak hanya untuk mesin milling 2-sumbu, atau kasus khusus untuk mesin bubut yang bergerak dibidang 2-dimensi.
3. Perintah gerak yang dikaji adalah untuk gerak cepat, interpolasi linear, dan gerak interpolasi melingkar.
4. Tipe perintah yang diolah harus jenis *modal command*.
5. Faktor yang terkait dengan titik referensi dan hal-hal terkait masih diabaikan.

1.5 Tujuan Penelitian

Penelitian ini bertujuan untuk melanjutkan fungsi-fungsi yang mampu mengeksplor dan membangun suatu algoritma pengujian, pengecitraan G-Code dan membangun formula global untuk gerakan interpolasi lintasan pahat pada bidang 2-dimensi. Algoritma ini kemudian ditujukan bagi pengembangan suatu perangkat lunak yang dapat menjembatani program NC pada komputer dengan unit pengendali meja mesin yang bergerak dalam bidang datar 2-dimensi. Algoritma ini nantinya secara bertahap akan dikembangkan agar dapat mendukung pembuatan perangkat lunak yang bisa dikaitkan dengan paket penelitian tentang pengendalian mesin milling CNC 3-sumbu secara keseluruhan dan bahkan untuk 4 atau 5 sumbu ataupun mesin-mesin perkakas lain yang dikendalikan menggunakan CNC. Algoritma ini harus mampu menjejak secara bertahap, setiap baris perintah G-Code yang dibuat oleh pemrogram NC, walaupun masih terdapat masalah penempatan kode yang terbalik penempatannya, sejauh aturan ketersediaan kode dipenuhi.

Algoritma yang dikembangkan akan dibangun dari susunan sederet fungsi dasar yang secara mandiri dapat diuji kebenarannya. Sementara keterkaitan antar fungsi juga harus dapat diuji performanya. Aliran informasi dari fungsi dasar awal hingga fungsi akhir diuji dengan contoh kasus pemotongan. Luaran yang didapat dari pengembangan algoritma

ini adalah (1) keabsahan semua kode yang dipakai/ditulis dalam suatu program NC, (2) sekumpulan data yang telah terpilah-pilah berdasarkan field/record informasi yang diekstrak, (3) suatu kelompok nilai yang disimpan dalam field tersebut sudah terkait dengan jenis-jenis gerakan dan parameter yang terkait dengannya. Kelompok data ini dapat digunakan untuk memerintah masing-masing motor penggerak mesin agar berputar dengan arah tertentu serta seberapa jauh lintasan yang harus dihasilkan.

1.6 Metode Penelitian

1. Studi Pendahuluan

Langkah ini bertujuan untuk mendapatkan pendalaman dan pengetahuan dasar tentang bagaimana suatu mesin CNC dikendalikan/dioperasikan dan bagaimana cara memprogramnya. Selanjutnya dapat dipahami karakteristik data yang diperlukan dalam pengendalian tersebut dan bagaimana alur informasi mengalir dari program NC menjadi perintah gerak bagi mesin CNC. Perumusan masalah diperlukan untuk menjabarkan masalah yang akan diteliti dalam bentuk yang lebih mudah dimengerti yakni dalam beberapa kalimat tanya serta untuk lebih memfokuskan arah penelitian ini. Disamping itu perlu dinyatakan beberapa batasan dalam penelitian ini karena pada dasarnya kajian yang terkait dengan pemrograman mesin CNC merupakan topik bahasan yang sangat luas. Dengan pembatasan yang diberlakukan, maka fokus penelitian bisa menjadi lebih tajam.

2. Studi Pustaka

Teori dasar yang mendukung penelitian ini sangat diperlukan untuk dirujuk agar arah penelitian ini tidak menyimpang dari arah perkembangan mesin perkakas pada umumnya. Dasar-dasar pengetahuan tentang proses pemesinan dengan mesin perkakas dijelaskan secara umum saja. Lebih dari itu, konsep dasar pemrograman mesin perkakas CNC menjadi acuan utama karena teknik pemrograman ini sebenarnya tidak sepenuhnya sama antara mesin perkakas satu dengan lainnya. Tergantung pada pembuat sistem kendali mesin, maka ada beberapa kode yang tidak sama walaupun fungsi operasi mesinnya sama dan identik. Oleh karena itu, disepakati bahwa terdapat kode-kode pemrograman yang secara umum dapat diterima oleh semua unit kendali mesin perkakas, sementara yang bersifat khusus ditulis mengikuti kesepakatan tertentu.

3. Pengembangan Algoritma Fungsi Khusus pada Program NC.

Langkah ini bertujuan untuk membangun fungsi-fungsi khusus yang diperlukan pada tahap lanjut bagi pemrograman NC. Fungsi-fungsi tersebut antara lain untuk menentukan titik rujukan pahat dan titik awal benda kerja, fungsi canned cycle untuk pengerjaan berulan dan perbaikan terhadap penggunaan kode G41 dan G42. Disamping itu dilakukan pula perhitungan yang diperlukan untuk mengkorelasikan informasi tentang satuan dimensi dalam program NC dengan satuan unit gerak dalam motor penggerak. Lebih dari itu perhatian tentang kecepatan gerak juga menjadi pembahasan yang mendalam.

4. Analisa Rancangan Algoritma.

Tahap ini menyajikan dan membahas seperti apa rancangan algoritma yang akan dibuat sebagai langkah awal untuk membangun perangkat lunak komputer pengendali yang diperlukan untuk mengendalikan gerak mesin. Dalam tahap ini

dilakukan juga analisis terhadap hasil rancangan algoritma, yakni konsistensi pengolahan blok informasi dalam program NC, kebebasan penulisan segmen asalkan aturan minimal masih dipenuhi.

5. Kesimpulan dan Saran.

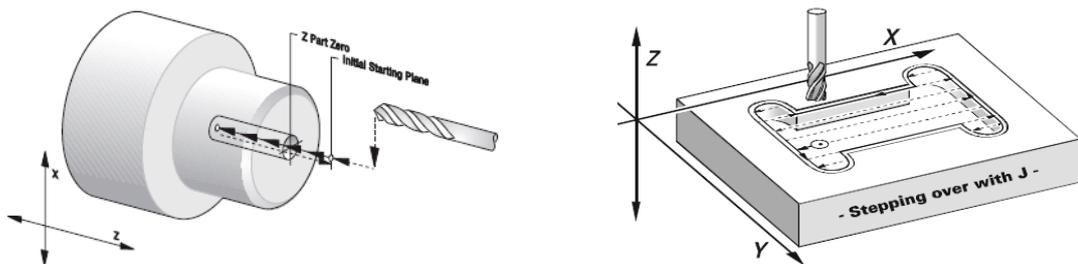
Dalam tahap ini dibuat kesimpulan yang bisa ditarik dari penelitian ini. Disamping itu, saran-saran yang terkait untuk perbaikan pada penelitian lebih lanjut juga disajikan.

BAB II

STUDI PUSTAKA

2.1 Mesin Perkakas CNC

Saat ini banyak mesin perkakas yang dikendalikan dengan sistem kendali berbasis komputer. Dengan bantuan komputer seseorang dapat membuat produk yang bentuknya cukup kompleks dengan kualitas yang sangat baik. Produk awal yang dikerjakan pada mesin perkakas biasanya produk hasil proses pengecoran ataupun hasil proses pemesinan lainnya. Berbagai produk yang dapat dihasilkan oleh mesin perkakas memiliki geometri yang cukup berbeda. Ada benda kerja yang disebut dengan istilah produk rotasional dan juga ada produk non-rotasional seperti ditunjukkan pada Gambar 2.1. Bentuk geometri ini akan sangat terkait dengan jenis mesin perkakas yang digunakan. Dari bentuk benda kerja yang ada, maka jenis mesin perkakas yang harus digunakan juga akan berbeda, misalnya benda-benda yang memiliki simetri putar akan memerlukan mesin bubut untuk memprosesnya, sementara benda kerja rektanguler akan banyak dikerjakan di dalam mesin milling dan sebagainya.



Gambar 2.1 Contoh dua produk dengan geometri yang berbeda, kiri: rotasional dan kanan: non-rotasional [Haas Automation, 2006].

Mesin milling memiliki meja mesin yang dapat bergerak translasional ke arah sumbu-X, -Y dan -Z. Sementara itu pahat pemotong benda kerja yang berputar pada porosnya dipegang oleh sebuah spindel yang stasioner pada suatu posisi tertentu di atas benda kerja. Benda kerja diletakkan di atas meja mesin dan dicekam dengan sempurna, sehingga tidak ada kebebasan gerak lagi. Posisi benda kerja terhadap pahat harus diketahui dengan baik. Karena adanya gerakan translasi pahat terhadap benda kerja yang diam, maka hasil pemotongan yang terbentuk adalah benda kerja non-rotasional.

Sementara mesin bubut memiliki *chuck* pemegang benda kerja yang diputar oleh suatu spindel pada sumbu-Z dari mesin. Karena benda kerja dicekam oleh *chuck* yang berputar maka benda kerja otomatis turut memiliki sumbu putar. Pahat potong dipasangkan pada *toolpost* yang dapat bergerak ke kiri, ke kanan pada sumbu-Z serta maju dan mundur pada sumbu-X, yang digerakkan oleh *carriage* bersama *cross slide*. Produk yang dihasilkan tak lain adalah produk non-rotasional dengan radius yang dapat berubah sepanjang sumbu putarnya.

Untuk meningkatkan kecepatan produksi dan tingkat ketelitian hasil proses pemesinan ini, maka mesin-mesin perkakas umumnya dikendalikan dengan sistem kendali numerik (NC). Dengan sistem kendali numerik ini, banyak hal positif yang didapat, antara lain adalah:

1. geometri produk dapat disimpan dalam data digital berupa part program dan setiap saat dapat dipanggil kembali bila diperlukan;
2. kecepatan dan ketelitian mesin perkakas CNC untuk mengeksekusi proses pemotongan sangat tinggi dibandingkan dengan kemampuan manusia, yang memiliki ketrampilan tinggi sekalipun;
3. bila harus dikerjakan pada mesin yang lain, proses penyesuaian dapat dilakukan dengan mudah, dan masih banyak lagi hal-hal positif lain yang ada.

Dengan perkembangan jaman ke arah dunia digital akhir-akhir ini, kemajuan komputer sangat mendukung para produsen mesin perkakas untuk menciptakan komputer pengendali numerik yang sangat cerdas. Mesin perkakas yang menggunakan program NC untuk mengerjakan produk teknik tadi kemudian disebut mesin perkakas CNC (*computerized numerical control*). Pemrograman NC ini mengikuti suatu aturan baku yang disepakati secara internasional sehingga para pemakai mesin CNC dapat dengan mudah bertukar informasi dan saling mengerti satu sama lainnya.

Suatu mesin perkakas CNC dapat dioperasikan dengan cara menyiapkan suatu program NC yang mengikuti aturan-aturan tertentu. Aturan-aturan tersebut telah disepakati oleh dunia industri secara internasional dan telah dibuatkan standarisasinya sehingga setiap pembuat mesin ataupun pengguna mesin dapat sepaham dalam hal pengoperasiannya. Kesepahaman ini akan memudahkan bagi semua pihak untuk memanfaatkan semaksimal mungkin kemampuan mesin perkakas yang ada.

Mesin perkakas CNC telah membawa perubahan yang sangat berarti bagi industri pemotongan logam. Mesin perkakas CNC telah memungkinkan suatu industri untuk menghasilkan produk secara konsisten dengan akurasi yang belum pernah diimpikan sebelumnya. Produk-produk yang sama dapat dihasilkan dengan tingkat akurasi yang sama setiap saat pada kondisi apapun apabila program CNC telah disiapkan dengan baik/sepurna dan komputer pengendali diprogram dengan benar. Perintah-perintah operasi yang mengendalikan mesin perkakas dieksekusi secara otomatis dengan kecepatan, akurasi, efisiensi dan keterulangan yang sangat mengagumkan. Dengan meningkatnya penggunaan CNC di industri telah menciptakan kebutuhan banyak tenaga pemrogram yang memiliki pengetahuan dan kesiapan tentang penyiapan program NC yang akan mengarahkan mesin perkakas menghasilkan produk dengan bentuk dan akurasi yang diinginkan (Krar, 1999).

Pembuatan program NC harus mengikuti suatu format penulisan yang sudah disepakati secara internasional dan yang paling umum digunakan adalah format berbasis *word address* (Groover, 2005). Berbagai jenis kode dapat digunakan dalam pembuatan program NC, dimana kode-kode perintah untuk mesin dikelompokkan menjadi beberapa macam *NC words*. *Word* ini memiliki fungsi-fungsi tertentu yang berkaitan dengan urutan blok informasi, jenis gerakan yang ingin dihasilkan, nilai nominal gerakan,

pengaktifan fungsi-fungsi khusus pada mesin dan lain sebagainya. Dalam satu blok informasi, belum tentu semua *word* akan muncul secara bersamaan.

