บทที่ 3

คลาสที่เกี่ยวกับการติดต่อกับผู้ใช้ของโปรแกรมจำลองการทำงานวงจรไฟฟ้า

บทนี้จะกล่าวถึงคลาสที่เกี่ยวข้องกับการติดต่อกับผู้ใช้ของโปรแกรมจำลองการทำงานวงจรไฟฟ้าที่ต้องการจะออกแบบ โดยคลาสเหล่านี้จะสืบทอดมาจากคลาสในคลาสไลบรารี่ MFC และสัญลักษณ์ของคลาสที่นำมาจากคลาสไลบรารี่ MFC จะมีชื่อขึ้นต้นด้วย "C" และจะมีพื้นเป็นสีเทา

3.1 คลาสที่เกี่ยวกับการติดต่อกับผู้ใช้ที่เป็นมาตรฐานของไมโครซอฟต์วินโดวส์

คลาสในหัวข้อนี้เป็นคลาสที่เกี่ยวกับการติดต่อกับผู้ใช้ที่เป็นมาตรฐานของไมโครซอฟต์วินโดวส์ ซึ่งคลาสที่จะกล่าวเหล่านี้เป็นคลาสที่สืบทอดมาจากคลาสของคลาสไลบรารี่ MFC โดยที่คลาสไลบรารี่นี้ได้จัดเตรียมคลาสที่ครอบคลุมการจัดการด้านการติดต่อกับผู้ใช้ไว้ให้นักเขียนโปรแกรมบนไมโครซอฟต์วินโดวส์ในรูปแบบของโครงร่างงาน (Framework)

การสืบทอดคลาสต่าง ๆ ของการติดต่อกับผู้ใช้ของโปรแกรมที่จะออกแบบจากคลาสของคลาสไลบรารี่ MFC แสดงดังรูปที่ 3.1 ซึ่งจะกล่าวถึงแต่ละคลาสที่สืบทอดมาจากคลาสไลบรารี่ MFC ในรูปที่ 3.1 โดยสรุปเป็นข้อ ๆ ดังนี้

3.1.1 LEK

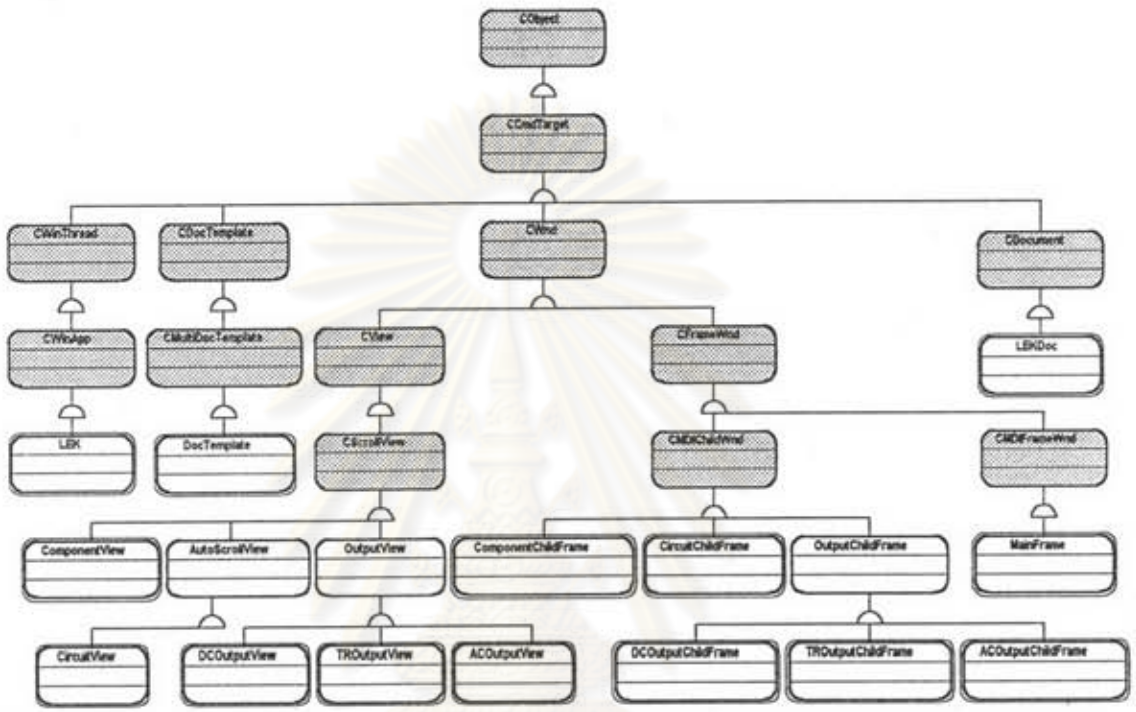
เป็นคลาสที่สำคัญของโปรแกรมต้นแบบ ซึ่งครอบคลุมการกำหนดตั้งค่าเริ่มต้นของการทำงานของโปรแกรม, การทำงานและการจบการทำงานของโปรแกรม

3.1.2 MainFrame

เป็นกรอบหน้าต่างหลัก (Main Frame Window) ของโปรแกรมต้นแบบ ทำหน้าที่จัดการกับวินโดวส์ที่เป็นลูกของมัน (Child Window) อย่างเช่น จัดหน้าต่างที่อยู่ภายในกรอบหน้าต่างหลักให้เรียงกันเป็นระเบียบ เป็นต้น

3.1.3 LEKDoc

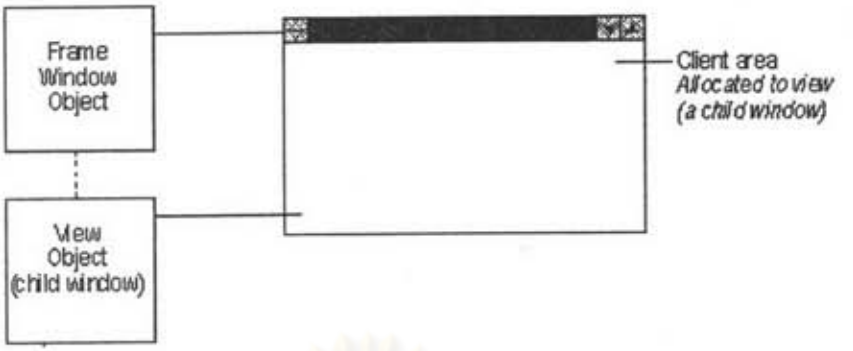
เก็บวงจรที่ผู้ใช้วาดในช่องหน้าต่าง Circuit View ซึ่งคลาส LEKDoc จะครอบคลุมการจัดเก็บวงจรลงไฟล์และอ่านวงจรจากไฟล์



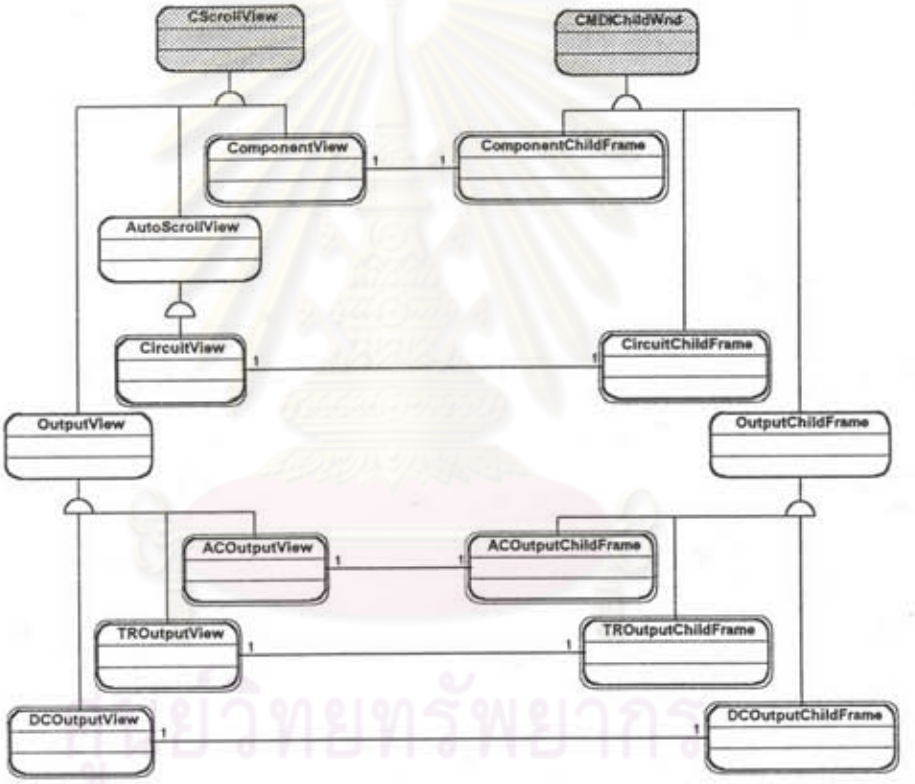
รูปที่ 3.1 ลำดับชั้นของคลาสที่สืบทอดมาจากคลาสของคลาสไลบรารี่ MFC

3.1.4 คลาสกรอบหน้าต่าง (Frame-Window Class) และคลาสช่องหน้าต่าง (View Class)

คลาสไลบรารี่ MFC กำหนดให้ช่องหน้าต่างถูกล้อมรอบด้วยกรอบหน้าต่าง ดังแสดงในรูปที่ 3.2 และจากรูปที่ 1.1 โปรแกรมที่จะออกแบบจะมีหน้าต่างทั้งหมด 5 ชนิด คือ หน้าต่าง Component View, Circuit View, DC Output View, TR Output View และ AC Output ทำให้ต้องสืบทอดคลาสกรอบหน้าต่างและคลาสช่องหน้าต่างจากคลาส CMDIChildWnd และ CScrollView ของคลาสไลบรารี่ MFC ดังแสดงในรูปที่ 3.1 และความสัมพันธ์ระหว่างคลาสของช่องหน้าต่างกับคลาสของกรอบหน้าต่างแสดงในรูปที่ 3.3



รูปที่ 3.2 ความสัมพันธ์ระหว่างวัตถุกรอบหน้าต่าง (Frame Object) กับวัตถุช่องหน้าต่าง (View Object)



รูปที่ 3.3 ความสัมพันธ์ระหว่างคลาสและวัตถุช่องหน้าต่างชนิดต่าง ๆ กับ คลาสและวัตถุกรอบหน้าต่างชนิดต่าง ๆ

3.1.4.1 CScrollView

กำหนดหน้าที่พื้นฐานเกี่ยวกับการแสดงข้อมูลในรูปแบบของรูปภาพ, การรับและตอบสนองข่าวสารจากวินโดวส์ และตอบสนองต่อคำสั่งของผู้ใช้ สำหรับคลาสที่สืบทอดจากคลาส CScrollView เช่น การรับค่าตำแหน่ง (x, y) ของตัวชี้ของเมาส์เมื่อมีการคลิ๊กใน

