รายการอ้างอิง

ภาษาไทย

ชูเกียรติ วิเชียรเจริญ. งานรังวัดดาวเทียม : ระบบการทำงานรังวัดในอนาคต. ใน งาน 8 รอบ ศาสตราจารย์พลโทพระยาศัลวิธานนิเทศ. กรุงเทพมหานคร : ม.ป.ท. , 2531

สวัสดิ์ชัย เกรียงไกรเพชร. การใช้ระบบ UTM ในงานรังวัด. วารสารวิศวกรรมสาร(กรกฎาคม 2527): 62-71

ภาษาอังกฤษ

Campbell, J. Introductory Cartography. NewJersy : Prentice-Hall, 1984.

Colwell, S. A year in transition for GPS and conventional total station sales. Geodetical Info. vol.6 no.3 (March 1992) : 29

Harvey, B.R. Transformation of 3D coordinate. The Australian Surveyor vol.33 no.2 (June 1986) : 105-125

Kay, D.E. Graphics File Formats. 1st ed. USA : Windcrest Books, 1992

Lachapelle, G. Precise surveying with a high performance C/A code GPS receiver. Geodetical Info. vol. 6 no. 3 (March 1992) : 47-49.

Landau, H. GPS positioning in real-time with centimetre accuracy. Geodetical Info. vol. 5 no. 9 (September 1991) : 39-42.

Lechner, W. Global navigation satellite systems : Market trends worldwide. Geodetical Info. vol.7 no. 3 (March 1993) : 64-67

Leick, A. GPS Satellite Surveying. 1st ed. New York : John Wiley & Sons, 1990

Richardus, P. Map projections for geodesists, cartographers and geographers. 1st ed. Amsterdam : North-Holland Publishing Co., 1972

บทที่ 6

บทสรุปและข้อเสนอแนะ

การวิจัยศึกษาความเป็นไปได้ในการแสดงตำแหน่งรถยนต์บนแผนที่โดยใช้ GPS สรุปผลว่าเป็นไปได้ที่จะทำระบบดังกล่าว และได้จัดทำตัวอย่างโปรแกรมการแสดงตำแหน่งรถยนต์บนแผนที่โดยใช้ GPS และได้ทำการทดสอบในการปฏิบัติการจริง(ดูรายละเอียดในบทที่ 5) ปรากฏผลว่าสามารถทำงานได้ดีถึงแม้ว่าตำแหน่งรถยนต์ในแผนที่จะคลาดเคลื่อนไปจากถนนบ้าง แต่ก็ยังสามารถทราบได้ว่าเดินทางอยู่บนถนนเส้นใด แต่ถ้าต้องการนำไปใช้งานที่ต้องการความถูกต้องสูงกว่านี้ ควรจะใช้เครื่องรับสัญญาณ GPS ที่มีวิธีการรังวัดที่ได้ค่าพิกัดที่ดีกว่านี้ วิธีการรังวัดที่จะให้ค่าพิกัดที่ดีกว่าที่น่าจะศึกษาคือ วิธีการรังวัดแบบ Kinematic GPS Positioning และการใช้อุปกรณ์เสริมนอกจากการใช้ GPS หาค่าพิกัดแล้วอาจติดอุปกรณ์วัดความเร็วกับทิศทางของรถยนต์เพื่อช่วยในการหาค่าพิกัดขณะที่รับสัญญาณดาวเทียมไม่ได้ นอกจากนี้ระบบแสดงตำแหน่งรถยนต์บนแผนที่สามารถนำไปเพิ่มเติมให้มีการเก็บค่าพิกัดในขณะที่ต้องการ เพื่อนำไปใช้ทำแผนที่หรือปรับปรุงแผนที่ที่มีอยู่ได้อย่างสะดวกและรวดเร็ว

6.1 <u>หัวข้อสรุปที่ได้จากการวิจัย</u>

- มีความเป็นไปได้ในการแสดงตำแหน่งรถยนต์บนแผนที่โดยใช้ GPS
- การใช้ทิศเหนือของแผนที่หันขึ้นด้านบนของจอภาพ ดีกว่าการใช้ทิศทางการเคลื่อนที่หันขึ้นด้านบนของจอภาพ
- มาตราส่วนของแผนที่ที่เหมาะสมในการแสดงตำแหน่งรถยนต์ขึ้นอยู่กับองค์ประกอบหลายอย่างเช่นความหนาแน่นของถนน, ความคลาดเคลื่อนของค่าพิกัด และวิธีการทำแผนที่การกับการใช้สัญญลักษณ์ (ดูรายละเอียดในหัวข้อที่ 3.2 เรื่อง "รูปแบบและมาตราส่วนที่เหมาะสม) โดยควรจะอยู่ในช่วง 1 : 10,000 ถึง 1 : 50,000 สำหรับพื้นที่ในเมือง 1 : 50,000 ถึง 1 : 100,000 สำหรับพื้นที่ชานเมือง และ 1 : 100,000 ถึง 1 : 2,500,000 สำหรับพื้นที่ระหว่างจังหวัด

ภาคผนวก

ภาคผนวก ก

การใช้โปรแกรม

<u>การติดค่าตัวแปร</u>

ตัวแปรของโปรแกรมจะเก็บอยู่ในแฟ้ม "CarMap.Cfg" ซึ่งประกอบด้วยข้อมูลดังนี้

FileProjectName	\thesis\project8.map	File Name of Project Map
ViewScale	2	Scale of Display = Scale of Map * ViewScale
ViewPointScaleX	51360	Points / Meter (X axis)
ViewPointScaleY	51360	Points / Meter (Y axis)
ViewLimitX	1200	No. Points Limit Before Load New Map
ViewLimitY	1200	No. Points Limit Before Load New Map
ViewLimitRotation	50	Angle Limit Before Load New Map (Degree)
NoCarCoordinate	8	No. Stock of CarCoordinate
GPSMode	GPS-Demo	Demo N,E N,E,H E,N E,N,H FixLength
GPSDataColStart	1	Column of Starting Data
GPSDataColOfN	1	Column of North
GPSDataColOfE	13	Column of East
GPSDataColOfH	24	Column of Height
GPSDataStrOfN	N=	Leading Text for North
GPSDataStrOfE	E=	Leading Text for East
GPSDataStrOfH	H=	Leading Text for Height
GPSDataSysErrN	0	Systematric Error (m)
GPSDataSysErrE	0	Systematric Error (m)
GPSDataSysErrH	0	Systematric Error (m)

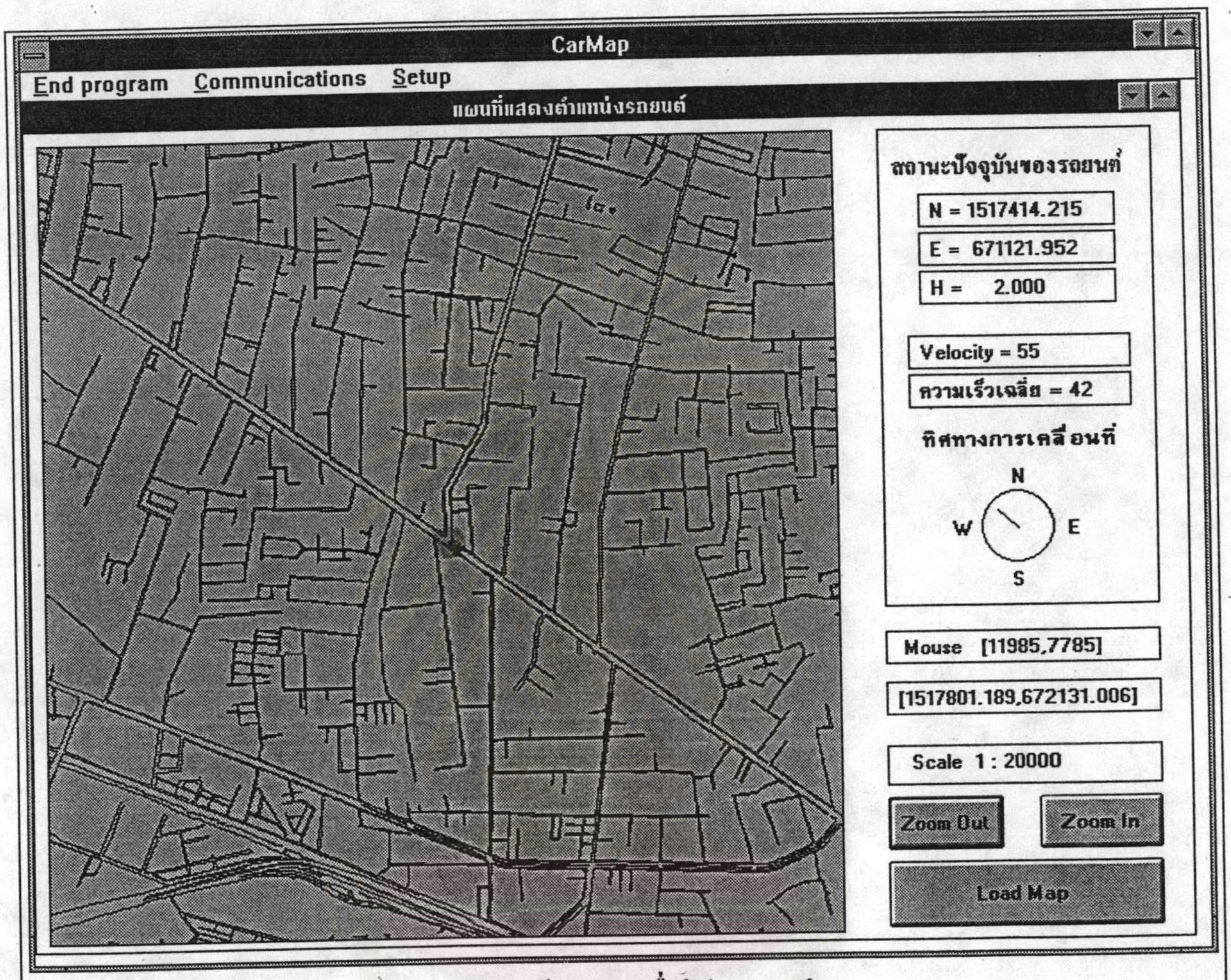
GPSDataRndErrN	15	Random Error (m)
GPSDataRndErrE	15	Random Error (m)
GPSDataRndErrH	15	Random Error (m)
GPSDataDemoSysErrN	0	Systematric Error (m)
GPSDataDemoSysErrE	0	Systematric Error (m)
GPSDataDemoSysErrH	0	Systematric Error (m)
GPSDataDemoRndErrN	10	Random Error (m)
GPSDataDemoRndErrE	10	Random Error (m)
GPSDataDemoRndErrH	10	Random Error (m)
CarMode	3	1 = Icon 2 = Circle 3 = Circle+Direction

