การทำมาตรฐานกระบวนการแยกสีสำหรับระบบการพิมพ์เฟล็กโซกราฟี

นาย สุรัช นัทธี

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาเทคโนโลยีทางภาพ ภาควิชาวิทยาศาสตร์ทางภาพถ่ายและเทคโนโลยีทางการพิมพ์
คณะวิทยาศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2544
ISBN 974-17-0131-4

STANDARDIZATION OF COLOR SEPARATION PROCESS FOR FLEXOGRAPHIC PRINTING

Mr. Surush Nuttee

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science in Imaging Technology
Department of Imaging and Printing Technology
Faculty of Science
Chulalongkorn University
Academic Year 2001

ISBN 974-17-0131-4

Thesis Title	STANDARDIZATION OF COLOR SEPARATION PROCESS FOR FLEXOGRAPHIC PRINTING
By	Mr. Surush Nuttee
Field of Study	Imaging Technology
Thesis Advisor	Associate Professor Aran Hansuebsai, Ph.D.
Thesis Co-advisor	Associate Professor Pontawee Pungrassamee, M.S.

Accepted by the Faculty of Science, Chulalongkorn University in Partial Fulfillment of the Requirements for the Master's Degree

……. Deputy Dean for Administrative Affairs

(Associate Professor Pipat Karntiang, Ph.D.) Acting Dean, Faculty of Science

THESIS COMMITTEE

…….. Chairman

(Professor Suda Kiatkamjornwong, Ph.D.)

……. Thesis Advisor

(Associate Professor Aran Hansuebsai, Ph.D.)

……. Thesis Co-advisor

(Associate Professor Pontawee Pungrassamee, M.S.)

…….. Member

(Associate Professor Wanna Sananpanichkul, Ph.D.)

นาย สุรัช นัทธี : การทำมาตรฐานกระบวนการแยกสีสำหรับระบบการพิมพ์เฟล็กโซกราฟี (STANDARDIZATION OF COLOR SEPARATION PROCESS FOR FLEXOGRAPHIC PRINTING) อ. ที่ปรึกษา : รศ.ดร.อรัญ หาญสืบสาย, อ. ที่ปรึกษาร่วม :รศ.พรทวี พึ่งรัศมี 100 หน้า. ISBN 974-17-0131-4

การมาตรฐานการพิมพ์เฟล็กโซกราฟี เป็นกระบวนผลิตฟิล์มแยกสี CMYK ให้ได้ตามคาดหวัง ในการวิจัยนี้เริ่มด้วยการออกแบบแบบทดสอบ แล้วนำไปพิมพ์เพื่อให้ได้ข้อมูลลักษณะจำเพาะของปัจจัยต่าง ๆ ที่เกี่ยวข้อง ได้แก่ กราฟแสดงค่าความดำ และการเกิดเม็ดสกรีนบวมของการพิมพ์แต่ละสี และสมดุลเทา (grey balance) นำค่าที่ได้พร้อมกำหนดน้ำหนักสีภาพที่ต้องการและชนิดของแม่พิมพ์ดำ (black printer) มาคำนวณการผลิตน้ำหนักสี (tone reproduction) ของฟิล์มแยกสี CMYK ผ่านโปรแกรมที่สร้างขึ้น การทดลองนี้ได้พิมพ์ทดสอบเพื่อประเมินผลโปรแกรม พบว่าสิ่งพิมพ์ที่ได้มีการผลิตน้ำหนักสีสอดคล้องกับข้อมูลที่กำหนดข้างต้น

ภาควิชา วิทยาศาสตร์ทางภาพถ่ายและเทคโนโลยีทางการพิมพ์ ลายมือชื่อนิสิต .....
สาขาวิชา เทคโนโลยีทางภาพ ลายมือชื่ออาจารย์ที่ปรึกษา.....
ปีการศึกษา 2544 ลายมือชื่ออาจารย์ที่ปรึกษาร่วม.....

4172510823: MAJOR IMAGING TECHNOLOGY
KEY WORD: TONE REPRODUCTION / GREY BALANCE / BLACK PRINTER /

SURUSH NUTTEE : STANDARDIZATION OF COLOR SEPARATION PROCESS FOR FLEXOGRAPHIC PRINTING. THESIS ADVISOR : ASSOC. PROF. ARAN HANSUEBSAI, Ph.D. THESIS CO-ADVISOR : ASSOC. PONTAWEE PUNGRASSAMEE, MS. 100 pp. ISBN 974-17-0131-4

Standardization of flexographic printing is a process to produce the predicted CMYK separation films. In this research, a test form is designed and printed to obtain the characteristics of the relevant factors including density and dot gain curves for each color and grey balance. The obtained data together with the input data: required tone reproduction and type of black printer were then used to calculate tone reproduction of each color separation film through the presently invented program. The experiment includes printing to evaluate the program. It is found that the obtained tone reproduction of print is close to the specification given before.

Department Imaging and Printing Technology Student's signature ……..
Field of study Imaging Technology Advisor's signature .....……
Academic year 2001 Co-advisor's signature ……..……

ACKNOWLEDGMENT

I would like to express my sincere respects and appreciations to my advisor, Associate Professor Dr. Aran Hansuebsai, and my co-advisor, Associate Professor Pontawee Pungrassamee, for their guidance and encouragement in several aspects, particularly in research management, as well as his friendliness and frankness. Special thanks go to Professor Suda Kiatkamjornwong, Ph.D. and Associate Professor Wanna Sananpanichkul, Ph.D. for their suggestions at the writing step of the thesis.

I would like to thank Mr.Pornpracha Wattanakijsiri, Plant Manager of Tetra Pak Manufacturing (Thai) Co.Ltd., for providing opportunities to research in the world class packaging factory. Thank also Ms.Ratchanee Tangkitshanguan, QC manager for preparing essential materials for my research and her concerns about progresses of my laboratory. I would like to thank the prepress and printing staff for their corporations and suggestions during my research at Tetra Pak Manufacturing (Thai) Co. Ltd.

Besides I would like to thank Miss Kanyarat**,** staff of Coates (Thailand) Co. Ltd, for her corporations and guidance in printing systems and ink properties and Mr. Weerawate, my friend, for reading and editing my thesis writing.

Finally, I would like to thank my father, my mother, my sisters and also their families for their moral supports and encouragements during study.

Surush Nuttee

CONTENTS

CONTENTS (continued)

PAGE

CONTENTS (continued)

LIST OF TABLES

LIST OF FIGURES

LIST OF FIGURES (continued)

FIGURE PAGE

CHAPTER 1

INTRODUCTION

Nowadays almost of color separation films for flexographic printing process are made based on the requirement for lithographic offset printing. The adjustment depends on the experience of the color separator. They derive dot area for highlight area through shadow area and including the grey balance by trial and error. Therefore the results in printing cannot be predictable. One reason is that the characterization of lithographic offset printing and flexographic printing are different, especially in dot gain occurred. This factor affects their tone reproduction directly. Besides the dot shape is also difference. In this research, standardization's concept is to collect the relevant data in the printing and platemaking process and to convert them into the specification for the color separation process. In addition to the parameter of the press operation, the characterizations of the original have to be taken into account.

1.1 Objective

To propose a system how specifications can be derived and how they are applied to generate the halftone separations.

1.2 Scope of the Research

Flexographic printing characterized data are required for color separation making. And the computer program will be developed to calculate the predicted color separation films from the input data, which compose of solid tone density, substrate, inks, density of original, characteristics of the black printer and tone reproduction.

1.3 Content of the Thesis

Chapter 2 deals with the overview of the theoretical considerations and literature reviews.

In the Chapter 3, the description on materials under study, the experiment procedures and apparatuses are described.

Chapter 4 contains the results and discussion on this research. Finally, the results are concluded in chapter 5 with some suggestions.

CHAPTER 2

THEORY AND LITERATURE REVIEW

2.1 Theoretical considerations

2.1.1 Definition of Flexography

Flexography is a method of direct rotary printing that uses resilient relief image plates of rubber or photopolymer material. The plates are mounted to plate cylinders of various repeat lengths, inked by the cell-structured ink-metering roll, with or without a reverse-angle doctor blade, and carrying a fast drying fluid ink to plates that print onto virtually any substrate, absorbent or nonabsorbent.

Flexography is a rotary printing method. For every revolution of the printing plate cylinder, an image is produced. The heart of the flexographic printing process is its simple ink system.

A thorough grasp of flexography's definition and the representative diagram are shown in Figure 2-1. The ink-fountain pan supplies ink to the rubber fountain roll, which supplies ink to the anilox roll. Then anilox roll will transfer uniform levels of ink from its cells in the anilox roll to the surface of the printing plates. The plates mounted to the printing plate cylinder carry the ink to the substrate as it travels through the press. The impression roll supports the substrate. [1]

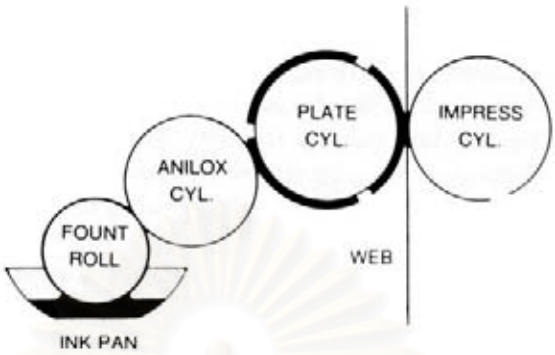

Figure 2-1 Flexographic printing station

2.1.2 Flexographic process color printing

Process color printing describes the technique of full color reproduction of a subject, rendering color original in full, continuous tone color. This is achieved in printing by first converting continuous tone original to four halftone separations namely cyan, magenta, yellow and black. Then plate are made from these separations and used to print these colors sequentially in register with each other. The black is used to improve the contrast and tone balance of the reproduction. This process is essentially the same for all printing methods, but corrections are needed for each method.

The objective of process color printing is to take an original colored subject and produce multiple copies that resemble the original as closely as possible. The process color printing requires the photographic or electronic separation of the colors of the original into four basic colors and the breaking up to the continuous tone image into halftone dots. The end result is four black and white photographic films, each containing the color and tone value of one basic color from the original. From these films, printing plates are made and these are used to sequentially print each color in register to give a reproduction of the original.

To successfully print process color with flexography, it is important to understand each step in the process and how to perform it. This step-by-step analysis starts by understanding the press so that corrections for the mechanics of the printing can be included at the manufacturing stage. (1)

2.1.3 Input data and their significance

2.1.3.1 Solid tone density and dot gain

Dot gain refers to the difference between the effective dot area of the halftone and the apparent dot area of the print. This specification is used to describe the shape of the characteristic curve between the dot area of halftone and the densities of print in the simplest way. (2)

2.1.3.2 Density range of the original

The density range of the original determines together with the maximum attainable density of print whether or not a facsimile reproduction can be achieved. Usually, the density range of transparencies is considerably higher than the density range attainable in printing, which it makes necessary to compress the tone reproduction curve. Since every compression is equivalent to a loss of tonal values and hence to a loss of information, the density range of originals should ideally not exceed 2.3. [(2)]

2.1.3.3 Tone reproduction

Tone reproduction is a used to control point for light and dark range compression in light or dark area of the original as required. In general the maximum density of the original is higher than the printing can reproduce. The density range of the transmission is 2.4 or higher and the reflection is 1.6 or higher. Therefore the compression of the tone scale is necessary to use. The uniform color value range compression from highlights to shadows produces a very unsatisfying reproduction. This is because the needed details can be lost. A better way to improvement is alternatively the selective compression of light and dark areas, which allowing the necessary tones for good definitions of image details to retain their original range of subtleties and brightness, while the unimportant tones absorbs the compression. [(3)]

The widest accepted type of compression is a visually uniform compression of Munsell gradation equation (Eq.1), which can be calculated from the density range of the print and the original.

$$y = 2\left\{1 - \log\left[10^{1-\frac{y_E}{2}} + \frac{10 - 10^{1-\frac{y_E}{2}}}{10 - 10^{1-\frac{x_E}{2}}}\left(10^{1-\frac{X}{2}} - 10^{1-\frac{x_E}{2}}\right)\right]\right\} \qquad \text{(Eq.1)}$$

When y = the density of the print

y_E = the maximun density of the print

x = the density of the original

x_E = the maximun density of the original

There are many types of gradation to relate with the original, normal key, low key and high key. It shows in Table 2-1. [4]

Table 2-1 Type of the original deviate from Munsell gradation

Type of original	Deviations from the Munsell gradation		
	Highlight	Midtone	Shadow
Normal / high key	No	Steeper	Flaltter
Normal / low key	Flaltter	Steeper	Flaltter
high key	Steeper	No	Flaltter
low key	Flaltter	Steeper	Steeper

High key original, the predominant tone range is in the highlights rather than in the midtone, the compression curve must be steeper in the highlights and flatter in the shadows.

Low key original, the shadows are dominant what is called a low key original, the tone reproduction must be steeper in the shadows. [5]

2.1.3.4 Shape of black printer

The black printer is a function to determine added black in the color reproduction. There are four reasons for using black as a replacement or suppliment: [6]

- To make the control of the other three colors less critical as to ink balance.
- To produce denser blacks and better shadow detail than the other three colors alone can produce.
- To substitute relatively inexpensive black ink for a part of the more costly colored inks.
- In high-speed wet printing, to avoid the piling up of several inks which do not print properly satisfactorily on top of each other.

The black printer can separate into 2 characters: [7]

- Skeleton black is only adding black onto the shadows. Usually it covers rarely exceeding 70%. The sample is UCR, under color removal, mean the partial replacement of chromatic ink in favor of black in the neutral area.

- Full black is starting in the highlight in order to stabilize the grey balance. The sample is GCR, grey component replacement, mean the replacement of the grey component of the chromatic ink by black in trichromatic areas.

There are two major differences between typical skeleton black and a full scale black. They are shown in Table 2-2. [8]

Table 2-2 Difference between a typical skeleton black and a full black

	Skeleton black	Full black
Maximun dot area in the black printer	60 - 80 %	75 - 90%
Maximun density of the original at where the black printer begins	0.6 - 1.0	0.2 - 0.4

2.1.3.5 Process inks

The three colors of a set of primaries define the grey balance condition: To reproduce a neutral grey, the three primaries must be superimposed in balance amounts. This relationship between neutral density and single color densities can be describes by three curves referred to as equivalent neutral density (END) curves. For given the set of process inks and a given set of densitometric filters the END curve can be considered to be constant. [8]

2.1.3.6 Substrate

The surface properties of substrate determine the attainable density in the overlap of all four primaries. In a strict sense, even the END curves depend on the substrate surface.

2.1.4 Output data and the its calculation

2.1.4.1 Tone reproduction

Tone reproduction is the relationship between levels of grey tone on the original and in the reproduction. The level of the grey tone is measured by density. If tone reproduction is ideal, the graph of this is 45^{o} straight line passing through the original. In fact, the typical tone reproduction is not because there is some loss in detail when the original reproduced. It shows in Figure 2-2. [9,10]

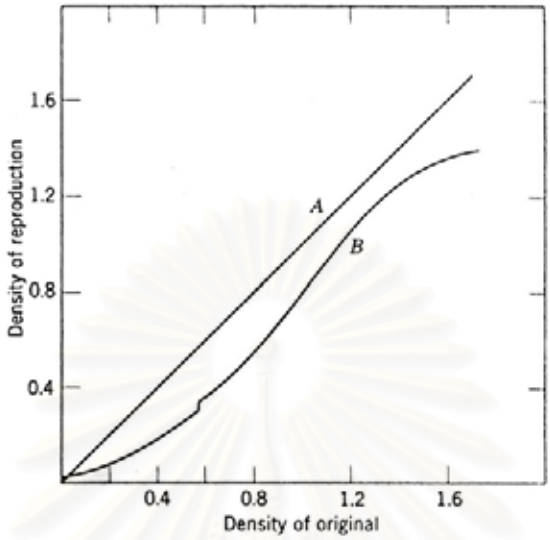

Figure 2-2 Comparison of the ideal tone reproduction (A) and the typical tone reproduction (B)

Tone reproduction curves refer to the relationship between the densities of the original and the density of four colors overlap. To plot this curve, first the attainable maximum density in the four colors overlap must be known.

Once the four colors curve is setup, it must be split off the three colors curve and the curve of the black printer in Figure 2-3. From both curves the starting and the end points are known as a result of the desired UCR and of the density of the original at which the black should begin. Basically, the shape of both curves can be selected freely. A very common way is to set up first the three colors curve by approaching the end point asymptotically. Then, the curve of black printer is a function of the three colors curve calculable by means of Yule-Nielsen equation (Eq.2). [11,12]

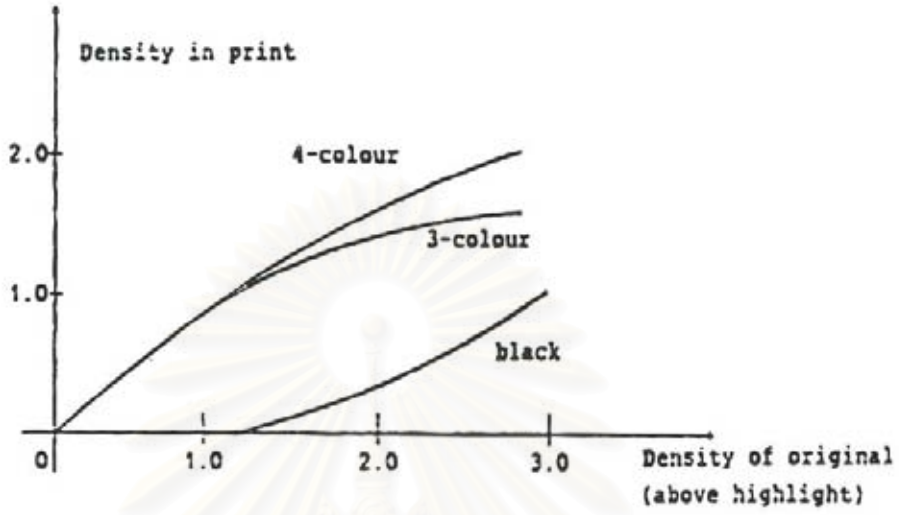

Figure 2-3 Reproduction curves where black is added to the 3 colors in the midtone and shadow.

$$D_{4C} = D_{3C} + D_K - D_{3C}.D_K.k \qquad \text{(Eq.2)}$$

When D_{4c} = the density of the 4 color printing

D_{3c} = the density of the 3 color printing

D_k = the density of genrated black

k = 0.30 for gloss coated paper

= 0.40 for supercalendered paper

= 0.45 for dull coated paper

= 0.55 for machine finished paper

2.1.4.2 Grey balance

Grey balance is ability of the process inks to reproduce neutral grey in printing process. It is necessity for the color reproduction because it is an important factor for color balance in the color reproduction. If the process inks are ideal ink, the equal amount of them can produce the neutral grey. But when the process inks are printed in equal amount, they cannot reproduce the proper grey. (13)

As the matter of fact, the process inks are not perfect color filters, with each color absorbing the proper third of the visible spectrum, the cyan, magenta and yellow would each absorb equal amounts of red, green and blue light, respectively. Equal amounts of cyan, magenta and yellow ink could then be printed, and the eye would receive equal amounts of red, green and blue light. This result would be seen as a neutral grey. In fact, this is not the case. Pigments that comprise the process colors are not as pure as the theory suggests. They possess some hue error as the following example indicates. Grey balance is not achieved by printing equally sized cyan, magenta and yellow dots. This combination would print a reddish brown. Rather, the size of yellow and magenta dots must be reduced in relation to cyan. The amount of the reduction is based on the contamination of the colors, so the dots absorb equal amounts of red, green and blue. (14)

This might be controlled in the color separation by unbalancing the three colors printing. The cyan printing is allowed to print larger dot sizes of ink while the magenta and yellow printing are reduced in their dot size by a controlled amount. A correct set of separations will be unbalanced to produce neutral values

throughout the value scale from highlights to shadows. At that point, grey balance is achieved. It shows in Figure 2-4. [15]

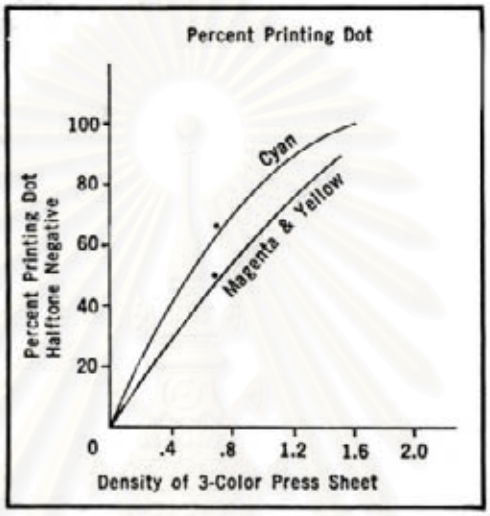

Figure 2-4 Relation of density of process color overprint and dot area of process inks.

To determine the amounts of cyan, magenta and yellow needed to print neutral grey, which is to find a specific Equivalent Neutral Density (END), the colorants to be printed must be evaluated in relation to the levels to be used during production.

One method to determine the grey balance is overprinting the color chart that contains the three process colors in different dot size on the same patch. It shows in Figure 2-5. The color chart should be covering all tone reproduction from highlight to shadow area. [16]

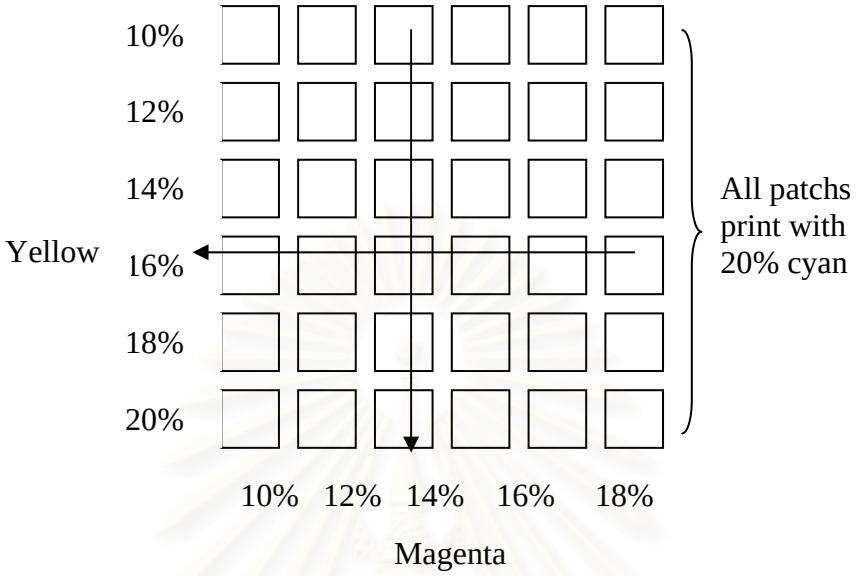

Figure 2-5 Halftone value for three colors grey test target

The methods used to evaluate grey patches of color chart overprint with the 3 process colors is:

- One method is visual evaluation. This is logical method to use human eye as the measuring device. Because it can detect grey accurately. The human is very sensitive to color difference in the bright illumination but it is poor sensitive in dimly light surround. The illumination level must be considered as a neutral color temperature. That is now established as 5000 Kelvin on the sample surface. For the grey reference, it uses a grey scale chart made to be perfect grey to comparison standard. The other reference can be used is the scale of grey step prints by black ink on the same substrate. [17]

- The other method used to determine the grey balance is measuring by the instrument, such as densitometer. The density value is not more different than 0.03 when it is measured through the red filter, the green filter and blue filter. Before using the densitometer it is necessary to calibrate and adjust the zero value on the same substrate. Since they affect the density value. [18,19]

Table 2-3 Sample of actual dot percentages required producing a neutral grey

Cyan	Magenta	Yellow	Black equivalent
5%	3%	3%	8%
10%	7%	7%	14%
30%	24%	24%	35%
70%	58%	58%	76%
90%	78%	78%	98%

2.1.4.3 Color reproduction

To achieve the correct color reproduction, it can be best controlled when hues are reproduced to matching exactly the process inks. In this case, each of the three primaries should only print in one color separation. Any dots in another separation must be considered as unwanted. Apart from this qualitative requirement, the dot area in the color separation can also be specified quantitatively for any density of the primary color in the original. For this the density of the original must be known in term of single color densities. This can be derived from the maximum grey density if it is split off in the three primaries according to the grey condition. Although the relationship between the grey density and the color density is

different for different film brands, an averaged set of curve can be used. From the maximum color density, which a primary color has in the original and in the print, the tone reproduction curve is calculated utilizing the same gradation type as in the grey scale. The resulting halftone densities of the print are then converted into dot area values of the halftone films. [20]

2.2 Literature Reviews

Schlaepfer [20] presented a system how specification can be derived and how they are applied to generate halftone separation for offset printing process. The required data is provided for halftone separation a computer program has been developed use as input data. They are solid tone density, paper, inks, density range of the original, UCR, characteristics of the black printer and tone reproduction. Then they are computed to generate halftone separation.

