

บทที่ 10

ระบบไอ/โอและระบบไฟล์

(I/O System and Files System)

การทำงานของคอมพิวเตอร์โดยรวมแล้วแบ่งหน้าที่การทำงานอย่างชัดเจน ในส่วนของการรับข้อมูล การประมวลผลและการแสดงผลลัพธ์ มีการควบคุมให้ทำงานร่วมกันอย่างต่อเนื่อง ซึ่งการรับข้อมูลเข้าสู่ระบบคอมพิวเตอร์ผ่านทางอุปกรณ์นำเข้า (Input Devices) ทำการประมวลผลที่ซีพียู และแสดงผลที่อุปกรณ์แสดงผล (Output Devices) ทั้งอุปกรณ์นำเข้าและอุปกรณ์แสดงผล ในการทำงานของระบบคอมพิวเตอร์นั้นมีหลากหลาย ระบบปฏิบัติการจะต้องควบคุมการทำงานให้ทุก ๆ อุปกรณ์ทำงานอย่างเป็นระบบและสอดคล้องกับซีพียูจึงเรียกระบบการทำงานในส่วนนี้ว่าระบบการนำเข้าและการส่งออก (Input / Output System : I/O System) หรือโดยทั่วไปเรียกระบบไอโอ สำหรับข้อมูลที่จัดเก็บในระบบคอมพิวเตอร์จะถูกจัดเก็บเป็นไฟล์ ซึ่งการจัดการไฟล์เป็นบริการอย่างหนึ่งของระบบปฏิบัติการ โดยที่ไฟล์เป็นที่เก็บข้อมูลที่เกี่ยวข้องและสัมพันธ์กันตามนิยามที่ผู้สร้างไฟล์กำหนดไว้ ไฟล์จะถูกระบบปฏิบัติการควบคุมและจัดการบนสื่อจัดเก็บที่จะกล่าวถึงต่อไปเรียกว่า ระบบไฟล์ (File System) รวมถึงการจัดการไดเรกทอรีของไฟล์ (File Directories) ระบบปฏิบัติการดำเนินการที่ติดต่อกับระบบไฟล์ การควบคุมไฟล์ขั้นพื้นฐานและในไดเรกทอรีของไฟล์ รวมทั้งการป้องกันการเข้าถึงไฟล์ข้อมูล และในส่วนท้ายบทได้กล่าวถึงพื้นฐานการป้องกันและการรักษาความปลอดภัย

การดำเนินงานของอุปกรณ์ ไอ/โอ

การรับส่งข้อมูลระหว่างหน่วยความจำหลักและอุปกรณ์ไอ/โอ มีการทำงาน 3 รูปแบบดังนี้

1. โหมดโปรแกรมไอ/โอ (Programmed I/O Mode) ซีพียู ต้องประมวลผล คำสั่งเพื่อเริ่มทำงานของไอ/โอ การทำงานในรูปแบบนี้จะช้า
2. โหมดการขัดจังหวะ (Interrupt Mode) เป็นการทำงานในช่วงการส่งผ่านข้อมูลในแต่ละไบต์ ซีพียู จะว่างงานในช่วงนี้
3. เข้าใช้หน่วยความจำโดยตรง (Direct Memory Access : DMA) เป็นการทำงานงานของ ไอ/โอ ที่ทำงานกับหน่วยความจำโดยซีพียูเพียงแต่สั่งการผ่านตัวควบคุมอุปกรณ์เท่านั้น การอ่านหรือบันทึกบนหน่วยความจำจะทำการโดยตัวควบคุมของไอ/โอ (I/O Channel) ผลดีก็คือ ซีพียูไม่ต้องคอยให้ไอ/โอ ทำงานจนเสร็จ ซีพียูสามารถงานอื่นไปได้พร้อมกับขณะที่ ไอ/โอ ทำงาน เมื่อไอ/โอ ทำงานเสร็จตัวควบคุมอุปกรณ์จะส่งสัญญาณขัดจังหวะของไอ/โอ (I/O Interrupt) ไปให้ ซีพียู

ประมวลผลโปรแกรมของผู้ควบคุมในระบบปฏิบัติการให้รู้ว่าไอ/โอทำงานเสร็จแล้ว ซึ่งจะได้อัด
ตารางเวลาการทำงานใหม่ต่อไป ทำให้ระบบได้เพิ่มประสิทธิภาพการทำงานของซีพียู

การขัดจังหวะ

การทำงานของระบบคอมพิวเตอร์ใช้การขัดจังหวะการทำงานของซีพียูชั่วคราว เพื่อเป็น
การส่งสัญญาณบอก ผู้ควบคุมให้จัดเงื่อนไขการทำงานภายในระบบใหม่ ทั้งนี้เราใช้ประโยชน์จาก
ข้อความสถานะของโปรแกรม (Program Status Words : PSW) โดยเมื่อมีการการขัดจังหวะ
เกิดขึ้น เราจะอ่านข้อมูลใหม่เข้าไปที่ PSW ทั้งนี้ข้อมูลเก่าบน PSW จะถูกฝากไว้ชั่วคราวใน
หน่วยความจำ เมื่อซีพียูประมวลผล PSW ตัวใหม่นี้จะเกิดการเปลี่ยนขั้นตอนการทำงาน และเมื่อ
การขัดจังหวะเสร็จสิ้น PSW ตัวเก่าจะถูกอ่านกลับคืนมา เพื่อให้ซีพียูกลับไปทำงานเดิมที่ค้างต่อไป
ได้ หากพบว่าเกิดการขัดจังหวะขึ้นพร้อมกันหลาย ๆ งาน ระบบจะมีกลไกการขัดจังหวะในการ
เลือกตัวที่มีความสำคัญ (Priority) สูงสุดเข้าไปทำงาน

ในการทำงานของระบบคอมพิวเตอร์ ต้องมีการขัดจังหวะการทำงานเพื่อให้การทำงานของ
เครื่องคอมพิวเตอร์งานหลายๆงานได้อย่างต่อเนื่อง เช่น ขณะเครื่องกำลังอ่านข้อมูลจากดิสก์
ผู้ใช้เครื่องยังสามารถป้อนข้อมูลผ่านคีย์บอร์ดและเครื่องก็อ่านข้อมูลจากการคีย์ในขณะนั้นได้ โดย
ความรู้สึกเหมือนเครื่องทำงานพร้อม ๆ กัน ได้หลาย ๆ งานในเวลาเดียวกัน โดยมีการขัดจังหวะ
การทำงานให้ซีพียูให้แบ่งเวลาทำการประมวลผลงานต่าง ๆ สลับกันไปตามความจำเป็นเร่งด่วน
ของแต่ละงาน

สัญญาณขัดจังหวะ

ระบบคอมพิวเตอร์มีการใช้การขัดจังหวะเพื่อให้ระบบทำงานประสานกันอยู่เสมือนทำงาน
พร้อมกัน เช่น ในเครื่องไมโครคอมพิวเตอร์ได้จัดการ การขัดจังหวะจากที่ต่าง ๆ ไว้ เช่น จากวงจร
เวลาส่งสัญญาณการขัดจังหวะเข้ามาเพื่อให้ซีพียูปรับคานาฬิกาให้ถูกต้อง เมื่อเครื่องกำลังทำงาน
อื่นใดก็ตาม แต่ถ้ามีการป้อนข้อมูลจากคีย์บอร์ด จะมีการขัดจังหวะเพื่อให้มีการรับค่าจากคีย์บอร์ด
เข้าไปเก็บในบัฟเฟอร์ก่อน ในการจัดระดับการขัดจังหวะที่จะมาจากพอร์ตอินพุต-เอาต์พุต
ผู้ออกแบบจะต้องวางรูปแบบลำดับความสำคัญเอาไว้ เช่น นาฬิกาเป็นส่วนที่ที่สำคัญที่สุด เพราะ
นาฬิกาจะต้องเดินต่อเนื่องและถูกต้องเสมอ คีย์บอร์ดมีความสำคัญรองลงมา ระบบสื่อสารแบบ
อนุกรมประเภทดิสก์ไครฟ์ ฮาร์ดดิสก์ และเครื่องพิมพ์เป็นสิ่งที่มีความสำคัญรองลงมาอีกเช่นกัน
การจัดลำดับความสำคัญนี้ จะต้องเป็นไปตามหลักการที่กำหนด เช่นเมื่อมีสัญญาณขัดจังหวะเข้ามา
อุปกรณ์ที่มีความสำคัญสูงกว่า จะต้องได้รับการบริการก่อน ส่วนพวกที่มีระดับต่ำกว่าจะรออยู่
ก่อน เช่นเมื่อเครื่องกำลังบริการดิสก์ไครฟ์ ตามการขอจากการขัดจังหวะของดิสก์ไครฟ์ แต่ถ้า
คีย์บอร์ดขอมา ซีพียูจะไปบริการคีย์บอร์ดก่อนแล้วกลับมาทำดิสก์ไครฟ์ต่อ แต่ถ้ากำลังบริการส่วน

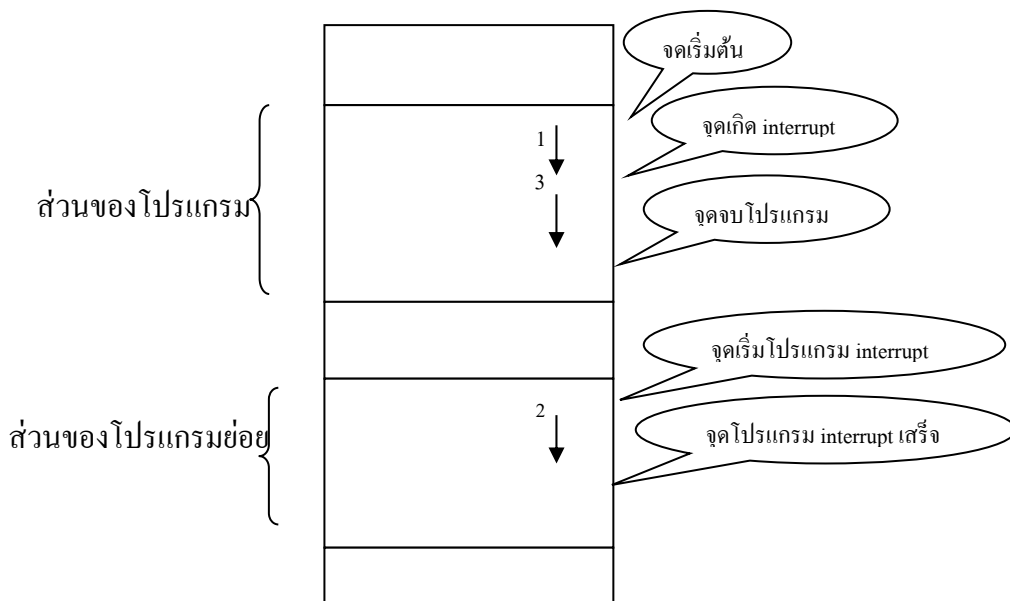
ที่มีลำดับความสำคัญสูงกว่า เช่น กำลังให้บริการปรับเวลาตามนาฬิกา แต่บังเอิญมีการกดคีย์บอร์ด ซีพียูจะทำการปรับปรุงเวลาให้เสร็จก่อน จึงค่อยให้บริการคีย์บอร์ด

การขัดจังหวะจึงทำให้ซีพียูกระโดดไปทำงานจากโปรแกรมบริการส่วนต่าง ๆ จนดูเหมือนทำงานพร้อมกันดังนั้น ระบบมัลติโปรแกรมมิ่งที่ใช้กับซีพียูตัวเดียวก็คือ ระบบที่ใช้การขัดจังหวะเพื่อแบ่งเวลา การขัดจังหวะ เป็นความสามารถพิเศษทางฮาร์ดแวร์ ที่จะทำให้ซีพียูหยุดการทำงานใด ๆ ไว้ชั่วคราวเพื่อปฏิบัติการโต้ตอบกับอีกเหตุการณ์ใดเหตุการณ์ สรุปว่าการขัดจังหวะทำให้เกิดการดำเนินการดังนี้ 1) การโต้ตอบเหตุการณ์พิเศษ 2) การเก็บสถานภาพการทำงานของซีพียูในปัจจุบันโดยฮาร์ดแวร์เพื่อใช้ในเวลาที่ทำงานต่อ 3) การส่งการควบคุมไปยังโปรแกรมพิเศษที่จัดการการขัดจังหวะที่เกิดขึ้น

เพื่อให้เกิดความเข้าใจง่ายขึ้นจึงแสดงการขัดจังหวะการทำงาน จากรูปที่ 10.1 โดยแสดงให้เห็นว่า เมื่อเกิดการขัดจังหวะ ณ ตำแหน่งหมายเลข 1 นั้น จะบังคับให้ซีพียูส่งการควบคุมไปยัง รوتينการขัดจังหวะ (Interrupt Routine) ซึ่งอยู่ที่ตำแหน่งหมายเลข 2 เมื่อเสร็จสิ้นการทำงานของ รوتينการขัดจังหวะแล้ว ซีพียู จะย้ายกลับมาทำงานในตำแหน่งหมายเลข 3 ต่อไปในโปรแกรมเดิม การขัดจังหวะที่เกี่ยวข้องกับ ไอ/โอ และการประมวลผลมีดังนี้

1. การขัดจังหวะของ ไอ/โอ (I/O Interrupt) ได้แก่

- เมื่อคำสั่ง ไอ/โอ ไม่ถูกต้อง
- เมื่อเสร็จสิ้นการทำงานของแชนแนล
- เมื่อเสร็จสิ้นการทำงานของเครื่องมือนำข้อมูลเข้าออก



รูปที่ 10.1 การเกิดการขัดจังหวะ

2. การขัดจังหวะของระบบควบคุม (Supervisor Call Interrupt) เป็นการเรียกใช้ระบบปฏิบัติการจากโปรแกรมของผู้ใช้เครื่องที่กำลังถูกประมวลผลอยู่ เช่น ใช้ให้ระบบปฏิบัติการหาพื้นที่ในหน่วยความจำเพิ่มขึ้นหรือส่งให้อ่านจากอุปกรณ์ใด ๆ เป็นต้น

3. การขัดจังหวะของเครื่อง (Machine-Check Interrupt) เป็นการขัดจังหวะเมื่อตรวจพบข้อผิดพลาดของเครื่อง การจัดการขัดจังหวะนี้สามารถกระทำได้โดยใช้ตัวจัดการสถานะ 2 ชนิดคือ

- ตัวจัดการสถานะการทำงานของโปรแกรม (Program Status Word : PSW) หรือเรียกว่า PSW ใช้สำหรับเก็บสถานะภาพของหน่วยประมวลผล จะมีสองสถานะคือ เก่าและใหม่

- ตัวจัดการสถานะการทำงานของตัวควบคุมการรับส่งข้อมูล (Channel Status Word : CSW) หรือใช้สำหรับเก็บสถานะภาพของแชนแนล

เมื่อเกิดการขัดจังหวะขึ้น สถานะการทำงานของซีพียูจะถูกเก็บไว้ใน PSW เพื่อใช้ในการย้อนกลับมาดูในภายหลัง หรือเมื่อจะทำงานต่อไป PSW ในปัจจุบันจะถูกบันทึกเก็บไว้ใน PSW ของมัน เมื่อจะส่งคอนโทรลไปยังโปรแกรมการขัดจังหวะ ก็เก็บสถานะภาพของคำสั่งใหม่ใน PSW อันใหม่ เมื่อเสร็จสิ้นการทำงานของโปรแกรมการขัดจังหวะ ก็จะมีคำสั่งพิเศษของระบบปฏิบัติการที่จะอ่าน PSW อันเก่าให้กลับมาเป็น PSW อันใหม่อีกครั้ง เพื่อจะทำงานของโปรแกรมที่ถูกขัดจังหวะต่อไป สังเกตไว้ว่า PSW อันใหม่นั้น จะใช้สำหรับการประมวลผลในขณะนั้น ส่วน PSW เก่าจะไว้สำหรับเก็บ PSW ของหน่วยประมวลผลเมื่อถูก ขัดจังหวะเพื่อจะนำมาใช้ในการประมวลผลครั้งต่อไป