โดยแต่ละบรรทัดประกอบด้วยชื่อตัวแปร, ค่าตัวแปร และหมายเหตุ มีตำแหน่งตัวอักษรแน่นอนดังนี้

- ชื่อตัวแปร เริ่มที่ตัวอักษรที่ 1
- ค่าตัวแปร เริ่มที่ตัวอักษรที่ 31
- หมายเหตุ เริ่มที่ตัวอักษรที่ 61

<u>การใช้คำสั่งของโปรแกรม</u>

รูปแบบของหน้าจอภาพเมื่อโปรแกรมทำงานเป็นดังนี้



สถานะปัจจุบันของรถยนต์

N = 1517414.215

E = 671121.952

H = 2.000

Velocity = 55

ความเร็วเฉลี่ย = 42

ทิศทางการเคลื่อนที่

Mouse [11985,7785]

[1517801.189,672131.006]

Scale 1 : 20000

รูปแบบของหน้าจอภาพเมื่อโปรแกรมทำงาน

Zoom Out	การลดมาตราส่วนลงครึ่งหนึ่ง
Zoom In	การขยายมาตราส่วนขึ้นหนึ่งเท่า
Load Map	การเรียกแผนที่จากแผ่นบันทึกข้อมูล

ภาคผนวก ข

รูปแบบการเก็บข้อมูลแผนที่ที่ใช้กับโปรแกรม

แฟ้มข้อมูลของแผนที่ที่นำมาทดสอบอยู่ในรูปของ DXF format (เป็นรูปแบบการเก็บภาพในแบบ Vector ที่เป็นมาตราฐานชนิดหนึ่ง มีใช้กันอย่างแพร่หลายเช่น โปรแกรม Autocad เป็นต้น รูปแบบการเก็บข้อมูลของสามารถหาอ่านในคู่มือของโปรแกรม Autocad) แต่แฟ้มข้อมูลที่จะใช้กับโปรแกรมเป็นรูปแบบ่ของ Microsoft Windows Metafile (WMF) ซึ่งจะต้องทำการแปลงข้อมูลโดยโปรแกรมแปลงข้อมูลภาพเช่น โปรแกรม Hijack เป็นต้น เหตุผลในการเลือกใช้รูปแบบข้อมูลของ WMF

- เก็บรายละเอียดได้ทั้งรูปภาพแบบลายจุด(Rester) และ รูปภาพแบบลายเส้น(Vector)
- มีโปรแกรมที่สามารถใช้รูปแบบข้อมูล WMF ได้มากมาย
- ความละเอียดในการเก็บค่าพิกัดเหมาะสมกับการใช้แสดงภาพบนจอเครื่องคอมพิวเตอร์
- เป็นรูปแบบมาตราฐานที่ใช้ใน Microsoft Windows

Kay และ Levine (1992) กล่าวถึงรายละเอียดของโครงสร้างของข้อมูล WMF ในบทที่ 23 ดังต่อไปนี้

23

WMF

Summary: WMF (Microsoft Windows Metafile)

Image type Display list

Intended use Storage and interchange under MS Windows

Owner Microsoft Corp., 1 Microsoft Way, Redmond WA 98052-6399, 1-800-426-9400

Latest revision/version/release date Windows 3.0, 1990

Platforms IBM PCs and clones

Supporting applications Visual Basic, Word for Windows, PageMaker, Ventura for Windows, Corel Draw, Drafix Windows CAD, and others, Most applications other than Visual Basic require the Placeable metafile header

Similar to CGM, somewhat

Overall assessment Best device-independent format under windows

Advantages and disadvantages

Advantages Device independent, files can be well-structured, files can be much smaller than corresponding bitmaps due to higher-level feature descriptions

Disadvantages Semantics closely tied to Windows imaging model, files relatively complex

Variants

Windows 3.x adds new record types to the Windows 2.x metafile format, but metafiles that avoid the new types should be backward compatible. Placeable metafiles have a prefix header containing size and scaling information.

Overview of file structure

A windows Metafile stores a list of Microsoft Windows graphical function calls. Only a subset of the Windows functions are allowed in metafiles, but it is a large subset containing most of the drawing calls. Although the metafile was originally intended as a sort of a graphical macroinstruction, it turns out to be more generally useful and is nower of scan as a general image interchange format among Windows applications.

A metafile consists of a short header followed by some number of records. An optional "placement" header can precede the file header. Each record corresponds to a Windows graphics device interface (GDI) call, and contains a size, a function number, and some argument data. In most cases, the argument data words are just the values that would be passed to the corresponding Windows GDI routine, although for some of the more complicated routines there are somewhat more complex data encodings.

Windows Imaging Model

Here is a brief overview of Windows graphics programming. For more details, consult the *Microsoft Windows Programmer's Reference.*

Geometry

All graphics in Windows are drawn in a *display context,* which contains a set of parameters, such as colors, scaling, defined drewing objects, etc. Drawing coordinates are relative to a logical window, which is mapped into a physical viewport on a screen or printer. Normally coordinates in the window correspond to the viewport one-to-one, with one unit being one pixel, although it is possible to scale and move both the window and the viewport. All coordinates are expressed as 16 bit signed integers.

Drawing is done with "objects," which are pens, brushes, bitmaps, fonts, and regions. A pen draws a line of some width that might be solid, dashed, or dotted. A brush lays down repeated copies of an array of pixels. A brush can use a built-in solid color or crosshatch brush pattern, or an arbitrary bitmap. One might outline a rectangle or circle with a pen, then use a brush to fill it in with a desired color or pattern. A region defines a clipping area-when a region is in effect, all graphics are clipped so that only material within the region is actually drawn.

All of the defined objects are assigned object numbers starting at zero. When a new object is defined, it is assigned the lowest unused number. After an object is deleted, its number is reused. Only one object of a particular kind is active at a time. The file header contains the maximum count of simultaneously defined objects, which is one more than the highest object number used.

To draw text, one first defines a font in terms of size, character style, color, and other attributes. Then a separate call draws the text, giving the position, the string, and some formatting options.

The color model uses a standard color map that is shared among all windows on the screen. Twenty entries are preallocated for standard colors. Windows compete for the rest of the entries. A set of color entries is known as a logical palette, which is realized when a window becomes visible by assigning the palette entries to entries in the physical color map. Within a palette entry, colors are specified as three values for red, green, and blue, with each value in the range 0 to 255 with 255 being the brightest. Colors in GDI calls are generally given as entry numbers in the active logical palette.

Color references

Many calls take as arguments a color reference, a 32-bit number in one of several forms. If the high byte is zero, it is an absolute color reference and the form is hex 00RRGGBB with RR, GG, and BB being red, green, and blue direct color values. If the high byte contains 1, the form is 0100NNNN where NNNN is a color index in the current logical palette. If the high byte contains 2, the form is 02RRGGBB where RRGGBB are again color values, but the system is to choose the closest existing color in the current logical palette.

Format details

Intel little-endian format is used for all data. Most values are 16-bit integers. All sizes are given in terms of 16 bit words, not bytes.

Header format

Every metafile starts with a file header, Table 23-1.

Table 23-1 WMF file header

Offset	Size	Description
0	2	Metafile type, always 1 for a disk file
2	2	Header size in words, generally 9.
4	2	Version number, hex 0300 for Windows 3
6	4	File size in words

10	2	Number of object handles used
12	4	Size in words of largest record
16	2	unused

The file header gives the maximum record size and number of simultaneously declared objects so that a program that reads the file can preallocate space.

Record format

Each record, Table 23-2, starts with a header containing a size word followed by the code that defines the function, and as many parameters as the function requires. The shortest possible record, with no parameters, is three words long.

Table 23-2 WMF record format

Offset	**Size**	**Description**
0	4	Size in words of the record
4	2	Function code
6	2*N	Function parameters

Metafile record types

Each record starts with the record size and function code as above. Function codes are all specified in hexadecimal. The parameters are usually the arguments to the corresponding Windows GDI function, in reverse order for some reason. For some of the functions, the parameters are more complicated, usually because the Windows function takes as a parameter a data structure more complex than a single value. In the descriptions below, if the parameters are all single words, they are merely listed in the order they appear in the record. If they are more complex, a structure description appears.

AnimatePalette

Code: 436 (new in Windows 3.x)

Parameters: See Table 23-3.

Table 23-3 AnimatePalette parameters

Offset	Size	Description
6	2	Starting palette entry number
8	2	Number of entries to animate
10	4*N	New patette entries

The new palette entries in the parameters immediately replace the given entries in the active logical palette.

Each entry is an absolute color reference.

Arc

Code: 817

Parameters: Y4, X4, Y3, X3, Y2, X2, Y1, X1

Draw an elliptical arc using the currently selected pen. The center of the ellipse is the center of the bounding rectangle with corners at (X1, Y1) and (X2, y2). The arc is drawn counterclockwise from (X3, Y3) to (X4, Y4).

BitBlt

Code: 922 (obsolete, Windows 2.x only)

Parameters: See Table 23-4.

Table 23-4 Old BitBlt parameters

Offset	Size	Description

6	2	Type of operation
8	2	Bitmap Y origin
10	2	Bitmap X origin
12	2	Destination area height
14	2	Destination area width
16	2	Destination Y origin
18	2	Destination X origin
20	2	Bitmap width in pixels
22	2	Bitmap height in pixels
24	2	Bitmap width in bytes
26	2	Bitmap number of planes
28	2	Pixel bits per plane, always 1
30	2	Bitmap data

Transfer a rectangle of bits into the image. The source and destination rectangles need not be the same size; if they are not the mode from the most recent SetStretchBltMode controls how the source image is resized. The source X and Y origins are disregarded and are usually zero.