Format penulisan program NC ini harus mengikuti suatu standar yang berisi kode-kode perintah, yang merujuk pada standar EIA (*Electronic Industries Association*) dan disusun dalam suatu urutan logis berupa blok informasi. *Word address* adalah format pemrograman yang paling umum digunakan untuk sistem pemrograman CNC. Format ini mengandung sejumlah besar dari berbagai kode perintah (*preparatory* dan *miscellaneous*) dan nantinya digunakan sebagai dasar untuk mengirimkan informasi program kepada penggerak servo mesin, relay-relay, *micro-switch*, dan lain-lain untuk memproses benda kerja yang dimaksud. Masing-masing blok harus mengandung cukup informasi dalam melakukan suatu operasi pemesinan.

Untuk mendapatkan informasi secara lengkap dari suatu blok informasi program NC, maka suatu langkah-langkah terstruktur telah dibangun untuk dapat mengekstrak informasi yang berkaitan dengan gerakan mesin dan pengaktifan fungsi-fungsi khusus pada mesin perkakas. Semua data yang tidak valid harus dikeluarkan atau segera mendapat perbaikan, dan semua data-data yang valid harus dikumpulkan dalam suatu kontainer. Isi kontainer ini diolah dengan algoritma yang telah dikembangkan dan hasilnya bisa dikirim ke unit pengendali mesin agar diolah dan dikirim ke unit-unit yang bertugas menjalankan fungsi-fungsi tertentu pada mesin perkakas.

Hingga saat ini, penelitian terdahulu yang dilakukan telah memberi langkah awal dalam kajian tentang teknik pemrograman NC suatu mesin milling. Disamping itu suatu media/fasilitas pemrograman yang mampu menguji keabsahan suatu program NC. Selanjutnya fasilitas ini juga dapat melakukan proses ekstraksi G-code yang terkait dengan gerakan mesin untuk operasi pemesinan tertentu. Untuk menyiapkan gerakan simulasi, beberapa formulasi lintasan mata pahat telah dijabarkan dengan cukup mendetail.

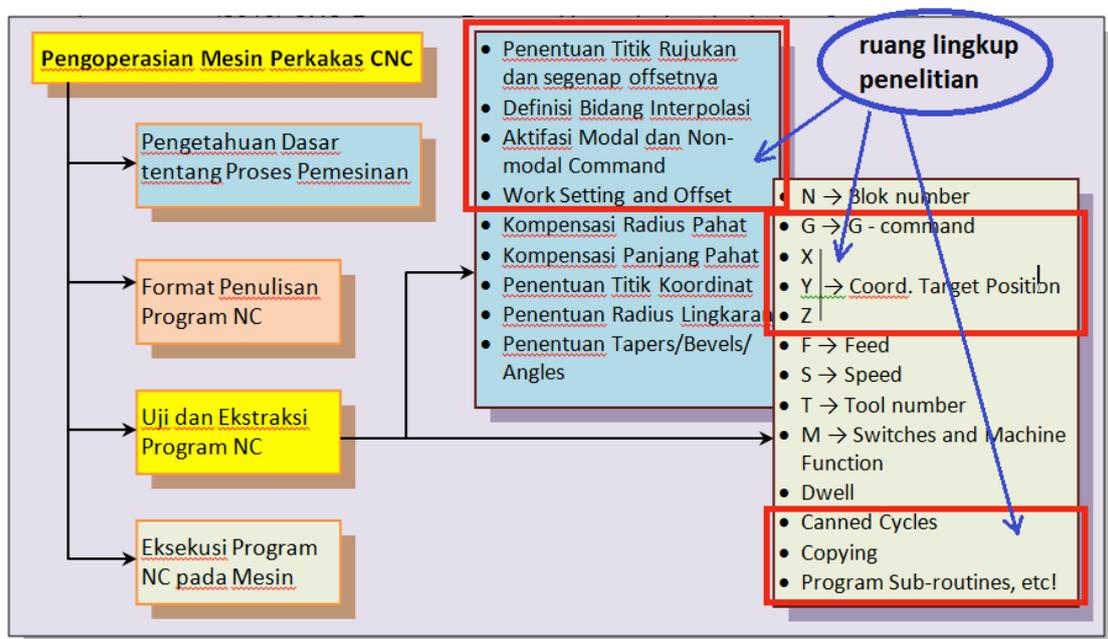
Akan tetapi beberapa fungsi utama untuk melakukan operasi pemotongan logam pada mesin perkakas milling CNC masih banyak yang harus dikerjakan. Fungsi-fungsi khusus untuk pengaturan benda kerja dan pahat terhadap titik rujukan mesin, gerakan-gerakan kombinasi yang lebih rumit dan fitur-fitur operasi duplikasi masih perlu mendapat pengolahan lebih mendetail.

BAB III

PENGEMBANGAN ALGORITMA PENGUJI DAN PENGEKSTRAK G-CODE

3.1 Ruang Lingkup Pemrograman Mesin Perkakas CNC

Secara garis besar dapat dijelaskan bahwa untuk mengoperasikan mesin perkakas CNC maka diperlukan penguasaan beberapa fungsi dasar seperti yang disajikan pada Gambar 3.1. Pada penelitian ini, fokus pekerjaan ditekankan pada fungsi-fungsi khusus seperti Penentuan titik rujukan, Definisi bidang interpolasi, Work setting dan offset, Kelengkapan G, X, Y dan Z word, Canned cycle, Copying, dimana didalamnya terdapat banyak fungsi-fungsi pendukungnya. Dalam penelitian inipun belum semua fungsi pendukung dapat dibahas karena luasnya topik bahasan.



Gambar 3.1 Ruang lingkup bahasan dalam pemrograman mesin CNC

3.2 Format Dasar Penulisan Pemrograman NC (Numerical Control)

Pembuatan program NC harus mengikuti suatu format penulisan yang sudah disepakati secara internasional dan yang paling umum digunakan adalah format berbasis *word address*. Jenis-jenis kode yang digunakan dalam pembuatan program NC secara umum disajikan dalam Tabel 1, dimana kode-kode perintah untuk mesin dikelompokkan menjadi beberapa macam *NC words*. *Word* ini memiliki fungsi-fungsi tertentu yang berkaitan dengan urutan blok informasi, jenis gerakan yang ingin dihasilkan, nilai nominal gerakan, pengaktifan fungsi-fungsi khusus pada mesin dan lain sebagainya. Dalam satu blok informasi, belum tentu semua *word* ini akan muncul secara bersamaan.

Tabel 1. Definisi dari beberapa NC words (G-code) yang umum digunakan [Groover, 2005].

NC Word	Penggunaan
N	<i>Sequence number</i> : mengidentifikasi nomor blok informasi
G	<i>Preparatory function</i> : memilih fungsi kontrol yang berbeda, termasuk berbagai rutin pemesinan yang telah diprogram sebelumnya
X, Y, Z, R, I, J, K	<i>Dimension coordinate data</i> : perintah gerak linier dan melingkar bagi sumbu-sumbu mesin
F	<i>Feed function</i> : menentukan kecepatan makan (feed rate) saat operasi
S	<i>Speed function</i> : menentukan kecepatan makan
T	<i>Tool function</i> : memberitahu mesin dimana lokasi pahat di dalam <i>tool holder</i> atau <i>tool turret</i>
M	<i>Miscellaneous function</i> : menghidupkan/mematikan coolant, membuka spindel, membalik putaran spindel, ganti pahat, dll.
EOB	<i>End of block</i> : menunjukkan kepada CPU bahwa semua blok informasi telah dihentikan.

Format penulisan program NC ini harus mengikuti suatu standar yang berisi kode-kode perintah, yang merujuk pada standar EIA (*Electronic Industries Association*) dan disusun dalam suatu urutan logis yang disebut dengan istilah blok informasi. *Word address* adalah format pemrograman yang paling umum digunakan untuk sistem pemrograman CNC. Format ini mengandung sejumlah besar dari berbagai kode perintah (*preparatory* dan *miscellaneous*) yang mengirimkan informasi program kepada penggerak servo mesin, relay-relay, micro-switches, dan lain-lain untuk memproses benda kerja yang dimaksud. Masing-masing blok harus mengandung cukup informasi untuk melakukan suatu operasi pemesinan. Kode perintah dasar yang paling sering digunakan adalah kode yang disajikan pada Tabel 2.

Kode M atau *miscellaneous codes* digunakan apakah untuk menghidupkan (ON) ataupun mematikan (OFF) berbagai fungsi khusus yang mengendalikan operasi tertentu pada mesin perkakas. Jenis-jenis M-code yang umum digunakan disajikan dalam Tabel 3. M-code tidak dikelompokkan ke dalam kategori tertentu, walaupun beberapa kode berbeda bisa saja mengendalikan jenis operasi yang sama seperti M03, M04, dan M05 yang mengendalikan *spindle* mesin perkakas, misalnya:

- M03 memutar spindle se-arah putaran jarum jam
- M04 memutar spindle berlawanan-arah putaran jarum jam
- M05 mematikan putaran spindle, dll.

Tabel 2. Sebagian dari kelompok G-code yang paling sering digunakan [Krar, 1999].

Group	Code	Function
01	G00	Rapid positioning
01	G01	Linear interpolation
01	G02	Circular interpolation clockwise (CW)
01	G03	Circular interpolation counterclockwise (CCW)
06	G20*	Inch input (in.)
06	G21*	Metric input (mm)
	G24	Radius programming (**)
00	G28	Return to reference point
00	G29	Return from reference point
	G32	Thread cutting (**)
07	G40	Cutter compensation cancel
07	G41	Cutter compensation left
07	G42	Cutter compensation right

Tabel 3. Beberapa M-code yang umum dipakai pada pemrograman NC [Krar, 1999].

Code	Function
M00	Program stop
M02	End of program
M03	Spindle start (forward CW)
M04	Spindle start (reverse CCW)
M05	Spindle stop
M06	Tool change
M08	Coolant on
M09	Coolant off

Contoh menuliskan satu perintah yang dikirim kepada suatu pengendali mesin adalah dalam bentuk satu blok informasi, sebagai berikut:

“N001 G90 G21 G01 X1.234 Y-5.678 Z0.321 S300 F100 M03”

dengan penjelasan:

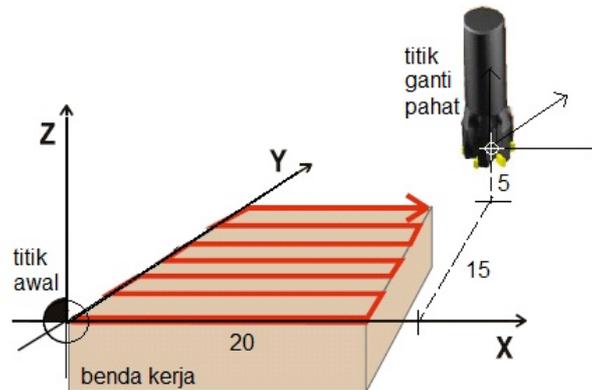
- N001** : menyatakan nomor urut operasi yang dilakukan.
- G90** : menyatakan mode dimensi yang dipakai adalah absolut.
- G21** : mode input unit dimensi yang digunakan adalah mm.
- G01** : menyatakan gerak interpolasi linear.
- X123** : akan menggerakkan meja 123 mm dalam arah positif sepanjang sumbu-X.
- Y-456** : akan menggerakkan meja 456 mm sepanjang sumbu-Y negatif.
- Z432** : akan menggerakkan meja 432 mm sepanjang sumbu-Z positif.
- S300** : memutar spindel dengan kecepatan 300 rpm
- F100** : menggerakkan spindel dengan kecepatan makan 100 mm/dtk
- M03** : menghidupkan spindel dengan arah CW.

