Dynamic Management for Multiple Rotation Printheads with Interlace Printing: Part I: Pattern Paging & Memory Arbitration

Chia-Ming Chang*, Chih-Jian Lin & Kevin Cheng, Industrial Technology Research Institute,Hsinchu, Taiwan, R. O. C.,E-Mail :cmchang@itri.org.tw

Abstract

Difficulties in ink-jet fabrication are due to the versatile tunable resolution required, and, an ink-jet printing system with the architecture of multi printhead is usually used to expedite the processing time of production. It is complex arbitration for data format between the printing resolution and pattern data, especially for multi-heads platform architecture. In this architecture, we managed the printing data and kept their consistency between individual head modules, to avoid the lapses of printing data that causes defects, named "mura" at borders of printing regions near head modules intersections. This article proposed a memory arbitration mechanism for patterning fine images in multiple printhead system. The collective objective pattern data were allotted to each printhead depended on the input of physical dimension and rotation angle of printhead, and generated a parallelogram memory architecture at printing resolution to be printed and the system throughput can be raised to a much better one and the memory usage of the system will be very much efficiency.

Introduction

This article describes consideration of the system architecture and its memory arbitration method for ink-jet fabrication of the Printed Circuit Board (PCB) and display application. This memory arbitration method, also called the dynamic image data management method, arranges the image data for the rotation relation of printhead[1] to forward the usage of stocking memory.

Cheng et al. [1-5] discussed the feasibility of ink-jet fabrication. It has the innate benefits of saving material, direct patterning [6][7], and decreasing the yield rate as the panel size increases. For ink-jet printing procedures, the printing system equipped with the architecture of multi printhead [8][9] is usually used to expedite the process time. However, its difficulty is on the complex of data management, especially for an ink-jet printing required the tunable resolution. General solution is to rotate the print head to fit the pitch of nozzles consistent with pattern resolution, but it also created challenge of data management. Lapses in the management of printing data will cause defects in printing quality, and inconsistency of actions between individual head modules will prolong the processing time due to unnecessary platform actions.

In multi printhead architecture fabrications, a good image data arbitration mechanism can help us obtain a fine image without defects at borders of the printing region of head modules. Furthermore, the system throughput can also be raised by this arbitration mechanism. The dynamic memory data management method and the multi printhead architecture proposed by this article can obtain non-defect fabrications and efficient memory usage for the multi-head printing platform.

Multi Printhead Arrangement

In many cases of digital fabrications by ink-jet printing, the printing system with architecture of multi printhead is prospected to operate at high throughput [8]. It is usually demarcated to several printhead modules, which is constituted by several print heads with identical resolution in both horizontal and vertical directions. To simplify the data management, all these nozzle intervals between each heads are all equivalent in the direction perpendicular to relative movement direction between the head module and the substrate, as shown in Fig. 1.

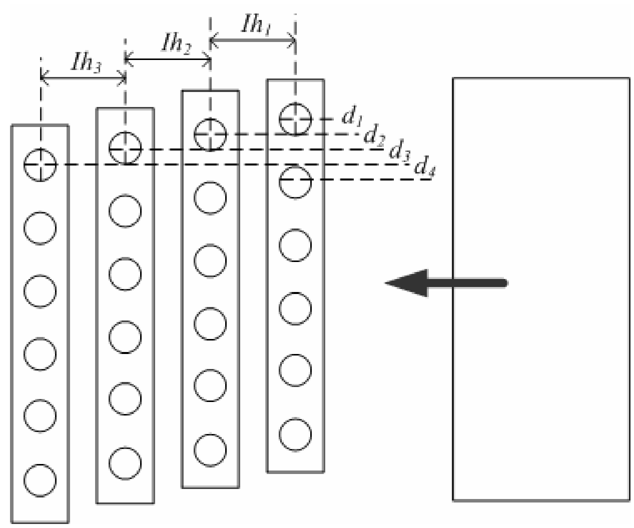

***Figure 1**. Architecture of the Collocated Printhead Module*

In Fig.1, d_1, d_2, d_3, and d_4 are equivalent in the direction perpendicular to the gray arrow. And this arrow indicates the direction of relative movement between the head module and the substrate. The interval of heads in horizontal direction: Ih_1, Ih_2, and Ih_3, are tunable values in this proposed patterning algorithm. The parallel printhead module, it is constituted by several parallel printheads with identical resolution in both horizontal and vertical directions. In this case, corresponding nozzles of each printhead, which are driven simultaneously, must be aligned in a straight line, as shown in Fig. 2.