The source bitmap is stored by plane by scan line, with the most significant plane first for each pixel row. Each scan line is padded out to a multiple of 16 bits, and the width in bytes at offset 24 reflects this padding. Color bitmaps depend on the target device, but usually store triples of red, green and blue planes in that order. (The new BitBlt3 operator below handles color images in a device independent way.)

The type of operation controls how the source bitmap is merged with the existing contents of the destination rectangle. Type 0020 means source overwrites destination. Type 0086 logically ORs the source with the destination. Type 00C6 logically ANDs the source with the destination. Type 0046 logically XORs the source with the destination. Other types exist but are less often used.

BitBlt3

Code: 940 (new in Windows 3.x)

Parameters: See Table 23-5.

Table 23-5 New BitBlt parameters

Offset	Size	Description
6	2	Type of operation
8	2	Bitmap Y origin
10	2	Bitmap X origin
12	2	Destination area height
14	2	Destination area width
16	2	Destination Y origin
18	2	Destination X origin
20	40	BITMAPINFOHEADER
60	4*N	color map
60+4N	S	bits

The Windows 3 BitBlt operation is the same as the Windows 2 version, except that the bitmap is specified in a device independent format. The BITMAPINFOHEADER, color map, and image bits are identical to those in the Windows Bitmap file described in chapter 10.

Chord

Code: 830

Parameters: Y4, X4, Y3, X3, Y2, X2, Y1, X1

Draw a chord, the intersection of an ellipse and a line. An elliptical arc is drawn the same way that arc does, then the endpoints are connected using the selected pen and the chord filled using the selected brush.

CreateBrushIndirect

Code: 2fc

Parameters: See Table 23-6.

Table 23-6

CreateBrushIndirect parameters

Offset	Size	Description
6	2	Brush style
8	4	Brush color
12	2	Cross-hatch line type

Create a new brush type and add it to the object table. The brush style is 0 for solid color, 1 for "hollow" (background color shows through), or 2 for crosshatched. It the brush is crosshatched the line type are 0 for horizontal, 1 for vertical, 2 for diagonal rising to the right, 3 for diagonal rising to the left, 4 for horizontal and vertical, and 5 for both diagonals.

The color of a solid or crosshatched brush is a 32 bit color reference.

CreateFontIndirect

Code: 2fb

Parameters: See Table 23-7.

Table 23-7 CreateFontIndirect parameters

Offset	Size	Description
6	2	Height
8	2	Width
12	2	Escapement
14	2	Orientation
16	2	Weight
18	1	Italic
19	1	Underline

20	1	Strikeout
21	1	Character set
22	1	Output precision
23	1	Clipping precision
24	1	Output quality
25	1	Two low bits: pitch, four high bits: font type
26	32	Typeface name

Select a font that best matches the attributes given and add it to the object table. Height is the character cell height if positive, average character height if negative, or selects a default size if zero. Width is the average character width, or a default if zero. Slant specifies in tenths of degrees the angle of slanted (pseudo-italic) characters, with positive being clockwise from horizontal. Orientation specifies in tenths of degrees the angle at which the font is drawn, with positive being clockwise from horizontal. Weight is in the range of 0 to 1000, with typical values being 400 for normal and 700 for bold characters. Italic, Underline, and Strikeout if nonzero specify the corresponding font attributes. Character set contains a character set code, typically 0 for ANSI. Output precision says how closely the character sizes and angles must match the ones specified, 0 for default, 1 for matching at the string level,and 2 for matching at the character level. Clip precision says how precisely to clip characters that are partially contained in the clipping region. Both are typically zero to use a default precision. Output quality is 0 for default pitch, 1 for draft, and 2 for proof quality. The two low-order bits of the pitch and family byte are 0 for a default 1 for fixed, and 2 for variable. The high four bits set the font family, 0 for a default or 10 H for Roman, 20 for Sans-Serif, 30 for Typewriter, 40 for Script, and 50 for Decorative font. Typeface name is the nullterminated ASCII name of the desired typeface.

CreatePalette

Code: 0f7 (new in Windows 3.x)

Parameters: See Table 23-8.

Table 23-8

CreatePalette parameters

Offset	Size	Description
6	2	Version
8	2	Number of entries
10	4*N	Palette entries

Create a color palette with the desired entries and add it to the object table. The palette has one or more entries, each specified by a 32-bit color reference. Version is 300 H for Windows 3.x.

CreatePatternBrush

Code: 1f9 (obsolete, Windows 2.x only)

Parameters: See Table 23-9.

Table 23-9

CreatePatternBrush parameters

Offset	Size	Description
6	2	Width in bits
8	2	Height in bits
10	2	Width in bytes
12	2	Number of planes
14	2	Pixel bits per plane, always 1
16	2	Pointer to bits, unused
18	N	Bitmap bits

Create a brush using the specified bitmap and enter it in the handle table. Each bit row is stored as an integral number of words, so the width in bytes must be a multiple of two, with part of the last word in each row possible unused. A monochrome bitmap is stored as a series of rows, from top to bottom, with the bits in each row stored from left to tight. A color bitmap is stored by row, so for a 9-bit deep bitmap, the first row of each plane is stored in the order red0, red1, red2, green0, green1, green2, blue0, blue1, blue2. Then the second row of each plane is stored, in the same order, and so forth.

CreatePatternBrush3

Code: 142 (new in Windows 3.x)

Parameters: See Table 23-10

Table 23-10 CreatePatternBrush3 parameters

Offset	Size	Description
6	2	Type
8	2	Usage
10	N	BITMAPINFO describing the bitmap
10+N	M	bits

Create a brush using the given bitmap and enter it in the handle table. Type field should be five to indicate that a device-independent bitmap folllows, and Usage zero to indicate that its palette contains explicit RGB values.

The BITMAPINFO and bits are identical to those in a Windows bitmap file, as described in chapter 10.

CreatePenIndirect

Code: 2fa

Parameters: Style, Width, unused, Color

Create a logical pen and enter it in the handle table. The style can be 0 solid, 1 dashed, 2 dotted, 3 alternating dot and dash, 4 alternating dash and two dots, 5 none, and 6 inside frame. The "inside frame" style is a variety of solid that draws the interior but not the boundary of primitive objects. Width is the stroke width in logical units. Color is a 32-bit color reference.

CreateRegion

Code: 6ff

Parameters: Unspecified

Create a region and enter it in the handle table. This function is not often used and its arguments are not documented.

DeleteObject

Code: 1f0

Parameters: Index

Delete the object identified by the handle table index, and free the entry in the handle table.

DrawText

Code: 62f

Parameters: See Table 23-11

Table 23-11 DrawText parameters

Offset	Size	Description
6	2	Format method
8	2	Character count, or zero for null-terminated string
10	2	X1
12	2	Y1
14	2	X2
16	2	Y2
18	N	character string

Draw the given text within the rectangle defined by (X1,Y1) and (X2,Y2). The format method is the logical OR of several sets formatting option bits.

Horizontal alignment 2 H for right aligned, 1 for centered, 0 for left aligned.

Vertical alignment 8 H for bottom of the rectangle, 4 for centered, 0 for the top.

Single line 20 H says to treat the text as a single line even if it contains carriage returns or line feeds.

Word Break 10 H says to break into multiple lines at spaces rather than always at the right boundary. Carriage returns and line feeds also force a new line unless the single line bit is set.

Tabs 40 H says to expand tab characters, and 80 H says to use the high byte as the format word as the number of spaces per tab stop (default 8).

Clipping 400 H says to extend the rectangle at the right for single lines or the bottom for multiple lines to make the text fit. 100 H says to draw the text without clipping.

Miscellaneous 200 H says to include the "external leading" of a font in size calculations, and 800 H suppresses the special treatment of the & character, which normally means to underscore the following character.

Ellipse

Code: 418

Parameters: Y2, X2, Y1, X1

Draw an ellipse bounded by the rectangle defined by (X1, Y1) and (X2, Y2).

Escape

Code:626

Parameters: See Table 23-12.

Table 23-12 Escape parameters

Offset	Size	Description

6	2	Escape number
8	2	Number of bytes of escape data
10	N	Escape data

Call an escape function defined by the output device driver.

ExcludeClipRect

Code: 415

Parameters: Y2, X2, Y1, X1

Create a new clipping region consisting of the current region minus the rectangle defined by (X1, Y1) and (X2, Y2).

ExtTextOut

Code: a32

Parameters: See Table 23-13.

Table 23-13 ExtTextOut parameters

Offset	Size	Description
6	2	Y character origin
8	2	X character origin
10	2	Option bits
12	2	X1
14	2	Y1
16	2	X2
18	2	Y2
20	N	text string
20+N	M	optional character spacings

Draw the text string in the rectangle defined by (X1, Y1) and (X2, Y2). Option bit 4 H clips the text to fit in the rectangle, and 2 fills the rectangle with the current background color. If neither bit is set, the rectangle data are ignored. The text string must be padded to an even boundary. If the character spacings are present, they are an array of words giving the width of each character cell.

FloodFill

Code: 419

Parameters: Color, Y, X

Fill an area using the current brush, starting at (X, Y) and extending out to a boundary whose color is given by the 32-bit color reference Color.

IntersectClipRect

Code: 416

Parameters: Y2, X2, Y1, X1

Create a new clipping region consisting of the intersection of the current region and the rectangle defined by (X1, Y1) and (X2, Y2).

LineTo

Code:213

Parameters: Y, X

Draw a line using the current pen from the current position to (X, Y).

MoveTo

Code: 214

Parameters: Y, X

Set the current position to (X, Y).

OffsetClipRgn

Code: 220

Parameters: Y, X

Move the current clipping region by the distance (X, Y).

OffsetViewportOrg

Code: 211

Parameters: Y, X

Move the viewport origin by adding (X, Y) to the current origin. X and Y are in device coordinates.

OffsetWindowOrg

Code: 20f

Parameters: Y, X

Move the window origin by adding (X, Y) to the current origin. X and Y are in logical coordinates.

PatBlt

Code: 61d

Parameters: Op, Height, Width, Y, X

Create a pattern in the rectangle with origin (X, Y) and the given Height and Width by combining the current brush with the existing contents in a manner defined by the 32-bit Op parameter. Values of Op are 00F00021 H to replace the contents with the rectangle with the brush pattern, 005A0049 H to XOR the brush pattern with the existing contents, 00550009 H to logically invert the existing contents disregarding the brush, 00000042 H to make the rectangle entirely black, and 00FF0062 H to make the rectangle entirely white.