ในระหว่างที่การขัดจังหวะของโปรแกรมถูกประมวลผลอยู่ มันสามารถดูใน PSW เก่าได้ว่าอะไรเป็นสาเหตุที่ทำให้เกิดการขัดจังหวะใน PSW เก่าจะมีรหัสของการขัดจังหวะ และตำแหน่งของโปรแกรมที่ถูกขัดจังหวะในขณะที่ประมวลผลอยู่

ในบางกรณี ห้ามไม่ให้มีการขัดจังหวะ เช่น เมื่อเกิดการขัดจังหวะไอ/โอ ของอุปกรณ์หนึ่ง เช่น เครื่องขบดิสก์ สถานะภาพในปัจจุบันถูกเก็บไว้ใน PSW เก่า ในขณะที่กำลังประมวลผลเกี่ยวกับการขัดจังหวะของโปรแกรมอยู่ ก็อาจจะเกิดขัดจังหวะของไอ/โอ จากเครื่องพิมพ์ ซึ่งทำให้สถานะภาพอันใหม่ถูกเก็บไว้ใน PSW และจะทำลายการขัดจังหวะอันเก่าเสีย ทำให้ไม่สามารถกลับไปทำงานของโปรแกรมอันแรกได้อีก ในลักษณะเช่นนี้มีวิธีการจัดการหลายวิธี เช่น

- ห้ามขัดจังหวะโดยเด็ดขาด ในขณะที่กำลังประมวลผลเกี่ยวกับการขัดจังหวะอันก่อน
- ห้ามการขัดจังหวะชั่วคราวจนกว่า PSW อันเก่าจะถูกจำลองถ่ายไปเก็บไว้ที่ไหนสักแห่งที่เรียกว่า แถวคอยการขัดจังหวะ (Interrupt Queuing)

ในการทำงานของระบบถ้าไม่มีงานที่เข้ามาดำเนินการก็จะไม่มีการให้บริการไอ/โอ และไม่มี การตอบสนองความต้องการของผู้ใช้ระบบปฏิบัติการ ก็จะไม่ว่าอะไรจนกว่าจะมีเหตุการณ์ บางอย่างเกิดขึ้น เหตุการณ์นี้จะเกิดขึ้นจากการส่งสัญญาณของการขัดจังหวะ หรือแท็ปรีบ (Trap) เป็น การขัดจังหวะที่เกิดจากซอฟต์แวร์ (Software-Generated Interrupt) ซึ่งเกิดจากการผิดพลาด (Error) เช่น การหารด้วย 0 หรือการเข้าใช้หน่วยความจำที่ไม่ถูกตำแหน่ง หรือเป็นการร้องขอจากโปรแกรม ของผู้ใช้ แล้วระบบปฏิบัติการก็จะให้บริการที่จุดนี้

ดังนั้น ระบบปฏิบัติการก็จะสร้างการขัดจังหวะขึ้น ตัวดำเนินการขัดจังหวะ (Interrupt-Driven) เป็นลักษณะหนึ่งของระบบปฏิบัติการ ซึ่งกำหนดเป็น โครงสร้างเมื่อเกิดการขัดจังหวะขึ้น ฮาร์ดแวร์ก็จะส่งตัวควบคุมไปยังระบบปฏิบัติการ โดยขั้นแรกระบบปฏิบัติการก็จะเก็บสถานะ การทำงานของซีพียู โดยการเก็บค่าไว้ในรีจิสเตอร์ และตัวชี้โปรแกรม (Program Counter) จากนั้นก็จะดู ว่าเป็นการขัดจังหวะชนิดไหน การที่จะกำหนดความต้องการทำการขัดจังหวะเราเรียกว่า โพลลิ่ง (Polling) หรือเราจะดูที่ลำดับความสำคัญของอุปกรณ์ ไอ/โอ หรือดูที่ตารางของการขัดจังหวะ (Vectored Interrupt) สำหรับแต่ละชนิดของการขัดจังหวะมันจะถูกส่งกลับไปยังโปรแกรมของผู้ใช้ สำหรับอีกวิธีหนึ่งคือมันจะส่งการควบคุมกลับไปยังโปรแกรมของผู้ใช้โดยไม่รอให้เสร็จสมบูรณ์ ก่อน

การที่จะรอจนไอ/โอทำงานเสร็จเรียบร้อยแล้ว คอมพิวเตอร์บางประเภทจะมีคำสั่งพิเศษ เรียกว่า “wait” คำสั่งนี้จะทำให้ซีพียูรอจนกว่าจะเจอคำสั่งการขัดจังหวะต่อไป วงรอบ (Loop) นี้จะ วิ่งต่อไปจนการขัดจังหวะเกิดขึ้นใหม่ มันจะส่งการควบคุมไปยังส่วนอื่นของระบบปฏิบัติการ คำสั่ง “wait” ช่วยให้งานดีขึ้น ถ้ามันสร้างลำดับของคำสั่งที่รับมา ซึ่งมันเป็นส่วนที่ช่วงชิงการใช้ งานในหน่วยความจำระหว่างอุปกรณ์ไอ/โอ กับซีพียู โดยที่อุปกรณ์ไอ/โอ ส่งข้อมูลส่วนซีพียูจะส่ง คำสั่ง

สิ่งที่เป็นประโยชน์อันหนึ่งที่สำคัญของการที่รอให้ ไอ/โอ ทำงานสมบูรณ์ ก็คือการดูว่าการ ทำไอ/โอ 1 ครั้งต้องใช้เวลาเท่าไร ดังนั้นเมื่อมีการขัดจังหวะของไอ/โอ เกิดขึ้นระบบปฏิบัติการ จะรู้ ว่าอุปกรณ์อะไรที่ทำให้เกิดการขัดจังหวะ ซึ่งเหมาะที่จะรวมการทำ ไอ/โอ อื่น ๆ ที่คล้าย ๆ กันเข้า ไปด้วยกัน

ในการดำเนินการอีกด้านหนึ่งให้เริ่มทำงานไอ/โอ และขณะเดียวกันก็ส่งคืนการควบคุมไป ยังโปรแกรมของผู้ใช้ โปรแกรมเรียกใช้ระบบหรือเรียกว่าคำสั่งเรียกระบบ (System Call) ก็จะบอก ให้ผู้ใช้รอจนกว่า ไอ/โอจะเสร็จ ดังนั้นเราจึงต้องการคำสั่ง wait ซึ่งก่อนหน้านั้นเราสามารถที่จะ เก็บความต้องการของหลาย ๆ ความต้องการไอ/โอ ในเวลาเดียวกัน โดยที่ระบบปฏิบัติการจะใช้แต่ ละตารางซึ่งบรรจุอุปกรณ์ ไอ/โอ เรียกว่าตารางสถานะของอุปกรณ์ (Device-status Table) ในแต่ละ

ช่องของตาราง จะชี้ไปยังที่อยู่และสถานะของอุปกรณ์แต่ละชนิดซึ่งสถานะของมันประกอบไปด้วย ไม่มีคำสั่ง (No Function) ว่าง (Idle) กำลังทำงาน (Busy) ถ้าเครื่องกำลังทำงานอยู่ตามคำสั่ง ชนิดของคำสั่งและพารามิเตอร์ตัวอื่น ก็จะเก็บไว้ในตารางสถานะของอุปกรณ์ด้วย ดังนั้นอาจจะมีโปรแกรมที่ต้องการทำงานหลาย ๆ อย่างบนอุปกรณ์เดียวกัน ซึ่งเราสามารถที่จะกำหนดรายการ (List) ของการรอ ดังนั้นในตารางอุปกรณ์ ไอ/โอ ระบบปฏิบัติการก็จะมีรายการของความต้องการอุปกรณ์ของแต่ละตัวอยู่ด้วย

เมื่อมีการขัดจังหวะมาที่อุปกรณ์ ไอ/โอ เพื่อที่ต้องการใช้งานระบบปฏิบัติการก็จะรู้ว่า มีอุปกรณ์ ไอ/โอ ตัวไหนที่ทำให้เกิดการขัดจังหวะมันก็จะสร้างตัวชี้ (Index) ในตารางอุปกรณ์ ไอ/โอ เพื่อที่จะกำหนดสถานะของอุปกรณ์นั้น และทำการแก้ไขตารางเพื่อแสดงสิ่งที่เกิดขึ้นจากการขัดจังหวะของอุปกรณ์ สัญญาณของการขัดจังหวะจะเสร็จสมบูรณ์เมื่อความต้องการของแต่ละ ไอ/โอ เสร็จ ถ้าหากว่ามีการเพิ่มความต้องการที่จะใช้อุปกรณ์นั้นอีก ระบบปฏิบัติการก็จะเริ่มทำการประมวลผลเพื่อแสดงความต้องการอุปกรณ์ต่อไป

ลำดับสุดท้ายการควบคุมก็จะถูกส่งกลับคืนมาจากการทำการขัดจังหวะ ไอ/โอ ถ้าโปรแกรมยังรอ เพื่อให้ความต้องการที่จะทำงานนั้นเสร็จสมบูรณ์ ระบบก็จะส่งคืนการควบคุมทั้งหลายให้กับโปรแกรม ในอีกแง่หนึ่งเราจะส่งคืนค่านั้นกลับไปยังงานก่อนที่เราจะทำการขัดจังหวะ ไอ/โอ เพื่อที่จะประมวลผลโปรแกรมของผู้ใช้ หรือขณะที่รออยู่ในวงรอบของการรอ (wait loop)

ในระบบโต้ตอบทันที (Interactive) หน่วยความจำที่เก็บข้อความผิดพลาดผลพลาดที่เกิดขึ้นจะบันทึกข้อความดังกล่าวลงเพิ่มข้อมูลของผู้ใช้ หรืออาจจะพิมพ์ออกมาตรวจสอบ หรืออาจจะตรวจสอบด้วยสายตาที่หน้าจอคอมพิวเตอร์ และบางครั้งอาจจะต้องแก้ไขและเริ่มโปรแกรมใหม่ การทำงานวิธีนี้จะให้ผลดีในระยะยาวเนื่องจากข้อผิดพลาดจะถูกตรวจพบโดยฮาร์ดแวร์เราต้องแน่ใจว่า ถึงอย่างไรก็ตามข้อผิดพลาดทั้งหมดจะถูกตรวจพบ แต่เราจะต้องป้องกันระบบปฏิบัติการและโปรแกรมอื่น ๆ รวมทั้งข้อมูลจากการทำงานที่ผิดพลาดของโปรแกรม การป้องกันจำเป็นมากสำหรับการระบบที่มีการแบ่งปันทรัพยากร วิธีที่จะนำเอาฮาร์ดแวร์มาช่วยกำหนดความแตกต่างในการทำงานระหว่างหลาย ๆ โหมดของการปฏิบัติงาน อย่างน้อยที่สุดเราจำเป็นต้องมีสองโหมดของการปฏิบัติงาน คือ โหมดผู้ใช้ (User Mode) และโหมดควบคุม (Monitor Mode หรือ บางทีเรียกว่า Supervisor Mode หรือ System Mode) ซึ่งวิธีนี้จะมีบิตที่ใช้สำหรับการควบคุม เรียกว่า ซึ่งโหมด-บิตนี้ถูกเพิ่มอยู่ในฮาร์ดแวร์ของคอมพิวเตอร์เพื่อแสดงโหมดปัจจุบัน โดยที่โหมดควบคุมมีค่าเป็น 0 และ โหมดผู้ใช้มีค่าเป็น 1

การควบคุมการทำงานด้วยโหมด-บิตนี้เราสามารถแบ่งโหมดการทำงานระบบปฏิบัติการได้ โดยที่โหมดหนึ่งจะแทนการกระทำของระบบปฏิบัติการ และอีกโหมดหนึ่งจะแทนการกระทำ

ของผู้ใช้ ซึ่งโหมคการทำงานปกติจะเป็นโหมคผู้ใช้ เมื่อไรก็ตามที่โปรแกรมของผู้ใช้เกิดการทำงานหยุดชะงัก ฮาร์ดแวร์จะเปลี่ยนจากโหมคผู้ใช้เป็นโหมคควบคุม คือ เปลี่ยนโหมค-บิตจาก 1 เป็น 0 ด้วยเหตุนี้ เมื่อไรก็ตามที่ระบบปฏิบัติการกำลังควบคุมคอมพิวเตอร์แสดงว่ามีสถานะเป็นโหมคควบคุม

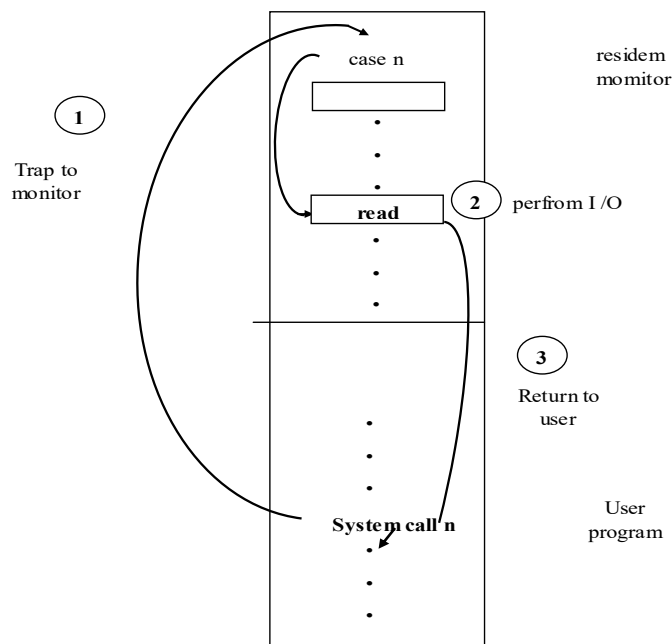
เพื่อป้องกันผู้ใช้จากการปฏิบัติงานของ ไอ/โอ ให้ถูกต้อง ซึ่งมีการศึกษาการให้คำแนะนำในการใช้ ไอ/โอ คือผู้ใช้จะไม่สามารถติดต่อกับระบบ ไอ/โอ ได้ โดยไม่มีระบบปฏิบัติการ สำหรับการป้องกันระบบ ไอ/โอ ที่สมบูรณ์แบบ เราต้องแน่ใจว่าโปรแกรมสำหรับผู้ใช้ไม่สามารถทำงานในโหมคควบคุมได้ ถ้าคอมพิวเตอร์จะทำการประมวลผลในโหมคผู้ใช้ มันจะต้องทำการสลับไปจากโหมคควบคุม เมื่อไรก็ตามที่เกิดการการขัดจังหวะและข้ามไปที่ส่วนเว็คเตอร์ขัดจังหวะ โปรแกรมก็จะถูกนำไปเก็บที่เว็คเตอร์ขัดจังหวะ โดย ตำแหน่งหน่วยความจำตัวใหม่จะเขียนทับตำแหน่งหน่วยความจำตัวเดิมที่มีอยู่แล้ว

เราต้องป้องกันเว็คเตอร์ขัดจังหวะจากผู้เขียน โปรแกรม และต้องป้องกันการแก้ไขส่วนของงานที่สนับสนุนการขัดจังหวะ ดังนั้นการได้รับการควบคุมจากงานสนับสนุนการขัดจังหวะ จะต้องประมวลผลในโหมคควบคุมด้วย ถ้ามิฉะนั้นแล้วผู้ใช้จะไม่สามารถอำนาจให้ใช้คอมพิวเตอร์ได้