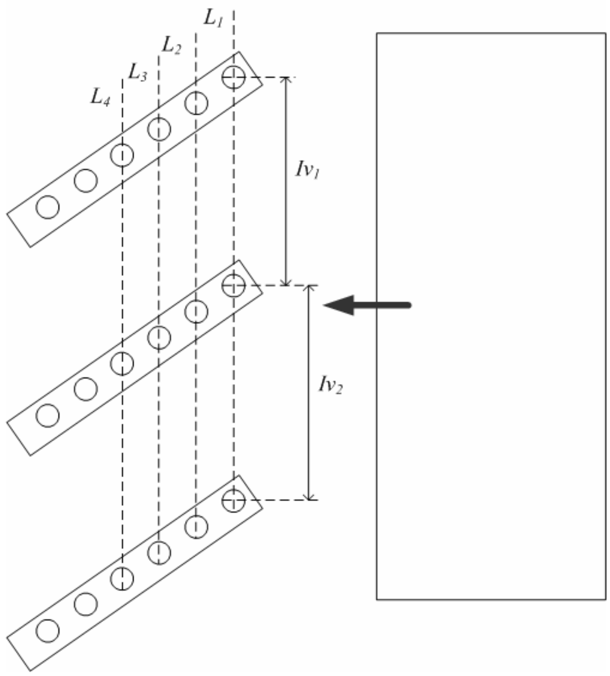

***Figure 2**. Architecture of the Parallel Printhead Module*

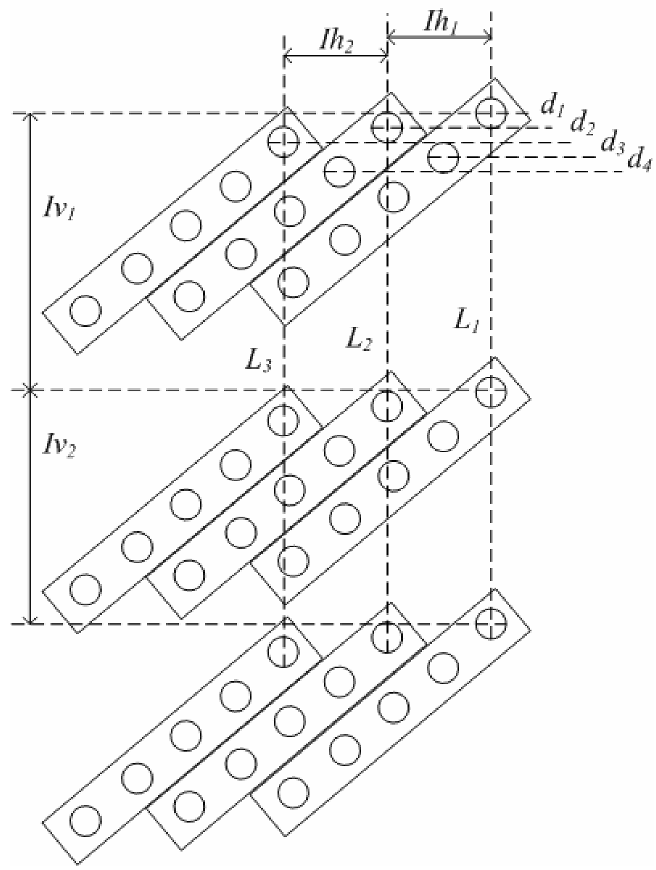

***Figure 3**. The Multi-Modules Architecture with Collocated and Parallel Heads Modules*

L_1, L_2, L_3, and L_4 in Fig. 2 are four straight lines crossing corresponding nozzles, and the interval of heads in vertical direction: Iv_1 and Iv_2 are tunable values in this proposed patterning algorithm using the method of dynamic data management. The collocated printhead module can be used to improve the printing resolution for each swath and reduce the interlace times, which expedites the processing efficiency. On the other hand, the parallel printhead module is used to increase the printable area on the substrate and can also expedite the processing efficiency of production.