Pie

Code: 81a

Parameters: Y4, X4, Y3, X3, Y2, X2, Y1, X1

Draw a pie shaped area bounded by an elliptical arc. The center of the ellipse is the center of the bounding rectangle with corners at (X1, Y1) and (X2, Y2). The arc is drawn counterclockwise from (X3, Y3) to (X4, Y4), then lines are drawn from the endpoints to the center and the area filled with the current brush.

Polygon

Code: 324

Parameters: See Table 23-14.

Table 23-14 Polygon

and Polyline parameters

Offset	Size	Description
6	2	Count
8	2	X1
10	2	Y1
...		

Draw a polygon. Count is the number of (X, Y) pairs that follow, each defining a vertex of the polygon. The polygon is filled with the current brush using the mode set by SetPolyFillMode.

Polyline

Code: 325

Parameters: See Table 23-14.

Draw a polyline. Count is the number of (X, Y) pairs that follow, each defining the end of a line segment. Lines are drawn from (X1, Y1) to (X2, Y2), then (X2, Y2) to (X3, Y3), etc.

PolyPolygon

Code: 538 (new in Windows 3.x)

Parameters: See Table 23-15.

Table 23-15

PolyPolygon parameters

Offset	Size	Description

6	2	Point count
8	N	Polygon counts
8+N	2	X1
10+N	2	Y1

Draw a set of polygons. Point count is the total number of (X, Y) pairs provided. Polygon counts is list of words each specifying the number of points that define a single polygon. The sum of the polygon counts must equal the point count.

RealizePalette

Code: 035 (new in Windows 3.x)

Parameters: none

Realize the current logical palette, i.e., select colors in the currnet display that correspond to those in the logical palette.

Rectangle

Code: 41b

Parameters: Y2, X2, Y1, X1

Draw the rectangle defined by (X1, Y1) and (X2, Y2) using the current pen.

ResizePalette

Code: 139 (new in Windows 3.x)

Parameters: Number of entries

Change the current logical palette so it contains the specified number of entries. Entries are added or deleted at the end, and new entries are all black.

RestoreDC

Code: 127

Parameters: Context, Handle

Restore a context saved by SaveDC from the context stack. Handle identifies the object to restore. Context should be - 1 to specify the context at the top of the stack.

RoundRect

Code: 61c

Parameters: Y3, X3, Y2, X2, Y1, X1

Draw the rectangle defined by (X1, Y1) and (X2, Y2) with rounded corners using the current pen and fill it with the current brush. X3 and Y3 define the width and height of the ellipse to use for the corners.

SaveDC

Code: 01e

Parameters: Handle

Save a copy of the object identified by Handle on the context stack

ScaleViewportExt

Code: 412

Parameters: Y1, X1, Y0, X0

Change the size of the viewport by multiplying its size by (X0/X1, Y0/Y1).

ScaleWindowExt

Code: 400

Parameters: Y1, X1, Y0, X0

Change the size of the window by multiplying its size by (X0/X1, Y0/Y1).

SelectClipRgn

Code: 12c

Parameters: Handle

Select the clipping region identified by Handle.

SelectObject

Code: 12d

Parameters: Handle

Select the bitmap, pen, brush, font, or region identified by Handle.

SelectPalette

Code: 234

Parameters: Handle

Select the palette identified by Handle.

SetBkColor

Code: 201

Parameters: Color

Set the background color identified by 32-bit color reference color.

SetBkMode

Code: 102

Parameters: Mode

If Mode is 2, fill the background with the current color before a pen, brush, or text is drawn. If Mode is 1, leave the background alone.

SetDIBitsToDevice

Code: d33 (new in Windows 3.x)

Parameters: See Table 23-16.

Table 23-16

SetDIBitsToDevice Parameters

Offset	Size	Description
6	2	Usage
8	2	Number of scan lines
10	2	First scan line

12	2	Bitmap Y origin
14	2	Bitmap X origin
16	2	Bitmap height
18	2	Bitmap width
20	2	Destination Y origin
22	2	Destination X origin
24	N	BITMAPINFO structure
24+N	M	bitmap data

Transfer the given bitmap directly to the output device. The source bitmap is the rectangle in the given bitmap defined by the bitmap X and Y origins, width, and height. Data can be transferred a band at a time using the First scan line and Number of scan lines to limit the amount of data transferred. Usage is 1 to say that the palette in the BITMAPINFO contains 16-bit indices into the current logical palette or zero to say that the palette contains explicit RGB values.

The BITMAPINFO and bits are identical to those in a Windows bitmap file, as described in chapter 10.

SetMapMode

Code: 103

Parameters: Mode

Set the units used to transform logical to physical device coordinates. Mode 1 maps units directly to pixels, 2 makes a logical unit 0.1 mm, 3 makes it 0.01 mm. 4 makes it 0.01 inch, 5 makes it 0.001 inch, 6 makes it 1/1440 inch, 7 and 8 make it arbitrary units defined by SetWindowExt and SetViewpotExt, with mode 7 forcing equal scaling on the two axes. In modes 1 the Y origin is at the bottom of the screen with increasing values toward the top, in modes 2 through 6 the Y origin is at the top of the screen with increasing values toward the bottom, and in modes 7 and 8 the origin and orientation depend on the scaling set.

SetMapperFlags

Code: 231

Parameters: Flag

Flag is a 32-bit values. If its high bit is 1, fonts will only be mapped if the X and Y aspect ratio of a logical font exactly matches the physical font.

SetPaletteEntries

Code: 037

Parameters: See Table 23-17.

Table 23-17

SetPaletteEntries Parameters

Offset	size	Description
6	2	First entry
8	2	Number of entries
10	4*N	Entries

Set the given entries in the current logical palette. Each entry is a 32-bit color reference. Changes are not visible until the next RealizePalette.

SetPixel

Code: 41f

Parameters: Color, X, Y

Set the pixel a (X, Y) to Color, a 32-bit color reference.

SetPolyFillMode

Code: 106

Parameters: Mode

If Mode is 1, nonsimple polygons are only filled in simply enclosed areas, if 2 the entire polygon is filled. For example, in a five-pointed star mode 1 only fills the points while mode 2 fills the entire star.

SetROP2

Code: 104

Parameters: Mode

Set drawing mode, the way that pen and display data are combined during drawing. The mode is computed as follows: Start with a mode value of 1. Add 1 to the mode if bits where the pen and display were both 0 should be 1. Add 2 to the mode if bits where the pen was 0 and the display was 1 should be 1. Add 4 to the mode if bits where the pen was 1 and the display was 0 should be 1. Add 8 to the mode if bits where the pen and display were both 1 should be 1. Common values are 13 to have the pen overlay the existing contents and 7 for XOR drawing.

SetStretchBltMode

Code: 107

Parameters: Mode

Set the disposition of deleted scan lines when a bitmap is compressed in a StretchBlt function. Mode 1 ANDs the eliminated lines with adjacent lines. Mode 2 ORs the eliminated lines with adjacent lines. Mode 3 simply discards the eliminated lines. Modes 1 and 2 can be useful to preserve significant foreground or background pixels in monochrome bitmaps.

SetTextAlign

Code: 12e

Parameters: Flags

Set text alignment options, based on bits in Flags. 01 H says to update the current graphics position after each TextOut or ExtTextOut call. The default is not to update.

By default, horizontal text alignment is based on the left side of the character bounding rectangle. 02 H says to align on the reight side and 06 H says to align on the center of the rectangle.

By defaul, vertical text alignment is based on the top side of the character bounding rectangle. 08 H says to align on the bottom, and hex 18 says to align on the text baseline.

SetTextCharExtra

Code: 108

Parameters: Width

Set the amount of extra space between characters to Width logical units.

SetTextColor

Code: 209

Parameters: Color

Set the foreground color of text to Color, a 32-bit color reference.

SetTextJustification

Code: 20a

Parameters: BreakCount, BreakExtra

Horizontally justify subsequent output text. BreakExtra is the distance in logical units to be distributed among the break characters in the line. BreakCount is the number of break characters (typically spaces) in the line. A BreakExtra of zero turns off justification.

SetViewportExt

Code: 20e

Parameters: Y, X

Make the viewport extent (X, Y), expressed in device units. Ignored if the mode set by SetMapMode is not 7 or 8.

SetViewportOrg

Code: 20d

Parameters: Y, X

Make the viewport origin (X, Y), expressed in device units.

SetWindowExt

Code: 20c

Parameters: Y, X

Make the window extent (X, Y), expressed in logical units. Ignored if the mode set by SetMapMode is not 7 or 8.

SetWindowOrg

Code: 20b

Parameters: Y, X

Make the window origin (X, Y), expressed in logical units.

StretchBlt

Code: b23 (obsolete, Windows 2.x only)

Parameters: See Table 23-18.

Table 23-18 StretchBlt parameters

Offset	Size	Description
6	4	Operation
10	2	Source height
12	2	Source width
14	2	Source Y origin
16	2	Source X origin
18	2	Destination height
20	2	Destination width
22	2	Destination Y origin
24	2	Destination X origin
26	2	Bitmap width in pixels
28	2	Bitmap height
30	2	Bitmap width in bytes

32	2	Bits per pixel per plane, always 1
34	N	Bitmap

Move a source rectangle from the bitmap to a destination rectangle, stretching or shrinking it to fit. The height and width values might be of opposite signs to produce mirror image transfers. The bitmap is stored the same way as for the BitBlt operator, above. The Operation type determines how the source and destination are combined. Type 00CC0020 H means source overwrites destination. Type 00EE0086 H logically ORs the source with the destination. Type 008800C6 H logically ANDs the source with the destination. Type 00660046 H logically XORs the source with the destination. Other types are possible but less common.

The source and destination bitmaps need not have the same depth. Black and white pixels in monochrome bitmaps are converted to the foreground and background colors in color bitmaps and vice-versa.

StretchBlt3

Code: b41 (new in Windows 3.x)

Parameters: See Tbalbe 23-19.

Table 23-19

StretchBlt3 parameters

Offset	Size	Description
6	4	Operation
10	2	Source height
12	2	Source width
14	2	Source Y origin
16	2	Source X origin
18	2	Destination height
20	2	Destination width

22	2	Destination Y origin
24	2	Destination X origin
26	2	BITMAPINFO structure
26+N	N	Bitmap data

Move a source rectangle from the bitmap to a destination rectangle, stretching or shrinking it to fit. The height and width values might be for opposite signs to produce mirror image transfers. The Operation values are the same as for StretchBlt, above.

The BITMAPINFO and bits are identical to those in a Windows bitmap file, as described in chapter 10.

StretchDIBits

Code: f43 (new in Windows 3.x)

Parameters: See Table 23-20.

Table 23-20

StretchBlt3 parameters

Offset	Size	Description
6	4	Operation
10	2	Usage
12	2	Source height
14	2	Source width
16	2	Source Y origin
18	2	Source X origin
20	2	Destination height
22	2	Destination width
24	2	Destination Y origin
26	2	Destination X origin
28	2	BITMAPINFO structure

28+N	N	Bitmap data

Move a source rectangle from the bitmap to a destination rectangle, stretching or shrinking it to fit. The height and width values may be of opposite signs to produce mirror image transfers. The Operation values are the same as for StretchBlt, above. Usage is 1 to say that the palette in the BITMAPINFO contains 16-bit indices into the current logical palette or zero to say that the palette contains explicit RGB values.

The BITMAPINFO and bits are identical to those in a Windows bitmap file, as described in chapter 10.

TextOut

Code: 521

Parameters: See Table 23-21.

Table 23-21

TextOut parameters

Offset	Size	Description
6	2	String length
8	N	String
8+N	2	Y origin
10+N	2	X origin

Draw the string starting at (X, Y). The text string must be padded to an even boundary.

Placeable metafiles

Placeable metafiles contain a header preceding the file header, Table 23-22.

Table 23-22

Placeable Metafile header

Offset	Size	Description
0	4	ID word, 9AC6CDD7
4	2	Unused, must be zero
6	2	Top left X
8	2	Top left Y
10	2	Bottom right X
12	2	Bottom right X
14	2	Scale, units per inch
16	4	Unused, must be zero
20	2	Checksum

The header contains the (X, Y) coordinates of the top left and bottom right corners of a bounding rectangle for the image drawn by the metafile. Scale is the number of units per inch at which the image should be drawn. The checksum is the XOR of the rest of the header, taken as ten 16-bit words.

Placeable metafile restrictions

A placeable metafile may not contain any of the following calls: BitBlt, Escape, OffsetClipRgn, SelectClipRgn, SetMapMode, SetViewportExt, SetViewportOrg, SetWindExt, SetWindOrg.

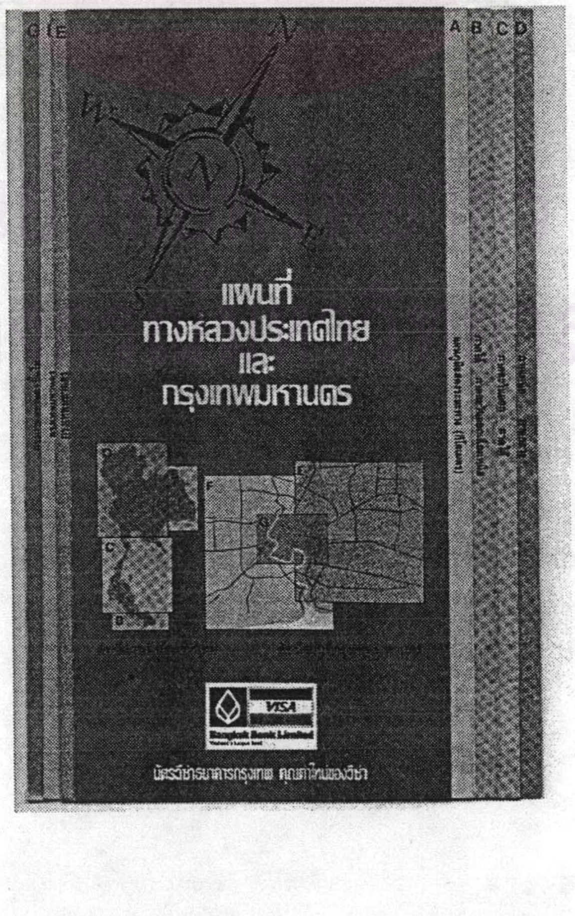
SaveDC and RestoreDC must be properly matched. Avoid pattern brushes because their appearance varies from one device to another.

ภาคผนวก ค

ตัวอย่างแผนที่เส้นทางเดินรถยนต์

ตัวอย่างแผนที่เส้นทางเดินรถยนต์ที่ยกมานี้เป็น แผนที่ที่ธนาคารกรุงเทพแจกให้กับลูกค้าบัตรเครดิตวีซ่าของธนาคาร ที่ออกแบบและจัดทำโดยห้างหุ้นส่วนจำกัดคาร์โตเมติก มีลักษณะเป็นแผ่นพับขนาดความสูงประมาณ 21 เซนติเมตร(ความกว้างของกระดาษมาตราฐาน A4) และกว้าง 13.5 เซนติเมตร มีทั้งหมด 7 หน้าคือหน้า A-G และมีมาตราส่วนโดยประมาณ(เนื่องจากการมีการแสดงมาตราส่วนในแบบกราฟิก)ดังต่อไปนี้ (ตัวอย่างแผนที่ได้รับความเอื้อเฟื้อจากห้างหุ้นส่วนจำกัดคาร์โตเมติก)

- กรุงเทพฯชั้นใน มาตราส่วนประมาณ 1 : 86,000
- กรุงเทพฯ (นอกจากกรุงเทพฯชั้นใน) มาตราส่วนประมาณ 1 : 210,000
- เส้นทางเดินรถยนต์ระหว่างจังหวัดของภาคต่างๆ มาตราส่วนประมาณ 1 : 3,400,000



A

แผนภูมิแสดงระยะทาง (กิโลเมตร)

หมายเลขทางหลวงที่มุ่งไปหรืออยู่ใน

ภาคเหนือ	1 1x 1xx 1xxxx
ภาคตะวันออกเฉียงเหนือ	2 2x 2xx 2xxxx
ภาคกลาง ตะวันออกและตะวันตก	3 3x 3xx 3xxxx
ภาคใต้	4 4x 4xx 4xxxx

คำอธิบายสัญลักษณ์

ทางหลวงสายประธาน สายรอง	━━
ทางหลวงจังหวัด ทางอื่น ๆ	──
ทางรถไฟ	─┼─
อาณาเขตประเทศ	‐‐‐
แม่น้ำลำคลอง	〰
ทะเลสาบ หนองบึง	▨
ที่ตั้งจังหวัด	●
ที่ตั้งอำเภอ	•
สถานที่ท่องเที่ยว	⊕
สนามบิน	✈

แผนภูมิระยะทางระหว่างจังหวัด(กม.)

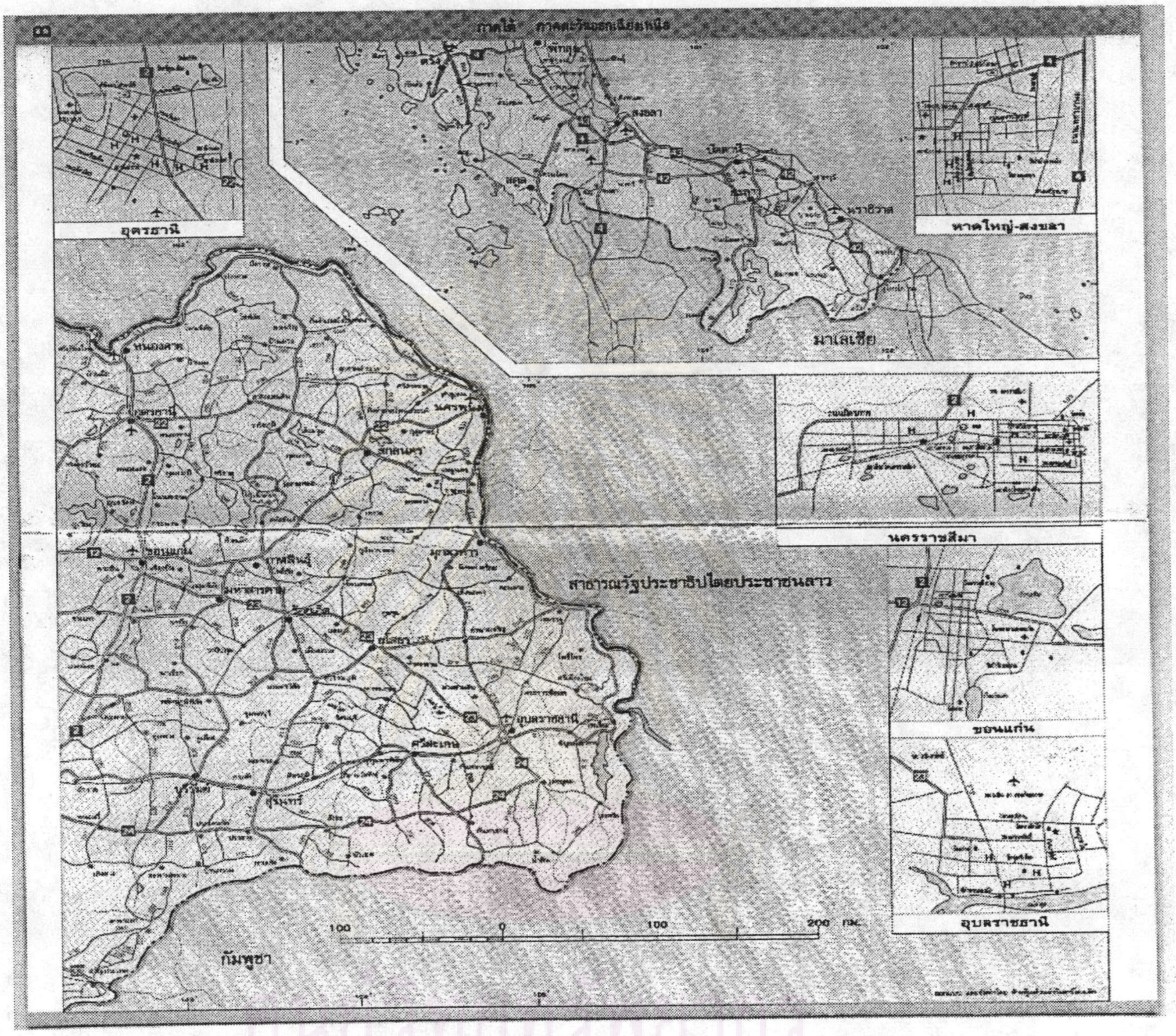
	กรุงเทพมหานคร	สิงห์บุรี	นครสวรรค์	ตาก	ลำปาง	เชียงใหม่	แม่ฮ่องสอน	เชียงราย	น่าน	พิษณุโลก	ลพบุรี	สระบุรี	เลย	นครราชสีมา	ขอนแก่น	อุดรธานี	หนองคาย	สกลนคร	นครพนม	มหาสารคาม	บุรีรัมย์	สุรินทร์	ยโสธร	อุบลราชธานี
สิงห์บุรี	145																							
นครสวรรค์	240	100																						
ตาก	425	285	185																					
ลำปาง	600	460	359	174																				
เชียงใหม่	685	550	450	265	92																			
แม่ฮ่องสอน	925	795	685	500	410	350																		
เชียงราย	785	645	540	400	225	180	635																	
น่าน	670	525	425	360	225	320	640	270																
พิษณุโลก	375	230	130	135	245	335	635	415	295															
ลพบุรี	155	33	130	315	490	580	815	670	555	260														
สระบุรี	105	79	175	360	535	625	860	720	600	305	46													
เลย	520	465	400	405	515	600	905	680	565	270	440	415												
นครราชสีมา	260	230	325	515	685	775	1010	870	750	455	200	150	345											
ขอนแก่น	450	420	495	455	565	850	955	730	815	320	390	340	205	190										
อุดรธานี	565	535	565	570	680	765	1070	845	730	435	505	455	150	305	115									
หนองคาย	615	585	615	875	730	820	1170	900	780	485	555	510	200	355	165	51								
สกลนคร	645	620	655	660	770	855	1160	935	820	525	585	540	310	390	205	160	210							
นครพนม	740	710	745	755	860	950	1255	1030	910	615	680	635	405	480	300	250	305	93						
มหาสารคาม	475	445	520	530	635	725	1030	805	685	390	415	370	280	215	73	190	240	170	265					
บุรีรัมย์	410	380	480	665	840	930	1165	1020	905	610	350	305	415	150	200	315	385	320	415	145				
สุรินทร์	455	430	525	710	885	975	1210	1070	950	655	395	350	460	200	255	375	420	310	350	175	110			
ยโสธร	530	505	600	640	745	840	1140	916	799	505	479	425	390	270	185	305	350	230	215	115	190	135		
อุบลราชธานี	830	600	700	740	845	935	1240	1014	897	600	570	520	490	370	283	400	450	285	270	215	290	225	98	
มุกดาหาร	840	615	710	700	805	900	1200	975	858	560	690	535	435	385	245	290	330	120	105	210	300	245	110	165

