© Universiti Tun Hussein Onn Malaysia Publisher's Office

IJIE

Journal homepage: http://penerbit.uthm.edu.my/ojs/index.php/ijie
ISSN : 2229-838X e-ISSN : 2600-7916

The International Journal of Integrated Engineering

Development of the PC-based Integrated Interface System of STEP file for CNC Machining Application: Circular Features

Muhammad Abdulrahim Rabbani Md Sharizam[1], Saiful Bahri Mohamed[*1], Tengku Mohd Shahrir Tengku Sulaiman[1], Zammeri Abd Rahman[1], Roslan Awang[1]

[1] Faculty of Innovative Design & Technology,
Universiti Sultan Zainal Abidin, Kuala Terengganu, MALAYSIA

*Corresponding author

DOI: https://doi.org/10.30880/ijie.2020.12.07.022
Received 3 September 2020; Accepted 12 October 2020; Available online 31 October 2020

Abstract: STEP is a general data format that observes the international standard ISO 10303-21. STEP means Standard for the Exchange of Product model data. It consists of the 3D geometry of a computer-aided design (CAD) model in the configuration of boundary representation (B-rep). By extracting, refining and decoding the geometric data correctly, the data can be utilized for writing G-code for Computer Numerical Control (CNC) machining application. Usually G-codes can either be manually generated by skilled machinists or automatically generated by computer-aided manufacturing (CAM) software. However, manually generated G-code is inefficient and susceptible to error. Meanwhile automated generation G-code requires significant setup cost. This paper describes the design and development of an integrated interface system, an instrument aimed to be used to analyze STEP files and generate machining tool path based on ISO 6983 format. This developed interface reduces the need for high setup cost as well as eliminates human limitations. The interface at present is able of detecting circular machining features on the workpiece. Circular machining features are created by 3D modelling software and retained as STEP file. The STEP file which contains geometrical data is then uploaded to the interface system as an input file which is structurally analyzed and processed. Finally, the ideal machining tool path in the G Code format is proposed and generated. By bypassing the CAM software and its proprietary post processor, the outcome of this research is important to enhance compatibility between different CNC machine systems

Keywords: STEP, B-rep, CAD model, CNC, G-code, interface

1. Introduction

Computer numerical control (CNC) machine is a machine that works on dedicated software to power the automatic machining. CNC machine is a vital link for the manufacturing automation. Its speed and precision make it ideal for mass production compared to conventional machining [1]. CNC machines employ specific programming language known as G-code. G-code tells a computerized machine tool how to move. The instructions in G-code contains cutter target location, speed, and the path to follow [2]. The type of G-code used in this research is ISO 6983 which is an open source international standard. G-code is useful to synchronize system design, to help encourage consistency of geometric programming systems, and to enhance interoperability of input systems.

G-code can be generated manually or automatically. For manual operation, a human operator will analyze the 3D CAD model, interpret the design, then write G-code according to the features on the CAD model. For automatic operation, CNC machine feeds on either G-code or STEP-NC files to operate. These files are usually generated by the add-on CAM software [3].

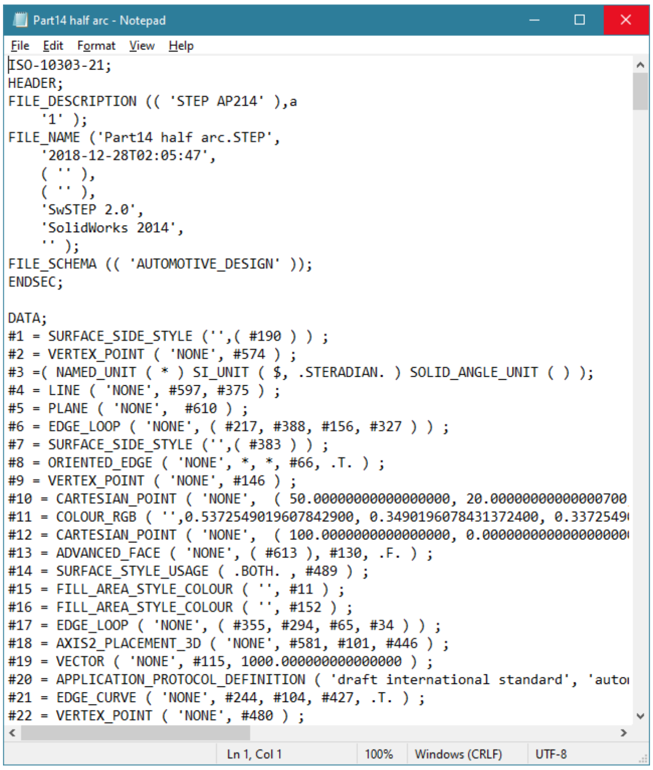
**Corresponding author saifulbahri@unisza.edu.my*