3.3 Penentuan Titik Referensi Mesin/Pahat Relatif Terhadap Benda Kerja

Untuk menjamin bahwa titik-titik tujuan yang dinyatakan dalam suatu blok informasi dapat dicapai mesin dengan benar, maka perlu dipastikan bahwa posisi awal benda kerja relatif terhadap titik referensi mesin telah diketahui sebelumnya. Dalam

teknik pemrograman NC, beberapa fungsi baku telah disiapkan salah satunya adalah kode G-92. Kode ini memerlukan informasi berupa jarak 3-dimensi titik awal benda kerja ke titik awal mesin (machine zero point/tool change position). Dalam satu blok informasi, penentuan titik rujukan uni dinyatakan dalam bentuk sbb dan secara grafis disajikan pada Gambar 3.2:

N002 G92 X20 Y15 Z5 (posisi tool 20mm di kanan, 15mm di depan dan 5mm di atas titik awal benda kerja)



Gambar 3.2 Posisi tools relatif terhadap titik awal benda kerja

Informasi awal ini akan digunakan untuk memodifikasi semua perintah gerak dengan kode G yang muncul setelah blok informasi ini. Pada penelitian sebelumnya [Arthaya, 2010], setiap blok informasi diekstrak oleh algortima khusus untuk mendapatkan semua data yang terkait dengan informasi gerak yang harus dilakukan oleh mesin ke setiap sumbu gerak. Dengan adanya kode G-92 sebelum G-code utama, maka proses modifikasi nominal gerak yang dihasilkan adalah sebagai berikut:

N002 G90 G92 X20 Y15 Z5 (penentuan referensi input pada koordinat absolut)

N003 G00 X10 Y5 Z2 (pergi dengan koordinat absolut ke titik 10,5,2 dengan gerakan cepat)

Konsekuensi dari kombinasi blok informasi ini (konversi data pada gerak cepat) adalah:

a) Konversi data titik referensi pahat;

- o $X_{t_ref} = N(i, x) = current_val$; (nilai aktual= 20)
- o $Y_{t_ref} = N(i, y) = current_val$; (nilai aktual= 15)
- o $Z_{t_ref} = N(i, z) = current_val$; (nilai aktual= 5)

b) Konversi data pada gerak cepat;

Pada gerak cepat, data yang didapat dari proses ekstraksi adalah:

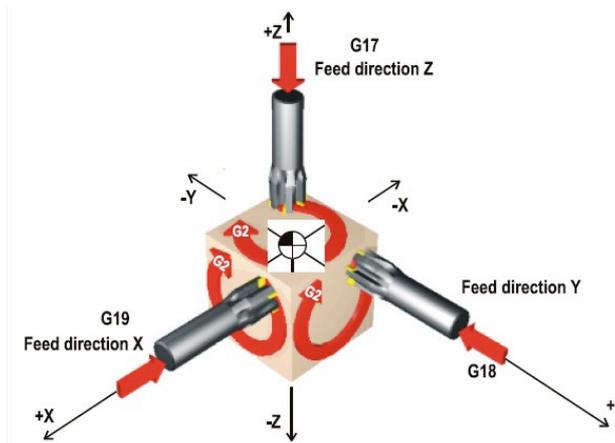
- o Ambil nilai BLU dan kec. maksimum motor penggerak max_sp (dlm BLU/s);
- o $Kec = N(i, vm) = max_sp$;
- o $X_rpid = N(i, x) = current_val$; (nilai aktual= 10)
- o $Y_rpid = N(i, y) = current_val$; (nilai aktual= 5)
- o $Z_rpid = N(i, z) = current_val$; (nilai aktual= 2)
- o $X_{dest} = X_rpid - X_{t_ref}$; $\rightarrow xPulsa = X_{dest}/BLU$; (nilai aktual $X_{dest} = -10$)
- o $Y_{dest} = Y_rpid - Y_{t_ref}$; $\rightarrow yPulsa = Y_{dest}/BLU$; (nilai aktual $Y_{dest} = -10$)
- o $Z_{dest} = Z_rpid - Z_{t_ref}$; $\rightarrow zPulsa = Z_{dest}/BLU$; (nilai aktual $Z_{dest} = -3$)
- o Nilai $xPulsa$, $yPulsa$ dan $zPulsa$ ini dikirim ke DDA-I untuk masing-masing unit pengendali motor penggerak.

3.4 Penentuan Bidang Interpolasi Tempat Pahat Bekerja

Setiap gerakan pahat potong di atas benda kerja dijabarkan terhadap suatu bidang kerja yang terdapat pada benda kerja. Secara alami (default) bidang kerja yang dikenali dalam pemrograman NC adalah bidang x-y bila sang programmer tidak menyatakan bidang kerja dimana program akan dilaksanakan.

Apabila proses pemotongan harus dilakukan pada bidang referensi lain, maka kode-G khusus harus dinyatakan sebelum blok informasi untuk proses pemotongan dilakukan. Kode-G yang digunakan adalah G18 untuk bekerja pada bidang y-z dan G19 untuk bekerja pada bidang z-x.

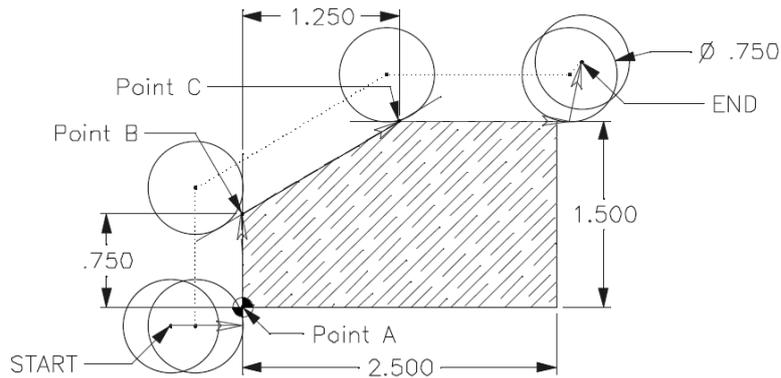
Secara prinsip proses perubahan bidang referensi berarti perubahan sumbu gerakan pahat terhadap titik awal benda kerja yang telah dibayangkan oleh sang programmer. Semua interpolasi gerak pada bidang x-y (G17) bila dikerjakan pada bidang y-z berarti bahwa nilai pada sumbu-x direalisasi pada sumbu y sedangkan nilai-nilai pada sumbu-y akan direalisasi pada sumbu z. Yang perlu diperhatikan adalah arah pahat potong sebaiknya datang dari arah tegak lurus terhadap bidang referensi seperti ditunjukkan pada Gambar 3.3.



Gambar 3.3 Arah pahat tegak lurus terhadap bidang referensi pada benda kerja

3.5 Penentuan Work Setting and Offset pada Benda Kerja

Sebelum benda kerja diproses pada mesin perkakas, maka benda kerja harus diposisikan dengan pasti pada meja mesin dan direferensikan kepada titik referensi pahat, dalam hal ini adalah tool change position. Dalam moda pemosisian absolut koordinat awal benda kerja biasanya didefinisikan pada salah satu titik pojok yang berupa pertemuan sisi-sisi benda kerja yang memiliki tingkat kepastian yang paling tinggi. Selanjutnya semua ukuran dimensi dirujuk dari titik acuan tersebut. Pada moda pemosisian inkremental, maka titik koordinat awal benda kerja selalu berubah pada setiap pergantian blok informasi gerakan yang akan dilaksanakan seperti yang disajikan pada Gambar 3.4.

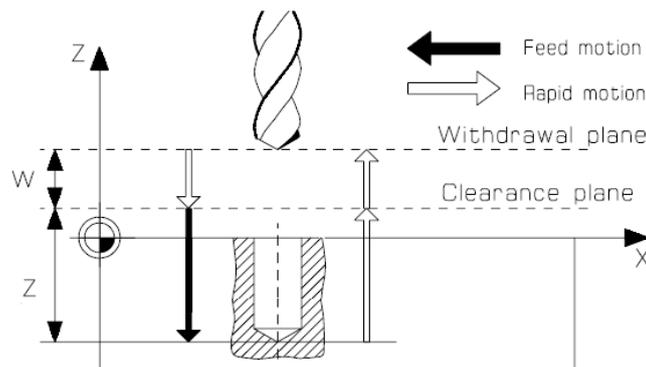


Gambar 3.4 Dimensi absolute menentukan titik tujuan secara langsung.

Semua titik koordinat pada benda kerja akan langsung menjadi titik tujuan pada setiap blok program NC dan titik. Titik awal benda kerja akan tetap pada titik A dan berlaku bagi titik B, C dan titik-titik lainnya. Sementara pada moda inkremental, jarak tempuh ke titik A adalah Y 0.750 dan setelah sampai di titik B, maka titik awal benda kerja berubah ke B dan jarak titik tujuan menjadi X 1.250 dan Y 0.750.

3.6 Pengembangan Canned Cycles untuk Fungsi Dwell

Canned cycle sangat banyak digunakan pada pemrograman mesin perkakas yang tujuannya adalah untuk mempermudah/menyederhanakan program NC untuk gerakan-gerakan yang berulang dengan pola gerak yang sejenis. Salah satu jenis cycle yang dikembangkan adalah gerakan masuk-keluar di dalam benda kerja pada saat seorang programmer membuat lubang. Sebagai contoh adalah program untuk membuat lubang seperti pada Gambar 3.5. Contoh kode-G yang digunakan untuk membuat fitur lubang ini adalah sebagai berikut:



Gambar 3.5 Langkah-langkah gerak inkremental untuk membuat lubang.

N010 G81 Z-20 W+10 (cycle yang dipakai G81, kedalaman Z -20 dari clearance plane sejauh W +10 ke withdrawal plane)
N011 G79 (kode untuk menonaktifkan kode G81)

Dari Gambar 3.5 terlihat bahwa untuk membuat sebuah lubang terdapat setidaknya 4 langkah gerak vertikal, yakni 2x rapid sepanjang W dan 1x feeding dan 1x rapid sepanjang Z. Disamping itu untuk kemudahan pemrograman, pada jenis cycle ini

pergerakan arah sumbu-Z dilakukan pada mode inkremental. Dengan adanya fungsi khusus drill ini maka perintah membuat lubang menjadi sangat sederhana. Untuk merealisasi dalam media simulator yang dikembangkan maka diperlukan perhitungan-perhitungan sebagai berikut:

N010 G81 Z-20 W+10 (fungsi cycle untuk membuat lubang sekali langkah)

- a) Konversi data parameter gerak pahat;
 - $Z_temp = N(i, z) = current_val$; (nilai aktual= -20)
 - $W_temp = N(i, w) = current_val$; (nilai aktual= +10)
- b) Konversi data pada gerak cepat dan gerak interpolasi;

Pada gerak cepat dan makan, data yang didapat dari proses ekstraksi adalah:

 - Ambil nilai BLU dan kec. maksimum motor penggerak max_sp (dlm BLU/s);
 - $Kec = N(i, vm) = max_sp$;
 - $Zdest = W_temp$; → $zPulsa = Zdest/BLU$; (nilai aktual $Zdest = -10$, mode G00)
 - Nilai $zPulsa$ ini dikirim ke DDA-I untuk unit pengendali motor penggerak Z
 - $Zdest = Z_temp$; → $zPulsa = Zdest/BLU$; (nilai aktual $Zdest = -20$, mode G01)
 - Nilai $zPulsa$ ini dikirim ke DDA-I untuk unit pengendali motor penggerak Z
 - $Zdest = Z_temp + W_temp$; → $zPulsa = Zdest/BLU$; (nilai aktual $Zdest = +20$, mode G00)
 - Nilai $zPulsa$ ini dikirim ke DDA-I untuk unit pengendali motor penggerak Z

Sementara gerak semacamnya dapat dilakukan dengan menciptakan gerakan secara bertahap seperti pada Gambar 3.6 (kiri) dengan memanfaatkan kode G82. Gerakan yang dilakukan gerak makan bertahap naik dan turun yang bertujuan untuk memberikan kesempatan bagi chip (geram) untuk patah dalam area pemotongan (dengan kata lain chip-breaking). Model perintah gerak untuk mendapatkan lubang dengan gerak makan secara bertahap adalah sebagai berikut:

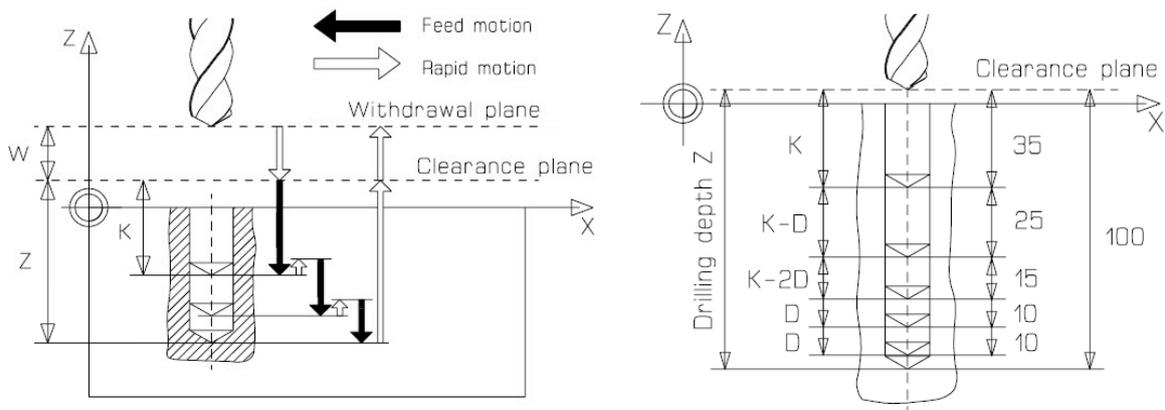
N010 G82 Z-47 W+5 B+0.5 D+5 K+20

- a) Konversi data parameter gerak pahat;
 - $Z_temp = N(i, z) = current_val$; (nilai aktual= -47)
 - $W_val = N(i, w) = current_val$; (nilai aktual= +5, jarak ke withdrawal plane)
 - $B_val = N(i, b) = current_val$; (nilai aktual= +0.5, waktu dwell)
 - $D_val = N(i, d) = current_val$; (nilai aktual= +5, angka degresi)
 - $K_val = N(i, k) = current_val$; (nilai aktual= +20, kedalaman drill pertama)
- b) Konversi data pada gerak cepat dan gerak interpolasi;