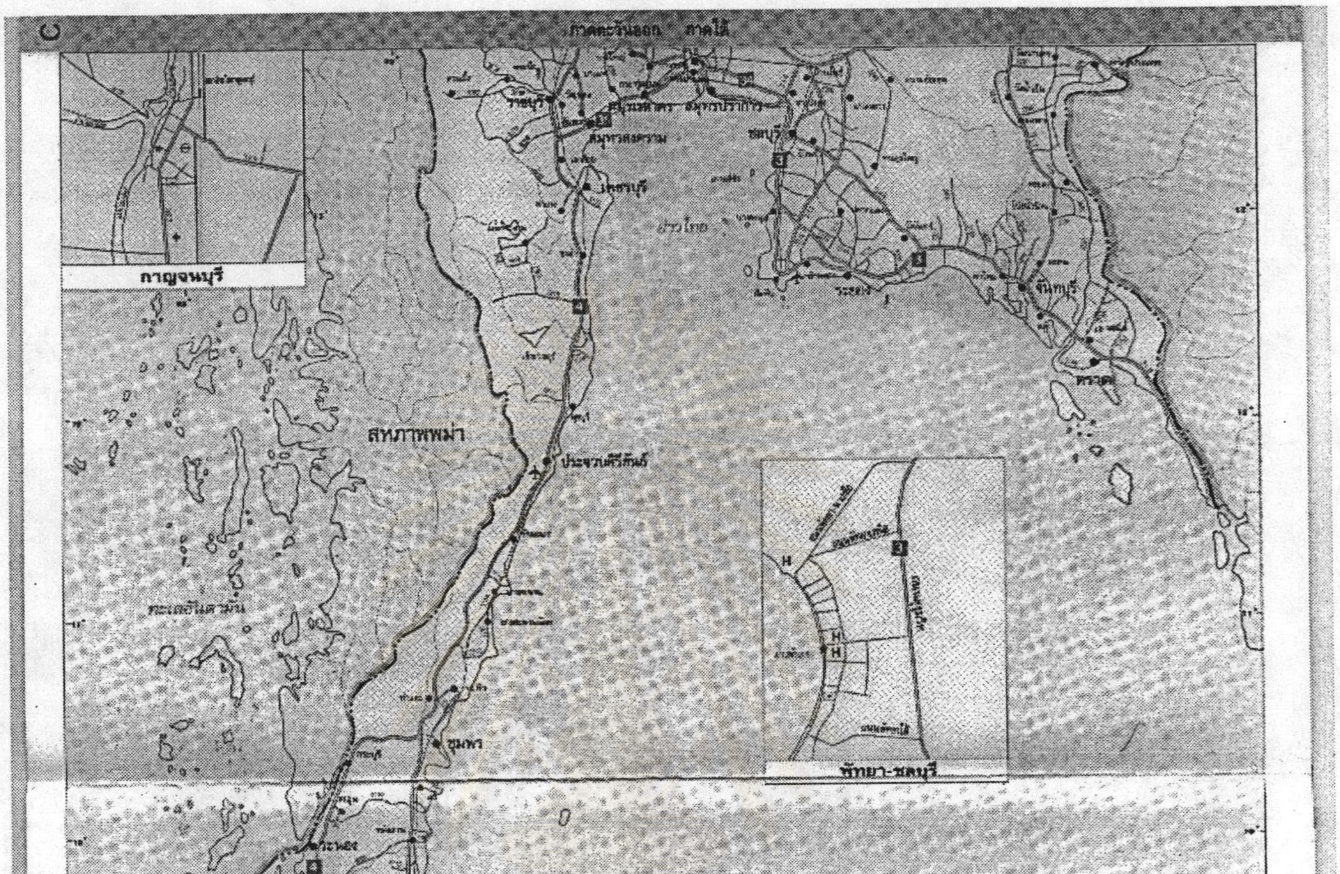
	นครนายก	ปราจีนบุรี	ตราด	จันทบุรี	ระยอง	ชลบุรี	ฉะเชิงเทรา	กรุงเทพมหานคร
นครราชสีมา	215	195	400	330	345	280	235	260
นครนายก		29	320	255	215	145	100	105
ปราจีนบุรี			310	245	155	120	76	135
ตราด				70	180	235	320	315
จันทบุรี					110	185	250	245
ระยอง						98	140	180
ชลบุรี							43	81
ฉะเชิงเทรา								82

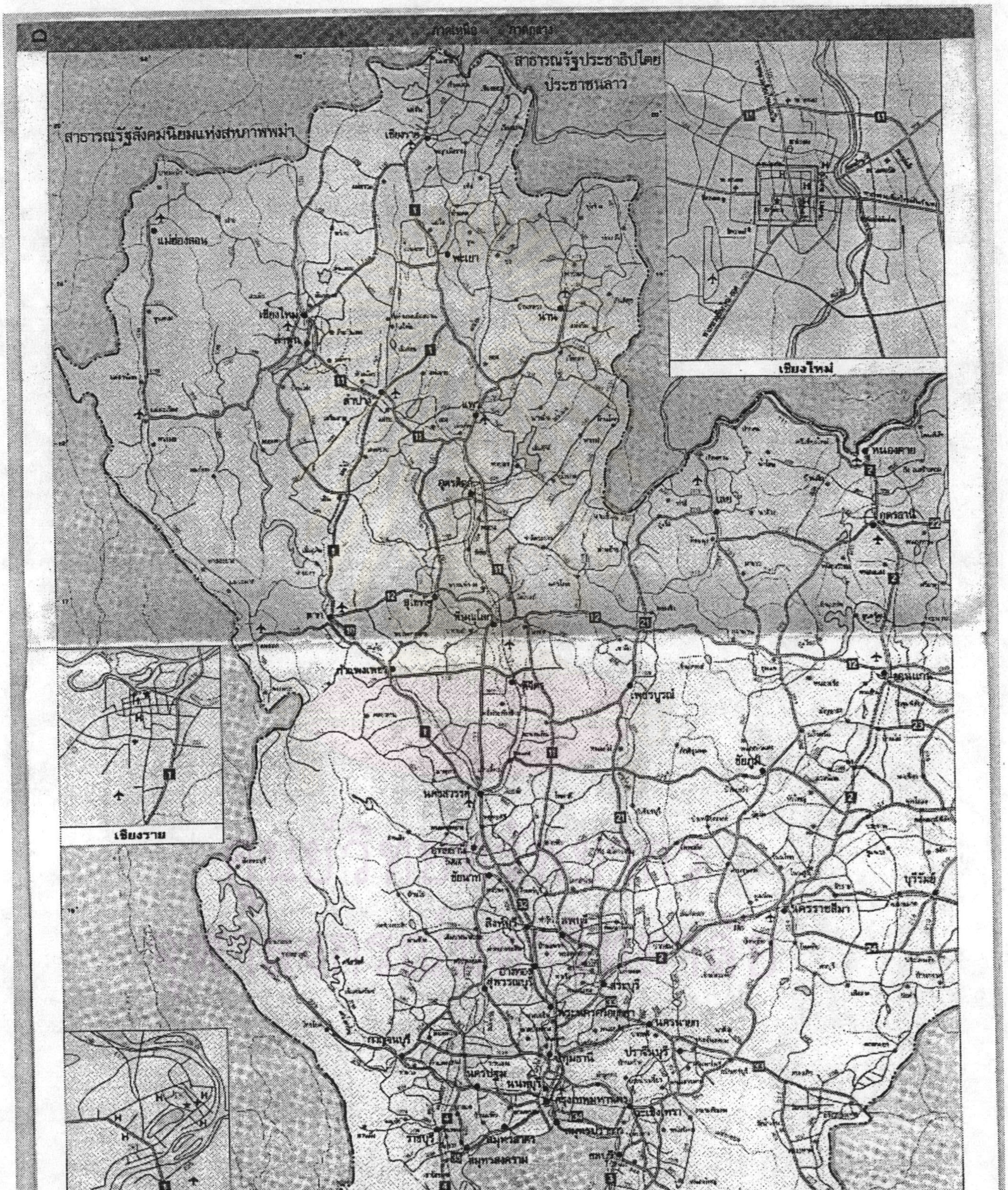
	ยะลา	ปัตตานี	สตูล	สงขลา	หาดใหญ่ - สงขลา	นครศรีธรรมราช	ภูเก็ต	สุราษฎร์ธานี	ระนอง	ชุมพร	ประจวบคีรีขันธ์	เพชรบุรี	ราชบุรี	กรุงเทพมหานคร
นราธิวาส	130	92	305	195	195	380	670	505	725	700	870	1025	1165	1150
ยะลา		35	235	130	130	295	605	435	660	635	805	960	1100	1085
ปัตตานี			210	99	105	265	575	410	630	605	775	930	1070	1055
สตูล				125	98	235	455	370	550	525	695	850	990	975
สงขลา					26	160	495	305	525	500	670	825	965	950
หาดใหญ่ - สงขลา						185	465	330	510	485	655	810	945	935
นครศรีธรรมราช							335	135	355	330	500	680	795	780
ภูเก็ต								265	300	410	580	740	875	860
สุราษฎร์ธานี									220	195	365	520	660	645
ระนอง										115	290	445	580	570
ชุมพร											185	340	480	465
ประจวบคีรีขันธ์												150	295	280
เพชรบุรี													145	125
ราชบุรี														130