Many researches have been done on automatic feature recognition of STEP file, but prior to this research, a practical system that involves recognizing circular feature in STEP file and translating it into G-code has not really taken off. The integrated interface system aims to fill the void.

2. Step File

STEP, which is an acronym for Standard for the Exchange of Product model data, is a general geometric data structure between CAD design and CNC machine. STEP, which is a high-level programming language, encodes the data of geometric entities of a CAD design. It is indexed under the ISO 10303-21, which is Part 21 of ISO 10303. Part 21 is the implementation of STEP data standard in ASCII format for easy readability [4]. As shown in Fig. 1, STEP contains data of geometric entities of a CAD design in the configuration of boundary representation (B-rep). All data of geometric entities obtained from CAD/CAM is passed to CNC controller without missing a single entity [5]. Modern machine controllers are able to analyze and extract geometric data from STEP files to construct and simulate machining tool paths [6]. STEP file does not encode any data about machining tool path, but owing to the nature of it being open sourced, it is used to analyze data of geometric entities.

Relatively for CNC machining, STEP is still new. Its development is important to create a standard flow of data of geometric entities from CAD design to CNC machining [7]. Around the world, STEP is still being continually improved under various research and development. It is a perfectly practical data format in encoding data of geometric entities of a CAD design. As development of STEP accelerates, it is anticipated to become more flexible and able to encode more types of geometric entities.


Part14 half arc - Notepad
File Edit Format View Help

```
ISO-10303-21;
HEADER;
FILE_DESCRIPTION (( 'STEP AP214' ),a
    '1' );
FILE_NAME ('Part14 half arc.STEP',
    '2018-12-28T02:05:47',
    ( '' ),
    ( '' ),
    'SwSTEP 2.0',
    'SolidWorks 2014',
    '' );
FILE_SCHEMA (( 'AUTOMOTIVE_DESIGN' ));
ENDSEC;

DATA;
#1 = SURFACE_SIDE_STYLE ('',( #190 ) ) ;
#2 = VERTEX_POINT ( 'NONE', #574 ) ;
#3 =( NAMED_UNIT ( * ) SI_UNIT ( $, .STERADIAN. ) SOLID_ANGLE_UNIT ( ) );
#4 = LINE ( 'NONE', #597, #375 ) ;
#5 = PLANE ( 'NONE',  #610 ) ;
#6 = EDGE_LOOP ( 'NONE', ( #217, #388, #156, #327 ) ) ;
#7 = SURFACE_SIDE_STYLE ('',( #383 ) ) ;
#8 = ORIENTED_EDGE ( 'NONE', *, *, #66, .T. ) ;
#9 = VERTEX_POINT ( 'NONE', #146 ) ;
#10 = CARTESIAN_POINT ( 'NONE',  ( 50.00000000000000000, 20.00000000000000700
#11 = COLOUR_RGB ( '',0.5372549019607842900, 0.3490196078431372400, 0.3372549
#12 = CARTESIAN_POINT ( 'NONE',  ( 100.0000000000000000, 0.000000000000000000
#13 = ADVANCED_FACE ( 'NONE', ( #613 ), #130, .F. ) ;
#14 = SURFACE_STYLE_USAGE ( .BOTH. , #489 ) ;
#15 = FILL_AREA_STYLE_COLOUR ( '', #11 ) ;
#16 = FILL_AREA_STYLE_COLOUR ( '', #152 ) ;
#17 = EDGE_LOOP ( 'NONE', ( #355, #294, #65, #34 ) ) ;
#18 = AXIS2_PLACEMENT_3D ( 'NONE', #581, #101, #446 ) ;
#19 = VECTOR ( 'NONE', #115, 1000.000000000000000 ) ;
#20 = APPLICATION_PROTOCOL_DEFINITION ( 'draft international standard', 'auto
#21 = EDGE_CURVE ( 'NONE', #244, #104, #427, .T. ) ;
#22 = VERTEX_POINT ( 'NONE', #480 ) ;
```

