Designing the Application of a Distributed Control System and Creating a Portable Mechanism with Pc-Based Numerical Control (Cnc)

Aliasghar Tabatabaei [1], Rana Hajıyeva[2], and Huseyn Gurbanov[2]

[1]Melorin Biotech LTD, London, UK

[2]Department of İnformation Technologies, Western Caspian University, Baku, Azerbaijan

Received: 12.05.2026

Accepted: 26.05.2026

Published: 29.06.2026

https://doi.org/10.54414/FZUM8064

Abstract: The trends in rapid production (RP) have largely impacted the development of technology in additional processes. Nevertheless, material incompatibility and accuracy issues with the additional processes have restricted the possibility of fabricating end-user products. The additional processes, including the more radical Computer Digital Control (CDC) machining, can in some cases be applied to RM. Using a 3-axis CDC milling machine equipped with an indexing device allows for increasing the accessibility of the cutting tools and avoiding numerous process restrictions. At the same time, more research should be conducted to advance the use of CDC in RM, which is the purpose of this thesis. This research aims to refine the process of deep machining and finishing operations and the application of cutting tools in CDC machining in order to make CDC machining applicable for RM. It is expected that the implementation of the proposed techniques will accelerate production and increase the quality of the parts produced.

Keywords: Distributed Control System (DCS), PC-Based CNC, G-code, Toolpath Optimization, Computer-Aided Manufacturing (CAM)

1.Introduction

Another way to increase the effectiveness of deep processing operations may include a change in the direction of cutting. Simulation analyses are conducted to change the values of orientations and calculate approximate cut-off times. Directions with a minimum processing time are chosen for performing deep processing operations. Further developments are aimed at changing geometrical tool characteristics and using straight and ball-nose cutters during the finishing process. An effective surface classification technique was developed to integrate milling tools and find out the cutting areas. The development of Computer-Aided Production (CAM) is implemented to conduct fast processing operations. It allows for the management of the CNC process with the help of special programming codes. Simulation study results are confirmed by the results of the processing experiment. First of all, this technology minimizes the time of rough work cutting in four independent directions and makes the process less sensitive to tool failures. Secondly, the use of milling tools increases the quality of the surfaces of machined parts. Finally, process planning programs provide effective simulation analysis and processing operations management. The application of computer numerical control (CNC) routers is very popular today because of the high-quality results that can be obtained [1]. This popularity is further supported by the widespread adoption of standardized data formats that facilitate diverse manufacturing applications [2].

The CNC machines are flexible enough that they use different tools, which allows them to undertake various operations like cutting, carving, and drilling of different types of material, including metal, wood, plastic, and steel. G-code instructions are used to control the movement of the CNC machine [3]. The correct interpretation of file structures, such as the RS274X standard, is critical for avoiding errors in CNC routing [4].

These instructions help to define where the machine is going to move and at what speed it will do so [5]. These codes are typically sent through the serial interface, such as a USB interface, from an external computer program [6]. This project deals with developing a computer program to run a CNC laser machine and generate G-code for the design of the pattern as per the plan. The primary application is the manufacturing of PCBs [7], [8], [9], [9]. Reliable CAD-to-CAM communication relies on modern format standards to ensure data accuracy and seamless integration [11].

The received results have been processed statistically.

Scientific novelty of the master's thesis.

• Use of PCB boards

• Installation of Stepper Motors

• Integration of open source software on hardware

Using a 3-axis CNC milling machine to ensure the availability of tools and justify many other technological priorities. But still, there is a need for improvement in the use of CNC in PC, and the task here is to improve deep machining and finishing operations, and the integration of cutting tools in CNC machining to adapt it to PC applications. The purpose of this research is to adapt CNC machining to rapid manufacturing [12], and it is assumed that implementation of the suggested approaches will help to accelerate the production process, improve the quality of parts, and make the process adaptable to PC. One of the ways to enhance deep machining operations is to apply various directions of cuts. The simulation analysis is conducted to manipulate the orientation values and obtain estimates of the cutoff times. For performing deep machining processes, directions with minimal processing times are chosen.

There are ongoing advancements to enable the combination of tools of varying geometry, and in the machining process, straight and spherical tools are applied. A technique for the classification of surfaces has been designed in order to facilitate the integration and definition of cut regions. The advancement of computer-aided manufacturing (CAM) technology is adopted to ensure rapid processing [13]. It enables you to control the planning process of the CNC process using individual programming codes. The results of the simulation study have been validated through the results of the processing experiment. Firstly, it ensures that there is a minimum roughing time in four separate directions while avoiding any sensitivity to tool breakage. Secondly, the combination of milling tools ensures better surface quality of the machined workpiece.

2. Running performance of the CNC-JS software

Tool guidance performance was assessed based on the performance of the application, meaning how efficient application processed the image into a G-code file, the run-time of the G-code file generated, and the cut-to-engrave process performed. The computer used in this experiment is a computer with an Intel i5-7200 dual-core (can run 4 threads at the same time) having a clock speed of 2.5 GHz and 8 GB of RAM; the following two images are used: 1. Image with a compact distribution of pixels to be deleted and 7 different resampling factors (Figure 1).

Image Properties				
Resample	Pixels		Pixels %	
Factor	Width	Height	To Remove	To Keep
0.125	125	103	39.25	60.75
0.25	250	205	38.08	61.92
0.5	500	410	37.34	62.66
1	1000	820	37.45	62.55
2	2000	1640	37.42	62.58
4	4000	3280	37.4	62.6
8	8000	6560	37.4	62.6

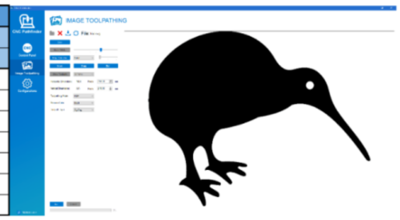

Figure 1. Compact distribution of pixels to be deleted.

An image with a more scattered distribution of pixels to be deleted and 7 different resampling factors (Figure 2).

Image Properties				
Resample	Pixels		Pixels %	
Factor	Width	Height	To Remove	To Keep
0.125	160	100	30.27	69.73
0.25	320	200	25.38	74.62
0.5	640	399	19.7	80.3
1	1280	798	21.27	78.73
2	2560	1596	21.3	78.7
4	5120	3192	21.28	78.72
8	10240	6384	21.27	78.73

Figure 2. Scattered distribution of pixels to be deleted.

As discussed earlier, the tool routing procedure can be broken down into five steps: image resampling, pixel segmentation, segment routing, segment linking, and guide generation. As image resampling uses the default interpolation methods, and both algorithms, zigzag and spiral, use the same resampled image, the execution time of image resampling is not considered. The following results were obtained for both Figures 1 and 2.

Table 1. G-code file generation time for Figure 1 with a compact distribution of pixels to be deleted

Tool Pathing	Image	File Generation Time [in seconds]								
Algorithm	Resample	Segmentation		Pathing		Connection		File		Total
	Factor	Time	%	Time	%	Time	%	Time	%	
ZigZag	0.125	0.0019	19.19	0.0026	26.26	0.0023	23.23	0.0031	31.31	0.0099
	0.25	0.0026	9.12	0.0009	3.16	0.008	28.07	0.017	59.65	0.0285
	0.5	0.0096	19.47	0.0021	4.26	0.0303	61.46	0.0073	14.81	0.0493
	1	0.0308	18.15	0.0044	2.59	0.1149	67.71	0.0196	11.55	0.1697
	2	0.1022	16.46	0.0142	2.29	0.4628	74.52	0.0418	6.73	0.621
	4	0.3984	16.48	0.0441	1.82	1.8611	76.98	0.1139	4.71	2.4175
	8	1.7011	17.18	0.1458	1.47	7.6036	76.78	0.4527	4.57	9.9032
Spiral	0.125	0.0005	1.82	0.0025	9.12	0.0009	3.28	0.0235	85.77	0.0274
	0.25	0.0163	16.65	0.0089	9.09	0.0024	2.45	0.0703	71.81	0.0979
	0.5	0.0463	14.92	0.0251	8.09	0.0109	3.51	0.228	73.48	0.3103
	1	0.1873	15.83	0.098	8.28	0.0458	3.87	0.8519	72.01	1.183
	2	0.7462	22.10	0.3821	11.32	0.0321	0.95	2.2157	65.63	3.3761
	4	3.0137	30.76	1.4901	15.21	0.0979	1.00	5.1969	53.04	9.7986
	8	12.1279	34.94	5.9168	17.04	0.2852	0.82	16.3849	47.20	34.7148

Table 2. Number of generated segments and Tool lines for Figure 1 with a compact distribution of pixels to be deleted.

Tool Pathing	Image	Number of Segments		Number of Lines				
Algorithm	Resample	Fill Segments		Tool On		Tool Off		Total
	Factor	Pre pathing	Post Pathing	Lines	%	lines	%	
ZigZag	0.125	103	167	167	49.85	168	50.15	335
	0.25	205	328	328	49.92	329	50.08	657
	0.5	410	659	659	49.96	660	50.04	1319
	1	820	1316	1316	49.98	1317	50.02	2633
	2	1640	2631	2631	49.99	2632	50.01	5263
	4	3280	5262	5262	50.00	5263	50.00	10525
	8	6560	10525	10525	50.00	10526	50.00	21051
Spiral	0.125	1	90	3008	97.06	91	2.94	3099
	0.25	1	178	11597	98.48	179	1.52	11776
	0.5	1	367	44061	99.17	368	0.83	44429
	1	1	732	177624	99.59	733	0.41	178357
	2	1	595	456097	99.87	596	0.13	456693
	4	1	967	1048047	99.91	968	0.09	1049015
	8	1	1670	3076772	99.95	1671	0.05	3078443

Table 3. G-code file generation time for Figure 2 with a scattered distribution of pixels to be deleted.

Tool Pathing	Image	File Generation Time [in seconds]								
Algorithm	Resample	Segmentation		Pathing		Connection		File		Total
	Factor	Time	%	Time	%	Time	%	Time	%	
ZigZag	0.125	0.0014	2.75	0.0008	1.57	0.043	84.48	0.0057	11.20	0.0509
	0.25	0.0055	1.44	0.0091	2.39	0.34	89.22	0.0265	6.95	0.3811
	0.5	0.0099	0.31	0.002	0.06	3.1009	97.39	0.0713	2.24	3.1841
	1	0.0319	0.30	0.0043	0.04	10.4474	98.56	0.1169	1.10	10.6005
	2	0.1261	0.28	0.076	0.17	44.1842	99.00	0.2444	0.55	44.6307
	4	0.409	0.22	0.0424	0.02	187.7034	99.48	0.5343	0.28	188.6891
	8	1.7478	0.21	0.1362	0.02	829.7499	99.61	1.357	0.16	832.9909
Spiral	0.125	0.0189	31.19	0.0095	15.68	0.0158	26.07	0.0164	27.06	0.0606
	0.25	0.0188	9.36	0.0107	5.33	0.1208	60.16	0.0505	25.15	0.2008
	0.5	0.0672	3.78	0.0156	0.88	1.3626	76.59	0.3336	18.75	1.779
	1	0.1994	4.71	0.077	1.82	3.4462	81.48	0.5068	11.98	4.2294
	2	0.7979	16.86	0.2441	5.16	2.4792	52.40	1.2102	25.58	4.7314
	4	3.1667	13.85	0.9318	4.07	15.6033	68.22	3.1692	13.86	22.871
	8	11.359	20.13	3.2112	5.69	33.7541	59.81	8.1123	14.37	56.4366

As can be seen in Tables 1 and 2, the Zigzag algorithm creates more segments and therefore spends more time connecting them. Note that the zigzag segments are controlled by only one plane, and the spiral groups move with pixels from the same "step" and thus control a larger area in the image of their middle segments. However, it also creates more space and not only spends more time on the road, but also calculates and converts physical positions in generating G-codes. Setting up more G-codes can lead to longer cutting and engraving operations. It may also be noted that the number of segments moving forward in a zigzag is equal to the height (in pixels) of the image, which is its largest size, and is equal to the number of segments moving in a spiral, a very compressed distribution of pixels to extract from a limited distance.

Table 4. Number of generated segments and tool lines for Figure 2 with a scattered distribution of pixels to be deleted.

Tool Pathing	Image	Number of Segments		Number of Lines				
Algorithm	Resample Factor	Fill Segments		Tool On		Tool Off		Total
		Pre pathing	Post Pathing	Lines	%	lines	%	
ZigZag	0.125	100	611	611	49.96	612	50.04	1223
	0.25	200	2025	2025	49.99	2026	50.01	4051
	0.5	399	6457	6457	50.00	6458	50.00	12915
	1	798	12554	12554	50.00	12555	50.00	25109
	2	1596	25246	25246	50.00	25247	50.00	50493
	4	3192	51480	51480	50.00	51480	50.00	102960
	8	6384	106307	106307	50.00	106308	50.00	212615
Spiral	0.125	49	341	2204	86.57	342	13.43	2546
	0.25	207	1146	7260	86.36	1147	13.64	8407
	0.5	889	3881	22535	85.30	3882	14.70	26417
	1	627	6342	83092	92.91	6343	7.09	89435
	2	628	5303	196524	97.37	5304	2.63	201828
	4	638	10910	515410	97.93	10911	2.07	526321
	8	669	17962	1462043	98.79	17963	1.21	1480006

Unlike previous tables, the Zigzag algorithm generates many more segments due to the scatter of pixels that should be removed, as the groups of pixels to be removed are larger on the same line and thus generate many more segments per column of the image. Of course, an obvious flaw is the linkage of segments. It will adversely affect the performance of the software for large images due to the quadratic dependency of the number of iterations of the sorting procedure on the number of segments, as defined by Eq. One approach could be to carry out a less resource-intensive search by finding several groups of segments, which are at the same distance from the origin (lower-left corner). But it will require additional time for the cutting and engraving process. The difference in the total number of lines between the two algorithms is smaller than that of the previous image set, but the spiral still generates more instructions. When considering the efficiency of both algorithms, the following values (CNC motion) were found by choosing a resolution of one pixel per millimeter:

The biggest difference between these algorithms is obviously the distance traveled by the CNC without actually engraving. If we consider that the CNC tool will move at constant speed regardless of acceleration and deceleration of direction changes, having in mind the properties of the CNC machine, which have a maximum feed rate of 1300 millimeters per minute, we can estimate that worst-case (8x) resampling factor would require approximately 31.49 hours to complete for zigzag algorithm and 34.39 hours for spiral, which is 9% slower. In this example, the zigzag algorithm proved faster at producing a G-code file, requiring just 9 seconds against 35 seconds for the other algorithm.

Table 5. Number of tool lines generated for Figure 1 with a compact distribution of pixels to be removed and the distance traveled by the CNC tool.

Tool Pathing Algorithm	Image Resample Factor	Number of Lines: Tool On Lines	Tool On %	Tool Off lines	Tool Off %	Total	Distance [in mm]: Tool On Distance	Tool On %	Tool Off Distance	Tool Off %	Total
ZigZag	0.125	167	49.85	168	50.15	335	39088	91.54	3611	8.46	42699
	0.25	328	49.92	329	50.08	657	76756	95.05	3994	4.95	80750
	0.5	659	49.96	660	50.04	1319	151766	97.53	3841	2.47	155607
	1	1316	49.98	1317	50.02	2633	305771	98.87	3501	1.13	309272
	2	2631	49.99	2632	50.01	5263	612337	99.30	4311	0.70	616648
	4	5262	50.00	5263	50.00	10525	1225334	99.61	4848	0.39	1230182
	8	10525	50.00	10526	50.00	21051	2452174	99.85	3773	0.15	2455947
Spiral	0.125	3008	97.06	91	2.94	3099	39704	74.17	13829	25.83	53533
	0.25	11597	98.48	179	1.52	11776	77356	73.80	27458	26.20	104814
	0.5	44061	99.17	368	0.83	44429	152350	73.79	54124	26.21	206474
	1	177624	99.59	733	0.41	178357	306355	74.93	102493	25.07	408848
	2	456097	99.87	596	0.13	456693	613355	92.96	46428	7.04	659783
	4	1048047	99.91	968	0.09	1049015	1226408	89.30	146934	10.70	1373342
	8	3076772	99.95	1671	0.05	3078443	2453281	91.47	228802	8.53	2682083

Table 6. Number of tool lines generated for Figure 2 with a scattered distribution of pixels to be deleted and the distance traveled by the CNC tool.

Tool Pathing Algorithm	Image Resample Factor	Number of Lines: Tool On Lines	Tool On %	Tool Off lines	Tool Off %	Total	Distance [in mm]: Tool On Distance	Tool On %	Tool Off Distance	Tool Off %	Total
ZigZag	0.125	611	49.96	612	50.04	1223	33856	72.57	12796	27.43	46652
	0.25	2025	49.99	2026	50.01	4051	56876	71.65	22499	28.35	79375
	0.5	6457	50.00	6458	50.00	12915	87704	69.84	37870	30.16	125574
	1	12554	50.00	12555	50.00	25109	204732	83.35	40897	16.65	245629
	2	25246	50.00	25247	50.00	50493	422419	90.49	44405	9.51	466824
	4	51480	50.00	51480	50.00	102960	856603	94.91	45900	5.09	902503
	8	106307	50.00	106308	50.00	212615	1724609	97.17	50236	2.83	1774845
Spiral	0.125	2204	86.57	342	13.43	2546	36016	62.08	21999	37.92	58015
	0.25	7260	86.36	1147	13.64	8407	60392	61.13	38407	38.87	98799
	0.5	22535	85.30	3882	14.70	26417	92856	58.85	64937	41.15	157793
	1	83092	92.91	6343	7.09	89435	210944	65.49	111167	34.51	322111
	2	196524	97.37	5304	2.63	201828	432391	80.45	105088	19.55	537479
	4	515410	97.93	10911	2.07	526321	866745	76.32	268973	23.68	1135718
	8	1462043	98.79	17963	1.21	1480006	1735652	78.22	483380	21.78	2219032

Table 6 shows that, again, the difference in distance is still significant when the tool is not cutting or engraving. Assuming the same bit rate as before, the estimated completion time for the worst case (8x resampling factor) would be about 22.75 hours for the zigzag and 28.45 hours for the spiral, a 25% increase in time. This means that despite the toolpath taking 14 minutes, the zigzag algorithm generated a G-Code file about 6 hours faster than the spiral, which took only one minute to toolpath. This shows that the fastest algorithm is not always the most efficient. However, there is still much room for improvement.

Toolpath Quality. The quality of the engraving depends very much on the type of material being engraved. Plywood was used to compare the print quality of both algorithms and the SVG outline, and the following result was obtained:

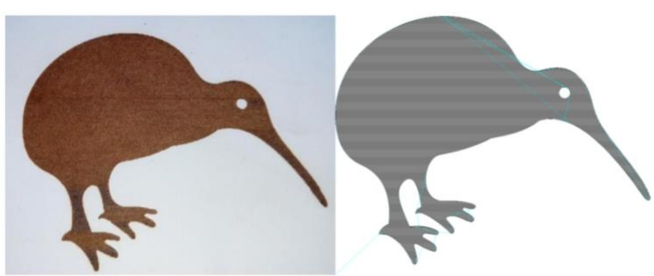

Figure 3. Image with a compact distribution of pixels printed using the zigzag algorithm. A. Print result (left); B. Simulation (right).

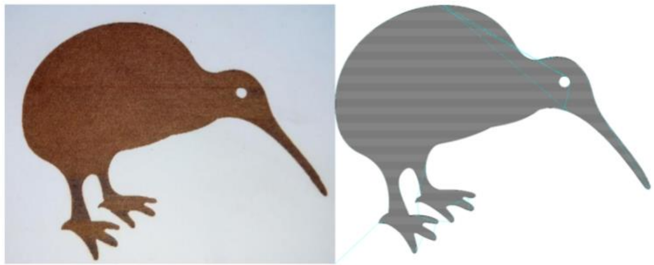

Figure 4. Image with a compact distribution of pixels printed using the spiral algorithm. A. Print result (left); B. Simulation (right).

The obtained results reveal that the engraving process using the zigzag algorithm is much smoother, as the number of turns or corners made by the CNC in the spiral algorithm leads to a greater duration of the laser being on in the same place [14]. The simulations have revealed a great difference between the distance covered when the laser is off, illustrated by the blue lines, thus resulting in a longer time for etching or cutting of the figure, as stated above. The physical size of the image was about 90 x 75 mm, and the duration of printing of both results was 14 minutes 28 seconds for zigzag and 19 minutes 14 seconds for spiral algorithms, respectively, using the feed rate of 1000 mm.

The main purpose of this project was to design a computer application that would be able to handle the CNC machine and tool path for all the most widely used types of image files. The CNC machine was controlled using the instructions given in G code, which explained how the CNC tool would move and what type of movement it should have [3]. A graphical user interface was created for the software using Windows Forms, which makes the user independent of the instructions and makes processes like calibration, printing, and orientation of the instruments easier. Such interface designs are essential for providing effective visual solutions that simplify the management of complex CNC workflows [15]. There is also a G-code simulator that analyzes the given file for the proper G-code instructions and displays the result so that the user can get an idea about the file and orient the CNC machine or the material before printing it. Coming to the image tool path, this project is related to the orientation of the raster images using the two most widely used methods, zigzag and spiral.. For the tests performed, the zig-zag method is the most efficient in terms of both time and quality of the cutting or engraving operation, although in some cases it takes longer to create the file. The selection and optimization of such path algorithms are critical for enhancing the efficiency of pocket milling processes [16]. While both try to minimize the distance traveled with the laser off, the spiral method also tries to create longer paths with the laser on, which results in fewer path segments with start and end points further apart.

The sorting algorithm that is employed to join the segments attempts to optimize only the proximity to the next path and not the entire length of the connection, making the latter unnecessarily elongated, sometimes even spirally. In order to address this issue, the sorting algorithm should analyze quite a few possible paths that may increase the time necessary to generate the toolpath file significantly. Utilization of the GPU for processing the toolpath would not only help to solve this problem but also speed up the process in general. Speaking about the SVG images, the conversion of Bezier and ellipses into simple lines and arcs was performed successfully, although there were some difficulties aligning the fill path with the outline. The latter were more noticeable for images with transformations like rotation or skew because the rendered version of the image did not correspond exactly to the outline path.

3. Result and Discussion

Although the project turned out to be significantly more complicated than I thought, it helped me learn about the topic of image processing and CNC machine control.

The results obtained demonstrate that the zigzag algorithm produces more uniform engravings since the number of turns made by the CNC in the spiral method results in longer exposure of the laser to the same place. In addition, the simulations reveal that there is a considerable difference in the distance traveled when the laser is not on, demonstrated by blue lines, thus making it more time-consuming to etch or cut a figure

The physical parameters of the image used were 90 by 75 millimeters, while it took 14 minutes 28 seconds and 19 minutes 14 seconds to print these two algorithms, respectively, at a feed rate of 1000 mm.

Practical significance of the research work: The results obtained are of great practical significance. So, the combination of CNC machines and software on the hardware, and its implementation on CNC-js JavaScript and distribution of algorithms may be significant.

Approval of work. The use of computer numerical control (CNC) routers has become extremely popular due to the high-quality results that can be achieved. CNC machines are versatile and can use different types of tools, allowing them to perform a wide variety of tasks such as cutting, engraving, and drilling different materials such as wood, plastic, or hard metals such as steel.

References

[1] House, N. (2011, July 20). *Building a cheap DIY CNC machine (1st attempt).* FoxyTronics. Accessed [April 20, 2026] https://www.foxytronics.com/blog/1/entry-13-building-a-cheap-diy-cnc-machine-1st-attempt/
[2] Artwork Conversion Software, Inc. (n.d.). *Applications of Gerber data*. Accessed [April 26, 2026] https://www.artwork.com/gerber/appl2.htm
[3] Denford Limited. (n.d.). *G and M programming for CNC milling machines*. Denford Limited. Accessed [April 20, 2026] https://cncmanual.com/denford-g-and-m-programming-manual-for-cnc-milling-machines/
[4] DiBartolomeo, S. (1995). *What's all this about RS274X anyway?* Artwork Conversion Software, Inc. Accessed [April 26, 2026] http://www.artwork.com/gerber/274x/rs274x.htm
[5] ManufacturingET. (2011, December 8). *CNC G code: G02 and G03*. Accessed [April 20, 2026] https://www.manufacturinget.org/2011/12/cnc-g-code-g02-and-g03/
[6] Roland DG Corporation. (2006). *NC code reference manual*. Accessed [April 20, 2026] http://www.imodela.co.uk/pdf/NC_Code_Reference_Manual.pdf
[7] Alberta Printed Circuits Ltd. (n.d.). *APCircuits*. Accessed [April 20, 2026] https://www.apcircuits.com
[8] Ucamco. (2024). *Gerber Layer Format Specification* (Revision 2024.05). Accessed [April 20, 2026] https://www.ucamco.com/en/guest/downloads
[9] Gerber, R. S. (1998). 274X format User's Guide. *Barco Gerber Systems*.
[10] Bittele Electronics Inc. (n.d.). *The difference between RS274D and RS274X Gerber formats*. 7PCB. Accessed [April 26, 2026] https://www.7pcb.com/blog/the-difference-between-rs274d-and-rs274x-gerber
[11] Ucamco. (2014, June 13). *Gerber X2: The new paradigm in CAD-to-CAM communication*. Accessed [April 26, 2026] https://www.ucamco.com/en/news/gerber-x2-the-new-paradigm-in-cad-to-cam-communication
[12] Bowyer, A. (2016, December 3). *RepRapWiki*. RepRap. Accessed [April 20, 2026] https://reprap.org/wiki/RepRap
[13] U.S. Congress, Office of Technology Assessment. (1984, April). *Computerized manufacturing automation: Employment, education, and the workplace* (OTA-CIT-235). U.S. Government Printing Office. Accessed [April 20, 2026] https://www.princeton.edu/~ota/disk3/1984/8408/840801.PDF

[14] Yao, Z., & Gupta, S. K. (2004). Cutter path generation for 2.5 D milling by combining multiple different cutter path patterns. *International journal of production research*, *42*(11), 2141-2161. https://doi.org/10.1080/00207540310001652879

[15] Bokland Custom Visuals. (2009). *Visual Solutions*. Accessed [April 26, 2026] http://www.bokland.com/page35/index.html

[16] Selvaraj, P., & Radhakrishnan, P. (2006). Algorithm for pocket milling using zig-zag tool path. *Defence Science Journal*, *56*(2), 117-127. https://doi.org/10.14429/dsj.56.1876