Pada gerak cepat dan makan, data yang didapat dari proses ekstraksi adalah:

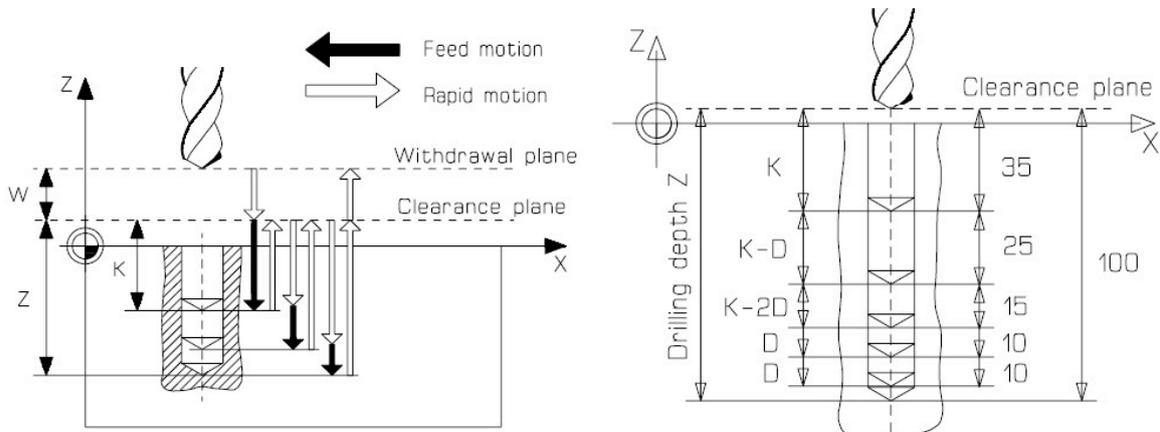
 - Ambil nilai BLU dan kec. maksimum motor penggerak max_sp (dlm BLU/s);
 - $Kec = N(i, vm) = max_sp$;
 - Panggil algoritma penghitung urutan kedalaman berdasarkan nilai total kedalaman Z, nilai kedalaman awal K dan nilai degresi D (didapat: 20, 15, 12 dan 2) disebut Z_degX .
 - $Zdest = W_temp$; → $zPulsa = Zdest/BLU$; (nilai aktual $Zdest = -5$, mode G00)
 - Nilai $zPulsa$ ini dikirim ke DDA-I untuk unit pengendali motor penggerak Z
 - $Zdest = Z_deg1$; → $zPulsa = Zdest/BLU$; (nilai aktual $Zdest = -20$, mode G01)
 - Nilai $zPulsa$ ini dikirim ke DDA-I untuk unit pengendali motor penggerak Z
 - $Zdest = 1$; → $zPulsa = Zdest/BLU$; (nilai aktual $Zdest = +1$, mode G00)
 - Nilai $zPulsa$ ini dikirim ke DDA-I untuk unit pengendali motor penggerak Z

- $Z_{dest} = Z_{deg2}$; $\rightarrow zPulsa = Z_{dest}/BLU$; (nilai aktual $Z_{dest} = -15$, mode G01)
- Nilai $zPulsa$ ini dikirim ke DDA-I untuk unit pengendali motor penggerak Z
- $Z_{dest} = 1$; $\rightarrow zPulsa = Z_{dest}/BLU$; (nilai aktual $Z_{dest} = +1$, mode G00)
- Nilai $zPulsa$ ini dikirim ke DDA-I untuk unit pengendali motor penggerak Z
- $Z_{dest} = Z_{deg3}$; $\rightarrow zPulsa = Z_{dest}/BLU$; (nilai aktual $Z_{dest} = -10$, mode G01)
- Nilai $zPulsa$ ini dikirim ke DDA-I untuk unit pengendali motor penggerak Z
- $Z_{dest} = 1$; $\rightarrow zPulsa = Z_{dest}/BLU$; (nilai aktual $Z_{dest} = +1$, mode G00)
- Nilai $zPulsa$ ini dikirim ke DDA-I untuk unit pengendali motor penggerak Z
- $Z_{dest} = Z_{deg4}$; $\rightarrow zPulsa = Z_{dest}/BLU$; (nilai aktual $Z_{dest} = -20$, mode G01)
- Nilai $zPulsa$ ini dikirim ke DDA-I untuk unit pengendali motor penggerak Z
- $Z_{dest} = Z_{temp} + W_{temp}$; $\rightarrow zPulsa = Z_{dest}/BLU$; (nilai aktual $Z_{dest} = +47$, mode G00)
- Nilai $zPulsa$ ini dikirim ke DDA-I untuk unit pengendali motor penggerak Z



Gambar 3.6 Langkah pembuatan lubang dengan jeda pemotongan chip.

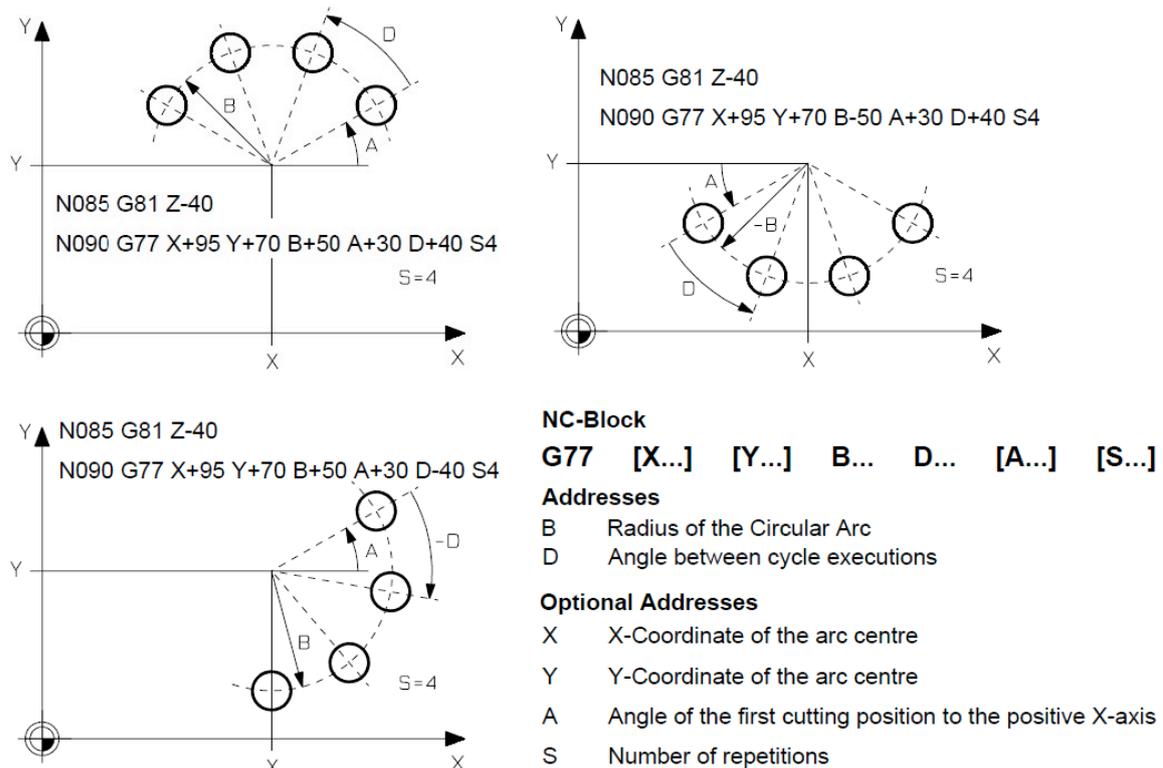
Dengan cara yang sama, drilling dengan kode G83 dapat dilakukan dengan perbedaan utama adalah adanya gerakan pahat total keluar hingga clearance plane selama dan diam dalam waktu tertentu untuk memberi kesempatan bagi chip untuk keluar sepenuhnya dari area pemotongan. Gerakan keluar ini dilakukan untuk setiap kedalaman degresi yang dihasilkan oleh algoritma pembangkit kedalaman degresi.



Gambar 3.7 Langkah pembuatan lubang dengan jeda pemotongan dan membuang chip keluar dari area lubang drill.

3.7 Pengembangan Algoritma Copying

Fungsi copying sangat diperlukan dalam rangka membuat fitur pemesinan sejenis dalam satu pola yang diinginkan. Mengcopy fitur pemesinan pada suatu lintasan tertentu sering diperlukan pada pembuatan cetakan dan lain-lain. Banyak macam fitur copying yang dapat dilakukan, salah satunya adalah mengcopy pada suatu lintasan berupa potongan busur seperti yang disajikan dalam Gambar 3.8.

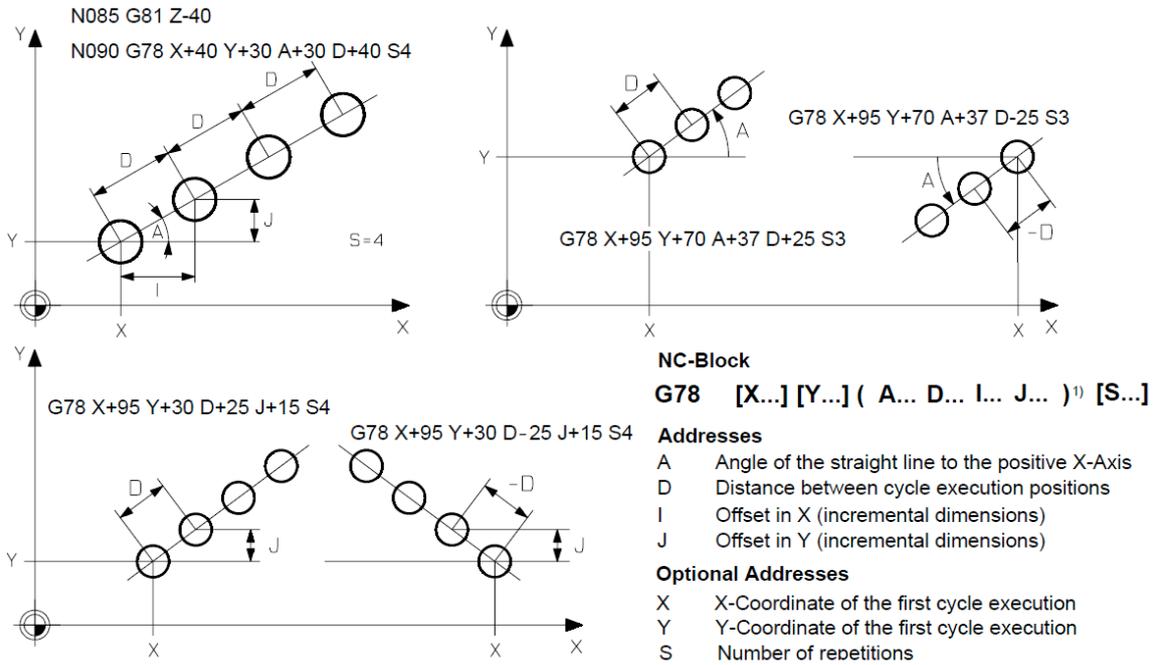


Gambar 3.8 Berbagai jenis pola copying pada suatu lintasan busur.

Jenis fitur copying yang lainnya juga sering digunakan untuk tujuan pembuatan lubang berurutan pada sepenggal garis lurus, seperti yang ditunjukkan pada Gambar 3.9. Beberapa parameter harus disiapkan agar arah dan jumlah batang yang harus dibuat terjadi pada tempat yang tepat.

Pada prinsipnya fitur copying ini bisa dilaksanakan dengan pembuatan sub-routine yang sangat sederhana, yakni suatu fungsi yang memiliki kemampuan untuk:

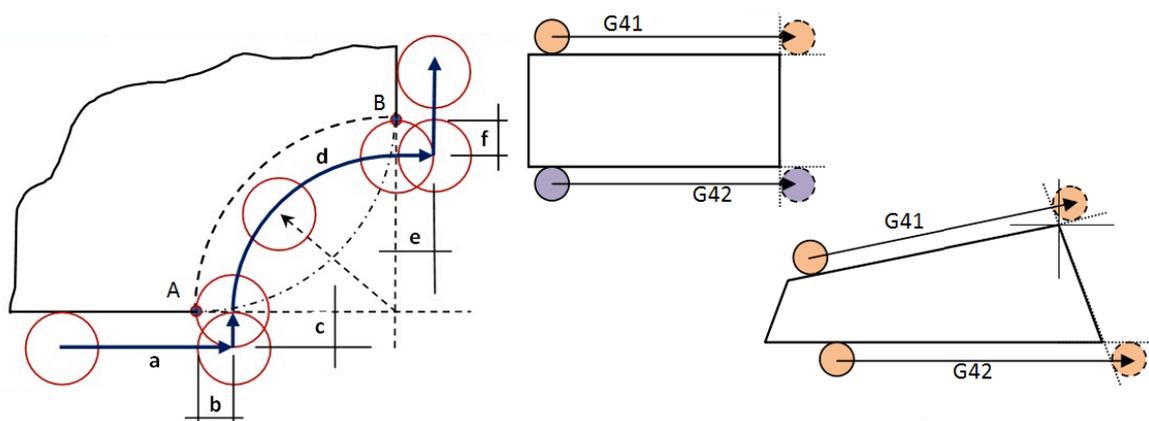
- Menghitung koordinat titik tujuan berikut terdekat, sesuai dengan moda lintasan copying yang dipilih, apakah lintasan busur ataupun garis lurus;
- Arah penentuan titik tujuan berikut terdekat;
- Memastikan moda gerakan menjadi inkremental;
- Gerakan awal hingga kedalaman yang diinginkan menggunakan kode G01;
- Gerakan kembali ke bidang clearance dengan kode G00;
- Gerakan ke titik tujuan berikut terdekat dengan kode G00.



