

เอกสารอ้างอิง

- Michael A. Harrison, 1989. Sara Baase Computer Algorithms second edition Introduction to Design and Analysis, Addison Wesley Publishing Company.
- George Omura, 1990. The ABC's of AutoLISP Asian edition, Tech Publications.
- Robert M. Thomas, 1991. Encyclopedia AutoCAD Release 11 Asian edition, Tech Publications.
- AutoCAD Release 11, AutoCAD Development System Programmer's Reference.
- Robert L. Kruse. Data Structure and Program Design, second edition, Prentice-Hall International Editions.
- Esen Ozkarahan, 1990. Database Management Concepts, Design and Practice, Prentice-Hall International Editions.
- dBASE III PLUS version 1.10, Getting started Learning Applications Generator.
- AutoCAD Release 11, AutoLISP Programmer's Reference.
- Outside Plant design Guideline, Telephone organization of Thailand.
- องค์การโทรศัพท์แห่งประเทศไทย , ทะเบียนเลขหมายโทรศัพท์ผู้เช่า .

การพัฒนาร

## ภาคผนวก ก

ตารางที่ ก.1 แสดงหมายเลขกำกับข้อมูลประเภทต่างๆ ของ AutoCAD  
ส่วนข้อมูลพื้นฐาน (Normal Entity Definition Data)

หมายเลข	ชนิดข้อมูล
0	แสดงประเภทของค็กรานิก
1	ข้อมูลตัวอักษร (ประเภทข้อมูลตัวอักษร)
2	ชื่อข้อมูล
3-4	ชื่อข้อมูลประเภทอื่นๆ
5	หมายเลขประจำตัวของค็กรานิกเป็นเลขฐานสิบหก
6	ประเภทของเส้น
7	ประเภทตัวอักษร
8	ชื่อชั้นของภาพ
9	กำหนดชื่อตัวแปร (DXF Format)
10	ค่าพิกัดจาก จุดเริ่มต้น
11-37	ค่าพิกัดอื่นๆ
38	ค่าระดับ
39	ความหนาของค็กรานิก
40-48	จำนวนจริง (มาตราส่วน, ความสูงตัวอักษร)
50-58	ข้อมูลเกี่ยวกับมุมต่างๆ
62	หมายเลขสี
66	องค็กรานิกถัดไป
70-78	จำนวนเต็ม
210, 220, 230	องค็กรานิกทิศทาง X, Y, Z
999	ค่าบรรยายต่างๆ

ตารางที่ ก.2 แสดงหมายเลขกำกับข้อมูลประเภทต่างๆ ของ AutoCAD  
ข้อมูลส่วนขยาย (Extended Entity Data)

หมายเลข	ชนิดข้อมูล
1000	ASCII text ความยาวไม่เกิน 255 ตัวอักษร
1001	ชื่อข้อมูลจัดทำทะเบียน (ไม่เกิน 31 ตัวอักษร)
1002	อักษรควบคุม ข้อมูลส่วนขยาย
1003	ชื่อชั้นของภาพ ข้อมูลส่วนขยาย
1005	หมายเลขประจำตัววงศ์กราฟิกเป็นเลขฐานสิบหก
1010, 1020, 1030	ค่าพิกัดฉาก X, Y, Z
1011, 1021, 1031	ค่าพิกัดฉาก X, Y, Z ในระบบรวม
1012, 1022, 1032	ระยะจัด X, Y, Z ในระบบรวม
1013, 1023, 1033	ทิศทาง X, Y, Z ในระบบรวม
1040	จำนวนจริง (floating point)
1041	ระยะทาง
1042	มาตราส่วน
1070	จำนวนเต็ม 16 บิต
1071	จำนวนนับ 32 บิต

## ภาคผนวก ข

การใช้งานระบบ ACAD-UIS

การทำงานของระบบ ACAD-UIS แบ่งออกเป็น 4 ส่วนใหญ่ คือ

1. การกำหนดสภาพแวดล้อม (SETUP ENVIRONMENT)
2. การจัดการฐานข้อมูล (DATABASE MANIPULATION)
3. การค้นหาข้อมูล (DATA QUERY)
4. การนำข้อมูลเข้าและการแสดงผล (INPUT/OUTPUT)

โดยมีรายละเอียดการใช้งานดังนี้

1. การกำหนดสภาพแวดล้อม (SETUP ENVIRONMENT)

1.1 สร้างไฟล์ควบคุมโครงสร้างฐานข้อมูล (CONTROL FILE) โดยการใช้ EDITOR สร้างไฟล์ควบคุม (\*.DBS) ตามรูปแบบที่กำหนดดังนี้

- 1.1.1 นามสกุลของไฟล์ต้องเป็น .DBS
- 1.1.2 ชื่อฟิลด์ของฐานข้อมูล ไม่เกิน 10 ตัวอักษร (คอลัมน์ 1-10)
- 1.1.3 ชนิดของฟิลด์ 3 ตัวอักษร (คอลัมน์ 11-13)
  - HAN หมายเลขประจำตัวองค์กราพิก
  - CHR ข้อมูลตัวอักษร
  - NUM ข้อมูลตัวเลข
  - DMY วัน เดือน ปี
- 1.1.4 ความกว้างฟิลด์ ในการเก็บข้อมูล (คอลัมน์ 16-20)
- 1.1.5 จำนวนทศนิยม กรณีข้อมูลตัวเลข (คอลัมน์ 21-25)

1.1.6 ข้อมูลไฟล์ควบคุมเก็บแบบตัวอักษร (ASCII FORMAT) ตัวอย่างเช่น ไฟล์  
NODE.DBS