Ln 1, Col 1 | 100% | Windows (CRLF) | UTF-8

Fig. 1 - A STEP file opened in Notepad

3. Boundary Representation

In STEP files, data of geometric entities are written in the configuration of B-rep model. B-rep model explains the 3D geometry as a set of enclosed surfaces. It describes objects using geometric limits. A CLOSED_SHELL (an enclosed solid object) consists of a set of linked surface entities forming a 3D object or solid [8]. B-rep model encodes more geometric information about a 3D object in comparison to the constructive solid geometry (CSG) model. Along with the model, there are also important geometric data like chamfer, fillet and extrusion. But B-rep model does not include the history of geometric construction, thus in this regard, B-rep falls behind CSG [9].

In any B-Rep model, there exist two sets of information: first is the geometric data for describing the coordinates, positions in 3D space and curves; second is the topological data for connecting multiple geometrical entities. Likewise, in any B-rep model, there also exist two sets of entity: the geometric entities that covers surfaces, points and curves; and the topological entities that covers faces, vertices and edges. A geometric entity is bound by a set of topological entities. [10]. As shown in Fig.2, a face is enclosed by a set of surfaces; an edge is enclosed by sets of lines and or curves; and a vertex is located somewhere on a spatial coordinate [11].

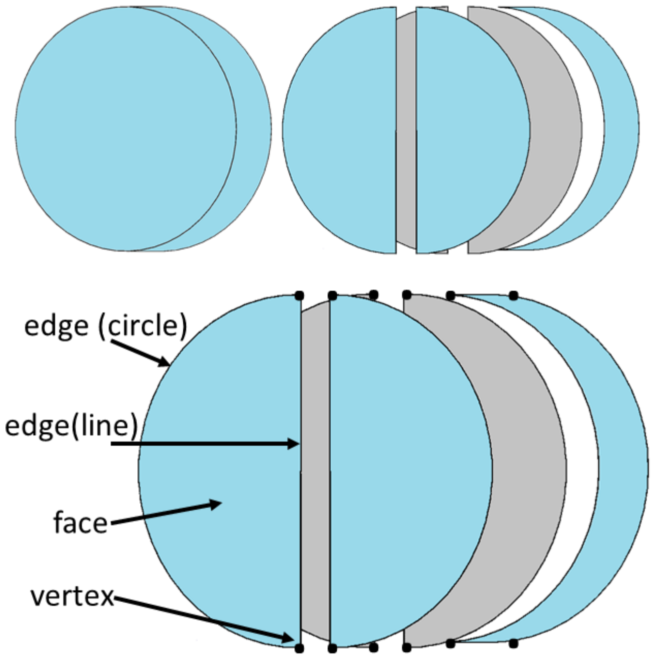


Fig. 2 - How a solid shape is represented according to B-rep model

4. Step Data Structure

The hierarchy of geometric entities in STEP file is shown in Fig. 3 which begins with a solid object. A CAD design representing a 3D solid shape is encoded as a CLOSED_SHELL. CLOSED_SHELL is enclosed by ADVANCED_FACEs. Every ADVANCED_FACE contains the geometric information of the FACE_BOUNDs, FACE_OUTER_BOUNDs and the type of surface.

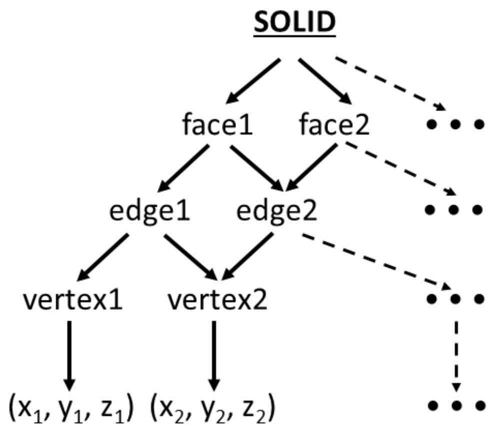


Fig. 3 - Hierarchy of geometric entities a STEP file according to B-rep model

A FACE_OUTER_BOUND is a loop of edges that encloses the outer limit of a surface, meanwhile a FACE_BOUND is a loop of edges bound internally that covers a pocket or hole of a model.