สถาปัตยกรรมระบบทั่วไป

เพื่อลดเวลาในการติดตั้งการปรับปรุงระบบคอมพิวเตอร์ให้มีความสามารถมากขึ้นการทำงานที่เป็นแบบแบท การทำบัพเฟอร์ การทำสพูล มัลติโปรแกรมมิ่ง และการแบ่งเวลาความสามารถเหล่านี้จะช่วยจัดสรรทรัพยากรของคอมพิวเตอร์ให้ทำงานใน โปรแกรมต่างกันและงานที่ต่างกัน การแบ่งปันงานจะนำไปสู่การแก้ไขโครงสร้างของระบบคอมพิวเตอร์โดยมีระบบปฏิบัติการทำหน้าที่ดูแลระบบคอมพิวเตอร์โดยเฉพาะเรื่อง ไอ/โอ ซึ่งต้องจัดการควบคุมให้มันทำงานต่อเนื่องคงเส้นคงวาและทำงานได้อย่างถูกต้อง ในการดูแลระบบควบคุมผู้พัฒนาจะใช้การประมวลผลแบบ 2 โหมคร่วมกันคือ โหมคผู้ใช้ และโหมคควบคุม วิธีนี้จะช่วยสนับสนุนแนวความคิดเกี่ยวกับคำสั่งพิเศษซึ่งสามารถประมวลผลเฉพาะในโหมคควบคุมเท่านั้นได้แก่คำสั่งเกี่ยวกับ ไอ/โอ, คำสั่งในการแก้ไขในหน่วยความจำ คำสั่งในการจัดการรีจิสเตอร์ หรือการตั้งเวลา คำสั่งในการหยุดการทำงานก็ถือว่าเป็นคำสั่งพิเศษ เพราะโปรแกรมของผู้ใช้ไม่สามารถที่จะหยุดการทำงานกรณียังไม่จบงานได้ คำสั่งที่ใช้ในระบบขัดจังหวะ การเรียกใช้และเลิกใช้การดำเนินงานของตัวจับเวลา และ ไอ/โอ จะถูกต้อง ขึ้นอยู่กับการขัดจังหวะได้ถูกต้อง คำสั่งที่เปลี่ยนจากโหมคผู้ใช้ไปเป็นโหมคควบคุม การเปลี่ยนไปใช้โหมค-บิตในหลาย ๆ เครื่อง ก็ถือเป็นคำสั่งพิเศษ

การที่ผู้ใช้จะใช้คำสั่งของ ไอ/โอ ต้องแปลงเป็นคำสั่งพิเศษก่อนเพื่อป้องกันโปรแกรมของผู้ใช้เข้ามาทำงานกับ ไอ/โอ โดยตรง การแก้ปัญหานี้คือเมื่อผู้ใช้ต้องการใช้คำสั่ง ไอ/โอ ให้ติดต่อผ่านตัวควบคุมก่อนในคอมพิวเตอร์สมัยใหม่ จะมีคำสั่งเรียกระบบ ถูกจัดให้เป็นคำสั่งพิเศษ คำสั่งเรียกระบบบางครั้งจะถูกส่งจากโปรแกรมของผู้ใช้ให้ทำบางคำสั่งเกี่ยวกับการใช้ไอ/โอ ซึ่งไม่มีรหัสคำสั่ง คือรหัสที่ไม่มีตัวดำเนินการฮาร์ดแวร์อธิบายความหมายของมัน มันจะสร้างคำสั่งที่ผิด (Illegal Instruction) แล้วถูกแปลงให้เป็นตัวควบคุม ในระบบปฏิบัติการส่วนใหญ่ใช้เทคนิคนี้ เมื่อคำสั่งเรียกระบบ ถูกประมวลผลมันจะถูกทำงาน โดยฮาร์ดแวร์ เปรียบเสมือนการขัดจังหวะด้วยซอฟต์แวร์ จะควบคุมผ่านทางเว็คเตอร์ขัดจังหวะ ไปยังรูทีนบริการ (Service Routine) ในระบบปฏิบัติการ จากนั้น โหมด-บิต ก็จะถูกตั้งค่าให้เป็นโหมดควบคุม ตัวควบคุมจะทดสอบคำสั่งขัดจังหวะที่ส่งมาจากคำสั่งเรียกระบบ จะมีพารามิเตอร์คอยชี้ว่าผู้ใช้ต้องการอะไรจากนั้นก็ให้ข้อมูลที่ต้องการส่งผ่านไปยังรีจิสเตอร์ที่จะทำ ไอ/โอ ของโปรแกรมผู้ใช้ และจะประมวลผลคำสั่งเรียกระบบเพื่อให้ระบบปฏิบัติการทำไอ/โอ ให้กับโปรแกรมของผู้ใช้ ทำในโหมดควบคุม และตรวจสอบว่าความต้องการของผู้ใช้ถูกต้องหรือไม่ถ้าความต้องการถูกต้อง ระบบปฏิบัติการก็จะทำสิ่งที่ผู้ใช้ต้องการแสดงดังรูปที่ 10.2 ตำแหน่งแรกเมื่อมีการขัดจังหวะระบบปฏิบัติการกำหนดเลขรหัสของการขัดจังหวะ เมื่อได้รับการตรวจสอบและได้รับการบริการ จะดำเนินการในตำแหน่งที่ 2 โดยดำเนินการไอ/โอตามหน้าที่จนเสร็จ ในตำแหน่งที่ 3 เมื่อไอ/โอทำงานเสร็จตามต้องการแล้วระบบปฏิบัติการจะส่งการควบคุมการทำงานกลับไปยังโปรแกรมของผู้ใช้ตามเดิม



รูปที่ 10.2 การใช้คำสั่งเรียกระบบ ทำ ไอ/โอ

การจัดการอุปกรณ์

อุปกรณ์ต่าง ๆ ที่ต่อพ่วงในระบบคอมพิวเตอร์จะถูกควบคุมการใช้งานโดยระบบปฏิบัติการ มีการกำหนดโครงสร้างและรูปแบบการจัดเก็บข้อมูล แตกต่างกันไปตามลักษณะของอุปกรณ์นั้น ๆ สำหรับอุปกรณ์ที่ใช้กันทั่วไปมีดังต่อไปนี้

การจัดการดิสก์

ลักษณะของแผ่นดิสก์เป็นแผ่นกลมแบน ซึ่งมีสารแม่เหล็กฉาบไว้ที่ผิวสำหรับใช้บันทึกข้อมูลต่าง ๆ ปกติจะวางหลาย ๆ แผ่นซ้อนกันเป็นตึก (Disk Pack) โดยใช้แกนหมุน (Spindle) ร่วมกัน ดิสแต่ละแผ่นแบ่งเป็นร่องหรือแทร็ก (Track) ทั้งสองด้าน ซึ่งเป็นแนววงแหวนซ้อนกันอยู่ ข้อมูลจะเก็บไว้ในแทร็กนี้เรียงต่อเนื่องกันบิดต่อบิด และแต่ละแทร็กจะถูกแบ่งออกเป็นเสี้ยวหรือเซกเตอร์ (Sector) แต่ละเซกเตอร์จะเก็บข้อมูลได้หนึ่งเรคคอร์ด สำหรับแทร็กที่อยู่ตรงกันในแนวตั้งของวงรอบ เรียกว่าไซลินเดอร์ (Cylinder) ตัวอย่างเช่น การจัดรูปแบบการเก็บข้อมูลของดิสก์บนเครื่องไมโครคอมพิวเตอร์ มีการจัดเก็บข้อมูลที่เห็นชัดเจนสำหรับระบบปฏิบัติการ DOS ดังนี้ มีการจัดจำนวนแทร็ก 40 แแทร็ก ต่อหน้า ดิสก์ 1 แผ่นบันทึกได้ 2 ด้าน ยกเว้นแผ่นแรกหน้าบนสุดและแผ่นสุดท้ายหน้าล่างสุดที่บันทึกได้หน้าเดียว แต่ละแทร็กมีจำนวนเซกเตอร์ต่อแทร็ก 9 เซกเตอร์ และจำนวนไบต์ต่อ เซกเตอร์ 512 ไบต์ โดยทั่วไปความจุของดิสก์ = จำนวนหน้า x จำนวนแทร็กต่อหน้า x จำนวนเซกเตอร์ต่อแทร็ก x ขนาดของเซกเตอร์ มีความจุเป็นจำนวนไบต์ สำหรับเครื่องคอมพิวเตอร์ระบบใหญ่ในดิสก์ไครฟ์ จะมีแผ่นดิสก์หลายแผ่นวางอยู่บนแกนหมุนแกนเดียวกัน ดิสก์ทุกแผ่นจะมีการแบ่งแทร็กเหมือนกัน แแทร็กที่อยู่ในแนวเดียวกันก็คือ อยู่บนรอบเดียวกัน เช่น แแทร็ก 00 ของทุกแผ่น ก็คือ 1 วงรอบ แแทร็ก 01 ของทุกแผ่นก็คืออีก 1 วงรอบ และต่อไป เป็นต้น ดิสก์ไครฟ์จะมีหัวอ่าน/เขียน (Read/Write Head) หรือเรียกสั้น ๆ ว่าหัวอ่าน ซึ่งทำหน้าที่บันทึกข้อมูลลงแผ่นดิสก์หรืออ่านข้อมูลจากแผ่นดิสก์ขึ้นมาส่งไปไว้ในหน่วยความจำหลัก หัวอ่านนี้จะเลื่อนเข้า-ออก ในแนวตั้งฉากกับจุดศูนย์กลางการหมุนของแผ่นดิสก์ เมื่อมีการเข้าถึงข้อมูล แผ่นดิสก์จะหมุนรอบแกนหมุนด้วยความเร็วรอบค่าหนึ่ง และหัวอ่านจะเลื่อนไปมาเพื่อทำการอ่านหรือเขียนข้อมูลบนแผ่นดิสก์โดยหัวอ่าน/บันทึก ซึ่งต่ออยู่กับแขน (boom) ที่ยื่นเข้าไปจากนอกดิสก์ เพื่อให้หัวอ่านตรงกับตำแหน่งของแทร็กที่ต้องการ จากนั้นก็รอให้ดิสก์หมุนไปจนเซกเตอร์ที่ต้องการบนแทร็กมาตรงกับหัวอ่าน/บันทึก จึงทำการอ่าน/บันทึกข้อมูลที่ต้องการ การเลื่อนหัวให้ไปอยู่เหนือแทร็กที่ต้องการเรียกว่า การค้นหาแทร็ก (seek) และเวลาที่ใช้ เรียกว่า เวลาในการค้นหาแทร็ก (seek time) ส่วนเวลาที่งานหมุนให้เซกเตอร์ที่ต้องการมาอยู่ใต้หัว เรียกว่า การค้นในแทร็ก (latency) และเรียกเวลาที่เสียไปสำหรับการนี้ว่า เวลาในการค้นในแทร็ก (latency time) สำหรับเวลาในการอ่านหรือบันทึกข้อมูลภายใต้หัวนั้น เรียกว่า เวลาในการโอนถ่ายข้อมูล (transfer time)

ดังนั้นเวลาในการเข้าถึงข้อมูล (access time) สำหรับดิสก์ประกอบด้วยเวลา 3 ส่วน คือ

1. เวลาที่ใช้ค้นหา (seek time) คือ ช่วงเวลาที่หัวอ่านเลื่อนจากแทร็กที่อยู่ในขณะนั้นไปอยู่เหนือแทร็กที่ต้องการ
2. เวลาแฝง (latency time) คือ ช่วงเวลาที่แผ่นดิสก์หมุนตัวเองเพื่อค้นหาข้อมูลในแทร็กจนกระทั่ง เซกเตอร์ที่ต้องการมาอยู่ใต้หัวอ่าน
3. เวลาที่ใช้โอนถ่ายข้อมูล (transfer time) คือ ช่วงเวลาที่หัวอ่านอ่านข้อมูล หรือเขียนข้อมูลลงบน แผ่นดิสก์

ดังนั้น เวลาในการเข้าถึงข้อมูล = เวลาค้นหา + เวลาแฝง + เวลาโอนถ่ายข้อมูล โดยปกติ ดิสก์จะหมุนด้วยความเร็วสูงทำให้เวลาแฝงมีค่าน้อยมากเมื่อเทียบกับส่วนอื่นจนบางครั้งอาจ ตัดทิ้งได้ สำหรับเวลาถ่ายเทข้อมูลจะขึ้นอยู่กับปริมาณข้อมูลที่ต้องการอ่านหรือเขียนและความเร็วของการส่ง ข้อมูล ซึ่งค่าทั้งสองนี้อยู่นอกเหนือการควบคุมของระบบปฏิบัติการ ดังนั้น จึงมีเพียงเวลา ค้นหาเท่านั้น ที่ระบบปฏิบัติการสามารถควบคุมได้บ้างบางส่วน เวลาในการเข้าถึงข้อมูลนี้มีผลกระทบต่อการทำงานของระบบ ถ้าเวลาในการเข้าถึงน้อยลง การทำงานของระบบก็จะดีขึ้น ทำงานได้เร็ว และได้ปริมาณงานเพิ่มขึ้นในเวลาเท่าเดิม หน้าที่อีกประการหนึ่งของระบบปฏิบัติการก็คือ พยายามลดเวลาในการเข้าถึงให้น้อยที่สุดเท่าที่จะทำได้ กลไกการจัดสรรดิสก์ (disk scheduling) จะทำหน้าที่ลดเวลาค้นหาให้น้อยที่สุดเท่าที่จะทำได้ กลไกการจัดสรรดิสก์มีหลายแบบ ขึ้นอยู่กับผู้ออกแบบระบบปฏิบัติการจะเลือกใช้ ซึ่งอัลกอริทึมการจัดตารางเวลาการเข้าถึงแทร็กต่าง ๆ ได้กล่าวไว้ในบทก่อนหน้า

การจัดการเทป

เทปเป็นอุปกรณ์จัดเก็บข้อมูลสำรองชนิดหนึ่ง ที่ใช้ทั่วไปในเครื่องระบบใหญ่เช่น มินิคอมพิวเตอร์หรือเมนเฟรมคอมพิวเตอร์ ถึงแม้การทำงานของเทปจะช้าแต่ก็ยังเป็นที่นิยมใช้กันอยู่ในปัจจุบัน เนื่องจากมีราคาถูก การเข้าถึงข้อมูลในเทปเป็นแบบตามลำดับคือต้องอ่านข้อมูลที่อยู่ส่วนหน้าขึ้นมาตรวจสอบทุกกลุ่ม จนกว่าจะพบข้อมูลที่ต้องการเช่นเดียวกับระบบปฏิบัติการเป็นผู้กำหนดรูปแบบการเก็บข้อมูลลงบนเทป เทปจะมีการจัดแบ่งการเก็บข้อมูลเป็นแทร็ก (Track) บล็อก (Block) และ เรคคอร์ด (Record) การแบ่งแทร็กจะแบ่งตามความยาวของเทป โดยทั่วไปแบ่งเป็น 9 แทร็ก สำหรับเก็บข้อมูลขนาด 8 บิต การเก็บข้อมูล 1 ไบต์ (8 บิต) จะเก็บตามแนวขวางของเทป 1 บิต ต่อ 1 แทร็ก แทร็กที่เหลือเป็นบิตตรวจสอบ (Parity Bit) ไว้ตรวจสอบความผิดพลาดของการอ่าน/เขียน หนึ่ง เรคคอร์ดจะมีหลายไบต์ เช่น 512 ไบต์ต่อเรคคอร์ด ระหว่างเรคคอร์ดจะมีเนื้อที่ว่างเรียกว่า ช่องว่างระหว่างเรคคอร์ด (Inter-record Gap) มีไว้เพื่อแบ่งแยกแต่ละเรคคอร์ดออกจากกันและไม่มีการเก็บข้อมูลลงใน ช่องว่างนี้ด้วย หลากๆ เรคคอร์ดรวมเป็น 1 บล็อก และ

เช่นเดียวกันระหว่างบล็อกแต่ละบล็อกจะมี เนื้อที่ว่างเรียกว่า ช่องว่างระหว่างบล็อก (inter-block gap) โดยทั่วไปช่องว่างระหว่างบล็อกจะกว้างกว่าช่องว่างระหว่างเรคคอร์ด

การจัดการคีย์บอร์ด

คีย์บอร์ดเป็นอุปกรณ์นำข้อมูลเข้าสู่ระบบคอมพิวเตอร์ที่มีรูปแบบการส่งข้อมูลเป็นสาย (Stream) นั่นคือลำดับของการส่งข้อมูลขึ้นอยู่กับลำดับของการกดคีย์แต่ละคีย์ การเรียงลำดับผิดก็จะเกิดความหมายผิดตาม ไปด้วย เช่น กด "ก" และตามด้วย "ข" จะเกิดลำดับของข้อมูลเป็น "ก ข" ต่างกับการกด "ข" ก่อนแล้วกด "ก" ซึ่งให้ลำดับของข้อมูลเป็น "ข ก" ดังนั้นลำดับของข้อมูลจึงมีความสำคัญ การจัดการคีย์บอร์ดของระบบปฏิบัติการแบ่งได้เป็น 2 โหมด คือ