EHAND	HAN	10	0
NODE_TYP	CHR	12	0
DESC	CHR	20	0

## 1.2 การกำหนดฐานข้อมูล (ACTIVE DATABASE)

การกำหนดประเภทของฐานข้อมูลในการใช้งานสามารถทำได้โดยใช้ฟังก์ชัน เพื่อ  
กำหนดสภาพแวดล้อมในการทำงาน (DBENV) ฟังก์ชัน DBENV มีรูปแบบการใช้งานดังนี้

Command : (DBENV "DBNAME")

โดย DBNAME คือ ชื่อของไฟล์ควบคุม เช่น NODE ให้พิมพ์ (DBENV "NODE") เป็น  
ต้น โดยฟังก์ชันจะแสดงชื่อฐานข้อมูลสำหรับใช้งาน ที่มุมล่างของ SCREEN MENU

## 1.3 การแก้ไขโครงสร้างฐานข้อมูล (DATABASE STRUCTURE MODIFICATION)

การแก้ไขโครงสร้างฐานข้อมูล เช่น เปลี่ยนชื่อฟิลด์ ความกว้าง หรือการเพิ่มฟิลด์ใน  
การจัดเก็บข้อมูล โดยมีคำสั่งในการแก้ไขโครงสร้างฐานข้อมูล (MODSTRUC)

คำสั่ง MODSTRUC ใช้ได้โดยตรงเช่นเดียวกับคำสั่งใน AutoCAD

รูปแบบของคำสั่ง MODSTRUC

Command : MODSTRUC

MODSTRUC จะจัดเก็บข้อมูลของฐานข้อมูลที่ใช้งานอยู่ไว้ในไฟล์ชั่วคราว โดยใช้ชื่อ  
ตามฐานข้อมูลที่ใช้งานมีนามสกุล .### เช่น NODE.### ซึ่งการทำงานจะหยุดให้ผู้แก้ไข  
ไฟล์ควบคุม (\*.DBS) แล้วไฟล์ที่จัดเก็บข้อมูลชั่วคราวตามความต้องการ แล้วจึงนำข้อมูลจาก  
เรคคอร์ดที่แก้ไขแล้ว จัดเก็บคืนให้กับองค์กราฟิกใดๆ

ประกอบด้วยฟังก์ชันหลัก 2 ส่วน คือ

- (MODSTROUT) ; เก็บข้อมูลไว้ในไฟล์ชั่วคราว
- (MODSTRIN) ; นำข้อมูลจากไฟล์ชั่วคราว (ASCII Format)  
; จัดเก็บร่วมกับองค์กราฟิก

## 2. การจัดการฐานข้อมูล (DATABASE MANIPULATION)

2.1 STDATT เพื่อกำหนดข้อมูลพื้นฐาน ให้กับองค์กราฟิกสำหรับงานสำธารณูปโภค เป็นการจัดเก็บฐานข้อมูล "LINE" ให้กับองค์กราฟิก Line และ จัดเก็บฐานข้อมูล "NODE" ให้กับองค์กราฟิก Node

รูปแบบของคำสั่ง STDATT

Command : STDATT

STDATT ให้ผู้ใช้เลือกองค์กราฟิกสำธารณูปโภค และทำการค้นหาองค์กราฟิกที่ต่อเนื่องกันทั้งระบบ ณ ตำแหน่งจุดต่อใดๆ จะทำการเขียนองค์กราฟิก Node เพื่อการกำหนดทิศทางของสำธารณูปโภค

ในการกำหนดทิศทางองค์กราฟิก STDATT จะให้ผู้ใช้เลือกกำหนดทิศทางของ Line ใดๆ ตามความต้องการ ซึ่งเป็นการกำหนดทิศทางจาก Node หนึ่งไปยังอีก Node ของ Line นั้นๆ

2.2 ACT-REC การกำหนด Attribute ของฐานข้อมูลที่ใช้งานอยู่ จัดเก็บไว้ใน Buffer

รูปแบบของคำสั่ง ACT-REC

Command : ACT-REC

จะปรากฏข้อความ

Active attribute by :

>>current Attribute/extract Entity :

โดยผู้ใช้เลือกวิธีการกำหนด Attribute จากบล็อกเดิม หรือเลือกองค์กราฟิก เพื่อนำ Attribute จากองค์กราฟิกนั้นเก็บไว้ใน Buffer

2.3 APATT การเพิ่มข้อมูลให้กับองค์กราฟิก ในฐานข้อมูลที่ใช้งานอยู่ในขณะนั้น โดยเลือกองค์กราฟิกเพื่อจัดเก็บข้อมูล

รูปแบบของคำสั่ง APATT

Command : APATT

จะปรากฏข้อความ ให้เลือกองค์กราฟิก

Select object :

ทำการเลือกองค์กราฟิก โดยใช้ MOUSE ถ้าองค์กราฟิกนั้นมีข้อมูลแล้ว จะปรากฏข้อความผิดพลาด "DATA RECORD ALREADY EXIST" แต่ถ้ายังไม่มีข้อมูล APATT จะแสดงพื้นที่ในการกำหนดข้อมูลแต่ละฟิลด์ เมื่อผู้ใช้ป้อนข้อมูลและยอมรับว่าข้อมูลนั้นถูกต้อง APATT จะจัดเก็บข้อมูลเรคคอร์ดนั้นไว้กับองค์กราฟิก



