บทที่ ๓

เพา เวอร์/วี เอส

เพา เวอร์/วี เอส เป็นชื่อของโปรแกรมควบคุมระบบ ซึ่งใช้กับ เครื่องไอบี เอ็ม โดยใช้ร่วมกับโปรแกรมควบคุมระบบดอส/วี เอส การทำงานภายใต้การควบคุมของ เพา เวอร์/วี เอส ทำให้ เกิดประโยชน์[๑] คือ

๑. เพิ่มปริมาณของงานในระบบ เนื่องจากงานที่กำลังทำอยู่ในพาติชันต่าง ๆ ไม่ต้อง เสีย เวลารอการรับ-ส่งข้อมูลจากหน่วยรับ-ส่งข้อมูลที่รับ-ส่งข้อมูลได้ทีละ ๑ ระ เบียน

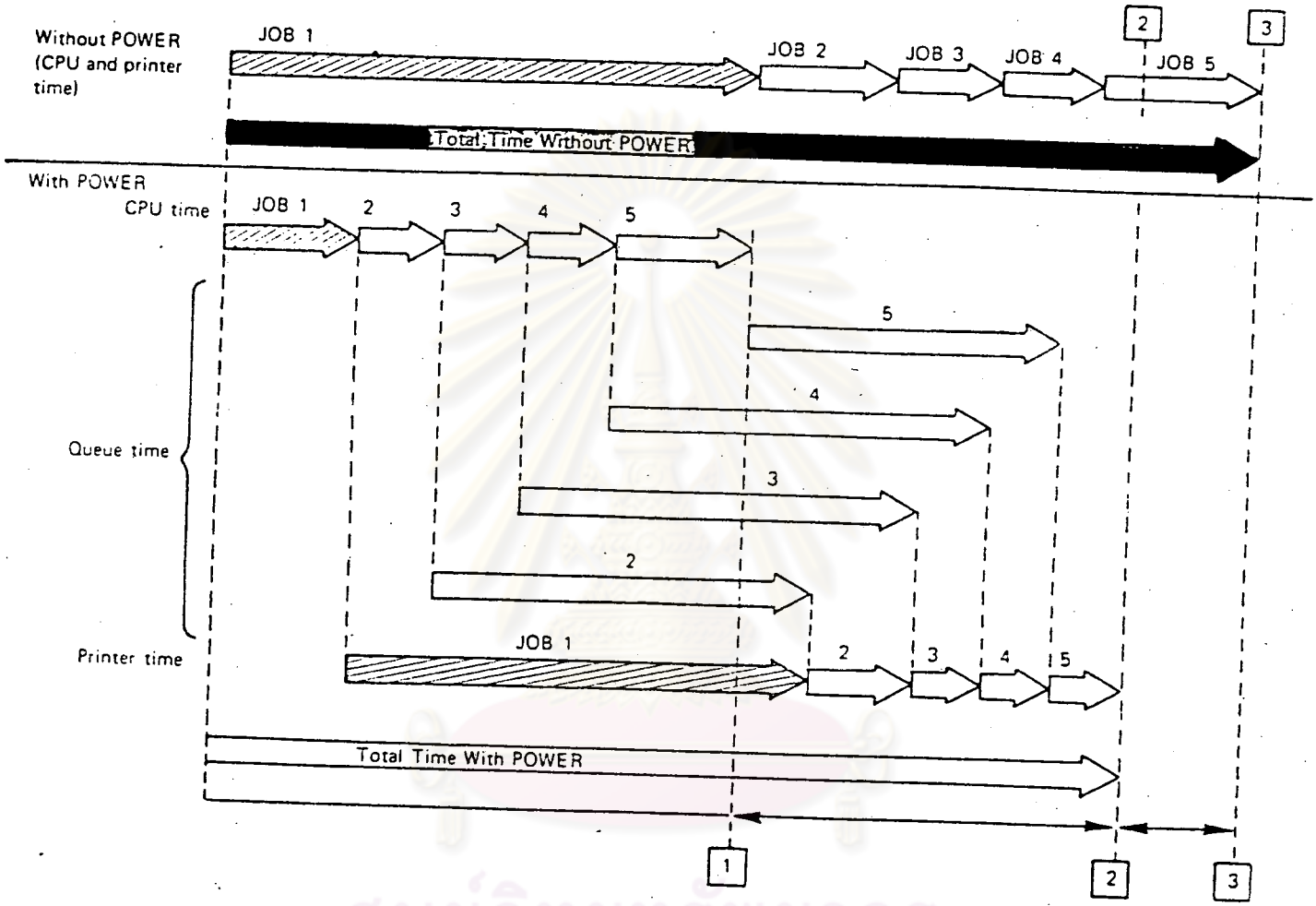
๒. เพิ่มประสิทธิภาพการใช้งานของอุปกรณ์ที่สามารถรับ-ส่งข้อมูลได้ทีละ ๑ ระ เบียน ให้สามารถใช้ร่วมกับงานหลาย ๆงานได้ใน เวลา เดียวกัน

๓. ช่วยในการรับ-ส่งข้อมูลระหว่าง เทอร์มินอลกับตัวรับคำสั่ง

เพา เวอร์/วี เอส เป็นโปรแกรมที่ควบคุมและจัดลำดับการทำงานของหน่วยรับ-ส่งข้อมูลกับงานที่กำลังทำอยู่หลาย ๆ งานในขณะ เดียวกัน ตั้งแต่ ๑ ถึง ๔ งานโดยใช้หลักการของสพูลลิ่ง ซึ่งทำงานในพาติชันหนึ่งของระบบดอส/วี เอสที่มีสิทธิทำงานสูงกว่าทุก ๆ พาติชันที่ เพา เวอร์/วี เอสให้บริการอยู่ทำงานในโหมดการทำงาน เสมือน (Virtual mode) ผลของการทำงานของ เพา เวอร์/วี เอสทำให้ เกิดการทำงาน เหลื่อมกัน (overlapping) ระหว่างหน่วยรับ-ส่งข้อมูลกับงานต่าง ๆ ที่กำลังทำอยู่ในขณะนั้น ดังรูปที่ ๓.๑

---

[๑] IBM, POWER/VS Installation Guide and Reference, Form GC 33-6048-1, (Bangkok: IBM Co., Ltd. (Thailand)), p. iii.



รูปที่ ๓.๑ การทำงานภายใต้การควบคุมของ เพา เวอร์/วี เอส

๓.๑ งานของ เพา เวอร์/วี เอส[๑]

แบ่งออกเป็น ๓ ส่วนใหญ่คือ

๑. งานอ่าน (READ) ทำการอ่านข้อมูลของงานจากเครื่องอ่าน แล้วนำไปเก็บไว้ในจานแม่เหล็ก หรือทำการนำข้อมูลเข้าไปทำงานในดอส/วี เอส

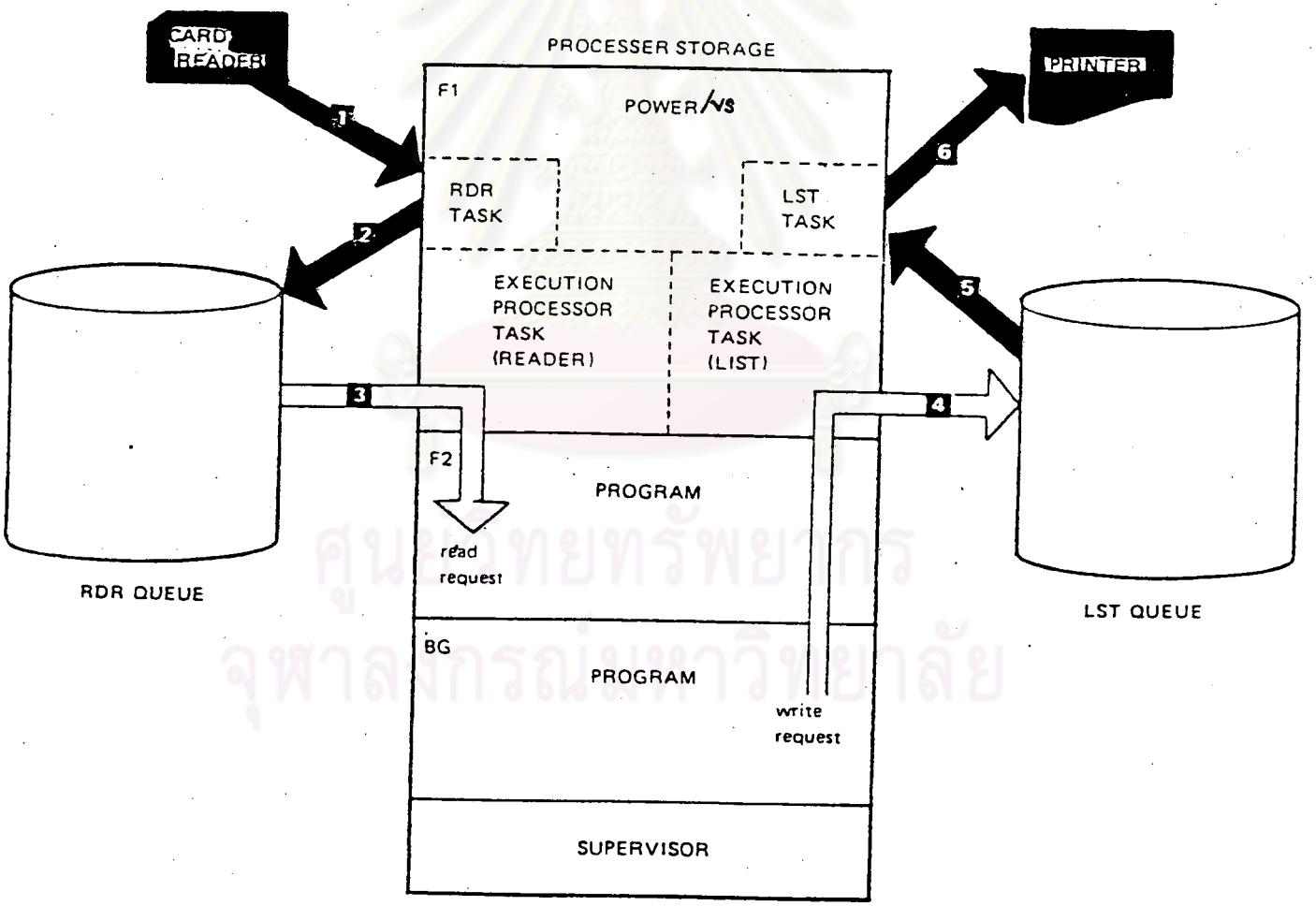
---

[๑]IBM, DOS/VS POWER/VS Logic 1, Form SY 33-8576-1, (Bangkok: IBM Co., Ltd. (Thailand)), p. 10.