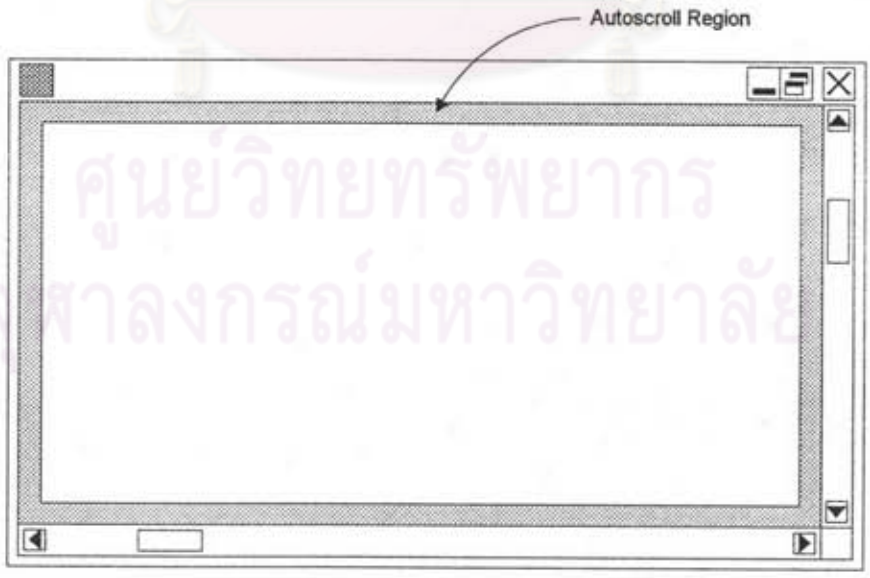
ช่องหน้าต่าง หรือ การเลื่อนข้อมูลที่อยู่ในช่องหน้าต่างเมื่อผู้ใช้กดปุ่มที่ Scroll Bar เป็นต้น

3.1.4.2 ComponentView

ทำหน้าที่สร้าง, กำหนดตำแหน่งที่อยู่ของ ComponentCreator ซึ่งเป็นตัวสร้างอุปกรณ์ที่ผู้ใช้สามารถเลือกนำไปสร้างวงจร, ส่งข่าวสารให้กับ ComponentCreator แต่ละตัวแสดงรูปของอุปกรณ์ที่ ComponentCreator นั้นสร้างที่ตำแหน่งที่กำหนด และเป็นตัวกลางในการติดต่อระหว่างผู้ใช้กับ ComponentCreator คือ เมื่อผู้ใช้ใช้เมาส์คลิ๊กในช่องหน้าต่าง ComponentView จะส่งตำแหน่ง (x, y) ของตัวชี้ไปให้กับ ComponentCreator รายละเอียดของ ComponentCreator จะกล่าวอีกครั้งในภายหลัง

3.1.4.3 AutoScrollView

เป็นเบสคลาสของ CircuitView ซึ่งคลาส AutoScrollView นี้ครอบคลุมการจัดการด้านการเลื่อนข้อมูลที่อยู่บน View โดยอัตโนมัติเมื่อผู้ใช้กดปุ่มด้านซ้ายของเมาส์ค้างแล้วเลื่อนตัวชี้ของเมาส์เข้าสู่บริเวณเลื่อนอัตโนมัติ (Autoscroll Region) ที่แสดงด้วยสีเทารอบ ๆ กรอบสี่เหลี่ยมในสุดในรูปที่ 3.4 และเมื่อผู้ใช้เลื่อนตัวชี้ออกนอกบริเวณเลื่อนอัตโนมัติ หรือปล่อยปุ่มของเมาส์ข้อมูลจะหยุดเลื่อน ทำให้ผู้ใช้สามารถเลื่อนอุปกรณ์เลือกจากหน้าต่าง Component View ไปในบริเวณที่มองไม่เห็นในหน้าต่าง Circuit View ได้ง่ายขึ้น



รูปที่ 3.4 บริเวณเลื่อนอัตโนมัติ (Autoscroll Region) ของ AutoScrollView

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

3.1.4.4 CircuitView

ทำหน้าที่แสดงรูปของอุปกรณ์ทุกตัวในวงจรที่อยู่ในช่องหน้าต่าง Circuit View และจัดการกับอุปกรณ์ภายในวงจรตามคำสั่งของผู้ใช้ที่เลือกจากเมนู เช่น การเพิ่ม, ลบอุปกรณ์ หรือ การกำหนดค่าพารามิเตอร์ของอุปกรณ์ เป็นต้น

3.1.4.5 OutputView

กำหนดทำหน้าที่พื้นฐานในการเก็บและแสดงผลของการจำลองการทำงานแบบต่าง ๆ ซึ่ง OutputView แบ่งออกเป็น 3 ชนิดตามชนิดของผลและวิธีการจำลองการทำงานวงจรไฟฟ้า คือ

1. DCOutputView สำหรับการจำลองเพื่อหาจุดทำงานสงบ
2. TROutputView สำหรับการจำลองการทำงานทางเวลา
3. ACOutputView สำหรับการจำลองการทำงานทางความถี่

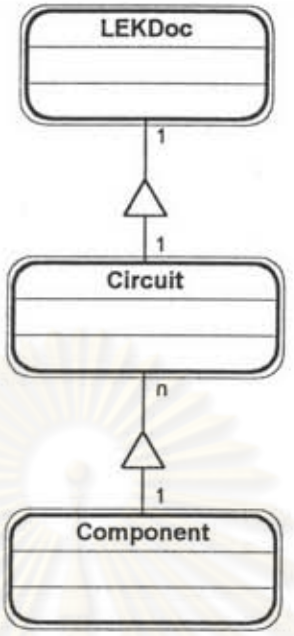
3.1.5 DocTemplate

ทำหน้าที่สร้างวัตถุของคลาส LEKDoc, CircuitView และ CircuitChild-Frame เมื่อผู้ใช้เปิดไฟล์ข้อมูลที่เก็บวงจร

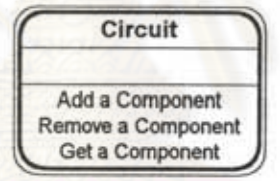
3.2 คลาสที่เกี่ยวกับวงจร

คลาสที่กล่าวในหัวข้อนี้เป็นคลาสที่เกี่ยวกับวงจรที่ผู้ใช้ป้อนให้กับโปรแกรมโดยการนำเอา Component ต่าง ๆ จากหน้าต่าง Component View มาประกอบกันเป็นวงจรในหน้าต่าง Circuit View ซึ่งคลาสนั้นก็คือ Circuit ประกอบด้วย Component ตั้งแต่ 0 ตัวจนถึง n ตัว และ Circuit นี้จะเป็นส่วนประกอบของ LEKDoc เพื่อให้วัตถุของคลาสช่องหน้าต่างทุกคลาสสามารถติดต่อกับ Circuit ได้ ลำดับชั้นแสดงส่วนประกอบของ LEKDoc และ Circuit ในรูปที่ 3.5

คลาส Circuit นี้ทำหน้าที่เหมือนกับรายการแบบเชื่อมโยงของ Component ชนิดต่าง ๆ มีวิธีที่สำคัญในการเพิ่ม, ลด และดึงเอา Component ที่ต้องการใน Circuit ดังแสดงในรูปที่ 3.6



รูปที่ 3.5 ลำดับชั้นแบบ Whole-Part Structure แสดงส่วนประกอบของ LEKDoc และ Circuit



รูปที่ 3.6 คลาสและวัตถุ Circuit

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

3.3 คลาสที่เกี่ยวกับอุปกรณ์และตัวสร้างอุปกรณ์

รูปที่ 3.7 แสดงลำดับชั้นของคลาสที่เกี่ยวกับอุปกรณ์และตัวสร้างอุปกรณ์ โดยจะกล่าวถึงคลาสในรูปที่ 3.7 ดังนี้

3.3.1 Shape

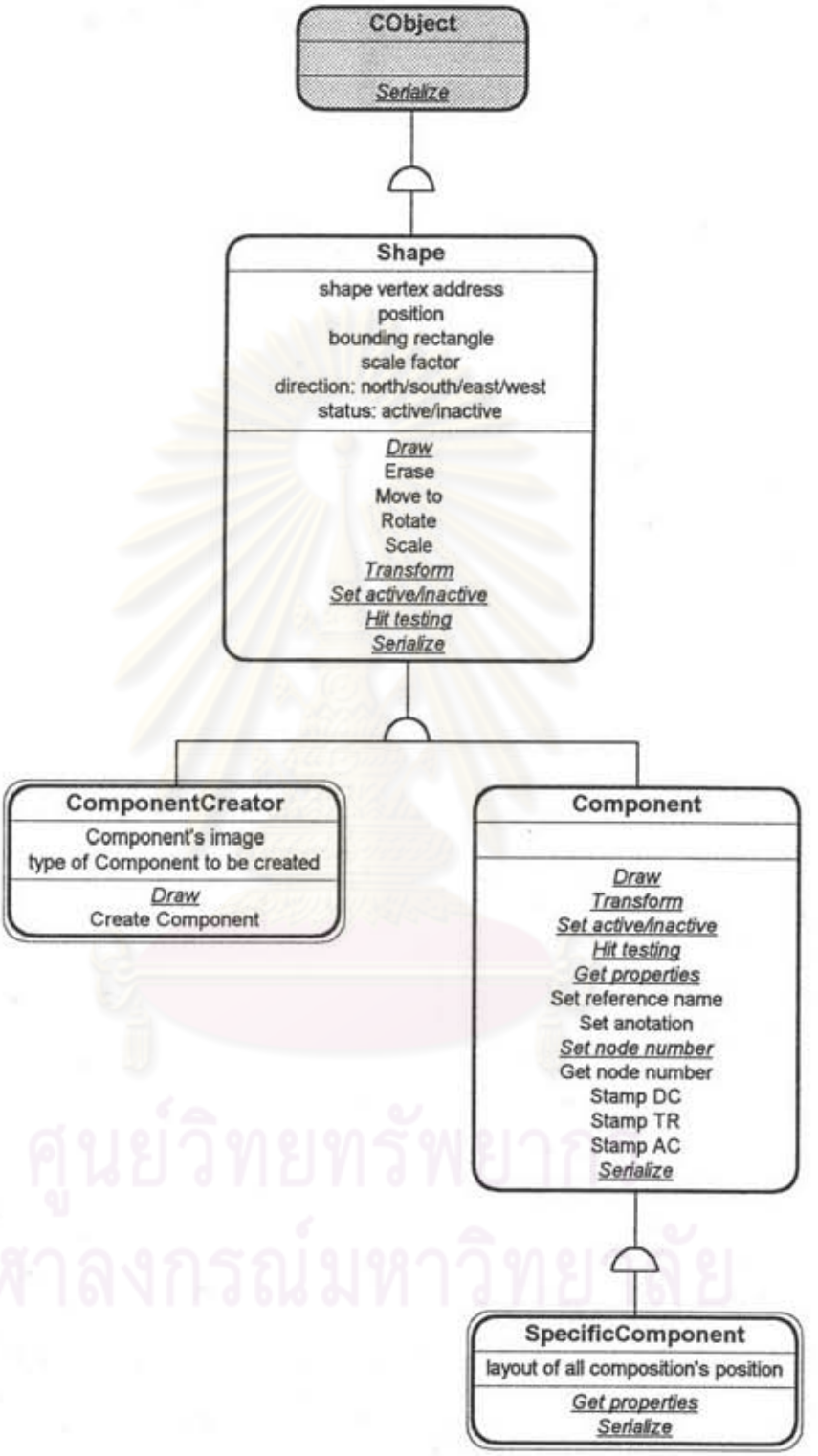
เป็นคลาสที่ครอบคลุมข้อมูลและวิธีการแสดงภาพ 2 มิติ ซึ่งมีคุณสมบัติและวิธีแสดงไว้ในรูปที่ 3.7 และจากรูปที่ 3.7 นี้จะเห็นว่าคลาส Shape สืบทอดมาจาคลาส CObject เพราะว่าต้องการวิธีในการเก็บคุณสมบัติของวัตถุลงไฟล์และอ่านคุณสมบัติของวัตถุจากไฟล์ ที่ได้รับมาจากการสืบทอดจาก CObject