Gambar 3.9 Berbagai jenis pola copying pada suatu lintasan garis lurus.

3.9 Formulasi Khusus Untuk Perpotongan Sembarang untuk G41 dan G42

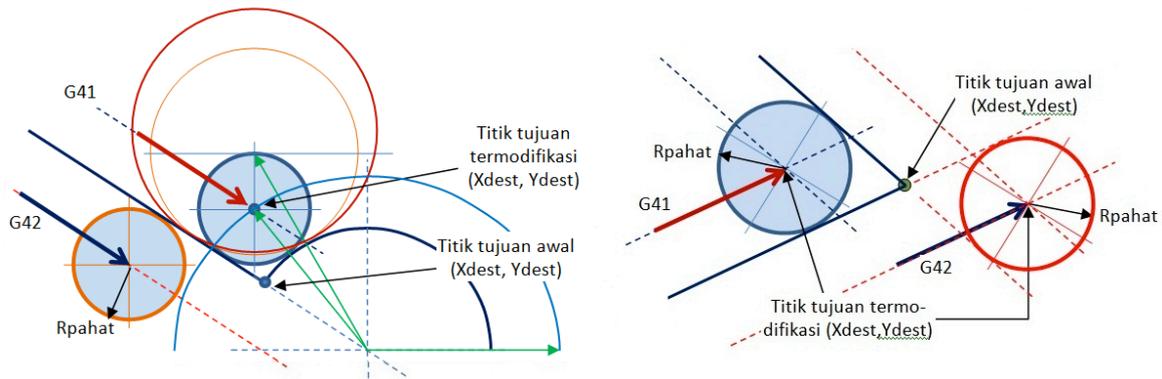
Pada penelitian sebelumnya [Arthaya, 2010] pengekstrasian data pada blok informasi yang menggunakan Cutter Radius Compensation (G41 dan G42) telah dibahas secara mendasar. Beberapa pola perpotongan sederhana telah berhasil dijabarkan terkait dengan penggunaan pahat yang memiliki radius bermacam-macam. Garis perpotongan sederhana seperti dua garis tegak lurus, dua garis lurus berpotongan dengan sudut tumpul tertentu, garis lurus berpotongan dengan lingkaran pada sudut 90° telah dijabarkan seperti ditunjukkan pada Gambar 3.10.



Gambar 3.10 Kompensasi radius posisi pahat pada G41/G42 di sekitar benda kerja

Pada penelitian tersebut, titik tujuan pahat dapat dengan mudah ditentukan dengan perhitungan geometri sederhana. Sejah lingkaran sebesar 90° berpotongan dengan garis lurus baik horizontal maupun garis miring, maka perhitungan koordinat tujuan termodifikasi dengan mudah dapat ditentukan.

Dalam penelitian kali ini dua macam gerakan dengan kode G41 dan G42 akan dibahas, yakni perhitungan titik tujuan termodifikasi saat perpotongan garis lurus dengan sudut tumpul dan perpotongan garis lurus dengan sepotong busur pada sembarang sudut potong. Pada gambar 3.11 terlihat dua kasus dimana titik tujuan awal mata pahat harus dimodifikasi agar fungsi kompensasi radius pahat dapat berjalan dengan baik.



Gambar 3.11 Perubahan titik tujuan pada berbagai posisi dengan kode G41 dan G42.

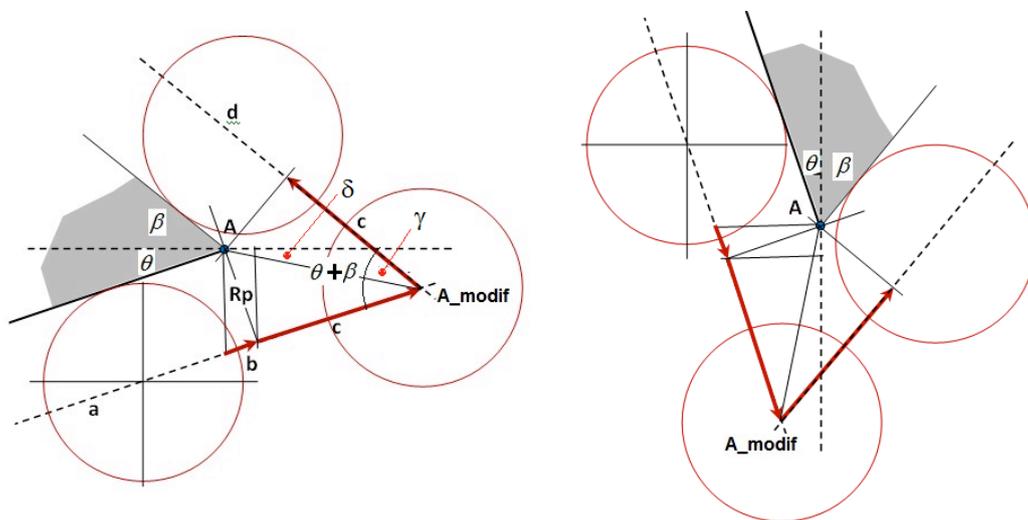
Untuk kasus perpotongan dua garis lurus dengan sudut sembarang, maka titik tujuan awal A harus dimodifikasi menjadi titik A_modif, sesuai dengan besarnya radius pahat seperti pada Gambar 3.12. Titik A_modif yang terjadi pada penggunaan kode G42, dapat dihitung dengan cara sebagai berikut:

- Kasus 1: sudut θ di kwadrant III dan β di kwadrant II, dan $\beta > \theta$ maka, posisi A_modif ada di kwadrant IV.
- Kasus 2: sudut θ di kwadrant III dan β di kwadrant II, dan $\beta < \theta$ maka, posisi A_modif ada di kwadrant I.
- Kasus 3: sudut θ di kwadrant II dan β di kwadrant I, dan $\beta > \theta$ maka, posisi A_modif ada di kwadrant III.
- Kasus 4: sudut θ di kwadrant II dan β di kwadrant I, dan $\beta < \theta$ maka, posisi A_modif ada di kwadrant IV.
- Kasus 5: sudut θ di kwadrant I dan β di kwadrant IV, dan $\beta > \theta$ maka, posisi A_modif ada di kwadrant II.
- Kasus 6: sudut θ di kwadrant I dan β di kwadrant IV, dan $\beta < \theta$ maka, posisi A_modif ada di kwadrant III.
- Kasus 7: sudut θ di kwadrant IV dan β di kwadrant III, dan $\beta > \theta$ maka, posisi A_modif ada di kwadrant I.
- Kasus 7: sudut θ di kwadrant IV dan β di kwadrant III, dan $\beta < \theta$ maka, posisi A_modif ada di kwadrant II.

Sebagai contoh dasar adalah perhitungan titik A_modif untuk Kasus 1:

- Panjang jarak A-A_modif, $L_{\text{modif}} = R_p \cdot \sin\{(\theta + \beta)/2\}$
- Sudut $\gamma = (\theta + \beta)/2$
- Sudut $\delta = (180 - \beta) - \gamma$
- $L_{\text{modif}_x} = L_{\text{modif}} \cos(\gamma)$
- $L_{\text{modif}_y} = -L_{\text{modif}} \sin(\gamma)$

- f) Konversi data pada gerak interpolasi linier dengan kompensasi radius pahat G42; Pada gerak interpolasi linier, data yang didapat dari proses ekstraksi adalah:
1. Ambil nilai BLU dari masing-masing sumbu gerak;
 2. Kec. makan = $N(i, f) = \text{current_val}$;
 3. $L = \sqrt{X_{\text{dest}}^2 + Y_{\text{dest}}^2}$; \rightarrow juga identikan dengan $L = \sqrt{a^2 + b^2}$;
 4. $\rightarrow x_{\text{Pulsa}} = X_{\text{dest}}/\text{BLU}$;
 5. $\rightarrow y_{\text{Pulsa}} = Y_{\text{dest}}/\text{BLU}$;
 6. $\text{FRN} = 10 * (\text{current_val} / L)$;
 7. Frek. untuk DDA-i (1&2) $\rightarrow f_o = \text{FRN}(f/2^m)$; dimana m: ukuran register ke-1&2
 8. Frek. untuk DDA-i (3) $\rightarrow f = 2^{m+n}/600$; dimana n: ukuran register ke-3
 9. Nilai x_{Pulsa} dan y_{Pulsa} ini dikirim ke DDA-i (gunakan perbandingan pulsa) untuk masing-masing unit pengendali motor penggerak.

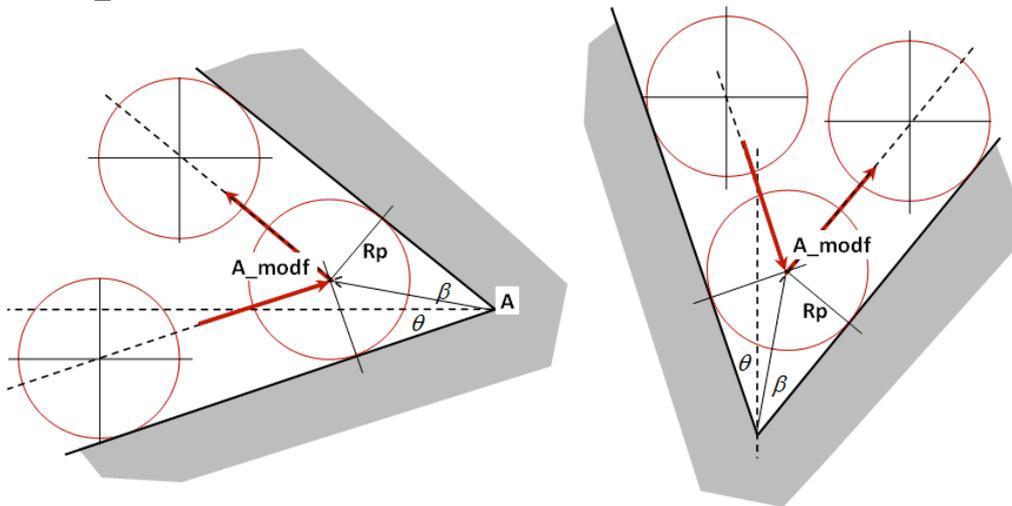


Gambar 3.12 Perubahan posisi titik tujuan pada perpotongan dua garis lurus sembarang dengan menggunakan kode G42.

Dengan cara yang sama maka perhitungan titik tujuan termodifikasi A_{modif} untuk kasus-kasus yang lain dapat dihitung dengan mudah. Pola perhitungan sejenis untuk penggunaan kode G41 dapat dijabarkan dengan ilustrasi pada Gambar 3.13. Bila diperhatikan dengan seksama, maka segi-3 penentu posisi titik tujuan A_{modif} adalah identik dengan perhitungan untuk kasus pada Gambar 3.12. Dengan pengertian sederhana ini maka untuk kasus penggunaan kode G41 didapat berbagai kondisi berikut:

- a) Kasus 1: sudut θ di kwadrant III dan β di kwadrant II, dan $\beta > \theta$ maka, posisi A_{modif} ada di kwadrant II.
- b) Kasus 2: sudut θ di kwadrant III dan β di kwadrant II, dan $\beta < \theta$ maka, posisi A_{modif} ada di kwadrant III.
- c) Kasus 3: sudut θ di kwadrant II dan β di kwadrant I, dan $\beta > \theta$ maka, posisi A_{modif} ada di kwadrant I.
- d) Kasus 4: sudut θ di kwadrant II dan β di kwadrant I, dan $\beta < \theta$ maka, posisi A_{modif} ada di kwadrant II.

- e) Kasus 5: sudut θ di kwadrant I dan β di kwadrant IV, dan $\beta > \theta$ maka, posisi A_{modif} ada di kwadrant IV.
- f) Kasus 6: sudut θ di kwadrant I dan β di kwadrant IV, dan $\beta < \theta$ maka, posisi A_{modif} ada di kwadrant I.
- g) Kasus 7: sudut θ di kwadrant IV dan β di kwadrant III, dan $\beta > \theta$ maka, posisi A_{modif} ada di kwadrant III.
- h) Kasus 7: sudut θ di kwadrant IV dan β di kwadrant III, dan $\beta < \theta$ maka, posisi A_{modif} ada di kwadrant IV.