1. โหมดดิบ (raw mode) โหมดนี้อาจเรียกได้ว่าเป็นการจัดการเชิงอักขระ (character-oriented) โดยมีการจัดการดังนี้ สมมติว่าผู้ใช้ตั้งใจจะพิมพ์คำว่า 'good' แต่เวลาพิมพ์จริงๆ พิมพ์ผิดเป็น 'ogod' ดังนั้นเมื่อต้องการแก้ไขต้องกด backspace 4 ครั้ง (เพื่อลบตัวอักษร 4 ตัวแล้วหลังออก) และกด 'g' 'o' 'd' ตามด้วย enter ตามลำดับ ในลักษณะนี้ โปรแกรมงานของผู้ใช้จะรับตัวอักษรทั้งหมด 13 ตัว (รวมตัวอักษรที่พิมพ์ผิด backspace และ enter)

2. โหมดสุก (cook mode) โหมดนี้อาจเรียกได้ว่าเป็นการจัดการเชิงบรรทัด (line-oriented) ลักษณะการส่งข้อมูลผ่านโหมดนี้จะต่างกับโหมดดิบเพียงเล็กน้อย ในกรณีตัวอย่างที่กล่าวไว้ในโหมดดิบ ระบบปฏิบัติการจะส่งตัวอักษรให้โปรแกรมทั้งหมด 13 ตัวอักษร แต่ถ้าโหมดสุก ระบบปฏิบัติการจะส่งเฉพาะตัวอักษรที่ถูกแก้ไขถูกต้องแล้วเท่านั้น นั่นคือจะส่งเพียง "good" ไปให้ในระบบปฏิบัติการบางตัวผู้ใช้สามารถเลือกโหมดการจัดการรับข้อมูลได้ทั้ง 2 โหมด

ในการรับส่งข้อมูลจากคีย์บอร์ด ระบบปฏิบัติการจะจองพื้นที่หน่วยความจำไว้ส่วนหนึ่งเรียกว่าบัฟเฟอร์ (buffer) ใช้สำหรับรับข้อมูลที่ป้อนเข้ามาจากคีย์บอร์ด ก่อนที่จะส่งไปให้โปรแกรมต่างๆ นำไปใช้ในการประมวลผล สำหรับโหมดสุกตัวอักษรที่ป้อนเข้ามาจะถูกนำมาจัดเก็บไว้ในบัฟเฟอร์ก่อน ถ้าผู้ใช้จะทำการแก้ไขก็จะแก้ไขข้อมูลที่เก็บไว้ในบัฟเฟอร์ เมื่อผู้ใช้กดแป้น ENTER ข้อมูลของตัวอักษรที่ได้แก้ไขแล้วในบัฟเฟอร์จะถูกส่งต่อไปให้โปรแกรมที่ต้องการใช้ข้อมูล ส่วนกรณีของโหมดดิบ การกดคีย์แต่ละคีย์ ระบบปฏิบัติการอาจส่งตัวอักษรไปให้โปรแกรมที่ต้องการใช้ข้อมูลได้ทันทีโดยไม่ต้องจัดเก็บลงในบัฟเฟอร์ก่อน แต่ถ้าโปรแกรมที่ต้องการใช้ข้อมูล ยังไม่พร้อมที่จะรับข้อมูลที่ป้อนเข้ามาจากผู้ใ้ ข้อมูลตัวอักษรต่างๆ ที่ผู้ใช้ป้อนเข้ามานั้นจะถูกเก็บลงในบัฟเฟอร์ และจะส่งให้โปรแกรมนั้นเมื่อมันพร้อมที่จะรับข้อมูล การจัดสรรบัฟเฟอร์ของคีย์บอร์ดมี 2 วิธีที่นิยมใช้กัน วิธีแรกคือ ระบบปฏิบัติการ จะมีบัฟเฟอร์ขนาดเล็ก ๆ เช่น มีขนาด 10 ไบต์หลายตัว รวมกันเป็นกลุ่มเรียกว่ากลุ่มบัฟเฟอร์กลาง เมื่อผู้ใช้ป้อนข้อมูลเข้ามาทางคีย์บอร์ด ระบบปฏิบัติการจะนำข้อมูลเหล่านั้นจัดเก็บลงในบัฟเฟอร์ 1 ตัว และแยกบัฟเฟอร์ตัวนั้นออกจาก

กลุ่มบัพเฟอร์กลาง (ถือว่าเป็นบัพเฟอร์ของผู้ใช้) ถ้าบัพเฟอร์ 1 ตัว เก็บข้อมูลได้ไม่หมด เป็นต้นว่า ผู้ใช้ป้อนข้อมูลเข้ามา 36 ตัวอักษร ระบบปฏิบัติการจะจัดบัพเฟอร์ให้ผู้ใช้ 4 ตัวจากบัพเฟอร์กลาง (4 ตัว เก็บได้ 40 ตัวอักษร) เมื่อข้อมูลนี้ส่งไปให้โปรแกรมนำไปใช้ในการประมวลผลแล้ว ระบบปฏิบัติการจะจัดการคืนบัพเฟอร์ทั้ง 4 ตัวนี้ให้กับกลุ่มบัพเฟอร์กลาง วิธีที่สอง ระบบปฏิบัติการ กำหนดบัพเฟอร์ให้กับผู้ใช้แต่ละคนอย่างอิสระที่ไม่ยุ่งเกี่ยวกับโดยไม่มีกลุ่มบัพเฟอร์กลาง เนื่องจากผู้ใช้อาจจะจำเป็นต้องป้อนข้อมูลเข้ามาเป็นจำนวนมาก ดังนั้น บัพเฟอร์สำหรับผู้ใช้แต่ละคนควรมีขนาดใหญ่พอสมควร เช่น มีขนาด 300 ไบต์ ถ้าในระบบการทำงานของคอมพิวเตอร์มีผู้ใช้เครื่องที่เป็นเทอร์มินัลอยู่ 200 เครื่อง ในเวลาเดียวกัน แสดงว่าระบบจะต้องเสียหน่วยความจำสำหรับจัดเป็นบัพเฟอร์ ขนาด $300 * 200 = 60,000$ ไบต์ ในขณะที่วิธีแรกใช้หน่วยความจำทำบัพเฟอร์เพียง 4,000 ไบต์ ในการกดคีย์แต่ละครั้ง ค่าจากคีย์บอร์ดอาจไม่ได้ส่งรหัสแอสกีไปให้ระบบปฏิบัติการโดยตรง แต่ส่งรหัสคีย์ (key code) ไปให้แทน ดังนั้นระบบปฏิบัติการจะต้องแปลงรหัสคีย์ที่ถูกส่งเข้ามาให้เป็นรหัสแอสกีของคีย์นั้น ๆ เช่นเมื่อผู้ใช้กดแป้น 'y' คีย์บอร์ดจะส่งรหัสคีย์ของ 'y' มาให้ระบบปฏิบัติการ ระบบปฏิบัติการจะต้องแปลงรหัสของคีย์นี้ให้เป็นรหัสแอสกีของตัวอักษร 'y' ตามตารางการแปลงรหัสแอสกีซึ่งระบบปฏิบัติการมีอยู่แล้ว เป็นการเพิ่มความยืดหยุ่นในการใช้งานคีย์บอร์ด โดยที่ผู้ใช้สามารถเปลี่ยนแปลงเป็นต่าง ๆ บนคีย์บอร์ดให้เป็นตัวอักษรตามที่ต้องการได้ โดยทำการปรับเปลี่ยนค่าในตารางแปลงรหัสแอสกีเท่านั้น การทำงานของระบบปฏิบัติการเป็นรูปแบบในเชิงตรรก คีย์บอร์ดและจอภาพเป็นอุปกรณ์ต่างชนิดกัน แต่เมื่อผู้ใช้กดแป้นใด ๆ บนคีย์บอร์ดหรือป้อนข้อมูลจากคีย์บอร์ดเข้ามาในระบบคอมพิวเตอร์แล้ว ผู้ใช้จะเห็นสิ่งที่ได้ป้อนเข้าไปโดยแสดงให้เห็นบนจอภาพ

ระบบไฟล์ (File System)

สำหรับผู้ใช้ไฟล์ข้อมูลจะไม่สนใจว่าระบบไฟล์มีรูปร่างหน้าตาอย่างไรเพราะ เป็นหน้าที่ของระบบปฏิบัติการที่เป็นตัวมองเห็นลักษณะไฟล์ที่เก็บข้อมูลและ โปรแกรมในเครื่องคอมพิวเตอร์ ระบบปฏิบัติการจะรู้ว่าการจัดการกับไฟล์ที่เป็นกลุ่มของอุปกรณ์ เก็บข้อมูลนั้นเป็นนามธรรม เช่น เทป และดิสก์ เป็นต้น ในบทนี้เราพิจารณาถึงวิธีต่าง ๆ ที่จะเข้าถึงไฟล์และอุปกรณ์นั้น ๆ การจัดการไฟล์ในไคเร็กทอรี จึงเป็นการสร้างความสะดวกสบายในการจัดการไฟล์เหล่านั้นซึ่งเราจะเห็นได้จากโครงสร้างไคเร็กทอรี และเมื่อผู้ใช้ ต่าง ๆ มีทางที่จะเข้าถึงไฟล์นั้นได้ จะต้องมีการควบคุมและต้องทราบว่าจะเข้าถึงไฟล์ต่าง ๆ ได้อย่างไร รวมถึงการป้องกันไฟล์ด้วย

การจัดการระบบไฟล์

การจัดการไฟล์เป็นอีกบริการหนึ่งของระบบปฏิบัติการคอมพิวเตอร์ที่สามารถจัดเก็บข้อมูลได้บนสื่อทางกายภาพหลายรูปแบบที่ต่างกัน เช่น ดิสก์ เทป และ ออปติคอลลิสต์ (Optical Disk)

ซึ่งเป็นสื่อที่ใช้เก็บข้อมูลทั่ว ๆ ไป และยังมีอุปกรณ์ในการเก็บข้อมูลที่เป็นลักษณะพิเศษและเป็น การจัดการทางกายภาพอีกหลายชนิด เช่น แฟลชไดรฟ์ (Flash Drive) และ ฮาร์ดดิสก์ภายนอก (External Hard Disk) เป็นต้น ดังนั้นระบบคอมพิวเตอร์ จึงเหมาะกับการใช้ระบบปฏิบัติการซึ่งเป็นระบบที่เป็นนามธรรม ทำการป้องกันทางกายภาพแก่อุปกรณ์เก็บข้อมูลโดยไม่ให้ผู้ใช้สามารถ เข้าถึงข้อมูลจากอุปกรณ์และสื่อจัดเก็บข้อมูล ซึ่งระบบปฏิบัติการได้ทำการป้องกันข้อมูลต่าง ๆ ที่ เก็บไว้บนสื่อจัดเก็บ โดยให้คำนิยามหน่วย ที่ใช้เก็บข้อมูล “ เรียกว่าไฟล์” ที่ระบบปฏิบัติการ สามารถค้นไฟล์ จากอุปกรณ์จัดเก็บทางกายภาพได้ ระบบไฟล์ประกอบด้วยวิธีการเก็บข้อมูล 2 ส่วน คือ

1. เก็บส่วนที่เป็นข้อมูลจริง ๆ นั่นคือบรรจุข้อมูลที่สัมพันธ์กัน
2. เก็บส่วนที่เป็นโครงสร้าง ไคเร็กทอรี ซึ่งส่วนนี้จะเป็นการป้องกันข้อมูลเกี่ยวกับไฟล์ ทั้งหมดที่จัดเก็บอยู่ในระบบสื่อจัดเก็บและอุปกรณ์

คำนิยามไฟล์

ไฟล์เป็นที่เก็บข้อมูลที่เกี่ยวข้องสัมพันธ์กัน อาจจะเป็น โปรแกรม (ทั้ง Source Program และ Object Program) หรือข้อมูล ไฟล์ข้อมูลอาจเป็นตัวเลข ตัวอักษร ตัวอักษรพิเศษ หรือ เลขฐานสอง ไฟล์อาจจะไม่อยู่ในรูปแบบ เช่น เทกซ์ไฟล์ (Text File) หรืออาจอยู่ในรูปแบบใดก็ได้ เช่น Binary File โดยทั่วไปแล้วไฟล์ เป็นลำดับของบิต ไบต์ หรือเรคคอร์ด ซึ่งผู้ที่ให้คำนิยามของ รูปแบบคือ ผู้สร้างหรือ ผู้ใช้ ที่ได้กำหนดขึ้นมาตามความต้องการใช้งาน

ในโครงสร้างของไฟล์จะต้องตั้งชื่อไฟล์ขึ้นมา การตั้งชื่อไฟล์ควรจะเกี่ยวข้องกับลักษณะ ของข้อมูลในไฟล์ เช่น ข้อมูลของพนักงานตั้งชื่อไฟล์ว่า “employee” และโครงสร้างของไฟล์ จะต้องมามีคุณสมบัติ เช่น ชนิด ของไฟล์ วันที่สร้าง ชื่อ (หรือเลขที่บัญชี) ของผู้สร้าง และความยาว หรือขนาดของไฟล์ เป็นต้น

ข้อมูลในไฟล์จะถูกนิยามโดยผู้สร้าง ชนิดข้อมูลที่แตกต่างกันอาจเก็บไว้ในไฟล์ ๆ หนึ่ง ได้ เช่น Source Program Object Program ข้อมูลที่เป็นตัวเลข เป็นตัวอักษร บัญชีเงินเดือน รูป กราฟฟิค และเสียง เป็นต้น ไฟล์จะต้องบรรจุโครงสร้างตามชนิดของมัน เช่น เทกซ์ไฟล์ จะต้อง เกี่ยวข้องกับตัวอักษรเป็นบรรทัด (เป็นหน้า) หรือไฟล์ Source Program เป็นไฟล์ที่มีการบันทึกงาน ที่ทำเป็นตามลำดับขั้นตอนของคำสั่ง และวงรอบตามหน้าที่ต่าง ๆ ที่กำหนดไว้ ส่วนไฟล์ Object Program เป็นไฟล์ที่เป็นไปตามการ โหลดขึ้นมาใช้งานในขณะที่ทำการประมวลผล

หลักในการพิจารณาว่า ระบบปฏิบัติการควรจะทราบหรือสนับสนุนชนิดของไฟล์ คือ ถ้า ระบบปฏิบัติการ ทราบชนิดของไฟล์ก็จะสามารถที่จะปฏิบัติตามไฟล์นั้น ได้อย่างสมเหตุสมผล เช่น ข้อผิดพลาดทั่วไปที่เกิดขึ้น เมื่อผู้ใช้พยายามที่จะพิมพ์ เลขฐาน 2 จากโปรแกรม ความพยายามนี้จะ

เหมือนกับไม่มีความหมายไม่มีค่าใด ๆ ที่จะนำไปใช้ได้ เหมือนเป็นข้อมูลขยะ แต่ระบบปฏิบัติการสามารถที่จะป้องกันปัญหาลักษณะนี้ได้ ถ้าระบบปฏิบัติการสามารถบอกกับไฟล์ได้ว่าข้อมูลที่ป้อนเข้ามานั้นเป็นเลขฐาน 2 การดำเนินการกับไฟล์จะดำเนินการต่อไปได้ แต่ถ้าระบบปฏิบัติการบอกไม่ได้ว่าเป็นเลขฐานสองข้อมูลที่พิมพ์เข้ามาเป็นเลขฐานสองจะถูกปฏิเสธ