3.3.2 ComponentCreator

ทำหน้าที่สร้าง Component เมื่อตำแหน่ง (x, y) ของตัวชี้ที่ได้รับจาก ComponentView อยู่บนตัวของ ComponentCreator แล้วส่ง Component ที่สร้างนั้นให้กับ ComponentView จัดการกับ Component นั้นต่อไป

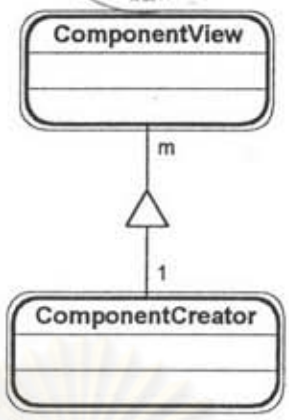
ในรูปที่ 1.1 รูปของ Component Creator ที่แสดงในช่องหน้าต่าง Component View นั้นจะเป็นรูปของส่วนประกอบที่นำไปใช้สร้างผังวงจรซึ่งสัมพันธ์กับ Component ที่ถูกสร้างโดย ComponentCreator นั้นเอง และ ComponentCreator เป็นสมาชิกของ ComponentView โดยมีจำนวนสมาชิกเท่ากับ m ตัว โดยที่ m เท่ากับจำนวนของส่วนประกอบที่ผู้ใช้สามารถเลือกนำไปวาดผังวงจร ดังลำดับชั้นในรูปที่ 3.8

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 3.7 ลำดับชั้นของคลาสที่เกี่ยวกับอุปกรณ์และตัวสร้างอุปกรณ์

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 3.8 ลำดับชั้นแสดงส่วนประกอบของ ComponentView

3.3.3 Component

เป็นคลาสที่ครอบคลุมข้อมูลและการจัดการพื้นฐานกับส่วนประกอบต่าง ๆ ของ Component ให้ทำงานร่วมกันและเป็นตัวกลางในการติดต่อระหว่างส่วนประกอบภายในกับวัตถุอื่นภายนอก

Component เป็นส่วนประกอบที่เล็กที่สุดของผังวงจรที่ผู้ใช้สามารถเคลื่อนย้ายส่วนประกอบนั้นได้ภายในช่องหน้าต่าง Circuit View ตัวอย่างของ Component ได้แก่ Resistor, Capacitor, Wire และ Probe ชนิดต่าง ๆ เป็นต้น

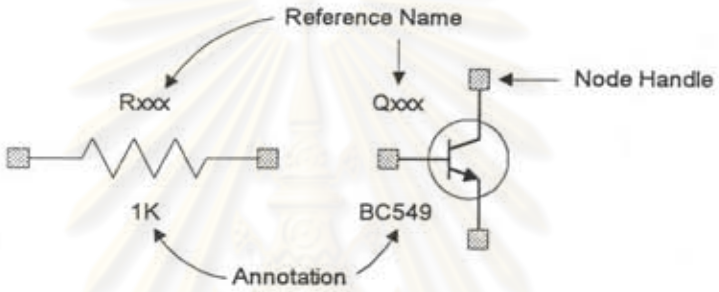
3.3.4 SpecificComponent

เป็นคลาสที่สืบทอดจากคลาส Component คลาสนี้แสดงไว้ในรูปที่ 3.7 เพื่อบอกว่าคลาสนี้จะถูกแทนด้วยคลาสของอุปกรณ์ชนิดใดชนิดหนึ่งที่ผู้ใช้สามารถเลือกนำไปสร้างวงจรได้ เช่น Resistor, Capacitor, Wire เป็นต้น ซึ่งอุปกรณ์เหล่านี้จะกล่าวถึงในหัวข้อ 3.5

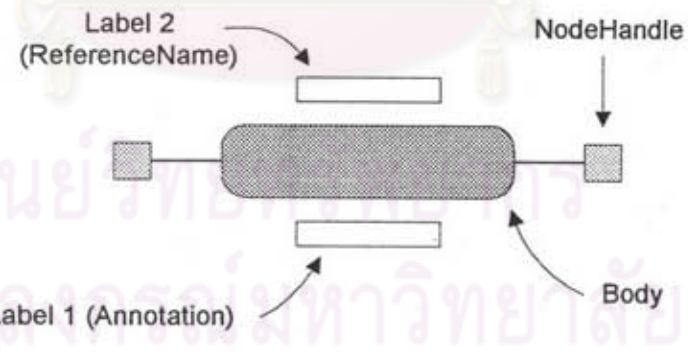
ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

3.4 รูปแบบทั่วไปของคลาส Component

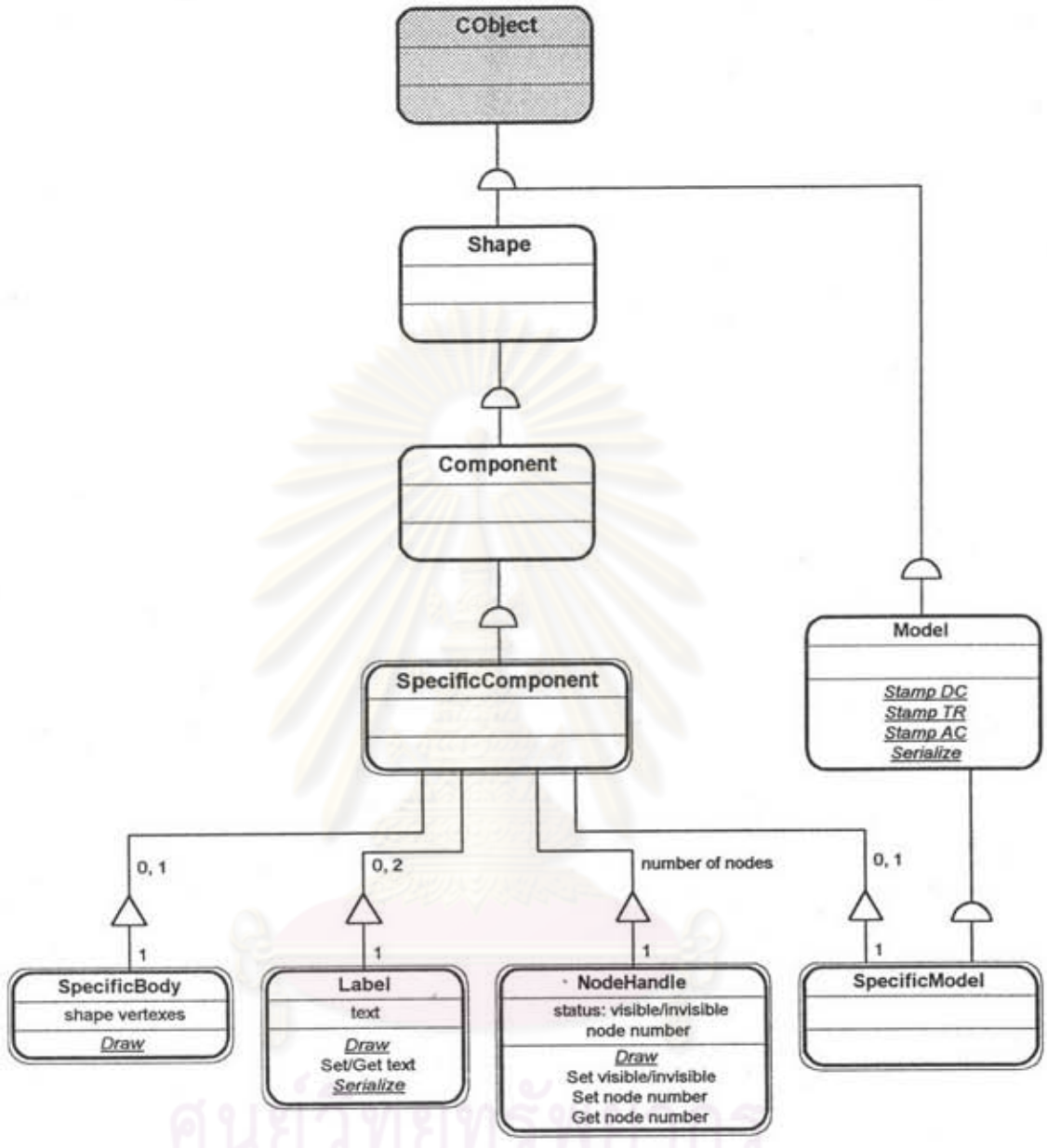
รูปที่ 3.9 แสดงตัวอย่างรูปของ Component ที่ปรากฏในช่องหน้าต่าง Circuit View เมื่อผู้ใช้ลากเอา Component จากช่องหน้าต่าง Component View มาวางในช่องหน้าต่าง Circuit View จะเห็นว่า Component มีวัตถุต่าง ๆ ประกอบกันเป็นรูปร่างทั่วไปดังรูปที่ 3.10 และวัตถุที่ประกอบกันเป็น Component นี้ได้แก่ Label, SpecificBody, NodeHandle นอกจากนั้นยังมีวัตถุที่สำคัญที่มีส่วนในการจำลองการทำงาน นั่นคือ Model ดังรูปที่ 3.11 และจะกล่าวเป็นข้อ ๆ ดังนี้



รูปที่ 3.9 ตัวอย่างรูปของ Component ที่ปรากฏในช่องหน้าต่าง Circuit View



รูปที่ 3.10 รูปร่างทั่วไปของรูป Component



รูปที่ 3.11 ลำดับชั้นของ Circuit, Component และส่วนประกอบของ Component

3.4.1 Label

ทำหน้าที่แสดงข้อความที่อธิบายถึง Component แต่ละตัว ในส่วนของ Reference Name และ Annotation

3.4.2 SpecificBody

ทำหน้าที่แสดงสัญลักษณ์ของ Component

3.4.3 NodeHandle

เปรียบเสมือนขั้วของ Component ชนิดต่าง ๆ ในวงจร ทำหน้าที่ช่วยต่อ Component ต่าง ๆ เข้าด้วยกัน เมื่อผู้ใช้ใช้เมาส์คลิ๊กที่ Node Handle ของ Component หนึ่งแล้วลากไปต่อยัง Node Handle ของอีก Component หนึ่ง CircuitView จะสร้างเส้นลวด (Wire) เชื่อมระหว่าง Node Handle ของ Component ทั้งสอง

นอกจากหน้าที่ดังกล่าวแล้ว NodeHandle มีคุณสมบัติ "node number" ที่เก็บหมายเลขประจำปมของส่วนประกอบของผังวงจรไฟฟ้า ซึ่งใช้ในการจำลองการทำงาน