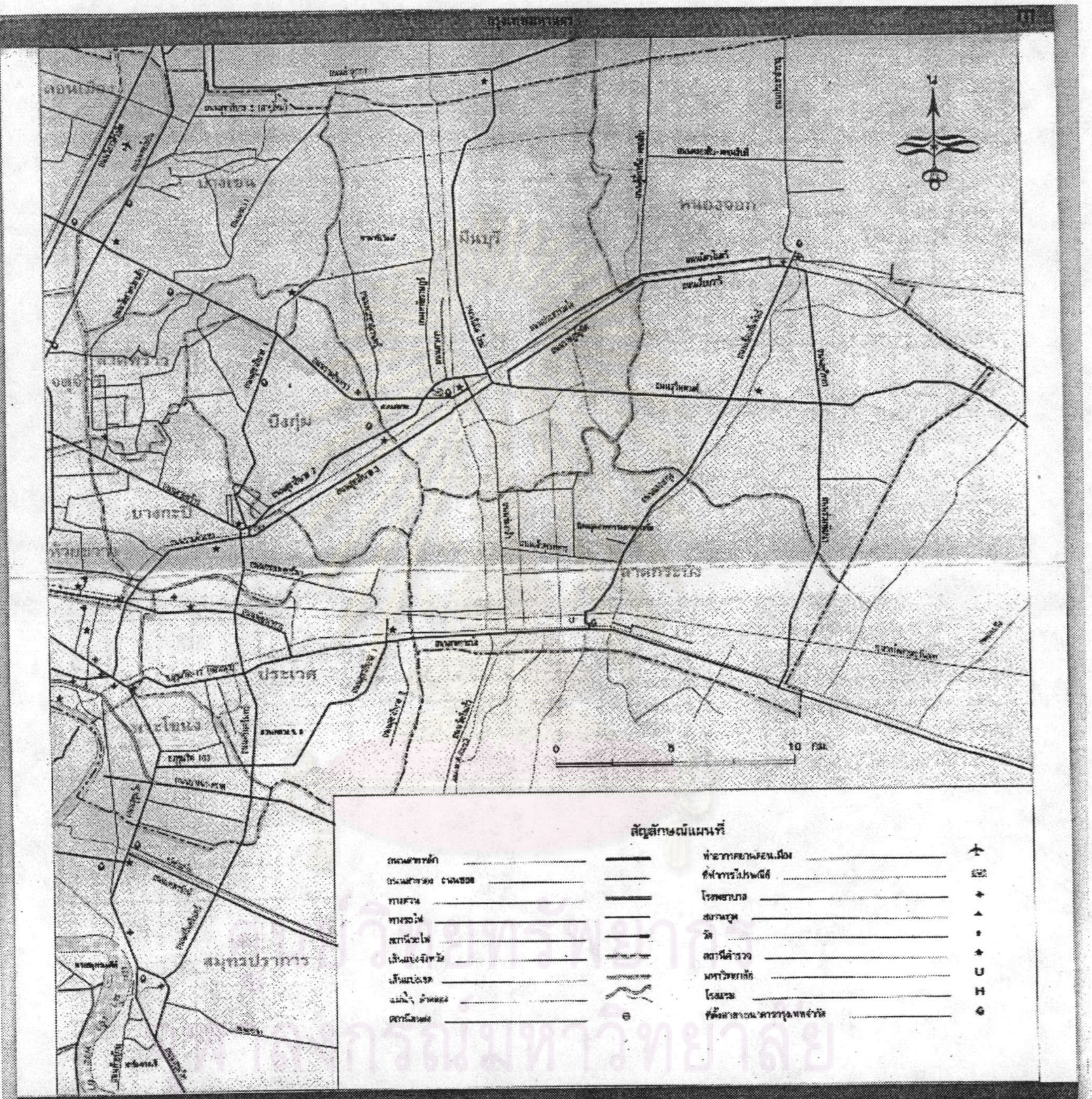
จุดให้บริการ เอ ที เอ็ม 24 ชั่วโมง

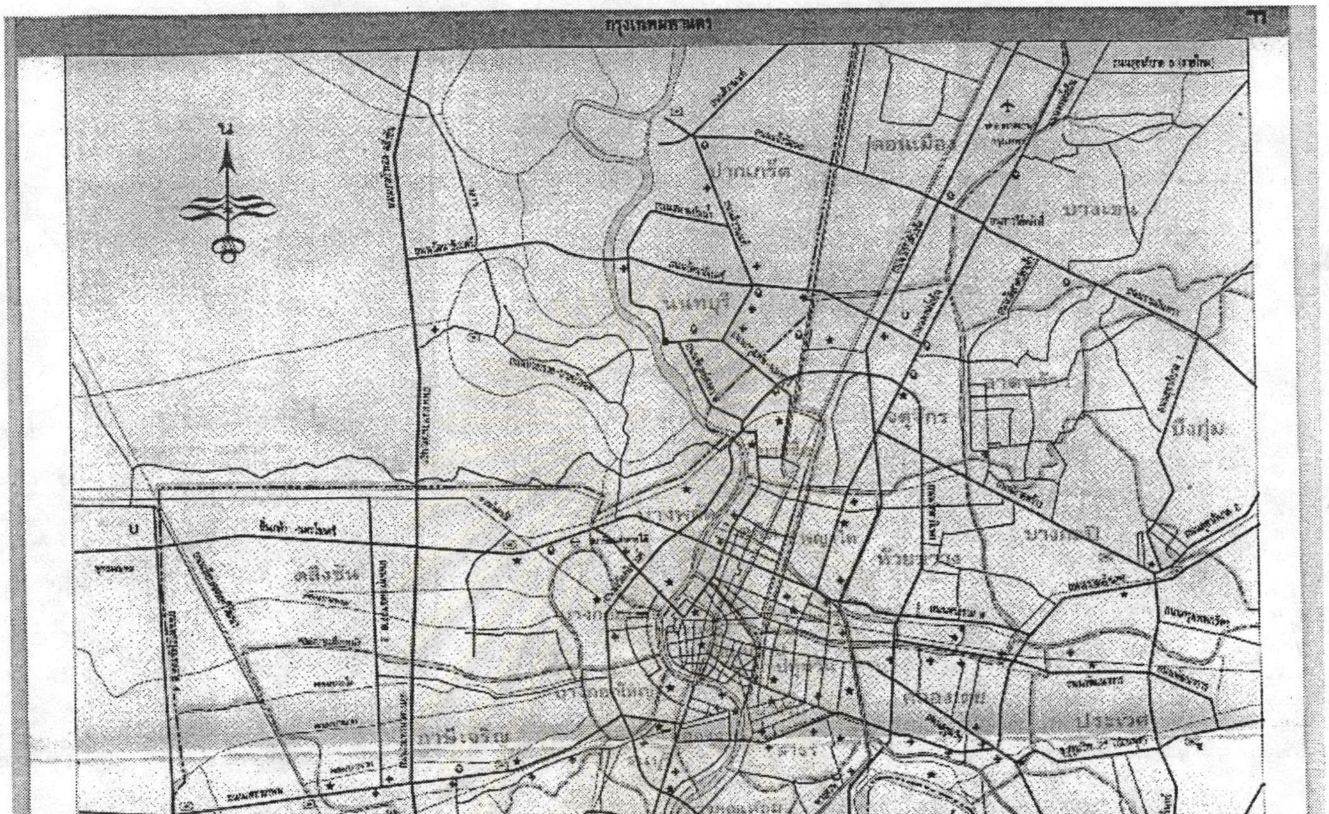
สำนักงานใหญ่สีลม
สาขาสะพานพระปิ่นเกล้า
สาขาหัวหมาก
สำนักสาขาพหลโยธิน
สาขาประตูน้ำ
สาขาบางแค
สาขาพระโขนง
สาขาลุมพินี
สาขาสุขสวัสดิ์
สาขาสำโรง
สาขาบางนา
สาขาบางกอกน้อย
สาขาถนนปู่เจ้าสมิงพราย
สาขาถนนเพชรบุรีตัดใหม่
สาขาถนนทองหล่อ
สาขาบางกรวย
สาขาถนนเทพารักษ์
สาขาบางปลาสร้อย จ.ชลบุรี
สาขาบางละมุง จ.ชลบุรี
สาขาเอี่ยมน้อย จ.สมุทรปราการ
สาขาปทุมธานี จ.ปทุมธานี
สาขาประตูช้างเผือก จ.เชียงใหม่
สาขาหาดใหญ่ จ.สงขลา
สาขาสงขลา จ.สงขลา
สำนักแลกเปลี่ยนเงินตราฯ ห้างนิวเวิลด์
ดีพาร์ทเม้นท์ สโตร์
ห้างเมอร์รี่คิงส์ วงเวียนใหญ่
ห้างโรบินสัน อนุสาวรีย์ชัยฯ
สำนักแลกเปลี่ยนเงินตราฯ พัทยา จ.ชลบุรี
ศูนย์การค้าพรพิงค์ไนท์บาซาร์ จ.เชียงใหม่
สาขาซอยอารี
สาขาสามแยกไฟฉาย
สาขาคลองจั่น
สาขาสะพานใหม่ดอนเมือง
สาขาบางเขน
[illegible]
สาขาสุรวงศ์
สาขาถนนประดิพัทธ์
สาขาสมุทรปราการ
สาขามีนบุรี
สาขารังสิต
สาขานานาเหนือ
สาขาลาดพร้าว 44
สาขาห้าแยกปากเกร็ด
สาขาถนนพัฒนาการ
สาขาปากคลองตลาด
สาขาชลบุรี
สาขาศรีราชา จ.ชลบุรี
สาขาพัทยา จ.ชลบุรี
สาขาระยอง
สาขาประตูน้ำพระอินทร์ จ.อยุธยา
สาขาศรีนครพิงค์ จ.เชียงใหม่
สาขาหาดใหญ่ย่านเพชรเกษม จ.สงขลา
สาขาภูเก็ต จ.ภูเก็ต
สำนักแลกเปลี่ยนเงินตราฯ สถานีรถไฟหัวลำโพง
ท่าอากาศยานกรุงเทพฯ
สหกรณ์กรุงเทพ เอกมัย
สำนักแลกเปลี่ยนเงินตราฯ หาดป่าตอง จ.ภูเก็ต
สาขาย่อยท่าอากาศยานกรุงเทพฯ