พื้นที่ใช้การกำหนดข้อมูล

Edit Attributes	
EHAND	FE
DP_ID	001
DP_TYP	DP2
PAIR_NO	1-20
INS_DATE	15/06/1992
Ok	Cancel

2.4 EDATT การแก้ไขข้อมูลที่จัดเก็บไว้กับองค์กราฟิก สามารถทำงานกับฐานข้อมูลที่ใช้งานขณะนั้น โดยเลือกองค์กราฟิกที่ต้องการแก้ไขข้อมูล

### รูปแบบของคำสั่ง EDATT

Command : EDATT

จะปรากฏข้อความให้เลือกองค์กราฟิก

Select Object :

ทำการเลือกองค์กราฟิก โดยใช้ MOUSE ถ้าองค์กราฟิกไม่มีข้อมูลจัดเก็บอยู่ จะปรากฏข้อความผิดพลาด "NO DATARECORD !!! <ชื่อฐานข้อมูล> ถ้ามีข้อมูลจะปรากฏพื้นที่ในการแก้ไขข้อมูล พร้อมทั้งข้อมูลเก่า เมื่อทำการแก้ไขและยอมรับการแก้ไขข้อมูล EDATT จะนำข้อมูลที่แก้ไขแล้วจัดเก็บแทนที่ข้อมูลเก่า

พื้นที่ในการแก้ไขข้อมูล

Edit Attributes	
EHAND	FE
DP_ID	001
DP_TYP	DP2
PAIR_NO	1-20
INS_DATE	15/06/1992
Ok	Cancel

2.5 (GETATTR) ฟังก์ชันเพื่อเรียกข้อมูลที่จัดเก็บไว้มาใช้งาน โดยต้องส่งค่าตัวแปร (Argument) ให้กับฟังก์ชัน

รูปแบบของฟังก์ชัน

Command : (GETATTR EN CODE)

โดย EN คือ ชื่อของค็กราฟิก (ENTITY NAME)

CODE คือ - หมายเลขกำกับข้อมูล ของ AutoCAD หรือ  
- ชื่อฟิลด์ของฐานข้อมูลที่ใช้งาน

ฟังก์ชันจะให้ค่าของข้อมูลตามตัวแปรที่ส่งให้แก่ฟังก์ชัน (ตัวแปร CODE) เช่น CODE คือ 62 จะได้ค่าหมายเลขสีของวงค็กราฟิก เป็นต้น

2.6 (PUTATTR) ฟังก์ชันเพื่อกำหนดค่าข้อมูลให้กับของค็กราฟิก โดยต้องส่งค่าตัวแปร (Argument) ให้กับฟังก์ชัน

รูปแบบของฟังก์ชัน

Command : (PUTATTR EN CODE XVAL)

โดย EN คือ ชื่อของค็กราฟิก (ENTITY NAME)

CODE คือ - หมายเลขกำกับข้อมูล และ AutoCAD หรือ  
- ชื่อฟิลด์ของฐานข้อมูล

XVAL คือ ค่าข้อมูลกำหนดให้ของค็กราฟิก

ฟังก์ชันจะนำค่าของ XVAL แทนที่ข้อมูลเก่าตามตัวแปรที่ส่งให้แก่ฟังก์ชัน (ตัวแปร CODE)

2.7 (MODSS) ฟังก์ชันเพื่อการแก้ไขข้อมูลจากกลุ่มองค์กราฟิกต่างๆ โดยต้องส่งค่าตัวแปร (Argument) ให้กับฟังก์ชัน

รูปแบบของฟังก์ชัน

Command : (MODSS SS CODE XVAL)

โดย SS คือ กลุ่มขององค์กราฟิก (Selection set)  
 CODE คือ - หมายเลขกำกับข้อมูล ของ AutoCAD หรือ  
 - ชื่อฟิลด์ของฐานข้อมูล  
 XVAL คือ ค่าข้อมูลที่ต้องการแก้ไข

ฟังก์ชัน จะนำค่า XVAL แก้ไขให้กับกลุ่มขององค์กราฟิกตามตัวแปรที่ส่งให้กับฟังก์ชัน (ตัวแปร CODE)

ฟังก์ชันนี้สามารถใช้ทำการแก้ไขลักษณะขององค์กราฟิก เช่น สี ประเภทของเส้น เพื่อใช้ในการแสดงผลข้อมูลกลุ่มองค์กราฟิกที่ได้จากการวิเคราะห์

#### หมายเหตุ

1. EN ในข้อ 2.5 และ 2.6 ถ้าไม่ใช่ชื่อขององค์กราฟิก ฟังก์ชันจะให้เลือกองค์กราฟิก
2. SS ในข้อ 2.7 ถ้าไม่ใช่กลุ่มองค์กราฟิกฟังก์ชันจะให้เลือกกลุ่มองค์กราฟิกโดยคำสั่งของ AutoCAD ในการเลือกองค์กราฟิก ซึ่งสามารถใช้ Window, Crossing

### 3. การค้นหาข้อมูล (DATA QUERY)

ข้อมูลที่จัดเก็บไว้กับองค์กราฟิค สามารถทำการค้นหาได้โดยใช้เงื่อนไขของข้อมูล  
ฟิลด์ที่กำหนด หรือโดยใช้ชื่อฐานข้อมูลที่ต้องการค้นหา มีรายละเอียดดังนี้

3.1 (SSAPP) ฟังก์ชันเพื่อทำการค้นหาข้อมูล โดยใช้ชื่อของฐานข้อมูลโดยการส่งค่า  
ตัวแปร (Argument) ให้กับฟังก์ชัน

รูปแบบของฟังก์ชัน

Command : (SSAPP SS DBNAME)