3.4.4 Model

เป็นแบบจำลองของอุปกรณ์อิเล็กทรอนิกส์ กล่าวถึงอีกครั้งในบทต่อไป

3.4.5 SpecificModel

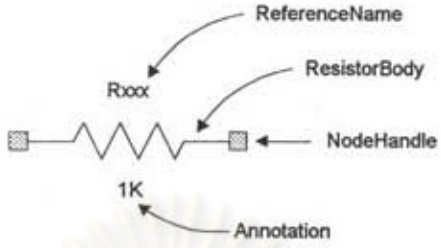
แสดงไว้ในแบบจำลองเพื่อแสดงว่า SpecificModel นี้จะถูกแทนด้วยแบบจำลองของคลาสอุปกรณ์อิเล็กทรอนิกส์ SpecificComponent

3.5 คลาสของ Component ชนิดต่าง ๆ

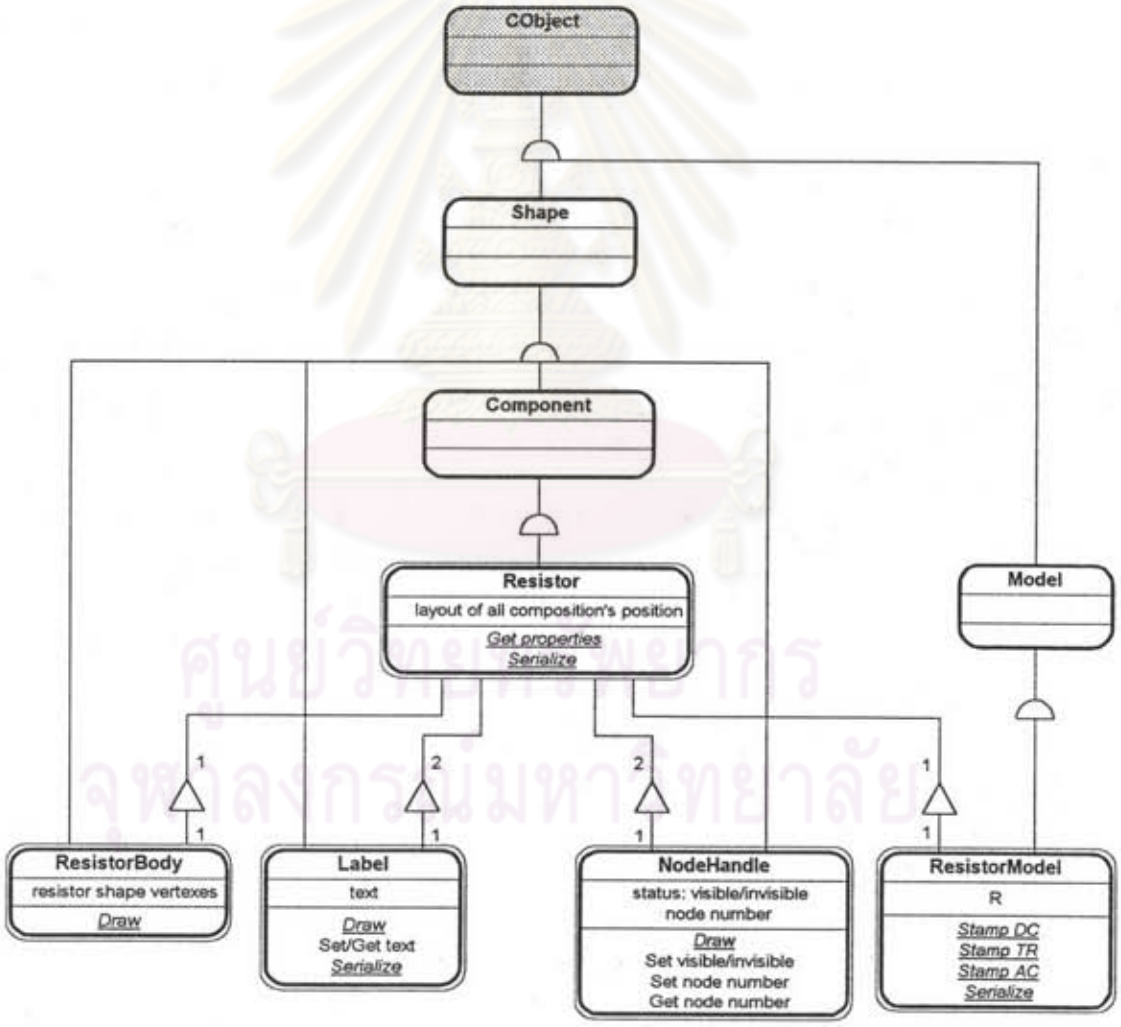
ในหัวข้อนี้จะกล่าวถึงคลาสที่สืบทอดจากคลาส Component ซึ่งจะแยกกล่าวเป็นข้อ ๆ โดยในแต่ละข้อจะแสดงรูปของ Component ชนิดนั้นและลำดับชั้นของคลาสนั้น ในลำดับชั้นของคลาส Component แต่ละตัวจะแสดงส่วนประกอบของ Component ที่ทุกตัวเพื่อความสมบูรณ์ สำหรับคลาส Component แต่ละชนิด ส่วนประกอบของคลาสที่มีส่วนเกี่ยวข้องกับการจำลองการทำงานและการแสดงผลการจำลองการทำงานจะกล่าวในบทที่ 4