The face type data tells whether a surface is a plane, cone, toroid, cylinder or sphere. [12]. The EDGE_LOOP is a complete set of edges enclosing the loop, and the ORIENTED_EDGE meanwhile contains data that describe the edge curves. The EDGE_CURVE holds the geometric information about the vertices and their respective spatial coordinates that encloses the edge.

The geometric data also consists of information like lines and circles that describe the type of EDGE_CURVE [13].

The lines that form the EDGE_CURVE are described as multiple sets of two cartesian points and a vector indicating the direction. Meanwhile the circles that form the EDGE_CURVE are described as multiple sets of two cartesian points, a vector indicating the direction, radius and center point of the circles.

5. Integrated Interface System

Integrated interface system is a computer program that works by translating the design in STEP files into G-code for CNC machine. Construction of this system is done using PHP language. It offers wide range of compatibility since it can be accessed on local server on internet browser. Additional user inputs are required which include feed rate, direction of rotation, spindle speed and depth of cut. These inputs are required to generate machining tool path, and eventually G-code.

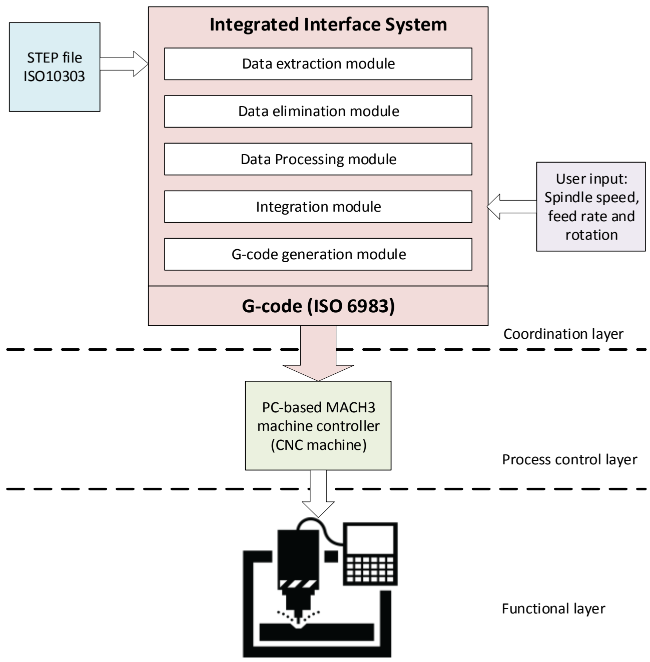


Fig. 4 - The modules in the integrated interface system

Integrated Interface System processes data on five different modules as in Fig. 4. The first stage is extracting data, then eliminating, processing, integrating and finally generating G-code. Module of data extraction will examine STEP files lines by lines to look for geometric entities. The module then copies the data into a database of circles. Data elimination module works by examining repeating data which are not pertinent to the pocket to be machined on the workpiece, then remove all of them from the circle database. The data is then brought to processing module to be analyzed and processed according to the specific set of instructions. Integrating data is done under a module which processes feed rate, direction of rotation, spindle speed and depth of cut keyed-in by the user. The final module writes G-code after analyzing and processing the STEP file and user input.

6. Data Processing

A CAD model, Part21_half_arc. SLDPRT is created using SolidWorks as shown in Fig. 5. The design consists of pocket to be removed on a single aluminum block to using CNC machine. The CAD design is then saved as STEP file, Part21_half_arc. STEP. The STEP file is then uploaded to the interface on internet browser.