สำหรับการจัดเก็บไฟล์ในรูปแบบอื่น ๆ การจัดเก็บหรือสนับสนุนจะไม่ใช่ชนิดของไฟล์ในระบบปฏิบัติการ ซึ่งได้แก่การทำงานของระบบปฏิบัติการ UNIX โดยที่ UNIX จะพิจารณาไฟล์อื่น ๆ ที่เป็นส่วนของ 8 บิต ระบบปฏิบัติการอื่นไม่แปลความหมายของบิตเหล่านี้เป็นไฟล์ ระบบปฏิบัติการ UNIX จะนิยามไฟล์ทั้งหมดเป็นสายของข้อมูล (Stream)

โครงสร้าง ไคเร็กทอรี

ไฟล์ในระบบคอมพิวเตอร์ จะแสดงรายการใน Device Directory หรือ Volume Table ของสารบัญใน Device Directory จะบันทึกข้อมูล เช่น ชื่อ ที่ตั้ง ขนาด และชนิด ของไฟล์

Device Directory อาจเพียงพอสำหรับผู้ใช้งานเพียงคนเดียว และจำกัดที่ว่างของการเก็บข้อมูล จำนวนของการเก็บข้อมูลและตัวเลขของผู้ใช้เพิ่มขึ้น อย่างไรก็ตามมันจะกลายเป็นการเพิ่มขึ้นที่ยากของการจัดเก็บ track ของไฟล์ทั้งหมด การแก้ไขปัญหานี้ คือ การจัดการโครงสร้างของ ไคเร็กทอรีบนระบบแฟ้ม โครงสร้าง ไคเร็กทอรี จะป้องกัน การจัดการไฟล์ในระบบแฟ้ม มันจะหมุน device กลับไปที่ขอบเขตของมันและรวบรวมดิสก์ที่แตกต่างกัน หรือหมุนดิสก์ที่คอมพิวเตอร์มองเห็นแตกต่าง ซึ่งวิธีนี้ผู้ใช้ต้องการให้เกี่ยวเนื่องกับ Logical Directory เท่านั้น และเกี่ยวข้องกับโครงสร้างแฟ้มเท่านั้นด้วยเช่นกัน

การจัดเก็บข้อมูลต่าง ๆ ของไฟล์ ในไคเร็กทอรี ในไคเร็กทอรี ไม่ได้จัดเก็บตำแหน่งของข้อมูลในไฟล์ทั้งหมดจะระบุตำแหน่งที่จัดเก็บตำแหน่งแรกเท่านั้น ระบบปฏิบัติการจะรับรู้ตำแหน่งที่จัดเก็บอื่น ๆ ด้วยการ Map จากตำแหน่งแรกของไคเร็กทอรีไปยังตารางการจัดเก็บข้อมูลของไฟล์ โครงสร้างของไคเร็กทอรี ประกอบด้วยส่วนต่าง ๆ ดังนี้

- ชื่อไฟล์ เป็นเอกลักษณ์พิเศษของไฟล์ เก็บข้อมูลไว้ใน human-readable form เท่านั้น
- ชนิดไฟล์ เป็นข้อมูลต้องการให้ระบบสนับสนุน ไฟล์ที่แตกต่างกัน
- ตำแหน่งที่เก็บ ตำแหน่งแรกของไฟล์ที่จัดเก็บ มีกรณีนี้ ซึ่งไปยังตำแหน่งในอุปกรณ์ที่จัดเก็บ
- ขนาด ขนาดปัจจุบันของไฟล์ มีขนาดเป็นไบต์ เวิร์ด หรือบล็อก
- ตำแหน่งปัจจุบัน ตัวชี้ (Pointer) ซึ่งตำแหน่งที่ จะอ่านหรือเขียนตำแหน่งปัจจุบันในไฟล์
- การป้องกัน การควบคุมทางเข้าของข้อมูลที่จะเข้า การอ่าน การเขียน และการประมวลผล เป็นต้น

- ตัวนับ จะบ่งชี้จำนวนของกระบวนการที่ใช้ในปัจจุบัน ที่เปิดอยู่ของไฟล์
- เวลา วันที่ และการแสดงกระบวนการ เป็นข้อมูลที่อาจจะเก็บค่า สำหรับการสร้าง สำหรับแก้ไขและสำหรับการใช้ครั้งสุดท้ายหรือล่าสุด ข้อมูลสามารถที่จะป้องกันและนับได้

ข้อมูลของไฟล์อาจจะใช้ตั้งแต่ 16 ถึงมากกว่า 1,000 ไบต์ ต่อเรคคอร์ด ในระบบ ประกอบด้วยตัวเลข ขนาดของ device directory อาจต้องการเก็บ device และนำหน่วยความจำ แบ่งเป็นส่วน ๆ ตามต้องการ

ระบบไดเรกทอรี (Directory System) จะมองสัญลักษณ์ต่าง ๆ แล้วแปลชื่อไว้ในไดเรกทอรี นั้น ถ้ามันแน่ใจไดเรกทอรีจะจัดการด้วยตัวมันเองตามวิธีของมัน เราสามารถที่จะแทรกหรือลบ หรือค้นหาชื่อและแสดงรายละเอียดใน ไดเรกทอรี ได้ ถ้าเราเลือกโครงสร้างข้อมูลที่เหมาะสมแล้วก็สามารถที่จะใช้อุปกรณ์ของโครงสร้างไดเรกทอรี (Directory Structure) ได้

ข้อเสียของไดเรกทอรี คือ linear ค้นหาในไฟล์นั้นแล้วรายการ (List) จำนวนนั้นอนุญาตให้ ค้นหาแบบการแบ่งครึ่ง (Binary Search) และลดค่าเฉลี่ยเวลาในการค้นหา แต่อย่างไรก็ตาม การ ค้นหา ของอัลกอริทึมจะซับซ้อนมากกว่า นอกจากนี้ List ต้องเก็บชนิดนั้น ๆ ด้วย ถ้าต้องการที่จะ compile สร้างและลบไฟล์เราจะต้องย้ายไดเรกทอรีที่เป็นหลักไปเก็บข้อมูลไว้ใน ไดเรกทอรี

ตัวอย่าง เช่น สมมติว่าเราใช้ตารางที่ซับซ้อนของ 64 ข้อมูล หน้าที่ซับซ้อนจะเปลี่ยนชื่อ ไฟล์เป็นตัวเลขจาก 0 ถึง 63 บางที่จะใช้วิธีการรวม 64 เข้าไปด้วยจะได้เป็น 65 ไฟล์ เราต้องจัดการ กับ directory entries ที่มีอยู่ เป็นหน้าที่ของฟังก์ชัน hash ที่จะดำเนินการให้ได้ค่าตำแหน่งจัดเก็บ ใหม่

การปฏิบัติการไฟล์

ไฟล์เป็นข้อมูลชนิดที่เป็นนามธรรม และมีคำนิยามของไฟล์ได้อย่างเหมาะสม เราต้อง พิจารณาการกระทำบนไฟล์ ระบบปฏิบัติการได้ป้องกันการทำงานบนไฟล์โดยใช้โปรแกรมเรียก ระบบ (System Calls) ในการดำเนินการ สร้าง เขียน อ่าน เริ่มต้นใหม่ และลบไฟล์ การพิจารณาว่า ระบบปฏิบัติการจะต้องดำเนินการพื้นฐานอะไรบ้างกับไฟล์ โดยการดำเนินการกับไฟล์ที่ผ่านการ ควบคุมของโปรแกรมเรียกระบบ มีดังต่อไปนี้

การสร้างไฟล์

มี 2 ขั้นตอน ที่จำเป็นต้องใช้ในการสร้างไฟล์ ประการแรกหาช่องว่างในระบบสำหรับ จัดเก็บไฟล์ให้พบก่อน ประการที่สอง คือหาส่วนที่ใช้สำหรับเก็บไฟล์ คือไดเรกทอรี จะบันทึกชื่อ ไฟล์และตำแหน่งแรกที่จัดเก็บไว้ในไดเรกทอรี

การเขียนไฟล์

การเขียนไฟล์เราใช้ system call ที่เป็นคำสั่งพิเศษของระบบปฏิบัติการ เป็นผู้ดำเนินการ ทั้งชื่อ และข้อมูลที่จะเขียนลงไฟล์ โดยที่ชื่อนั้น ระบบจะค้นหาจาก ไดรฟ์ทอรี เพื่อให้พบที่ตั้งของไฟล์ ส่วน directory entry จะต้องเก็บข้อมูลจาก pointer ที่ชี้อยู่ที่ block ปัจจุบันของไฟล์นั้น ๆ (เริ่มจากต้นไฟล์) การใช้ pointer จะต้องคำนวณ address ของ block ถัดไปและเขียนข้อมูล pointer ที่เขียนต้อง updated ไปด้วย

การอ่านไฟล์

การอ่านไฟล์เราใช้ system call ที่เป็นคำสั่งพิเศษของระบบปฏิบัติการซึ่งอยู่ในหน่วยความจำหลัก อ่านจากชื่อไฟล์ใน block ถัดไปของไฟล์ที่ถูกจัดเก็บ จากนั้น ไดรฟ์ทอรี ถูกค้นหา directory entry ที่เชื่อมโยงกับไดรฟ์ทอรี จะให้ pointer ชี้ไปที่ block ถัดไปจาก block ที่ถูกอ่าน pointer จะถูกปรับเปลี่ยนไปด้วย

การ ย้ายตำแหน่งภายในไฟล์

ไดรฟ์ทอรี จะค้นหา entry ที่ต้องการ และตำแหน่งไฟล์ปัจจุบัน เพื่อทำการ reset ค่าเริ่มต้นของไฟล์ การ Reset ไฟล์ที่ไม่จำเป็นต้องเกี่ยวข้องกับ I/O อย่างแท้จริง

การลบไฟล์

การลบไฟล์ เราจะค้นหา ชื่อไฟล์ จาก ไดรฟ์ทอรี หากพบ directory entry ที่ต้องการจะดำเนินการแทนที่ไฟล์นั้นให้เป็นไฟล์ว่าง และจะถือว่า ไฟล์เดิมใน directory entry นั้นถูกยกเลิกไป

เราอาจตั้งข้อสังเกตได้ว่า การ กระทำนั้นคือการค้นหา ไดรฟ์ทอรี เพื่อจะได้ลบชื่อไฟล์ที่ไม่ต้องการออกและบรรจุ ข้อมูลสำคัญที่ต้องการลงไป ในไฟล์นั้น การค้นหาจะเปิดไฟล์เมื่อไฟล์แรก เป็นกิจกรรมที่ต้องการใช้และระบบปฏิบัติการจะเก็บข้อมูลที่เกี่ยวกับไฟล์ที่เปิดนั้นไว้ เมื่อการดำเนินการไฟล์ (File Operation) ต้องการก็จะแสดงเป็นตารางออกมาใช้ แต่จะไม่ค้นหาตามต้องการ และทำนองเดียวกัน จะปิดและเคลื่อนย้าย จากตารางที่เปิดไฟล์อยู่

การกระทำกับไฟล์ที่กล่าวมา ครอบคลุมถึง ความต้องการของการดำเนินการกับไฟล์ โดยปกติแล้วเราต้องการ edit ไฟล์ และ modify ตามที่กำหนด การ modify จะเป็นการเพิ่มเติมข้อมูลใหม่ เข้าไป หรือทำให้ว่าง เราอาจต้องการสร้างหรือ copy ไฟล์ โดยเรียกใช้ จาก I/O device เช่น Hard Disk Printer หรือ display เราสามารถทำการเปลี่ยนชื่อไฟล์ที่มีอยู่ได้ เป็นต้น

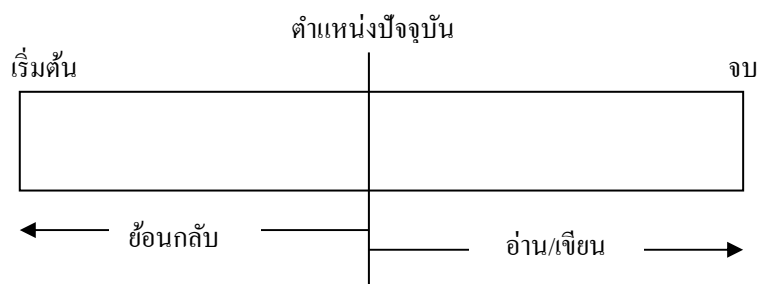
วิธีการแอกเซสไฟล์

ไฟล์เก็บข้อมูล เมื่อจะใช้ข้อมูลต้องทำการ access และอ่านจาก หน่วยความจำ มีหลายวิธีที่ข้อมูลในไฟล์สามารถที่จะ access ได้ บางระบบจะป้องกันวิธีการ access ไฟล์ไว้เพียงอย่างเดียว

เช่น IBM จะมีวิธีการ access ที่แตกต่างจากบริษัทอื่น คือ จะสนับสนุนและเลือกใช้ตามแอปพลิเคชัน โปรแกรม วิธีการ Access ไฟล์ มีหลายวิธีคือ

การ Access ไฟล์แบบ Sequential

การดำเนินการของ Sequential file จะทำตามข้อมูลในไฟล์ ตามลำดับที่ละเรคคอร์ด การกระทำบนไฟล์ทั้งมีทั้งอ่านและเขียน ในการดำเนินการอ่านจะอ่านในส่วนของ file ในตำแหน่งที่ pointer ชี้อยู่เมื่ออ่านแล้วจะเลื่อน pointer ไปยังตำแหน่งถัดไปโดยอัตโนมัติ จนกระทั่งจบไฟล์ และการเขียนไฟล์จะทำในทำนองเดียวกันกับการอ่าน แต่เมื่อจบไฟล์ต้องบันทึกจุดสิ้นสุดของไฟล์ลงไป ด้วย ไฟล์ที่ reset จะเริ่มที่ต้นไฟล์และข้าม forward หรือ backward โดยต้องผ่าน record ที่อยู่ก่อนเสมอ ดังรูปที่ 10.3



รูปที่ 10.3 แสดงการเรียกใช้ Sequential ไฟล์

การแอกเซสไฟล์แบบตรง

การแอกเซสไฟล์แบบตรง (Direct – access File) จะทำการเข้าถึงข้อมูลในไฟล์โดยตรง เป็นไฟล์ที่มีข้อมูลขนาดใหญ่ และข้อมูลที่ต้องการให้ประมวลผลได้ตอบทันที จึงเหมาะกับแอกเซสไฟล์ชนิดนี้ ซึ่งข้อมูลต่าง ๆ จะถูกเก็บไว้เป็นบล็อก ๆ ตามชนิดและประเภทของข้อมูลทั้งยังมีการป้องกันบล็อกที่อ่านไปแล้วโดยตรง

วิธีการแอกเซสไฟล์ ด้วยวิธีอื่น ๆ

วิธีการ Access ด้วยวิธีอื่น สามารถสร้างจากวิธีของ direct – access และการเพิ่มโครงสร้างของดัชนีเข้าไป วิธีนี้ ดัชนี จะทำหน้าที่เหมือนกับดัชนีของหนังสือทั่ว ๆ ไป ภายในดัชนีบรรจุ pointer และ block ต่าง ๆ เมื่อพบว่าต้องการหารายการ(entry) ใดในไฟล์แล้ว จะค้นหาจากดัชนี และใช้ pointer ชี้ไปยังไฟล์ที่ดำเนินการ(Access File) และค้นหารายการที่ต้องการ

ตัวอย่าง เช่น ราคาขายปลีกอาจจะเป็น code เป็น items อาจจะเป็นเลข 6 หลัก หรือใช้ 16 ไบต์ ถ้าดัชนีมี 1024 ไบต์/บล็อก จะเป็น 64 entry/บล็อก การค้นหาราคาในส่วนของ Item เราสามารถดำเนินการ (Access) ข้อมูลแบบ sequential เทียบกับการ access แบบ direct ได้ ดังรูป 10.4

จะเห็นว่า การ Access แบบ Sequential ระบบจะเลื่อนตัวชี้ตำแหน่ง record ถัดไปให้โดยอัตโนมัติ
 ส่วนการ Access แบบ Direct จะต้องมีการสั่งในการเลื่อนตัวชี้ตำแหน่ง record