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

3.5.1 Resistor (รูปที่ 3.12 และ 3.13)

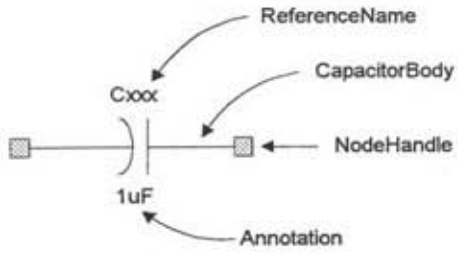


รูปที่ 3.12 รูปของส่วนประกอบของ Resistor

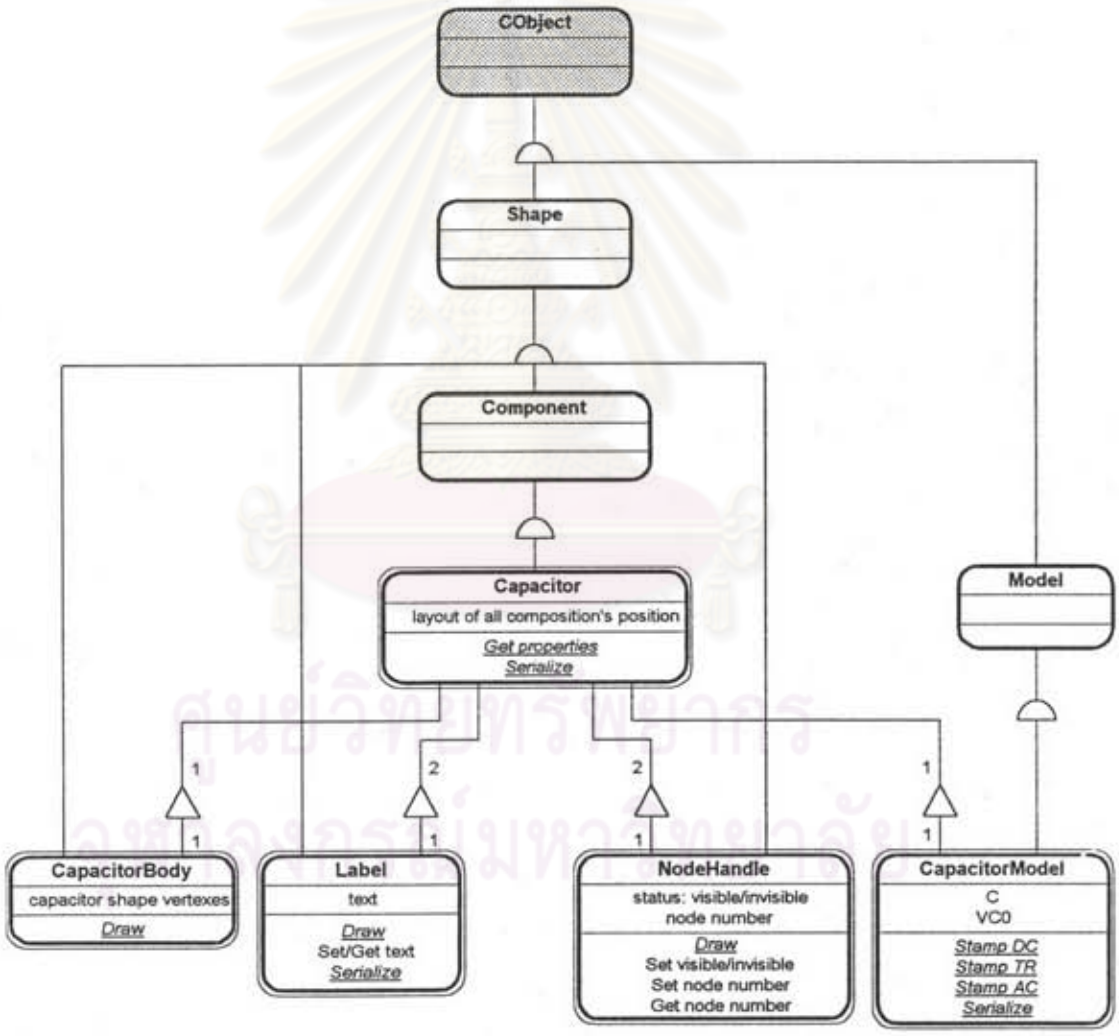


รูปที่ 3.13 คลาส Resistor

3.5.2 Capacitor (รูปที่ 3.14 และ 3.15)



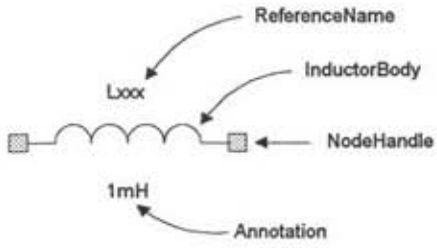
รูปที่ 3.14 รูปส่วนประกอบของ Capacitor



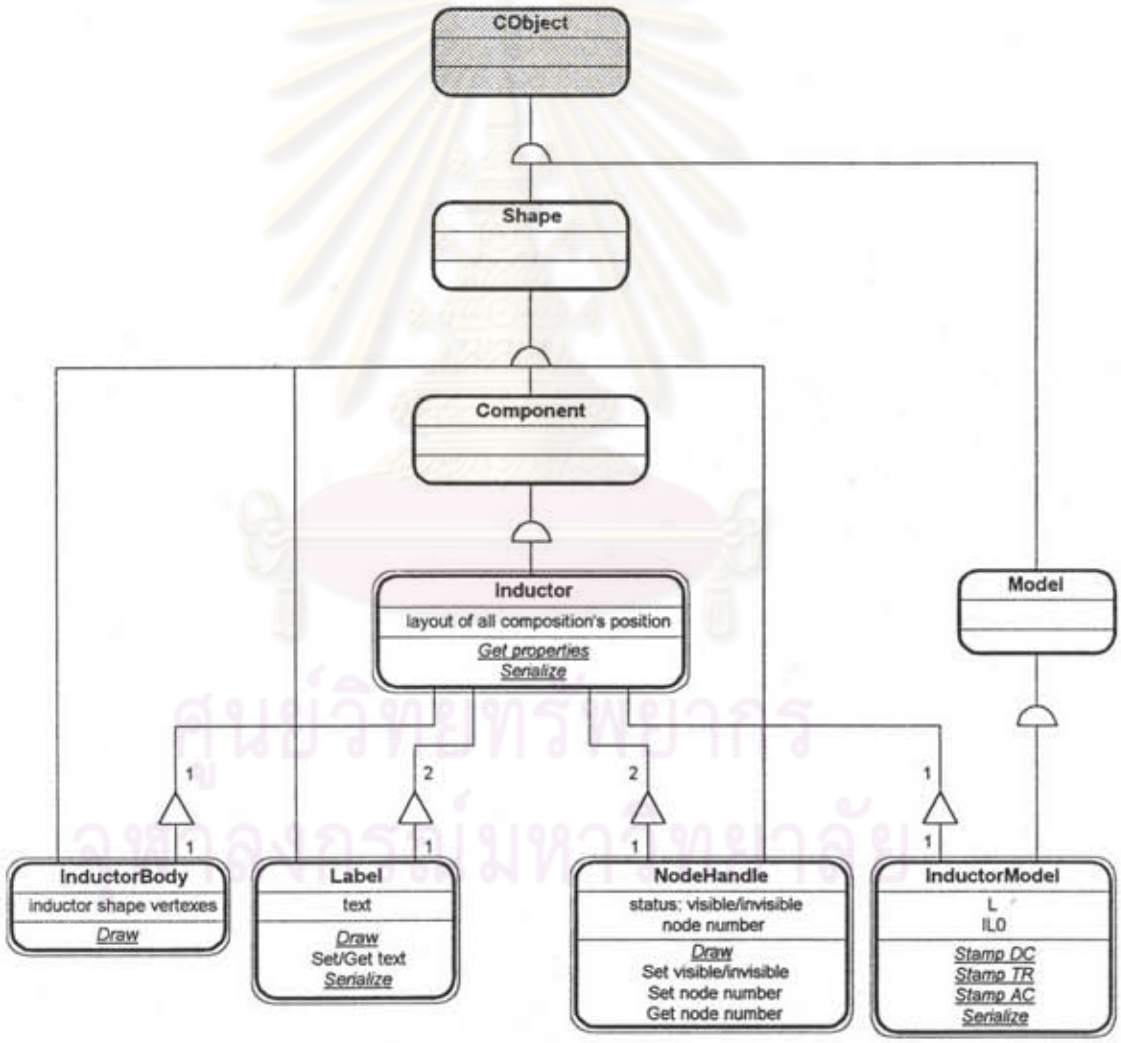
รูปที่ 3.15 คลาส Capacitor

I 17012911

3.5.3 Inductor (รูปที่ 3.16 และ 3.17)

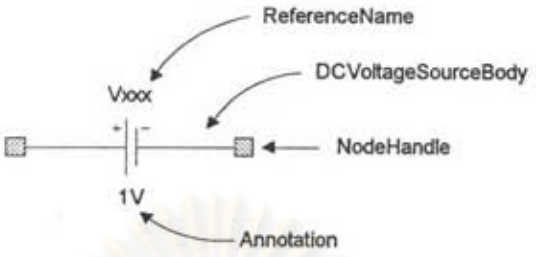


รูปที่ 3.16 รูปของส่วนประกอบของ Inductor

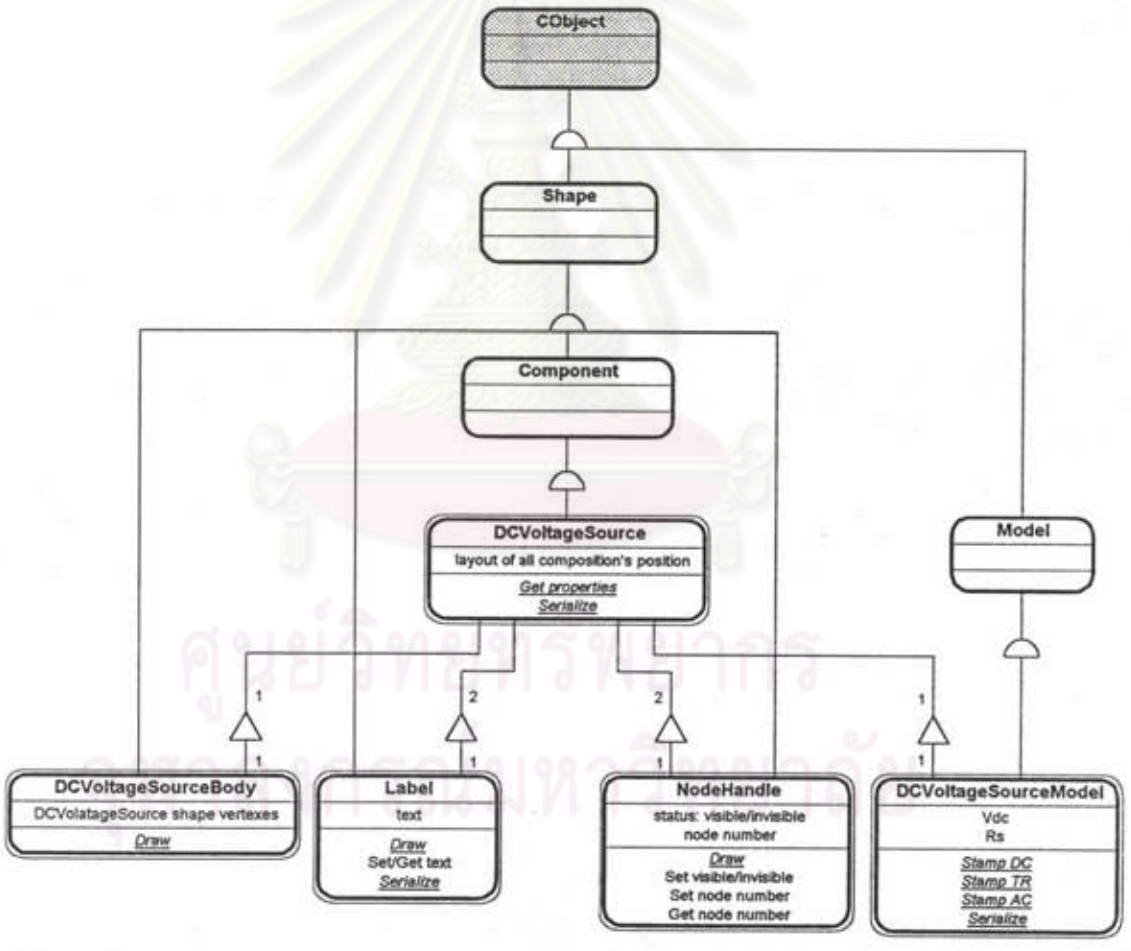


รูปที่ 3.17 คลาส Inductor

3.5.4 DCVoltageSource (รูปที่ 3.18 และ 3.19)

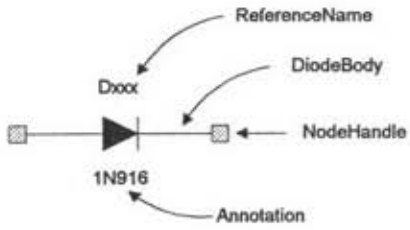


รูปที่ 3.18 รูปส่วนประกอบของ DCVoltageSource

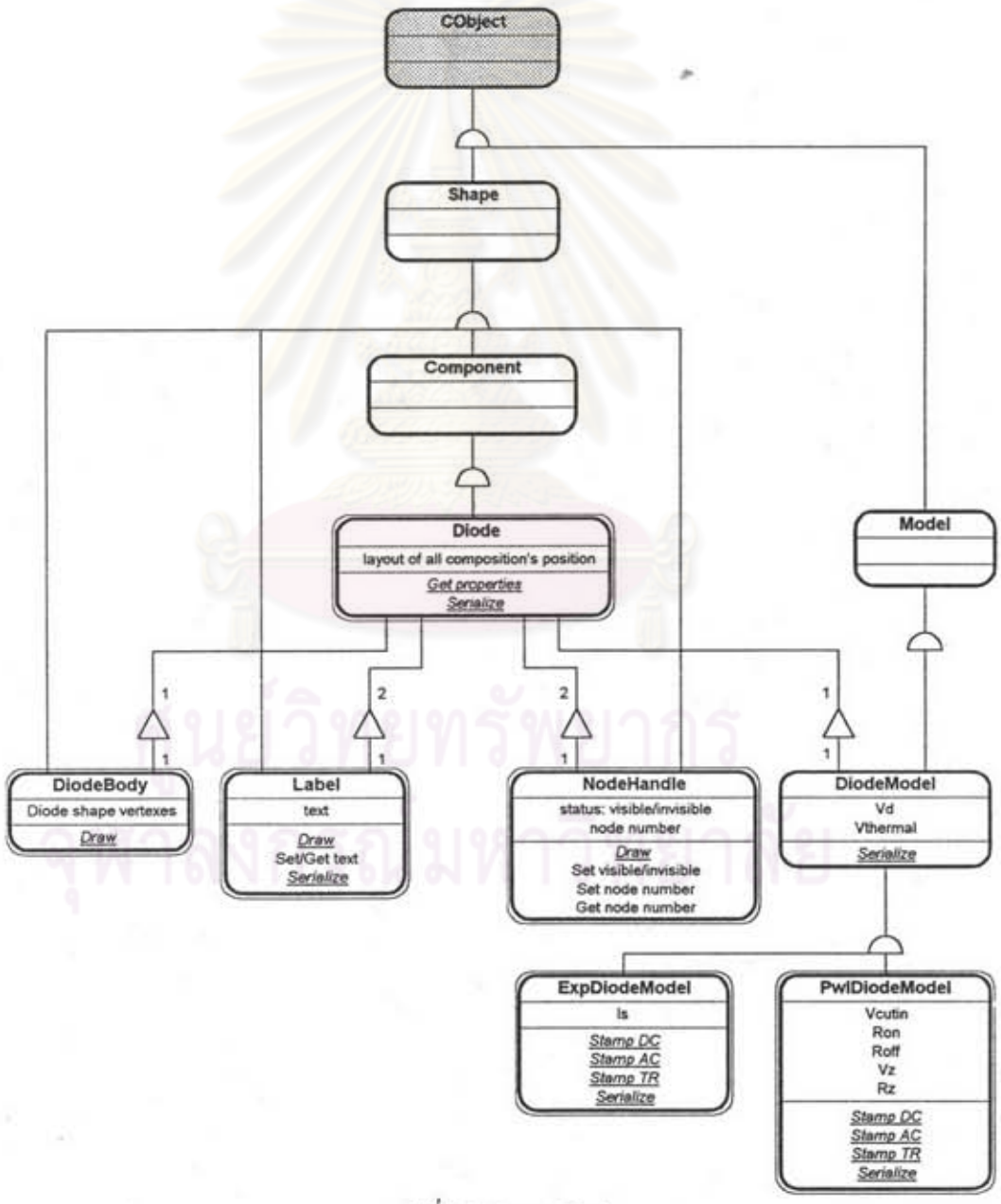


รูปที่ 3.19 คลาส DCVoltageSource

3.5.5 Diode (รูปที่ 3.20 และ 3.21)



รูปที่ 3.20 รูปของส่วนประกอบของ Diode



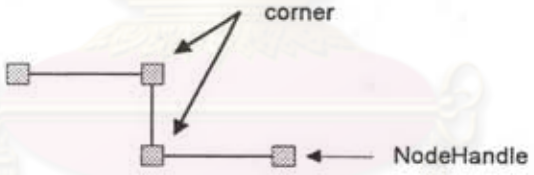
รูปที่ 3.21 คลาส Diode

3.5.6 Wire (รูปที่ 3.22 และ 3.23)

ซึ่ง Wire นี้ประกอบด้วย NodeHandle ตั้งแต่ 2 ขึ้นไป มีจำนวนเท่ากับมุม (Corner) ของเส้นดังรูปที่ 3.22 ใน Wire นี้ไม่มี WireBody เพราะว่า Wire ที่ผู้ใช้ลากจะมีมุมไม่แน่นอนตายตัว ดังนั้นฟังก์ชัน Draw ในการวาดรูป Wire จะใช้วิธีลากเส้นเชื่อมระหว่างตำแหน่งของ NodeHandle ทั้งหมดของ Wire