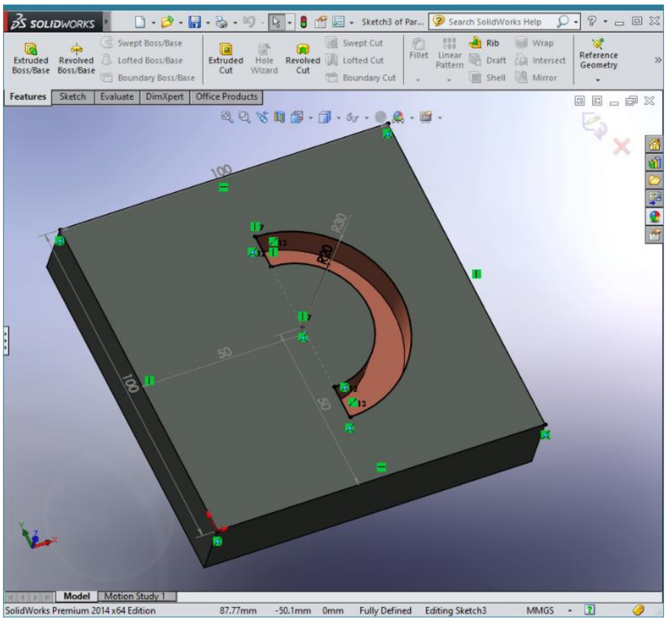


Fig. 5 - A CAD model, Part21_half_arc.SLDPRT

The first step the interface does is recording user inputs which are spindle speed, depth of cut, direction of rotation and feed rate. Then it scans the whole STEP file for circular feature. As shown in the example highlighted in turquoise, circular feature can be traced by the word “CIRCLE”. The interface then creates a database for every circle and each of their related features.

```
#87 = CIRCLE ( 'NONE', #592, 20.000 );
```

The feature of the circle can be traced by its reference number to find its center point and radius. As shown in the example, the reference number #592 is highlighted in grey, and it always begin with “#” mark. Meanwhile, the geometric entities are highlighted in turquoise, and usually indicated after the “=” mark. Searching for line #592 returns to another reference number #529. The line #529 indicates a cartesian point with three numbers that represents the spatial coordinates of the circle’s center point. The center points and radii data are subsequently recorded in the database.

```
#87 = CIRCLE ( 'NONE', #592, 20.000 ) ;
#592 = AXIS2_PLACEMENT_3D ( 'NONE', #529,
   #157, #254 ) ;
#529 = CARTESIAN_POINT ( 'NONE',  ( 50.000, 0.000,
   -10.000 ) ) ;
```

However, not all circles in STEP file are related to the machining feature. The redundant circles, if not filtered, will create confusion in the interface system and eventually lead to generating invalid G-code. By having the data about the center point of each circle, the circles that lie on top of the surface of the workpiece will be eliminated. This is because there will be another circle below it the with the same feature, center point and curve but at different height.

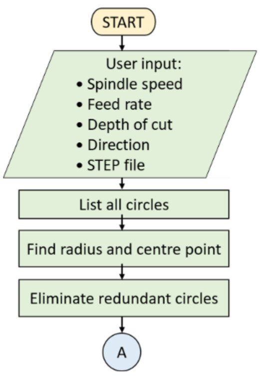


Fig. 6 - Data extraction and elimination

Then, the start and end point of each circle are recorded by tracing it through the EDGE_CURVE and EDGE_LOOP. EDGE_CURVE #556 contains the reference numbers #314 and #576. These reference numbers subsequently lead to the start point and end point of the circle. By having the start point, end point, radius and center point of each circle, the database is complete and can be sent to processing module to determine the ideal machining tool path. Summary of the data extraction module is shown in Fig. 6.

```
#370 = CIRCLE ( 'NONE', #317, 30.000 ) ;
#317 = AXIS2_PLACEMENT_3D ( 'NONE', #537,
    #302, #99 ) ;
#537 =CARTESIAN_POINT ( 'NONE',  ( 50.000,
    50.000, 0.000 ) ) ;
#556 = EDGE_CURVE ( 'NONE', #314, #576, #370, .T.
    ) ;
#314 = VERTEX_POINT ( 'NONE', #594 ) ;
#594 = CARTESIAN_POINT ( 'NONE',  ( 80.000,
    50.000, 0.000 ) ) ;
#576 = VERTEX_POINT ( 'NONE', #311 ) ;
#311 = CARTESIAN_POINT ( 'NONE',  ( 50.000,
    80.000, 0.000 ) ) ;
```

Then, the processed data proceed to the data processing module as explained in Fig. 7. Depending on the user preference of direction of tool path, whether clockwise and counter-clockwise, the interface eliminates the redundant circles. If the user chose clockwise direction, then the interface will eliminate larger circles. But if the user chose counter-clockwise direction, then the interface will eliminate smaller circle. This is because for every cylindrical surface, there is an opposite cylindrical surface that forms a pocket on the workpiece. From the remaining circles, the tool path is generated.