Sequential access	direct access
Reset	cp := 0
Read next	read cp; cp := cp + 1;
Write next	write cp; cp := cp + 1;

รูปที่ 10.4 แสดงการจำลอง การ access แบบ sequential เทียบกับ การ access แบบ direct

กฎเกณฑ์ความคงสภาพของไฟล์

กฎเกณฑ์ความคงสภาพของไฟล์เป็นส่วนสำคัญในการประเมินตามมาตรฐานของ
 แฟ้มข้อมูลที่สนับสนุนการใช้ไฟล์ร่วมกัน มันเป็นลักษณะพิเศษของระบบกฎเกณฑ์ มีการจัดการ
 พิเศษของ ผู้ใช้ ที่หลากหลายได้ เข้ามา access ไฟล์ ในส่วนของกฎเกณฑ์จะดำเนินการเป็นพิเศษคือ
 เมื่อแก้ไขข้อมูลของ ผู้ใช้ เพียงคนหนึ่งแล้ว ผู้ใช้ คนอื่นที่ใช้ไฟล์ร่วมกันต้องสามารถเห็นสิ่งที่แก้ไขได้
 การจัดการอาจสมมติเป็น Series ของการ access ไฟล์ ที่ถูกอ่านและเขียน โดยผู้ใช้ที่จะพยายามใช้
 แต่อยู่ระหว่างเปิดและปิด ต้องมีการกระทำบนไฟล์เป็น session คือผู้ใช้ที่กำลังแก้ไขอยู่ใน session
 เปิดของไฟล์ผู้ใช้คนอื่นที่ใช้ไฟล์เดียวกันจะอยู่ใน session ปิดของไฟล์

กฎเกณฑ์ของ UNIX

ระบบแฟ้มของ UNIX ใช้พิจารณา กฎเกณฑ์ ดังนี้

- เขียนเมื่อเปิด โปรแกรมโดยผู้ใช้ที่มีอุปกรณ์ที่มองเห็นได้ไปหาผู้ใช้ที่เปิดไฟล์อยู่
 ขณะนั้น

- มี mode ที่ใช้แบ่งผู้ใช้โดย pointer ปัจจุบันเป็นตัวแบ่ง คือ ข้อดีของ pointer ที่ใช้ผู้ใช้
 เพียง 1 คน มีประสิทธิภาพมากกว่าการ share ผู้ใช้หลายคน มีกฎเกณฑ์ให้นำเครื่องมือของมันมาใช้
 กับไฟล์ได้อย่างเหมาะสมกับทรัพยากร

กฎเกณฑ์ของเซสชัน (Session Semantics)

ระบบแฟ้มตามกฎเกณฑ์ของเซสชัน พิจารณา ดังนี้

- เขียนเมื่อไฟล์เปิดโดยที่ ผู้ใช้ ไม่จำเป็นต้องมีอุปกรณ์ไปยัง ผู้ใช้ ที่เปิดไฟล์นั้นอยู่
- ในขณะที่ไฟล์เปิดจะมีการเปลี่ยน mode ให้มองเห็นเพียง sessions เท่านั้น

กฎเกณฑ์ที่สำคัญของใช้ไฟล์ร่วมกัน (Immutable – Shared – File Semantics)

ความแตกต่างที่ไม่เหมือนกับวิธีอื่นคือการใช้ไฟล์ร่วมกัน ซึ่ง file จะถูกประกาศไว้โดยผู้สร้างมันไม่สามารถที่จะแก้ไขสิ่งใด ๆ ได้เลย ไฟล์ที่ไม่เปลี่ยนแปลง (Immutable) แบ่งเป็น 2 ส่วนสำคัญคือ ชื่อจะต้องไม่เคยใช้ และเนื้อหาจะต้องไม่เปลี่ยนแปลง ดังนั้นชื่อจะต้องมีความหมายที่สื่อถึงเนื้อหาของไฟล์

การจัดการ โครงสร้างไดเรกทอรี

ไฟล์ไดเรกทอรี จะต้องแสดงรายการให้แก่ผู้ใช้ ผู้ใช้รู้ว่ามีการใดบ้าง ไดเรกทอรีจำเป็นต้องใช้ตารางสัญลักษณ์ โดยที่ระบบปฏิบัติการจะนำชื่อไฟล์ สัญลักษณ์พิเศษ ไปค้นหาชื่อไฟล์จากไดเรกทอรี เพื่อพิจารณาจากในส่วนของโครงสร้างไดเรกทอรี การปฏิบัติการบนไดเรกทอรี มีดังนี้

- ค้นหา ต้องสามารถค้นหาไดเรกทอรีได้ทั้งชื่อที่เป็นสัญลักษณ์พิเศษและชื่อที่สัมพันธ์กับไฟล์ , ต้องสามารถพบไฟล์ทั้งหมดที่จับคู่ได้ดังรูปแบบ

- สร้างไฟล์ ไฟล์ใหม่ที่ถูกสร้างขึ้นหรือไฟล์ที่ต้องการจะสร้างไฟล์ใหม่และต้องการเพิ่มเข้าไปใน ไดเรกทอรี นั้น

- ลบไฟล์ เมื่อไฟล์นั้นไม่ต้องการเราต้องลบออกจาก ไดเรกทอรี

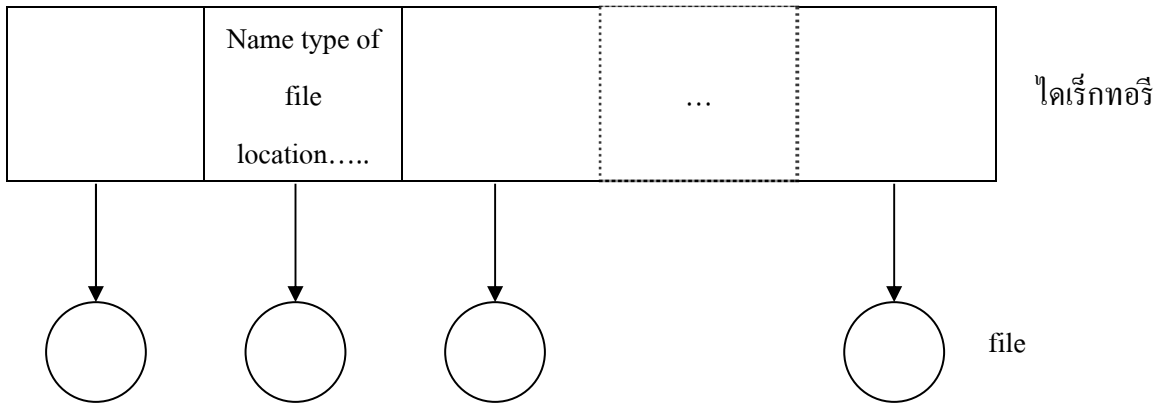
- แสดงรายละเอียดของ ไดเรกทอรี ต้องแสดงรายละเอียดของไฟล์ใน ไดเรกทอรี และเนื้อหาของ ไดเรกทอรี ของไฟล์อื่นๆ ที่แสดงขึ้นมาบน list

- ไฟล์สำรอง เป็นไฟล์ที่บันทึกไว้สำรองหรืออาจจะเรียกว่าเป็นการคัดลอกไฟล์เก็บไว้ในเทปก็ได้ ซึ่งเทคนิคนี้ถือว่าการป้องกันการทำให้ไฟล์สำรองไว้ในแคชของระบบ

ไดเรกทอรีชั้นเดียว

โครงสร้างไดเรกทอรีเป็นไดเรกทอรีเดี่ยว ๆ ซึ่งอุปกรณ์ไดเรกทอรีเป็นตัวอย่างของไดเรกทอรีชั้นเดียว ไฟล์ทั้งหมดที่มีอยู่ได้ถูกบรรจุไว้ใน ไดเรกทอรี ซึ่งเป็นการง่ายที่จะสนับสนุนและเข้าใจ ดังรูปที่ 10.5

ไดเรกทอรี ชั้นเดียวมีคำจำกัดความพิเศษเป็นเครื่องหมายแต่เมื่อจำนวนของไฟล์เพิ่มขึ้น หรือมี ผู้ใช้ ใช้มากกว่าหนึ่งไฟล์ทั้งหมดจะเป็นเหมือนกัน ไดเรกทอรีจะต้องมีชื่อที่ไม่ซ้ำกัน ถ้ามี ผู้ใช้ 2 คน เรียกใช้ไฟล์ข้อมูลในไฟล์ที่ชื่อ test แล้วชื่อไม่เหมือนกันถือว่าการละเมิด เช่น ในโปรแกรม class มีนักเรียนอยู่ 20 คนเรียกโปรแกรมนี้เป็นชั้นที่ 2 จะต้องมีโปรแกรมอยู่ 2 โปรแกรม อย่างไรก็ตามชื่อไฟล์โดยปกติแล้วถูกเลือกให้มีเนื้อหาครอบคลุมไฟล์ มีการจำกัดความยาวด้วย เช่น MS – DOS อนุญาตให้ตั้งชื่อไฟล์ได้ไม่เกิน 11 ตัวอักษร แต่ใน UNIX อนุญาตให้ใช้ได้ 255 ตัวอักษร



รูปที่ 10.5 แสดงไดเรกทอรีชั้นเดียว (Single – level directory)

ไดเรกทอรี 2 ชั้น

ข้อเสียหลักๆ ของ ไดเรกทอรี ชั้นเดียวคือ จะทำให้สับสนในเรื่องของไฟล์ระหว่าง ผู้ใช้หลายคน การแก้ปัญหามาตรฐานคือสร้างไดเรกทอรีแยกออกมาอีกสำหรับผู้ใช้อื่น ๆ ซึ่งการจัดการต่าง ๆ จะต้องเป็นเชิงตรรกะ (logical) จึงดีกว่ากายภาพ (physical)

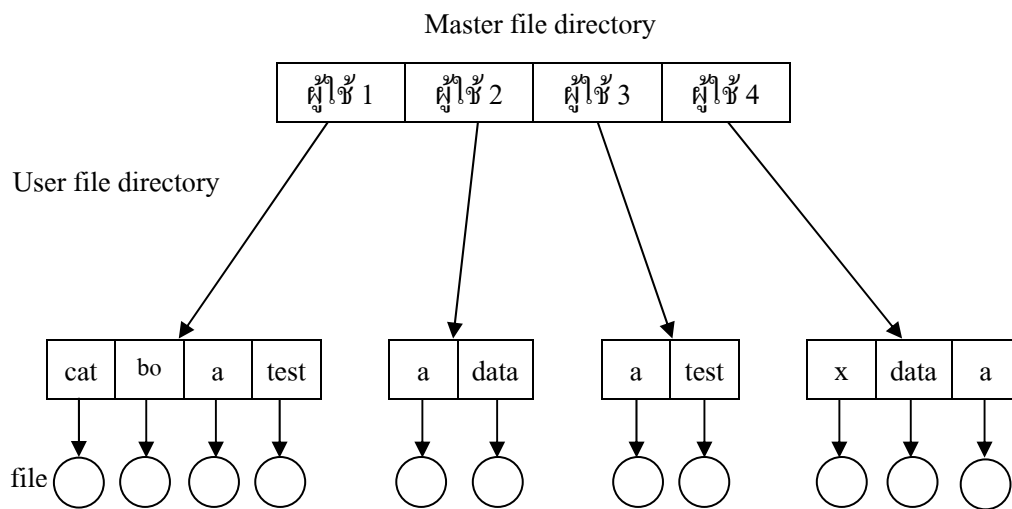
ในไดเรกทอรี 2 ชั้นนี้ ผู้ใช้แต่ละคนจะต้องมีไฟล์ ไดเรกทอรี เป็นของตนเอง มีลักษณะคล้ายๆ กับโครงสร้าง linear binary หรือ hashes ซึ่งแต่ละรายละเอียดของไฟล์จะต้องเป็นเหมือนกับ ผู้ใช้ เพียงคนเดียว เมื่อ ผู้ใช้ เริ่มทำงานตัว ระบบในไฟล์ไดเรกทอรีหลัก (Master File Directory: MFD) จะถูกค้นหา ซึ่งถ้า ผู้ใช้ คนอื่นมีไฟล์ที่มีชื่อเหมือนกับของผู้ใช้ก็จะแสดงชื่อไฟล์นั้นภายในไดเรกทอรีของผู้ใช้ (User File Directory: UFD) เพื่อให้ทราบว่าเป็นไฟล์ของผู้ใช้เอง ไม่ใช่ไฟล์เดียวกันกับไดเรกทอรีอื่น

การสร้างไฟล์ของผู้ใช้ระบบปฏิบัติการจะค้นหาที่ไดเรกทอรีของผู้ใช้เท่านั้น หากไฟล์ที่ผู้อื่นมีชื่อนี้แล้วก็จะแจ้งให้ทราบและยอมทำตามคำสั่ง ส่วนการลบไฟล์ระบบปฏิบัติการจะค้นหาที่ส่วนของ UFD ซึ่งมันไม่ได้เป็นการบังเอิญที่จะลบไฟล์ของผู้ใช้คนอื่นที่มีชื่อเหมือนกัน ไดเรกทอรีของผู้ใช้ต้องสร้างและลบอย่างจำเป็น โปรแกรมระบบพิเศษจะ run ชื่อหรือเลขที่บัญชีของ ผู้ใช้ ที่ใกล้เคียงมาให้ทราบด้วย โปรแกรมสร้างไฟล์ ผู้ใช้ ใหม่และเพิ่ม entry เข้าไปที่ master Directory โปรแกรมที่กำลังทำการ execute อยู่อาจต้องถูกจำกัดพื้นที่ดิสก์ที่ว่างอยู่นั้น ผู้ใช้ สามารถที่จะใช้ได้ตามหลักหรือเทคนิคที่มีอยู่

การเข้าถึงไฟล์ในไดเรกทอรี เช่น ถ้าต้องการเข้าถึงไฟล์ test ของผู้ใช้ 1 ในการดำเนินการ ชั้นเริ่มต้นจาก การค้นหาไดเรกทอรีของผู้ใช้ 1 ใน MFD เมื่อพบจึงเข้าดำเนินการต่อในไดเรกทอรีของผู้ใช้

ใช้ 1 แล้วทำการค้นหาไฟล์ test เมื่อพบจึงใช้งาน จะไม่เกี่ยวกับไฟล์ test ในไดเรกทอรีของผู้ใช้ 3 ที่มีชื่อไฟล์เหมือนกัน ดังรูปที่ 10.6

ปัญหาของไดเรกทอรี 2 ชั้น คือประสิทธิภาพที่แยกระหว่าง ผู้ใช้ 1 คน กับผู้ใช้คนอื่นๆ ข้อดีคือ ผู้ใช้เป็นอิสระ ข้อเสีย คือ ถ้าผู้ใช้ต้องการแยกที่จะได้ตอบงานและ access file กับผู้ใช้คนอื่นๆ ไม่ได้