๒. งานพิมพ์ (LIST) ทำการเก็บผลลัพธ์ที่ได้จากการประมวลผลของงานในดอส/วีเอสไปจัดเก็บไว้ในจานแม่เหล็ก ก่อนหน้าที่จะพิมพ์ออกทางเครื่องพิมพ์

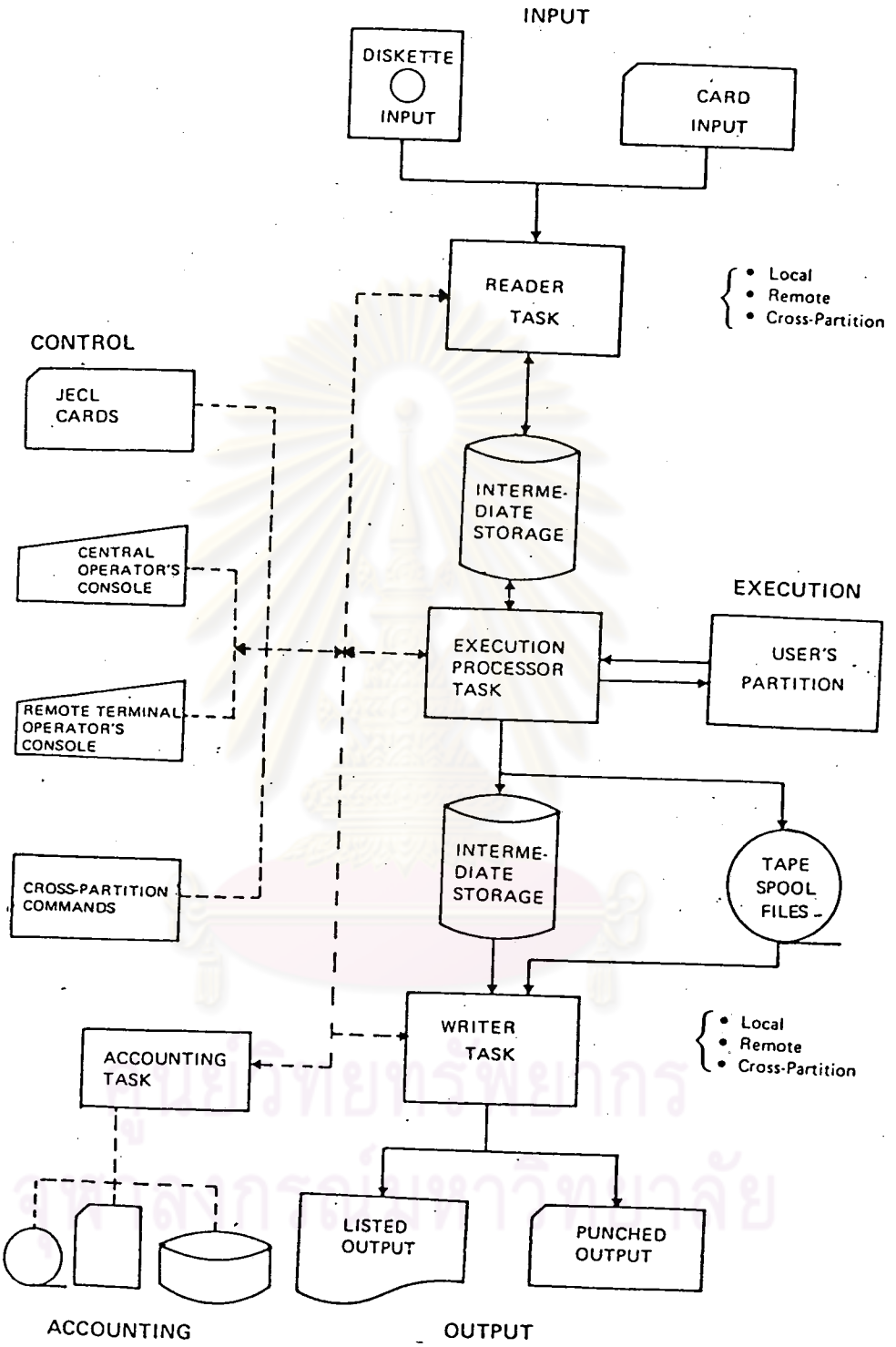
๓. งานเจาะบัตร (PUNCH) ทำการเก็บผลลัพธ์ของงานที่ต้องการเจาะออกทางเครื่องเจาะบัตรไว้ในจานแม่เหล็กก่อนที่จะนำไปยังเครื่องเจาะบัตร

งานหลักทั้งสามนี้มีขั้นตอนและความสัมพันธ์กับข้อมูลของงานที่ผู้ใช้และผู้ควบคุมเครื่องกำหนดให้กับเพาเวอร์/วีเอส ดังแสดงในรูปที่ ๓.๒, ๓.๓



รูปที่ ๓.๒ งานอ่านและงานพิมพ์ของเพาเวอร์/วีเอส

015197



————Data Flow

--------Operator Communication

รูปที่ ๓.๓ ความสัมพันธ์ของการทำงานของ เพา เวอร์/วีเอส ข้อมูลและการติดต่อกับผู้ควบคุม เครื่อง

๓.๒ โครงสร้างภายในของเพาเวอร์/วีเอส[๑]

ระบบการทำงานของเพาเวอร์/วีเอส ประกอบด้วยเฟส (phase) ต่าง ๆ ที่ทำหน้าที่ดังต่อไปนี้

๓.๒.๑ เอกเทอร์นอลรูทีน (EXTERNAL ROUTINE) ทำหน้าที่ติดต่อและควบคุมการทำงานของเครื่องอ่าน เครื่องพิมพ์และเครื่องเจาะบัตร โดยทำการส่งข้อมูลจากเครื่องอ่านเข้าสู่อินเทอร์นอลรูทีน หรือทำการส่งข้อมูลจากอินเทอร์นอลรูทีนไปยังเครื่องพิมพ์, เครื่องเจาะบัตร

๓.๒.๒ อินเทอร์นอลรูทีน (INTERNAL ROUTINE) ทำหน้าที่ประสานงานระหว่างเอกเทอร์นอลรูทีนกับฟังก์ชันภายในของเพาเวอร์/วีเอส โดยจะเป็นตัวเรียกฟังก์ชันมาโคร (FUNCTION MACRO) ต่าง ๆ เข้ามาทำงาน

๓.๒.๓ ฟังก์ชันมาโคร (FUNCTION MACRO) ทำหน้าที่ช่วยการทำงานของรูทีนต่าง ๆ อาจถูกเรียกให้เข้ามาทำงานโดยตรงจากเอกเทอร์นอลรูทีนได้ เช่นการรับ-ส่งข้อมูลจากจานแม่เหล็ก เป็นต้น

๓.๒.๔ เซอร์วิสมาโคร (SERVICE MACRO) ทำหน้าที่ให้บริการการใช้แหล่งข้อมูล (resource) และข้อมูลที่จำเป็นแก่รูทีนและฟังก์ชันมาโครต่าง ๆ เช่น การใช้เนื้อที่ในจานแม่เหล็ก (Disk Management) การเลือกงานที่จะทำ (Task Management) เป็นต้น

การติดต่อระหว่างเฟสต่าง ๆ เหล่านี้ จะมีลิงค์เคด (LINKAGE) ซึ่งเป็นมาโคร (MACRO) ที่ใช้กำหนดรูปแบบของบลอคควบคุม (control block) ซึ่งใช้ในการทำงานของเพาเวอร์/วีเอส เนื้อที่ของข้อมูลที่ใช้ส่งผ่านหรือจัดการทำงานเกี่ยวกับฟังก์ชันพิเศษโดยใช้การติดต่อผ่านทางรีจิสเตอร์ (Register) ลิงค์เคดเหล่านี้ได้แก่

---

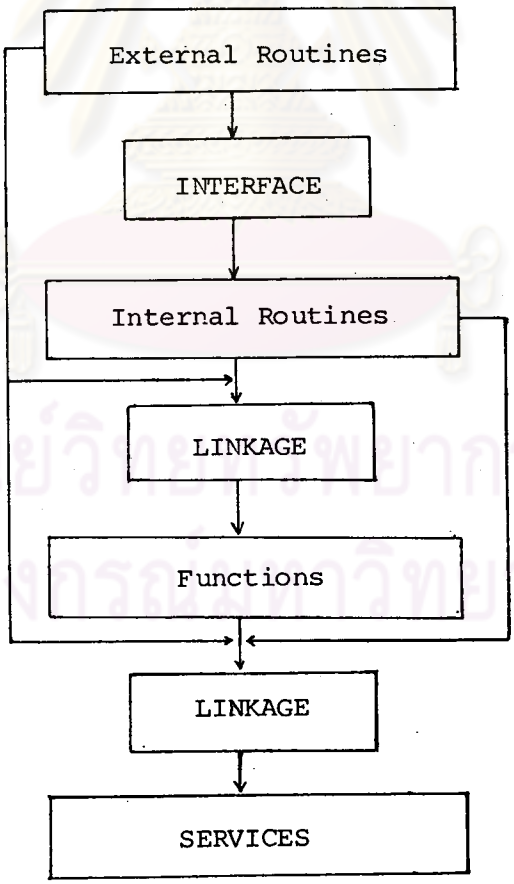
[๑] IBM, DOS/VS POWER/VS Logic 2, Form SY 33-8577-1, (Bangkok: IBM Co., Ltd. (Thailand)), p. 8.

