NPRU
นครปฐม

การจัดการโปรเซส

T.Kunlaya Charoenmongkonvilai

Agenda

- โปรเซส (Process)
- เทรด (Thread)
- การประสานเวลาของโปรเซส
- บัฟเฟอร์ (Buffer)

Reference : สุจิตรา อดุลย์เกษม,ทฤษฎีระบบปฏิบัติการ.กรุงเทพฯ:โปรวิชั่น,2552.

โปรเซส (Process)

Reference : สุจิตรา อดุลย์เกษม,ทฤษฎีระบบปฏิบัติการ.กรุงเทพฯ:โปรวิชั่น,2552.

ความหมายของโปรเซส

- โปรแกรมที่กำลังถูกเอ็กซิคิวต์
- โปรแกรมที่อยู่ระหว่างการทำงาน
- กิจกรรมที่มีการทำงานสัมพันธ์กัน
- สิ่งที่กำลังใช้งานโปรเซสเซอร์อยู่

Reference : สุจิตรา อดุลย์เกษม,ทฤษฎีระบบปฏิบัติการ.กรุงเทพฯ:โปรวิชั่น,2552.

องค์ประกอบของโปรเซส

- หมายเลขโปรเซส (Process ID)
- โค้ดโปรแกรม (Program Code)
- ข้อมูล (Data)
- บล็อกควบคุมโปรเซส (Process Control Block) หรือ PCB
- PSW (Program Status Word)
- คุณสมบัติของโปรเซส
  - ลำดับความสำคัญของโปรเซส (Priority)
  - อำนาจหน้าที่ของโปรเซส (Authority)
  - คุณสมบัติอื่น ๆ

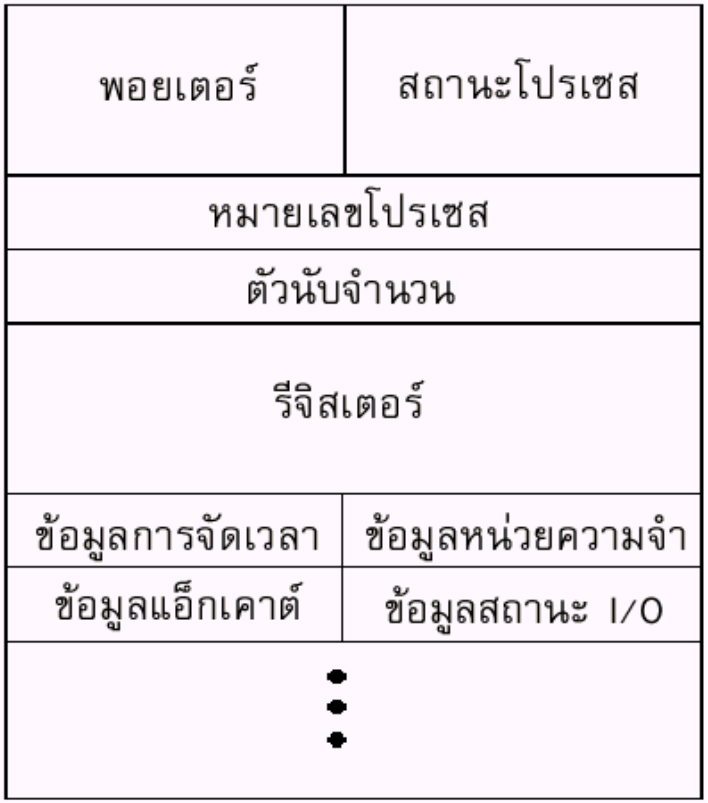
Reference : สุจิตรา อดุลย์เกษม,ทฤษฎีระบบปฏิบัติการ.กรุงเทพฯ:โปรวิชั่น,2552.

บล็อกควบคุมโปรเซส (PCB)

- ตัวชี้ (Pointer)
- สถานะของโปรเซส (Process state)
- หมายเลขของโปรเซส (Process ID)
- ตัวนับโปรเซส (Program counter)
- รีจิสเตอร์ (Register)
- ข้อมูลการจัดการเวลาของซีพียู (CPU scheduling information)
- ข้อมูลการจัดการหน่วยความจำ (Memory management information)
- ข้อมูลแอ็กเคาต์ (Account information)
- ข้อมูลสถานะอินพุต/เอาต์พุต (I/O status information)

Reference : สุจิตรา อดุลย์เกษม,ทฤษฎีระบบปฏิบัติการ.กรุงเทพฯ:โปรวิชั่น,2552.