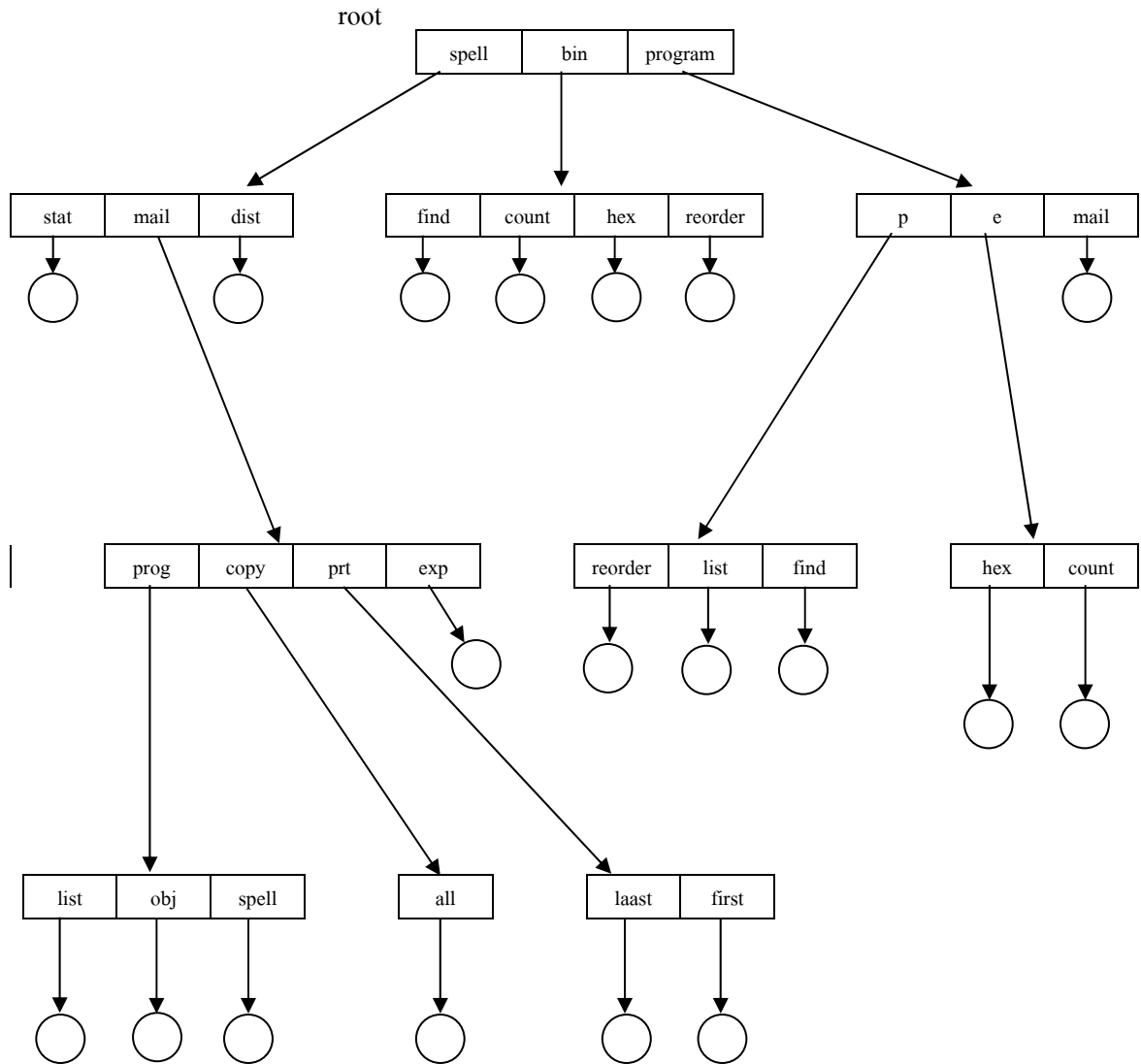
รูปที่ 10.6 แสดง โครงสร้าง ไดเรกทอรี แบบ 2 ชั้น (Two – Level directory structure)

ไดเรกทอรีโครงสร้างต้นไม้

การที่เราจะมองเห็นภาพของ ไดเรกทอรี 2 ชั้น อย่างไรนั้น ก็เปรียบเสมือนกับที่เราได้มองเห็นโครงสร้าง ไดเรกทอรี ต้นไม้แล้วเช่นกันเพราะเปรียบได้กับโครงสร้างต่างๆ ไปของต้นไม้สูงๆ ที่มีกิ่ง ก้าน ใบ แตกออกมาจากลำต้น (ดูรูป ที่ 10.7) โดยใน ไดเรกทอรี นี้จะอนุญาตให้ผู้สร้าง Sub Directory ของตัวเองและการจัดการไฟล์ตามที่ต้องการ ในระบบของ MS-DOS จะมีโครงสร้าง ต้นไม้นี้อยู่โดยที่รากของต้นไม้เปรียบได้กับ root ของไดเรกทอรีทุก ๆ ไฟล์ ในระบบจะต้องมีชื่อที่ไม่ซ้ำกัน Path ของชื่อต้องเป็น Path ที่ผ่านมาจาก Sub Directory ก็ได้

ไดเรกทอรีย่อยหรือ Sub Directory บรรจุทั้ง ไฟล์และ Sub Directory ไว้ และไดเรกทอรีทั้งหมด มีโครงสร้างรูปแบบภายในที่คล้าย ๆ กัน สามารถที่จะสร้างหรือลบไดเรกทอรี ได้เช่นกัน

โดยปกติผู้ใช้จะต้องมีไดเรกทอรีปัจจุบันเพื่อเก็บไฟล์ต่าง ๆ ไว้ภายในไดเรกทอรีของผู้ใช้



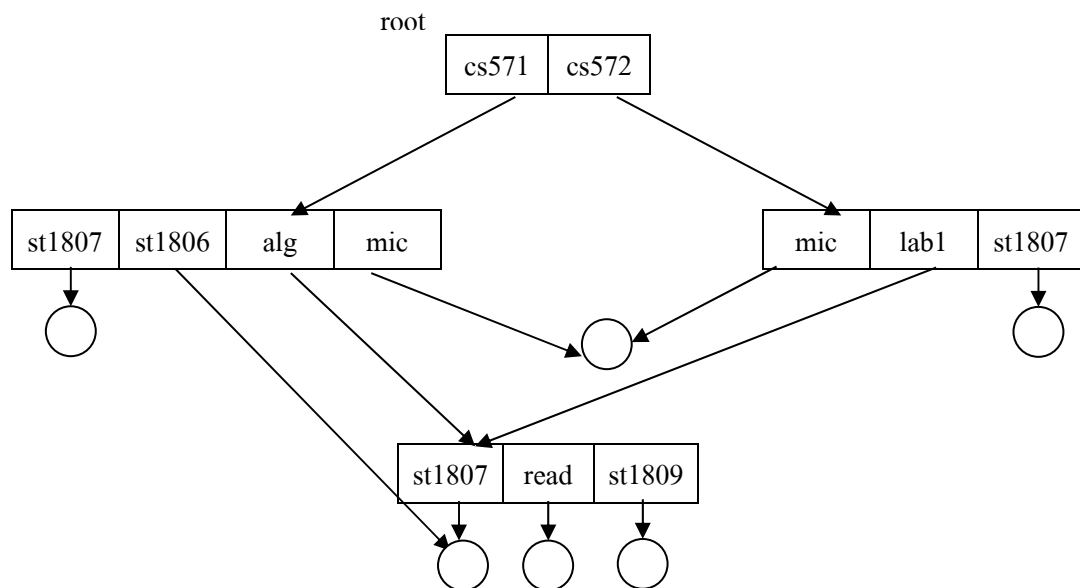
รูปที่ 10.7 แสดงโครงสร้าง ตรีโกทรีแบบต้นไม้

เมื่อต้องการที่จะอ้างถึงไฟล์ใดเรียกทอรีปัจจุบันจะทำหน้าที่เป็นผู้ค้นหา ถ้าพบว่าไฟล์ที่ต้องการไม่ได้อยู่ในไคเรียกทอรีปัจจุบันแล้ว ผู้ใช้ต้องเปลี่ยนชื่อหรือเปลี่ยน path ใหม่ให้กับไคเรียกทอรี เมื่อผู้ใช้เริ่มทำงานระบบปฏิบัติการก็จะเริ่มค้นหาเลขที่ไฟล์ที่ผู้ใช้ต้องการค้นหานั้น โดยที่ข้อมูลที่เก็บไว้ภายในเลขที่ไฟล์จะถูก pointer เป็นตัวชี้อยู่ด้วย การกำหนด path ให้กับไฟล์ที่ต้องการเข้าถึง จะทำการระบุชื่อไคเรียกทอรีของแต่ละระดับ โดยเริ่มจาก root ค้นด้วย “\” จนถึงไคเรียกทอรีที่ไฟล์นั้นอยู่ เช่นจากรูปที่ 10.7 path ของ ไฟล์ first ซึ่งอยู่ในไคเรียกทอรีของผู้ใช้ prt คือ \spell\mail\prt

Acyclic – Graph ไคเร็กทอรี

จากการพิจารณาการทำงานของโปรแกรม 2 โปรแกรมซึ่งกำลังทำงานอยู่ใน Project ใด ๆ ไฟล์ต่าง ๆ ที่อยู่ใน Project สามารถที่จะเก็บข้อมูลจาก ไคเร็กทอรีย่อย (Sub Directory) และสามารถที่จะเก็บข้อมูลจาก Project อื่น ๆ ได้อีกทั้ง 2 โปรแกรม ซึ่งปกติแล้วไคเร็กทอรีย่อย จะต้อง share กับไคเร็กทอรีหรือไฟล์ที่กำลัง exist อยู่ในระบบแฟ้มใน 2 หรือ 1 แห่ง มีข้อสังเกต คือการ share file หรือ share ไคเร็กทอรีไม่ได้เป็นการคัดลอกไฟล์เป็น 2 สำเนาแต่อย่างใด ซึ่งผู้เขียนโปรแกรมกลับมองว่าเป็นการคัดลอกจากต้นฉบับ ถ้าผู้เขียนโปรแกรม เปลี่ยนแปลงไฟล์ การเปลี่ยนแปลงนั้นจะไม่ปรากฏให้เห็นในการคัดลอก เป็นข้อดีของการแชร์ที่สามารถเห็นข้อมูลที่ถูกแก้ไขแล้ว แต่ถ้าเป็นการคัดลอกจะไม่เห็นข้อมูลที่ถูกแก้ไขแล้ว

โครงสร้างต้นไม้จะห้ามไม่ให้มีการ share file หรือ share Directory แต่ acyclic – graph จะอนุญาตให้มีการ Share Directory กับ Sub Directory หรือกับไฟล์ได้ ดังรูปที่ 10.8 ไฟล์ที่เหมือนกันหรือ Sub Directory เหมือนกัน อาจจะมีได้ในทั้ง 2 ไคเร็กทอรี ที่มีชื่อต่างกัน ดังนั้น



รูปที่ 10.8 แสดงโครงสร้างไคเร็กทอรี แบบ Acyclic – Graph

ลักษณะของ acyclic – graph (ไม่ได้เป็นวงจร) เป็นการทำงานทั่ว ๆ ไปของโครงสร้าง ไคเร็กทอรี ซึ่งเป็นข้อดีของโครงสร้างไคเร็กทอรีแบบ acyclic – graph ที่มีความยืดหยุ่นกว่าโครงสร้างไคเร็กทอรีแบบต้นไม้

การ Share ไฟล์และ Sub Directory สามารถที่จะทำได้หลายทาง ทางหนึ่งก็นิยมทำกันใน Unix ก็คือ สร้าง ไคเร็กทอรี ขึ้นมาใหม่ เราเรียกว่า “link” การ link นี้เป็นการทำงานของ pointer ที่ชี้ไปยังไฟล์หรือ ไคเร็กทอรีย่อย เช่นอาจจะชี้ไปที่ที่มีความสัมพันธ์ระหว่าง path name เมื่อเรา

การป้องกันไฟล์

การป้องกันอาจทำได้หลายทาง เช่น ในกรณีที่เป็นระบบที่มี ผู้ใช้ ใช้คนเดียวอาจจะป้องกันได้จากการเคลื่อนย้ายลงไปไว้ใน floppy disks และ lock ไว้ในพื้นที่ส่วนเฉพาะของไฟล์ (cabinet file) แต่ถ้ามีผู้ใช้หลายคนในระบบอาจจะต้องมีองค์ประกอบหลาย ๆ อย่างที่จำเป็นต้องใช้อีก

การป้องกันอาจจะป้องกันได้จากทาง access ไฟล์ระบบจะปิดกั้นไม่ให้ access ไฟล์จากของผู้ใช้ คนอื่นที่ไม่ใช่ ผู้ใช้ เจ้าของ ดังนั้นการป้องกันทาง access อาจเป็นอีกทางหนึ่งที่สามารถช่วยป้องกันได้ แต่ถ้าหากว่าเป็นการป้องกันจาก access ที่ไม่มีการป้องกันเอาไว้ก็จะไม่สามารถที่จะป้องกัน โดยการ access ได้เลย ผู้ใช้คนอื่นสามารถเข้ามาใช้ access ได้เหมือนกันทุกคน

องค์ประกอบของการป้องกันและควบคุมการ access โดยการจำกัดชนิดของไฟล์ ที่จะเข้ามา access นั้นจะต้องถูกจำกัดหน้าที่ลักษณะทั่วไป ชนิดของการ access ที่ต้องการจะใช้เพื่อให้ปฏิบัติงานได้ถูกต้องตามที่กำหนดไว้แต่แรกโดยสามารถดำเนินการได้ต่อไปนี้

- อ่าน อ่านจากไฟล์
- เขียน เขียนลงไฟล์
- Execute โหลดไฟล์จากหน่วยความจำและทำการ execute
- เพิ่ม เป็นการเขียนข้อมูลใหม่และเก็บไว้เป็นไฟล์ล่าสุด
- ลบ จะลบไฟล์เพื่อทำให้เนื้อที่นั้นว่างแล้วนำเนื้อที่บริเวณนั้นกลับมาใช้ใหม่อีกครั้ง

ยังมีการทำงานอื่น ๆ อีก เช่น การเปลี่ยนชื่อ คัดลอกไฟล์ แก้ไขไฟล์ หรือการควบคุมของระบบ เป็นการป้องกันเบื้องต้นเท่านั้น ซึ่งผู้ใช้อื่นสามารถที่จะอ่าน access แล้วทำการคัดลอกหรือพิมพ์ไปได้ การป้องกันไฟล์มีวิธีการดังต่อไปนี้

Naming

การป้องกันชนิด Naming ก็คือ การที่ให้ผู้ ใช้ ที่จะเข้าไปใช้ access ระบุชื่อไฟล์ ถ้าผู้ใช้ไม่สามารถที่จะระบุได้ก็จะไม่อนุญาตให้เข้าไปทำงานภายในได้

Passwords

มีลักษณะเหมือนกับการใส่ passwords ทั่ว ๆ ไปนั่นคือ access จะถูกควบคุมด้วย password ถ้าไม่รู้ password ก็ไม่สามารถเข้าไปใช้ได้แต่อาจจะมีการสุ่ม password แล้วบังเอิญสุ่มถูกต้องก็สามารถที่จะเข้าไปใช้ได้ก็เช่นกันมีข้อเสียคือ

- ถ้าใช้ password แล้วลืมเพียงตัวใดตัวหนึ่งก็ไม่สามารถเข้าไปใช้งานใน access ได้
- จะต้องใช้ passwords ที่ไม่ซ้ำกันหลาย ๆ ไฟล์เพราะหากว่ามี ผู้ใช้ คนอื่นทราบ ก็จะสามารถทราบทุกไฟล์ที่เราได้ทำการป้องกันไว้ทำให้ ผู้ใช้ ผู้อื่นสามารถที่จะเข้าไปใช้ access ในส่วนนั้นได้

Access Lists

ผู้ใช้งานต่าง ๆ ต้องการให้ access ต่างชนิดกัน ดังนั้น access lists ถึงเหมาะสมและใกล้เคียงกับชนิดของไฟล์และไดเรกทอรีที่ผู้ใช้งานต้องการใช้ด้วย ถ้าผู้ใช้งานต้องการใช้ access ในส่วนของไฟล์ ระบบปฏิบัติการจะต้องตรวจสอบ access ที่ใกล้เคียงกับไฟล์นั้นด้วยและถ้าผู้ใช้งานที่ต้องการที่จะดูรายละเอียดของ access นั้นก็ต้องยอมให้ผู้ใช้งานได้

Access Group

ปัญหาหลักของ access list คือความยาว ถ้าต้องการที่จะอ่านไฟล์ของผู้ใช้ ทุกคนจะต้องมีรายละเอียดของผู้ใช้ทั้งหมดเก็บไว้ใน access ด้วย ซึ่งเทคนิคนี้อยู่ภายใต้กฎ 2 ข้อ คือ

- การจัดการให้ list ทำงานช้าและทำงานโดยมีการหยุดถ้าไม่พบสิ่งที่ต้องการค้นหา
- ไดเรกทอรี จะดูขนาดที่เหมาะสมหากขนาดใหญ่เกินไปจะต้องทำการ Compile ใหม่อีก

ครั้ง

เมื่อพิจารณาความยาวของ access list ระบบจะจัดการไฟล์โดย :

Owner จะเป็นผู้สร้างที่สามารถใช้ได้ทุกอย่าง

Group เมื่อต้องการสร้าง ผู้ใช้ จะต้องสร้างระบบ group ด้วยโดยอาจจะ access ข้อมูลได้เหมือนกับ owner

Universe ผู้ใช้ ทั้งหมดที่ไม่ได้อยู่ใน group เดียวกันกับ Owner

เมื่อจัดเป็นหมวดหมู่แล้วจะต้องมีการป้องกันทำได้โดยแบ่งฟิลด์ออกเป็น 3 บิต คือ rwx

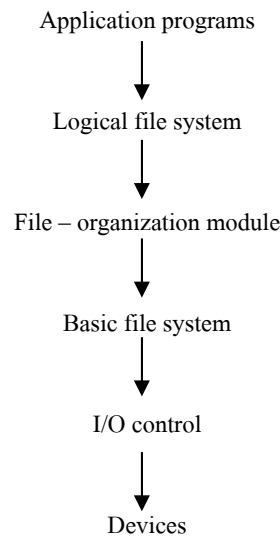
โดยที่

- | | |
|---|-----------------|
| r | ควบคุมการอ่าน |
| w | ควบคุมการเขียน |
| x | ควบคุมการใช้งาน |

access bit มี 9 บิตคือ rwx rwx rwx ทั้งในไฟล์และในการทำงาน

การใช้งาน

การดำเนินการบนไฟล์ประกอบด้วยหลาย ๆ Level โดยนำ path name ไปค้นหาในไดเรกทอรี เพื่อหา entry ของคุณสมบัติ ใช้ในการเปิดไฟล์ในโปรแกรมประยุกต์แล้วเก็บ path ไว้ในตารางเปิดไฟล์ในหน่วยความจำ ใช้ชื่อไฟล์มา access ข้อมูลที่เกี่ยวข้องกับไฟล์ในไดเรกทอรี ชนิดของไฟล์จะถูกอ่านโดย basic file system I/O เป็นผู้ส่งให้ Block อ่าน ดังรูปที่ 10.10



รูปที่ 10.10 แสดงระดับชั้นการดำเนินการไฟล์

การเปลี่ยนแปลงที่เกี่ยวข้องกับ Directory entry จะทำหน้าที่ คัดลอกและเมื่อเปิดไฟล์ระบบจะคัดลอกกลับไปเก็บที่ ไคเร็กทอรี ในดิสก์

หากเกิด System fail ก่อนที่จะมีการเขียนข้อมูลลงดิสก์ซึ่งจะทำให้ข้อมูลของระบบจะมีโปรแกรมพิเศษมา run ตอนที่ reboot เพื่อตรวจสอบและแก้ไข

การป้องกันและความปลอดภัย

หน้าที่สำคัญของระบบปฏิบัติการอีกหน้าที่หนึ่งคือการป้องกันและการรักษาความปลอดภัยของระบบการทำงานของคอมพิวเตอร์ ถึงแม้โปรแกรมการทำงานในส่วนนี้ไม่ได้เป็นส่วนหนึ่งของโปรแกรมระบบ (System Program) ทั้งหมดก็ตาม แต่ระบบปฏิบัติการส่วนใหญ่จำเป็นจะต้องมีการป้องกันและรักษาความปลอดภัยของระบบคอมพิวเตอร์ซึ่งเป็น โปรแกรมที่เพิ่มเติมในระบบปฏิบัติการทั่วไป โดยเฉพาะอย่างยิ่งระบบปฏิบัติการเครือข่าย (Network Operation System: NOS) เพื่อให้ระบบปฏิบัติการมีประสิทธิภาพการทำงานมากขึ้น สามารถแบ่งเป็น 2 ลักษณะคือ การป้องกัน (Protection) และการรักษาความปลอดภัย (Security) โดยที่การป้องกันจะเป็นโปรแกรมที่ติดตั้งเพื่อทำการควบคุมการดำเนินการที่เน้นเกี่ยวกับการป้องกันข้อมูลและโปรแกรมการทำงานต่าง ๆ ของผู้ใช้งาน เพื่อไม่ให้ผู้ใช้คนอื่น ๆ สามารถเข้าถึงข้อมูลต่าง ๆ ที่สำคัญจากระบบคอมพิวเตอร์ได้ ส่วนความปลอดภัยเป็นโปรแกรมที่ติดตั้ง เพื่อทำหน้าที่รักษาความปลอดภัยป้องกันข้อมูลและโปรแกรมต่าง ๆ ของผู้ใช้งานในระบบไม่ให้ถูกรั่วไหล หรือไม่ให้บุคคลภายนอกสามารถเจาะเข้ามาดูข้อมูลได้ ความแตกต่างระหว่างการติดตั้งระบบป้องกันและระบบความปลอดภัยของระบบปฏิบัติการมีข้อสังเกตได้ดังนี้ ระบบความปลอดภัยจะมีข้อจำกัดในการอนุญาต

ให้ผู้ใช้งานเข้ามาใช้โปรแกรมได้ตามที่อนุญาตให้มีสิทธิ์เข้าใช้ ถ้าเกิดมีบุคคลที่ไม่ได้รับสิทธิ์ อนุญาตให้เข้าใช้แล้วพยายามที่จะเข้าใช้โปรแกรมหรือข้อมูลเหล่านั้น ก็จะมีการแสดงข้อความ ขึ้นมาเตือน อย่างไรก็ตามระบบโปรแกรมป้องกันและรักษาความปลอดภัยมีหลายรูปแบบและหลายวิธีในการติดตั้ง ขึ้นกับลักษณะของระบบงานที่ใช้ ในการติดตั้งระบบโปรแกรมการป้องกัน และการรักษาความปลอดภัยของระบบปฏิบัติการนั้น ก่อนข้างจะยุ่งยากและซับซ้อน ที่ผู้ใช้ โปรแกรมภายนอกไม่สามารถเข้าไปติดตั้งหรือใช้โปรแกรมต่าง ๆ ได้เลย เนื่องจากการติดตั้งระบบ การรักษาความปลอดภัยของระบบปฏิบัติการนั้น จะมีขั้นตอนหลายขั้นตอนที่จะให้ผู้ใช้ โปรแกรมใส่ ชื่อ หรือรหัสส่วนตัว ก่อนเข้าไปใช้งานในโปรแกรมต่าง ๆ เพื่อเป็นการเก็บรักษา ข้อมูล ส่วนระบบการป้องกันจะมีการให้ใส่ ID ของผู้ใช้ก่อนจะเข้าไปดูโปรแกรมต่าง ๆ นั้นอาจจะ หมายถึงชื่อผู้ใช้งานหรือชื่อเฉพาะที่ถูกติดตั้งไว้ นอกจากนี้ยังมีค่าที่เป็นรหัสในการใช้เข้าในการ ติดตั้งระบบป้องกันและความปลอดภัย หมวดความลับ (Secrecy) และ ส่วนบุคคล (Privacy) หมวดความลับ จะยอมให้ผู้ใช้งานที่ได้รับอนุญาตในการติดตั้งเท่านั้นเข้ามาใช้งาน ส่วนบุคคล จะเป็นระบบโปรแกรมที่ติดตั้งขึ้นเพื่อตามวัตถุประสงค์ที่จะใช้งานเท่านั้น หรือต้องการจะ เผยแพร่ ข้อมูลต่างๆ ให้ผู้ใช้งานคนอื่นๆ ได้เห็น

ตัวอย่างเช่น ในการติดตั้งระบบปฏิบัติการเพื่อการทำงานสำหรับการป้องกัน โดยการ ใช้ ระบบป้องกันการ ใช้ข้อมูลสารสนเทศร่วมกัน (Sharing of information) สิ่งสำคัญต้องเข้ารหัสให้ ถูกต้องและถ้าต้องการแสดงข้อมูลต่างๆ ให้ ผู้ใช้คนอื่นเห็น ต้องทำการติดตั้ง โปรแกรมควบคุมการ ใช้ไฟล์ร่วมกัน (controlled sharing file) โดยการ ใช้ อย่างแรกต้องดูว่าต้องการให้ใครบ้างที่จะเห็น ไฟล์ข้อมูล และข้อมูลอะไรบ้างที่ต้องการแสดง โดยมีข้อกำหนด ดังตาราง ที่ 10.1

จากตารางที่ 10.1 ความปลอดภัยข้อมูล มีนโยบายให้ดำเนินการลงรายชื่อสำหรับผู้ที่ จะเข้า ใช้งาน และกลวิธีในการใช้ ให้ผู้ใช้สามารถ เพิ่มหรือลบข้อมูลได้ และทำการติดตั้งชื่อผู้ที่สามารถ เข้าไปใช้งานได้ สำหรับระบบป้องกัน มีนโยบายให้ระบุและกำหนดผู้ที่สามารถเข้ามาใช้ข้อมูล

ตารางที่ 10.1 นโยบายและกลวิธีในความปลอดภัยและการป้องกัน

การควบคุม	นโยบาย (Policy)	กลวิธี (Mechanisms)
ความปลอดภัยข้อมูล (Security)	ให้ลงรายชื่อสำหรับผู้ที่ จะเข้า ใช้งาน	1. เพิ่มหรือลบข้อมูล 2. ติดตั้งชื่อผู้ที่สามารถเข้าไปใช้งานได้
ระบบป้องกัน (Protection)	ให้ระบุและกำหนดผู้ที่ สามารถเข้ามาดูข้อมูลได้	1. ติดตั้งหรือเปลี่ยนข้อมูล 2. เลือกไฟล์ ที่จะแสดง

มีกลวิธีในการใช้ โดยให้ผู้ใช้สามารถ เพิ่มข้อมูลหรือเปลี่ยนแปลงแก้ไขข้อมูลได้ และสามารถเลือกไฟล์ที่จะดำเนินการได้

สำหรับนโยบายและกลไกของระบบการป้องกันและความปลอดภัย จากตารางที่ 10.1 แสดงให้เห็นถึงวิธีการใช้และการติดตั้งระบบการป้องกันและความปลอดภัยข้อมูล ซึ่งสามารถนำไปใช้ติดตั้งให้ผู้ใช้สามารถใช้ได้ ผู้ที่เข้ามาใช้โปรแกรมโดยปกตินั้นจะมีการให้ผู้ใช้สร้างไฟล์ส่วนตัวขึ้นมาเพื่อที่จะสามารถเข้าไปทำงานได้ รวมถึงระบบการป้องกันและความปลอดภัยจะคอยเตือนระหว่างการใช้งาน โปรแกรมต่าง ๆ เหล่านั้น ในขั้นตอนการทำงานมีการใส่ข้อมูลของผู้ใช้ให้สามารถเข้าไปใช้โปรแกรมการทำงานได้อย่างถูกต้อง ขั้นตอนเหล่านี้เราเรียกว่า การแสดงข้อมูลของผู้ใช้ที่มีอยู่ (Authentication) ซึ่งจะถูกนำมาใช้เมื่อมีการใช้โปรแกรมในการทำงาน ระบบกลไกความปลอดภัยก็จะถูกติดตั้งและแสดงขึ้นมาอย่างอัตโนมัติ ในการสอบถามว่า ต้องการให้ผู้ใช้โปรแกรมคนอื่น ๆ เห็นข้อมูลของเราหรือไม่ หรือว่าเราต้องการเข้าไปดูข้อมูลของคนอื่น ๆ ด้วยหรือไม่

การแสดงข้อมูลของผู้ใช้ที่มีอยู่ เป็นระบบที่ถูกควบคุมโดยกลไกของระบบความปลอดภัย ซึ่งระบบปฏิบัติการหลัก (OS Domain) จะเป็นตัวกำหนดผู้ใช้คนไหนที่ได้รับการอนุญาตในการใช้โปรแกรมได้อย่างถูกต้อง โดยที่ผู้ใช้ต้องเป็นคนใส่ข้อมูลส่วนตัวต่าง ๆ เข้าไปในระบบ ในส่วนของโปรแกรมระบบปฏิบัติการและส่วนการติดต่อผู้ใช้ (User Interface: UI) รวมถึงมีการใส่รหัสลับส่วนตัวสำหรับการเข้าทำงานในโปรแกรมต่าง ๆ ณ เวลาที่เราจะเข้าไปใช้โปรแกรมจะมีหน้าต่างขึ้นมาให้ใส่ชื่อและรหัส ถ้ารหัสถูกต้องก็สามารถเข้าไปดูหรือทำงานในโปรแกรมต่าง ๆ ได้ แต่ถ้าใส่รหัสผิดก็จะไม่สามารถเข้าไปดูหรือทำงานภายในโปรแกรมต่าง ๆ ได้ การเข้าสู่ระบบการใช้งานนอกจากการใส่ชื่อและรหัสลับของผู้ใช้แล้ว ยังสามารถใช้วิธีการไบโอเมตริก (Biometrics) เป็นระบบในเชิงชีววิทยา สามารถใช้กลไกในลักษณะการตรวจสอบบางส่วนจากร่างกาย เช่น ตรวจสอบจากลายนิ้วมือ ลูกตา หรือม่านตา แต่ปัจจัยที่จะทำให้เป็นไปได้หรือไม่ นั่น ก็ขึ้นอยู่กับเครื่องมือที่ใช้ทำการทดสอบเหล่านี้ ว่ามีคุณภาพและราคาแพงมากน้อยเพียงไร เนื่องจากเครื่องมือทางชีววิทยาจะมีราคาค่อนข้างสูงแต่มีความถูกต้องแม่นยำสูง

ปฏิบัติการบทที่ 10

1. ให้นักศึกษาแต่ละคน ค้นหาอุปกรณ์ input และ output ต่าง ๆ ที่สามารถต่อพ่วงเข้ามาใช้งานในระบบคอมพิวเตอร์และมือถือ มีอะไรบ้างและใช้ประโยชน์อย่างไร
2. เขียนแผนภาพ (Mind Mapping) แสดงการเชื่อมต่อระหว่าง CPU กับ I/O controller
3. ให้นักศึกษาโหวตเลือกงานแผนภาพที่ดีที่สุด 5 งาน มานำเสนอ

คำถามท้ายบท

1. ระบบ I/O เกี่ยวข้องกับการทำงานของ CPU อย่างไรบ้าง
2. การรับส่งข้อมูลระหว่าง I/O กับ CPU มีวิธีการ อย่างไรบ้าง
3. การขัดจังหวะ (Interrupt) คืออะไร จะเมื่อขึ้นเมื่อไร เมื่อเกิดการขัดจังหวะขึ้น มีขั้นตอนการดำเนินการอย่างไรบ้าง
4. เมื่อผู้ใช้มีการเรียกใช้ I/O มีขั้นตอนการดำเนินการอย่างไรบ้าง
5. ระบบไฟล์คืออะไรมีความสำคัญอย่างไร
6. อธิบายความแตกต่างกันหรือเหมือนกันของไฟล์และไดเรกทอรี
7. โครงสร้างไดเรกทอรีแบบใดที่มีความยืดหยุ่นมากที่สุด และมีข้อดีข้อด้อยอย่างไร
8. ระบบปฏิบัติการ จะมีการป้องกันการเข้าถึงไฟล์จากอุปกรณ์ โดยมีการดำเนินการพื้นฐานกับไฟล์อย่างไรบ้าง
9. จงอธิบายกฎเกณฑ์ที่สำคัญสำหรับความคงสภาพของไฟล์ (Consistency Semantics) ในการใช้ไฟล์ร่วมกัน
10. จงอธิบายเกี่ยวกับระบบป้องกันและความปลอดภัยของข้อมูล ในระบบปฏิบัติการมีหน้าที่และความสำคัญอย่างไร
11. จงอธิบายเกี่ยวกับนโยบายและกลวิธีของระบบป้องกันและความปลอดภัยของข้อมูล ในระบบปฏิบัติการมีหน้าที่และความสำคัญอย่างไร
12. ใ้บรรยายว่าทำไมการแสดงข้อมูลของผู้ใช้ที่มีอยู่ (Authentication) จึงมีความสำคัญกับระบบการป้องกันข้อมูล