- อินเทอร์เฟสลิงค์เคด (INTERFACE LINKAGE) ถูกเรียกใช้เมื่อมีการส่งผ่านการควบคุมระหว่างอินเทอร์นอลรูทีนกับเอกเทอร์นอลรูทีน ได้แก่ การเปิดการติดต่อทางตรรกภาพ (OPEN LOGICAL INTERFACE), การรับ-ส่งข้อมูล เป็นต้น

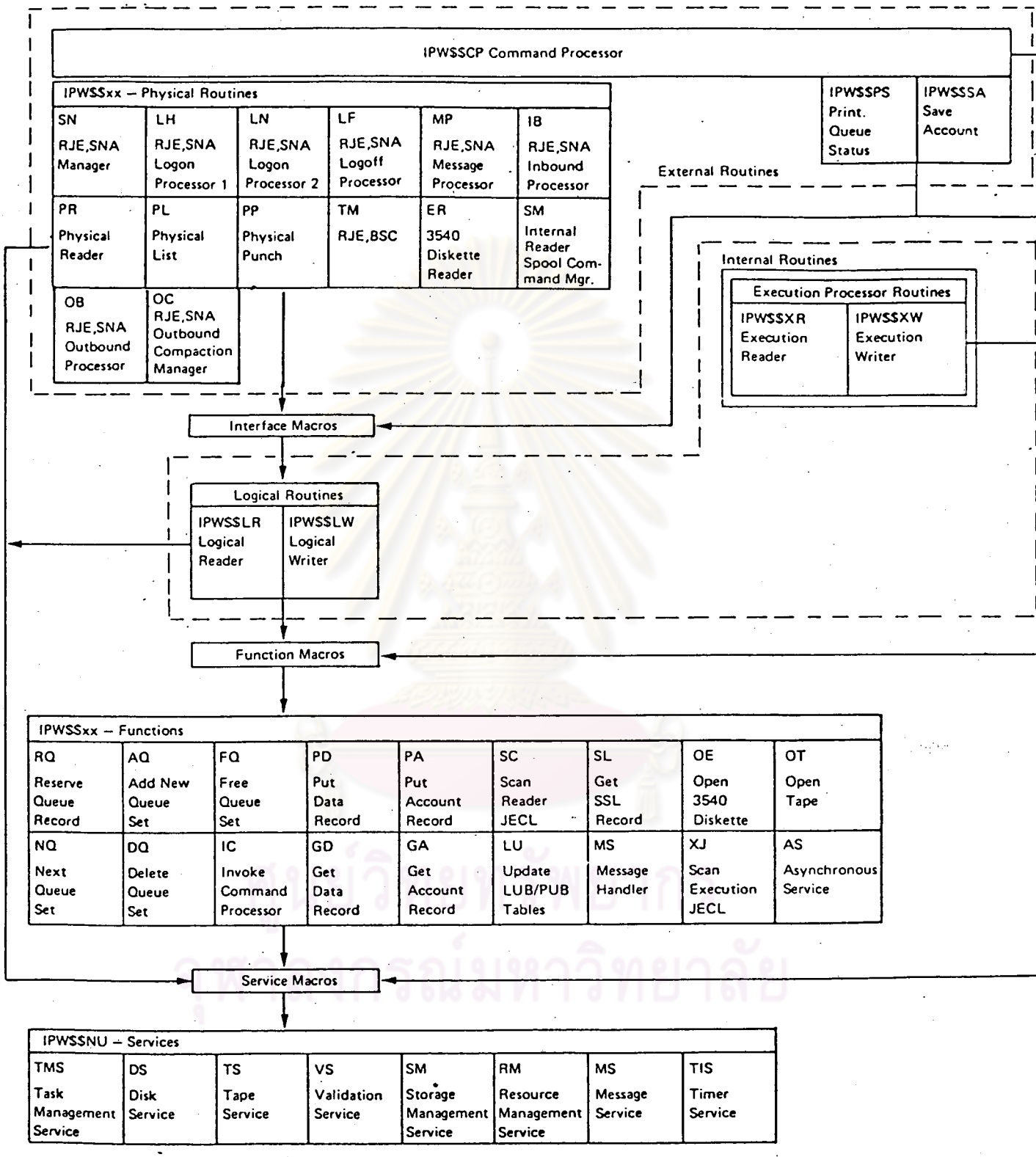
- ฟังก์ชันลิงค์เคด (FUNCTION LINKAGE) ถูกเรียกใช้เมื่ออินเทอร์นอลรูทีนเรียกใช้ฟังก์ชันมาโคร เช่นการจัดลำดับการเข้าออกของงาน

- เซอร์วิสลิงค์เคด (SERVICE LINKAGE) ถูกเรียกใช้เมื่อรูทีนและฟังก์ชันมาโครต่าง ๆ เรียกใช้ เช่นการสำรองเนื้อที่ในหน่วยความจำ

การทำงานเป็นระดับของเพาเวอร์/วีเอส พอสรุปเขียนเป็นแผนผังได้ดังรูปที่ ๓.๔ และรูปที่ ๓.๕



รูปที่ ๓.๔ ระดับการทำงานในเพาเวอร์/วีเอส



รูปที่ ๓.๕ การประสานงานโดยละเอียดของเพาเวอร์/วีเอส

๓.๓ ขั้นตอนการทำงานของเพาเวอร์/วีเอส[๑]

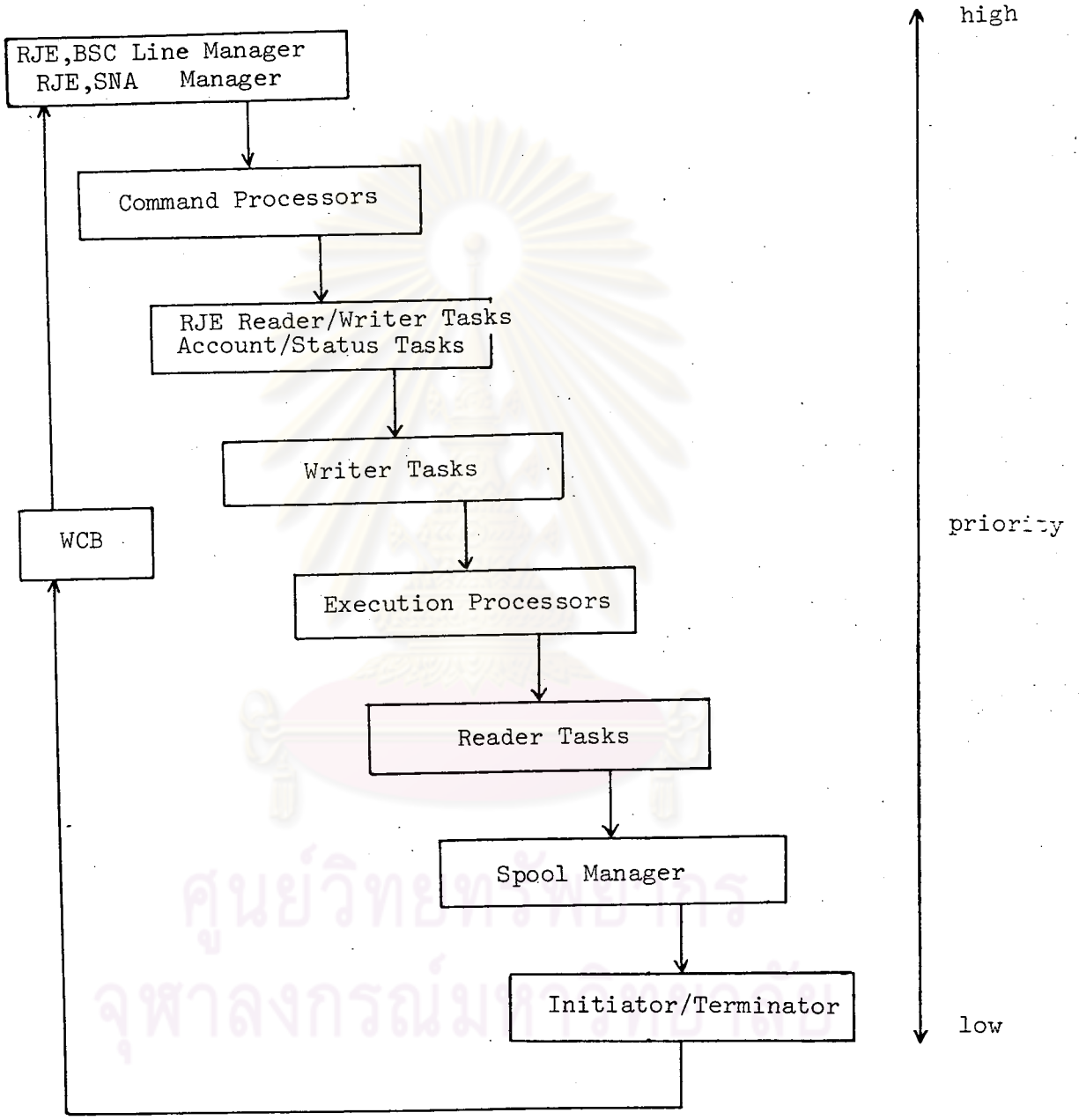
การทำงานของแต่ละส่วนในโครงสร้างของเพาเวอร์/วีเอสจะจัดทำเป็นส่วนงานย่อย (Task) โดยนิวเคลียสของเพาเวอร์/วีเอส (IPW$$NU) จะเป็นตัวเลือกและจัดการงานย่อยแต่ละงานตามลำดับความสำคัญก่อนหลัง (Task Management) ให้เข้ามาทำงาน

ลักษณะการเลือกและจัดการงานย่อยเหล่านี้ มีโครงสร้างเป็น link list โดยมีบลอคควบคุมที่เรียกว่า WCB (Wait Control Block) เป็นจุดเริ่มต้นและจุดสุดท้ายของ link list ของบลอคควบคุมที่เรียกว่า TCB (Task Control Block)

TCB เป็นบลอคควบคุมที่ถูกสร้างขึ้นมาในงานย่อยแต่ละงาน ใช้เก็บข้อมูลต่าง ๆ ที่จำเป็นแก่นิวเคลียสของเพาเวอร์/วีเอสในการจัดการให้งานย่อยนั้นให้มาทำงาน แต่ละ TCB แยกเป็นประเภทต่าง ๆ แต่ละประเภทมีลำดับความสำคัญแตกต่างกันดังรูปที่ ๓.๖

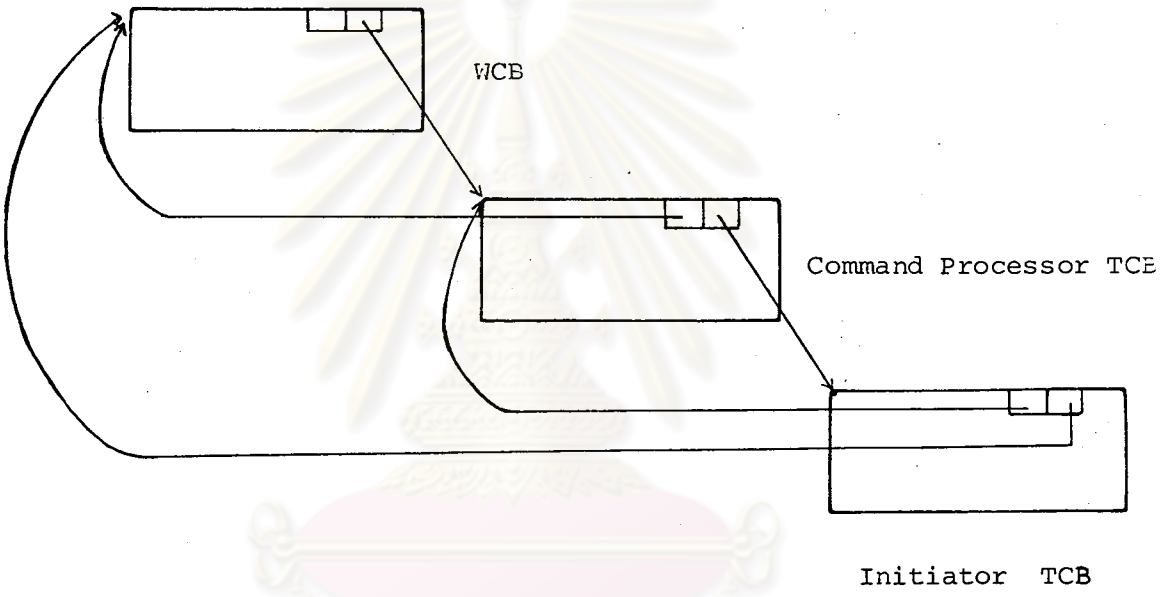
---

[๑] IBM, DOS/VS POWER/VS Logic 1, Form SY 33-8576-1, (Bangkok : IBM Co., Ltd. (Thailand)), p. 38.