บล็อกควบคุมโปรเซส (PCB)



Reference : สุจิตรา อดุลย์เกษม,ทฤษฎีระบบปฏิบัติการ.กรุงเทพฯ:โปรวิชั่น,2552.

สถานะของโปรเซส

สถานะของโปรเซสที่สำคัญมี 3 สถานะ คือ

- สถานะพร้อม (Ready State)
- สถานะการทำงาน (Running State)
- สถานะรอ,พัก (Waiting State)

Reference : สุจิตรา อดุลย์เกษม,ทฤษฎีระบบปฏิบัติการ.กรุงเทพฯ:โปรวิชั่น,2552.

การจัดการโปรเซส

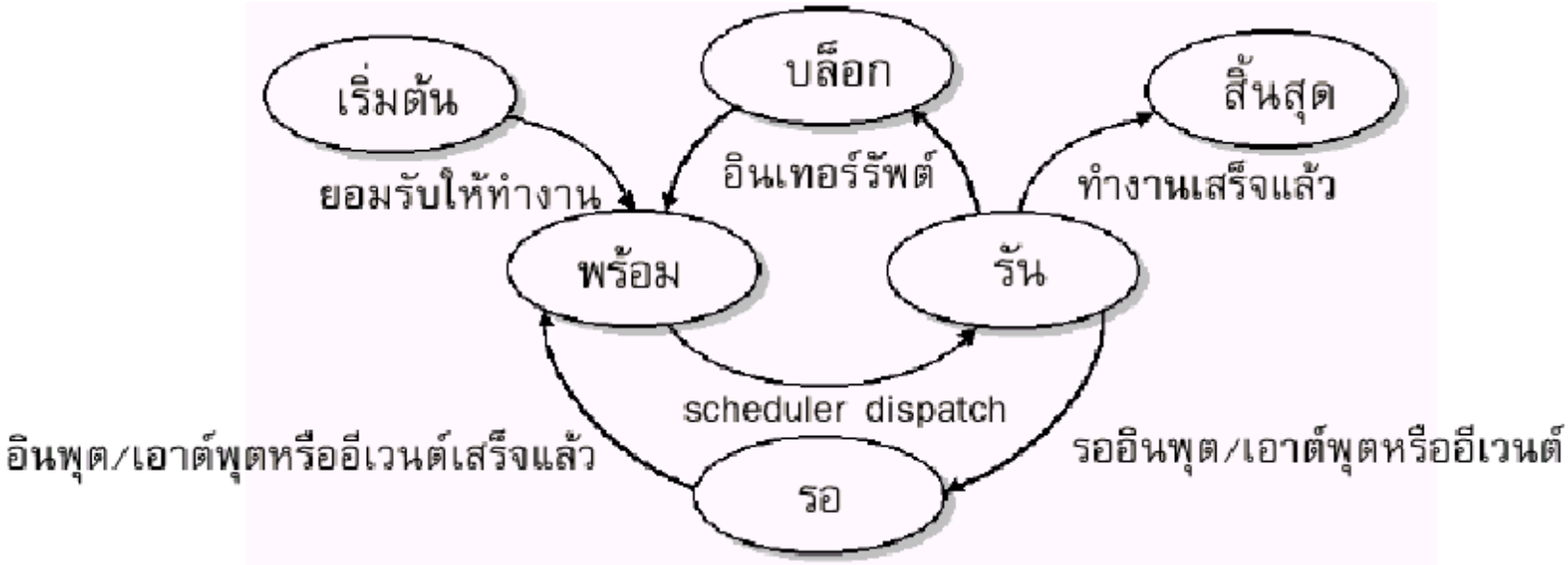


Diagram Status Process

Reference : สุจิตรา อดุลย์เกษม,ทฤษฎีระบบปฏิบัติการ.กรุงเทพฯ:โปรวิชั่น,2552.