Jorgensen et al. [21] described the development of a computer program to this end and also presented the result of a trial in which the several repro companies were asked to prepare color separations for offset printing using their own internal standards and using data provided by computerized program.

William et al. [22] presented the method to improved color reproduction of newspaper by using the IFRA color test form. It composes of register mark, grey balance, halftone stepwedge, maximum four color density, solid color bar, slur target and vernier register mark. And it included two colors, three colors and four colors overprint. It characterized the reproduction stage and press by running the test form. It also checked the performance by densitometer or spectrophotometer. The results were used to adjust in the original processing stages necessary to optimize the color reproduction.

CHAPTER 3

EXPERIMENT

3.1 Material

3.1.1 Duplex clay coated paper 274 grams

3.1.2 UV process ink, Coates.

- Cyan color ink (LM 3-610)
- Magenta color ink (LM 3-310)
- Yellow color ink (LM 3-110)
- Black color ink (LM 3-810)

3.1.3 Asahi Photopolymer plate, AFP-HD 1.70 mm

3.1.4 Kodak grey scale chart

- The chart contains patches of grey scale

3.1.5 Original

- The original is used in this experiment is normal key including the grey scale patch for measuring density.

3.2 Apparatus

3.2.1 Gretag D 200 II Transmission densitometer

3.2.2 Gretag D 19 C Reflective densitometer

3.2.3 Anilaflex 700 Printing machine

3.2.4 Pentium III series personal computer

3.2.5 Linotype Hell Saphir scanner

3.2.6 Linotronic 260 Imagesetter

3.2.7 Software

- Microsoft Visual Basic 6.0 plus Services Pack 4
- Adobe Photoshop version 6.0
- Adobe Illustrator version 8.0

3.2.8 Standard light cabinet with the illuminant D_{50}

3.3 Procedure

3.3.1 Test form design

Test form is designed in a digital form by the Illustrator program. The test form composes of following elements:

- Halftone stepwedge 0 to 100% for the four process colors.
- Solid density of three colors (CMY) overprint.
- Solid density of four colors overprint.

- Grey component chart of three colors (CMY) overprint with different halftone dot area combination.

3.3.2 Characterization of printing machine

Test form is separated into four color separation negative films (CMYK) and making the photopolymer plates from them. Then they are used to print on the printing machine according to the standard printing procedure of the factory. The duplex paper is used as the substrate.

The following data are measured and comparison are made:

- Digital dot area vs. dot area on color separation negative films.
- Dot area on films vs. dot area on print.
- Tone reproduction curve of each process color
- Dot gain curve of each process color

Grey component of the print is found by visually with Kodak grey scale chart in the standard viewing condition D_{50}. Then the densities of the selected patches are measured with densitometer (visual filter). The END, Equivalent Neutral Density, curve is report.

3.3.3 Determination dot area for color separation

3.3.3.1 Determination the density of predicted print

The density of grey patches in the original (D_o) is measured by reflective densitometer. The predicted density (D_r) is calculated by using Munsell gradation equation (Eq.1). Then the relation of density of the original and density of the predicted print are shown.

$$y = 2\left\{1 - \log\left[10^{1-\frac{y_E}{2}} + \frac{10 - 10^{1-\frac{y_E}{2}}}{10 - 10^{1-\frac{x_E}{2}}}\left(10^{1-\frac{X}{2}} - 10^{1-\frac{x_E}{2}}\right)\right]\right\} \qquad \text{(Eq.1)}$$

When y = density of print

y_E = maximum density of print

x = density of original

x_E = maximum density of original

3.3.3.2 Determination the black printer

The skeleton black is chosen for generating black printer in this experiment. The density of original 0.80 is used to the starting point of the black printer. The end point, the maximum density of the black printer, is calculated from by Yule-Nielsen equation (Eq.2). This research is not concern the k factor of the

surface properties of paper in this equation. Due to the paper in this experiment is the paper with gloss coated, so the k factor 0.30 is used according to the surface properties of paper in the URGA Project 70 research. [23] The density of four colors (D_{4c}) is substituted with the maximum predicted density. And the density of three colors (D_{3c}) is substituted with maximum solid density of three colors from test form printing. The perpendicular line is used to join the starting point and the end point. The black printer curve is shown.

$$D_{4C} = D_{3C} + D_K - D_{3C}.D_K.k \qquad \text{(Eq.2)}$$

When D_{4c} = density of four colors

D_{3c} = density of three colors

D_k = density of black

k is a constant depending on surface properties of the paper.

k = 0.30 for gloss coated paper

k = 0.40 for supercalendered paper

k = 0.45 for dull coated paper

k = 0.55 for machine finished paper

3.3.3.3 Determination the density of three colors

The predicted density of three colors is equal to the density of four colors in the highlight area until the black printer starts to print. The density of three colors is found by using Yule-Nielsen equation (Eq.2) and the k factor 0.30 is used for the surface properties of paper. The density of four colors (D_{4c}) is substituted with the density of four colors. The density of black printer (D_k) is found by the black printer curve. Draw a perpendicular line from any point of the predicted density of four colors across to the X-axis. The black density is obtained from the crossing of the black printer curve. Then the density of three colors is calculated from the Eq.2. The curve of the density of three colors is shown.

3.3.3.4 Determination the dot area for color separation negative films

The dot area for color separation of each color is obtained by comparison the curve of the three colors density and the curve of the black printer density with the grey balance curve and the tone reproduction curve of process color in 3.3.2. The relation of original density and dot area for color separation is shown.

3.3.4 The color separation computing program

The Microsoft Visual Basic is used for programming. The program contains tables for input data as below:

- The input data of digital file
- Dot area on film
- Dot area on print
- Density on print
- Grey balance
- The maximum solid density of three colors (CMY)
- The maximum solid density of four colors (CMYK)
- The density of grey patches in the original

After filling the data in all table, it calculates and shows the relation the dot area of each process color (CMYK) for using color separation.

3.3.5 The production test print

The original is separated by using dot area data from the color separation program in 3.3.4. The Photoshop program is used to adjust the original following the data. Then the original is printed according to the standard printing procedure of the factory. The grey patches in the original are measured to obtain the tone reproduction of production test print. It compares with the tone reproduction of the predicted print.

Besides the tone reproduction of each color is measured to compare with the tone reproduction of characterization test print in 3.3.2.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Test form design

Test form is designed as shown in Figure 4-1. It contains the standard digital image and the standard target for specifying the printing parameter.

Figure 4-1 Test form

4.2 Characterization of the printing system

The relation of input data and dot area on film is shown in Figure 4-2. The curve is almost a 45^{o} straight line passing through the origin. This relation can use to calibration the imagesetter. Besides it can be used to checking the film making condition.

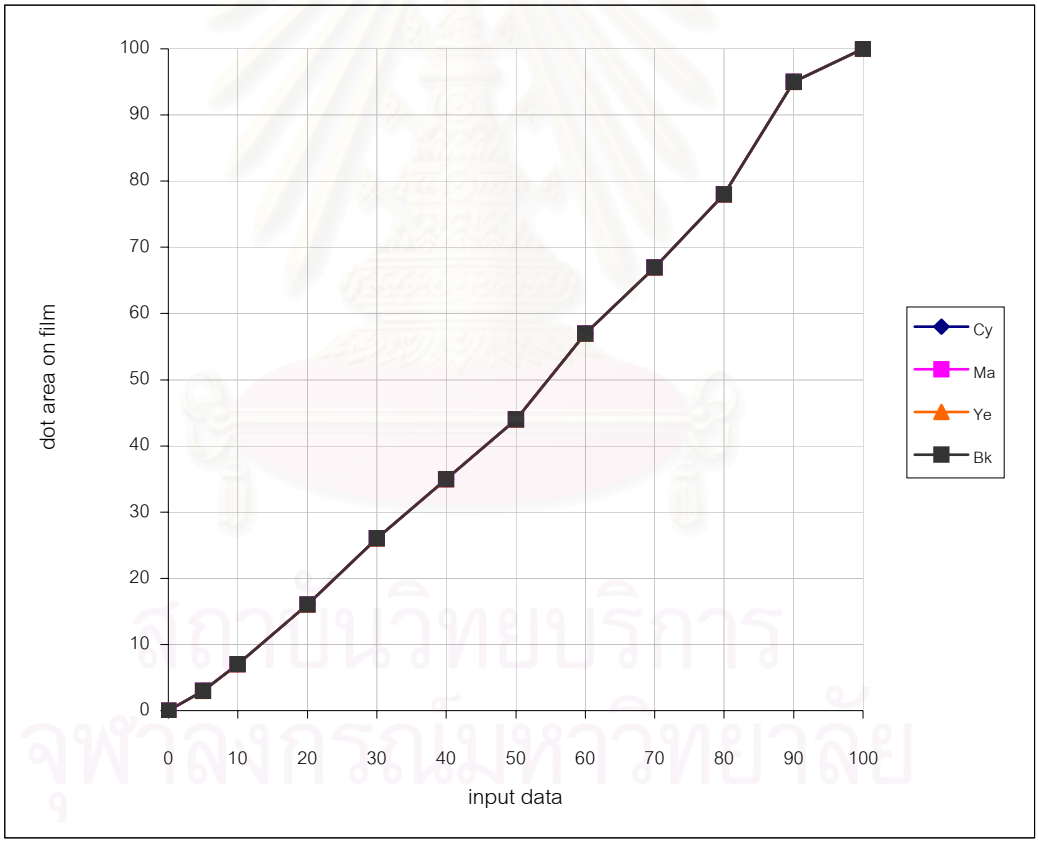

Figure 4-2 The relation of input data vs. dot area on film

The relation of dot area on film and dot area on print of each process color (CMYK) is shown in Figure 4-3. This graph shows characteristic of tone reproduction of each color in term of dot area. Normally each process color curve is not straight line since dot gain effects.

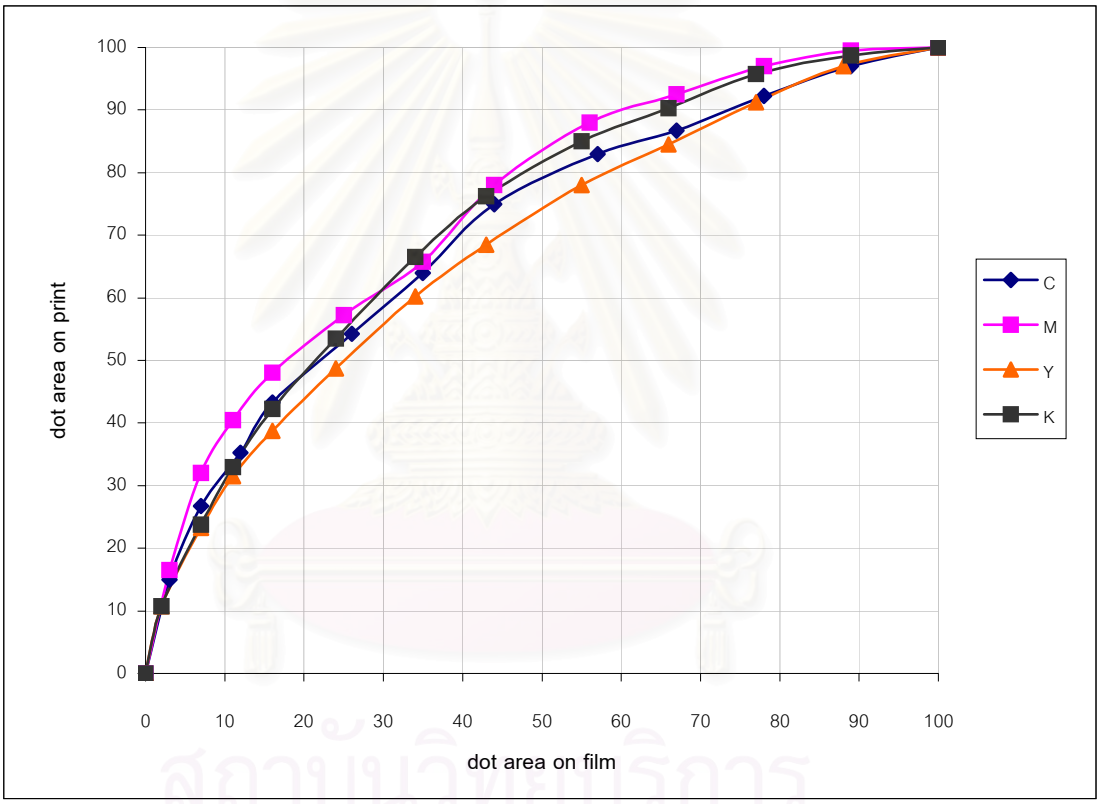

Figure 4-3 The relation of dot area on film vs. dot area on print

The comparison of dot area on print and dot area on film were made. The dot gain curve is shown in Figure 4-4. The dot gain curve of each color is not equal according to the characteristic of each color is different. In this research, magenta has the largest dot gain regarding to highlight and midtone area, meanwhile yellow has the lowest dot gain.

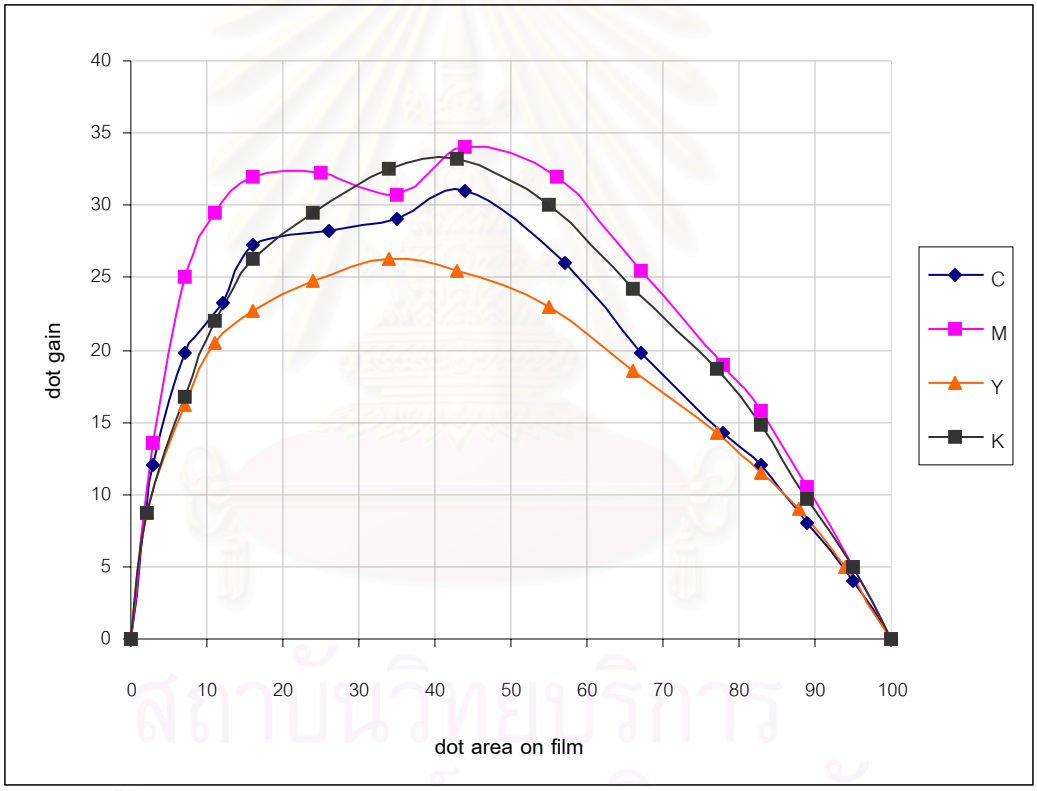

Figure 4-4 Dot gain of CMYK

The tone reproduction curve of each process color is shown in Figure 4-5. Black has the largest solid density (D_k = 1.59) and yellow has the lowest solid density (D_y = 1.15). The solid density of magenta and cyan are same (D_c = D_m = 1.38). In highlight area, magenta has more density than the others.

The results of this experiment are used to adjust the printing condition for the production run to obtain the consistency reproduction.

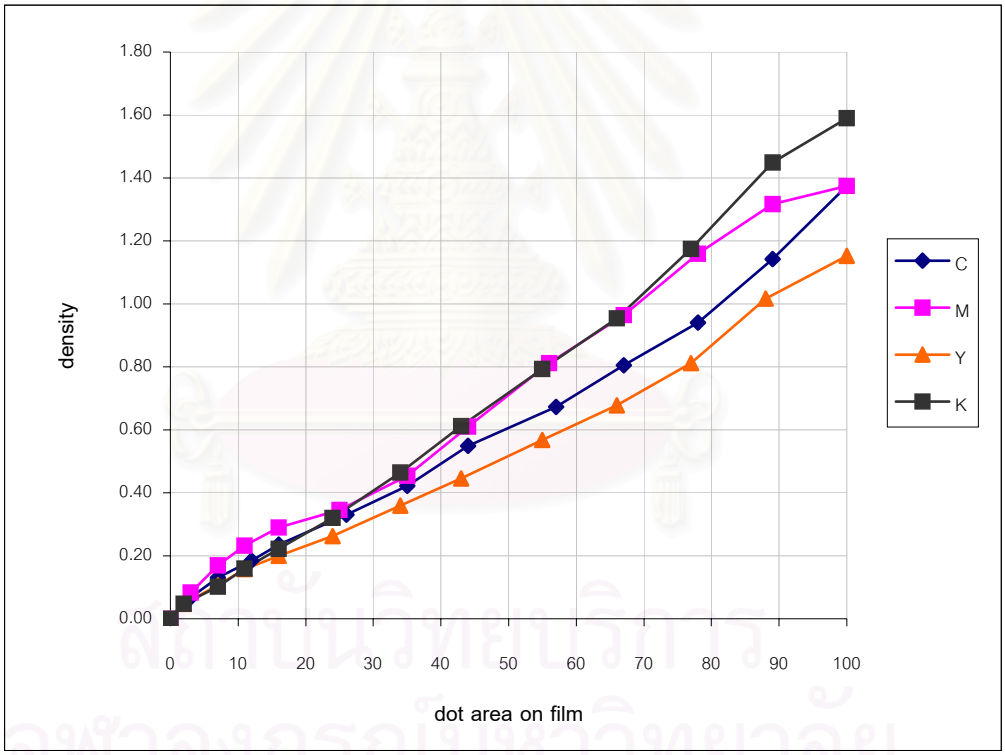

Figure 4-5 Tone reproduction curve of CMYK

Grey balance curve shows the relation of dot area on film of three colors (CMY) producing grey and the density of three colors overprint. It is shown in Figure 4-6. The dot areas of magenta and yellow are almost equality to producing greys. But the cyan required larger dot area than magenta and yellow especially in midtone and shadow area.

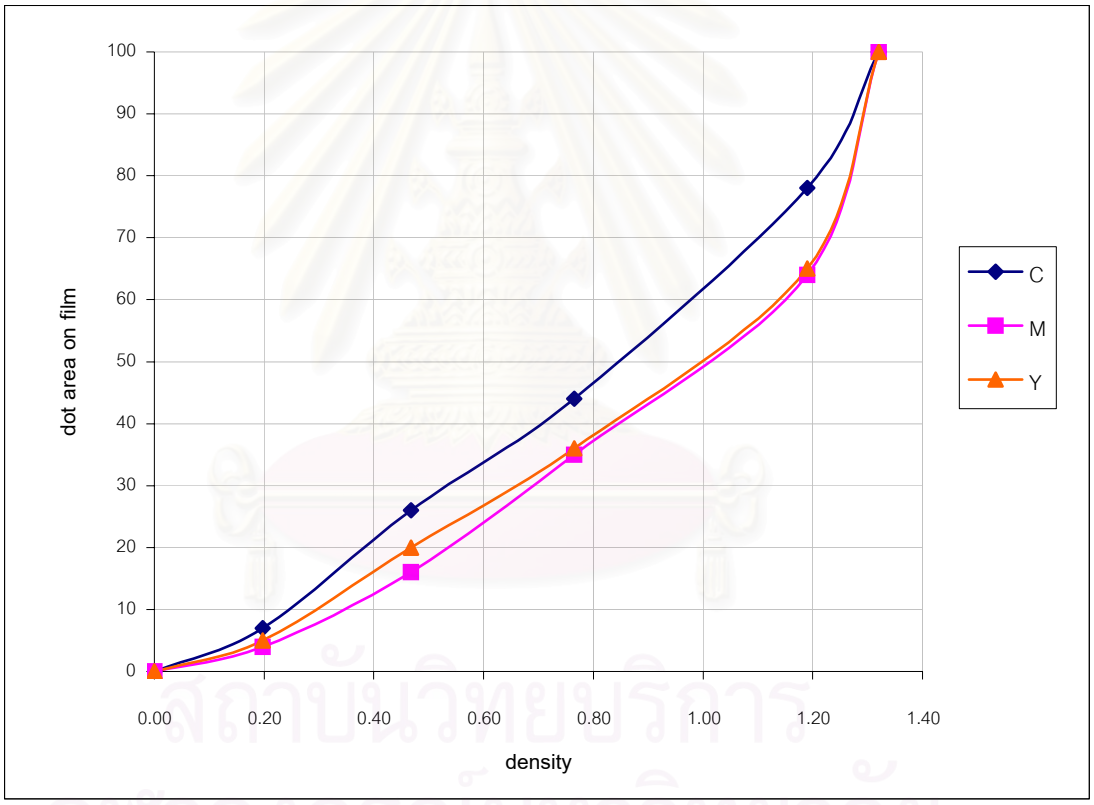

Figure 4-6 Grey balance curve

4.3 Determination dot area for color separations

4.3.1 Determination the density of predicted print

The density of grey patches of the original and the predicted density, which calculates from Munsell gradation equation, are shown in Table 4-1. The tone reproduction of predicted print is shown in Figure 4-7. The graph is almost straight line according to normal key original.

Table 4-1 Density of original (D_o) and density of predicted print (D_r)

Do	Dr
0.00	0.00
0.09	0.09
0.14	0.15
0.30	0.31
0.54	0.57
1.01	1.09
1.51	1.69
1.75	2.01

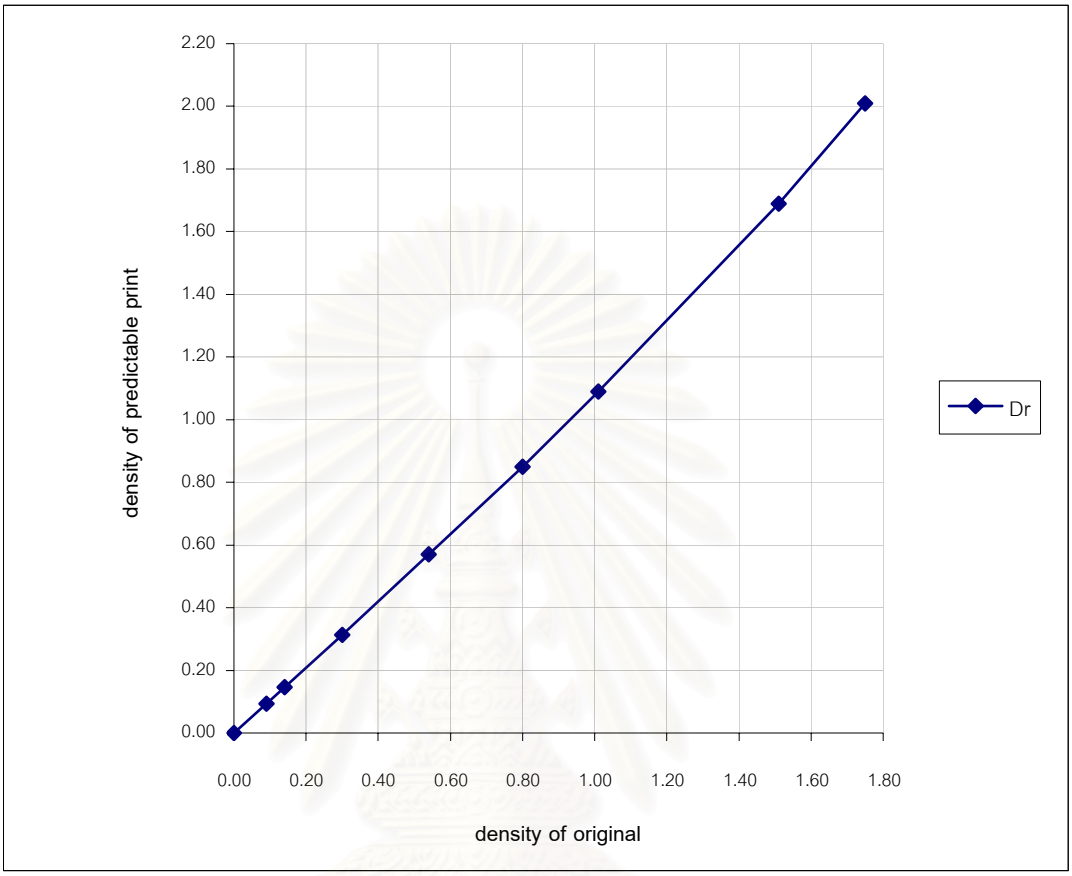

Figure 4-7 Tone reproduction of predicted print

4.3.2 Determination the black printer

The constructed black printer curve is shown in Figure 4-8. The density of the original 0.8 is the starting point of the black printer. The predicted density 1.14 is the end point of the black printer calculated from Yule-Nielsen equation (Eq.2).

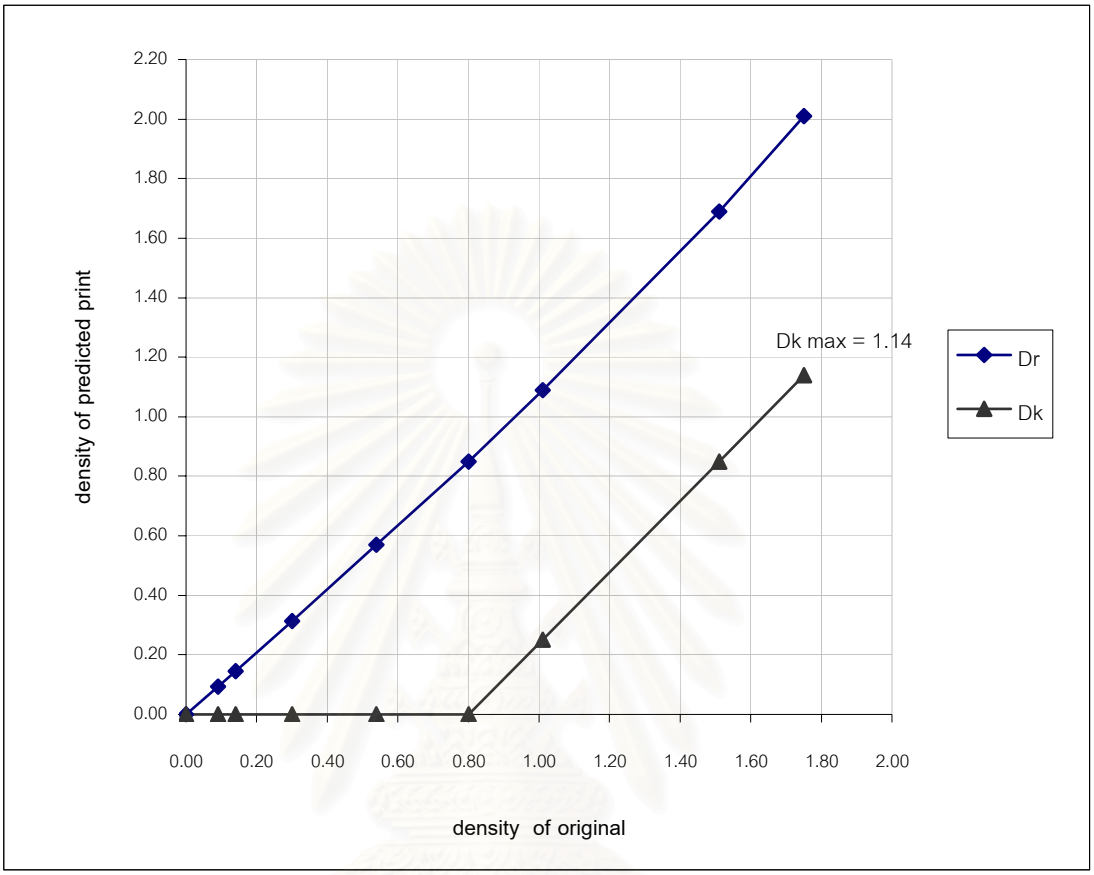

Figure 4-8 The black printer curve

4.3.3 Determination the density of three colors

The density of three colors is obtained from calculating any point of the predicted density and the black density by using Yule-Nielsen equation (Eq.2). It is shown in Figure 4-9.

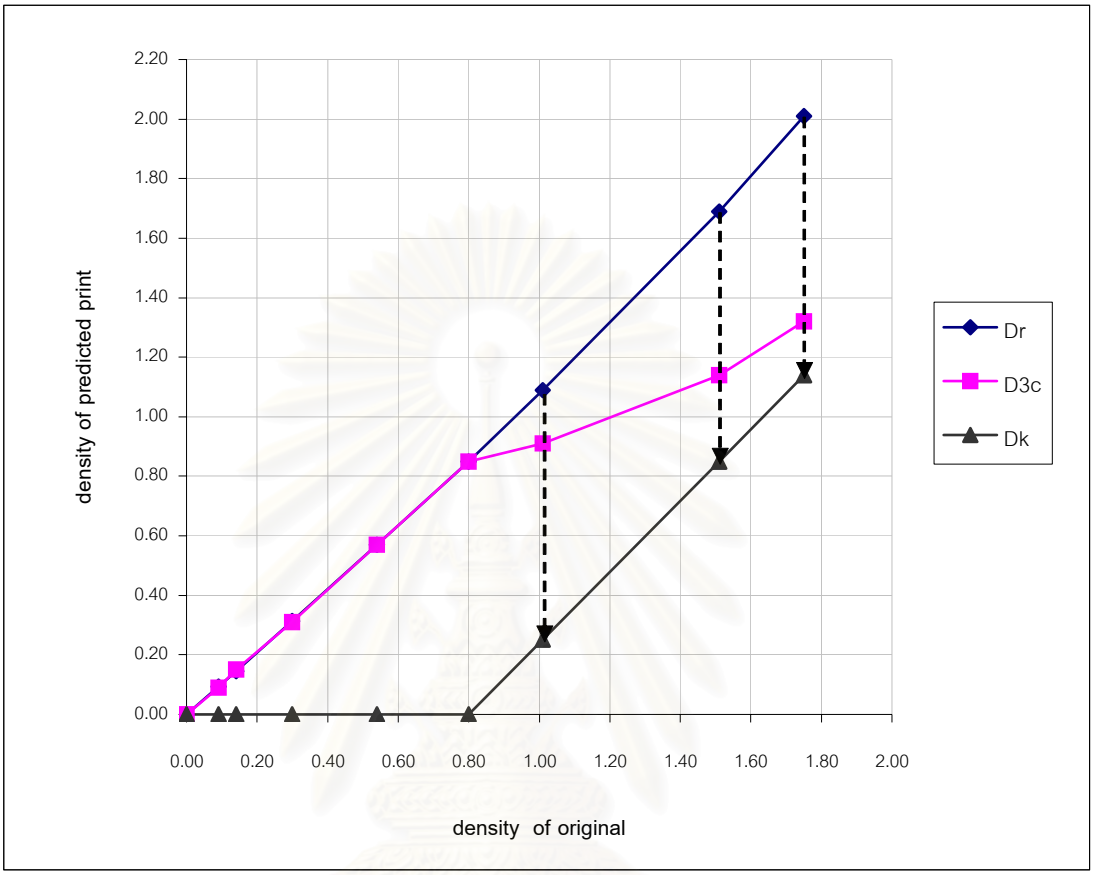

Figure 4-9 The relation of D_{4c}, D_{3c} and D_k

4.3.4 Determination the dot area for color separation negative films

The dot area for color separation negative films is shown in Figure 4-10. It is obtained from the method in 3.3.4

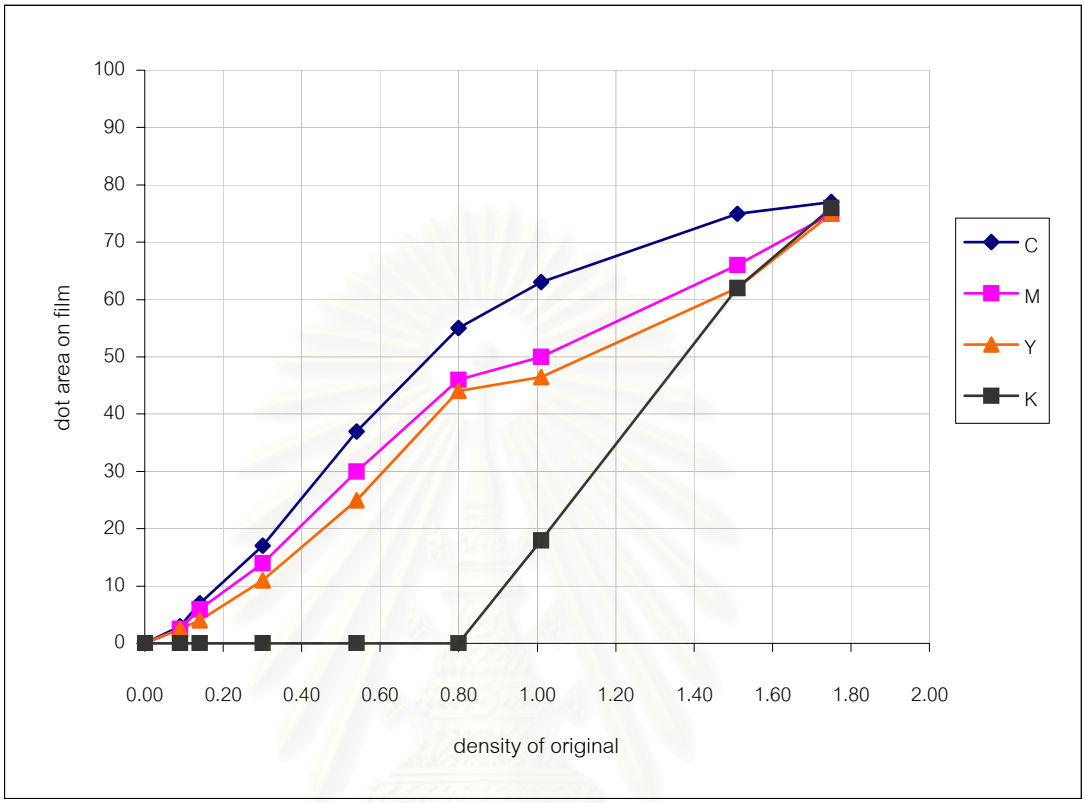

Figure 4-10 The relation of the original density and dot area for color separation

4.4 The color separation computing program

The tables for inputting the characterization data are shown in Figure 4-11 to Figure 4-14. After filling the data, the graph of dot area for color separation negative films is shown in Figure 4-15.

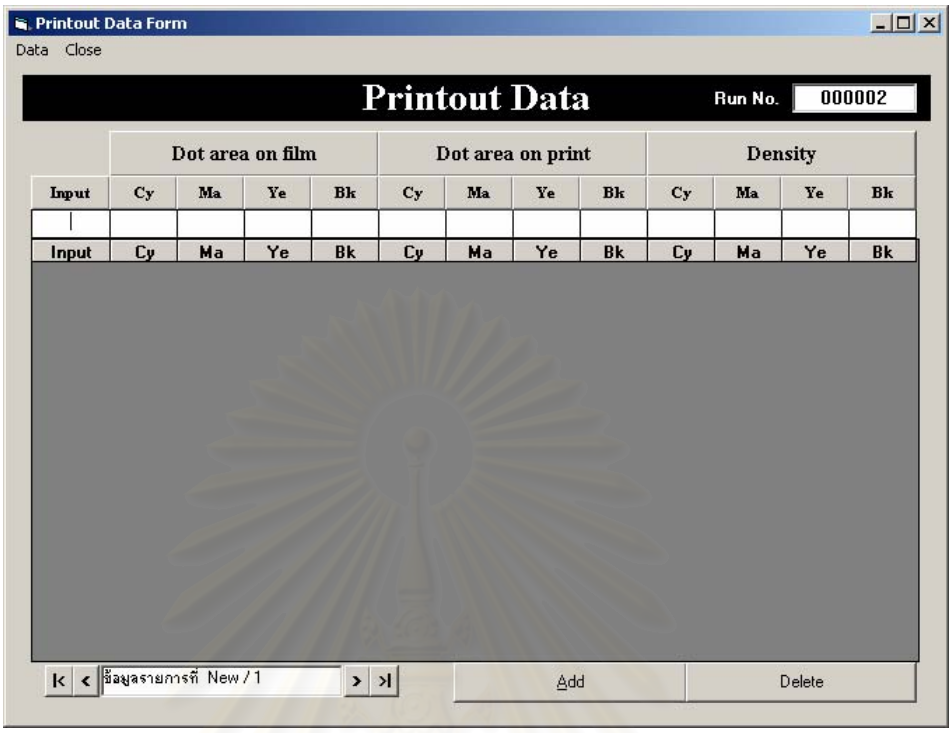

Figure 4-11The table for input digital data, dot area on film, dot area on print and density on print

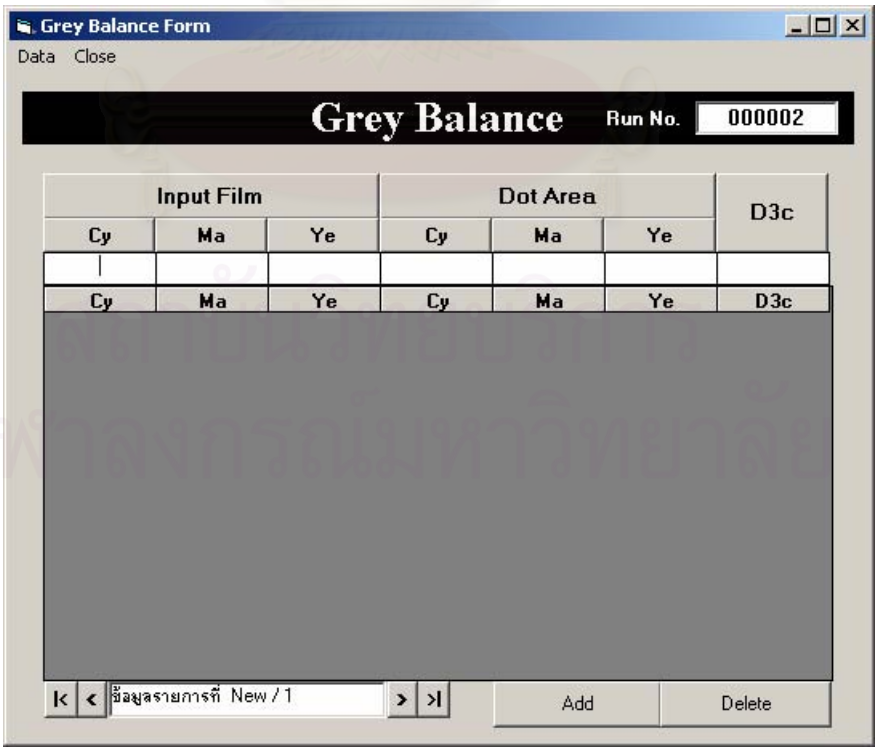

Figure 4-12The table for input grey balance

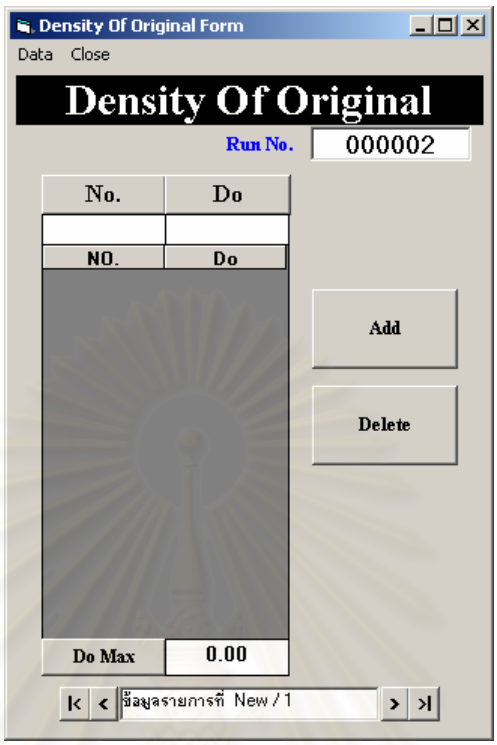

Figure 4-13The table for input the original density

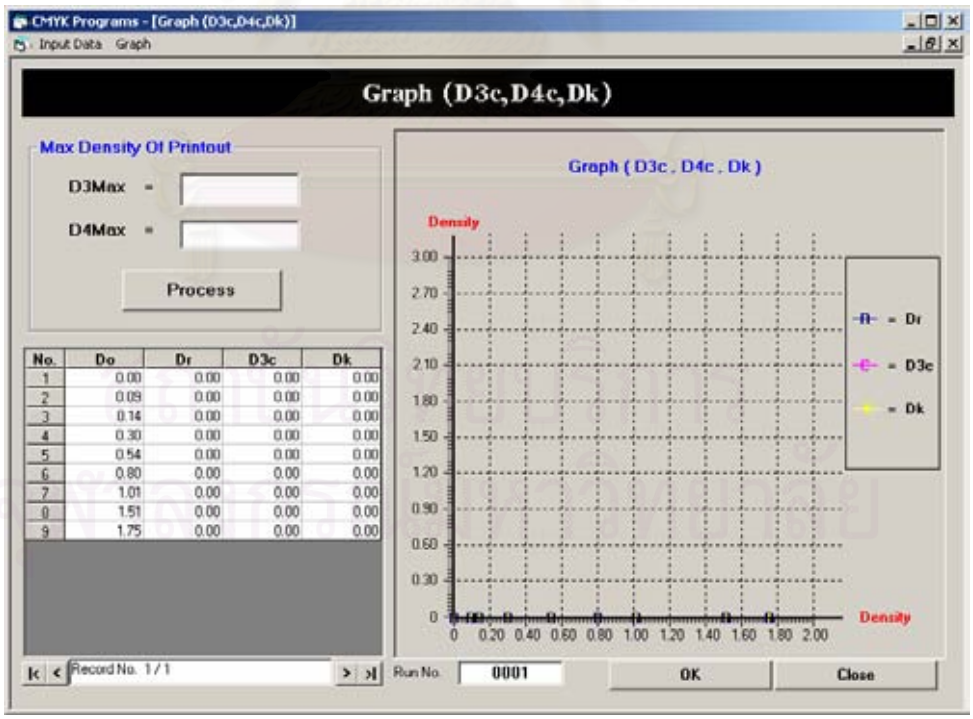

Figure 4-14The table for input maximum solid density of four colors overprint (D_4max) and three colors overprint (D_3max)

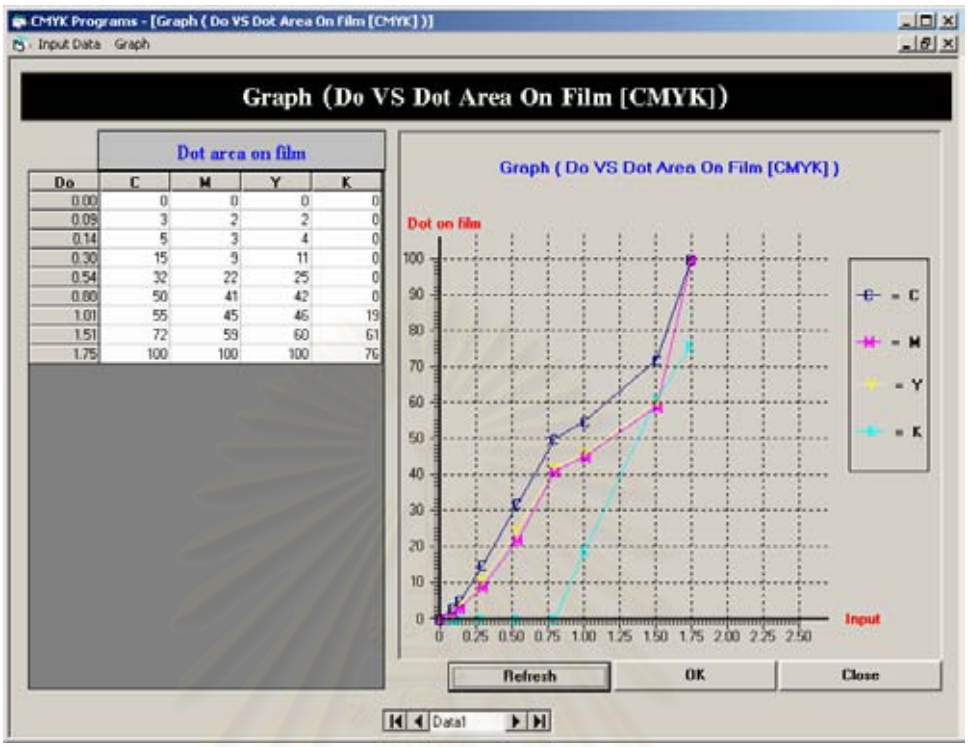

Figure 4-15 The dot area for color separation negative films

4.5 Result of production test print

The comparison of the tone reproduction of production test print and predicted print are shown in Figure 4-16. The tolerance density is chosen for convenient evaluation. The tolerance 0.1 is used for this research regarding to standard printing procedure of the factory. The character of production test print is similar to the curve of predicted print. The density of production test print is higher the predicted print in highlight area to midtone area. But the density of production test print is lower than the predicted print in the shadow area.

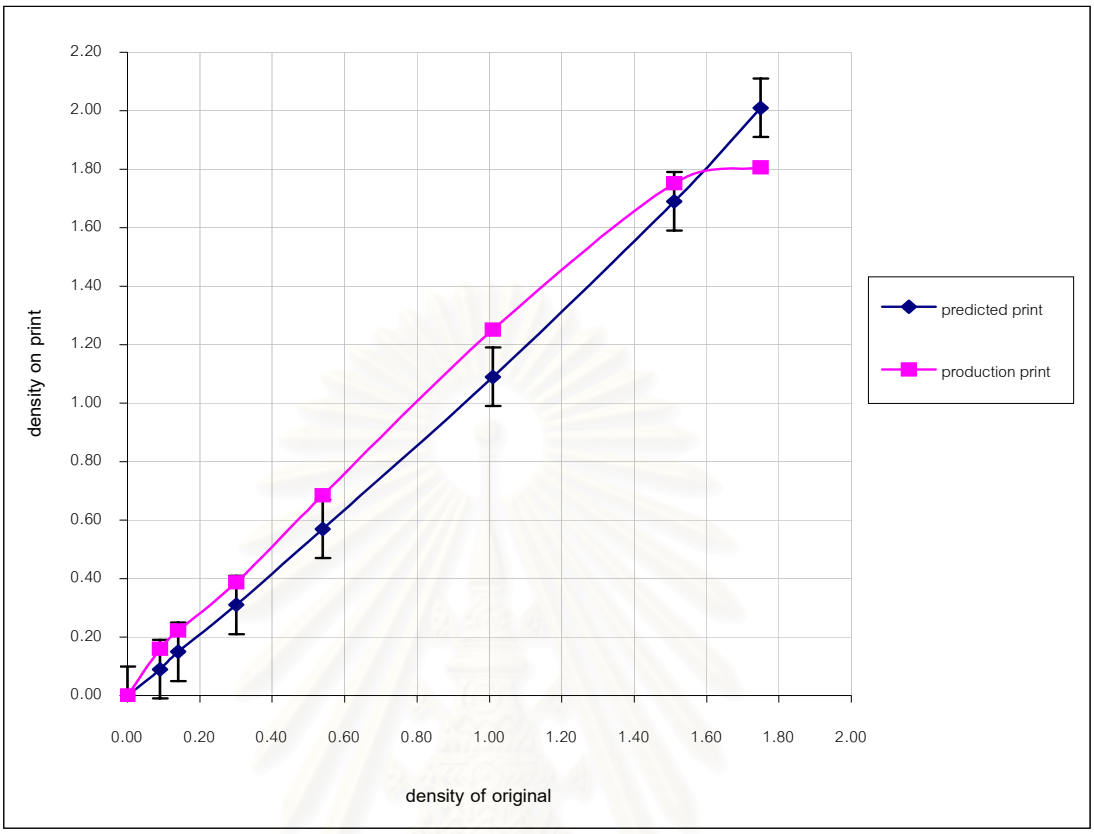

Figure 4-16The tone reproduction of production test print and predicted print

The comparison of cyan density between production test print and characterization test print are shown in Figure 4-17. The density of production test print is lower than the density of characterization test print from highlight area to shadow area. The difference of maximum solid density of both prints is 0.32.

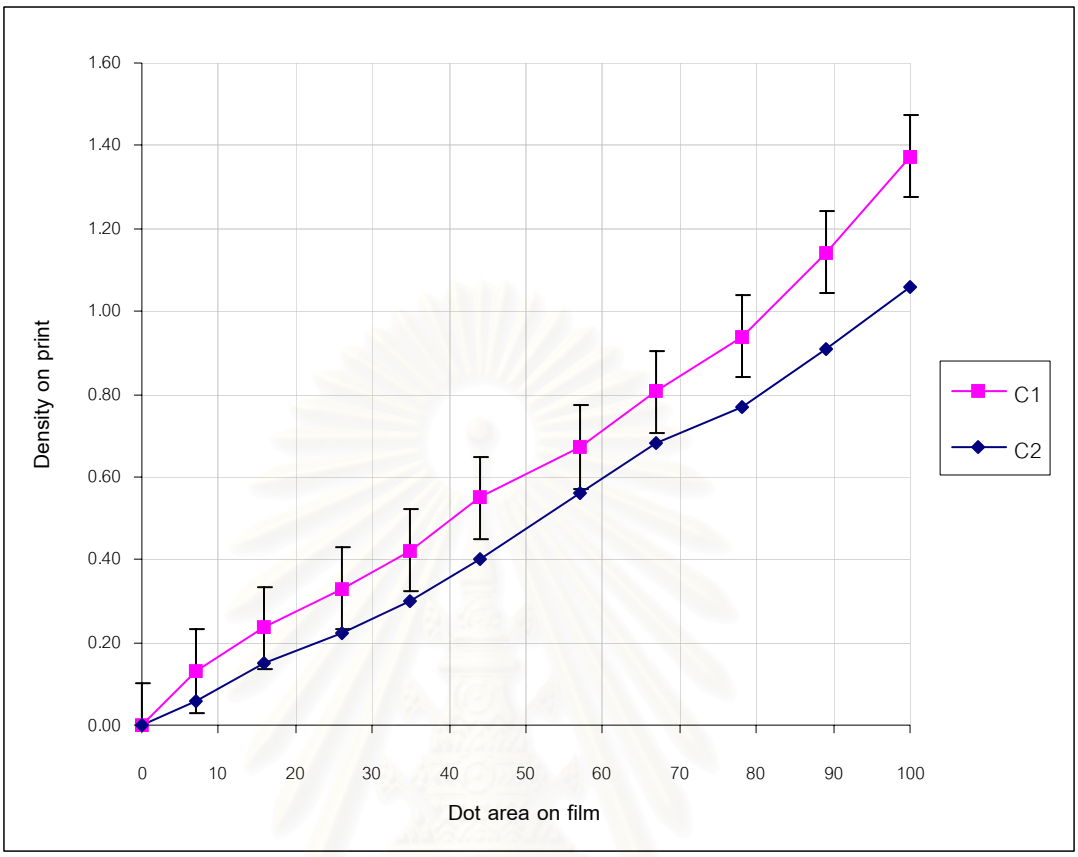

Figure 4-17 Tone reproduction curves of the production test print (C2) and the characterization test print (C1) of cyan

The comparison of magenta density between production test print and characterization test print are shown in Figure 4-18. The density of production test print is lower than the density of characterization test print from highlight area to shadow area. The difference of maximum solid density of both prints is 0.12.

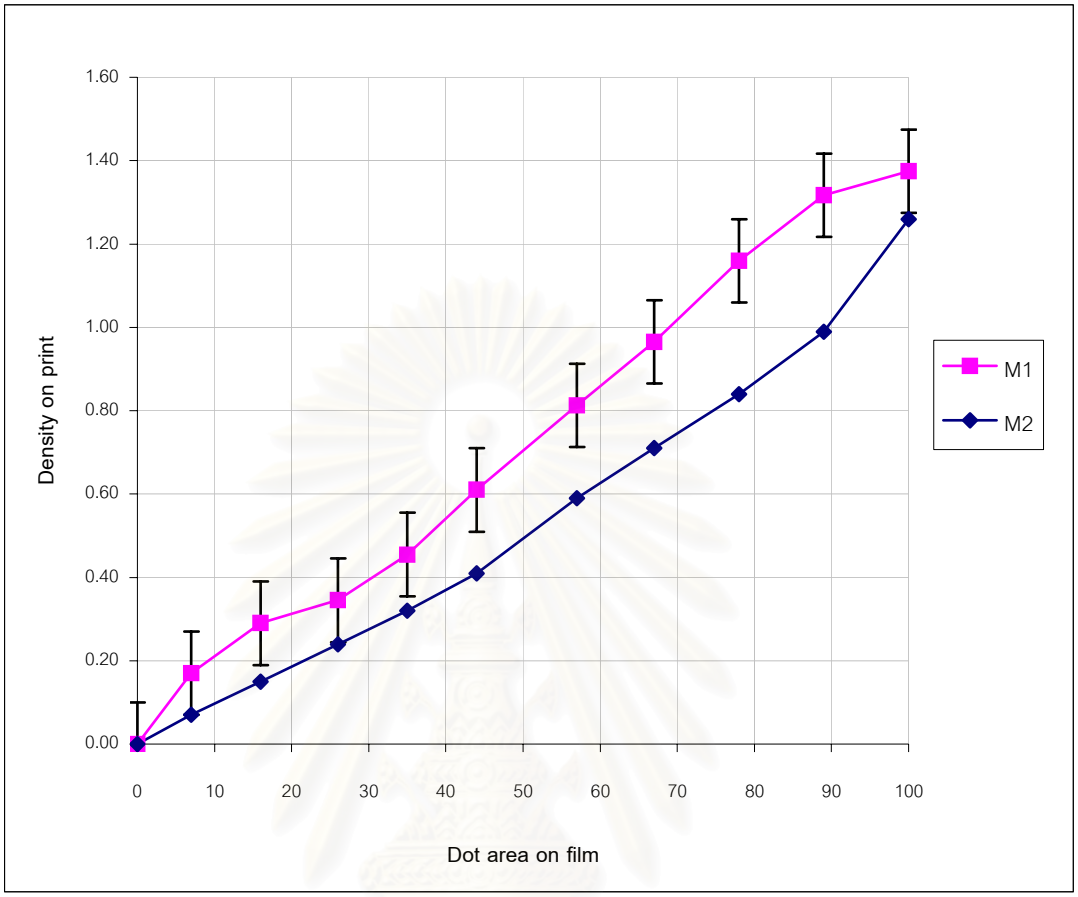

Figure 4-18 Tone reproduction curves of the production test print (M2) and the characterization test print (M1) of magenta

The comparison of yellow density between production test print and characterization test print are shown in Figure 4-19. The density of production test print is nearly the density of characterization test print. The difference of maximum solid density of both prints is 0.06.

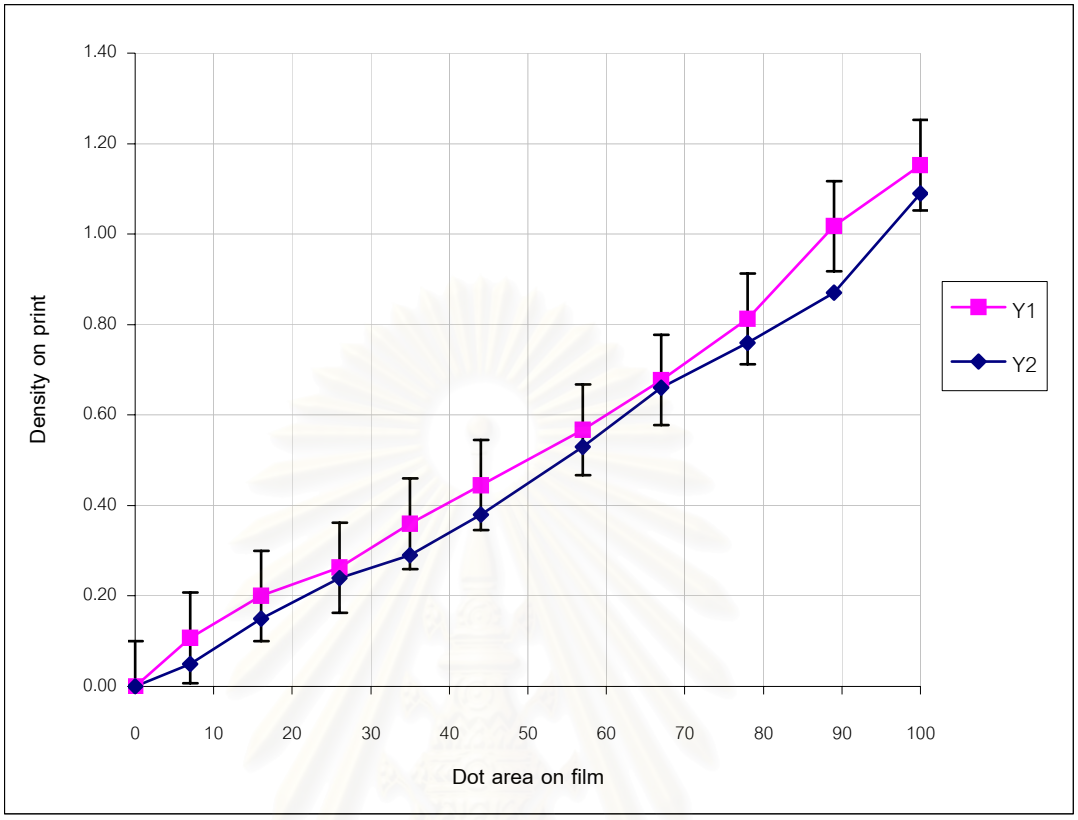

Figure 4-19 Tone reproduction curves of the production test print (Y2) and the characterization test print (Y1) of yellow

The comparison of black density between production test print and characterization test print are shown in Figure 4-20. The density of production test print is nearly the density of characterization test print. But it is lower in midtone area to shadow area. The maximum solid density of both prints is equal.

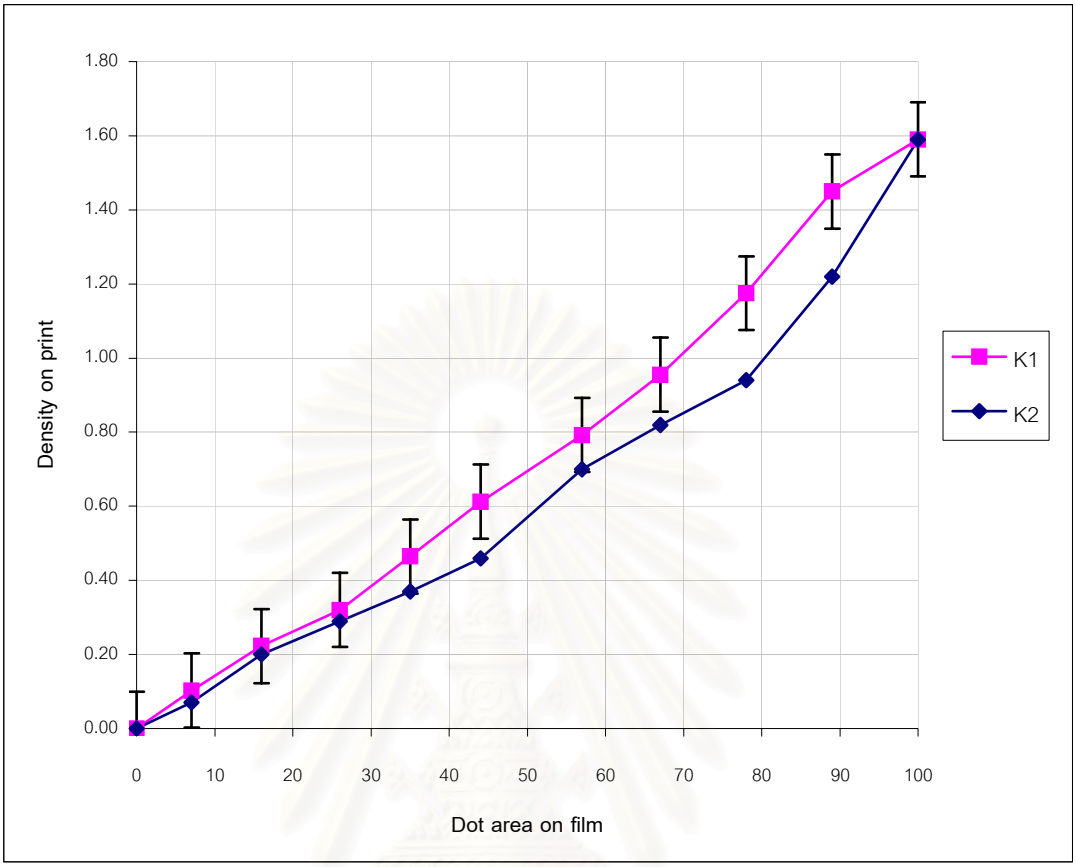

Figure 4-20 Tone reproduction curves of the production test print (K2) and the characterization test print (K1) of black

CHAPTER 5

CONCLUSION AND SUGGESTION

5.1 Conclusion

From the results and discussion, it can be concluded that the printing results of flexographic printing is predictable by the color separation computing program based on visually uniform compression of Munsell gradation. The tone reproduction curve of production test print is similar to the curve of predicted print. Some colors are closely predictable. But for some colors are not closely predictable, they can be adjusted to close.

The material quality control plays the important role because it affects the printing results. In this experiment, the batch lot of cyan and magenta inks used for the characterization process and production test print process are different. They affect the tone reproduction of cyan and magenta production print to deviate from that of the characterization process. Furthermore they also reduce the maximum density of four colors. For yellow and black, the batch lot of both colors is same. The tone reproduction of both colors is closely related.

5.2 Suggestion

This research tests only the original with normal key. The printing result is acceptable in the production. To confirm this program, the originals with the high key and low key should be studied.

In this research, the linear skeleton black is chosen for the black printer. However the other form of the skeleton black should be investigated too.

Normally, the relief printing is popularly used the full scale black. Because it believes that the full scale black produce smoother. Therefore the black printer with full scale is worth to studying.

REFERENCES

1. Parisi, J.G. *Flexography Principles and Practices*. 4th ed. New York: FTA, 1991, pp. 4-5, 407-408.
2. Schlaepfer, K. *Standardization of Color Reproduction: A Program for Computing Specifications for Halftone Separation*, UGRA Project 70, November 1977, pp. 3-4.
3. Parisi, J.G. *Flexography Principles and Practices*. 4th ed. New York: FTA, 1991, pp. 411, 422.
4. Schlaepfer, K. *Standardization of Color Reproduction: A Program for Computing Specifications for Halftone Separation*, UGRA Project 70, November 1977, pp. 5-6.
5. Schlaepfer, K. *Standardization in the Pre-press Area*, Proceedings for International Conference on Standardization of Color Reproduction and Printing, The Imperial Hotel, Bangkok, June 6-8, 1988, pp. 50-51.
6. Yule, J.A.C. *Principles of Color Reproduction*. New York: John Wiley & Son, 1967, pp. 282-283.
7. Schlaepfer, K. *Standardization in the Pre-press Area*, Proceedings for International Conference on Standardization of Color Reproduction and Printing, The Imperial Hotel, Bangkok, June 6-8, 1988, pp. 49-50.
8. Schlaepfer, K. *Standardization of Color Reproduction: A Program for Computing Specifications for Halftone Separation*, UGRA Project 70, November 1977, pp. 4-5.

9. Mortimer, A. *Colour Reproduction in Digital Age*. Survey: Pira Internatioinal, 1998, p. 110.

10. Yule, J.A.C. *Principles of Color Reproduction*. New York: John Wiley & Son, 1967, pp. 85-86.

11. Jorgensen, H.O. et.al. *An Approach to the Development of Standard Procedures in Scanner Operation.* Proceedings for International Conference on Standardization of Color Reproduction and Printing, The Imperial Hotel, Bangkok, June 6-8, 1988, p. 65.

12. Schlaepfer, K. *Standardization of Color Reproduction: A Program for Computing Specifications for Halftone Separation,* UGRA Project 70, November 1977, pp. 4,11.

13. Parisi, J.G. *Flexography Principles and Practices*. 4th ed. New York: FTA, 1991, p. 412.

14. Randazzo, J.A. et.al. *FIRST*. New York: FTA , 1997, pp. 33-34.

15. Southworth, M. *Color Separation Technique*, 2nd ed. New York: Graphic Arts Publishing, 1979, p. 190.

16. Mortimer, A. *Colour Reproduction in Digital Age*. Survey: Pira Internatioinal, 1998, pp. 270-274.

17. Southworth, M. *Color Separation Technique*, 2nd ed. New York: Graphic Arts Publishing, 1979, pp. 185-186.

18. Yule, J.A.C. *Principles of Color Reproduction*. New York: John Wiley & Son, 1967, pp. 104-107.

19. Randazzo, J.A. et.al. *FIRST*. New York: FTA, 1997, p. 34.

20. Schlaepfer, K. *Standardization of Color Reproduction: A Program for Computing Specifications for Halftone Separation*. UGRA Project 70, November 1977, pp. 1-17.

21. Jorgensen, H.O. et.al. *An Approach to the Development of Standard Procedures in Scanner Operation.* Proceedings for International Conference on Standardization of Color Reproduction and Printing, The Imperial Hotel, Bangkok, June 6-8, 1988, pp. 59-76.

22. William, A. et.al. *Newspaper Printing-On the Way to Standardization Part 2: Improvement of color print quality*. IFRA Special Report 2.22, IFRA, Darmstadt, 1989, pp. 1-56.

23. Schlaepfer, K. *Standardization of Color Reproduction: A Program for Computing Specifications for Halftone Separation*. UGRA Project 70, November 1977, p. 4.

APPENDICES

APPENDIX A

1.1 Results of the characterization of the printing system

Table A-1 Digital input data vs. Dot area on film

Input	Dot area on film			
	C	M	Y	K
0	0	0	0	0
5	3	3	3	3
10	7	7	7	7
20	16	16	16	16
30	26	26	26	26
40	35	35	35	35
50	44	44	44	44
60	57	57	57	57
70	67	67	67	67
80	78	78	78	78
90	95	95	95	95
100	100	100	100	100

Table A-2 Dot are on film vs. Dot area on print

dot area on film				dot area on print			
C	M	Y	K	C	M	Y	K
0	0	0	0	0	0	0	0
3	3	3	3	15	17	11	11
7	7	7	7	27	32	23	24
16	16	16	16	43	48	39	42
26	26	26	26	54	57	49	54
35	35	35	35	64	66	60	67
44	44	44	44	75	78	69	76
57	57	57	57	83	88	78	85
67	67	67	67	87	93	85	90
78	78	78	78	92	97	91	96
89	89	89	89	97	100	97	99
100	100	100	100	100	100	100	100

Table A-3 Dot area on film vs. Dot gain

dot area on film				dot gain			
C	M	Y	K	C	M	Y	K
0	0	0	0	0	0	0	0
3	3	3	3	12	14	8	8
7	7	7	7	20	25	16	17
16	16	16	16	27	32	23	26
26	26	26	26	28	31	23	28
35	35	35	35	29	31	25	32
44	44	44	44	31	34	25	32
57	57	57	57	26	31	21	28
67	67	67	67	20	26	18	23
78	78	78	78	14	19	13	18
89	89	89	89	8	11	8	10
100	100	100	100	0	0	0	0

Table A-4 Dot area on film vs. Density on print

Dot area on film				Density on print			
C	M	Y	K	C	M	Y	K
0	0	0	0	0.00	0.00	0.00	0.00
3	3	2	2	0.07	0.08	0.05	0.05
7	7	7	7	0.13	0.17	0.11	0.10
16	16	16	16	0.24	0.29	0.20	0.22
26	26	26	26	0.33	0.35	0.26	0.32
35	35	35	35	0.42	0.46	0.36	0.47
44	44	44	44	0.55	0.61	0.45	0.61
57	57	57	57	0.67	0.81	0.57	0.79
67	67	67	67	0.81	0.97	0.68	0.96
78	78	78	78	0.94	1.16	0.81	1.18
89	89	88	89	1.14	1.32	1.02	1.45
100	100	100	100	1.38	1.38	1.15	1.59

Table A-5 Grey balance

setting for grey			dot area on film of CMY			D_{3C}
C	M	Y	C	M	Y	
0	0	0	0	0	0	0.00
10	6	8	7	4	5	0.20
30	20	24	26	16	20	0.47
50	40	41	44	35	36	0.77
80	67	68	78	64	65	1.19
100	100	100	100	100	100	1.32

1.2 Determination dot area for color separations

Table A-6 The relation of Do, D4c, D3c and Dk

Do	Dr	D_{3C}	D_K
0.00	0.00	0.00	0.00
0.09	0.09	0.09	0.00
0.14	0.15	0.15	0.00
0.30	0.31	0.31	0.00
0.54	0.57	0.57	0.00
0.80	0.85	0.85	0.00
1.01	1.09	0.91	0.25
1.51	1.69	1.14	0.85
1.75	2.01	1.32	1.14

Table A-7 The relation of Do and dot area on films of CMYK

Do	C	Y	M	K
0.00	0	0	0	0
0.09	3	3	3	0
0.14	7	4	6	0
0.30	17	11	14	0
0.54	37	25	30	0
0.80	55	44	46	0
1.01	63	47	50	18
1.51	75	62	66	62
1.75	100	100	100	76

1.3 Results of production test print

Table A-8 Tone reproduction of production test print and predicted print

original	predict	production
0.00	0.00	0.00
0.09	0.09	0.11
0.14	0.15	0.17
0.30	0.31	0.33
0.54	0.57	0.60
1.01	1.09	1.12
1.51	1.69	1.65
1.75	2.01	1.85

Table A-9 Tone reproduction of production test print and predicted print of CMYK

dot area on film				Production test print				Predicted print			
C	M	Y	K	C	M	Y	K	C	M	Y	K
0	0	0	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
7	7	7	7	0.06	0.07	0.05	0.07	0.13	0.17	0.11	0.10
16	16	16	16	0.15	0.15	0.15	0.20	0.24	0.29	0.20	0.22
26	26	26	26	0.22	0.24	0.24	0.29	0.33	0.35	0.26	0.32
35	35	35	35	0.30	0.32	0.29	0.37	0.42	0.46	0.36	0.47
44	44	44	44	0.40	0.41	0.38	0.46	0.55	0.61	0.45	0.61
57	57	57	57	0.56	0.59	0.53	0.70	0.67	0.81	0.57	0.79
67	67	67	67	0.68	0.71	0.66	0.82	0.81	0.97	0.68	0.96
78	78	78	78	0.77	0.84	0.76	0.94	0.94	1.16	0.81	1.18
89	89	89	89	0.91	0.99	0.87	1.22	1.14	1.32	1.02	1.45
100	100	100	100	1.06	1.26	1.09	1.59	1.38	1.38	1.15	1.59

APPENDIX B

1.1 Software Development and Details

1.1.1 Hardware Requirements

To run this software, it requires at least:

- **CPU** Pentium 166 MMX
- **Hard Disk** 1.2 Gb
- **RAM** 32 Mb
- **Program** *Microsoft Visual Basic 6.0 plus Services Pack 4*
- **Floppy** 1.44 Mb

1.1.2 Software's details

```
Dim Pos As Long

Private Sub CloseForm_Click()
Unload Me
End Sub

Private Sub DeleteData_Click()
If MsgBox("ยืนยันการลบข้อมูล คลิกปุ่ม Yes ยกเลิกการลบคลลิกปุ่ม No", vbInformation + vbYesNo, "Delete Data") =
vbNo Then Exit Sub
Adodc2.RecordSource = "Select * From Original Where Run = '" & Text1 & "'"
Adodc2.Refresh
If Adodc2.Recordset.EOF Then MsgBox "'ไม่สามารถลบข้อมูลรายการนี้ได้ โปรดตรวจสอบข้อมูลใหม่", vbCritical +
vbOKOnly, "ยกเลิกการลบข้อมูล": Exit Sub
Do While Not Adodc2.Recordset.EOF
   Adodc2.Recordset.Delete
   Adodc2.Refresh
Loop
Adodc2.Refresh: Adodc1.Refresh
If Adodc1.Recordset.EOF Then
   Text1 = ""
   G1.Rows = 1
   Pos = 0
   Else
   Text1 = Adodc1.Recordset("Run")
   Pos = 1
End If
```

```
Label3.Caption = "ข้อมูลรายการที่  " & Pos & " / " & Adodc1.Recordset.RecordCount
End Sub

Private Sub Form_Load()
Adodc1.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" & App.Path &
"\Data.mdb;Persist Security Info=False"
Adodc2.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" & App.Path &
"\Data.mdb;Persist Security Info=False"
Data1.DatabaseName = App.Path & "\Data.mdb"
Data1.RecordSource = "Select Distinct Run From QOriginal"
Data1.Refresh
Adodc1.RecordSource = "Select Distinct Run From QOriginal"
Adodc1.Refresh
Dim i As Byte
For i = 0 To 1
   G1.ColWidth(i) = 1220
   G1.ColAlignment(i) = 5
Next
   G1.TextMatrix(0, 0) = "NO.": G1.TextMatrix(0, 1) = "Do"
   G1.Rows = 1
If Not Adodc1.Recordset.EOF Then
   Pos = 1: Text1 = Adodc1.Recordset("run")
Else
   Pos = 0: Text1 = ""
End If
Label3.Caption = "ข้อมูลรายการที่  " & Pos & " / " & Adodc1.Recordset.RecordCount
End Sub

Private Sub NewData_Click()
Adodc1.Refresh
If Adodc1.Recordset.EOF Then
   Text1 = "000001"
   Label3 = "ข้อมูลรายการที่  New / 0"
   Else
   Adodc1.Recordset.MoveLast
   Text1 = Format(Val(Adodc1.Recordset("Run")) + 1, "000000")
   Label3 = "ข้อมูลรายการที่  New / " & Adodc1.Recordset.RecordCount
   Pos = Adodc1.Recordset.RecordCount + 1
   Adodc1.Recordset.MoveNext
End If
Text(0).SetFocus
End Sub

Private Sub SaveData_Click()
If Text1 = "" Then MsgBox "ยกเลิกการบันทึก โปรดเลือกรายการ New Data ก่อนบันทึก", vbCritical + vbOKOnly,
"ยกเลิกการบันทึก": Exit Sub
If G1.Rows = 1 Then MsgBox "ยกเลิกการบันทึก โปรดเพิ่มรายการก่อนบันทึก", vbCritical + vbOKOnly, "ยกเลิกการ
บันทึก": Exit Sub
If MsgBox("ยืนยันการบันทึกข้อมูล คลิกปุ่ม Yes ยกเลิกการบันทึกคลิกปุ่ม No", vbInformation + vbYesNo, "Save Data")
= vbNo Then Exit Sub
Adodc2.RecordSource = "Select * From Original Where Run = '" & Text1 & "'"
Adodc2.Refresh
Do While Not Adodc2.Recordset.EOF
   Adodc2.Recordset.Delete
   Adodc2.Refresh
```

```
Loop
Dim i As Integer: Dim a As Byte
With Adodc2.Recordset
For i = 1 To G1.Rows - 1
   .AddNew
   .Fields(0) = Text1
   For a = 0 To 1
      If G1.TextMatrix(i, a) <> Empty Then .Fields(a + 1) = G1.TextMatrix(i, a)
   Next
   .Update
Next
End With
Adodc2.Refresh: Adodc1.Refresh
Pos = 1
Text1 = Adodc1.Recordset("Run")
Label3.Caption = "ข้อมูลรายการที่  " & Pos & " / " & Adodc1.Recordset.RecordCount
End Sub

Private Sub Text_KeyPress(Index As Integer, KeyAscii As Integer)
If KeyAscii = 13 Then
   If Index < 1 Then Text(Index + 1).SetFocus
   If Index >= 1 Then Text(0).SetFocus
   KeyAscii = 0
End If
End Sub

Private Sub Text1_Change()
Adodc2.RecordSource = "Select * From Q_Original Where Run = '" & Text1 & "'": Adodc2.Refresh
Call ReGrid
End Sub

Public Sub ReGrid()
G1.Rows = 1
If Data1.Recordset.RecordCount > 0 Then
   Data1.Recordset.FindFirst ("Run = '" & Text1 & "'")
   If Data1.Recordset.NoMatch Then
      Label3 = "ข้อมูลรายการที่  EOF / " & Data1.Recordset.RecordCount
      Else
      Label3 = "ข้อมูลรายการที่  " & Data1.Recordset.AbsolutePosition + 1 & " / " &
Data1.Recordset.RecordCount
   End If
   Else
      Label3 = "ข้อมูลรายการที่  EOF / " & Data1.Recordset.RecordCount
End If
If Not Adodc2.Recordset.EOF Then
Dim Rcount As Long: Dim i As Long
Rcount = Adodc2.Recordset.RecordCount
With Adodc2.Recordset
For i = 1 To Rcount
   G1.AddItem .Fields(1) & vbTab & Format(.Fields(2), "0.00")
    Adodc2.Recordset.MoveNext
Next
End With
End If
Dim Num As Double
Num = 0
```

```
For i = 1 To G1.Rows - 1
   If G1.TextMatrix(i, 1) > Num Then Num = G1.TextMatrix(i, 1)
Next
Text2 = Format(Num, "##0.00")
End Sub

Private Sub Command1_Click(Index As Integer)
Text(0).SetFocus
End Sub

Private Sub Command2_Click()
   G1.AddItem Text(0) & vbTab & Text(1)
   Text(0) = "": Text(1) = ""
   Text(0).SetFocus
Dim i As Long: Dim Num As Double
Num = 0
For i = 1 To G1.Rows - 1
   If G1.TextMatrix(i, 1) > Num Then Num = G1.TextMatrix(i, 1)
Next
Text2 = Format(Num, "##0.00")
End Sub

Private Sub Command3_Click()
If G1.Rows <= 2 Then
   G1.Rows = 1
   Else
   G1.RemoveItem (G1.Row)
End If
Dim i As Long: Dim Num As Double
Num = 0
For i = 1 To G1.Rows - 1
   If G1.TextMatrix(i, 1) > Num Then Num = G1.TextMatrix(i, 1)
Next
Text2 = Format(Num, "##0.00")
End Sub

Private Sub Command4_Click()
If Pos <= 1 Then Exit Sub
Adodc1.Recordset.MovePrevious
Pos = Pos - 1
Label3.Caption = "ข้อมูลรายการที่  " & Pos & " / " & Adodc1.Recordset.RecordCount
Text1 = Adodc1.Recordset("Run")
End Sub

Private Sub Command5_Click()
If Pos >= Adodc1.Recordset.RecordCount Then Exit Sub
Adodc1.Recordset.MoveNext
If Adodc1.Recordset.EOF Then Exit Sub
Pos = Pos + 1
Label3.Caption = "ข้อมูลรายการที่  " & Pos & " / " & Adodc1.Recordset.RecordCount
Text1 = Adodc1.Recordset("Run")
End Sub

Private Sub Command6_Click()
On Error GoTo Err_
Pos = 0
Text1 = ""
```

```
Adodc1.Recordset.MoveFirst
Text1 = Adodc1.Recordset("Run")
Pos = 1
Err_:
Label3.Caption = "ข้อมูลรายการที่  " & Pos & " / " & Adodc1.Recordset.RecordCount
End Sub

Private Sub Command7_Click()
On Error GoTo Err_
Text1 = ""
Adodc1.Recordset.MoveLast
Text1 = Adodc1.Recordset("Run")
Err_:
Pos = Adodc1.Recordset.RecordCount
Label3.Caption = "ข้อมูลรายการที่  " & Pos & " / " & Adodc1.Recordset.RecordCount
End Sub

Sub PGraph()
Dim i As Byte
Dim sleft As Long
Do While C.Count > 1
   Unload C(C.Count - 1): Unload M(M.Count - 1): Unload Y(Y.Count - 1): Unload K(K.Count - 1)
   If LC.Count > 1 Then Unload LC(LC.Count - 1): Unload LM(LM.Count - 1): Unload LY(LY.Count
- 1): Unload LK(LK.Count - 1)
Loop
   If G1.Rows <= 1 Then Exit Sub
      For i = 1 To G1.Rows - 1
         sleft = (Val(G1.TextMatrix(i, 0)) * 40) * 45
         Load C(C.Count): C(C.Count - 1).Visible = True: C(C.Count - 1).Left = (sleft + (495 - 70))
         Load M(M.Count): M(M.Count - 1).Visible = True: M(M.Count - 1).Left = (sleft + (495 - 70))
         Load Y(Y.Count): Y(Y.Count - 1).Visible = True: Y(Y.Count - 1).Left = (sleft + (495 - 70))
         Load K(K.Count): K(K.Count - 1).Visible = True: K(K.Count - 1).Left = (sleft + (495 - 70))
         C(C.Count - 1).Top = 6075 - ((Val(G1.TextMatrix(i, 1)) * 45) + 70)
         M(M.Count - 1).Top = 6075 - ((Val(G1.TextMatrix(i, 2)) * 45) + 70)
         Y(Y.Count - 1).Top = 6075 - ((Val(G1.TextMatrix(i, 3)) * 45) + 70)
         K(K.Count - 1).Top = 6075 - ((Val(G1.TextMatrix(i, 4)) * 45) + 70)
         'Adodc1.Recordset.MoveNext
      Next
      For i = 1 To C.Count - 2
         Load LC(LC.Count): LC(LC.Count - 1).Visible = True
         LC(LC.Count - 1).X1 = C(i).Left + 70
         LC(LC.Count - 1).X2 = C(i + 1).Left + 70
         LC(LC.Count - 1).Y1 = C(i).Top + 70
         LC(LC.Count - 1).Y2 = C(i + 1).Top + 70
         Load LM(LM.Count): LM(LM.Count - 1).Visible = True
         LM(LM.Count - 1).X1 = M(i).Left + 70
         LM(LM.Count - 1).X2 = M(i + 1).Left + 70
         LM(LM.Count - 1).Y1 = M(i).Top + 70
         LM(LM.Count - 1).Y2 = M(i + 1).Top + 70
         Load LY(LY.Count): LY(LY.Count - 1).Visible = True
         LY(LY.Count - 1).X1 = Y(i).Left + 70
         LY(LY.Count - 1).X2 = Y(i + 1).Left + 70
         LY(LY.Count - 1).Y1 = Y(i).Top + 70
         LY(LY.Count - 1).Y2 = Y(i + 1).Top + 70
         Load LK(LK.Count): LK(LK.Count - 1).Visible = True
         LK(LK.Count - 1).X1 = K(i).Left + 70
         LK(LK.Count - 1).X2 = K(i + 1).Left + 70
```

```
        LK(LK.Count - 1).Y1 = K(i).Top + 70
        LK(LK.Count - 1).Y2 = K(i + 1).Top + 70
      Next
End Sub

Sub Ado2()
If Data1.Recordset.RecordCount > 0 Then
   If Data1.Recordset.EOF Then Data1.Recordset.MoveFirst
   If Data1.Recordset.BOF Then Data1.Recordset.MoveLast
   Label10 = "Record No.  " & Data1.Recordset.AbsolutePosition + 1 & " / " &
Data1.Recordset.RecordCount
   Text1 = Data1.Recordset("Run")
   Else
   Label10 = "Record No.  0 / 0"
   Text1 = ""
   Exit Sub
End If
   Adodc1.RecordSource = "Select * From PrintOutData Where Run = '" & Data1.Recordset("Run") &
"'": Adodc1.Refresh
End Sub

Private Sub Command1_Click()
Unload Me
End Sub

Private Sub Command2_Click()
Me.WindowState = 1
End Sub

Private Sub Command4_Click()
If Not Data1.Recordset.EOF And Not Data1.Recordset.BOF Then
   Data1.Recordset.MovePrevious
   Else
   If Data1.Recordset.RecordCount > 0 Then Data1.Recordset.MoveFirst
End If
   Call Ado2
End Sub

Private Sub Command5_Click()
If Not Data1.Recordset.EOF And Not Data1.Recordset.BOF Then
   Data1.Recordset.MoveNext
   Else
   If Data1.Recordset.RecordCount > 0 Then Data1.Recordset.MoveLast
End If
   Call Ado2
End Sub

Private Sub Command6_Click()
If Data1.Recordset.RecordCount > 0 Then
   Data1.Recordset.MoveFirst
End If
   Adodc1.RecordSource = "Select * From PrintOutData Where Run = '" & Data1.Recordset("Run") &
"'": Adodc1.Refresh
   Call Ado2
End Sub

Private Sub Command7_Click()
```

```
If Data1.Recordset.RecordCount > 0 Then
   Data1.Recordset.MoveLast
End If
   Call Ado2
End Sub

Private Sub Command3_Click()
Dim i As Byte: Dim C As Double: Dim M As Double: Dim Y As Double
Dim Dk As Double: Dim Bk As Double
Dim OFBk As Double: Dim ODBk As Double
Dim Diff1 As Double: Dim Diff2 As Double
Dim D3c As Double: Dim UpD3c As Double: Dim DownD3c As Double
Dim UpF As Double: Dim DownF As Double
G1.Rows = 1
For i = 1 To GraphD34cDk.G1.Rows - 1
   G1.AddItem GraphD34cDk.G1.TextMatrix(i, 1)
Next
For i = 1 To GraphD34cDk.G1.Rows - 1
   Bk = 0: Diff1 = 0: Diff2 = 0: OFBk = 0: ODBk = 0
   Dk = GraphD34cDk.G1.TextMatrix(i, 4)
   GraphDoVSDensity.Adodc1.Refresh
   Do While Not GraphDoVSDensity.Adodc1.Recordset.EOF
         If GraphDoVSDensity.Adodc1.Recordset("DensityBk") >= Dk Then Exit Do
         GraphDoVSDensity.Adodc1.Recordset.MoveNext
   Loop
         If GraphDoVSDensity.Adodc1.Recordset("DensityBk") > Dk Then
               GraphDoVSDensity.Adodc1.Recordset.MovePrevious
               Diff1 = GraphDoVSDensity.Adodc1.Recordset("DensityBk")
               Diff2 = GraphDoVSDensity.Adodc1.Recordset("FilmBk")
               ODBk = Diff1
               OFBk = Diff2
               GraphDoVSDensity.Adodc1.Recordset.MoveNext
               Diff1 = GraphDoVSDensity.Adodc1.Recordset("DensityBk") - Diff1
               Diff2 = GraphDoVSDensity.Adodc1.Recordset("FilmBk") - Diff2
               Bk = OFBk + ((Diff2 / Diff1) * (Dk - ODBk))
         End If
         If GraphDoVSDensity.Adodc1.Recordset("DensityBk") = Dk Then
               Bk = GraphDoVSDensity.Adodc1.Recordset("FilmBk")
         End If
         G1.TextMatrix(i, 4) = Format(Bk, "##0")
Next
Dim c1 As Double
Dim d1 As Double
Dim Txt As Double
GraphGrey.Adodc1.Refresh
If GraphGrey.Adodc1.Recordset.EOF Then Exit Sub
For i = 1 To GraphD34cDk.G1.Rows - 1
   Txt = Val(GraphD34cDk.G1.TextMatrix(i, 3))
   Do While Not GraphGrey.Adodc1.Recordset.EOF
      If Txt > GraphGrey.Adodc1.Recordset("D3c") Then
         GraphGrey.Adodc1.Recordset.MoveNext
         Else
         Exit Do
      End If
   Loop
   Diff1 = 0:    Diff2 = 0
   c1 = 0:    c2 = 0
```

```
   C = 0: M = 0: Y = 0
'************************** หาค่า C **************************
      GraphGrey.Adodc1.Recordset.MovePrevious
   If Not GraphGrey.Adodc1.Recordset.BOF Then
      Diff1 = GraphGrey.Adodc1.Recordset("DotCy")
      Diff2 = GraphGrey.Adodc1.Recordset("D3c")
      c1 = GraphGrey.Adodc1.Recordset("D3c")
      d1 = GraphGrey.Adodc1.Recordset("DotCy")
   End If
      GraphGrey.Adodc1.Recordset.MoveNext
      Diff1 = GraphGrey.Adodc1.Recordset("DotCy") - Diff1
      Diff2 = GraphGrey.Adodc1.Recordset("D3c") - Diff2
   If Diff2 > 0 Then
      C = ((Diff1 / Diff2) * (Txt - c1)) + d1
      Else
      C = 0
   End If
   G1.TextMatrix(i, 1) = Format(C, "##0")
'************************** หาค่า M **************************
      GraphGrey.Adodc1.Recordset.MovePrevious
   If Not GraphGrey.Adodc1.Recordset.BOF Then
      Diff1 = GraphGrey.Adodc1.Recordset("DotMa")
      Diff2 = GraphGrey.Adodc1.Recordset("D3c")
      c1 = GraphGrey.Adodc1.Recordset("D3c")
      d1 = GraphGrey.Adodc1.Recordset("DotMa")
   End If
      GraphGrey.Adodc1.Recordset.MoveNext
      Diff1 = GraphGrey.Adodc1.Recordset("DotMa") - Diff1
      Diff2 = GraphGrey.Adodc1.Recordset("D3c") - Diff2
   If Diff2 > 0 Then
      M = ((Diff1 / Diff2) * (Txt - c1)) + d1
      Else
      M = 0
   End If
   G1.TextMatrix(i, 2) = Format(M, "##0")
'************************** หาค่า Y **************************
      GraphGrey.Adodc1.Recordset.MovePrevious
   If Not GraphGrey.Adodc1.Recordset.BOF Then
      Diff1 = GraphGrey.Adodc1.Recordset("DotYe")
      Diff2 = GraphGrey.Adodc1.Recordset("D3c")
      c1 = GraphGrey.Adodc1.Recordset("D3c")
      d1 = GraphGrey.Adodc1.Recordset("DotYe")
   End If
      GraphGrey.Adodc1.Recordset.MoveNext
      Diff1 = GraphGrey.Adodc1.Recordset("DotYe") - Diff1
      Diff2 = GraphGrey.Adodc1.Recordset("D3c") - Diff2
   If Diff2 > 0 Then
      Y = ((Diff1 / Diff2) * (Txt - c1)) + d1
      Else
      Y = 0
   End If
   G1.TextMatrix(i, 3) = Format(Y, "##0")
GraphGrey.Adodc1.Recordset.MoveFirst
Next
Call PGraph
End Sub
```

```
Private Sub Form_Activate()
'
End Sub

Private Sub Form_Load()
Dim i As Byte
Dim a As Byte
G1.Rows = 1
For i = 0 To 4
   G1.ColWidth(i) = 890
   G1.ColAlignment(i) = 7
Next
G1.Row = 0
For i = 0 To 4
G1.Col = i
G1.CellAlignment = 5
G1.CellFontBold = True
Next
G1.TextMatrix(0, 0) = "Do"
G1.TextMatrix(0, 1) = "C": G1.TextMatrix(0, 2) = "M": G1.TextMatrix(0, 3) = "Y": G1.TextMatrix(0,
4) = "K"
a = 0
Adodc1.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" & App.Path &
"\Data.mdb;Persist Security Info=False"
Data1.DatabaseName = App.Path & "\Data.MDB"
For i = 1 To 100
   a = a + 1
   Load L3(L3.Count)
   L3(L3.Count - 1).Visible = True
   L3(L3.Count - 1).X1 = 420
   L3(L3.Count - 1).Y1 = L3(L3.Count - 2).Y1 - 45
   L3(L3.Count - 1).Y2 = L3(L3.Count - 2).Y2 - 45
   Load L4(L4.Count)
   L4(L4.Count - 1).Visible = True
   L4(L3.Count - 1).Y2 = 6165
   L4(L4.Count - 1).X1 = L4(L4.Count - 2).X1 + 45
   L4(L4.Count - 1).X2 = L4(L4.Count - 2).X2 + 45
   If a = 10 Then
      L3(L3.Count - 1).X1 = 380
      L4(L3.Count - 1).Y2 = 6210
      Load L9(L9.Count)
      If Ln3.Count < 10 Then
         Load Ln3(Ln3.Count)
         Load Ln4(Ln4.Count)
         Ln3(Ln3.Count - 1).Visible = True
         Ln4(Ln4.Count - 1).Visible = True
         Ln3(Ln3.Count - 1).Y1 = Ln3(Ln3.Count - 2).Y1 - 450
         Ln3(Ln3.Count - 1).Y2 = Ln3(Ln3.Count - 2).Y2 - 450
         Ln4(Ln4.Count - 1).X1 = Ln4(Ln4.Count - 2).X1 + 450
         Ln4(Ln4.Count - 1).X2 = Ln4(Ln4.Count - 2).X2 + 450
      End If
      L9(L9.Count - 1).Visible = True
      L9(L9.Count - 1).Top = L3(L3.Count - 1).Y1 - 110
      L9(L9.Count - 1).Caption = L9(L9.Count - 2).Caption + 10
      Load L10(L10.Count)
      L10(L10.Count - 1).Visible = True
```

```
      L10(L10.Count - 1).Left = L4(L4.Count - 1).X1 - (125 + 120)
      L10(L10.Count - 1).Caption = Format(Val(L10(L10.Count - 2).Caption) + 0.25, "0.00")
      a = 0
   End If
Next
Data1.RecordSource = "Select Distinct Run From PrintoutData"
Data1.Refresh
Call Ado2
End Sub

Private Sub Text1_Change()
Adodc1.RecordSource = "Select * From PrintoutData Where Run = '" & Text1 & "'"
Adodc1.Refresh
If Adodc1.Recordset.RecordCount > 0 Then
   Data1.Recordset.FindFirst ("Run = '" & Text1 & "'")
   Call Ado2
   Else
   Label10 = "Record No.  * / " & Data1.Recordset.RecordCount
End If
   Call PGraph
End Sub

Sub PGraph()
Dim sleft As Long
Dim i As Byte
  Do While C.Count > 1
   Unload C(C.Count - 1): Unload M(M.Count - 1): Unload Y(Y.Count - 1)
   Loop
   Do While LC.Count > 1
      Unload LC(LC.Count - 1): Unload LM(LM.Count - 1): Unload LY(LY.Count - 1)
   Loop
      If G1.Rows <= 1 Then Exit Sub
      For i = 1 To G1.Rows - 1
         sleft = (Val(G1.TextMatrix(i, 1)) * (100 / 2)) * 45
         Load C(C.Count): C(C.Count - 1).Visible = True: C(C.Count - 1).Left = sleft + (735 - 70)
         Load M(M.Count): M(M.Count - 1).Visible = True: M(M.Count - 1).Left = sleft + (735 - 70)
         Load Y(Y.Count): Y(Y.Count - 1).Visible = True: Y(Y.Count - 1).Left = sleft + (735 - 70)
         C(C.Count - 1).Top = 6075 - (((Val(G1.TextMatrix(i, 2)) * (100 / 3)) * 45) + 70)
         M(M.Count - 1).Top = 6075 - (((Val(G1.TextMatrix(i, 3)) * (100 / 3)) * 45) + 70)
         Y(Y.Count - 1).Top = 6075 - (((Val(G1.TextMatrix(i, 4)) * (100 / 3)) * 45) + 70)
      Next
      For i = 1 To C.Count - 2
         Load LC(LC.Count): LC(LC.Count - 1).Visible = True
         LC(LC.Count - 1).X1 = C(i).Left + 70
         LC(LC.Count - 1).X2 = C(i + 1).Left + 70
         LC(LC.Count - 1).Y1 = C(i).Top + 70
         LC(LC.Count - 1).Y2 = C(i + 1).Top + 70
         Load LM(LM.Count): LM(LM.Count - 1).Visible = True
         LM(LM.Count - 1).X1 = M(i).Left + 70
         LM(LM.Count - 1).X2 = M(i + 1).Left + 70
         LM(LM.Count - 1).Y1 = M(i).Top + 70
         LM(LM.Count - 1).Y2 = M(i + 1).Top + 70
         Load LY(LY.Count): LY(LY.Count - 1).Visible = True
         LY(LY.Count - 1).X1 = Y(i).Left + 70
         LY(LY.Count - 1).X2 = Y(i + 1).Left + 70
         LY(LY.Count - 1).Y1 = Y(i).Top + 70
         LY(LY.Count - 1).Y2 = Y(i + 1).Top + 70
```

```
      Next
End Sub

Sub Ado2()
If Data1.Recordset.RecordCount > 0 Then
   If Data1.Recordset.EOF Then Data1.Recordset.MoveLast
   If Data1.Recordset.BOF Then Data1.Recordset.MoveFirst
   Label10.Caption = "Record No.  " & Data1.Recordset.AbsolutePosition + 1 & " / " &
Data1.Recordset.RecordCount
   Text1 = Data1.Recordset("Run")
   Else
   Label10.Caption = "Record No.  0 / 0"
   Text1 = ""
   Exit Sub
End If
End Sub

Private Sub Command1_Click()
Unload Me
End Sub

Private Sub Command2_Click()
Me.WindowState = 1
End Sub

Private Sub Command3_Click()
If G1.Rows > 1 And Val(Text2) > 0 And Val(Text3) > 0 Then
   Dim i As Variant
   Dim S1 As Double: Dim S2 As Double: Dim S3 As Double:  Dim DoMax As Double: Dim Dr As
Double
   S1 = 0: S2 = 0: S3 = 0:  DoMax = 0
   For i = 1 To G1.Rows - 1
      If G1.TextMatrix(i, 1) > DoMax Then DoMax = G1.TextMatrix(i, 1)
   Next
   For i = 1 To G1.Rows - 1
      S1 = (10 ^ (1 - (G1.TextMatrix(i, 1) / 2))) - (10 ^ (1 - (DoMax / 2)))
      S2 = (10 - (10 ^ (1 - (Val(Text3) / 2)))) / (10 - (10 ^ (1 - (DoMax / 2))))
      S3 = (10 ^ (1 - (Val(Text3) / 2))) + S2 * S1
      Dr = 2 * (1 - (Log(S3) / Log(10)))
      G1.TextMatrix(i, 2) = Format(Dr, "##0.00")
   Next
   i = 0
   Do
   If Val(Text3) - Val(Text2) <= i - (Val(Text2) * i * 0.3) Then i = i - 0.01: Exit Do
   i = i + 0.01
   Loop
   Do
   i = i + 0.0001
   If Val(Text3) - Val(Text2) <= i - (Val(Text2) * i * 0.3) Then: Exit Do
   Loop
   Dim Dk As Double
   Dk = (i / (DoMax - 0.8))
   For i = 1 To G1.Rows - 1
      If Val(G1.TextMatrix(i, 1)) > 0.8 Then
         G1.TextMatrix(i, 4) = Format(Dk * (Val(G1.TextMatrix(i, 1)) - 0.8), "##0.00")
         Else
         G1.TextMatrix(i, 4) = "0.00"
```

```
      End If
   Next
   For i = 1 To G1.Rows - 1
   Dr = 0
   If Val(G1.TextMatrix(i, 1)) > 0.8 Then
      Do
      If Val(G1.TextMatrix(i, 2)) - Val(G1.TextMatrix(i, 4)) <= Dr - (Val(G1.TextMatrix(i, 4)) * Dr *
0.3) Then Dr = Dr - 0.01: Exit Do
      Dr = Dr + 0.01
      Loop
      Do
      Dr = Dr + 0.001
      If Val(G1.TextMatrix(i, 2)) - Val(G1.TextMatrix(i, 4)) <= Val(Format(Dr - (Val(G1.TextMatrix(i,
4)) * Dr * 0.3), "###0.00")) Then Exit Do
      Loop
      'MsgBox Val(G1.TextMatrix(i, 2)) - Val(G1.TextMatrix(i, 4)) & "  " & Val(Format(Dr -
(Val(G1.TextMatrix(i, 4)) * Dr * 0.3), "###0.00")) & "   " & Dr
      G1.TextMatrix(i, 3) = Format(Dr, "##0.00")
      Else
      G1.TextMatrix(i, 3) = G1.TextMatrix(i, 2)
   End If
   Next

End If
Call PGraph
End Sub

Private Sub Command4_Click()
If Not Data1.Recordset.EOF And Not Data1.Recordset.BOF Then
   Data1.Recordset.MovePrevious
   Else
   If Data1.Recordset.RecordCount > 0 Then Data1.Recordset.MoveFirst
End If
   Call Ado2
End Sub

Private Sub Command5_Click()
If Not Data1.Recordset.EOF And Not Data1.Recordset.BOF Then
   Data1.Recordset.MoveNext
   Else
   If Data1.Recordset.RecordCount > 0 Then Data1.Recordset.MoveLast
End If
   Call Ado2
End Sub

Private Sub Command6_Click()
If Data1.Recordset.RecordCount > 0 Then
   Data1.Recordset.MoveFirst
End If
   Call Ado2
End Sub

Private Sub Command7_Click()
If Data1.Recordset.RecordCount > 0 Then
   Data1.Recordset.MoveLast
End If
   Call Ado2
```

```
End Sub

Private Sub Form_Load()
Dim i As Byte
Dim a As Byte
Adodc1.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" & App.Path &
"\Data.mdb;Persist Security Info=False"
Data1.DatabaseName = App.Path & "\Data.mdb"
Data1.RecordSource = "Select Distinct Run From Original"
Data1.Refresh
G1.ColWidth(0) = 500: G1.TextMatrix(0, 0) = "No."
G1.TextMatrix(0, 1) = "Do": G1.TextMatrix(0, 2) = "Dr"
G1.TextMatrix(0, 3) = "D3c": G1.TextMatrix(0, 4) = "Dk"
G1.ColAlignment(0) = 5
G1.Row = 0
G1.Col = 0
G1.CellFontBold = True
For i = 1 To 4
   G1.ColAlignment(i) = 7
   G1.Col = i
   G1.CellAlignment = 5
   G1.CellFontBold = True
   G1.ColWidth(i) = 990
Next
Adodc1.Refresh
With Adodc1.Recordset
G1.Rows = 1
Do While Not .EOF
   G1.AddItem .Fields("No") & vbTab & Format(.Fields("Do"), "##0.00") & vbTab & "0.00" & vbTab
& "0.00" & vbTab & "0.00"
   .MoveNext
Loop
End With
a = 0
For i = 1 To 100
   a = a + 1
   Load L3(L3.Count)
   L3(L3.Count - 1).Visible = True
   L3(L3.Count - 1).X1 = 645
   L3(L3.Count - 1).Y1 = L3(L3.Count - 2).Y1 - 45
   L3(L3.Count - 1).Y2 = L3(L3.Count - 2).Y2 - 45
   Load L4(L4.Count)
   L4(L4.Count - 1).Visible = True
   L4(L3.Count - 1).Y2 = 6165
   L4(L4.Count - 1).X1 = L4(L4.Count - 2).X1 + 45
   L4(L4.Count - 1).X2 = L4(L4.Count - 2).X2 + 45
   If a = 10 Then
      L3(L3.Count - 1).X1 = 600
      L4(L3.Count - 1).Y2 = 6210
      Load L9(L9.Count)
      If Ln3.Count < 10 Then
         Load Ln3(Ln3.Count)
         Load Ln4(Ln4.Count)
         Ln3(Ln3.Count - 1).Visible = True
         Ln4(Ln4.Count - 1).Visible = True
         Ln3(Ln3.Count - 1).Y1 = Ln3(Ln3.Count - 2).Y1 - 450
         Ln3(Ln3.Count - 1).Y2 = Ln3(Ln3.Count - 2).Y2 - 450
```

```
            Ln4(Ln4.Count - 1).X1 = Ln4(Ln4.Count - 2).X1 + 450
            Ln4(Ln4.Count - 1).X2 = Ln4(Ln4.Count - 2).X2 + 450
        End If
        L9(L9.Count - 1).Visible = True
        L9(L9.Count - 1).Top = L3(L3.Count - 1).Y1 - 110
        L9(L9.Count - 1).Caption = Format(L9(L9.Count - 2).Caption + 0.3, "0.00")
        Load L10(L10.Count)
        L10(L10.Count - 1).Visible = True
        L10(L10.Count - 1).Left = L4(L4.Count - 1).X1 - 225
        L10(L10.Count - 1).Caption = Format(L10(L10.Count - 2).Caption + 0.2, "0.00")
        a = 0
    End If
Next
Adodc1.Refresh
Call Ado2
End Sub

Private Sub Text1_Change()
Adodc1.RecordSource = "Select * From Original Where Run = '" & Text1 & "' Order By Do":
Adodc1.Refresh
If Not Adodc1.Recordset.EOF Then
    Data1.Recordset.FindFirst ("Run = '" & Text1 & "'")
    Label10.Caption = "Record No.  " & Data1.Recordset.AbsolutePosition + 1 & " / " &
Data1.Recordset.RecordCount
    Else
    Label10.Caption = "Record No.  EOF / " & Data1.Recordset.RecordCount
End If
With Adodc1.Recordset
G1.Rows = 1
Do While Not .EOF
    G1.AddItem .Fields("No") & vbTab & Format(.Fields("Do"), "##0.00") & vbTab & "0.00" & vbTab
& "0.00" & vbTab & "0.00"
    .MoveNext
Loop
End With
'*************************************************************************
If G1.Rows > 1 And Val(Text2) > 0 And Val(Text3) > 0 Then
    Dim i As Variant
    Dim S1 As Double: Dim S2 As Double: Dim S3 As Double:  Dim DoMax As Double: Dim Dr As
Double
    S1 = 0: S2 = 0: S3 = 0:  DoMax = 0
    For i = 1 To G1.Rows - 1
        If G1.TextMatrix(i, 1) > DoMax Then DoMax = G1.TextMatrix(i, 1)
    Next
    For i = 1 To G1.Rows - 1
        S1 = (10 ^ (1 - (G1.TextMatrix(i, 1) / 2))) - (10 ^ (1 - (DoMax / 2)))
        S2 = (10 - (10 ^ (1 - (Val(Text3) / 2)))) / (10 - (10 ^ (1 - (DoMax / 2))))
        S3 = (10 ^ (1 - (Val(Text3) / 2))) + S2 * S1
        Dr = 2 * (1 - (Log(S3) / Log(10)))
        G1.TextMatrix(i, 2) = Format(Dr, "##0.00")
    Next
    i = 0
    Do
    If Val(Text3) - Val(Text2) <= i - (Val(Text2) * i * 0.3) Then Exit Do
    i = i + 0.01
    Loop
    Dim Dk As Double
```

```
   Dk = (i / (DoMax - 0.8))
   For i = 1 To G1.Rows - 1
      If Val(G1.TextMatrix(i, 1)) > 0.8 Then
         G1.TextMatrix(i, 4) = Format(Dk * (Val(G1.TextMatrix(i, 1)) - 0.8), "##0.00")
         Else
         G1.TextMatrix(i, 4) = "0.00"
      End If
   Next
   For i = 1 To G1.Rows - 1
   Dr = 0
   If Val(G1.TextMatrix(i, 1)) > 0.8 Then
      Do
      If Val(G1.TextMatrix(i, 2)) - Val(G1.TextMatrix(i, 4)) <= Dr - (Val(G1.TextMatrix(i, 4)) * Dr *
0.3) Then Exit Do
      Dr = Dr + 0.01
      Loop
      G1.TextMatrix(i, 3) = Format(Dr, "##0.00")
      Else
      G1.TextMatrix(i, 3) = G1.TextMatrix(i, 2)
   End If
   Next

End If
'*************************************************************************
Call PGraph
End Sub

Sub PGraph()
   Do While C.Count > 1
      Unload C(C.Count - 1): Unload M(M.Count - 1): Unload Y(Y.Count - 1): Unload K(K.Count - 1)
      If LC.Count > 1 Then Unload LC(LC.Count - 1): Unload LM(LM.Count - 1): Unload
LY(LY.Count - 1): Unload LK(LK.Count - 1)
   Loop
   If Not Adodc1.Recordset.EOF Then
      Dim sleft As Long
      Do While Not Adodc1.Recordset.EOF
         sleft = Adodc1.Recordset("InputData") * 45
         Load C(C.Count): C(C.Count - 1).Visible = True: C(C.Count - 1).Left = sleft + (495 - 70)
         Load M(M.Count): M(M.Count - 1).Visible = True: M(M.Count - 1).Left = sleft + (495 - 70)
         Load Y(Y.Count): Y(Y.Count - 1).Visible = True: Y(Y.Count - 1).Left = sleft + (495 - 70)
         Load K(K.Count): K(K.Count - 1).Visible = True: K(K.Count - 1).Left = sleft + (495 - 70)
         C(C.Count - 1).Top = 6075 - ((Adodc1.Recordset("FilmCy") * 45) + 70)
         M(M.Count - 1).Top = 6075 - ((Adodc1.Recordset("FilmMa") * 45) + 70)
         Y(Y.Count - 1).Top = 6075 - ((Adodc1.Recordset("FilmYe") * 45) + 70)
         K(K.Count - 1).Top = 6075 - ((Adodc1.Recordset("FilmBk") * 45) + 70)
         Adodc1.Recordset.MoveNext
      Loop
      Dim i As Byte
      For i = 1 To C.Count - 2
         Load LC(LC.Count): LC(LC.Count - 1).Visible = True
         LC(LC.Count - 1).X1 = C(i).Left + 70
         LC(LC.Count - 1).X2 = C(i + 1).Left + 70
         LC(LC.Count - 1).Y1 = C(i).Top + 70
         LC(LC.Count - 1).Y2 = C(i + 1).Top + 70
         Load LM(LM.Count): LM(LM.Count - 1).Visible = True
         LM(LM.Count - 1).X1 = M(i).Left + 70
         LM(LM.Count - 1).X2 = M(i + 1).Left + 70
```

```
            LM(LM.Count - 1).Y1 = M(i).Top + 70
            LM(LM.Count - 1).Y2 = M(i + 1).Top + 70
            Load LY(LY.Count): LY(LY.Count - 1).Visible = True
            LY(LY.Count - 1).X1 = Y(i).Left + 70
            LY(LY.Count - 1).X2 = Y(i + 1).Left + 70
            LY(LY.Count - 1).Y1 = Y(i).Top + 70
            LY(LY.Count - 1).Y2 = Y(i + 1).Top + 70
            Load LK(LK.Count): LK(LK.Count - 1).Visible = True
            LK(LK.Count - 1).X1 = K(i).Left + 70
            LK(LK.Count - 1).X2 = K(i + 1).Left + 70
            LK(LK.Count - 1).Y1 = K(i).Top + 70
            LK(LK.Count - 1).Y2 = K(i + 1).Top + 70
        Next
    End If
Adodc1.Refresh
End Sub

Sub Ado2()
If Data1.Recordset.RecordCount > 0 Then
    If Data1.Recordset.EOF Then Data1.Recordset.MoveFirst
    If Data1.Recordset.BOF Then Data1.Recordset.MoveLast
    Label10 = "Record No.  " & Data1.Recordset.AbsolutePosition + 1 & " / " &
Data1.Recordset.RecordCount
    Text1 = Data1.Recordset("Run")
    Else
    Label10 = "Record No.  0 / 0"
    Text1 = ""
    Exit Sub
End If
    Adodc1.RecordSource = "Select * From PrintOutData Where Run = '" & Data1.Recordset("Run") &
"' Order By InputData": Adodc1.Refresh
End Sub

Private Sub Command1_Click()
Unload Me
End Sub

Private Sub Command2_Click()
Me.WindowState = 1
End Sub

Private Sub Command4_Click()
If Not Data1.Recordset.EOF And Not Data1.Recordset.BOF Then
    Data1.Recordset.MovePrevious
    Else
    If Data1.Recordset.RecordCount > 0 Then Data1.Recordset.MoveFirst
End If
    Call Ado2
End Sub

Private Sub Command5_Click()
If Not Data1.Recordset.EOF And Not Data1.Recordset.BOF Then
    Data1.Recordset.MoveNext
    Else
    If Data1.Recordset.RecordCount > 0 Then Data1.Recordset.MoveLast
End If
    Call Ado2
```

```
End Sub

Private Sub Command6_Click()
If Data1.Recordset.RecordCount > 0 Then
   Data1.Recordset.MoveFirst
End If
   Adodc1.RecordSource = "Select * From PrintOutData Where Run = '" & Data1.Recordset("Run") &
"'": Adodc1.Refresh
   Call Ado2
End Sub

Private Sub Command7_Click()
If Data1.Recordset.RecordCount > 0 Then
   Data1.Recordset.MoveLast
End If
   Call Ado2
End Sub

Private Sub Form_Load()
Dim i As Byte
Dim a As Byte
a = 0
Adodc1.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" & App.Path &
"\Data.mdb;Persist Security Info=False"
Set DataGrid1.DataSource = Adodc1
Data1.DatabaseName = App.Path & "\Data.MDB"
For i = 1 To 100
   a = a + 1
   Load L3(L3.Count)
   L3(L3.Count - 1).Visible = True
   L3(L3.Count - 1).X1 = 420
   L3(L3.Count - 1).Y1 = L3(L3.Count - 2).Y1 - 45
   L3(L3.Count - 1).Y2 = L3(L3.Count - 2).Y2 - 45
   Load L4(L4.Count)
   L4(L4.Count - 1).Visible = True
   L4(L3.Count - 1).Y2 = 6165
   L4(L4.Count - 1).X1 = L4(L4.Count - 2).X1 + 45
   L4(L4.Count - 1).X2 = L4(L4.Count - 2).X2 + 45
   If a = 10 Then
      L3(L3.Count - 1).X1 = 380
      L4(L3.Count - 1).Y2 = 6210
      Load L9(L9.Count)
      If Ln3.Count < 10 Then
         Load Ln3(Ln3.Count)
         Load Ln4(Ln4.Count)
         Ln3(Ln3.Count - 1).Visible = True
         Ln4(Ln4.Count - 1).Visible = True
         Ln3(Ln3.Count - 1).Y1 = Ln3(Ln3.Count - 2).Y1 - 450
         Ln3(Ln3.Count - 1).Y2 = Ln3(Ln3.Count - 2).Y2 - 450
         Ln4(Ln4.Count - 1).X1 = Ln4(Ln4.Count - 2).X1 + 450
         Ln4(Ln4.Count - 1).X2 = Ln4(Ln4.Count - 2).X2 + 450
      End If
      L9(L9.Count - 1).Visible = True
      L9(L9.Count - 1).Top = L3(L3.Count - 1).Y1 - 110
      L9(L9.Count - 1).Caption = L9(L9.Count - 2).Caption + 10
      Load L10(L10.Count)
      L10(L10.Count - 1).Visible = True
```

```
      L10(L10.Count - 1).Left = L4(L4.Count - 1).X1 - 125
      L10(L10.Count - 1).Caption = L10(L10.Count - 2).Caption + 10
      a = 0
   End If
Next
Data1.RecordSource = "Select Distinct Run From PrintoutData"
Data1.Refresh
Call Ado2
End Sub

Private Sub Text1_Change()
Adodc1.RecordSource = "Select * From PrintoutData Where Run = '" & Text1 & "' Order By
InputData"
Adodc1.Refresh
If Adodc1.Recordset.RecordCount > 0 Then
   Data1.Recordset.FindFirst ("Run = '" & Text1 & "'")
   Call Ado2
   Else
   Label10 = "Record No.  * / " & Data1.Recordset.RecordCount
End If
   Call PGraph
End Sub

Sub PGraph()
      Do While C.Count > 1
            Unload C(C.Count - 1): Unload M(M.Count - 1): Unload Y(Y.Count - 1): Unload K(K.Count
- 1)
            If LC.Count > 1 Then Unload LC(LC.Count - 1): Unload LM(LM.Count - 1): Unload
LY(LY.Count - 1): Unload LK(LK.Count - 1)
      Loop
   If Not Adodc1.Recordset.EOF Then
      Dim CLeft As Long: Dim MLeft As Long: Dim YLeft As Long: Dim KLeft As Long
      Do While Not Adodc1.Recordset.EOF
         CLeft = (Adodc1.Recordset("FilmCy") * 45)
         MLeft = (Adodc1.Recordset("FilmMa") * 45)
         YLeft = (Adodc1.Recordset("FilmYe") * 45)
         KLeft = (Adodc1.Recordset("FilmBk") * 45)
         Load C(C.Count): C(C.Count - 1).Visible = True: C(C.Count - 1).Left = CLeft + (675)
         Load M(M.Count): M(M.Count - 1).Visible = True: M(M.Count - 1).Left = MLeft + (675)
         Load Y(Y.Count): Y(Y.Count - 1).Visible = True: Y(Y.Count - 1).Left = YLeft + (675)
         Load K(K.Count): K(K.Count - 1).Visible = True: K(K.Count - 1).Left = KLeft + (675)
         C(C.Count - 1).Top = 6075 - (((Adodc1.Recordset("DensityCy") * 100) * 45) + 180) / 2
         M(M.Count - 1).Top = 6075 - (((Adodc1.Recordset("DensityMa") * 100) * 45) + 180) / 2
         Y(Y.Count - 1).Top = 6075 - (((Adodc1.Recordset("DensityYe") * 100) * 45) + 180) / 2
         K(K.Count - 1).Top = 6075 - (((Adodc1.Recordset("DensityBk") * 100) * 45) + 180) / 2
         Adodc1.Recordset.MoveNext
      Loop
      Dim i As Byte
      For i = 1 To C.Count - 2
         Load LC(LC.Count): LC(LC.Count - 1).Visible = True
         LC(LC.Count - 1).X1 = C(i).Left + 70
         LC(LC.Count - 1).X2 = C(i + 1).Left + 70
         LC(LC.Count - 1).Y1 = C(i).Top + 70
         LC(LC.Count - 1).Y2 = C(i + 1).Top + 70
         Load LM(LM.Count): LM(LM.Count - 1).Visible = True
         LM(LM.Count - 1).X1 = M(i).Left + 70
         LM(LM.Count - 1).X2 = M(i + 1).Left + 70
```

```
        LM(LM.Count - 1).Y1 = M(i).Top + 70
        LM(LM.Count - 1).Y2 = M(i + 1).Top + 70
        Load LY(LY.Count): LY(LY.Count - 1).Visible = True
        LY(LY.Count - 1).X1 = Y(i).Left + 70
        LY(LY.Count - 1).X2 = Y(i + 1).Left + 70
        LY(LY.Count - 1).Y1 = Y(i).Top + 70
        LY(LY.Count - 1).Y2 = Y(i + 1).Top + 70
        Load LK(LK.Count): LK(LK.Count - 1).Visible = True
        LK(LK.Count - 1).X1 = K(i).Left + 70
        LK(LK.Count - 1).X2 = K(i + 1).Left + 70
        LK(LK.Count - 1).Y1 = K(i).Top + 70
        LK(LK.Count - 1).Y2 = K(i + 1).Top + 70
      Next
   End If
Adodc1.Refresh
End Sub

Sub Ado2()
If Data1.Recordset.RecordCount > 0 Then
   If Data1.Recordset.EOF Then Data1.Recordset.MoveFirst
   If Data1.Recordset.BOF Then Data1.Recordset.MoveLast
   Label10 = "Record No.  " & Data1.Recordset.AbsolutePosition + 1 & " / " &
Data1.Recordset.RecordCount
   Text1 = Data1.Recordset("Run")
   Else
   Label10 = "Record No.  0 / 0"
   Text1 = ""
   Exit Sub
End If
   Adodc1.RecordSource = "Select * From PrintOutData Where Run = '" & Data1.Recordset("Run") &
"' Order By InputData": Adodc1.Refresh
End Sub

Private Sub Command1_Click()
Unload Me
End Sub

Private Sub Command2_Click()
Me.WindowState = 1
End Sub

Private Sub Command4_Click()
If Not Data1.Recordset.EOF And Not Data1.Recordset.BOF Then
   Data1.Recordset.MovePrevious
   Else
   If Data1.Recordset.RecordCount > 0 Then Data1.Recordset.MoveFirst
End If
   Call Ado2
End Sub

Private Sub Command5_Click()
If Not Data1.Recordset.EOF Then
   Data1.Recordset.MoveNext
   Else
   If Data1.Recordset.RecordCount > 0 Then Data1.Recordset.MoveLast
End If
   Call Ado2
```

```
End Sub

Private Sub Command6_Click()
If Data1.Recordset.RecordCount > 0 Then
   Data1.Recordset.MoveFirst
End If
   Call Ado2
End Sub

Private Sub Command7_Click()
If Data1.Recordset.RecordCount > 0 Then
   Data1.Recordset.MoveLast
End If
   Call Ado2
End Sub

Private Sub Form_Load()
Dim i As Byte
Dim a As Byte
a = 0
Adodc1.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" & App.Path &
"\Data.mdb;Persist Security Info=False"
Set DataGrid1.DataSource = Adodc1
Data1.DatabaseName = App.Path & "\Data.MDB"
For i = 1 To 100
   a = a + 1
   Load L3(L3.Count)
   L3(L3.Count - 1).Visible = True
   L3(L3.Count - 1).X1 = 600
   L3(L3.Count - 1).Y1 = L3(L3.Count - 2).Y1 - 45
   L3(L3.Count - 1).Y2 = L3(L3.Count - 2).Y2 - 45
   Load L4(L4.Count)
   L4(L4.Count - 1).Visible = True
   L4(L3.Count - 1).Y2 = 6165
   L4(L4.Count - 1).X1 = L4(L4.Count - 2).X1 + 45
   L4(L4.Count - 1).X2 = L4(L4.Count - 2).X2 + 45
   If a = 10 Then
      L3(L3.Count - 1).X1 = 560
      L4(L3.Count - 1).Y2 = 6210
      Load L9(L9.Count)
      If Ln3.Count < 10 Then
         Load Ln3(Ln3.Count)
         Load Ln4(Ln4.Count)
         Ln3(Ln3.Count - 1).Visible = True
         Ln4(Ln4.Count - 1).Visible = True
         Ln3(Ln3.Count - 1).Y1 = Ln3(Ln3.Count - 2).Y1 - 450
         Ln3(Ln3.Count - 1).Y2 = Ln3(Ln3.Count - 2).Y2 - 450
         Ln4(Ln4.Count - 1).X1 = Ln4(Ln4.Count - 2).X1 + 450
         Ln4(Ln4.Count - 1).X2 = Ln4(Ln4.Count - 2).X2 + 450
      End If
      L9(L9.Count - 1).Visible = True
      L9(L9.Count - 1).Top = L3(L3.Count - 1).Y1 - 110
      L9(L9.Count - 1).Caption = Format(L9(L9.Count - 2).Caption + 0.2, "0.00")
      Load L10(L10.Count)
      L10(L10.Count - 1).Visible = True
      L10(L10.Count - 1).Left = L4(L4.Count - 1).X1 - 125
      L10(L10.Count - 1).Caption = L10(L10.Count - 2).Caption + 10
```

```
      a = 0
   End If
Next
Data1.RecordSource = "Select Distinct Run From PrintoutData"
Data1.Refresh
Call Ado2
End Sub

Private Sub Text1_Change()
Adodc1.RecordSource = "Select * From PrintoutData Where Run = '" & Text1 & "' Order By
InputData"
Adodc1.Refresh
If Adodc1.Recordset.RecordCount > 0 Then
   Data1.Recordset.FindFirst ("Run = '" & Text1 & "'")
   Call Ado2
   Else
   Label10 = "Record No.  * / " & Data1.Recordset.RecordCount
End If
   Call PGraph
End Sub

Sub PGraph()
      Do While C.Count > 1
            Unload C(C.Count - 1): Unload M(M.Count - 1): Unload Y(Y.Count - 1): Unload K(K.Count
- 1)
            If LC.Count > 1 Then Unload LC(LC.Count - 1): Unload LM(LM.Count - 1): Unload
LY(LY.Count - 1): Unload LK(LK.Count - 1)
      Loop
   If Not Adodc1.Recordset.EOF Then
      Dim CLeft As Long: Dim MLeft As Long: Dim YLeft As Long: Dim KLeft As Long
      Do While Not Adodc1.Recordset.EOF
         CLeft = Adodc1.Recordset("FilmCy") * 45
         MLeft = Adodc1.Recordset("FilmMa") * 45
         YLeft = Adodc1.Recordset("FilmYe") * 45
         KLeft = Adodc1.Recordset("FilmBk") * 45
         Load C(C.Count): C(C.Count - 1).Visible = True: C(C.Count - 1).Left = CLeft + (495 - 70)
         Load M(M.Count): M(M.Count - 1).Visible = True: M(M.Count - 1).Left = MLeft + (495 - 70)
         Load Y(Y.Count): Y(Y.Count - 1).Visible = True: Y(Y.Count - 1).Left = YLeft + (495 - 70)
         Load K(K.Count): K(K.Count - 1).Visible = True: K(K.Count - 1).Left = KLeft + (495 - 70)
         C(C.Count - 1).Top = 6075 - ((Adodc1.Recordset("PrintCy") * 45) + 70)
         M(M.Count - 1).Top = 6075 - ((Adodc1.Recordset("PrintMa") * 45) + 70)
         Y(Y.Count - 1).Top = 6075 - ((Adodc1.Recordset("PrintYe") * 45) + 70)
         K(K.Count - 1).Top = 6075 - ((Adodc1.Recordset("PrintBk") * 45) + 70)
         Adodc1.Recordset.MoveNext
      Loop
      Dim i As Byte
      For i = 1 To C.Count - 2
         Load LC(LC.Count): LC(LC.Count - 1).Visible = True
         LC(LC.Count - 1).X1 = C(i).Left + 70
         LC(LC.Count - 1).X2 = C(i + 1).Left + 70
         LC(LC.Count - 1).Y1 = C(i).Top + 70
         LC(LC.Count - 1).Y2 = C(i + 1).Top + 70
         Load LM(LM.Count): LM(LM.Count - 1).Visible = True
         LM(LM.Count - 1).X1 = M(i).Left + 70
         LM(LM.Count - 1).X2 = M(i + 1).Left + 70
         LM(LM.Count - 1).Y1 = M(i).Top + 70
         LM(LM.Count - 1).Y2 = M(i + 1).Top + 70
```

```
            Load LY(LY.Count): LY(LY.Count - 1).Visible = True
            LY(LY.Count - 1).X1 = Y(i).Left + 70
            LY(LY.Count - 1).X2 = Y(i + 1).Left + 70
            LY(LY.Count - 1).Y1 = Y(i).Top + 70
            LY(LY.Count - 1).Y2 = Y(i + 1).Top + 70
            Load LK(LK.Count): LK(LK.Count - 1).Visible = True
            LK(LK.Count - 1).X1 = K(i).Left + 70
            LK(LK.Count - 1).X2 = K(i + 1).Left + 70
            LK(LK.Count - 1).Y1 = K(i).Top + 70
            LK(LK.Count - 1).Y2 = K(i + 1).Top + 70
        Next
    End If
Adodc1.Refresh
End Sub

Sub Ado2()
If Data1.Recordset.RecordCount > 0 Then
    If Data1.Recordset.EOF Then Data1.Recordset.MoveFirst
    If Data1.Recordset.BOF Then Data1.Recordset.MoveLast
    Label10 = "Record No.  " & Data1.Recordset.AbsolutePosition + 1 & " / " &
Data1.Recordset.RecordCount
    Text1 = Data1.Recordset("Run")
    Else
    Label10 = "Record No.  0 / 0"
    Text1 = ""
    Exit Sub
End If
    Adodc1.RecordSource = "Select * From PrintOutData Where Run = '" & Data1.Recordset("Run") &
"' Order By InputData": Adodc1.Refresh
End Sub

Private Sub Command1_Click()
Unload Me
End Sub

Private Sub Command2_Click()
Me.WindowState = 1
End Sub

Private Sub Command4_Click()
If Not Data1.Recordset.EOF And Not Data1.Recordset.BOF Then
    Data1.Recordset.MovePrevious
    Else
    If Data1.Recordset.RecordCount > 0 Then Data1.Recordset.MoveFirst
End If
    Call Ado2
End Sub

Private Sub Command5_Click()
If Not Data1.Recordset.EOF And Not Data1.Recordset.BOF Then
    Data1.Recordset.MoveNext
    Else
    If Data1.Recordset.RecordCount > 0 Then Data1.Recordset.MoveLast
End If
    Call Ado2
End Sub
```

```
Private Sub Command6_Click()
If Data1.Recordset.RecordCount > 0 Then
   Data1.Recordset.MoveFirst
End If
   Call Ado2
End Sub

Private Sub Command7_Click()
If Data1.Recordset.RecordCount > 0 Then
   Data1.Recordset.MoveLast
End If
   Call Ado2
End Sub

Private Sub Form_Load()
Dim i As Byte
Dim a As Byte
Adodc1.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" & App.Path &
"\Data.mdb;Persist Security Info=False"
Set DataGrid1.DataSource = Adodc1
Data1.DatabaseName = App.Path & "\Data.MDB"
a = 0
For i = 1 To 100
   a = a + 1
   Load L3(L3.Count)
   L3(L3.Count - 1).Visible = True
   L3(L3.Count - 1).X1 = 420
   L3(L3.Count - 1).Y1 = L3(L3.Count - 2).Y1 - 45
   L3(L3.Count - 1).Y2 = L3(L3.Count - 2).Y2 - 45
   Load L4(L4.Count)
   L4(L4.Count - 1).Visible = True
   L4(L3.Count - 1).Y2 = 6165
   L4(L4.Count - 1).X1 = L4(L4.Count - 2).X1 + 45
   L4(L4.Count - 1).X2 = L4(L4.Count - 2).X2 + 45
   If a = 10 Then
      L3(L3.Count - 1).X1 = 380
      L4(L3.Count - 1).Y2 = 6210
      Load L9(L9.Count)
      If Ln3.Count < 10 Then
         Load Ln3(Ln3.Count)
         Load Ln4(Ln4.Count)
         Ln3(Ln3.Count - 1).Visible = True
         Ln4(Ln4.Count - 1).Visible = True
         Ln3(Ln3.Count - 1).Y1 = Ln3(Ln3.Count - 2).Y1 - 450
         Ln3(Ln3.Count - 1).Y2 = Ln3(Ln3.Count - 2).Y2 - 450
         Ln4(Ln4.Count - 1).X1 = Ln4(Ln4.Count - 2).X1 + 450
         Ln4(Ln4.Count - 1).X2 = Ln4(Ln4.Count - 2).X2 + 450
      End If
      L9(L9.Count - 1).Visible = True
      L9(L9.Count - 1).Top = L3(L3.Count - 1).Y1 - 110
      L9(L9.Count - 1).Caption = L9(L9.Count - 2).Caption + 10
      Load L10(L10.Count)
      L10(L10.Count - 1).Visible = True
      L10(L10.Count - 1).Left = L4(L4.Count - 1).X1 - 125
      L10(L10.Count - 1).Caption = L10(L10.Count - 2).Caption + 10
      a = 0
   End If
```

```
Next
Data1.RecordSource = "Select Distinct Run From PrintoutData"
Data1.Refresh
Call Ado2
End Sub

Private Sub Text1_Change()
Adodc1.RecordSource = "Select * From PrintoutData Where Run = '" & Text1 & "' Order By
InputData"
Adodc1.Refresh
If Adodc1.Recordset.RecordCount > 0 Then
   Data1.Recordset.FindFirst ("Run = '" & Text1 & "'")
   Call Ado2
   Else
   Label10 = "Record No.  * / " & Data1.Recordset.RecordCount
End If
   Call PGraph
End Sub

Sub PGraph()
      Do While C.Count > 1
            Unload C(C.Count - 1): Unload M(M.Count - 1): Unload Y(Y.Count - 1): Unload K(K.Count
- 1)
            If LC.Count > 1 Then Unload LC(LC.Count - 1): Unload LM(LM.Count - 1): Unload
LY(LY.Count - 1): Unload LK(LK.Count - 1)
      Loop
   If Not Adodc1.Recordset.EOF Then
      Dim CLeft As Long: Dim MLeft As Long: Dim YLeft As Long: Dim KLeft As Long
      Do While Not Adodc1.Recordset.EOF
         CLeft = (Adodc1.Recordset("D3c") * 50) * 45
         MLeft = (Adodc1.Recordset("D3c") * 50) * 45
         YLeft = (Adodc1.Recordset("D3c") * 50) * 45
         KLeft = (Adodc1.Recordset("D3c") * 50) * 45
         Load C(C.Count): C(C.Count - 1).Visible = True: C(C.Count - 1).Left = CLeft + (495 - 70)
         Load M(M.Count): M(M.Count - 1).Visible = True: M(M.Count - 1).Left = MLeft + (495 - 70)
         Load Y(Y.Count): Y(Y.Count - 1).Visible = True: Y(Y.Count - 1).Left = YLeft + (495 - 70)
         Load K(K.Count): K(K.Count - 1).Visible = True: K(K.Count - 1).Left = KLeft + (495 - 70)
         C(C.Count - 1).Top = 6075 - ((Adodc1.Recordset("DotCy") * 45) + 70)
         M(M.Count - 1).Top = 6075 - ((Adodc1.Recordset("DotMa") * 45) + 70)
         Y(Y.Count - 1).Top = 6075 - ((Adodc1.Recordset("DotYe") * 45) + 70)
         K(K.Count - 1).Top = 6075 - ((Adodc1.Recordset("K") * 45) + 70)
         Adodc1.Recordset.MoveNext
      Loop
      Dim i As Byte
      For i = 1 To C.Count - 2
         Load LC(LC.Count): LC(LC.Count - 1).Visible = True
         LC(LC.Count - 1).X1 = C(i).Left + 70
         LC(LC.Count - 1).X2 = C(i + 1).Left + 70
         LC(LC.Count - 1).Y1 = C(i).Top + 70
         LC(LC.Count - 1).Y2 = C(i + 1).Top + 70
         Load LM(LM.Count): LM(LM.Count - 1).Visible = True
         LM(LM.Count - 1).X1 = M(i).Left + 70
         LM(LM.Count - 1).X2 = M(i + 1).Left + 70
         LM(LM.Count - 1).Y1 = M(i).Top + 70
         LM(LM.Count - 1).Y2 = M(i + 1).Top + 70
         Load LY(LY.Count): LY(LY.Count - 1).Visible = True
         LY(LY.Count - 1).X1 = Y(i).Left + 70
```

```
        LY(LY.Count - 1).X2 = Y(i + 1).Left + 70
        LY(LY.Count - 1).Y1 = Y(i).Top + 70
        LY(LY.Count - 1).Y2 = Y(i + 1).Top + 70
        Load LK(LK.Count): LK(LK.Count - 1).Visible = True
        LK(LK.Count - 1).X1 = K(i).Left + 70
        LK(LK.Count - 1).X2 = K(i + 1).Left + 70
        LK(LK.Count - 1).Y1 = K(i).Top + 70
        LK(LK.Count - 1).Y2 = K(i + 1).Top + 70
      Next
   End If
Adodc1.Refresh
End Sub

Sub Ado2()
If Data1.Recordset.RecordCount > 0 Then
   If Data1.Recordset.EOF Then Data1.Recordset.MoveFirst
   If Data1.Recordset.BOF Then Data1.Recordset.MoveLast
   Label10 = "Record No.  " & Data1.Recordset.AbsolutePosition + 1 & " / " &
Data1.Recordset.RecordCount
   Text1 = Data1.Recordset("Run")
   Else
   Label10 = "Record No.  0 / 0"
   Text1 = ""
   Exit Sub
End If
   Adodc1.RecordSource = "Select * From GreyBalance Where Run = '" & Data1.Recordset("Run") &
"' Order By FilmCy": Adodc1.Refresh
End Sub

Private Sub Command1_Click()
Unload Me
End Sub

Private Sub Command2_Click()
Me.WindowState = 1
End Sub


Private Sub Command4_Click()
If Not Data1.Recordset.EOF And Not Data1.Recordset.BOF Then
   Data1.Recordset.MovePrevious
   Else
   If Data1.Recordset.RecordCount > 0 Then Data1.Recordset.MoveFirst
End If
   Call Ado2
End Sub

Private Sub Command5_Click()
If Not Data1.Recordset.EOF And Not Data1.Recordset.BOF Then
   Data1.Recordset.MoveNext
   Else
   If Data1.Recordset.RecordCount > 0 Then Data1.Recordset.MoveLast
End If
   Call Ado2
End Sub

Private Sub Command6_Click()
```

```
If Data1.Recordset.RecordCount > 0 Then
   Data1.Recordset.MoveFirst
End If
   Call Ado2
End Sub

Private Sub Command7_Click()
If Data1.Recordset.RecordCount > 0 Then
   Data1.Recordset.MoveLast
End If
   Call Ado2
End Sub

Private Sub Form_Activate()
Dim Rs As New ADODB.Recordset
Dim K As Double
Dim DotK As Double: Dim DenK As Double
Dim DotKUp As Double: Dim DotKDown As Double
Dim DenKUp As Double: Dim DenKDown As Double
Set Rs = GraphDoVSDensity.Adodc1.Recordset
Adodc1.Refresh
Adodc1.Refresh
Do While Not Adodc1.Recordset.EOF
   Adodc1.Recordset("K") = 0#
   Adodc1.Recordset.Update
   Adodc1.Recordset.MoveNext
Loop
Adodc1.Refresh
Adodc1.Refresh
If Rs.RecordCount > 0 Then Rs.MoveFirst
Do While Not Adodc1.Recordset.EOF
   K = Adodc1.Recordset("D3c")
   If K > 0 Then
   Do While Not Rs.EOF
      If K <= Rs("DensityBk") Then
            Rs.MovePrevious
            DenKUp = Rs("DensityBk")
            DotKUp = Rs("FilmBk")
            Rs.MoveNext
            DenKDown = Rs("DensityBk")
            DotKDown = Rs("FilmBk")
            DotK = DotKUp - DotKDown
            DenK = DenKUp - DenKDown
            K = ((K - DenKDown) * (DotK / DenK)) + DotKDown
        Adodc1.Recordset("K") = K
        Adodc1.Recordset.Update
        Exit Do
      End If
      Rs.MoveNext
   Loop
   End If
   If Rs.RecordCount > 0 Then Rs.MoveFirst
   Adodc1.Recordset.MoveNext
Loop
Adodc1.Refresh
Call Ado2
End Sub
```

```
Private Sub Form_Load()
Dim i As Byte
Dim a As Byte
GraphDoVSDensity.Show
GraphDoVSDensity.WindowState = 1
Me.WindowState = 2
Adodc1.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" & App.Path &
"\Data.mdb;Persist Security Info=False"
Set DataGrid1.DataSource = Adodc1
Data1.DatabaseName = App.Path & "\Data.MDB"
Set DataGrid2.DataSource = GraphDoVSDensity.Adodc1
a = 0
For i = 1 To 100
   a = a + 1
   Load L3(L3.Count)
   L3(L3.Count - 1).Visible = True
   L3(L3.Count - 1).X1 = 420
   L3(L3.Count - 1).Y1 = L3(L3.Count - 2).Y1 - 45
   L3(L3.Count - 1).Y2 = L3(L3.Count - 2).Y2 - 45
   Load L4(L4.Count)
   L4(L4.Count - 1).Visible = True
   L4(L3.Count - 1).Y2 = 6165
   L4(L4.Count - 1).X1 = L4(L4.Count - 2).X1 + 45
   L4(L4.Count - 1).X2 = L4(L4.Count - 2).X2 + 45
   If a = 10 Then
      L3(L3.Count - 1).X1 = 380
      L4(L3.Count - 1).Y2 = 6210
      Load L9(L9.Count)
      If Ln3.Count < 10 Then
         Load Ln3(Ln3.Count)
         Load Ln4(Ln4.Count)
         Ln3(Ln3.Count - 1).Visible = True
         Ln4(Ln4.Count - 1).Visible = True
         Ln3(Ln3.Count - 1).Y1 = Ln3(Ln3.Count - 2).Y1 - 450
         Ln3(Ln3.Count - 1).Y2 = Ln3(Ln3.Count - 2).Y2 - 450
         Ln4(Ln4.Count - 1).X1 = Ln4(Ln4.Count - 2).X1 + 450
         Ln4(Ln4.Count - 1).X2 = Ln4(Ln4.Count - 2).X2 + 450
      End If
      L9(L9.Count - 1).Visible = True
      L9(L9.Count - 1).Top = L3(L3.Count - 1).Y1 - 110
      L9(L9.Count - 1).Caption = L9(L9.Count - 2).Caption + 10
      Load L10(L10.Count)
      L10(L10.Count - 1).Visible = True
      L10(L10.Count - 1).Left = L4(L4.Count - 1).X1 - 180
      L10(L10.Count - 1).Caption = Format(L10(L10.Count - 2).Caption + 0.2, "0.00")
      a = 0
   End If
Next
Data1.RecordSource = "Select Distinct Run From GreyBalance"
Data1.Refresh
End Sub

Private Sub Text1_Change()
Adodc1.RecordSource = "Select * From GreyBalance Where Run = '" & Text1 & "' Order By FilmCy"
Adodc1.Refresh
If Adodc1.Recordset.RecordCount > 0 Then
```

```
   Data1.Recordset.FindFirst ("Run = '" & Text1 & "'")
   Call Ado2
   Else
   Label10 = "Record No.  * / " & Data1.Recordset.RecordCount
End If
   Call PGraph
End Sub

Dim Pos As Long
Dim KeyEnter As Integer

Public Sub Search(Inx As Integer)
Adodc1.RecordSource = "Select * From PrintOUTData Where Run = '" & Text1 & "' Order By
FilmCy"
Adodc1.Refresh
If Adodc1.Recordset.EOF Then Exit Sub
Dim Var1 As Integer
Dim Var2 As Integer
Dim Var3 As Integer
Dim Var4 As Integer
Do While Not Adodc1.Recordset.EOF
   If Val(Text(Inx).Text) < Adodc1.Recordset("InPutData") Then
      Adodc1.Recordset.MovePrevious
      If Adodc1.Recordset.EOF Then Exit Sub
      Var1 = Adodc1.Recordset("InPutData")
      Var3 = Adodc1.Recordset(Inx + 2)
      Adodc1.Recordset.MoveNext
      If Adodc1.Recordset.EOF Then Exit Sub
      Var2 = Adodc1.Recordset("InPutData")
      Var4 = Adodc1.Recordset(Inx + 2)
      Exit Do
   End If
   If Val(Text(Inx).Text) = Adodc1.Recordset("InPutData") Then
      Text(Inx + 3) = Adodc1.Recordset(Inx + 2)
      Exit Sub
   End If
      Adodc1.Recordset.MoveNext
Loop
Dim Var5 As Integer
Dim Var6 As Integer
Dim Var7 As Integer
Var5 = Var2 - Var1
Var6 = Var4 - Var3
Var7 = ((Var6 / Var5) * (Text(Inx) - Var1)) + Var3
Text(Inx + 3) = Var7
End Sub

Private Sub CloseForm_Click()
Unload Me
End Sub

Private Sub Form_Load()
Adodc1.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" & App.Path &
"\Data.mdb;Persist Security Info=False"
Adodc2.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" & App.Path &
"\Data.mdb;Persist Security Info=False"
Adodc1.RecordSource = "Select Distinct Run From QGreyBalance"
```

```
Adodc1.Refresh
Dim i As Byte
For i = 0 To 6
   G1.ColAlignment(i) = 5
   G1.ColWidth(i) = 1050
Next
G1.TextMatrix(0, 0) = "Cy": G1.TextMatrix(0, 1) = "Ma": G1.TextMatrix(0, 2) = "Ye"
G1.TextMatrix(0, 3) = "Cy": G1.TextMatrix(0, 4) = "Ma": G1.TextMatrix(0, 5) = "Ye"
G1.TextMatrix(0, 6) = "D3c": G1.Rows = 1
If Not Adodc1.Recordset.EOF Then
Pos = 1: Text1 = Adodc1.Recordset("run")
Else
Pos = 0: Text1 = ""
End If
Label3.Caption = "ข้อมูลรายการที่  " & Pos & " / " & Adodc1.Recordset.RecordCount
End Sub


Private Sub Text_KeyDown(Index As Integer, KeyCode As Integer, Shift As Integer)
If KeyCode = 17 Then KeyEnter = KeyCode
If KeyCode = 13 And KeyEnter = 17 Then
   G1.AddItem Text(0) & vbTab & Text(1) & vbTab & Text(2) & vbTab & Text(3) & vbTab &
Text(4) & vbTab & Text(5) & vbTab & Text(6)
Text(0).SetFocus
Dim i As Byte
   For i = 0 To 6
      Text(i) = ""
   Next
KeyEnter = 0
End If
End Sub

Private Sub Text_KeyPress(Index As Integer, KeyAscii As Integer)
If KeyAscii = 8 And Text(Index).SelStart = 0 Then
   If Index = 0 Then
      Text(Text.Count - 1).SetFocus
      Else
      Text(Index - 1).SetFocus
   End If
End If
If KeyAscii = 13 Then
   If Index < 6 Then Text(Index + 1).SetFocus
   If Index >= 6 Then Text(0).SetFocus
End If
If KeyAscii >= 48 And KeyAscii <= 57 Or KeyAscii = 8 Or KeyAscii = 46 Then Exit Sub
   KeyAscii = 0
End Sub

Private Sub Text_LostFocus(Index As Integer)
If Index = 0 Then Call Search(Index)
If Index = 1 Then Call Search(Index)
If Index = 2 Then Call Search(Index)
End Sub

Private Sub Text1_Change()
Adodc2.RecordSource = "Select * From GreyBalance Where Run = '" & Text1 & "'": Adodc2.Refresh
Call ReGrid
```

```
End Sub

Private Sub Text_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)
KeyEnter = 0
End Sub

Public Sub ReGrid()
G1.Rows = 1
If Not Adodc2.Recordset.EOF Then
Dim Rcount As Long: Dim i As Long
Rcount = Adodc2.Recordset.RecordCount
With Adodc2.Recordset
For i = 1 To Rcount
   G1.AddItem .Fields(1) & vbTab & .Fields(2) & vbTab & .Fields(3) & vbTab & .Fields(4) & vbTab
& .Fields(5) & vbTab & .Fields(6) & vbTab & .Fields(7)
    Adodc2.Recordset.MoveNext
Next
End With
End If
End Sub

Private Sub Command1_Click(Index As Integer)
Text(0).SetFocus
End Sub

Private Sub Command2_Click()
   G1.AddItem Text(0) & vbTab & Text(1) & vbTab & Text(2) & vbTab & Text(3) & vbTab &
Text(4) & vbTab & Text(5) & vbTab & Text(6)
Dim i As Byte
   For i = 0 To 6
      Text(i) = ""
   Next
Text(0).SetFocus
End Sub

Private Sub Command3_Click()
If G1.Rows <= 2 Then
   G1.Rows = 1
   Else
   G1.RemoveItem (G1.Row)
End If
End Sub

Private Sub Command4_Click()
If Pos <= 1 Then Exit Sub
Adodc1.Recordset.MovePrevious
Pos = Pos - 1
Label3.Caption = "ข้อมูลรายการที่  " & Pos & " / " & Adodc1.Recordset.RecordCount
Text1 = Adodc1.Recordset("Run")
End Sub

Private Sub Command5_Click()
If Pos >= Adodc1.Recordset.RecordCount Then Exit Sub
Adodc1.Recordset.MoveNext
If Adodc1.Recordset.EOF Then Exit Sub
Pos = Pos + 1
Label3.Caption = "ข้อมูลรายการที่  " & Pos & " / " & Adodc1.Recordset.RecordCount
```

```
Text1 = Adodc1.Recordset("Run")
End Sub

Private Sub Command6_Click()
On Error GoTo Err_
Pos = 0
Text1 = ""
Adodc1.Recordset.MoveFirst
Text1 = Adodc1.Recordset("Run")
Pos = 1
Err_:
Label3.Caption = "ข้อมูลรายการที่  " & Pos & " / " & Adodc1.Recordset.RecordCount
End Sub

Private Sub Command7_Click()
On Error GoTo Err_
Text1 = ""
Adodc1.Recordset.MoveLast
Text1 = Adodc1.Recordset("Run")
Err_:
Pos = Adodc1.Recordset.RecordCount
Label3.Caption = "ข้อมูลรายการที่  " & Pos & " / " & Adodc1.Recordset.RecordCount
End Sub


Private Sub NewData_Click()
Adodc1.Refresh
If Adodc1.Recordset.EOF Then
   Text1 = "000001"
   Label3 = "ข้อมูลรายการที่  New / 0"
   Else
   Adodc1.Recordset.MoveLast
   Text1 = Format(Val(Adodc1.Recordset("Run")) + 1, "000000")
   Label3 = "ข้อมูลรายการที่  New / " & Adodc1.Recordset.RecordCount
   Pos = Adodc1.Recordset.RecordCount + 1
   Adodc1.Recordset.MoveNext
End If
Text(0).SetFocus
End Sub

Private Sub SaveData_Click()
If Text1 = "" Then MsgBox "ยกเลิกการบันทึก โปรดเลือกรายการ New Data ก่อนบันทึก", vbCritical + vbOKOnly,
"ยกเลิกการบันทึก": Exit Sub
If G1.Rows = 1 Then MsgBox "ยกเลิกการบันทึก โปรดเพิ่มรายการก่อนบันทึก", vbCritical + vbOKOnly, "ยกเลิกการ
บันทึก": Exit Sub
If MsgBox("ยืนยันการบันทึกข้อมูล คลิกปุ่ม Yes ยกเลิกการบันทึกคลลิกปุ่ม No", vbInformation + vbYesNo, "Save Data")
= vbNo Then Exit Sub
Adodc2.RecordSource = "Select * From GreyBalance Where Run = '" & Text1 & "'"
Adodc2.Refresh
Do While Not Adodc2.Recordset.EOF
   Adodc2.Recordset.Delete
   Adodc2.Refresh
Loop
Dim i As Integer: Dim a As Byte
With Adodc2.Recordset
```

```
For i = 1 To G1.Rows - 1
   .AddNew
   .Fields(0) = Text1
   For a = 0 To 6
      If G1.TextMatrix(i, a) <> Empty Then .Fields(a + 1) = G1.TextMatrix(i, a)
   Next
   .Update
Next
End With
Adodc2.Refresh: Adodc1.Refresh
Pos = 1
Text1 = Adodc1.Recordset("Run")
Label3.Caption = "ข้อมูลรายการที่  " & Pos & " / " & Adodc1.Recordset.RecordCount
End Sub

Private Sub DeleteData_Click()
If MsgBox("ยืนยันการลบข้อมูล คลิกปุ่ม Yes ยกเลิกการลบคลลิกปุ่ม No", vbInformation + vbYesNo, "Delete Data") =
vbNo Then Exit Sub
Adodc2.RecordSource = "Select * From GreyBalance Where Run = '" & Text1 & "'"
Adodc2.Refresh
Adodc2.Refresh
If Adodc2.Recordset.EOF Then MsgBox "ไม่สามารถลบข้อมูลรายการนี้ได้ โปรดตรวจสอบข้อมูลใหม่", vbCritical +
vbOKOnly, "ยกเลิกการลบข้อมูล": Exit Sub
Do While Not Adodc2.Recordset.EOF
   Adodc2.Recordset.Delete
   Adodc2.Refresh
Loop
Adodc2.Refresh: Adodc1.Refresh
If Adodc1.Recordset.EOF Then
   Text1 = ""
   G1.Rows = 1
   Pos = 0
   Else
   Text1 = Adodc1.Recordset("Run")
   Pos = 1
End If
Label3.Caption = "ข้อมูลรายการที่  " & Pos & " / " & Adodc1.Recordset.RecordCount
End Sub

Private Sub DenOr_MN_Click()
DensityOfOriginal.Show , Me
End Sub

Private Sub DoVSDensity_MN_Click()
GraphDoVSDensity.Show
End Sub

Private Sub FilmVSPrint_MN_Click()
GraphFilmVSPrint.Show
End Sub

Private Sub GBal_MN_Click()
GreyBalance.Show , Me
End Sub

Private Sub GraphCMYK_MN_Click()
```

```
GraphD34cDk.Show
GraphD34cDk.WindowState = 1
GraphDoVSDensity.Show
GraphDoVSDensity.WindowState = 1
GraphCMYK.Show
GraphCMYK.WindowState = 2
End Sub

Private Sub GraphD34cDk_MN_Click()
GraphD34cDk.Show
End Sub

Private Sub GraphDAOF_MN_Click()
GraphDAOF.Show
End Sub

Private Sub GraphGrey_MN_Click()
GraphGrey.Show
End Sub

Private Sub PoData_MN_Click()
PrintOutData.Show , Me
End Sub

Dim Pos As Long
Dim KeyEnter As Integer

Public Sub ReGrid()
G1.Rows = 1
If Not Adodc2.Recordset.EOF Then
Dim Rcount As Long: Dim i As Long
Rcount = Adodc2.Recordset.RecordCount
With Adodc2.Recordset
For i = 1 To Rcount
   G1.AddItem .Fields(1) & vbTab & .Fields(2) & vbTab & .Fields(3) & vbTab & .Fields(4) & vbTab
& .Fields(5) & vbTab & .Fields(6) & vbTab & .Fields(7) _
    & vbTab & .Fields(8) & vbTab & .Fields(9) & vbTab & Format(.Fields(10), "0.00") & vbTab &
Format(.Fields(11), "0.00") & vbTab & Format(.Fields(12), "0.00") & vbTab & Format(.Fields(13),
"0.00")
    Adodc2.Recordset.MoveNext
Next
End With
End If
End Sub

Private Sub CloseFrom_Click()
Unload Me
End Sub

Private Sub Command1_Click(Index As Integer)
Text(0).SetFocus
End Sub

Private Sub Command2_Click()
Dim i As Byte
Dim a As Byte
For i = 1 To G1.Rows - 1
```

```
   If Text(0) = G1.TextMatrix(i, 0) Then
      For a = 0 To 12
         G1.TextMatrix(i, a) = Text(a)
      Next
      Exit For
   End If
Next
If i = G1.Rows Then
   G1.AddItem Text(0) & vbTab & Text(1) & vbTab & Text(2) & vbTab & Text(3) & vbTab &
Text(4) & vbTab & Text(5) & vbTab & Text(6) _
    & vbTab & Text(7) & vbTab & Text(8) & vbTab & Text(9) & vbTab & Text(10) & vbTab &
Text(11) & vbTab & Text(12)
End If
'Adodc2.RecordSource = "Select * From PrintOutData Where Run = '" & Text1 & "' Order By
InputData": Adodc2.Refresh
'Call ReGrid
   For i = 0 To 12
      Text(i) = ""
   Next
Text(0).SetFocus
End Sub

Private Sub Command3_Click()
If G1.Rows <= 2 Then
   G1.Rows = 1
   Else
   G1.RemoveItem (G1.Row)
End If
End Sub

Private Sub Command4_Click()
If Pos <= 1 Then Exit Sub
Adodc1.Recordset.MovePrevious
Pos = Pos - 1
Label3.Caption = "ข้อมูลรายการที่  " & Pos & " / " & Adodc1.Recordset.RecordCount
Text1 = Adodc1.Recordset("Run")
End Sub

Private Sub Command5_Click()
If Pos >= Adodc1.Recordset.RecordCount Then Exit Sub
Adodc1.Recordset.MoveNext
If Adodc1.Recordset.EOF Then Exit Sub
Pos = Pos + 1
Label3.Caption = "ข้อมูลรายการที่  " & Pos & " / " & Adodc1.Recordset.RecordCount
Text1 = Adodc1.Recordset("Run")
End Sub

Private Sub Command6_Click()
On Error GoTo Err_
Pos = 0
Text1 = ""
Adodc1.Recordset.MoveFirst
Text1 = Adodc1.Recordset("Run")
Pos = 1
Err_:
Label3.Caption = "ข้อมูลรายการที่  " & Pos & " / " & Adodc1.Recordset.RecordCount
```

```
End Sub

Private Sub Command7_Click()
On Error GoTo Err_
Text1 = ""
Adodc1.Recordset.MoveLast
Text1 = Adodc1.Recordset("Run")
Err_:
Pos = Adodc1.Recordset.RecordCount
Label3.Caption = "ข้อมูลรายการที่  " & Pos & " / " & Adodc1.Recordset.RecordCount
End Sub

Private Sub Form_Load()
Dim i As Byte
G1.ColWidth(0) = 845
G1.ColAlignment(0) = 5
For i = 1 To G1.Cols - 1
   G1.ColAlignment(i) = 5
   G1.ColWidth(i) = 720
Next
G1.TextMatrix(0, 0) = "Input": G1.TextMatrix(0, 1) = "Cy": G1.TextMatrix(0, 2) = "Ma":
G1.TextMatrix(0, 3) = "Ye": G1.TextMatrix(0, 4) = "Bk"
G1.TextMatrix(0, 5) = "Cy": G1.TextMatrix(0, 6) = "Ma": G1.TextMatrix(0, 7) = "Ye":
G1.TextMatrix(0, 8) = "Bk"
G1.TextMatrix(0, 9) = "Cy": G1.TextMatrix(0, 10) = "Ma": G1.TextMatrix(0, 11) = "Ye":
G1.TextMatrix(0, 12) = "Bk"
G1.Rows = 1
Adodc1.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" & App.Path &
"\Data.mdb;Persist Security Info=False"
Adodc2.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" & App.Path &
"\Data.mdb;Persist Security Info=False"
Adodc1.RecordSource = "Select Distinct Run From QPrintOutData"
Adodc1.Refresh
If Not Adodc1.Recordset.EOF Then
Pos = 1: Text1 = Adodc1.Recordset("run")
Else
Pos = 0: Text1 = ""
End If
Label3.Caption = "ข้อมูลรายการที่  " & Pos & " / " & Adodc1.Recordset.RecordCount
End Sub

Private Sub G1_DblClick()
Dim i As Byte
For i = 0 To 12
   Text(i) = G1.TextMatrix(G1.Row, i)
Next
End Sub

Private Sub NewData_Click()
Adodc1.Refresh
If Adodc1.Recordset.EOF Then
   Text1 = "000001"
   Label3 = "ข้อมูลรายการที่  New / 0"
   Else
   Adodc1.Recordset.MoveLast
   Text1 = Format(Val(Adodc1.Recordset("Run")) + 1, "000000")
```

```
   Label3 = "ข้อมูลรายการที่  New / " & Adodc1.Recordset.RecordCount
   Pos = Adodc1.Recordset.RecordCount + 1
   Adodc1.Recordset.MoveNext
End If
Text(0).SetFocus
End Sub

Private Sub SaveData_Click()
If Text1 = "" Then MsgBox "ยกเลิกการบันทึก โปรดเลือกรายการ New Data ก่อนบันทึก", vbCritical + vbOKOnly,
"ยกเลิกการบันทึก": Exit Sub
If G1.Rows = 1 Then MsgBox "ยกเลิกการบันทึก โปรดเพิ่มรายการก่อนบันทึก", vbCritical + vbOKOnly, "ยกเลิกการ
บันทึก": Exit Sub
If MsgBox("ยืนยันการบันทึกข้อมูล คลิกปุ่ม Yes ยกเลิกการบันทึกคลิกปุ่ม No", vbInformation + vbYesNo, "Save Data")
= vbNo Then Exit Sub
Adodc2.RecordSource = "Select * From PrintoutData Where Run = '" & Text1 & "' Order By
InputData"
Adodc2.Refresh
Do While Not Adodc2.Recordset.EOF
   Adodc2.Recordset.Delete
   Adodc2.Refresh
Loop
Dim i As Integer: Dim a As Byte
With Adodc2.Recordset
For i = 1 To G1.Rows - 1
   .AddNew
   .Fields(0) = Text1
   For a = 0 To 12
      If G1.TextMatrix(i, a) <> Empty Then .Fields(a + 1) = G1.TextMatrix(i, a)
   Next
   .Update
Next
End With
Adodc2.Refresh: Adodc1.Refresh
Pos = 1
Text1 = Adodc1.Recordset("Run")
Label3.Caption = "ข้อมูลรายการที่  " & Pos & " / " & Adodc1.Recordset.RecordCount
End Sub

Private Sub DeleteData_Click()
If MsgBox("ยืนยันการลบข้อมูล คลิกปุ่ม Yes ยกเลิกการลบคลิกปุ่ม No", vbInformation + vbYesNo, "Delete Data") =
vbNo Then Exit Sub
Adodc2.RecordSource = "Select * From PrintoutData Where Run = '" & Text1 & "' Order By
InputData"
Adodc2.Refresh
If Adodc2.Recordset.EOF Then MsgBox "ไม่สามารถลบข้อมูลรายการนี้ได้ โปรดตรวจสอบข้อมูลใหม่", vbCritical +
vbOKOnly, "ยกเลิกการลบข้อมูล": Exit Sub
Do While Not Adodc2.Recordset.EOF
   Adodc2.Recordset.Delete
   Adodc2.Refresh
Loop
Adodc2.Refresh
Adodc1.Refresh
If Adodc1.Recordset.EOF Then
   Text1 = ""
   G1.Rows = 1
```

```
   Pos = 0
   Else
   Text1 = Adodc1.Recordset("Run")
   Pos = 1
End If
Label3.Caption = "ข้อมูลรายการที่  " & Pos & " / " & Adodc1.Recordset.RecordCount
End Sub

Private Sub Text_KeyDown(Index As Integer, KeyCode As Integer, Shift As Integer)
If KeyCode = 17 Then KeyEnter = KeyCode
If KeyCode = 13 And KeyEnter = 17 Then
   G1.AddItem Text(0) & vbTab & Text(1) & vbTab & Text(2) & vbTab & Text(3) & vbTab &
Text(4) & vbTab & Text(5) & vbTab & Text(6) _
    & vbTab & Text(7) & vbTab & Text(8) & vbTab & Text(9) & vbTab & Text(10) & vbTab &
Text(11) & vbTab & Text(12)
Text(0).SetFocus
Dim i As Byte
   For i = 0 To 12
      Text(i) = ""
   Next
KeyEnter = 0
End If
End Sub

Private Sub Text_KeyPress(Index As Integer, KeyAscii As Integer)
If KeyAscii = 8 And Text(Index).SelStart = 0 Then
   If Index = 0 Then
      Text(Text.Count - 1).SetFocus
      Else
      Text(Index - 1).SetFocus
   End If
End If
If KeyAscii = 13 Then
   If Index < 12 Then Text(Index + 1).SetFocus
   If Index >= 12 Then Text(0).SetFocus
   KeyAscii = 0
End If
If KeyAscii >= 48 And KeyAscii <= 57 Or KeyAscii = 8 Or KeyAscii = 46 Then Exit Sub
End Sub

Private Sub Text_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)
KeyEnter = 0
End Sub

Private Sub Text1_Change()
Adodc2.RecordSource = "Select * From PrintOutData Where Run = '" & Text1 & "' Order By
InputData": Adodc2.Refresh
Call ReGrid
End Sub
```

VITA

Mr. Surush Nuttee was born on May 25, 1972 in Bangkok, Thailand. He received his B.Sc. degree in Packaging Technology from the Department of Packaging Technology, Faculty of Agro-Industry, Kasetsart University in 1995, and he has been a graduate student in the Imaging Technology Program, Graduate school, Chulalongkorn University since 1998.