โดย SS คือ กลุ่มขององค์กราฟิค  
- 0 หมายถึงองค์กราฟิคทั้งหมด

DBNAME คือ ชื่อฐานข้อมูลที่ต้องการ

ฟังก์ชัน SSAPP จะค้นหาองค์กราฟิคที่มีฐานข้อมูลที่กำหนดจัดเก็บอยู่ ไว้ในตัวแปร  
ENS และส่งผลกลุ่มองค์กราฟิคในตัวแปร ENS ให้โปรแกรมหลักที่เรียกใช้ฟังก์ชัน

3.2 (SSFLD) ฟังก์ชันเพื่อค้นหาข้อมูล โดยใช้เงื่อนไขของฟิลด์ต่างๆ โดยการส่งค่าตัวแปร (Argument) ให้กับฟังก์ชัน

รูปแบบของฟังก์ชัน

Command : (SSFLD SS FLD OPER XVAL)

โดย SS คือ - กลุ่มขององค์กราฟิก  
- ถ้าไม่ใช่กลุ่มองค์กราฟิก จะให้เลือกองค์กราฟิก

FLD คือ ชื่อฟิลด์ของฐานข้อมูล

OPER คือ ฟังก์ชันในการเปรียบเทียบ เช่น =, /=, <, >, <=, >=

XVAL คือ ค่าที่ใช้เปรียบเทียบกับข้อมูลของฟิลด์

ผลการทำงานของฟังก์ชัน คือกลุ่มองค์กราฟิกตามเงื่อนไขที่กำหนด เก็บไว้ในตัวแปร ENS และผลดังกล่าวถูกส่งค่ากลับให้โปรแกรมหลักด้วย

3.3 (SSCNT) ฟังก์ชันเพื่อค้นหาข้อมูลที่มีตัวอักษรที่กำหนด เป็นองค์ประกอบของข้อมูลในฟิลด์ที่กำหนด โดยการส่งค่าตัวแปร (Argument) ให้กับฟังก์ชัน

รูปแบบของฟังก์ชัน

Command : (SSCNT SS FLD CONTENT)

โดย SS คือ กลุ่มขององค์กราฟิก

FLD คือ ชื่อฟิลด์ของฐานข้อมูล

CONTENT คือ ข้อมูลตัวอักษรที่ต้องการค้นหา

ผลการทำงานของฟังก์ชัน คือกลุ่มขององค์กราฟิกที่ข้อมูลของฟิลด์ที่กำหนด มีค่าของ CONTENT เป็นองค์ประกอบของข้อมูล เก็บไว้ในตัวแปร ENS และผลดังกล่าวถูกส่งค่ากลับให้โปรแกรมหลักที่เรียกใช้ฟังก์ชันนี้

3.4 (SSLU) ฟังก์ชันเพื่อค้นหาข้อมูล ที่มีค่าของฟิลด์ตามช่วงข้อมูลที่กำหนด โดยการส่งค่าตัวแปร (Argument) ให้กับฟังก์ชัน

รูปแบบของฟังก์ชัน

Command : (SSLU SS FLD LOW UP)

โดย	SS	คือ	กลุ่มขององค์กราฟิก
	FLD	คือ	ชื่อฟิลด์ของฐานข้อมูล
	LOW	คือ	ค่าตัวเลขน้อย
	UP	คือ	ค่าตัวเลขมาก

ผลการทำงานของฟังก์ชัน คือกลุ่มขององค์กราฟิกที่มีค่าของฟิลด์เป็นค่าตัวเลข มีค่าน้อยกว่าตัวเลขน้อย หรือมีค่ามากกว่าตัวเลขมาก เก็บไว้ในตัวแปร ENS และค่าดังกล่าวถูกส่งกลับให้โปรแกรมหลักที่เรียกใช้ฟังก์ชันนี้

3.5 (SSRNG) ฟังก์ชันเพื่อค้นหาข้อมูล ที่มีค่าของฟิลด์ตามช่วงที่กำหนด โดยการส่งค่าตัวแปร (Argument) ให้กับฟังก์ชัน

รูปแบบของฟังก์ชัน

Command : (SSRNG SS FLD LOW UP)

โดย	SS	คือ	กลุ่มขององค์กราฟิก
	FLD	คือ	ชื่อฟิลด์ของฐานข้อมูล
	LOW	คือ	ค่าตัวเลขน้อย
	UP	คือ	ค่าตัวเลขมาก

ผลการทำงานของฟังก์ชัน คือกลุ่มขององค์กราฟิกที่มีค่าของฟิลด์เป็นตัวเลข มีค่าอยู่ระหว่าง หรือเท่ากับค่าตัวเลขน้อยและค่าตัวเลขมาก เก็บไว้ในตัวแปร ENS และค่าดังกล่าวถูกส่งกลับให้โปรแกรมหลักที่เรียกใช้ฟังก์ชันนี้

3.6 (BOOL2) ฟังก์ชันเพื่อจัดการกับกลุ่มองค์กราฟิก 2 กลุ่ม โดยการส่งค่าตัวแปร (Argument) ให้กับฟังก์ชัน

รูปแบบของฟังก์ชัน

Command : (BOOL2 OPER S1 S2)