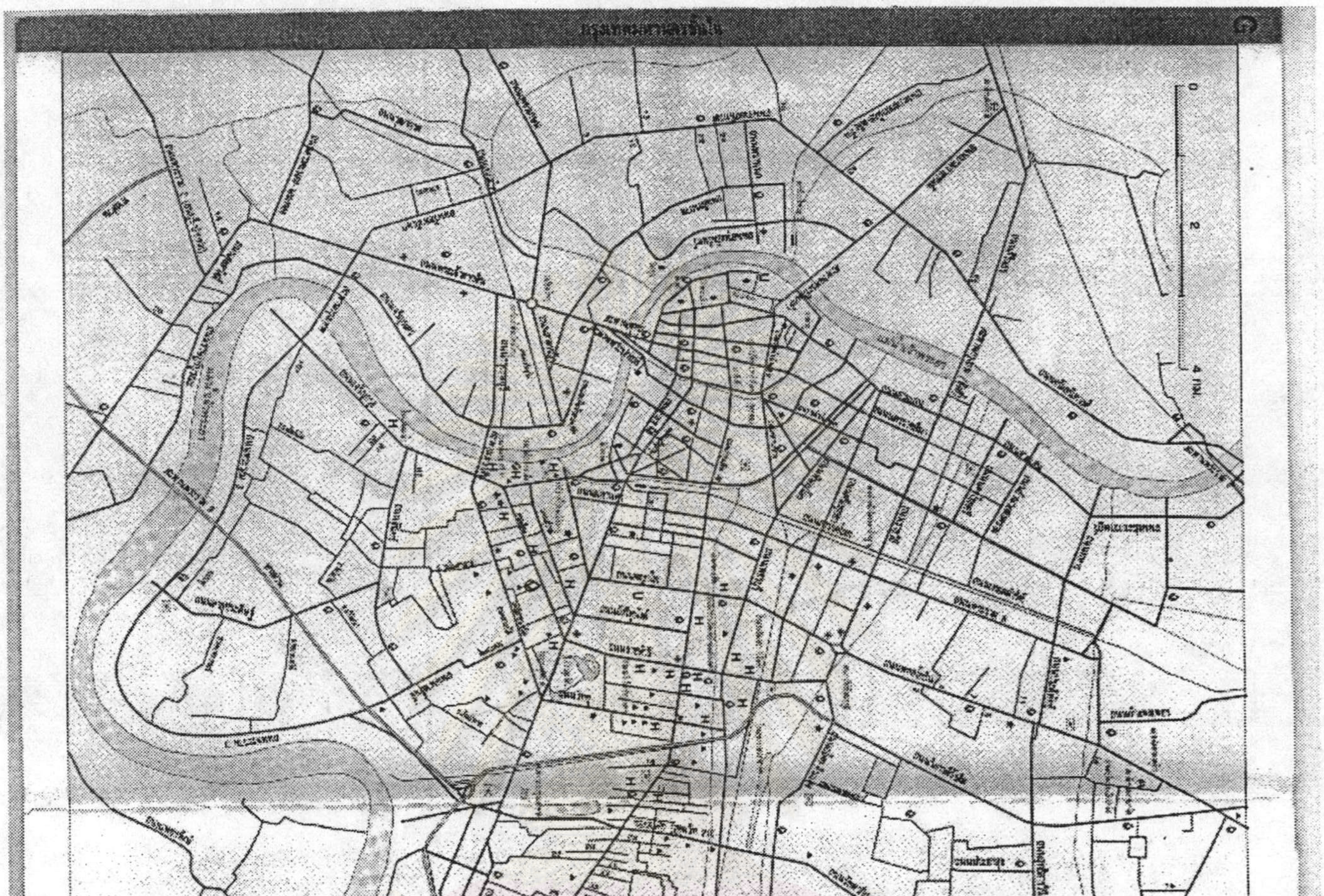

ภาคใต้ ภาคตะวันออกเฉียงเหนือ
อุดรธานี
หาดใหญ่-สงขลา
พัทลุง
สงขลา
ปัตตานี
นราธิวาส
มาเลเซีย
นครราชสีมา
ขอนแก่น
อุบลราชธานี
หนองคาย
อุดรธานี
นครพนม
สกลนคร
ขอนแก่น
กาฬสินธุ์
มหาสารคาม
มุกดาหาร
ร้อยเอ็ด
ยโสธร
สาธารณรัฐประชาธิปไตยประชาชนลาว
อุบลราชธานี
ศรีสะเกษ
บุรีรัมย์
สุรินทร์
100
0
100
200 กม.
กัมพูชา


ภาคตะวันออก - ภาคใต้
กาญจนบุรี
ราชบุรี
สมุทรสาคร
สมุทรปราการ
สมุทรสงคราม
เพชรบุรี
ชลบุรี
ระยอง
จันทบุรี
ตราด
อ่าวไทย
สหภาพพม่า
ประจวบคีรีขันธ์
ทะเลอันดามัน
ชุมพร
พัทยา-ชลบุรี
ระนอง
สุราษฎร์ธานี
นครศรีธรรมราช
พังงา
กระบี่
ภูเก็ต
พัทลุง
สงขลา
ปัตตานี
เกาะสมุย-สุราษฎร์ธานี
ภูเก็ต


D
ภาคเหนือ - ภาคกลาง
สาธารณรัฐประชาธิปไตยประชาชนลาว
สาธารณรัฐสังคมนิยมแห่งสหภาพพม่า
เชียงราย
แม่ฮ่องสอน
พะเยา
เชียงใหม่
ลำพูน
น่าน
ลำปาง
แพร่
อุตรดิตถ์
สุโขทัย
ตาก
พิษณุโลก
กำแพงเพชร
พิจิตร
เพชรบูรณ์
นครสวรรค์
อุทัยธานี
ชัยนาท
สิงห์บุรี
ลพบุรี
อ่างทอง
สุพรรณบุรี
สระบุรี
พระนครศรีอยุธยา
นครนายก
กาญจนบุรี
ปทุมธานี
ปราจีนบุรี
นครปฐม
นนทบุรี
กรุงเทพมหานคร
ฉะเชิงเทรา
ราชบุรี
สมุทรสาคร
สมุทรปราการ
สมุทรสงคราม
ชลบุรี
เพชรบุรี
หนองคาย
อุดรธานี
เลย
ขอนแก่น
ชัยภูมิ
นครราชสีมา
บุรีรัมย์
เชียงใหม่
เชียงราย
นครสวรรค์


กรุงเทพมหานคร
ดอนเมือง
บางเขน
มีนบุรี
หนองจอก
ลาดพร้าว
บึงกุ่ม
บางกะปิ
ห้วยขวาง
ลาดกระบัง
ประเวศ
พระโขนง
สมุทรปราการ
0 5 10 กม.
สัญลักษณ์แผนที่
ถนนสายหลัก
ถนนสายรอง ถนนซอย
ทางด่วน
ทางรถไฟ
สถานีรถไฟ
เส้นแบ่งจังหวัด
เส้นแบ่งเขต
แม่น้ำ, ลำคลอง
ตลาดสด
ท่าอากาศยานดอนเมือง
ที่ทำการไปรษณีย์
โรงพยาบาล
สถานทูต
วัด
สถานีตำรวจ
มหาวิทยาลัย
โรงแรม
ที่ตั้งสาขาธนาคารกรุงเทพจำกัด


กรุงเทพมหานคร
น
ปากเกร็ด
ดอนเมือง
บางเขน
นนทบุรี
จตุจักร
บึงกุ่ม
บางกะปิ
ห้วยขวาง
ตลิ่งชัน
ภาษีเจริญ
คลองเตย
ประเวศ
หนองแขม
พระโขนง
ราษฎร์บูรณะ
บางขุนเทียน
สมุทรปราการ
สมุทรสาคร
อ่าวไทย



ประวัติผู้เขียน

นายทวีศักดิ์ พุทธวรรณไชย เกิดวันที่ 21 กรกฎาคม พ.ศ. 2506 ที่กรุงเทพมหานครฯ สำเร็จการศึกษาปริญญาตรีวิศวกรรมศาสตรบัณฑิต ภาควิชาวิศวกรรมสำรวจ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2529 และเข้าศึกษาต่อในหลักสูตรวิศวกรรมศาสตรมหาบัณฑิต ที่จุฬาลงกรณ์มหาวิทยาลัย เมื่อ พ.ศ. 2531