รูปที่ ๓.๖ บลอคควบคุม (WCB, TCBs) ที่ประกอบกันเป็นวงจรการทำงานตามลำดับความสำคัญก่อนหลัง

Link List นี้มีชื่อเรียกว่า "วงจรการทำงาน" (TSL ย่อมาจาก Task Selection List) จะถูกสร้างขึ้นมาครั้งแรกเมื่อเริ่มต้นให้เพาเวอร์/วีเอสทำงาน โดยมีบลอคควบคุม ๓ ตัวประกอบกันเป็นวงจรการทำงานได้แก่ WCB, Command Processor TCB, Initiator TCB



รูปที่ ๓.๗ วงจรการทำงานที่ถูกสร้างขึ้นเมื่อเริ่มต้นให้เพาเวอร์/วีเอสทำงาน

หลังจากนั้นตัวจัดงานย่อย (Task Management) จะเป็นตัวจัดการเอางานย่อยให้มาทำงาน และดึงเอางานที่ทำเสร็จแล้วออกจากวงจรการทำงานนี้

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

งานย่อย เหล่านี้ได้แก่

- งานติดต่อกับผู้ควบคุม เครื่องและการติดต่อสื่อสารภายนอก (Communication and Line Manager) ทำหน้าที่รับ-ส่งข้อมูลระหว่างคอนโซล (console) ของเครื่องหรือสถานีข้อมูลซึ่งอยู่ไกล (Remote Data Station) กับตัวรับคำสั่ง

- งานรับคำสั่ง (Command Processor) เป็นตัวรับคำสั่ง และส่งคำสั่งไปยังรูทีนหรือมาโครที่ทำหน้าที่ตามคำสั่งนั้น ๆ

- งานอ่าน/เขียนข้อมูลระหว่างแฟ้มข้อมูลในจานแม่เหล็กกับสถานีข้อมูล ซึ่งอยู่ไกล (RJE Reader/ Writer Tasks)

- งาน เก็บทะเบียนสถิติการทำงานของเครื่อง (Account / Status Tasks)

- งาน เขียนข้อมูล (Writer Tasks) เป็นส่วนทำงานระหว่างแฟ้มข้อมูลในจานแม่เหล็กกับเครื่องพิมพ์หรือเครื่องเจาะบัตร

- งานประมวลผล (Execution Processors) เป็นส่วนทำงานระหว่างแฟ้มข้อมูลในจานแม่เหล็กกับงานที่กำลังทำอยู่ในดอส/วีเอส

- งานอ่านข้อมูล (Reader Tasks) เป็นส่วนทำงานระหว่างแฟ้มข้อมูลในจานแม่เหล็กกับเครื่องอ่าน

- งานรับ-ส่งข้อมูลข้ามพาติชัน (Spool Manager) เป็นส่วนทำงานระหว่างแฟ้มข้อมูลในจานแม่เหล็กกับงานที่กำลังทำอยู่ในดอส/วีเอสในพาติชันที่ไม่อยู่ภายใต้การควบคุมของเพาเวอร์/วีเอส ทำหน้าที่นำข้อมูลเข้าหรือออกจากแฟ้มข้อมูลเพาเวอร์/วีเอสให้กับงานในพาติชันที่ไม่อยู่ภายใต้การควบคุมของเพาเวอร์/วีเอส

- งานควบคุมงานย่อยอื่น (Initiator / Terminator Tasks) เป็นส่วนนำเอางานต่าง ๆ ดังกล่าวข้างต้นเข้ามาหรือออกไปจากวงจรการทำงานโดยรับคำสั่งจากส่วนรับคำสั่ง

งานของตัวจัดงานย่อยมี ๓ งานคือ

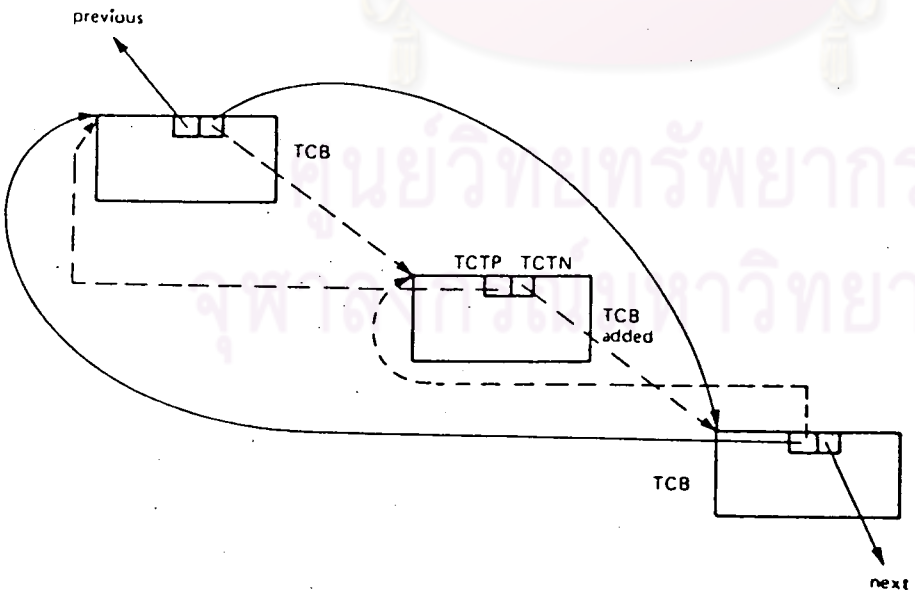
๑) นำงานเข้ามาอยู่ในวงจรการทำงาน (Task Initiation) และกำหนดภาวะของ TCB ของงานนั้น ภาวะของ TCB มี ๒ ภาวะคือ

TCB ที่พร้อมจะทำงานทันทีมีภาวะเป็น D (dispatch)

TCB ที่จะทำงานโดยมีเงื่อนไขได้แก่ภาวะ L,F,S,C,Q,M

๒) การเลือกงาน (Task Selection) ทำการหางานต่อไปในวงจรการทำงานให้มาทำงาน โดยเปลี่ยนภาวะของงานนั้นเป็น R(running) เริ่มทำงานเมื่องานใดที่กำลังทำงานอยู่ มอบการควบคุมผ่านทางมาโคร IPW$WFx (wait for 'x' ค่าของ x ได้แก่ I,O,L,M,R,C,S,D) ถ้าหากไม่มีงานใดอยู่ในวงจรการทำงานเลย จะมอบการควบคุมคืนให้ดอส/วีเอสผ่านทาง SVC7 และรอจนกว่าดอส/วีเอสจะผ่านการควบคุมคืนให้เพาเวอร์/วีเอส เช่น เมื่อมีการส่งผ่านข้อมูลทาง I/O ตัวใดเรียบร้อยแล้ว เป็นต้น

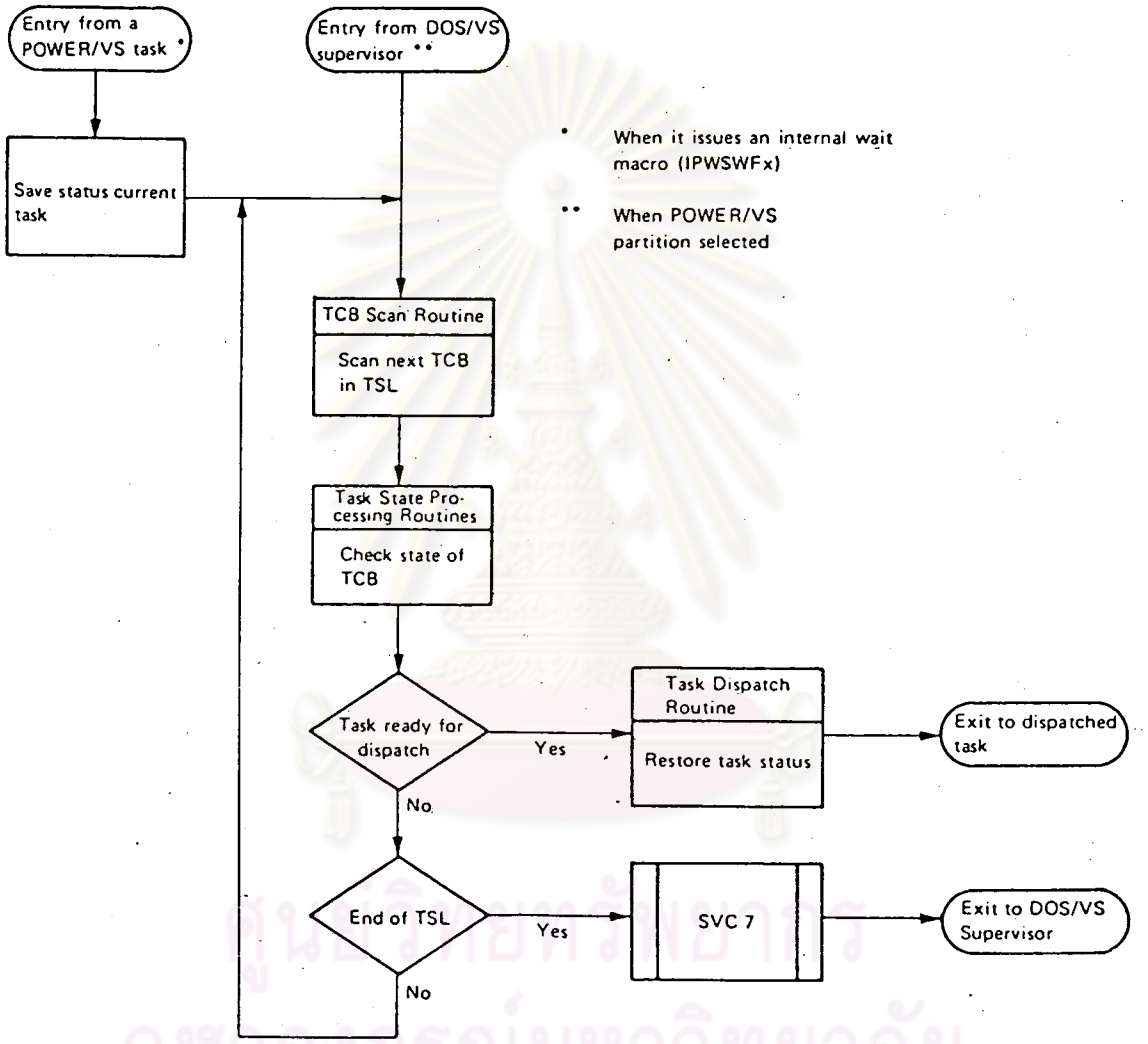
๓) นำงานออกจากวงจรการทำงานเมื่องานนั้นเสร็จแล้ว (Task Termination) และมอบการควบคุมให้ตัวเลือกงานทำการเลือกงานต่อไป