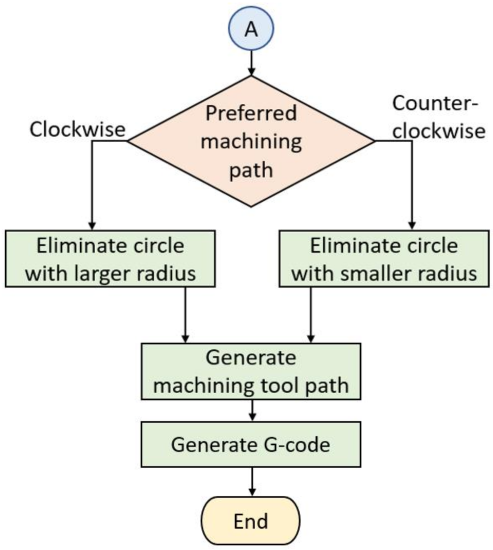


Fig. 7 - Data processing module of the interface

7. Interface Development

The integrated interface system is developed using PHP language. PHP is chosen because it can be accessed on local server on internet browser. This expands its compatibility in which it can be accessed on any system that has web browser.

When accessing the interface, user will be greeted with blank spaces to key in input like spindle speed, feed rate, direction of rotation and depth of cut. User also need to upload a STEP file containing the CAD design and saved in text document format (.TXT).

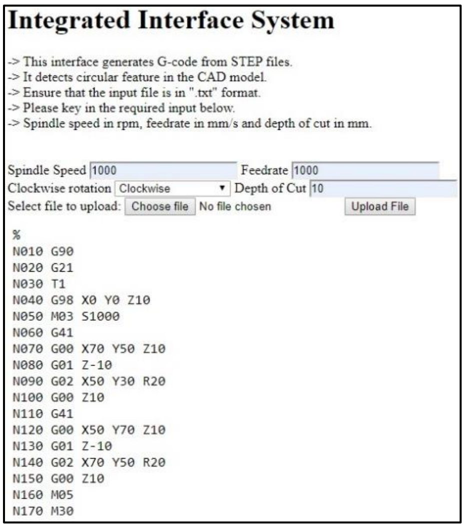


Fig. 8 - G-code generated by the interface

The whole STEP file is scanned lines by lines, then a tool path is generated. Then as shown in Fig. 8, G-code will be generated based on the tool path.\

8. G-Code Validation

The G-code generated by the interface is tested through CNC simulation on a computer as in Fig. 9. The controller for PC-based simulation used is Mach3 machine controller. Mach3 is selected because of its public availability of its source code. Mach3 does not rely on certain machine vendors and their proprietary or closed source software to operate the CNC machine. The software in case, CNC Simulator Pro generates machining tool path by having G-code generated by the interface as an input. The tool path generated will be put into comparison with the pocket on the CAD design.

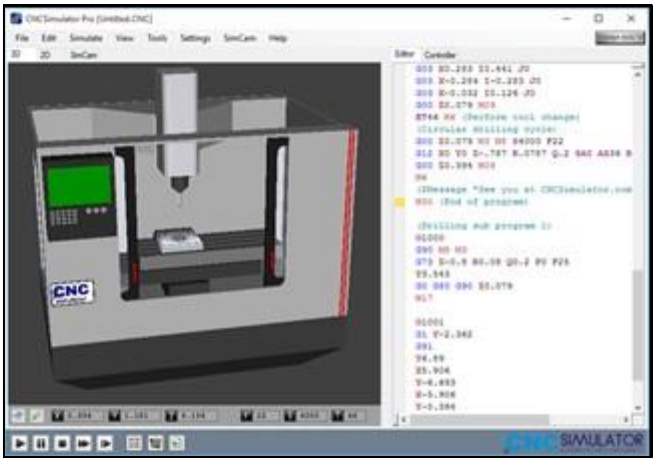


Fig. 9 - Tool path simulation on a PC-based CNC controller

9. Result and Discussion

CNC Simulator Pro generates tool path according to the G-code written by the interface. Fig. 11 shows two tool paths generated by the interface. The tool path in Fig. 11 (a) covers half of the machining feature, while Fig. 11 (b) completes the other half. The tool path shown by the software is excellently aligned with the pocket on the CAD model in Fig. 10. One important thing to note is that the interface system does not split the tool path into two, instead it is the CAD software that defines the geometry when generating STEP file. The interface just works on available geometric information encoded by the CAD software. The interface lives up to its intended function. It is able to accurately write G-code that matches the tool path needed to machine workpiece.