โดย OPER คือ วิธีการจัดการกลุ่มองค์กราฟิก

- "I" = INTERSECT
- "S" = SUBTRACT
- "U" = UNION

S1 คือ กลุ่มองค์กราฟิก ชุดที่ 1

S2 คือ กลุ่มองค์กราฟิก ชุดที่ 2

ผลการทำงานของฟังก์ชัน คือกลุ่มองค์กราฟิกตามวิธีการจัดการกลุ่มกราฟิกที่ส่งให้กับฟังก์ชัน และกลุ่มองค์กราฟิกถูกส่งกลับให้โปรแกรมหลักที่เรียกใช้ฟังก์ชันนี้

#### หมายเหตุ

1. กลุ่มขององค์กราฟิกส่งให้กับฟังก์ชัน สามารถค้นหาข้อมูลตามเงื่อนไขต่างๆ ตามลำดับขั้นต่อเนื่องได้ ทำให้สามารถสร้างเงื่อนไขในการค้นหาข้อมูลหลายเงื่อนไขซ้อนกันได้

2. ชื่อฟิลด์ของฐานข้อมูล ถ้าไม่ใช่ฐานข้อมูลที่ใช้งานต้องกำหนดชื่อฐานข้อมูลกำกับไว้ด้วย เช่น "LINE->L\_LENGTH" โดยมีเครื่องหมาย "->" ระหว่างชื่อฐานข้อมูล และชื่อฟิลด์เป็นต้น

3. รูปแบบการส่งค่ากลับโปรแกรมหลัก

Command : (SETQ S1 (SSFLD SS "DESC" = "CABLE"))

ผลที่ได้คือกลุ่มองค์กราฟิกตามเงื่อนไขเก็บไว้ในตัวแปร ENS และส่งค่านั้นเก็บไว้ในตัวแปร S1 ของโปรแกรมหลัก



#### 4. การนำข้อมูลเข้าและการแสดงผล (INPUT/OUTPUT)

ACAD-UIS สามารถนำข้อมูลที่จัดเก็บจากโปรแกรมอื่นๆ เช่น LOTUS-123 dBASE ซึ่งจัดเก็บไว้ในไฟล์ข้อมูลตัวอักษร (ASCII Format) นำเข้าสู่ระบบเพื่อจัดเก็บร่วมกับบงค์กรราฟิค โดยที่ใช้เลขประจำตัวบงค์กรราฟิคเป็นส่วนเชื่อมต่อข้อมูล และสามารถนำข้อมูลที่จัดเก็บไว้แสดงผลในรูปแบบต่างๆ ได้ โดยใช้ฟังก์ชันต่างๆ ดังนี้

4.1 (SEES) ฟังก์ชันเพื่อใช้ในการแสดงผลกลุ่มบงค์กรราฟิคที่เลือกไว้แล้ว โดยต้องส่งค่าตัวแปร (Argument) ให้กับฟังก์ชัน

รูปแบบของฟังก์ชัน

Command : (SEES SS)

โดย SS คือ กลุ่มของบงค์กรราฟิค (Selection set)

ฟังก์ชันจะแสดงผลของกลุ่มบงค์กรราฟิคโดยการเปลี่ยนสีของบงค์กรราฟิคนั้นเป็นสีน้ำเงินชั่วคราว เพื่อให้ผู้ใช้สามารถเห็นบงค์กรราฟิคได้ชัดเจนยิ่งขึ้น

4.2 (MODSTRIN) ฟังก์ชันเพื่อนำข้อมูลที่จัดเก็บไว้ในไฟล์ข้อมูลตัวอักษร ตามแบบฟอร์มของแต่ละฐานข้อมูล (CDF Format) เข้าจัดเก็บไว้ร่วมกับบงค์กรราฟิค

รูปแบบของฟังก์ชัน

Command : (MODSTRIN)

ฟังก์ชันจะทำการอ่านข้อมูลจากไฟล์ DBNAME.### โดย DBNAME คือ ชื่อฐานข้อมูลที่ละบรรทัดๆ 1 เรคคอร์ด โดยที่หมายเลขประจำตัวบงค์กรราฟิคเป็นตัวเชื่อมในการนำข้อมูลแต่ละเรคคอร์ด จัดเก็บร่วมกับบงค์กรราฟิค

4.3 PR-DB-SCR คำสั่งเพื่อแสดงผลข้อมูลแต่ละเรคคอร์ดที่จัดเก็บไว้กับองค์ราฟิ  
กบนจอภาพ

รูปแบบของคำสั่ง

Command : PR-DB-SCR

จะปรากฏชื่อตัวแปรกลุ่มมององค์ราฟิที่เมนูด้านข้างให้เลือก

ผลคือมีการพิมพ์ข้อมูลที่ใช้งานอยู่ขององค์ราฟิที่เลือกทุกเรคคอร์ดบนจอภาพ

4.4 PR-DB-FILE คำสั่งเพื่อใช้การแสดงผล ข้อมูลแต่ละเรคคอร์ด ที่จัดเก็บไว้  
กับองค์ราฟิ โดยสามารถกำหนดชื่อไฟล์สำหรับการจัดเก็บผลที่ได้

รูปแบบของคำสั่ง

Command : PR-DB-FILE

- ปรากฏข้อความให้ผู้ใช้งานกำหนดชื่อไฟล์ข้อมูลในการจัดเก็บผลที่ได้
- ปรากฏชื่อตัวแปรกลุ่มมององค์ราฟิที่เมนูด้านข้างให้เลือก

ผลคือมีการพิมพ์ข้อมูลขององค์ราฟิที่เลือกทุกเรคคอร์ดลงในไฟล์ข้อมูลที่กำหนด

หมายเหตุ การกำหนดชื่อไฟล์ในการเก็บข้อมูล ถ้ากำหนดเป็น "PRN" จะพิมพ์ข้อมูลออกเครื่องพิมพ์ (PRINTER)

ดูรายละเอียดเพิ่มเติม คำสั่ง SELFLD เพื่อการแสดงผล



## ตัวอย่างการแสดงผลข้อมูลสายโทรศัพท์

EHAND	CABLE_ID	C_TYP	PAIR_NO	INS_DATE	CAB_ID
143C	01	AS50	1-50	10/06/1991	001
143B	01	AS50	1-50	10/06/1991	001
143A	01	AS20	1-20	10/06/1991	001
1439	01	AS20	1-20	10/06/1991	001
1438	02	AS150	1-150	15/07/1992	001
1437	02	AS150	1-150	15/07/1992	001
1436	02	AS20	1-20	15/07/1992	001
1435	02	AS100	1-100	15/07/1992	001
1434	02	AS100	1-100	15/07/1992	001
1433	02	AS20	21-40	15/07/1992	001
1432	02	AS20	51-70	15/07/1992	001
1431	03	AS100	1-75	15/07/1992	001
1430	03	AS100	1-75	15/07/1992	001
142F	03	AS50	1-50	15/07/1992	001
142E	03	AS50	1-30	15/07/1992	001
142D	03	AS10	1-10	15/07/1992	001
142C	03	AS10	1-10	15/07/1992	001
142B	04	AS25	1-25	15/07/1992	001

4.5 SELFLD คำสั่งเมื่อกำหนดชื่อฟิลด์ที่ต้องการแสดงผล สำหรับคำสั่ง PR-DB-SCR และ PR-DB-FILE

รูปแบบของคำสั่ง

Command : SELFLD

จะปรากฏชื่อฟิลด์ทั้งหมดของฐานข้อมูลที่ใช้งานอยู่ เพื่อให้ผู้ใช้เลือกฟิลด์ที่ต้องการแสดงผล หรือสลับตำแหน่งของฟิลด์ใดๆ ถ้าไม่ต้องการแสดงผลข้อมูลฟิลด์ใดก็ลบชื่อฟิลด์นั้นออกได้

การกำหนดฟิลด์เพื่อการแสดงผลสามารถกำหนดฟิลด์ของฐานข้อมูลอื่น นอกจากฐานข้อมูลที่ใช้งาน โดยอ้างถึงชื่อของฐานข้อมูลที่ต้องการ เช่น ฐานข้อมูลที่ใช้งานคือ "CABLE" แต่ต้องการแสดงผลความยาวซึ่งเก็บไว้ในฐานข้อมูล "LINE" ต้องพิมพ์ชื่อฟิลด์ "LINE->L\_LENGTH" โดยใช้เครื่องหมาย "->" ระหว่างชื่อฐานข้อมูล และชื่อฟิลด์

4.6 (SEEDIR) ฟังก์ชันเพื่อแสดงทิศทางของกลุ่มองค์กราฟิกประเภท Line  
รูปแบบของฟังก์ชัน

Command : (SEEDIR SS)

โดย SS คือ กลุ่มขององค์กราฟิกประเภท Line

การทำงานของฟังก์ชันจะแสดงลูกศรชี้แสดงทิศทางของ Line ต่างๆ ให้ผู้ใช้ตรวจสอบว่าทิศทางที่กำหนดไว้ถูกต้องหรือไม่

## ขั้นตอนการใช้งาน ACAD-UIS

1. กำหนดแผนพื้นฐานสำหรับพื้นที่ระบบสาธารณูปโภค
2. ออกแบบฐานข้อมูลประเภทต่างๆ ของระบบสาธารณูปโภค
3. สร้างไฟล์ข้อมูล AutoCAD ของระบบสาธารณูปโภค
4. กำหนดให้ไฟล์ แผนพื้นฐานเป็นไฟล์อ้างอิงของระบบสาธารณูปโภค (Xref)
5. กำหนดแนวทางของระบบสาธารณูปโภค โดยอ้างอิงแผนพื้นฐาน
6. จัดเก็บข้อมูลแต่ละประเภทร่วมกับองค์กราฟิก ของระบบสาธารณูปโภคโดยใช้คำสั่ง

และฟังก์ชันของ ACAD-UIS

7. สร้างโปรแกรมส่วนในการวิเคราะห์ผลลักษณะต่าง ๆ ของการใช้งาน
8. การแสดงผลข้อมูลจากการวิเคราะห์ สามารถทำได้โดยฟังก์ชันและคำสั่งพื้นฐาน

ของ ACAD-UIS

## ภาคผนวก ค

ตัวอย่างไฟล์ควบคุม

## ไฟล์ควบคุม "NODE.DBS"

EHAND	HAN	10	0	; Handle entity
NODE_TYP	CHR	12	0	; Node type
DESC	CHR	20	0	; Description

## ไฟล์ควบคุม "LINE.DBS"

EHAND	HAN	10	0	; Handle entity
FR_EHAND	HAN	10	0	; From handle entity
TO_EHAND	HAN	10	0	; To handle entity
LINE_TYP	CHR	12	0	; Line type
L_LENGTH	NUM	12	3	; Line length
DESC	CHR	20	0	; Description

## ไฟล์ควบคุม "JOINT.DBS"

EHAND	HAN	10	0	; Handle entity
JOINT_ID	CHR	12	0	; Joint identification
JOINT_TYP	CHR	15	0	; Joint type

## ไฟล์ควบคุม "CABINET.DBS"

EHAND	HAN	10	0	; Handle entity
CAB_ID	CHR	12	0	; Cabinet identification
CAB_TYP	CHR	15	0	; Cabinet type
PAIR_NO	CHR	12	0	; Secondary pair number
PRI_NO	CHR	12	0	; Primary pair number

## ไฟล์ควบคุม "CABLE.DBS"

EHAND	HAN	10	0	; Handle entity
CABLE_ID	CHR	12	0	; Cable identification
C_TYP	CHR	15	0	; Cable type
PAIR_NO	CHR	12	0	; Secondary pair number
INS_DATE	DMY	12	0	; Installation date
CAB_ID	CHR	12	0	; Cabinet identification

## ไฟล์ควบคุม "DP.DBS"

EHAND	HAN	10	0	; Handle entity
DP_ID	CHR	12	0	; Distribution point identification
DP_TYP	CHR	15	0	; Distribution point type
PAIR_NO	CHR	12	0	; Secondary pair number
INS_DATE	DMY	12	0	; Installation date
CABLE_ID	CHR	12	0	; Cable identification

## ไฟล์ควบคุม "SUBSCR.DBS"

EHAND	HAN	10	0	; Handle entity
SUB_ID	CHR	12	0	; Subscriber identification
STAT	NUM	5	0	; Status of subscriber
OWNER	CHR	25	0	; Owner's Name of subscriber
ADDRESS	CHR	40	0	; Address of subscriber
DP_ID	CHR	12	0	; Distribution point identification

## ภาคผนวก ง

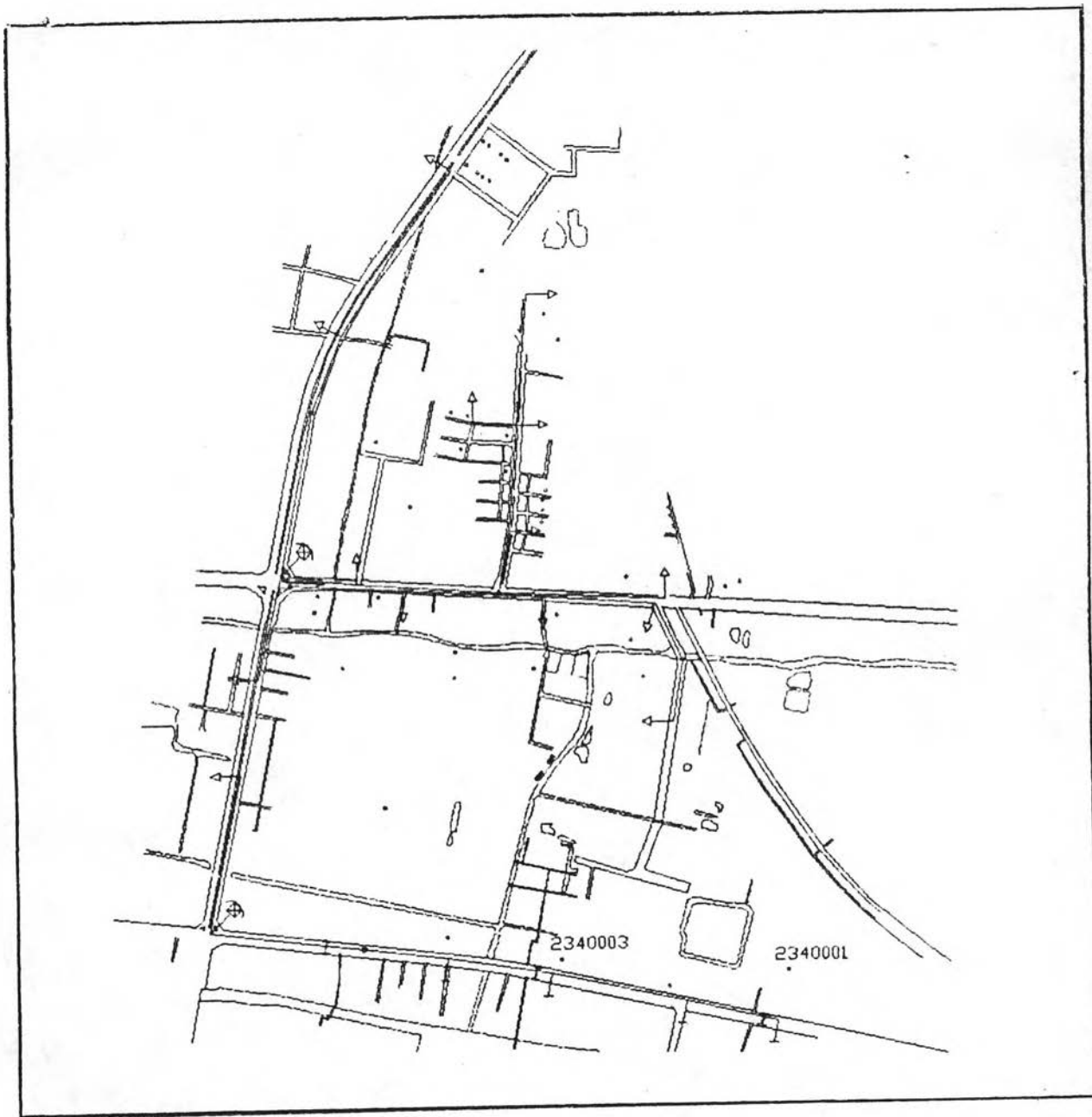
ตัวอย่างการแสดงผล

## ข้อมูลจุดกระจายสายโทรศัพท์ (DP)

EHAND	DP_ID	DP_TYP	PAIR_NO	INS_DATE	CABLE_ID
150E	, 002005	, DP1	, 1-25	, 15/06/1992	, 02
150D	, 002001	, DP2	, 1-20	, 15/06/1992	, 01
150C	, 002002	, DP2	, 21-40	, 15/06/1992	, 01
150B	, 002003	, DP2	, 51-75	, 15/06/1992	, 01
150A	, 002004	, DP2	, 76-100	, 15/06/1992	, 01
142A	, 001002	, DP1	, 21-40	, 10/06/1991	, 01
1427	, 001001	, DP1	, 1-20	, 10/06/1992	, 01
1425	, 001013	, DP1	, 1-25	, 15/07/1992	, 04
1423	, 001012	, DP1	, 51-75	, 15/07/1992	, 03
1422	, 001011	, DP1	, 31-50	, 15/07/1992	, 03
1421	, 001010	, DP1	, 21-30	, 15/07/1992	, 03
141F	, 001009	, DP1	, 1-10	, 15/07/1992	, 03
141E	, 001008	, DP1	, 101-120	, 15/07/1992	, 02
141C	, 001007	, DP1	, 81-100	, 15/07/1992	, 02
141B	, 001006	, DP1	, 71-80	, 15/07/1992	, 02
141A	, 001005	, DP1	, 51-70	, 15/07/1992	, 02
1419	, 001004	, DP1	, 21-40	, 15/07/1992	, 02
1418	, 001003	, DP1	, 1-20	, 15/07/1992	, 02



ตัวอย่างการแสดงผลเชิงกราฟิก



รูปที่ ง.1 ตัวอย่างการแสดงผลเชิงกราฟิก

## ภาคผนวก จ

โปรแกรม ACAD-UIS

```

;;; This routine set command name to ACAD
;;; for managing the utility information system
;;; by Mr. Boonlert Nitiattananon

; to set standard attribute
(defun C:STDATT ()
  (setq olderr *error*          ; Redefine error handler.
        *error* fiserr)
  (statt)
  (setq *error* olderr)        ; Reset the error function.
  (prin1)                      ; Quiet exit.
);end def

;;;-----

; to set active attribute record
(defun C:ACT-REC ( / cnd)
  (setq olderr *error*          ; Redefine error handler.
        *error* fiserr)
  (initget 1 "Attribute Entity")
  (prompt "\nActive attribute by :")
  (setq cnd
    (getkword "\n>>current Attribute/extract Entity : ")
  )
)

```

```

(if (= cnd "Entity")
  (actatt)
  (def_currec)
)

(setq *error* olderr)           ; Reset the error function.
(prin1)                         ; Quiet exit.
);end def

```

```
;;;-----
```

```
; to append attribute
```

```

(defun C:APATT ()
  (setq olderr *error*         ; Redefine error handler.
        *error* fiserr)
  (apatr 0)
  (setq *error* olderr)       ; Reset the error function.
  (prin1)                      ; Quiet exit.
);end def

```

```
;;;-----
```

```
; to edit attribute
```

```

(defun C:EDATT ()
  (setq olderr *error*         ; Redefine error handler.
        *error* fiserr)
  (edattr 0)
  (setq *error* olderr)       ; Reset the error function.
  (prin1)                      ; Quiet exit.
);end def

```

```
;;;-----
```

; to define direction of line

```
(defun C:DEF-DIR ()  
  (setq olderr *error*           ; Redefine error handler.  
        *error* fiserr)  
  (entdir 0)  
  (setq *error* olderr)         ; Reset the error function.  
  (prin1)                       ; Quiet exit.  
);end def
```

;;;

; to append subscriber

```
(defun C:APSUB ()  
  (setq olderr *error*           ; Redefine error handler.  
        *error* fiserr)  
  (apsubscr)  
  (setq *error* olderr)         ; Reset the error function.  
  (prin1)                       ; Quiet exit.  
);end def
```

;;;

; to append distribution point

```
(defun C:APDP ()  
  (setq olderr *error*           ; Redefine error handler.  
        *error* fiserr)  
  (apdp 1)  
  (setq *error* olderr)         ; Reset the error function.  
  (prin1)                       ; Quiet exit.  
);end def
```

;;;

; to append cabinet

(defun C:APCAB ())

(setq olderr \*error\* ; Redefine error handler.

\*error\* fiserr)

(apdp 0)

(setq \*error\* olderr) ; Reset the error function.

(prin1) ; Quiet exit.

);end def

;;;-----

; to modify structure of data base

(defun C:MODSTRUC ())

(setq olderr \*error\* ; Redefine error handler.

\*error\* fiserr)

(modstruc)

(setq \*error\* olderr) ; Reset the error function.

(prin1) ; Quiet exit.

);end def

;;;-----

; to Select field name of data base for report

(defun C:SELFLD ())

(setq olderr \*error\* ; Redefine error handler.

\*error\* fiserr)

(def\_fld)

(setq \*error\* olderr) ; Reset the error function.

(prin1) ; Quiet exit.

);end def

;;;-----

## ประวัติผู้เขียน

นายบุญเลิศ นิตวัฒนานนท์ เกิดเมื่อวันที่ 27 กันยายน พ.ศ. 2508 ที่อำเภอองครักษ์ จังหวัดนครนายก สำเร็จการศึกษาวิศวกรรมศาสตรบัณฑิตจากคณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ปีการศึกษา 2529 ปัจจุบันอยู่บ้านเลขที่ 18/129 ซอยจตุรพักตรพิมาน ถนนงามวงศ์วาน อำเภอเมือง จังหวัดนนทบุรี โทร. 580-4779 ประกอบวิชาชีพอิสระ