---------- new pointers

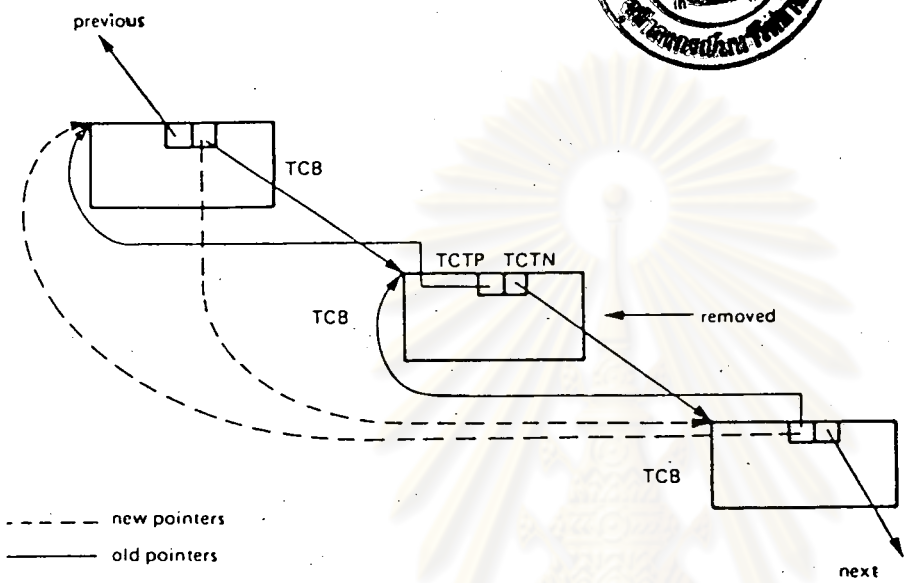
_____ old pointers

รูปที่ ๓.๘ การเปลี่ยนตัวชี้ในวงจรการทำงานเมื่อนำงานเข้ามาอยู่ในวงจรการทำงาน



ผังงานที่ ๓.๑ ขั้นตอนของการเลือกงาน

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



---------- new pointers

_____ old pointers

รูปที่ ๓.๙ การเปลี่ยนตัวชี้ในวงจรการทำงาน เมื่อนำงานออกจากวงจรการทำงาน

๓.๔ แฟ้มข้อมูลของเพาเวอร์/วีเอส[๑]

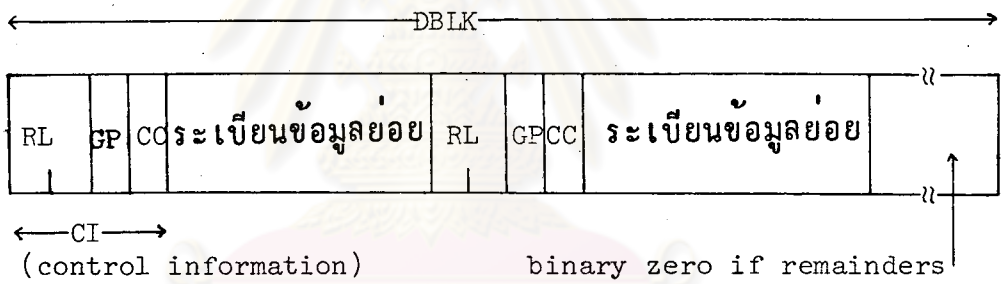
เพาเวอร์/วีเอส สร้างระบบแฟ้มข้อมูลเป็น ๓ ส่วนคือ

---

๑ IBM, DOS/VS POWER/VS Logic 1, Form SY 33-8576-1, (Bangkok: IBM Co., Ltd. (Thailand)), p. 16.

๓.๔.๑ แฟ้มข้อมูล (DATA FILE)

เป็นแฟ้มข้อมูลที่เก็บข้อมูลต่าง ๆ ที่อ่านได้จากเครื่องอ่านและข้อมูลต่าง ๆ ที่ได้จากงานในดอส/วีเอส เพื่อรอนำออกไปยังเครื่องพิมพ์หรือเครื่องเจาะบัตร ใช้วิธีเข้าถึงข้อมูลโดยตรง (DIRECT ACCESS) ขนาดของระเบียนที่ใหญ่ที่สุดในแฟ้มข้อมูลนี้ที่สามารถกำหนดได้คือ ๒๐๐๘ ไบต์ เล็กที่สุด ๕๔๔ ไบต์ เรียกแต่ละระเบียนว่า DBLK (data block size) ข้อมูลในแต่ละ DBLK ประกอบด้วยระเบียนข้อมูลย่อยหลายระเบียน มีรูปแบบการเก็บตามรูปที่ ๓.๑๑



รูปที่ ๓.๑๐ รูปแบบการเก็บระเบียนข้อมูลย่อยใน DBLK

การเก็บระเบียนข้อมูลย่อยใน DBLK จะมีสารสนเทศควบคุม (control information) ของทุก ๆ ระเบียนย่อยกำกับด้วย บันทึกอย่างต่อเนื่องกัน ถ้าที่ว่างใน DBLK ไม่พอสำหรับระเบียนย่อย จะนำระเบียนย่อยนั้นไปเก็บใน DBLK ถัดไป ส่วนที่ว่างนั้นจะใส่ค่าศูนย์ (Binary Zero) ไว้

สารสนเทศควบคุมมีความยาว ๔ ไบต์ ประกอบด้วย ๓ เขตข้อมูลคือ

๑) RL เป็นความยาวของระเบียนข้อมูลย่อย (รวมความยาวของสารสนเทศควบคุม) มีขนาด ๒ ไบต์

๒) CC เป็นรหัสคำสั่ง (command code) ที่ใช้สำหรับระเบียนข้อมูลย่อยนั้นในการนำระเบียนข้อมูลย่อยนั้นออกพิมพ์หรือเจาะบัตร (ถ้ามีค่าเป็น ๐๐ หมายถึงระเบียนข้อมูลย่อยนั้นเป็นข้อมูลเข้าหรือเป็นข้อมูลสถิติการใช้เครื่อง) มีขนาด ๑ ไบต์

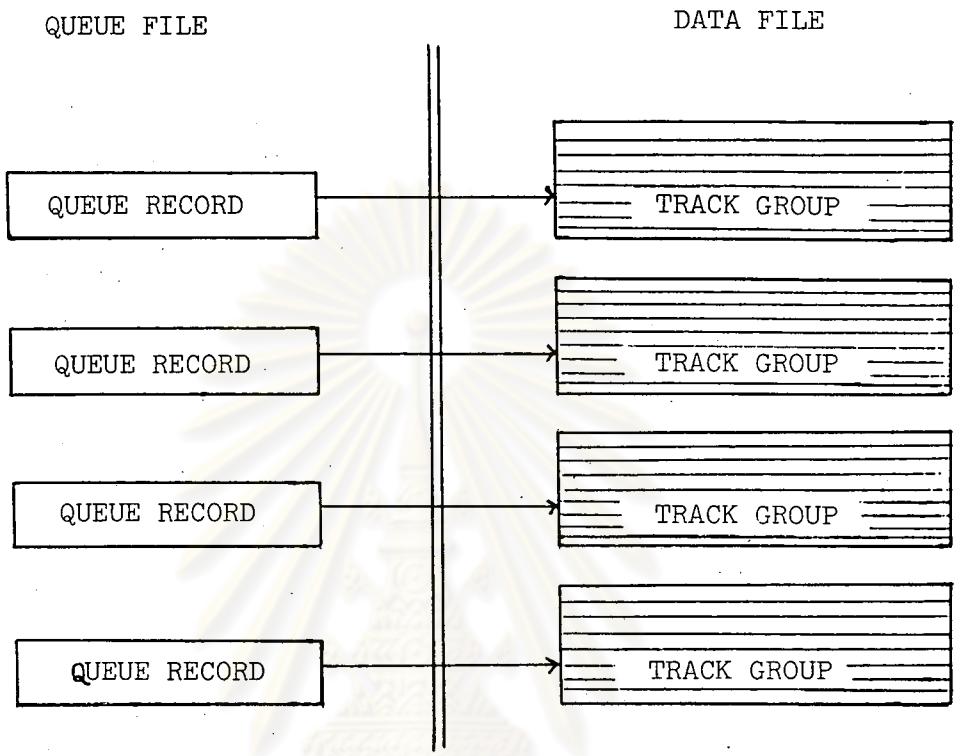
๓) GP บอกชนิดของระเบียนย่อย มีค่าเป็นไปได้ดังต่อไปนี้

00 = normal record	08 = break record
02 = 3540 data record	10 = end of block
04 = end of data	20 = end of 3540 data

มีขนาด ๑ ไบต์ โดยบิทที่ ๗ ของไบต์นี้ถ้ามีสถานะเป็น 'ON' จะบอกถึงว่ามีการขึ้นบรรทัดใหม่หรือขึ้นบัตรใหม่หลังการพิมพ์หรือเจาะข้อมูลย่อยนั้น