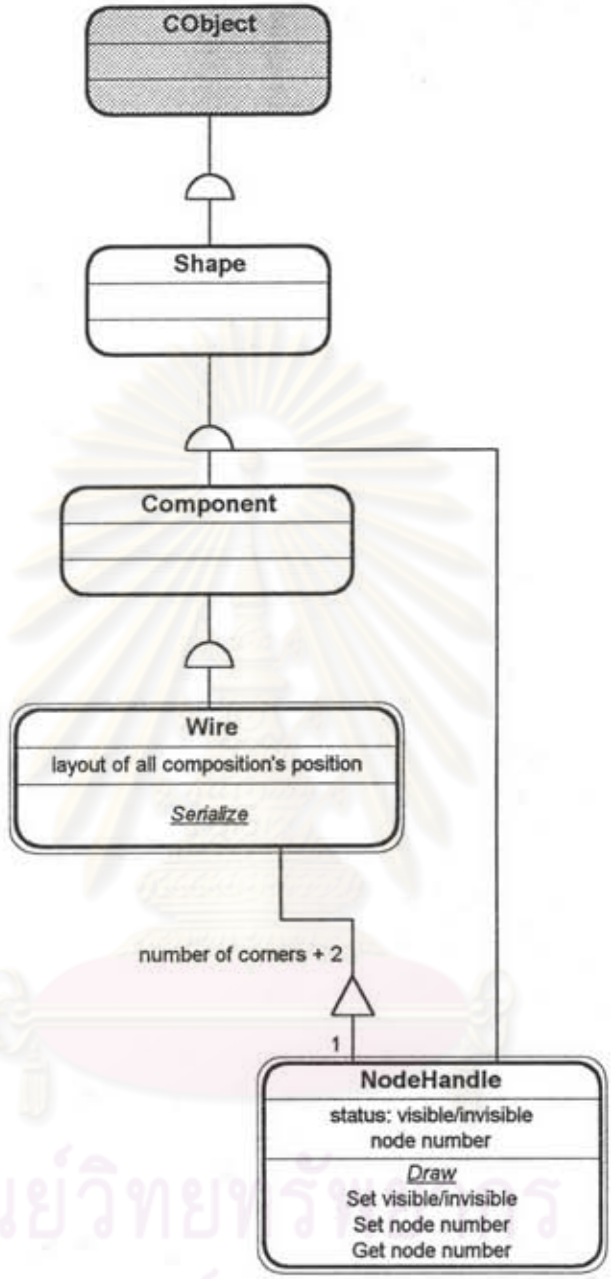
แต่ในต้นแบบโปรแกรมจำลองการทำงานวงจรไฟฟ้านี้ ได้กำหนดจำนวนของ NodeHandle ไว้เท่ากับ 3 เท่านั้น แต่ว่าการเก็บ NodeHandle ของ Component ทุกชนิดที่มี NodeHandle จะเก็บเป็นรายการแบบเชื่อมโยง ดังนั้นจึงสามารถที่จะเขียนวิธีในการลาก Wire ใหม่ให้มีการสร้าง NodeHandle แทรกในรายการแบบเชื่อมโยงของ NodeHandle ได้เมื่อจุดปลายของ Wire มีการเคลื่อนที่แล้วทำให้เกิดเป็นมุมขึ้น

ในการจำลองการทำงานวงจรไฟฟ้า Wire ไม่ได้เข้าไปมีส่วนร่วมในการคำนวณเลย เพียงแต่ช่วยในการกำหนดค่าหมายเลขประจำปมของแต่ละขั้วของ Component ชนิดอื่น เพราะว่าทุก NodeHandle ของ Wire จะมีคุณสมบัติ node number มีค่าเท่ากัน เมื่อกำหนดค่าหมายเลขประจำปม (Node Number) ให้กับ Wire



รูปที่ 3.22 รูปของส่วนประกอบของ Wire

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

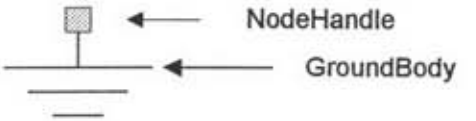


รูปที่ 3.23 คลาส Wire

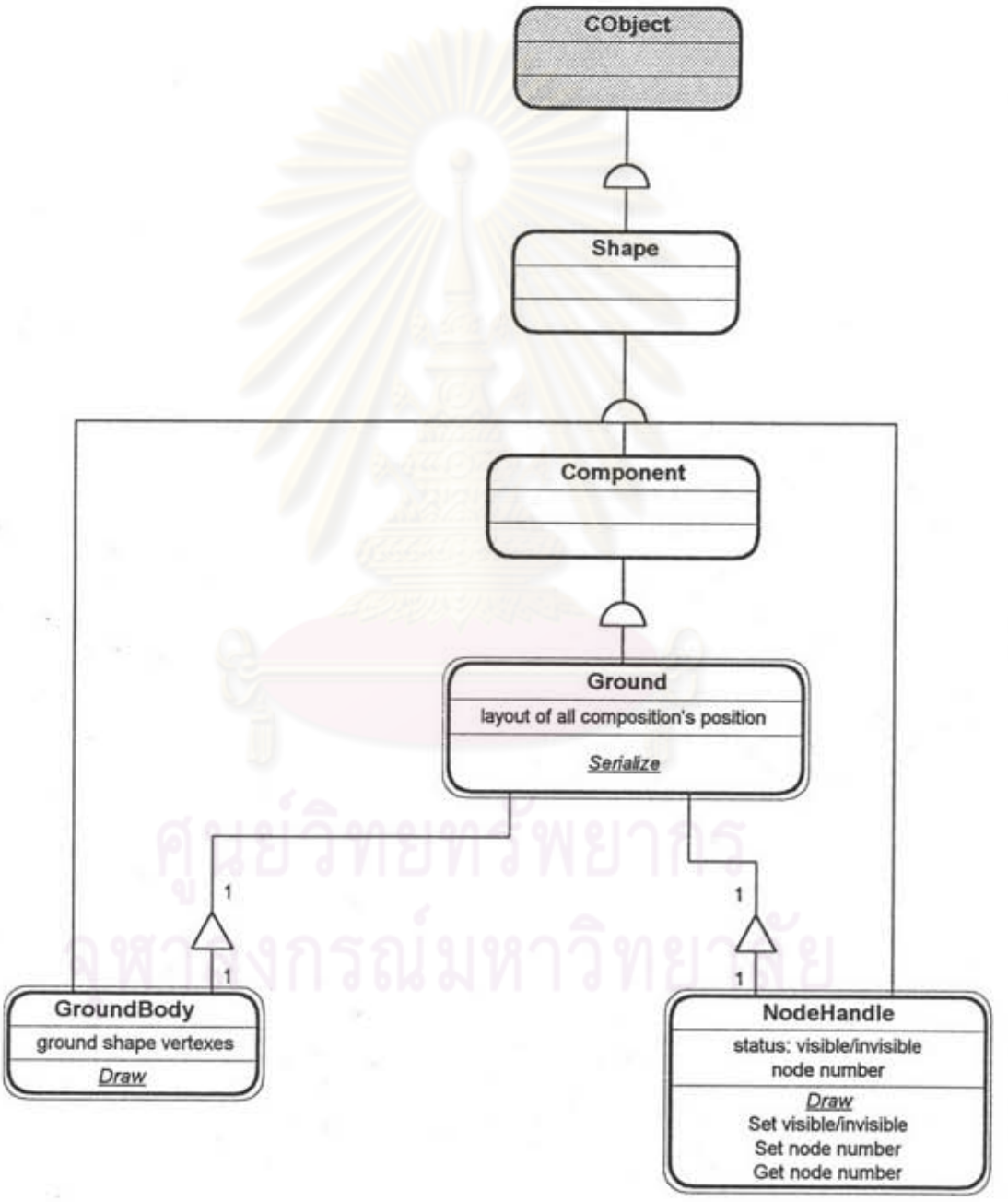
3.5.7 Ground (รูปที่ 3.24 และ 3.25)

Ground ไม่ได้มีส่วนร่วมในการคำนวณในการจำลองการทำงานวงจรไฟฟ้า เพียงช่วยในการกำหนดหมายเลขประจำปม โดย NodeHandle ของ Component อื่นที่ต่ออยู่กับ NodeHandle ของ Ground นี้จะถูกกำหนดให้มีค่าหมายเลขประจำปม (คุณสมบัติ node number) เป็น 0

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 3.24 รูปของส่วนประกอบของ Ground



รูปที่ 3.25 คลาส Ground

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

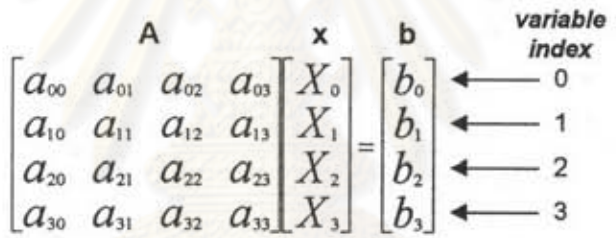
3.5.8 Probe (รูปที่ 3.27, 3.28, 3.29 และ 3.30)

ทำหน้าที่บอกให้กับ OutputView รู้ว่าจะต้องนำเอาค่าของตัวแปรวงจรไฟฟ้าในสมการวงจรไฟฟ้าที่จะนำมาแสดงที่หน้าต่าง Output View ซึ่ง Probe จะมีคุณสมบัติ variable index ซึ่งเก็บดัชนีที่ชี้ไปตัวแปรในเวกเตอร์ x ดังตัวอย่างในรูปที่ 3.26 และกำหนดวิธีให้ OutputView ติดต่อกับ VoltageProbe และ CurrentProbe

3.5.8.1 VoltageProbe (รูปที่ 3.27 และ 3.28)

3.5.8.2 CurrentProbe (รูปที่ 3.29 และ 3.30)

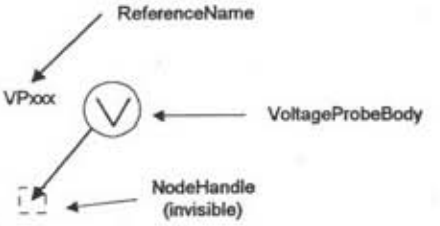
CurrentProbe บอกให้ OutputView รู้ว่าค่ากระแสใดที่จะนำมาแสดงได้เฉพาะกระแสที่เป็นตัวแปรวงจรไฟฟ้าในสมการวงจรไฟฟ้า นั่นคือกระแสของแหล่งกำเนิดแรงดัน และกระแสของตัวเหนี่ยวนำ เท่านั้น