Even more, a printing system can be composed by several collocated head modules and parallel head modules. This multi-module architecture can decrease the swath number and increase the system throughput during processing. In Fig. 3, d1, d2, d3, and d4 are equivalent. L1, L2, and L3 are straight lines. Iv1 and Iv2 are tunable intervals of heads in the vertical direction, and Ih1, Ih2, and Ih3, are tunable intervals of heads in horizontal direction. In this article, Iv also called the module distance and Ih also called collocated distance. In this research and following [10], we disclosed a completed flow paging and its memory arbitration method, the so-called interlace-rotation patterning algorithm and its complexity of splitting swath data.

By use of the memory arbitration mechanism, so-called dynamic data management, proposed by this article in the multi printhead system, a collectively objective pattern can be obtained. And the printed data can be allotted to each printhead without observing any defect in quality. Moreover, the efficiency of printing process can be raised to a better one because actions of head modules in the platform are identical. Below will introduce the detail of flow paging and the memory management arbitration.

Flow Paging & Memory Arbitration

This article presented an image paging flow and its memory data mapping architecture, to make an effective and fast processing for multi-heads ink-jet printing. The specialty is the combination of the position shifting between print heads into the image data assignment to avoid from landing deviation, and high accuracy printing can be realized. It is important to include relative operation parameters like the patterning resolution, interlace times, nozzle shift delay count, module distance and collocated distance, as input to construct the completed image data. Thus, the image data will contain the print head shifting and process information to be printed.

After the original image had established, then step 1-3 page the image data, and arbitrate to memory at step 4-5, described below.

<Step1> Data Format Transfer

Before starting the data management steps, the image data must be transferred to a bit stream from the format of Bitmap, TIFF, JPEG, Gerber or other data formats. These image data are processed by halftone algorithm [11] to become printable bit stream data. If it needs, a pre-processing of the image trimming [1][12] steps can make the produced pattern become more conformable to the substrate surface property, and output this trimmed image in raster data to be ready for ink-jet printing.

<Step2> Distributing Printable Data to Each Head

A raster data, is needed to slice into parts according to the print head modules arrangement, printing resolution, the rotation angle of print head, interlace times, and the size of printing etc. The concept is to allocate each head module with partial raster data, according to the position distance between head modules. In this idea, for example, the original image data was divided to several parts and distributed to each module and then further separated those parts data down to for each head in each module. Fig.4 sketched the detail. Where we defined the vertical splitting length L of image computed by

$$L= Iv/PR \tag{1}$$

, in which where L is the image splitting length, Iv is the module distance, and PR is the patterning resolution.

As an example, head module 1 and head module 2, will split the raster data into L_1, L_2, and L_3 parts. In printable area 1, all the heads of module were in same operation, the head module 1 & head module 2 were responsible for same printing area, but in different data contents. But for printable area 2, the reminders parts of the image data, it can't be full coverage by all head module, only head module 1 was operated in this case, the L3 data contents. In the same concept, this image splitting methodology can expand to plurality of head modules, which each module can further include plurality of heads.