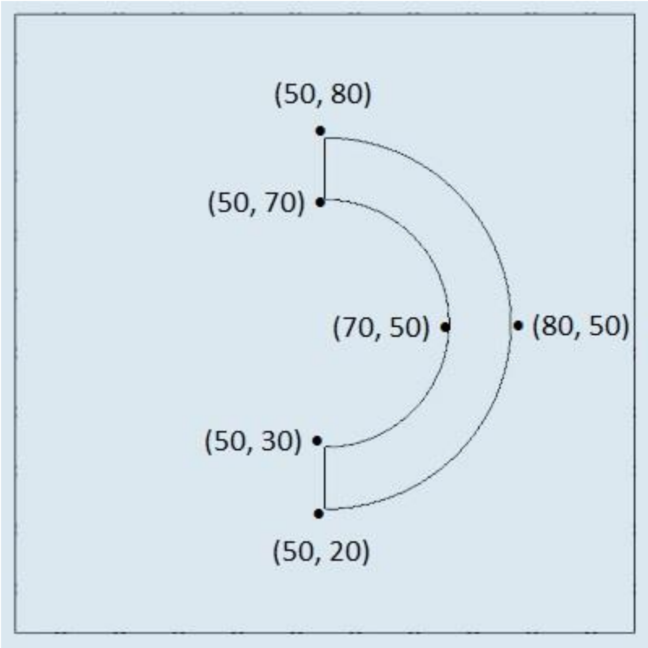
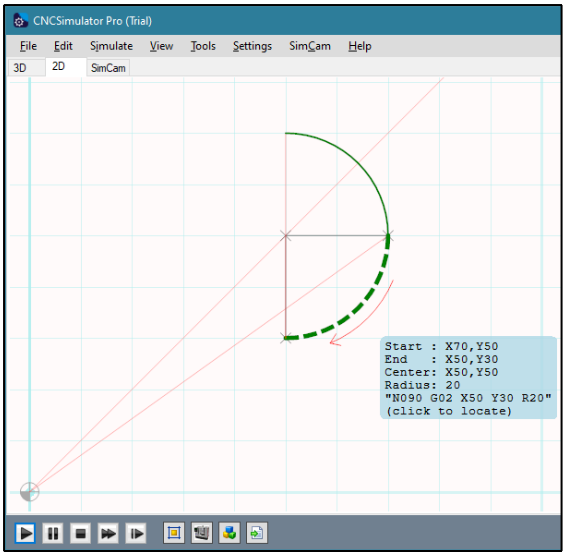
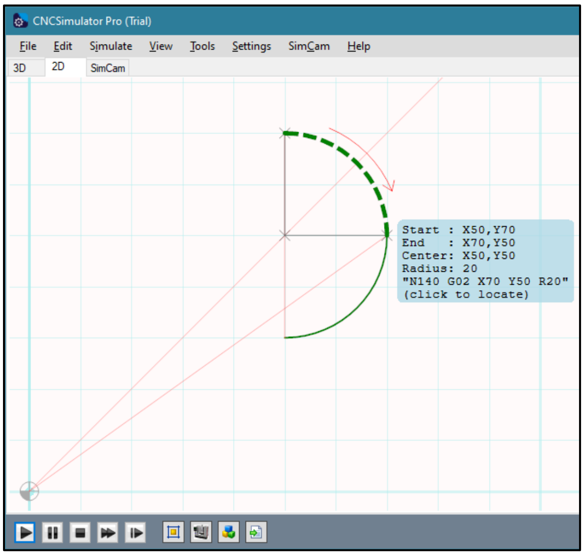


Fig. 10 - The feature on the CAD design, Part21_half_arc.SLDRT



(a)



(b)

Fig. 11 - Both tool paths generated by the CNC Simulator Pro suits the pocket required to be machined on the CAD design

G-code is produced by the interface according to the circles and their associated features in the STEP files. Interface chooses either the end point or start point as the end point of arc in G03 or G04 instruction for circular interpolation. Integrated interface system can function as a link in machining automation. It extracts data of geometric entities in the STEP file, relay that information, and processed it into G-code. The role of CAM software in generating G-code can be minimized with the advent interface system. This minimizes the need for closed source software and enhance flexibility as the interface can be accessed from local server on internet browser.