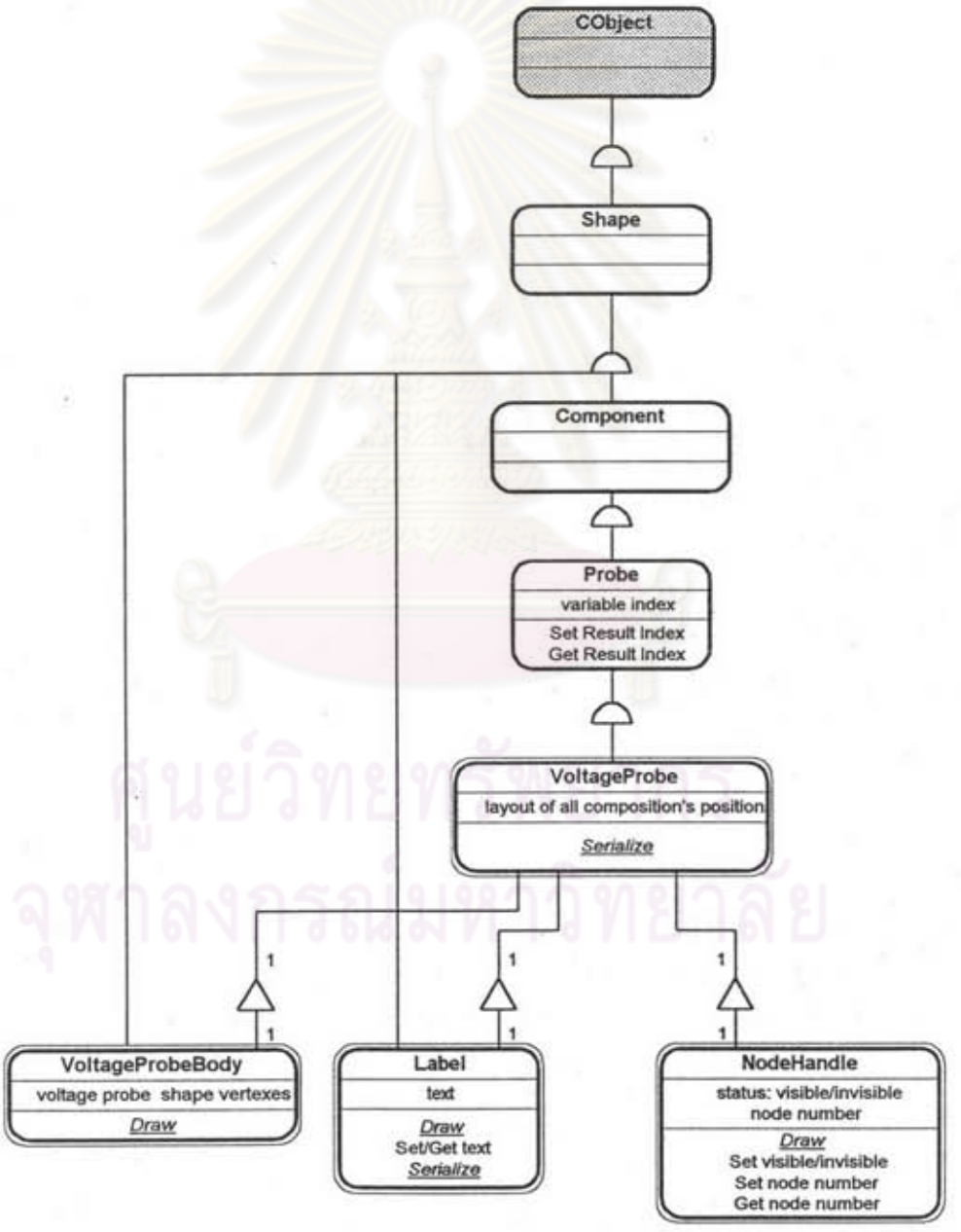
รูปที่ 3.26 variable index ที่ชี้ในสมการเมตริกซ์ของวงจรไฟฟ้า

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

3.5.8.1 VoltageProbe (รูปที่ 3.27 และ 3.28)



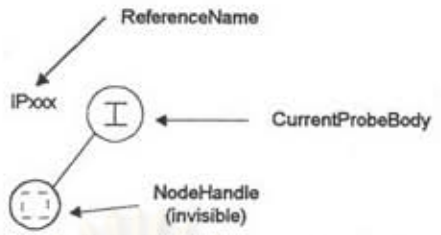
รูปที่ 3.27 รูปของส่วนประกอบของ VoltageProbe



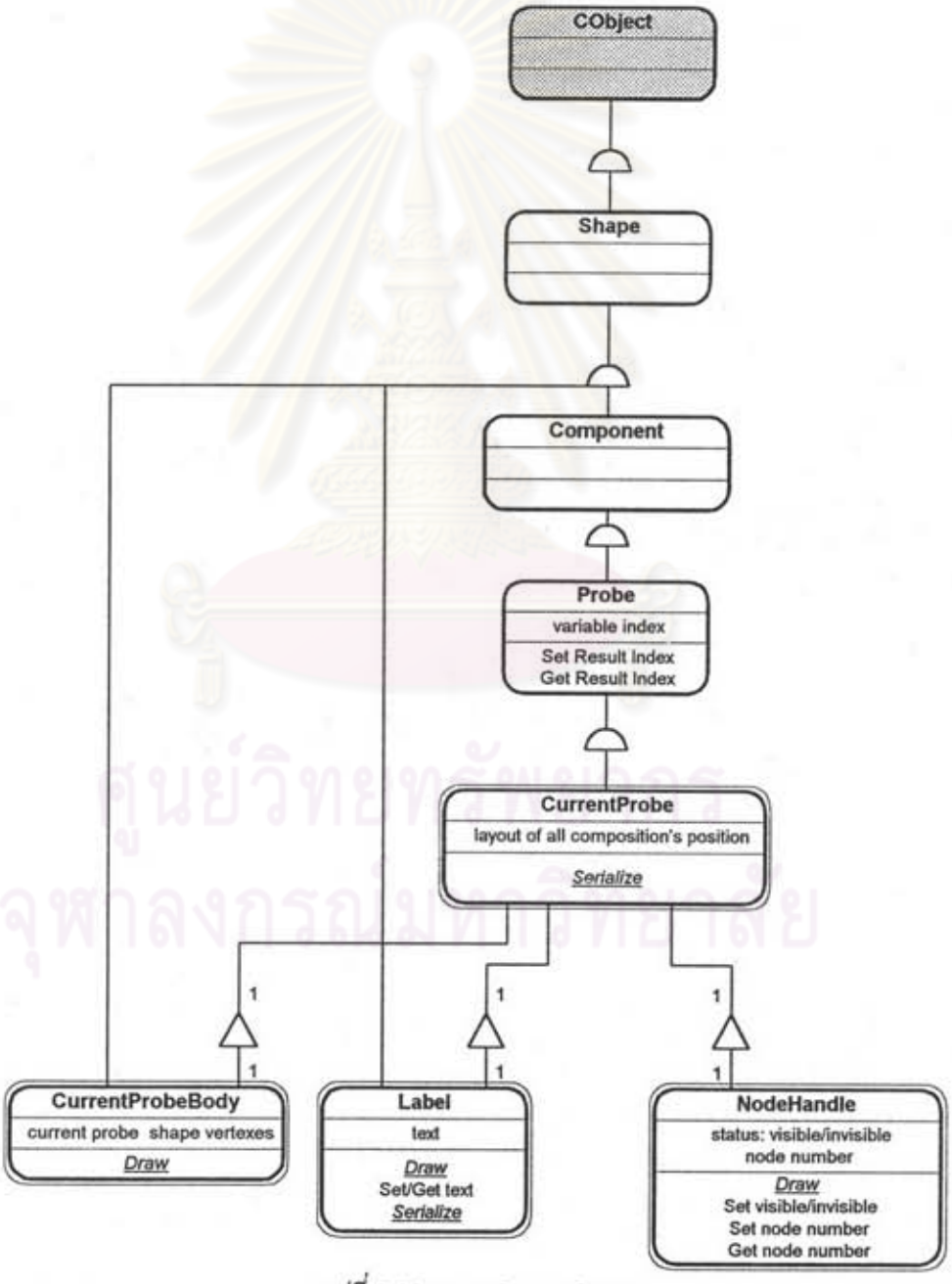
รูปที่ 3.28 คลาส VoltageProbe

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

3.5.8.2 CurrentProbe (รูปที่ 3.29 และ 3.30)



รูปที่ 3.29 รูปของส่วนประกอบของ CurrentProbe



รูปที่ 3.30 คลาส CurrentProbe

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

3.6 พฤติกรรมของวัตถุ

ในหัวข้อนี้จะกล่าวและแสดงถึงพฤติกรรมของวัตถุที่สำคัญ ๆ ของโปรแกรม ออกมาในรูปของแผนผัง Object Life-History ของวัตถุ

3.6.1 พฤติกรรมของ ComponentView

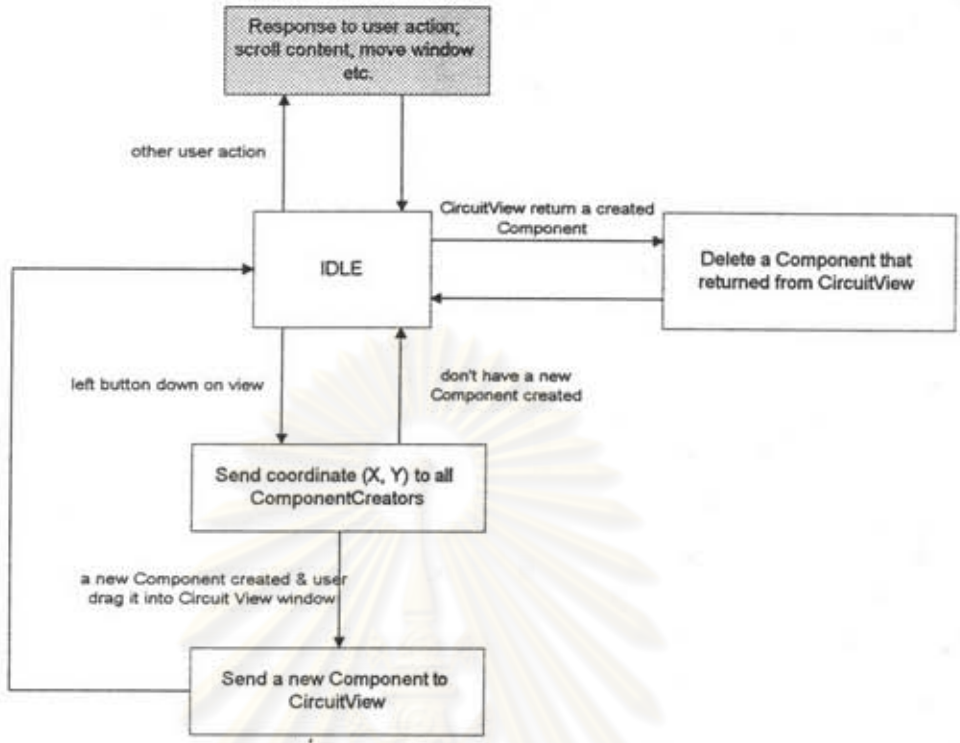
รูปที่ 3.31 แสดงพฤติกรรมของ ComponentView ที่เกิดขึ้นในขณะที่ผู้ใช้คลิ๊กในหน้าต่าง Component View วัตถุ ComponentView จะส่งค่าตำแหน่ง (x, y) ของตัวชี้ไปให้ ComponentCreator ที่อยู่บนหน้าต่าง Component View ถ้าตำแหน่ง (x, y) นั้นอยู่บนตัว ComponentCreator ตัวใด ComponentCreator ตัวนั้นจะสร้าง Component ให้กับ ComponentView เพื่อให้ ComponentView ส่ง Component นั้นไปให้ CircuitView จัดการกับ Component นั้นต่อไป เมื่อ CircuitView รับ Component ไป ถ้าผู้ใช้วาง Component นอกหน้าต่าง Circuit View CircuitView จะส่ง Component นั้นกลับคืนให้กับ ComponentView ทำลาย