จำนวนของแทรค (Track) ที่ใช้ประกอบกันเป็นหน่วยข้อมูลที่มีความสัมพันธ์กับ ๑ ระเบียนข้อมูลของแฟ้มคิว เรียกว่า 'TRACK GROUP' ขนาดของแทรคกรุ๊ฟสามารถกำหนดขนาดได้ตั้งแต่ ๑ แทรค จนถึง ๑ ไซลินเดอ (Cylinder) เพาเวอร์/วีเอสที่สถาบันฯ กำหนดขนาดของ DBLK เป็น ๑๙๖๖ ไบต์ และขนาดของแทรคกรุ๊ฟเป็น ๑๒ แทรค

ลักษณะการเก็บข้อมูลในแฟ้มข้อมูลถูกควบคุมโดยระเบียนคิว (queue record) ในแฟ้มคิว ถ้าระเบียนคิวที่สัมพันธ์โดยตรงกับแทรคกรุ๊ฟเปลี่ยนแปลงไปใช้กับงานใหม่ ค่าในแทรคกรุ๊ฟ จะถูกทับด้วยข้อมูลของงานใหม่นั้นแทน โดยถือการสิ้นสุดของข้อมูลของงานด้วยค่าของระเบียนข้อมูลย่อยเป็น* $$ EØJ ส่วนข้อมูลอื่น ๆ ของงานเก่าที่ค้างในแทรคกรุ๊ฟนั้น จะไม่ถูกนำมาเป็นส่วนของงานใหม่เลย



รูปที่ ๓.๑๑ ความสัมพันธ์ระหว่างระเบียนคิวกับแทรคกรุ๊ฟ

๓.๔.๒ แฟ้มคิว (QUEUE FILE)

เป็นแฟ้มข้อมูลที่ใช้วิธีเข้าถึงข้อมูลโดยตรง ประกอบด้วยระเบียน ๓ ประเภทคือ

๑) ระเบียนแรก (Master Record) เป็นระเบียนแรกของแฟ้มคิว ใช้เก็บข้อมูลที่จำเป็นในการเริ่มให้เพาเวอร์/วีเอสทำงานแบบ warm start (warm start เป็นการเริ่มให้เพาเวอร์/วีเอสทำงานโดยใช้ค่าของพารามิเตอร์เดิมของการเริ่มทำงานครั้งที่แล้ว) มีความยาว ๑๘๔ ไบต์

๒) ระเบียนคิว (Queue Records) เป็นระเบียนที่ใช้เก็บข้อมูลต่าง ๆ ที่ผู้ใช้กำหนดขึ้น เพื่อใช้ในการควบคุมการทำงานสำหรับงานแต่ละงาน และเก็บดัชนีหาตำแหน่งข้อมูลของงานนั้นในแฟ้มข้อมูล มีความยาว ๑๕๒ ไบต์ ระเบียนประเภทนี้มีได้มากที่สุด ๑๐๒๔ ระเบียน เพาเวอร์/วีเอสที่สถาบันมี ๓๙๔ ระเบียน

งาน ๑ งานอาจประกอบด้วย ๑ ระเบียนคิวหรือมากกว่าเรียกว่า ๑ คิวเซ็ท (Queue set)

ระเบียนคิวและแทรคกรุ๊ฟที่สัมพันธ์กันของงานงานหนึ่งเรียกว่า ๑ คิวข้อมูลเข้า (Queue Entry)

๓) ระเบียนคิวว่าง (Free Queue Records) เป็นระเบียนประเภทเดียวกับระเบียนในแฟ้มคิวที่ไม่มีงานไหนใช้อยู่ ค่าในเขตข้อมูล QCQI (queue record identifier) จะมีค่าเป็น F และ QCDF (data file seek address) เก็บตำแหน่งข้อมูลของงานนั้นในแฟ้มข้อมูล และ QCNS (next queue record in set) เก็บตำแหน่งของระเบียนคิวถัดไปของคิวเซ็ทนี้ ค่าในเขตข้อมูลอื่น ๆ เก็บเป็นศูนย์ (ดูรูปที่ ๓.๑๗)

ความหมายของแต่ละเขตข้อมูลในแฟ้มคิวแสดงในภาคผนวก ค.

๓.๔.๓ แฟ้มข้อมูลทะเบียนสถิติการใช้เครื่อง (ACCOUNT FILE)

เป็นแฟ้มข้อมูลที่เก็บข้อมูลทางสถิติการใช้เครื่องของผู้ใช้ ใช้วิธีเข้าถึงข้อมูลโดยตรง ประกอบด้วยระเบียนประเภทต่าง ๆ ๔ ประเภทคือ

๑) ทะเบียนข้อมูลเข้า (Reader account records) เป็นระเบียนที่บันทึกรายละเอียดเกี่ยวกับงานต่าง ๆ ที่เข้ามาในระบบ มีความยาว ๕๗ ไบต์

๒) ทะเบียนงานพิมพ์ (List account records) เป็นระเบียนที่บันทึกรายละเอียดเกี่ยวกับงานต่าง ๆ ที่จัดออกทางเครื่องพิมพ์ มีความยาว ๗๑ ไบต์

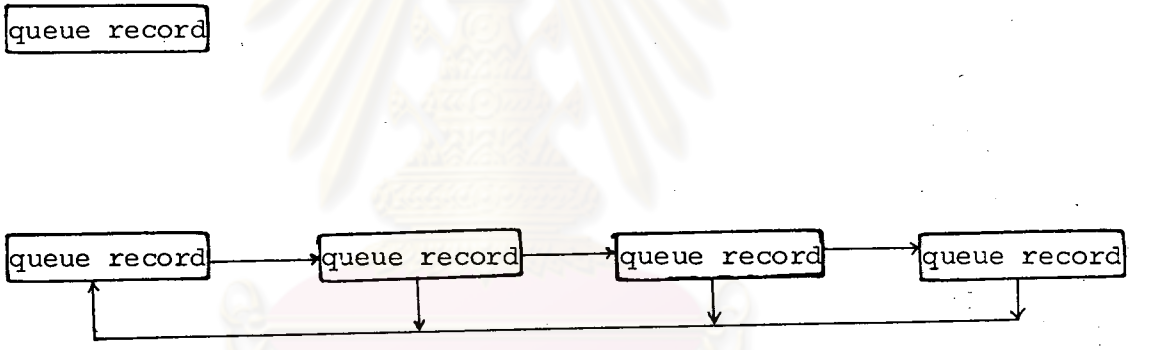
๓) ทะเบียนงานเจาะบัตร (Punch account records) เป็นระเบียนที่บันทึกรายละเอียดเกี่ยวกับงานต่าง ๆ ที่เจาะออกทางเครื่องเจาะบัตร มีความยาว ๖๗ ไบต์

๔) ทะเบียนงานประมวลผล (Execution account records) เป็นระเบียนที่บันทึกรายละเอียดเกี่ยวกับงานต่าง ๆ ที่ทำงานอยู่ในระบบ มีความยาว ๑๒๘ ไบต์

๓.๕ ลูกโซ่ (CHAIN) ในแฟ้มคิว

ลักษณะของลูกโซ่ในแฟ้มคิวมี ๓ ประเภทคือ

๓.๕.๑ ลูกโซ่ในคิวเซ็ท เป็นลูกโซ่ระหว่างระเบียนคิวของงาน ๑ งาน ที่จำเป็นต้องใช้แทรคกรุ๊ปมากกว่า ๑ ลักษณะของลูกโซ่เป็นแบบ Two way ring with one pointer to the head



รูปที่ ๓.๑๒ แสดงคิวเซ็ทที่ประกอบด้วย ๑ ระเบียนคิวจะไม่มีการสร้างลูกโซ่ เปรียบเทียบกับคิวเซ็ทที่ประกอบด้วย ๔ ระเบียนคิว

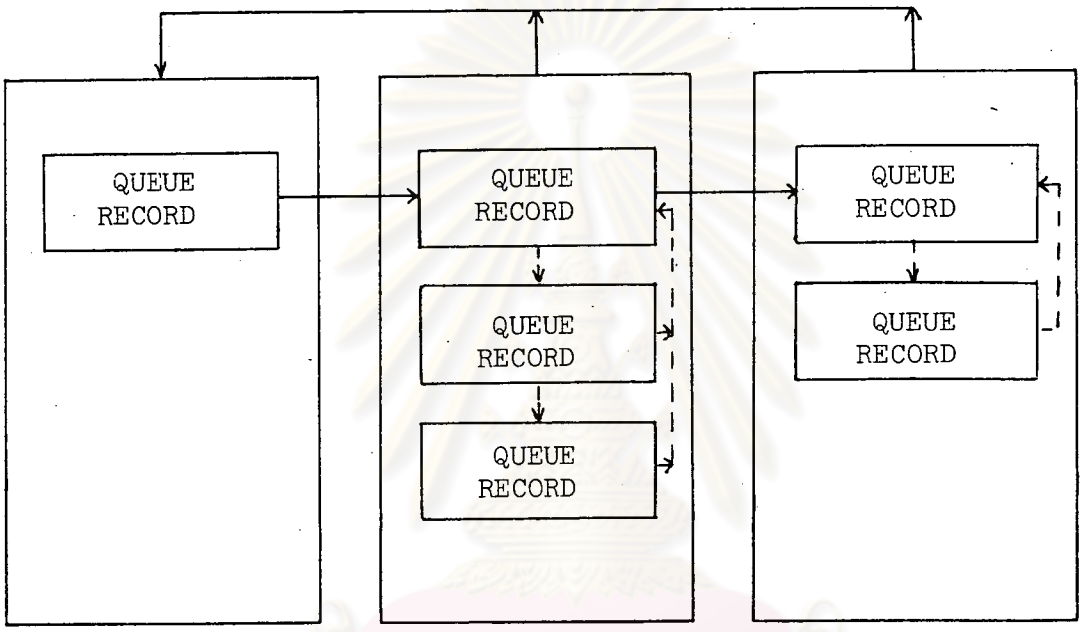
ค่าในเขตข้อมูลของระเบียนคิว ที่แสดงความสัมพันธ์ในคิวเซ็ทคือค่าของ QCQN

ถ้า QCQN = 0 หมายถึงระเบียนในแฟ้มคิวนี้เป็นระเบียนคิวสุดท้ายในคิวเซ็ทนี้

ถ้า QCQN ≠ 0 หมายถึงตำแหน่งของระเบียนคิวถัดไปของคิวเซ็ทนี้

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

๓.๕.๒ ลูกโซ่ระหว่างคิวเซ็ทของงานที่อยู่กลุ่มเดียวกัน เรียกว่า "ลูกโซ่กลุ่ม" (class chain)



รูปที่ ๓.๑๓ ลูกโซ่ระหว่างคิวเซ็ทที่ประกอบด้วย ๓ คิวเซ็ท

จะมีการสร้างลูกโซ่ระหว่างระเบียนคิวแรกของแต่ละคิวเซ็ท ในแบบ Two way ring with one pointer to the head เช่นเดียวกับลูกโซ่ในคิวเซ็ท

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ค่าของ เขตข้อมูลในระ เบียนคิวแรกของคิว เซ็ทที่มีความสัมพันธ์ในลูกโซ่ระหว่าง คิว เซ็ท คือค่าของ QCQP และ QCQN

QCQP = 0
QCQN = 0
หมายถึง ลูกโซ่กลุ่มนี้ประกอบด้วยคิว เซ็ทนี้ เพียง เซ็ท เดียว

QCQP = 0
QCQN ≠ 0
หมายถึง คิว เซ็ทนี้ เป็นคิว เซ็ทแรกในลูกโซ่กลุ่มนี้

QCQP ≠ 0
QCQN = 0
หมายถึง คิว เซ็ท เป็นคิว เซ็ทสุดท้ายในลูกโซ่กลุ่มนี้

QCQP ≠ 0
QCQN ≠ 0
หมายถึง คิว เซ็ทนี้อยู่ตรงกลางลูกโซ่กลุ่มนี้

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

The contents of field QCFS can be X'00' or X'01'.

X'01' indicates that the queue record is the first in the queue set.
X'00' indicates that the queue record is not the first in the queue set.

This determines the meaning of the fields QCNS, QCQP, and QCQN as follows:

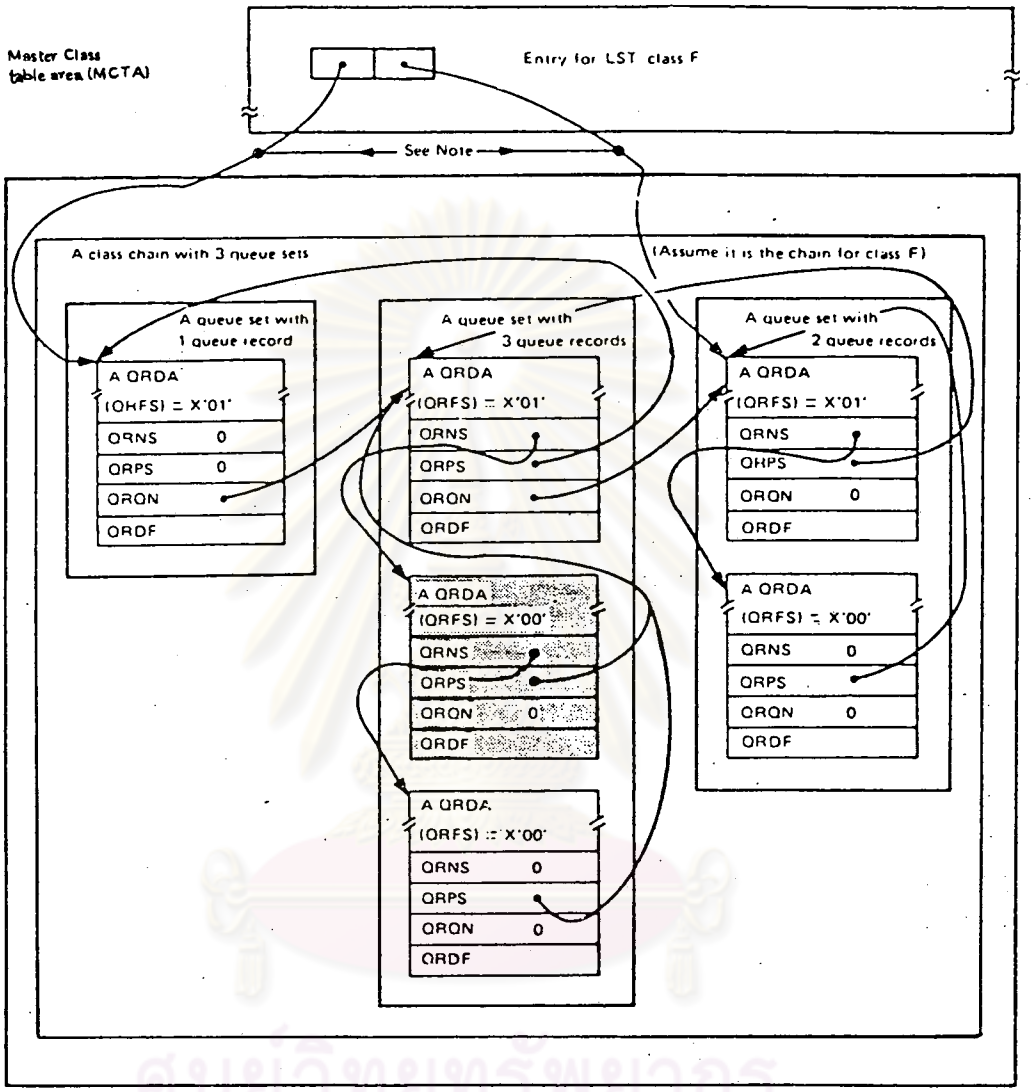
QCFS = X'00' (This queue record is not first in Q set)

Label of Field	Field contains zero	Field does not contain zero
QCNS	This queue record is the last in this queue set.	It is seek address of next queue record in this queue set.
QCQP	Cannot be zero.	It is the seek address of the first queue record in this queue set.
QCQN	Must be zero.	Must be zero.

QRFS = X'01' (This queue record is first in queue set)

Label of Field	Field contains zero	Field does not contain zero
QCNS	This queue record is last in same queue set.	It is the seek address of the next queue record in this queue set.
QCQP	This queue record is the first in the first queue set in this class chain.	It is the seek address of the first queue record of the previous queue set.
QCQN	This queue record is the first queue record of the last queue set in this class chain.	It is the seek address of the next queue set in this class chain.

ตารางที่ ๓.๑ ความหมายของเขตข้อมูลในระเบียนคิว ที่ใช้บอกสภาพของระเบียนคิวในลูกโซ่ต่าง ๆ

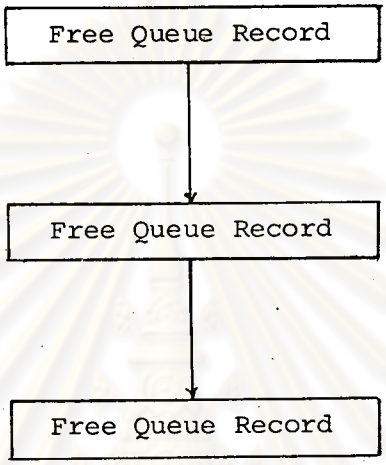


รูปที่ ๓.๑๔ ความสัมพันธ์ของค่าในเขตข้อมูลในระเบียนคิวในลูกโซ่กลุ่ม

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

I16503119

๓.๕.๓ ลูกโซ่ระหว่างระเบียนคิวว่างในเซ็ทคิวว่าง (Free Queue set)



รูปที่ ๓.๑๕ เซ็ทคิวว่าง

เซ็ทคิวว่างนี้แตกต่างจากคิว เซ็ท เนื่องจากไม่มีตัวชี้ (pointer) ชี้กลับไปยังระเบียนคิวแรกของเซ็ทคิวว่าง ลูกโซ่นี้จึงมีลักษณะเป็น link list ค่าของ QCQP, QCQN ในระเบียนคิวจะมีค่า = 0 เมื่อใดที่ระเบียนคิวในเซ็ทคิวว่างถูกเรียกใช้จนหมด งานของเพาเวอร์/วีเอสที่เรียกใช้ระเบียนคิวว่างในขณะนั้นจะหยุดรอจนกว่าจะมีการคืนระเบียนคิวให้มาอยู่ในเซ็ทคิวว่างจากงานส่วนอื่นของเพาเวอร์/วีเอสจึงจะทำงานต่อไปได้

```
                                                    QCQI
                                                     ↓
CHAR  22/07/83   %   %                  ABJBBML1 'L  00E  Y3                                  D              '
ZONE  FF6FF6FF019601964444444444444444CCDCCDDF07D14FFC00EF000300F04444000000000020000000C00000001E001E0407
NUMR  22107183204C204C0000000000000000121224310D303005008300070 1F10000000060000 0D0000004100000050005002 0D
      1...5...10...15...20...25...30...35...40...45...50...55...60...65...70...75...80...85...90...95.....

                                  ←QCNS→|←QCQP→|←QCQN→|←QCDF→|
CHAR
ZONE  4444000000000000000000009000|00009000|00000000|00002000|
NUMR  0000010000000000000040014078|40014038|00000000|8000A003|
    101...5...10...15...20...25...|30...35...|40...45...|50...|
```

รูปที่ ๓.๑๖ รูปแบบของข้อมูลที่เก็บอยู่ในระเบียนในคิวของลูกโซ่กลุ่ม

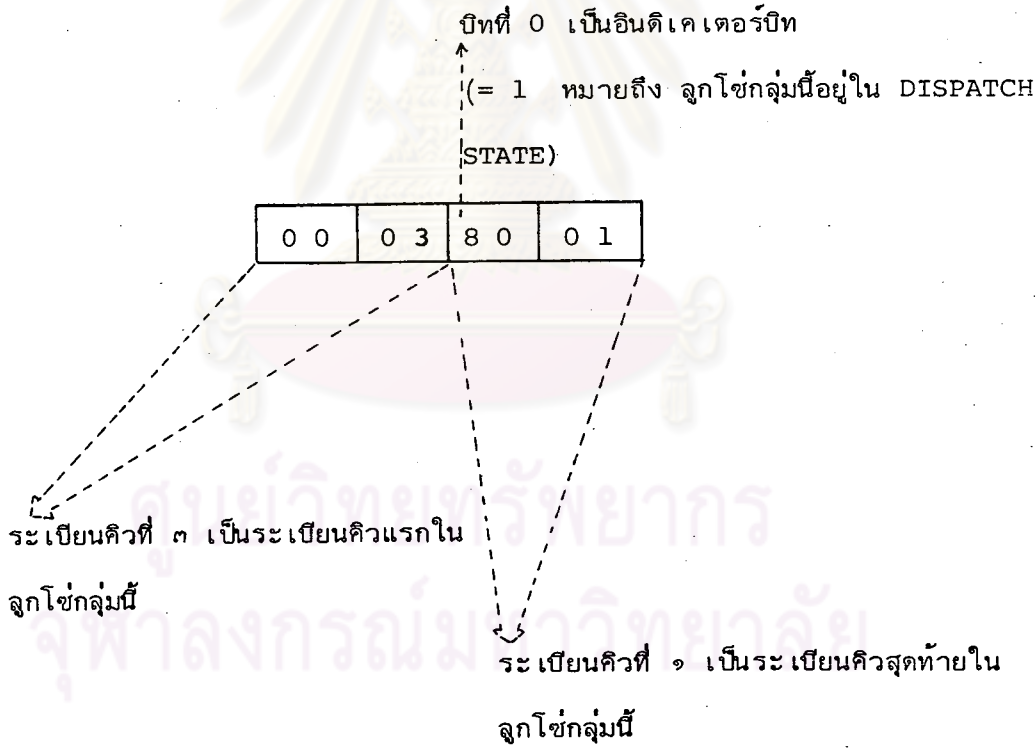
```
                                                                  QCQI
                                                                   ↓
CHAR                                                               F
ZONE  0000000000000000000000000000000000000000000C000000000000000000000000000000000000000000000000000000000000
NUMR  0000000000000000000000000000000000000000000600000000000000000000000000000000000000000000000000000000000000
      1...5...10...15...20...25...30...35...40...45...50...55...60...65...70...75...80...85...90...95.....
```

```
                                  |←QCNS→|←QCQP→|←QCQN→|←QCDF→|
CHAR
ZONE  00000000000000000000000090000000000000000000020000
NUMR  00000000000000000000400140220000000000000000800050001
    101...5...10...15...20...25...30...35...40...45...50...|
```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

รูปที่ ๓.๑๗ รูปแบบข้อมูลที่เก็บอยู่ในระเบียนคิวของเซ็ทคิวว่าง

เพื่อให้การทำงานเป็นไปอย่างรวดเร็ว ในโปรแกรมเพาเวอร์/วีเอสจะมีพื้นที่ถาวร (Permanent area) ส่วนหนึ่งใช้กับ DISK MANAGEMENT CONTROL BLOCK (DMB) ซึ่งในบลอคควบคุมตัวนี้จะมี master class table area ของแต่ละกลุ่มของงานอ่าน,งานพิมพ์และงานเจาะบัตร พื้นที่ส่วนนี้ใช้บอกว่าแต่ละกลุ่มนี้มีจุดเริ่มต้นและสิ้นสุดของลูกโซ่ที่ระเบียนคิวใด โดยเก็บเป็นค่าของตำแหน่งของระเบียนคิวที่สัมพันธ์กับระเบียนคิวแรกในคิวไฟล์ และมีอินดิเคเตอร์บิท (indicator bit) บอกว่าลูกโซ่กลุ่มนั้นอยู่ใน DISPATCH STATE หรือไม่ ใช้เป็นข่าวสารของ TASK CONTROL BLOCK ในการทำงานของเพาเวอร์/วีเอส



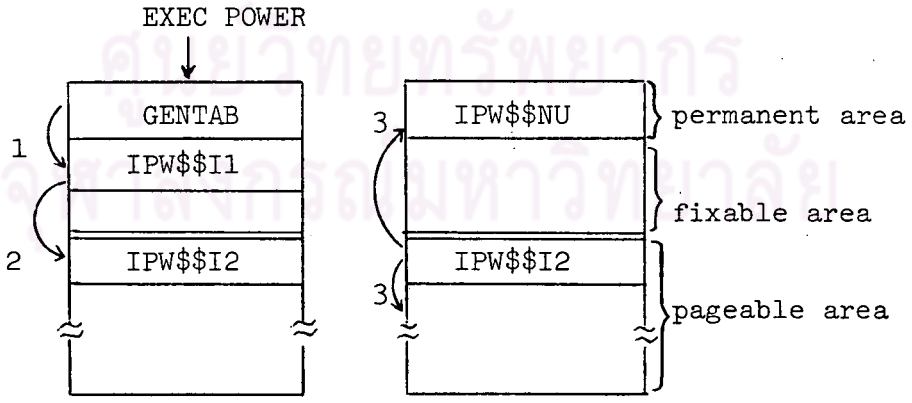
รูปที่ ๓.๑๘ ข้อมูลที่เก็บใน master class table ของแต่ละกลุ่ม

ทุกครั้งที่มีการเริ่มให้เพาเวอร์/วีเอสทำงาน (initialize POWER) มีเฟสที่เกี่ยวข้อง ๓ เฟสคือ

๑) POWER เฟสนี้ประกอบด้วยคอนโทรลบลอคที่จำเป็นในการโหลดเฟส IPW$$I1 และตัวโหลด (Loader) ตัวเล็ก ๆ

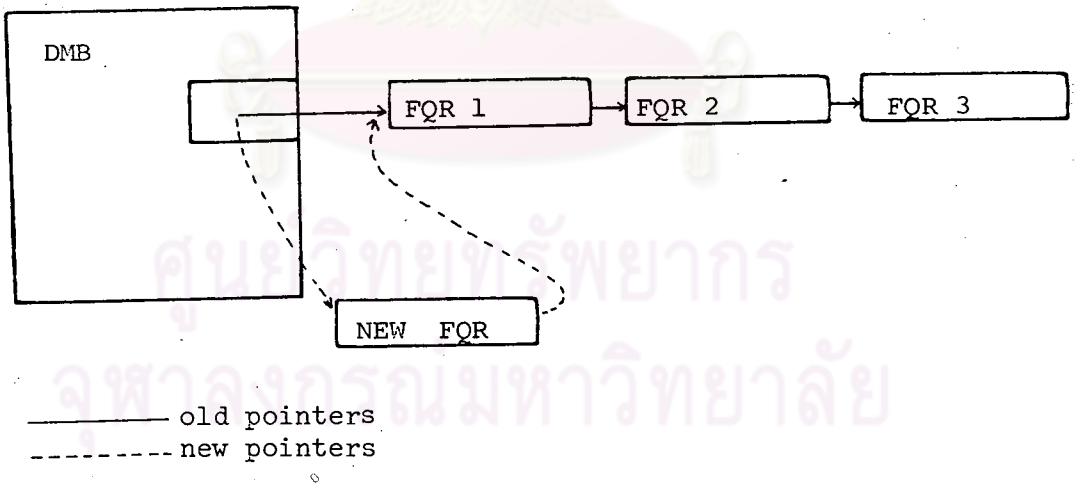
๒) IPW$$I1 เฟสนี้จะถูกโหลดโดยเฟส POWER เข้ามาอยู่ในหน่วยความจำเสมือนที่ต่อจากเฟส POWER ใช้ในการโหลดเฟส IPW$$I2 ให้เข้ามาอยู่ในหน่วยความจำเสมือนส่วนที่เป็น pageable area

๓) IPW$$I2 เฟสนี้เมื่อถูกโหลดเข้ามาจะทำหน้าที่ให้ข่าวสารแก่บลอคควบคุมต่าง ๆ เช่นค่าต่าง ๆ ใน master class table area ของ DISK MANAGEMENT CONTROL BLOCK แล้วทำการโหลดเฟสที่จำเป็นเช่น IPW$$NU เข้ามาในหน่วยความจำเสมือนส่วนที่เป็นพื้นที่ถาวร นอกจากนั้นยังทำการโหลดเฟสที่จำเป็นในการทำงานของแต่ละงาน ให้เข้ามาอยู่ในหน่วยความจำเสมือนส่วนที่เป็น pageable area



รูปที่ ๓.๑๙ ขั้นตอนการเริ่มให้เพาเวอร์/วีเอสทำงาน

ดังนั้น ทุกครั้งที่มีการเริ่มให้เพาเวอร์/วีเอสทำงาน เฟส IPW$$I2 จะทำการสำรวจแฟ้มคิว เพื่อหาระเบียนคิวแรก และสุดท้ายของแต่ละลูกโซ่กลุ่ม และหาระเบียนคิวว่างระเบียนแรกในเซ็ทคิวว่าง ข้อมูลเหล่านี้จะเก็บในหน่วยความจำเสมือนส่วนที่เป็น master class table area (ระบบหน่วยความจำของเพาเวอร์/วีเอสแสดงในภาคผนวก ข.) ใช้ในการควบคุมการใช้แฟ้มคิว ลักษณะการควบคุมการใช้แฟ้มคิวนี้ เป็นการควบคุมโดยใช้เนื้อที่ในหน่วยความจำเสมือนของพาติชันที่เพาเวอร์/วีเอสทำงานอยู่ เมื่องานใดที่ทำงานอยู่ของเพาเวอร์/วีเอสปลดปล่อยระเบียนคิวที่ใช้อยู่ จะมีการเปลี่ยนแปลงตัวชี้ในระเบียนคิวนั้นให้หลุดไปจากลูกโซ่กลุ่มที่อยู่ ไปอยู่ในเซ็ทคิวว่าง และทำการใส่ค่าศูนย์ให้แก่เขตข้อมูลของระเบียนคิวไบต์ที่ ๑-๑๒๐, ๑๒๔-๑๔๔ ส่วน QCNS จะเก็บตำแหน่งของระเบียนคิวว่างถัดไป ลักษณะการต่อระเบียนคิวที่ถูกปลดปล่อยให้เซ็ทคิวว่าง จะต่อเข้าทางตำแหน่งแรกของลูกโซ่ของ link list (ตำแหน่งแรกของลูกโซ่ของ link list เก็บอยู่ใน DISK MANAGEMENT CONTROL BLOCK)



รูปที่ ๓.๒๐ การต่อระเบียนคิวเข้าในเซ็ทคิวว่าง

การปลดปล่อยระเบียนคิวนี้ จะทำการปลดปล่อยทั้งคิวเซ็ทและการปลดปล่อยคิวเซ็ทที่เป็นคิวเซ็ทแรกและสุดท้ายของลูกโซ่กลุ่มมีผลให้ต้องปรับค่าตำแหน่งจุดเริ่มและจุดสิ้นสุดของลูกโซ่กลุ่มนั้นใน master class table ของ DMB ด้วย

จากรายละเอียดของเขตข้อมูลในระเบียนคิว และลักษณะการควบคุมการใช้แฟ้มคิวของเพาเวอร์/วีเอส จะนำมาใช้เป็นข้อพิจารณาในการดึงแฟ้มข้อมูลของเพาเวอร์/วีเอสของงานที่ต้องการที่เก็บอยู่ในจานแม่เหล็กไปจัดเก็บไว้ในเทปแม่เหล็กในลักษณะที่สามารถนำมาประมวลผลภายใต้การควบคุมของเพาเวอร์/วีเอส

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย