Automatic Drilling Machine for Printed Circuit Board

H. Ferdinando, I. N. Sandjaja, G. Sanjaya
Department of Electrical Engineering Petra Christian University

Abstract – **Using Printed Circuit Board (PCB) in the electronics instrumentation is common. In the experiment phase, not the final phase, one has to drill the hole and via by himself. The number of holes and vias in the PCB influence the speed of drilling time if it is done manually. The idea is to implement an automatic drilling machine. The system consists of two main parts, i.e. software in PC and the drilling machine itself. The coordinates for holes and vias were generated by PCB drawing software as text file. PC will extract and then send them to the machine via RS-232 connection. The drilling machine is equipped with MCS-51 family minimum system. This controls the mechanism of the drilling machine. To move the drill to the specific location, two stepper motor are used. This makes the control system in open loop. From the experiments, X- and Y-axes have different error, i.e. 2.34% and 4.34% respectively. This is due to the bending in worm system connected between the motor and the mechanics. For 90 holes (9x10) with 100 mils distance, the average time is around 5 minutes**

Keywords – **PCB, drilling machine, hole, via**

I. INTRODUCTION

The development of electronic circuit usually uses PCB or Printed Circuit Board. One of the processes in making PCB is to make a hole, whether for vias and pads. For small PCB, this process can be finished soon. But for larger one, this can take longer time.

This paper discusses the development of automatic drilling machine for PCB. After etching process, the PCB is placed to this system then it is drilled automatically.

The same system has been made but the controller is PC and uses camera to detect the coordinate position [1]. Another system tried to optimize the mechanism using Genetic Algorithm in order to shorten elapse drilling time [2]. The last research was about replacing the controller with PLC [3]. All previous systems have slow response.

System consists of a microcontroller (89C51), a mechanics for drilling process and PC. The mechanism of drilling process is 89C51's responsibility, but it is PC which sends the coordinates of the hole.

The mechanics bring the drill to the coordinate by stepper motors, X- and Y-axis. There is one DC motor which controls the drill up and down. Figure 1 shows the mechanics of the system.

All motors are controlled by the 89C51, while the 89C51 receives information from PC. PC gets coordinate information by reading a text file generated by software for PCB design. The data is communicated via serial port RS-232.

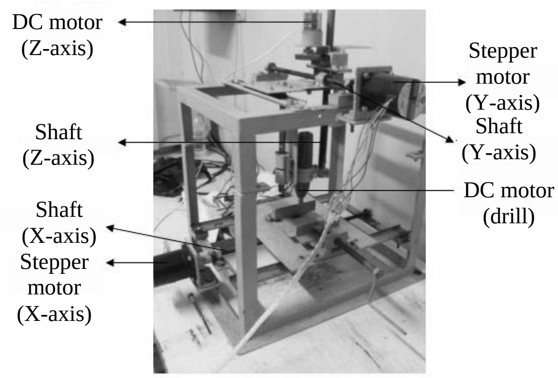

Fig. 1 The mechanics of the system [1]

The goal of this research is to shorten the elapse drilling time. It makes the performance better than the previous systems.

II. PREVIOUS SYSTEM

The previous systems have several disadvantages. One of them is the elapse drilling time is long. From the direct inspection, it was found that the shafts for both X- and Y-axis bend. This makes the movement of the drill not optimal. Sometimes, when the stepper motor rotates, the drill is still.

The joint between the shaft and the motor is also not good. This also gives contribution to the ineffectiveness of the movement.

System with PLC cannot move fast for the stepper motor's speed is limited by the speed of the PLC's output. The stepper motor works with pulse. If the pulse is too fast then there is lost pulse, but if it is too slow, the stepper motor is also slow. Unfortunately, we cannot use the PLC's output beyond its ability.

III. FIRST PROCESS

The first process is to fix the mechanics. The bending shafts are replaced with the new ones. But the movement is still not satisfied.

The joint between the shaft and the stepper motor is loosely. This makes the movement not efficient. The new join was made for this.

After all these things, the movement is not satisfied. The problem lies on the fact that the position of the stepper motors are not good.

Another additional component is limit switch. Two limit switches are added to the system. These switches are placed at the Z-axis.

IV. Hardware Design

The system is controlled by 89C51 general purposed microcontroller. The configuration is single chip and MAX232 is used as peripheral for serial communication with PC.

Several motor drivers are made, two for stepper motor (X- and Y-axis), one for DC motor (Z-axis) and one for the drill. The drivers use optocoupler to isolate the digital system from the analog one.

V. Software Design

There are two parts, i.e. for PC and 89C51. Both have different function but they have to collaborate to run the system. The PC gets the coordinate information and send it to 89C51 via serial port. Upon receiving this data, the 89C51 moves the drill to the position and then drill it. After finishing one hole, 89C51 asks PC for the next coordinate.

First, we will discuss software for the 89C51. Its role is to communicate data with PC, to control all mechanism of the system.

To control the stepper motor, 89C51 has to generate pulse. From the pre experiments, it is known that the delay for consecutive pulse is 2.5ms. This is the optimal delay. Shorter delay makes the pulse is lost while the longer delay makes the system slow.

The 89C51 also monitor the limit switch while rotating the stepper motor. This will guarantee that the drill will not move beyond the system.

The drill moves up and down by using the DC motor. At Z-axis, there is two limit switches to guard the movement at this axis.

Software in the PC reads a text file generated by PCB design software [4]. The PCB design software can generate a file text. In the file, we can find information about the coordinate of the opened file. Therefore, the PCB must be designed with this software also.

Figure 2 shows the opening screen of the PC software. User clicks the *Open File* button to open the text file. The coordinates from that file are placed in a database. Figure 3 shows the database screen.

User can also load the previous database and also save the current database. Here, one can select whether manual selected coordinate is chosen of automatic. With manual option, one can choose the coordinate then send it to the 89C51.

In the automatic mode, the software will sort the coordinate from X first then Y. This makes the drill move efficiently.

Once the drilling process is in progress, user can stop it by clicking *Cancel* button. *Reset* button is used to reset the position of the drill.

Coordinate from PC is sent with several formats. This depends on the number of digits in the coordinate. For each movement, the axis uses 4 bytes memory, i.e. 10s mm (at 040h), 1s mm (at 041h), 0.1s mm (at 042h) and 0.01s mm (at 043h). There must be some rule to place the coordinate data in the memory. The rules are little bit complex. But this is compulsory for the information from the text file is fixed.

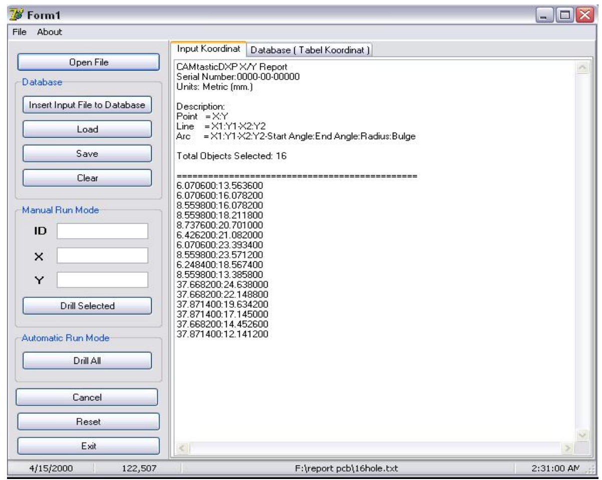

Fig. 2 The opening screen of the PC software

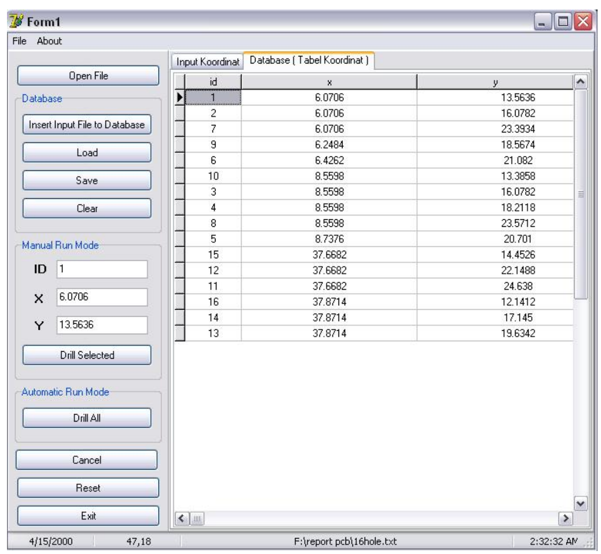

Fig. 3 The database screen of the system

Table 1a and 1b shows the memory address for each axis.

TABLE 1a
MEMORY ADDRESS FOR X-AXIS

Memory Address	Function
040h	To control every 10mm
041h	To control every 1mm
042h	To control every 0.1 mm
043h	To control every 0.01 mm

TABLE 1b
MEMORY ADDRESS FOR Y-AXIS

Memory Address	Function
044h	To control every 10mm
045h	To control every 1mm
046h	To control every 0.1 mm
047h	To control every 0.01 mm

Rule 1 is for 9 bytes coordinate. For example, the coordinate is 1.23:4.56. This data is 9 bytes long (point and colon are included). ‘1’ will be placed in memory address 030h, ‘.’ is in 031h, ‘2’ is in 032h, and so on. This uses ASCII code; therefore the value must be subtracted with 030h first.

Memory address 040h to 043h is cleared this will reset the position of the drill. The value in 030h is subtracted by the value in 041h. The result is used to move the drill to the desired location. If the result negative, then the drill move backward. The 89C51 will not calculate from address 031h, 034h and 036h for the values are not number.

Rule 2 is for 10 bytes coordinate. For example, 1.23:45.67 and 12.34;5.67. Rule 3 is for 11 bytes coordinate. For example, 12.34:56.78. With these rules, 89C51 process coordinate to move the drill.

VI. EXPERIMENTS

The PCBs for experiment are shown in figure 4. The first three are the same PCB from the previous systems.

The experiments are compared with the previous system. System 1 used camera [1], system 2 is optimized with GA [2] and system 3 used PLC [3].

Table 2 shows the performance of X- and Y-axis. The performance of X-axis is better than Y-axis. From the mechanics point of view this makes sense. The reason is that the shaft of Y-axis still has little bending. Another reason is the position of the Y-axis stepper motor is not positioned well.

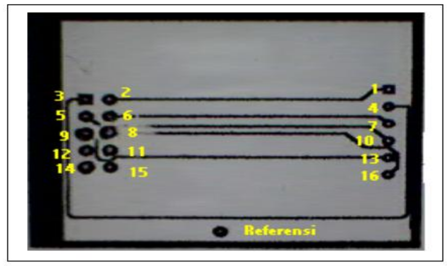

Fig. 4a PCB with 16 holes

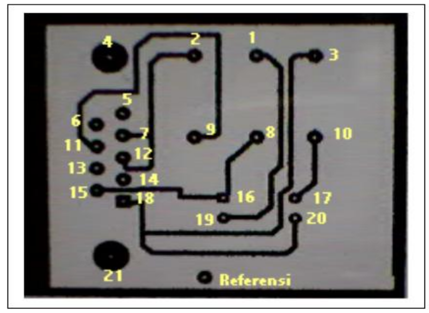

Fig. 4b PCB with 21 holes

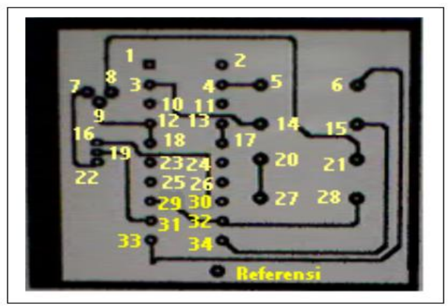

Fig. 4c PCB with 34 holes

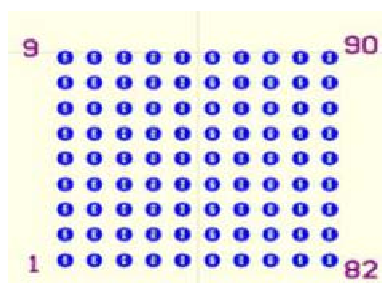

Fig. 4d PCB with 90 holes.

TABLE 2a
PERFORMANCE AT X-AXIS

Movements	***Full Stepping***				***Half Stepping***			
	Design (mm)	**Result (mm)**	**Error (mm)**	**Error (%)**	**Design (mm)**	**Result (mm)**	**Error (mm)**	**Error (%)**
360º	1.588	1.561	0.027	1.7	1.588	1.582	0.0055	0.346
720º	3.176	3.171	0.004	0.151	3.176	3.182	0.0064	0.201
0.1 mm	0.1	0.1	0	0	0.1	0.081	0.019	19
0.5 mm	0.5	0.463	0.0361	7.222	0.5	0.486	0.0139	2.78
1 mm	1	0.988	0.011	1.12	1	0.977	0.0222	2.22
5 mm	5	4.961	0.038	0.778	5	4.924	0.0751	1.502
1 cm	10	10.027	0.0278	0.278	10	10.027	0.0278	0.278
5 cm	50	50.175	0.175	0.35	50	50.1	0.1	0.2

TABLE 2b
PERFORMANCE AT X -AXIS

Movements	***Full Stepping***				***Half Stepping***			
	Design (mm)	**Result (mm)**	**Error (mm)**	**Error (%)**	**Design (mm)**	**Result (mm)**	**Error (mm)**	**Error (%)**
360º	1.588	1.522	0.065	4.143	1.588	1.566	0.021	1.341
720º	3.176	3.143	0.033	1.039	3.176	3.119	0.056	1.769
0.1 mm	0.1	0.105	0.005	5.5	0.1	0.094	0.005	5.6
0.5 mm	0.5	0.4872	0.0128	2.56	0.5	0.481	0.019	3.8
1 mm	1	0.95556	0.04444	4.444	1	0.975	0.025	2.5
5 mm	5	4.98333	0.01667	0.3334	5	4.9944	0.0056	0.112
1 cm	10	9.975	0.025	0.25	10	9.9875	0.0125	0.125
5 cm	20	19.9667	0.0333	0.1665	20	19.95	0.05	0.25

TABLE 3
COMPARISON BETWEEN CURRENT SYSTEM AND THE PREVIOUS ONE FOR PCB 16 HOLES

System	**Avg. Error X (mm)**	**Avg. Error X (%)**	**Avg. Error Y (mm)**	**Avg. Error Y (%)**
1	0.554	3.232	0.38	2.29
2	0.821	4.7894	0.307	1.852
3	0.48625	2.875	0.2639	1.605
Current	0.38055	2.22	0.4151	2.50462

TABLE 4
COMPARISON BETWEEN CURRENT SYSTEM AND THE PREVIOUS ONE FOR PCB 21 HOLES

System	**Avg. Error X (mm)**	**Avg. Error X (%)**	**Avg. Error Y (mm)**	**Avg. Error Y (%)**
1	0.331	2.2214	0.5833	5.0264
2	0.651	4.368	0.221	1.904
3	0.5428	3.642	0.6365	4.055
Current	0.3288	2.2066	0.4556	3.926

TABLE 5
COMPARISON BETWEEN CURRENT SYSTEM AND THE PREVIOUS ONE FOR PCB 34 HOLES

System	Avg. Error X (mm)	Avg. Error X (%)	Avg. Error Y (mm)	Avg. Error Y (%)
1	0.4755	3.003	0.7705	4.9165
2	0.672	4.244	0.488	3.133
3	0.3785	2.391	0.504	3.216
Current	0.5779	3.65	0.168	1.072

Tables 3 to 5 compare the performance of the current with the previous system. It is shown that the whole performance for the current system is not always better.

TABLE 6
ELAPSE DRILLING TIME

System	16 Hole (min)	21 Hole (min)	34 Hole (min)
1	16:40	13:09	19:52
2	7:49	9:59	12:17
3	27:36	19:13	29:27
Current	2:21	3:44	4:38

The elapse drilling time for various PCB holes as shown in figure 4 is different. The current system can upgrade this performance. System 1 depends on the image processing and limited pin out of the parallel port, while system 2 depends on the elapse time of the GA operation. The system 3 depends on the ability of the I/O port of the PLC. Usually, the ordinary discrete I/O of PLC has limited capability in handling the frequency. The current system can minimize the delay of the stepper motor. This gives significant changes for the elapse drilling time.

VII. CONCLUSION

The current system has better performance than the previous system. This result is achieved by replacing the both shaft for X- and Y-axis. The performance of the system can be upgraded by using smaller delay for the stepper motor.

VIII. REFERENCES

[1] Sompie, S. R. U., 2001, *Automatic Drilling Machine With Camera As Coordinated Detection Sensor*. Undergraduate final project no. No. 01/E/EL/020/534

[2] Rustan, T., 2003, *The Application of Genetic Algorithm to Optimize the Automatic Drilling Machine*. Undergraduate final project no. 02/610/ELK/2003. 2003

[3] Gunawan, D., 2005, *Automatic Drilling Machine with PLC*. Undergraduate final project no. 02010706/ELK/2005

[4] Sanjaya, G., 2005, *Automatic Drilling Machine with 89C51*. Undergraduate final project no. 02010741/ELK/2005.