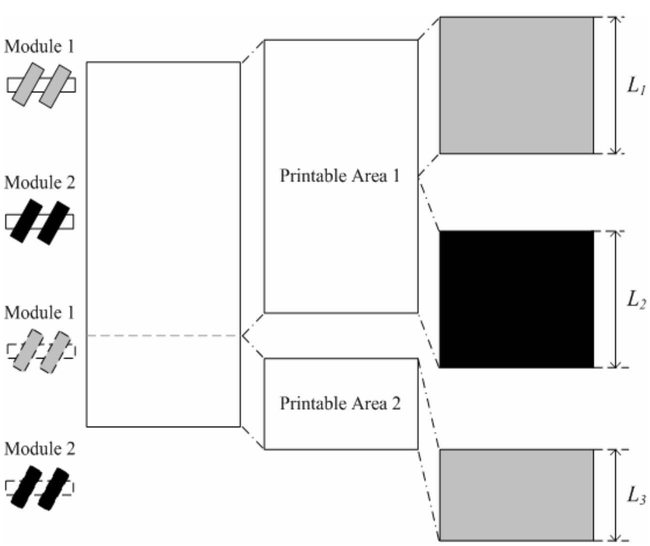

***Figure 4**. The Data Distribution Operated for Each Head Module in Different Swath Printing. (Data Printed by Module 1, Gray Area, Data Printed by Module 2, Black Area)*

<Step3> Data Interlacing for each Head

Said L_1, L_2, and L_3 were a printing area operated by certain head module. But for each module, it may contain several heads, need to further distribute data contents to each head, depends on the head numbers, the nozzles adopted by each head, the operation of interlace times (related to printing resolution) and the angle of the print head rotation. .

1.Data Shifting

To fit the image resolution need, the print head may rotate at certain angle, and it makes the timing delay for nozzles. The timing lag between nozzles are transferred as dummy data, and inserted into the image contents. Here we explained this mechanism for one head module with multiple heads. In Fig.5, D_0, D_1, D_2, D_3, and D_4, are delay counts of nozzles caused by printheads rotation, Ih_1 and Ih_2 are the collocated distances of multi printhead architecture.

$$\begin{aligned} &D_0=0,\ D_1=Ih_1,\ D_2=Ih_2, \\ &D_3=D,\ D_4=Ih_1+D,\ D_5=Ih_2+D, \\ &D_6=2D,\ D_7=Ih_1+2D,\ D_8=Ih_2+2D, \\ &\quad\vdots \end{aligned} \tag{2}$$

In the above list, D is the nozzle position shift delay count [10] relative to the physical position of printheads. When the nozzle number is N and collocated head number is n, the nozzle shift delay count of collocated printheads can be computed by:

$$D(N, AH)=Ih_{AH}+N\times D \tag{3}$$

, where AH are the number of collocated printheads, which from the closest with the substrate to the farthest one is equal to 0, 1, 2,…. And Ih_{AH} are the collocated distances of each printhead, from the closest with the substrate to the farthest one is equal to 0, Ih_1, Ih_2,....

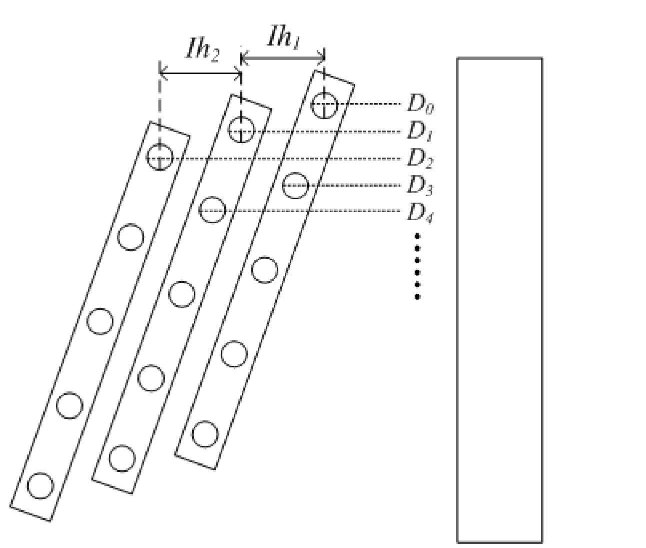

***Figure 5**. The Timing Delay and Its Physical Sketch Due to the Head Alignment at Certain Angle.*

2.Margin Limitation

In operation, the printing data will piping into memory, and then trigger a signal to drive the output voltage connected with nozzle. Sometimes, especially at the boundary of image, the image data for printing at the boundary are not match with the chose nozzles. The over range parts will insert dummy data for the nozzles over the physical margin (i.e, send 0 for not firing), to consist with the data format at each swath. In Fig. 6, the Region A is the normal swath data to print, within the printable area. Region B & C are the swath data face to the printing margin, need to be special process. We created dummy region C data (not firing) coupling with region B data, to complete a full use of chose nozzle similar region A, i.e, the data format is consistence due to A equals to B plus C.

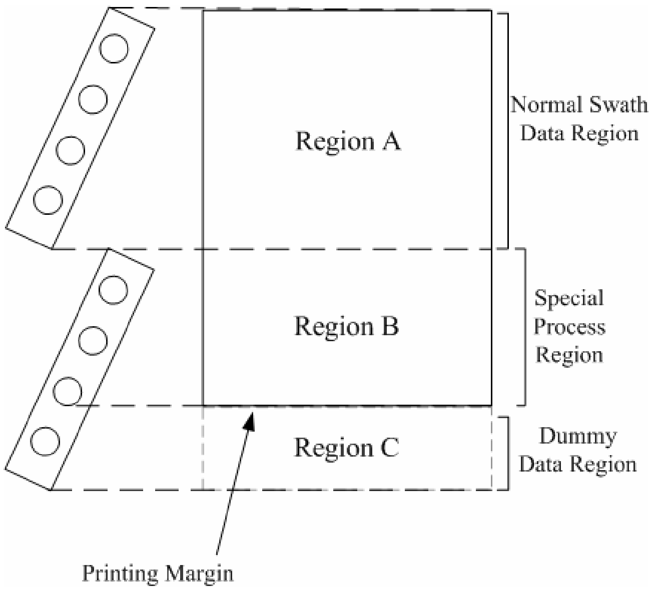

***Figure 6**. Normal Area and Filled Area*

<Step4> Data Allotting

After several swath data were made up by step3, an arbitrative mechanism is executed to allot these sequence swath data to particular printheads for firing.. An array is used to record the number of swath data that is allotted to each printhead. For example, if there are three printheads in a module and two modules in a printing system. The head swath array in format of [3, 2, 2; 1, 1, 1] means module #1 Head #1 will printing three swaths, and in the same way, module #1 Head #2 for two swaths, module #1 Head #3 for two swaths, etc. In manipulating, the swath indexes and head indexes are computed by following formulas:

$$HI=AN\times MN+(SN\%AH) \quad (4)$$

$$SI= HAS(MN, AN) \quad (5)$$

MN=0, 1, 2,...; AH=0, 1, 2,...

, where HSA is the component of head swath array, HI is the head index, SI is the swath index, SN is the swath number, MH is the module number and AH is the collocated head number.

According to the head index HI and swath index SI, the swath data can be allotted to the SIth swath of the HIth head. And the head swath array must be updated by:

$$HAS(MN, AN)=HAS(MN, AN)+1 \quad (6)$$

For ensuring actions of each head are consistent. The maximum value in the head swath array should be found and all components in the array must be equal to this value, and dummy data will be allotted to other printheads. In the previous example, the head swath array will be updated to [3, 3, 3; 3, 3, 3].

In summary, the index of start rows must be recorded for computing the position array latter.

$$(\text{Start Row})_{n+1} = (\text{Start Row})_n + 1 \quad (7)$$

By the above five allotting steps, swath data can be allotted to each printheads to execute data deposition. An example of the swath data allotting is shown in Fig. 7.

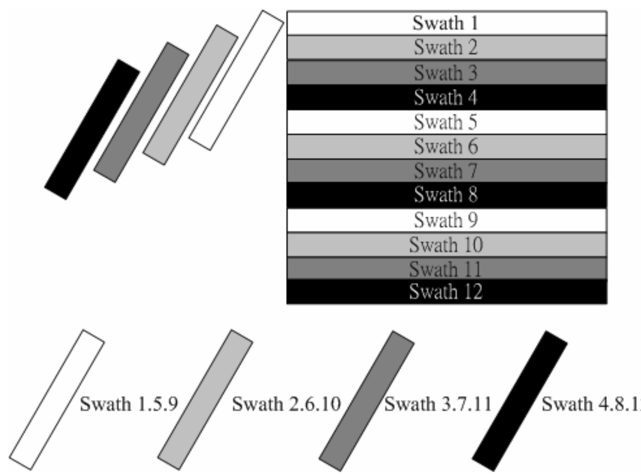

***Figure 7**. Data Allot Mechanism Operated for Multiple Heads in a Print Head Module.*

<Step5> Building Position Array

Position array is an important parameter to do positional control in print systems. It can be computed by start row indexes that are recorded in the allotting swath step. The printing data can be produced by the above dynamic data management steps and the manipulated data can be printed by the firing module to obtain fine images, as shown in Fig.8. Where, the original image data are divided to several parts to each printhead modules. Through swath data slicing, nozzle data shifting, module data filling and swath allotting steps, sequence swath data are made up and become efficient forms to be deposited.

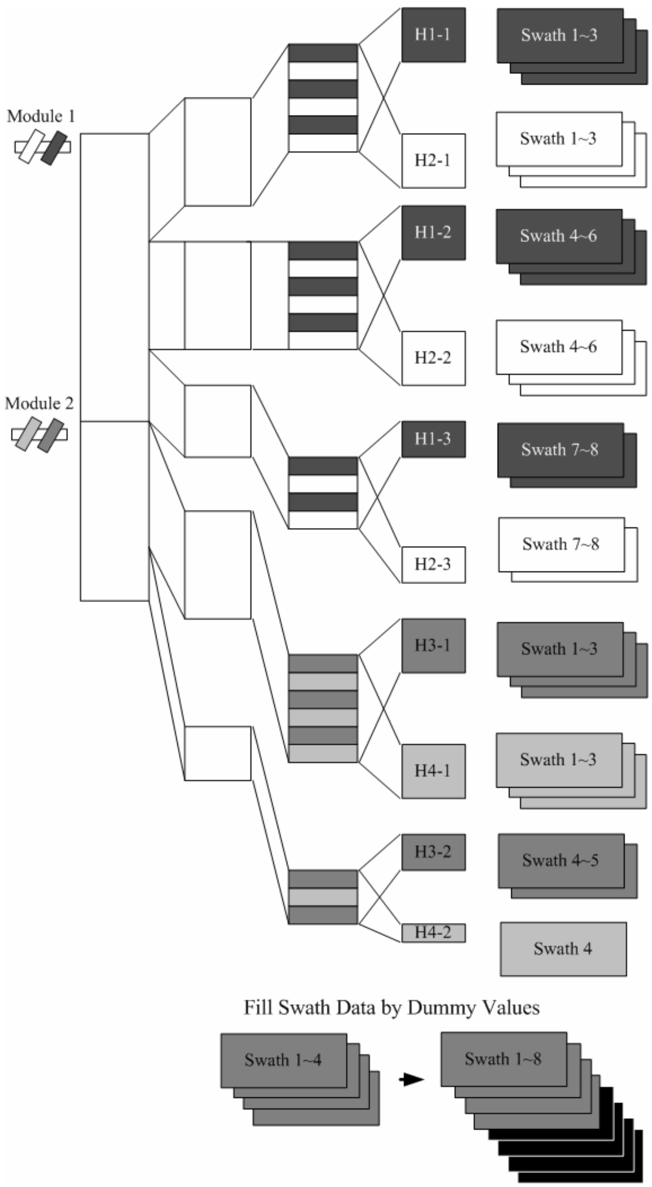

***Figure 8**. The Paging Flow of Image Data Distributed Into Memory for Multiple Modules with Plurality of Heads.*

Conclusion

In this article, we disclosed a patterning paging flow and memory data arbitration method for multi printhead architecture. Though this memory arbitration mechanism, a collectively objective pattern can be obtained and the printed data can be allotted to each printhead without observing any defect in quality. Besides, the efficiency of printing process can be raised to a better one because actions of head modules in the platform are identical and memory usage during the process is economical.

In multi printhead architecture fabrications, image data are mapped through the image paging flow, including swath data slicing, nozzle data shifting, module data filling and swath allotting steps, to make an effective and fast processing for multi-heads ink-jet printing. Then, a fine image without defects at borders of the printing region of head modules can be obtained. Furthermore, the system throughput can also be raised by this arbitration mechanism for the multi-head printing platform.

Reference Preparation

[1] Chih-Jian Lin, Chih-Hsuan Chiu, Chia-Ming Chang, Gian-Hung Liu, Yung-Kuo Ho, Jane Chang & Kevin Cheng, "Tunable Resolution and Patterning Method for Ink-Jet Printing Process", Digital Fabrication 2005 - Final Program and Proceedings, 2005, p 89-92

[2] Kevin Cheng et al, "A Novel Application of Ink-Jet Printing Technology on Manufacturing Color Filter for Liquid Crystal Display", NIP 17: International Conference on Digital Printing Technologies, pp.739-743, 2001.

[3] Wanda Wan-Wen Chiu & Kevin Cheng et al, "Ink-Jet Printing Technology on Manufacturing Color Filter for Liquid Crystal Display Part I: Ink-Jet Manufacturing Processes", NIP 19: International Conference on Digital Printing Technologies, pp.303-308, 2003.

[4] Kevin Cheng et al, "Ink-Jet Printing Technology on Manufacturing Color Filter for Liquid Crystal Display Part II: Printing Quality Improvement", NIP 19: International Conference on Digital Printing Technologies, pp.309-313, 2003.

[5] Chih-Hsuan Chiu, Chung-Wei Wang, Ming-Huan Yang, Chih-Jian Lin, Jane Chang, Jinn-Cherng Yang, Chun-Jung Chen & Kevin Cheng, " Image Segmentation and Trimming for Ink-Jet Fabrication of Electronic Circuits", Digital Fabrication 2005 - Final Program and Proceedings, 2005, p 68-71

[6] Richard Bennett and Dave Albertalli, "Use of Industrial Inkjet Printing in Flat Panel Displays", International Display Manufacturing Conference (IDMC), 2005.

[7] David Albertalli, "Gen 7 FPD Inkjet Equipment – Development Status", Society for Information Display (SID), 2005.

[8] Sawyer B. Fuller, Dric J. Wilhelm, and Joseph M. Jacobson, "Ink-Jet Printed Nanoparticle Microelectromechanical Systems", IEEE Journal of Microelectromechanical System, VOL. 11, NO. 1, February 2002.

[9] Fujitsu Lab Ltd, Kawasaki, Jpn "Indirect Discharge Recording", Conference Record - IAS Annual Meeting (IEEE Industry Applications Society), 1979, p 30-33.

[10] Chia-Ming Chang, Chih-Jian Lin & Kevin Cheng, "Dynamic Management for Multiple Rotation Printheads with Interlace Printing: Part II: Patterning Algorithm", accepted by International Conference on Digital Fabrication Technologies, 2006.

[11] Li Pingshan; Allebch, Jan P., "Look-Up-Table Based Halftoning Algorithm", IEEE International Conference on Image Processing, v 2, 1998, p 34-38

[12] M. Moganti and F. Ercal, "A Subpattern Level Inspection System for Printed Circuit Boards", Computer Vision and Image Understanding 70, 1998, 51-62.

Author Biography

Chia-Ming Chang received the B.S. degree in Aerospace engineering from Tamkang University, Tamsui, Taiwan in 2001. and M.S. degree in electrical & control engineering from Nation Chiao Tung University, HsinChu, Taiwan in 2003. He is currently working in Display Technology Center of Industrial Technology Research Institute, Hsinchu, Taiwan, as an associate Engineer. His work has primarily focused on the image processing and industrial ink-jet printing algorithm development.