10. Conclusion

The benefit of this research is that generating G-code will no longer depend solely on CAM software, which relies on proprietary or closed source software to analyze the CAD model and generate G-code. On top of that, it removes the need to invest on CAM software and its significant setup cost. The system also bypasses the proprietary post processor required to run the CNC machine. This means enhanced compatibility between different CNC machine systems since a single type of file can be used as an input to run different model of CNC machines. The integrated interface system presents an alternative to G-code generation by providing speed and accuracy while enhancing compatibility across different systems.

The system motivates novices to learn CNC machining; hence expanding the talent pool. The knowledge about the encoding of geometric entities gained from this research will improve our understanding of the data of geometric entities in STEP files. For now, the interface is only capable to generate circular tool path. The future research can be expanded to include detection for planes, cones, spheres and toroid. When the scope of feature detection scope is widened beyond circular feature, the integrated interface system will become a practical alternative to CAM software in generating G-code.

Acknowledgment

The authors convey the deepest appreciation to Universiti Sultan Zainal Abidin and especially to the Ministry of Higher Education Malaysia for granting the funding to this research under the Fundamental Research Grant Scheme (Grant No. FRGS/1/2018/TK03/UNISZA/02/1).

References

[1] S.S. Yeh, Z.H. Tsai, P.L. Hsu, "Applications of integrated motion controllers for precise CNC machines" in The International Journal of Advanced Manufacturing Technology, vol. 44, Article 906, 2009

[2] S.S. Suh, S.K. Kang, D.H. Chung, I. Stroud (2008) Theory and Design of CNC Systems. Springer Science & Business Media

[3] N. Zeroudi, M. Fontaine, K. Necib, "Prediction of cutting forces in 3-axes milling of sculptured surfaces directly from CAM tool path" in Journal of Intelligent Manufacturing, vol. 23, 2012, p. 1573-1587

[4] M. Minhat, V. Vyatkin, X. Xu, S. Wong, & Z. Al-Bayan, "A novel open CNC architecture based on STEP-NC data model and IEC 61499 function blocks" in Robotics and Computer-Integrated Manufacturing, vol. 25, Issue 3, Jun. 2009, p. 560-569

[5] S.B. Mohamed, M. Minhat, T.M.S.T. Sulaiman, M.S.M. Kasim, A.S. Mohamed, "Integrated interface system for tool path generation of STEP file" in AIP Conference Proceedings 2129 (1), 020012, Jul. 2019

[6] S.B. Mohamed, A. Jameel, M. Minhat, "A review on intelligence STEP-NC data model and function blocks CNC machining protocol" in Advanced Materials Research 845, Dec. 2013, p. 779-785

[7] J. Kim, M.J. Pratt, R.G. Iyer, R.D. Sriram, "Standardized data exchange of CAD models with design intent" in Computer-Aided Design, vol.40, Issue 7, Jul. 2008, p. 760-777

[8] D. Seeramulu, C.S. Rao, "A New Methodology for Recognition of Milling Features from STEP File" in 3rd International Journal of Applied Management and Technology; vol. 6, Issue. 3, 2008, p. 172

[9] F. Boussuge, J.C. Leon, S. Hahmann, L. Fine, "Extraction of generative processes from B-Rep shapes and application to idealization transformations" in 3rd International Conference on Computer Science and Information Technology, Jul. 2010

[10] C. Ellul, M.M. Haklay, "Using a B-Rep Structure to Query 9-Intersection Topological Relationships in 3D GIS – Reviewing the Approach and Improving Performance" in 3D Geo-Information Sciences, 2009, p. 127-151

[11] A. Bormann, S. Schraufstetter, E. Rank, "Implementing Metric Operators of a Spatial Query Language for 3D Building Models: Octree and B-Rep Approaches" in Journal of Computing in Civil Engineering, vol. 23, Issue 1, Jan. 2009

[12] M.P. Bhandarkar, B. Downie, M. Hardwick, R. Nagi, "Migrating from IGES to STEP: one to one translation of IGES drawing to STEP drafting data" in Computers in Industry, vol. 41, Issue 3, May 2000, pp. 261-277

[13] I.H. Song, S.C. Chung, "Web-based CAD viewer with dimensional verification capability through the STEP translation server" in Journal of Mechanical Science and Technology, vol. 21, Aug. 2007, pp. 1235–1243