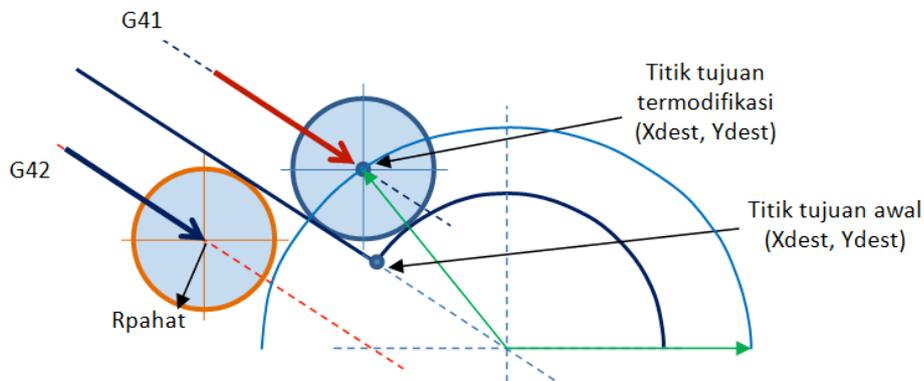


Gambar 3.13 Penentuan koordinat posisi titik tujuan termodifikasi pada perpotongan dua garis lurus sembarang.

Sebagai contoh dasar adalah perhitungan titik A_{modif} untuk Kasus 1:

- a) Panjang jarak $A-A_{\text{modif}}$, $L_{\text{modif}} = R_p \cdot \sin\{(\theta + \beta)/2\}$
- b) Sudut $\gamma = (\theta + \beta)/2$
- c) Sudut $\delta = (180 - \beta) - \gamma$
- d) $L_{\text{modif}_x} = -L_{\text{modif}} \cos(\gamma)$
- e) $L_{\text{modif}_y} = L_{\text{modif}} \sin(\gamma)$
- f) Konversi data pada gerak interpolasi linier dengan kompensasi radius pahat G42; Pada gerak interpolasi linier, data yang didapat dari proses ekstraksi adalah:
 1. Ambil nilai BLU dari masing-masing sumbu gerak;
 2. Kec. makan = $N(i, f) = \text{current_val}$;
 3. $L = \sqrt{X_{\text{dest}}^2 + Y_{\text{dest}}^2}$; \rightarrow juga identikan dengan $L = \sqrt{a^2 + b^2}$;
 4. $\rightarrow x_{\text{Pulsa}} = X_{\text{dest}}/\text{BLU}$;
 5. $\rightarrow y_{\text{Pulsa}} = Y_{\text{dest}}/\text{BLU}$;
 6. $\text{FRN} = 10 * (\text{current_val} / L)$;
 7. Frek. untuk DDA-i (1&2) $\rightarrow f_0 = \text{FRN}(f/2^m)$; dimana m: ukuran register ke-1&2
 8. Frek. untuk DDA-i (3) $\rightarrow f = 2^{m+n}/600$; dimana n: ukuran register ke-3
 10. Nilai x_{Pulsa} dan y_{Pulsa} ini dikirim ke DDA-i (gunakan perbandingan pulsa) untuk masing-masing unit pengendali motor penggerak.

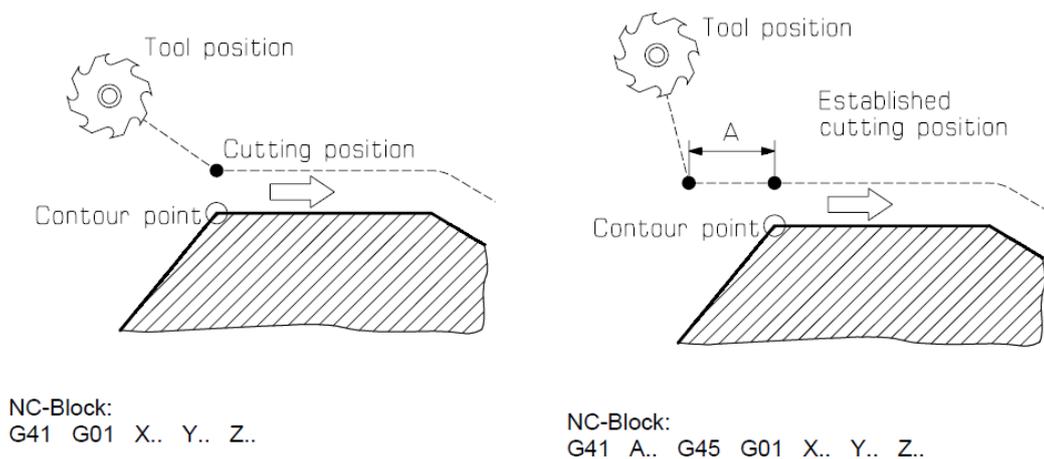
Untuk penentuan titik tujuan termodifikasi pada perpotongan garis lurus dengan kemiringan tertentu dengan lingkaran pada titik sembarang, merupakan kasus yang paling rumit. Gambaran skematik disajikan pada Gambar 3.14. Titik termodifikasi dapat dihitung dengan cara menentukan koordinat titik perpotongan antara garis lurus menuju titik A dan lingkaran baru dengan radius diperbesar sebesar Rpahat.



Gambar 3.14 Contoh perubahan posisi titik tujuan dibagian luar garis lingkaran.

Posisi titik tujuan termodifikasi ini akan dapat ditentukan oleh factor-factor antara lain: kwadrant tempat titik perpotongan itu terjadi, besar radius pahat yang digunakan dan sudut terkecil dari garis horizontal dan vertical yang bersinggungan dengan lingkaran pahat pada diameter terbesar.

Penggunaan kode G41 atau G42 sering kali dilengkapi dengan titik approaching pada jarak tertentu sebelum aksi kompensasi dilakukan. Gambar 3.15 menunjukkan contoh aksi kompensasi yang disiapkan untuk pemotongan pada pemotongan antara dua garis lurus.

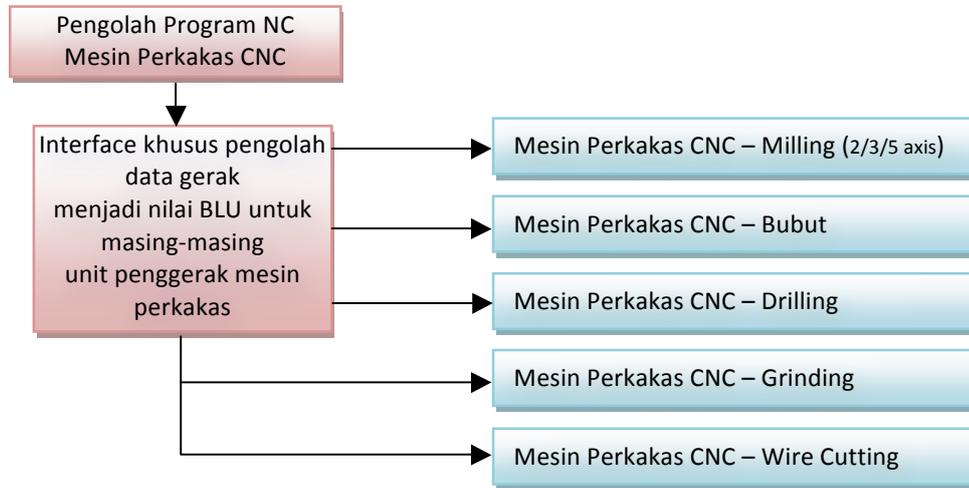


Gambar 3.15 Dua macam gerak awal pemotongan dengan kompensasi pahat.

3.8 Proposal Tampilan Perangkat Lunak Pemrograman NC

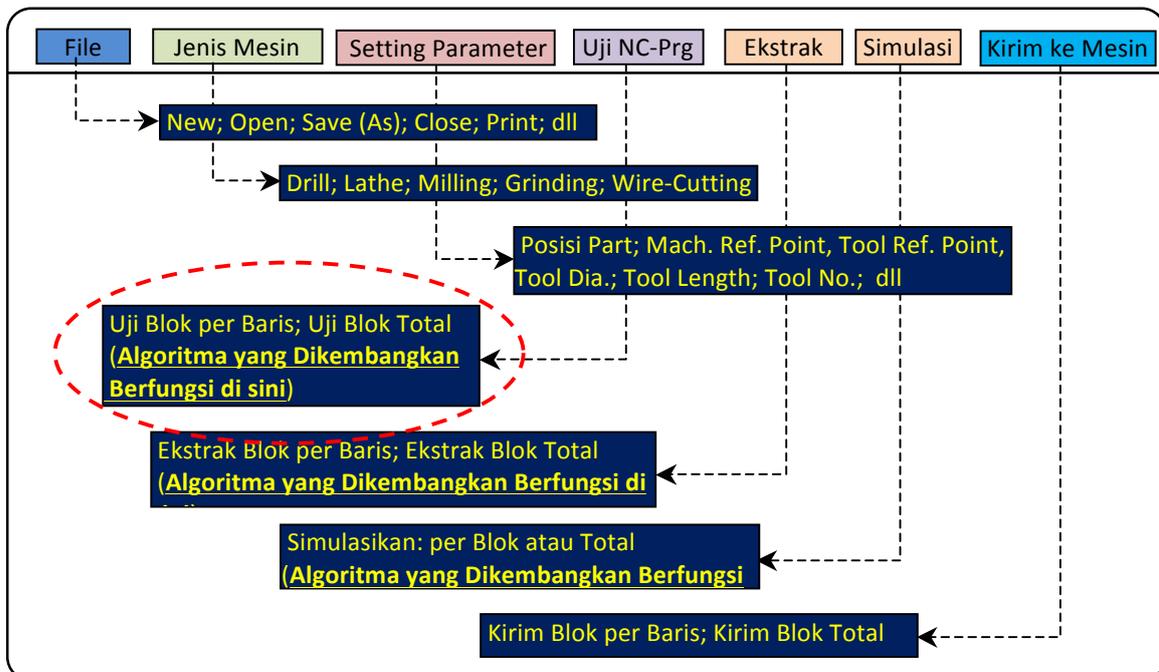
Untuk memudahkan pengguna berkomunikasi dengan perangkat keras/mesin maka perlu dibangun suatu antar muka (*interface*) yang bersifat mudah digunakan bagi pengguna. Fungsi-fungsi utama seperti adanya aksesibilitas terhadap file yang akan diuji,

pemilihan mesin perkakas yang akan digunakan, pengaturan-pengaturan yang terkait dengan parameter proses dan mesin, pengujian program NC dan juga perintah simulasi sebaiknya tersedia dalam antar muka ini seperti yang disajikan pada Gambar 3.15.



Gambar 3.15 Unit penghubung antara pengolah program NC dengan berbagai mesin perkakas berbasis CNC

Secara global tampilan perangkat lunak sebagai antar muka antara pengguna mesin/ pemrogram NC dapat dilihat pada Gambar 3.16. Fungsi-fungsi dasar yang diperlukan dikelompokkan dalam menu-menu yang bersesuaian dengan fungsi-fungsi khusus tersebut. Sesuai dengan batasan pada penelitian ini dimana pengembangan yang dilakukan difokuskan pada pengembangan algoritma, maka tampilan ini pada tahap pengembangan selanjutnya dapat dijadikan rujukan yang memadai.



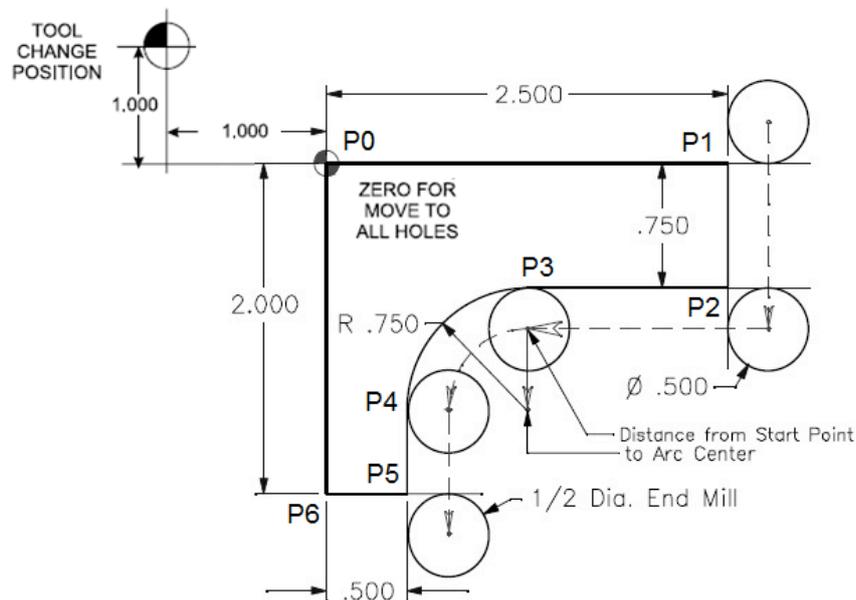
Gambar 3.16 Gambaran umum perangkat lunak yang dapat digunakan untuk menguji pengembangan algoritma

BAB IV

ANALISIS RANCANGAN ALGORITMA

4.1 Analisis Penentuan Titik Referensi Pahat

Sebagai langkah awal adalah penentuan titik referensi pahat relatif terhadap titik awal benda kerja. Hal ini dapat dipastikan keabsahannya dengan menganalisis satu contoh penentuan titik rujukan sebagai berikut. Tool change position ada berada pada jarak 1.000" di sebelah kiri, 1.000" di sebelah depan dan 1.000" di sebelah atas titik awal benda kerja.



Gambar 4.1 Penentuan titik posisi tool change relatif terhadap benda kerja.

Program NC yang dibuat untuk pemotongan pada Gambar 4.1 adalah sebagai berikut:

(Posisi awal pahat P0 pada jarak 1" di sebelah kiri, depan dan atas benda kerja, dan harus memotong dari bagian kanan menuju kanan bawah seperti pada gambar, dan selanjutnya kemudian kembali ke titik P0; Dia. pahat 0,5" dan harus memotong dengan kedalaman 0,4")

```
%  
O2000  
N00 G20 G91  
N10 M06 T03  
N20 S1500 M03  
N30 G92 X-1 Y+1 Z+1  
N40 G00 X2.75 Y0.25 Z0.1  
  
N50 G01 Z-0.5 F5  
  
N60 Y-1.25 F10
```

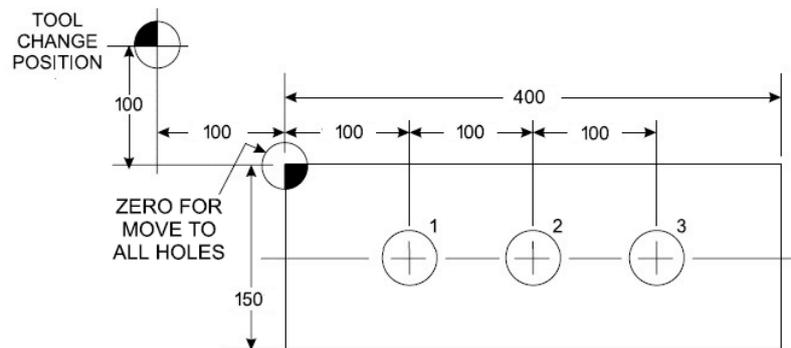
awalan program
nomor atau nama program (O2000)
input dimensi inch; mode dimensi inkremental
ganti pahat dgn pahat di no.3 (D03=0,5")
set spindle pada 1500 rpm dan arah CW
posisi pahat ada di kiri, depan atas titik nol
gerak cepat ke P1(2.0,0) dan pada 0,1" di atas permukaan benda kerja
gerak makan, memotong hingga -0,4" dibawah permukaan, kec. makan 5"/menit
gerak interpolasi, potong ke sb. -Y hingga P2 dengan kec. makan 10"/menit

N70 X-1.5	gerak interpolasi linier, potong ke sb. -X hingga titik P3
N80 G03 X-0.5 Y-0.5 R0.5	gerak interpolasi melingkar, hingga titik P4
N90 G01 Y-0.75	gerak potong ke sb. -Y hingga titik P5
N100 G00 Z0.5	gerak cepat hingga 0,1" di atas benda kerja
N110 G28	gerak cepat ke titik referensi di titik tool change position
N120 M05	matikan spindel
N139 M30	akhir dari program.

Pada contoh ini secara jelas dapat dilihat bahwa posisi titik rujukan (tool change position) dengan mudah dapat ditentukan dan titik tujuan selanjutnya harus didefinisikan dengan hati-hati dan teliti.

4.2 Analisis Fungsi Copying untuk Lubang Segaris

Setelah dua contoh operasi pemotongan dijabarkan di atas dan program NC yang disiapkan maka secara tahap demi tahap jalannya algoritma yang dirancang dapat dianalisis. Performa algoritma dapat dilihat dengan mengamati output yang dihasilkan oleh setiap fungsi dalam algoritma, yakni :



Gambar 4.2 Perhitungan koordinat titik-titik copying pada garis lurus

Bagian program NC pada Gambar 4.2 yang berfungsi untuk membuat 3-buah copy lubang akan tertulis sebagai berikut:

N05 G21 G90	input dimensi mm; mode dimensi absolut
N10 M06 T01	ganti pahat dgn pahat di no.1 (D01=15)
N15 S1200 F100 M03	set spindle 1200rpm, feed 100mm/s, arah CW
N20 G92 X-100 Y+100 Z+100	posisi pahat ada di kiri, depan atas titik nol
N25 G81 Z-15	penentuan drilling cycle, kedalaman 15mm
N30 G78 X100 Y-75 D+100 S3	lubang dimulai pada koordinat (100, -75) dengan jarak 100 sebanyak 3 lubang
N35 G79	cycle di-invoked oleh G79

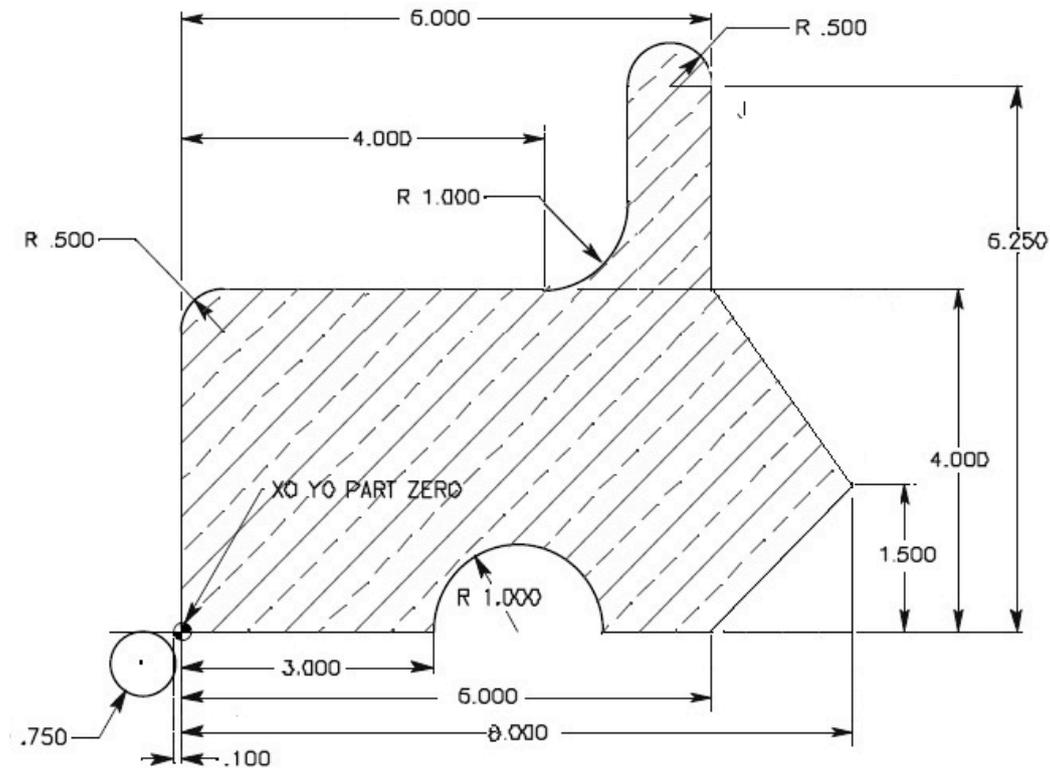
- o Blok pengolah copying cycle dimulai pada blok N25, menyiapkan drilling cycle yang merupakan canned cycle yang paling utama;
- o Selanjutnya pada baris N30, fungsi copying diawali dengan: jenis copying adalah pada garis lurus (G78); titik awal lubang adalah (100, -75) ; arah garis lurus

horizontal (A=0); jarak antar masing-masing lubang D=100; jumlah lubang yang harus dibuat S=3;

- o Program canned cycle ini di-invocated dengan kode G79;

4.3 Analisis Penggunaan G41 dan G42 pada Perpotongan Garis Sembarang

Penentuan titik tujuan pada pemotongan dua garis lurus tanpa memperhatikan ukuran radius pahat merupakan gerakan interpolasi garis lurus yang sederhana. Apabila CRC (cutter radius compensation) mulai digunakan, maka fungsi khusus untuk menangani titik-titik tujuan pahat harus diterapkan.



Gambar 4.3 Hasil pemotongan (kiri) dengan dan (kanan) tanpa kompensasi pahat membuat ukuran benda kerja hasil pemotongan banyak berubah

Bagian program NC pada Gambar 4.3 yang berfungsi untuk membuat 3-buah copy lubang akan tertulis sebagai berikut:

N05 G20 G90	input dimensi inch; mode dimensi absolut
N10 M06 T01	ganti pahat dgn pahat di no.1 (D01=0.75)
N15 G92 X0 Y0 Z+2.000	posisi pahat ada 2" atas titik nol
N20 S1000 F150 M03	set spindle 1000rpm, feed 150mm/s, arah CW
N25 G42 G00 X-0.750 Z0	titik approaching di kiri titik nol dengan G42
N30 G01 Z-1.000	gerak potong ke masuk sedalam 1"
N35 G01 X3.000	gerak potong ke kanan hingga x=3"
N40 G02 X5.000 R1.000	gerak melingkar setengah lingkaran
N40 G01 X6.000	gerak lurus hingga titik x=6"
N40 G01 X8.000 Y1.500	gerak miring hingga titik (8, 1.5) dst.

4.4 Arah Pengembangan Pada Tahap Selanjutnya

Seperti telah disajikan pada Gambar 1.2 bahwa masih banyak fungsi-fungsi pendukung pemrograman NC yang belum ditindak lanjuti walaupun pada penelitian ini sudah menambahkan fungsi-fungsi baru. Oleh sebab itu, dipandang perlu untuk membahas lebih mendalam beberapa fungsi berikut yaitu:

- 1) Penambahan berbagai canned cycle yang belum ditambahkan;
Fungsi-fungsi khusus untuk melakukan gerakan khusus dapat didefinisikan dalam canned cycle yang belum didefinisikan. Contoh fungsi khusus yang memerlukan canned cycle adalah pembuatan ulir dalam dan luar dengan gerakan pahat secara helikal, pembuatan lubang besar dengan gerakan helikal, dan lain-lain.
- 2) Pengolahan fungsi-fungsi bevel, taper, angle dan lain-lain;
Pada pemrograman NC, secara otomatis bidang interpolasi yang berlaku adalah G17 dimana interpolasi dilakukan pada bidang X-Y. Apabila diperlukan operasi pada bidang yang lain, maka perlu dinyatakan kode G18 atau G19.
- 3) Penentuan Fungsi Boring dan pembuatan lubang terkait lainnya;
Pada proses pembuatan lubang, terdapat pola lubang sejenis seperti counter boring, counter sinking, reverse counter sinking dan lain-lain. Urutan menekan kedalam bagi pahat saat membuat lubang, dapat ditentukan untuk keseragam kerja dan hal ini dilakukan dalam Canned cycle.
- 4) Penentuan Canned Cycles untuk pembuatan pocket, island dan bentuk-bentuk khusus;
Adakalanya beberapa fitur pemesinan harus dilakukan secara berulang-ulang. Apabila tidak ada fungsi khusus yang menangani ini maka program NC khusus harus dikembangkan menjadi bentuk fungsi standar yang dapat digunakan berkali-kali.

BAB V

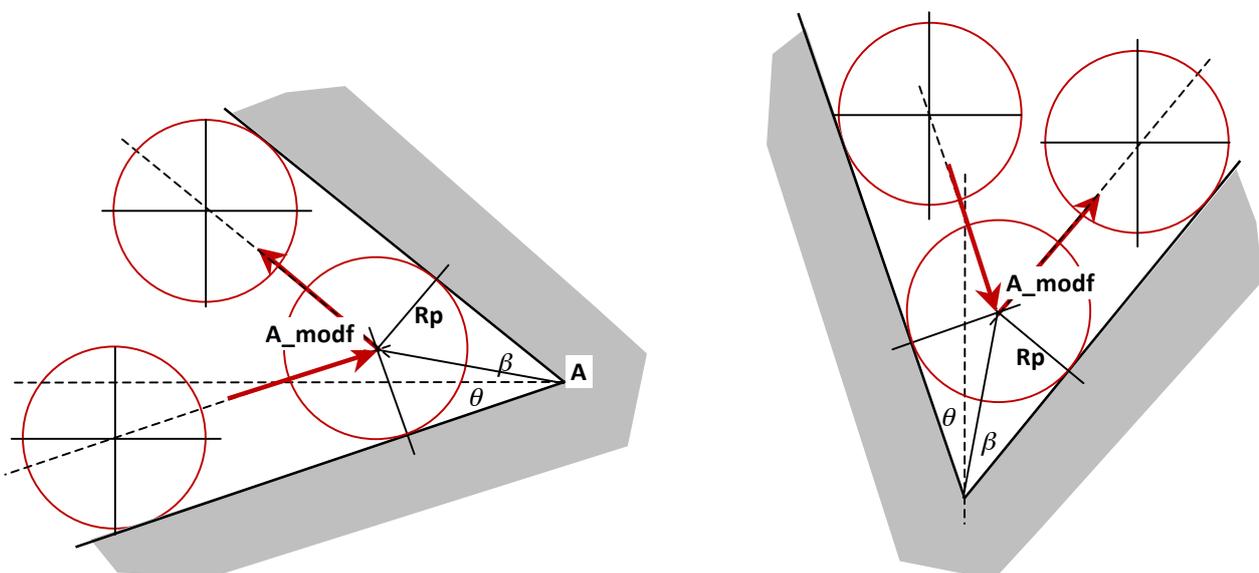
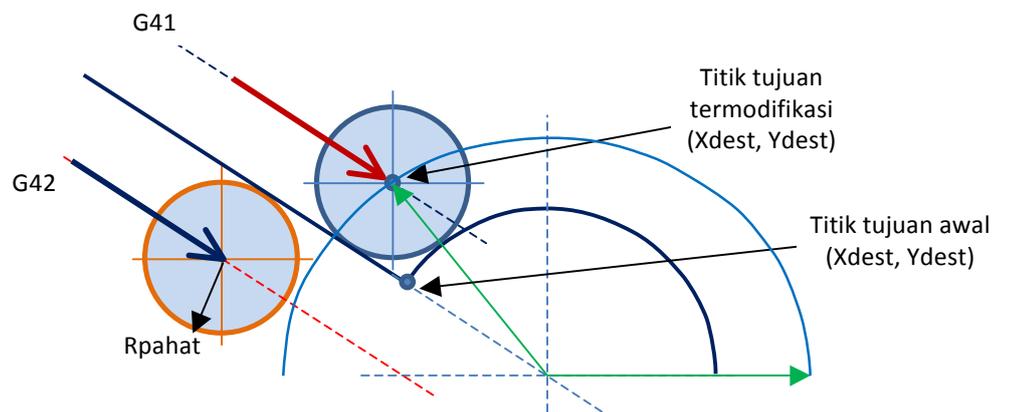
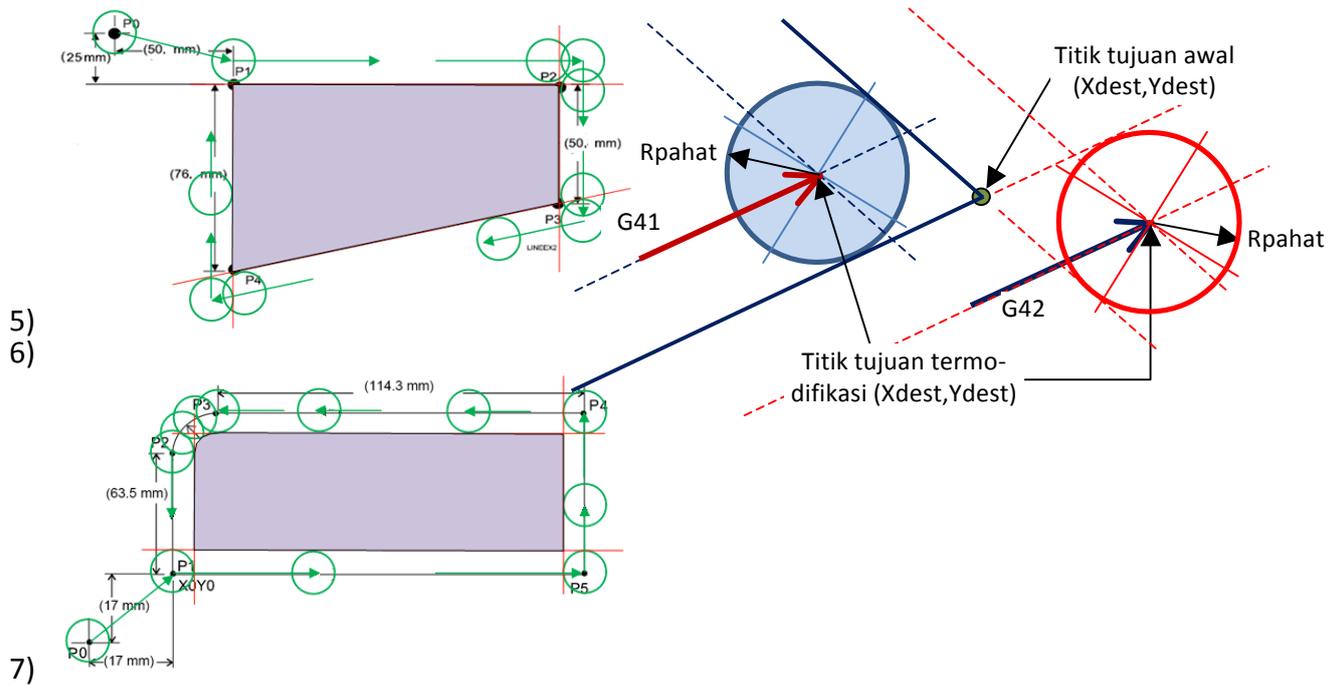
KESIMPULAN DAN SARAN

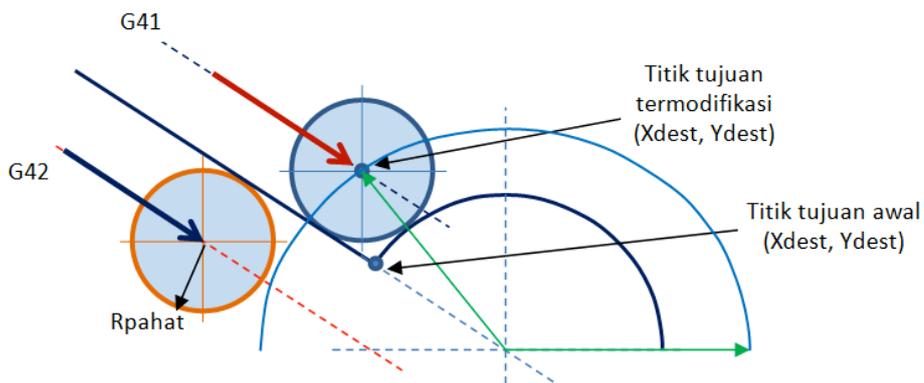
Dari kajian yang dijabarkan dalam penelitian ini, maka dapat ditarik beberapa kesimpulan mendasar, dan tentu masih membutuhkan kajian yang lebih mendalam. Berdasarkan rumusan masalah yang dijabarkan di atas, maka beberapa hal penting dapat dinyatakan sebagai berikut:

- Penentuan titik referensi merupakan langkah awal penting dalam pemrograman NC agar pahat dapat mengetahui jarak tempuh yang pasti dan akurat saat melakukan gerakan. Titik referensi mesin/pahat terhadap benda kerja dapat didefinisikan dengan kode G92. Selanjutnya nilai2 data koordinat yang dibawa oleh blok informasinya dijadikan nilai offset untuk menyesuaikan setiap titik tujuan yang dinyatakan pada blok-blok informasi yang akan datang.
- Titik awal benda kerja ditentukan pada sisi-sisi yang dapat dijadikan titik acuan. Misalnya bidang hasil pemesinan yang berpotongan tegak lurus dapat dipilih menjadi bidang acuan dimana semua ukuran dimensi akan merujuk pada bidang ini. Titik awal benda kerja ini dengan jelas disajikan pada gambar teknik yang menyertai benda kerja yang akan dimesin.
- Penentuan fungsi-fungsi khusus yang dapat disebut sebagai canned cycle dipilah berdasarkan karakteristik alamiah proses pemotongan yang dimiliki. Ada beberapa fungsi canned cycle yang dikembangkan yakni:
 - Fungsi dwell dilakukan untuk pembuatan lubang, dengan cara memberikan waktu khusus bagi pahat untuk diam sesaat pada ketinggian tertentu saat membuat lubang. Hal ini bertujuan untuk memastikan patahnya geram (chip) dan juga ditambahkan fungsi untuk mengeluarkan geram dari area pemotongan. Kode G yang terkait adalah G81, G82 dan G83.
 - Fungsi untuk melakukan gerakan berulang misalnya mengcopy lubang pada suatu lintasan lurus maupun sepotong busur dilakukan dengan mendefinisikan titik lubang pertama, jarak antar lubang, arah lubang dan jumlah lubang. Kode G yang terkait adalah G77, dan G78.
- Fungsi kompensasi radius pahat merupakan fungsi yang sangat penting agar programmer dapat dengan mudah melakukan tugasnya, maka fungsi ini harus dapat digunakan pada kondisi pemotongan apapun. Dalam penelitian ini ditambahkan fungsi kompensasi pahat untuk perpotongan dua garis lurus dengan arah sembarang sehingga titik-titik tujuan awal harus dimodifikasi sesuai dengan gradien garis dan ukuran radius pahat. Perhitungan titik tujuan termodifikasi itu dihitung dengan perhitungan goneometri sederhana seperti yang telah dijabarkan sebelumnya. Fungsi kompensasi pahat ini harus dapat digunakan pada setiap fitur pemotongan yang terkait langsung dengan dimensi akhir benda kerja yang akan dibuat

DAFTAR PUSTAKA

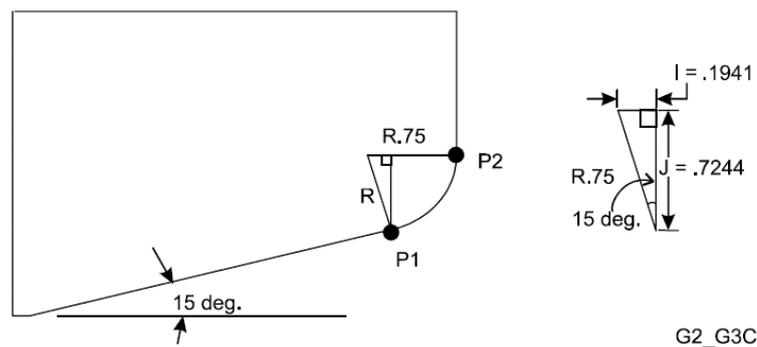
1. _____, (2010). CNC Program, Daewoo Haeavy Industries Ltd, ep2.profweb.qc.ca/.../Daewoo_programming_manual.pdf
2. _____, (2006). Programming Workbook - Milling, Haas Automation, Inc. 2800 Sturgis Rd. Oxnard, Ca 93030, www.HaasCNC.com
3. _____, (2006). Programming Workbook - Turning, Haas Automation, Inc. 2800 Sturgis Rd. Oxnard, Ca 93030, www.HaasCNC.com
4. _____, (2006). 5000M CNC Programming and Operations Manual, ACU-RITE Companies, Inc., www.anilam.com
5. _____, (1995). MTS: Mathematisch Technische Software-Entwicklung GmbH, Berlin, www.mts-cnc.com
6. Arthaya B, (2010). Pengembangan Perangkat Lunak G-Code Untuk Penyiapan Perintah Gerak Mesin Cnc (Pokok Bahasan Algoritma Uji, Ekstraksi, Formulasi Lintasan), Laporan Penelitian LPPM, Unpar.
7. Krar S., and Gill A., (1999). Computer Numerical Control Programming Basics, *Industrial Press, Inc.*, New York.
8. Rochim, T. (2000). Pemrograman NC. *Institut Teknologi Bandung*, Bandung.
9. Singh, R., (2006). Introduction to Manufacturing Processing and Workshop Technology, *New Age International (P) Ltd, Publishers*, New Delhi.
10. Stenerson, J., Curran, K. (1997). Computer Numerical Control. *Prentice Hall Inc.* New Jersey.
11. Koren Y., (1983). Computer Control of Manufacturing Systems, McGraw-Hill Int. Book Co, Auckland





Penentuan Kompensasi Pahat pada Kombinasi Lintasan;

Pada algoritma yang telah dikembangkan dalam penelitian ini, kompensasi radius pahat hanya diterapkan pada lintasan garis lurus saja. Sementara pada lintasan melingkar kompensasi ini belum diterapkan. Khususnya pada kasus garis miring yang berlanjut dengan radius lingkaran memerlukan perhitungan geometri yang lebih teliti. Kemiringan garis lurus akan secara langsung menentukan sampai dimana lintasan pahat harus dikerjakan, sehingga segmen lintasan yang satu bisa bertemu dengan sempurna dengan segmen yang lain.



Gambar 4.9 Kompensasi radius pahat memerlukan geometri detail benda kerja (Acu-rite,2006)