การจัดลำดับโปรเซส (Process Scheduling)

- ใช้ในระบบมัลติโปรแกรมมิ่ง คือการจัดโปรเซสให้รันตลอดเวลา เพื่อให้ใช้ประโยชน์ซีพียูได้สูงสุด
- โปรเซสที่ระบบปฏิบัติการสร้างขึ้นมาใหม่จะอยู่ในสถานะพร้อมและเก็บ อยู่ในคิว เรียกว่า Ready Queue เพื่อรอการรัน
- ต้องพิจารณาหน่วยความจำของระบบมีเพียงพอหรือไม่
- Ready Queue จะเก็บโปรเซสในลักษณะลิงค์ลิสต์ (Link list) ที่ส่วนหัวของคิว จะประกอบด้วยพอยเตอร์ของ PCB เริ่มต้น และท้ายสุดของลิสต์ และแต่ละ PCB จะมีพอยเตอร์ชี้ไปยัง PCB ตัวต่อไปของคิว

Reference : สุจิตรา อดุลย์เกษม,ทฤษฎีระบบปฏิบัติการ.กรุงเทพฯ:โปรวิชั่น,2552.

การจัดลำดับโปรเซส (Process Scheduling)

อย่างไรก็ตาม ในการจัดการโปรเซสที่อยู่ในคิวต่างๆ จำเป็นต้องมีโปรแกรมที่ช่วยกำหนดและจัดลำดับของโปรเซส โปรแกรมดังกล่าวเรียกว่า “กำหนดการ (Scheduler)”

โดยแบ่งออกเป็น

- กำหนดการระยะยาว (Long-term Scheduler หรือ Job Scheduler)

ทำหน้าที่เลือกโปรเซสที่รออยู่ในหน่วยความจำสำรองเข้าไปอยู่ที่หน่วยความจำหลัก โดย **จะพิจารณาเลือกโปรเซสจากหน่วยความจำสำรองไปเข้าคิวพร้อมที่หน่วยความจำหลัก เพื่อรอจนกว่าซีพียูจะว่าง**

Reference : สุจิตรา อดุลย์เกษม,ทฤษฎีระบบปฏิบัติการ.กรุงเทพฯ:โปรวิชั่น,2552.

การจัดลำดับโปรเซส (Process Scheduling)

- กำหนดการระยะสั้น (Short-term Scheduler หรือ CPU Scheduler) ทำหน้าที่ เลือกโปรเซสจาก คิวพร้อม และมอบหมายซีพียูให้แก่โปรเซสทีละโปรเซส เป็นผลให้โปรเซสเปลี่ยนสถานะจาก *พร้อม* เป็น *ทำงาน*

ข้อแตกต่างระหว่างกำหนดการทั้ง 2 ข้อ คือ

- กำหนดการระยะสั้นจะถูกเรียกใช้งานมากกว่ากำหนดการระยะยาว

Reference : สุจิตรา อดุลย์เกษม,ทฤษฎีระบบปฏิบัติการ.กรุงเทพฯ:โปรวิชั่น,2552.

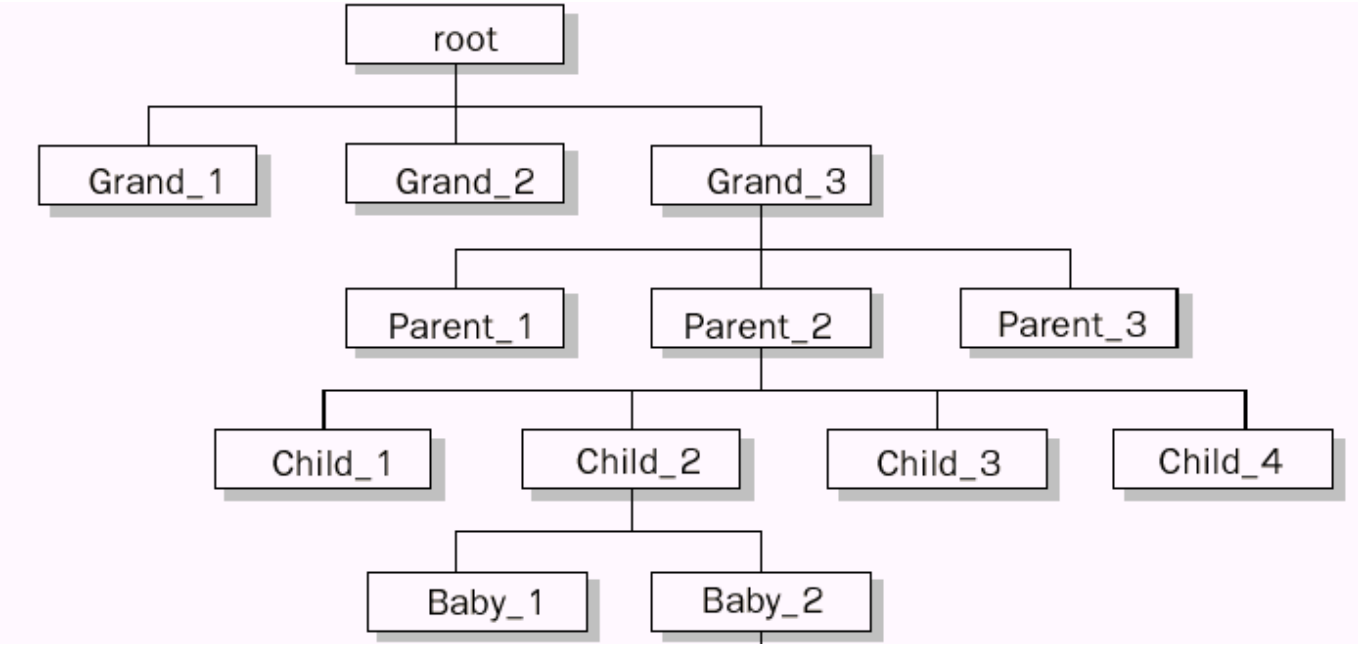
การดำเนินการบนโปรเซส (Operation on Process)

ระบบปฏิบัติการจำเป็นต้องมีวิธีการหรือกระบวนการสร้างและทำลายโปรเซสได้ ด้วยการเรียกใช้ “คำสั่งระบบ (System command)” ของระบบปฏิบัติการ

- โปรเซสพ่อ (Parent Process) โปรเซสที่สร้างโปรเซสอื่น
- โปรเซสลูก (Child Process) โปรเซสที่ถูกสร้าง

Reference : สุจิตรา อดุลย์เกษม,ทฤษฎีระบบปฏิบัติการ.กรุงเทพฯ:โปรวิชั่น,2552.

การดำเนินการบนโปรเซส (Operation on Process)



การเอ็กซิคิวต์ (พร้อมกัน หรือรอให้โปรเซสลูกเสร็จก่อนแม่จึงเอ็กซิคิวต์)

แอ็ดเดรส (โปรเซสลูกสำเนาจากโปรเซสแม่ หรือโหลดแอ็ดเดรสให้ตัวเอง)

Reference : สุจิตรา อดุลย์เกษม,ทฤษฎีระบบปฏิบัติการ.กรุงเทพฯ:โปรวิชั่น,2552.

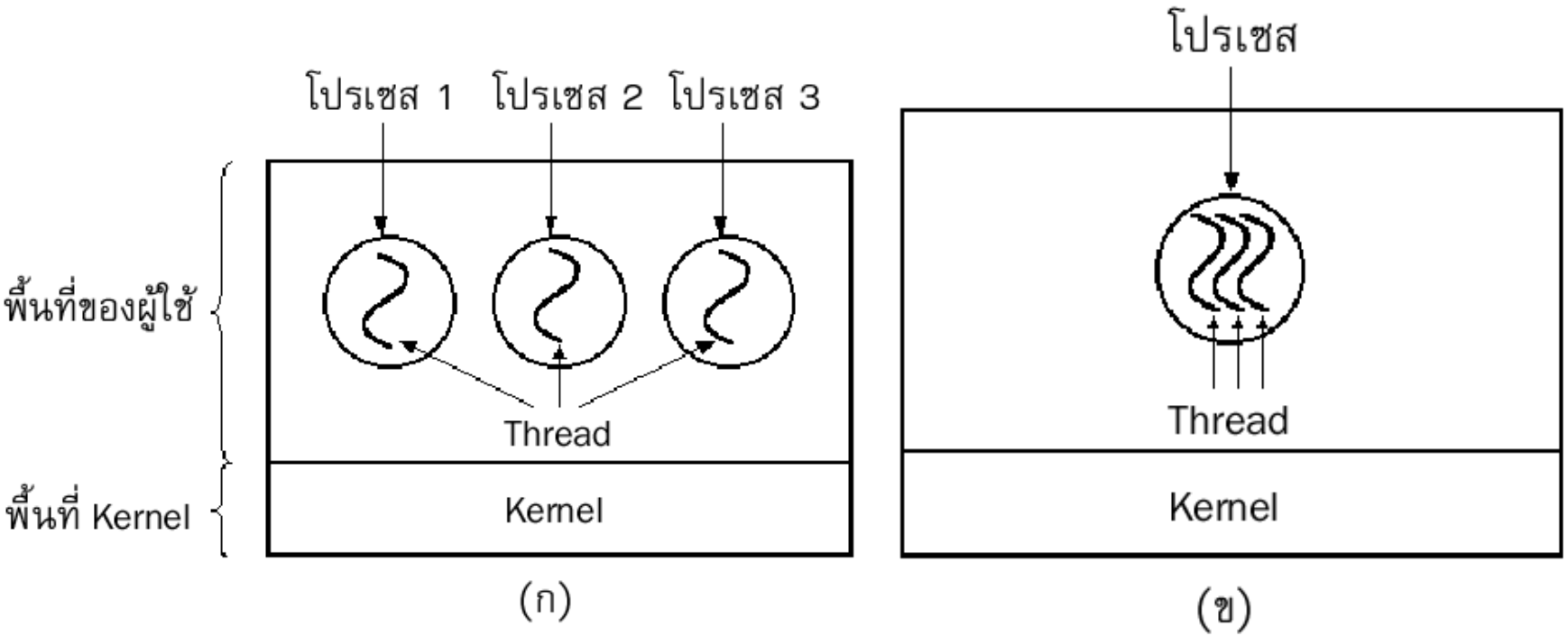
Thread

- Thread เป็นส่วนย่อยของโปรเซสที่เรียกว่า “Lightweight Process”
- แต่ละโปรเซสอาจจะมีหนึ่งเทรด หรือมีหลายเทรดก็ได้
- ข้อดีของการมีหลายเทรด(Multithreaded process) เช่น
  - การตอบสนอง
  - การใช้ทรัพยากรร่วมกัน
  - ความประหยัด
  - การเอื้อประโยชน์ของสถาปัตยกรรมมัลติโปรเซสเซอร์

Reference : สุจิตรา อดุลย์เกษม,ทฤษฎีระบบปฏิบัติการ.กรุงเทพฯ:โปรวิชั่น,2552.

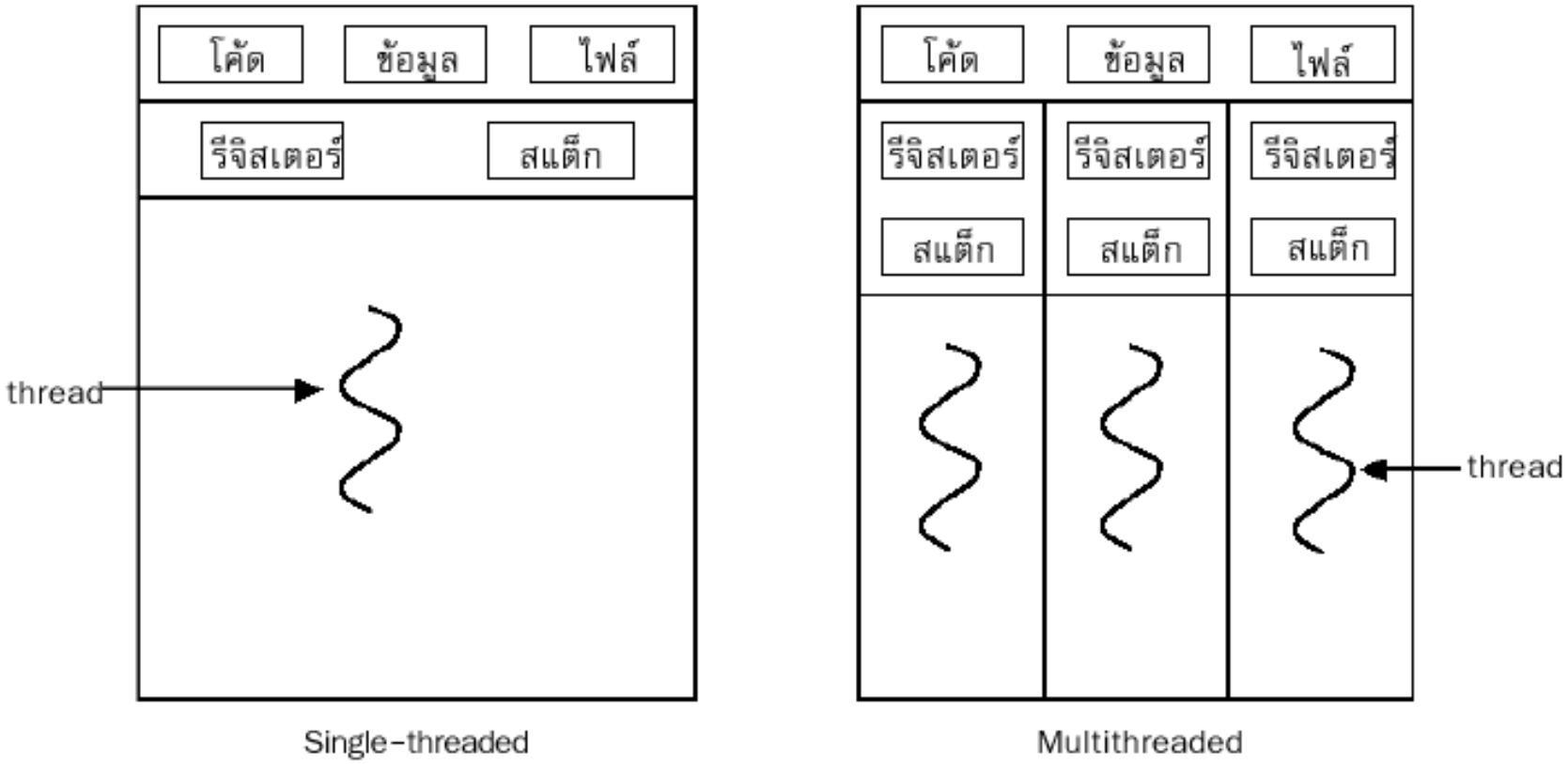
การจัดการโปรเซส

Thread



Reference : สุจิตรา อดุลย์เกษม,ทฤษฎีระบบปฏิบัติการ.กรุงเทพฯ:โปรวิชั่น,2552.

Thread



Reference : สุจิตรา อดุลย์เกษม,ทฤษฎีระบบปฏิบัติการ.กรุงเทพฯ:โปรวิชั่น,2552.

Thread

ส่วนประกอบของ Thread

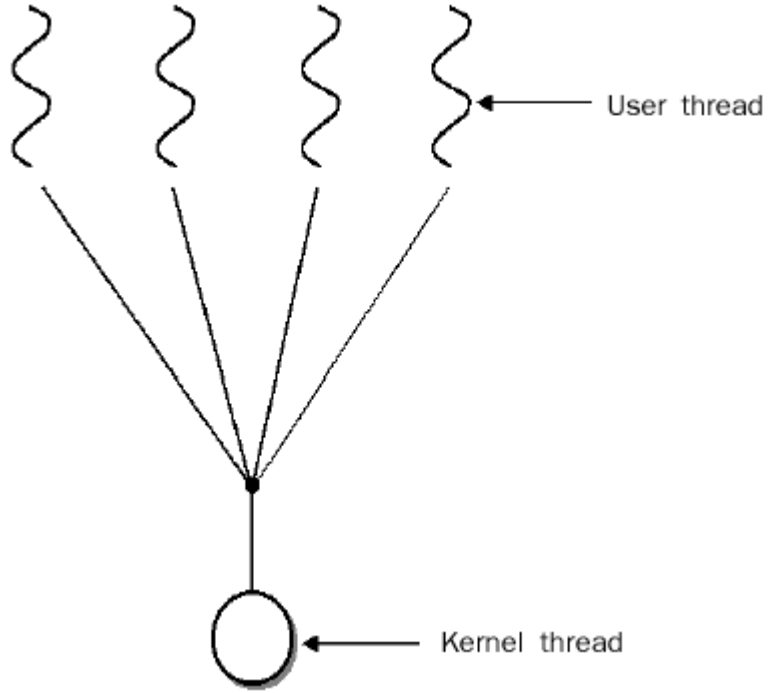
- หมายเลข Thread
- ตัวนับ : ติดตามให้ทราบคำสั่งต่อไปที่จะเอ็กซิคิวต์
- ชุดของรีจิสเตอร์ : เก็บค่าตัวแปรที่ทำงานอยู่
- สแต็ก : เก็บประวัติการเอ็กซิคิวต์

Reference : สุจิตรา อดุลย์เกษม,ทฤษฎีระบบปฏิบัติการ.กรุงเทพฯ:โปรวิชั่น,2552.

Thread

โมเดลของ MultiThread

Many-to-One : 1 Kernel กับหลาย User thread การจัดการ threadอยู่ในพื้นที่ของผู้ใช้ซึ่งมีประสิทธิภาพ แต่ถ้า thread บล็อก system call จะทำให้โปรเซสถูกบล้อกไปด้วย

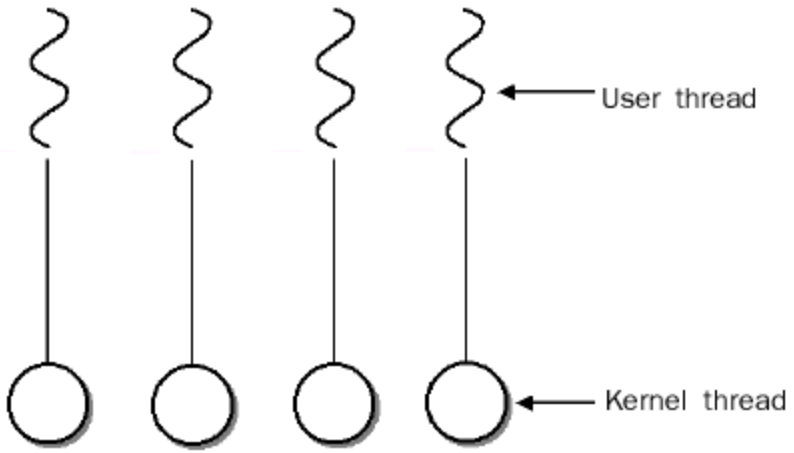


Reference : สุจิตรา อดุลย์เกษม,ทฤษฎีระบบปฏิบัติการ.กรุงเทพฯ:โปรวิชั่น,2552.

Thread

โมเดลของ Multithread

One-to-One : 1 Kernel กับ 1 User thread ทำงานได้พร้อมกันดีกว่าแบบแรก โดยยอมให้ thread อื่นรันได้เมื่อ thread บล็อก system call ต้องคำนึงการสร้าง kernel และ user thread ต้องสัมพันธ์กัน การสร้าง kernel thread เป็นประสิทธิภาพของโปรแกรม

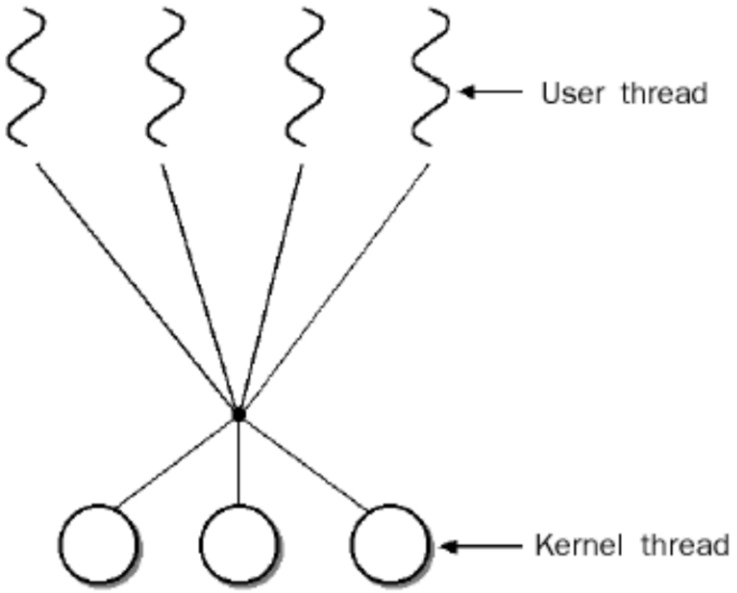


Reference : สุจิตรา อดุลย์เกษม,ทฤษฎีระบบปฏิบัติการ.กรุงเทพฯ:โปรวิชั่น,2552.

Thread

โมเดลของ Multithread

Many-to-many : User thread มากว่าหรือเท่ากับ Kernel thread ได้ kernel thread กำหนดแอปพลิเคชันMany-to-One ยอมให้ผู้พัฒนาสร้าง user thread ได้ตามต้องการ แต่ไม่สามารถรันได้พร้อมกัน เนื่องจาก Kernel จัดเวลาให้ครั้งละ thread



Reference : สุจิตรา อดุลย์เกษม,ทฤษฎีระบบปฏิบัติการ.กรุงเทพฯ:โปรวิชั่น,2552.

การประสานเวลาของโปรเซส

- P0 : read a ; //อ่านค่าตัวแปร a
- a = a + 1 ; //เพิ่มค่าตัวแปร a
- print a ; //พิมพ์ค่าตัวแปร a
- P1 : read b ; // อ่านค่าตัวแปร b
- b = b + 1 ; // เพิ่มค่าตัวแปร b
- print b ; // พิมพ์ค่าตัวแปร b
- ตัวอย่างการทำงานของโปรเซสที่ไม่ส่งผลกระทบต่อกัน
- เป็นอิสระต่อกัน

Reference : สุจิตรา อดุลย์เกษม,ทฤษฎีระบบปฏิบัติการ.กรุงเทพฯ:โปรวิชั่น,2552.

การประสานเวลาของโปรเซส

- P0 : read x ; //อ่านค่าตัวแปร x
- x = x + 1 ; //เพิ่มค่าตัวแปร x
- print x ; //พิมพ์ค่าตัวแปร x
- P1 : read x ; // อ่านค่าตัวแปร x
- x = x + 1 ; // เพิ่มค่าตัวแปร x
- print x ; // พิมพ์ค่าตัวแปร x
- ตัวอย่างการทำงานของโปรเซสที่ส่งผลกระทบต่อกัน
- ไม่เป็นอิสระต่อกัน

Reference : สุจิตรา อดุลย์เกษม,ทฤษฎีระบบปฏิบัติการ.กรุงเทพฯ:โปรวิชั่น,2552.

Buffer กับการแสดงข้อมูล

กำหนดให้ Buffer มีขนาดเท่ากับ 7 ให้แสดงข้อมูลที่บรรจุใน Buffer และค่าของตัวแปร Full และ Buffer เมื่อ

N = 7

Full = 0

Empty = N = 7

Reference : สุจิตรา อดุลย์เกษม,ทฤษฎีระบบปฏิบัติการ.กรุงเทพฯ:โปรวิชั่น,2552.

Buffer กับการแสดงข้อมูล

Process Programmer นำอักขระ “A” บรรจุใน Buffer

Full = 0 + 1 = 1

Empty = 7 – 1 = 6

Reference : สุจิตรา อดุลย์เกษม,ทฤษฎีระบบปฏิบัติการ.กรุงเทพฯ:โปรวิชั่น,2552.

Buffer กับการแสดงข้อมูล

Process User นำอักขระ “A” ออกจาก Buffer

Full = 1 -1 = 0

Empty = 6 + 1 = 7

Reference : สุจิตรา อดุลย์เกษม,ทฤษฎีระบบปฏิบัติการ.กรุงเทพฯ:โปรวิชั่น,2552.