พฤติกรรมของ ComponentView แสดงไว้ในรูปที่ 3.31 ในกรอบที่เป็นสีเทา แสดงถึงการตอบสนองต่อคำสั่งของผู้ใช้โดยทั่ว ๆ ไป เช่น การเลื่อนข้อมูลในหน้าต่าง Component View , การสั่งให้ ComponentCreator แสดงรูปของ Component ที่ให้ผู้ใช้เลือกนำไปวาดผังวงจร เป็นต้น

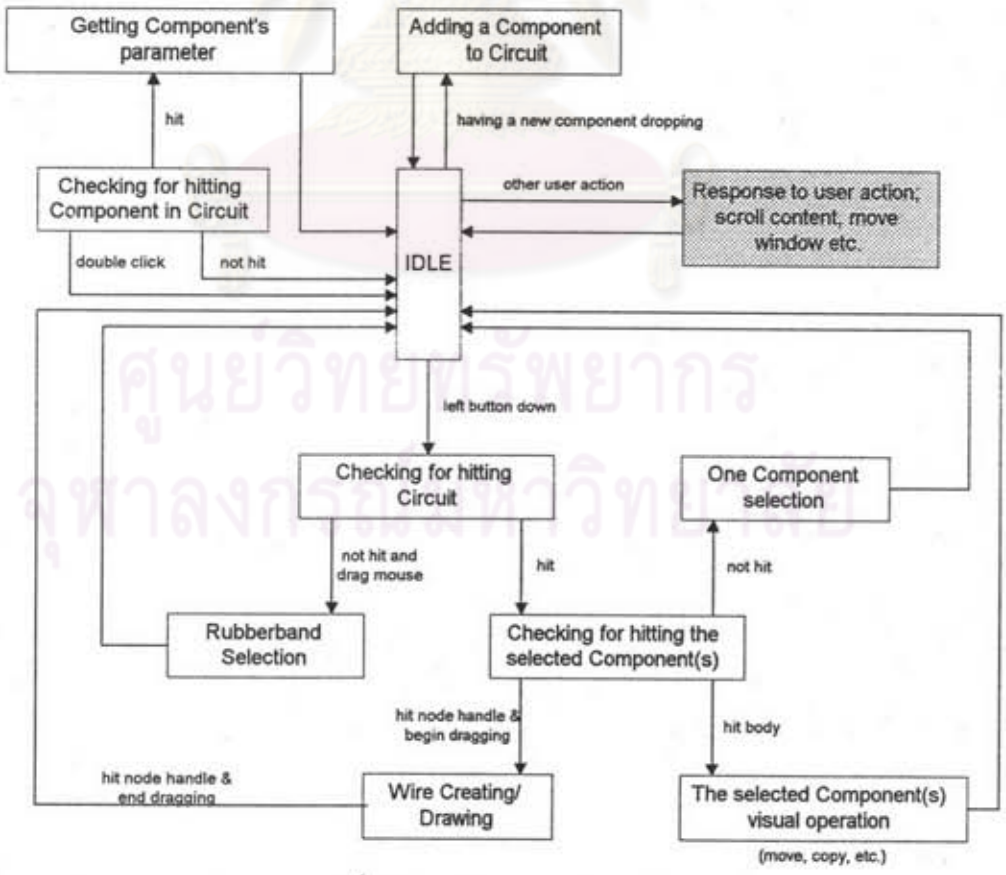
3.6.2 พฤติกรรมของ CircuitView

ในรูปที่ 3.32 แสดงพฤติกรรมที่เกิดขึ้น

- เมื่อผู้ใช้ลาก Component ที่ได้จาก ComponentView มาวางในหน้าต่าง Circuit View ซึ่ง CircuitView จะนำเอา Component นั้นเพิ่มใน Circuit และพฤติกรรมที่เกิดขึ้นเมื่อผู้ใช้ใช้เมาส์คลิ๊กในหน้าต่าง Circuit View
- เมื่อผู้ใช้เลือก Component เป็นตัวโดยการใช้เมาส์คลิ๊กที่ตัว Component ที่ต้องการ หรือเป็นกลุ่ม (Rubberband Selection) โดยใช้เมาส์ลากกรอบสี่เหลี่ยมล้อมรอบกลุ่มของ Component ที่ต้องการ
- เมื่อผู้ใช้ลากเส้นลวด (Wire) ต่อ Component เข้าด้วยกัน



รูปที่ 3.31 พฤติกรรมของ ComponentView



รูปที่ 3.32 พฤติกรรมของ CircuitView

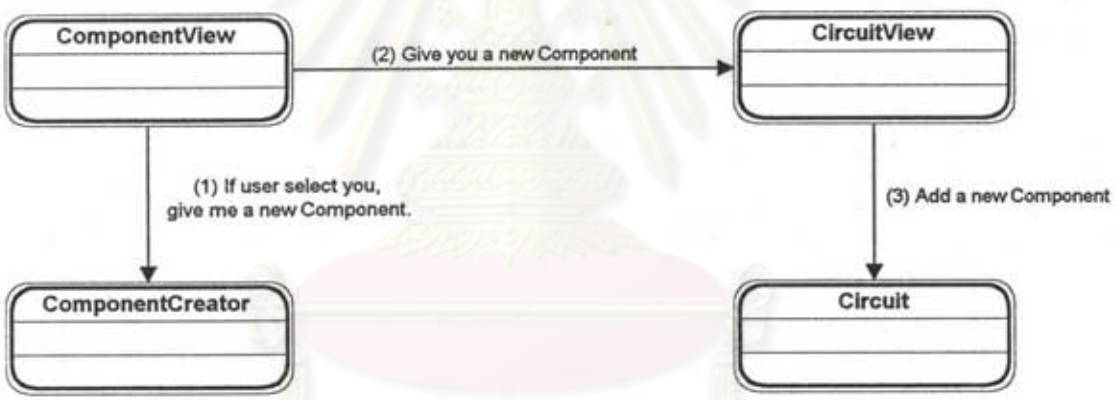
ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

3.7 การติดต่อส่งข่าวสารระหว่างวัตถุ

ในหัวนี้จะกล่าวถึงการติดต่อกันระหว่างวัตถุ ในส่วนที่สำคัญ ๆ ของการทำงานของโปรแกรมจำลองการทำงานวงจรไฟฟ้า

3.7.1 การติดต่อระหว่างวัตถุเมื่อผู้ใช้เลือกรูปอุปกรณ์ไปวางในหน้าต่าง Circuit View

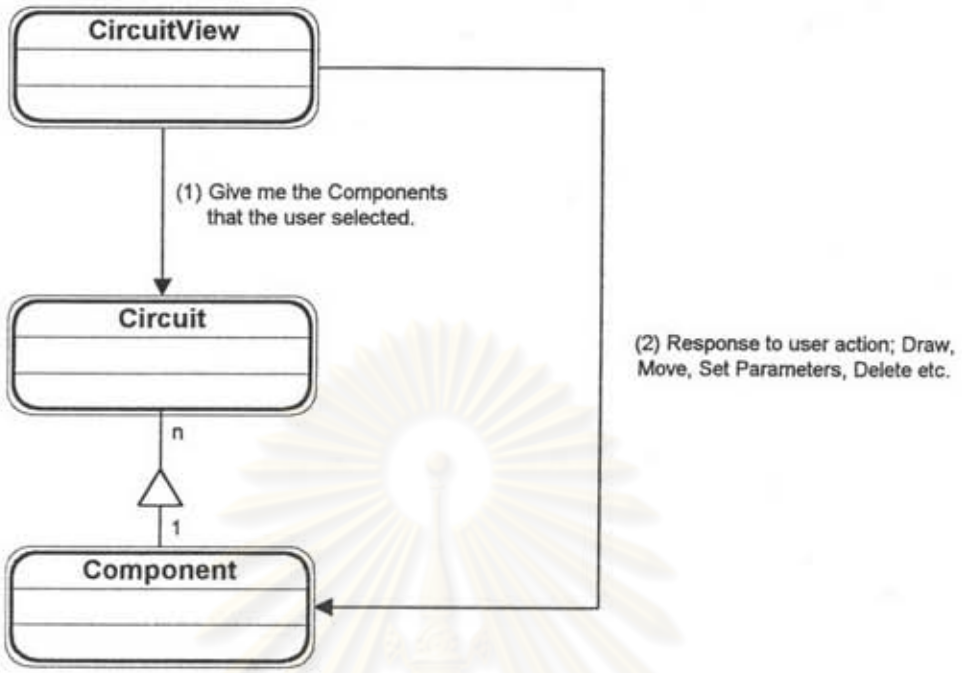
รูปที่ 3.33 แสดงการติดต่อระหว่างวัตถุเมื่อผู้ใช้คลิ๊ก (Click) บนหน้าต่าง Component View ตัว ComponentView จะถาม ComponentCreator แต่ละตัวที่เป็นส่วนประกอบของ ComponentView ว่าตำแหน่ง (x, y) ของตัวชี้ของเมาส์อยู่บน ComponentCreator หรือไม่ ถ้าอยู่ ComponentCreator จะสร้าง Component ให้กับ ComponentView เพื่อส่ง Component นั้นให้กับ CircuitView จัดการกับ Component ต่อไป



รูปที่ 3.33 การติดต่อกันระหว่างวัตถุเมื่อผู้ใช้ใช้เมาส์ลากรูปอุปกรณ์ไปวาดผังวงจร

3.7.2 การติดต่อระหว่างวัตถุเมื่อผู้ใช้จัดการกับ Component ใน Circuit

รูปที่ 3.34 แสดงการจัดการกับกลุ่มของ Component ที่ผู้ใช้เลือกจาก Circuit โดยนำ CircuitView จะขอ Component ที่ถูกเลือกมาจัดการตามคำสั่งที่ผู้ใช้เลือกจากเมนู หรือทูลบาร์ เป็นตัว ๆ ไป



รูปที่ 3.34 การติดต่อกันระหว่างวัตถุเมื่อผู้ใช้จัดการกับ Component ที่เลือก

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย