

โปรแกรมที่เกี่ยวข้องกับแฟ้มข้อมูลแบบอื่น

โปรแกรมที่เกี่ยวข้องกับแฟ้มข้อมูลแบบอื่นของเครื่องคาคาโทปที่มีอยู่หลายโปรแกรม เช่น EDIT/CMD, REFORMAT/CMD, LIST/CMD, TAPE/CMD, SAPP/CMD INDEX/CMD, SORT/CMD, EDIT/DBC และโปรแกรมที่ผู้ใช้เขียนขึ้นเอง ความต้องการเป็นต้น ส่วนโปรแกรมอื่นที่ไม่ได้กล่าวถึงจะสามารถศึกษารายละเอียดได้จากหนังสือ DISK OPERATING SYSTEM User's Guide⁽⁴⁾

3.1 โปรแกรมที่ใช้ในการสร้างแฟ้มข้อมูลแบบอื่น (EDIT/CMD)

EDIT/CMD ถูกพัฒนาขึ้นมาเพื่อใช้ในการสร้างและปรับปรุงแก้ไขแฟ้มข้อมูลแบบอื่นโดยเฉพาะ ดังนั้นแฟ้มข้อมูลที่เป็น Non space compression file นำมาไว้กับโปรแกรม EDIT /CMD จะทำให้แฟ้มข้อมูลนี้ถูกเปลี่ยนไปเป็น Space compression file นี้เนื่องจากแฟ้มข้อมูลที่น่ามาไว้กับโปรแกรม EDIT /CMD นี้จะต้องถูกบันทึกอยู่ในงานแม่เพื่ออีกการทำ Space compression ของโปรแกรม EDIT/CMD จึงทำให้ใช้เนื้อที่ในงานแม่เพื่ออีกน้อยลงเท่าที่จะสามารถทำได้ ข้อจำกัดของโปรแกรม EDIT /CMD ก็คือขนาดความยาวของเรกคอร์ดจะคงไม่เกิน 79 ตัวอักษร ดังนั้นผู้ใช้สามารถที่จะใช้โปรแกรม EDIT/CMD ในการเตรียม Source โปรแกรมภาษาต่าง ๆ เช่น แอสเซมบลี, คาคาโทป, เบสิก, โคบอล, อาร์ที จี เป็นต้น หรือใช้ในการเตรียมแฟ้มข้อมูลที่มีขนาดความยาวของเรกคอร์ดไม่เกิน 79 ตัวอักษร

3.1.1 การเรียกใช้โปรแกรม EDIT/CMD มีรูปแบบการเรียกใช้ดังนี้

EDIT <f₁> [₁<f₂>] [₁<f₃>] [;<options>]

<f₁> เป็นชื่อของแฟ้มข้อมูลหรือชื่อของการสร้างใหม่หรือเป็นชื่อของแฟ้มข้อมูลที่มีอยู่แล้ว แต่ต้องการนำมาแก้ไขใหม่ ถ้าหากผู้ใช้ไม่กำหนด Extension ให้จะถูก default เป็น "TXT"

<f₂> เป็นชื่อของแฟ้มข้อมูลที่ผู้ใช้เป็น Work file ในระหว่างการทำงานของ

โปรแกรม EDIT/CMD ถ้าหากผู้ใช้ไม่กำหนดชื่อแฟ้มข้อมูลหรือ/หรือ Extension ให้ จะถูก default เป็น "SCRATCH/TXT"

[<F3>] เป็นชื่อของแฟ้มข้อมูลที่ใช้เป็น Configuration Overlay จะทำหน้าที่ เก็บรายละเอียดต่าง ๆ ของตัวพารามิเตอร์ที่ผู้ใช้กำหนดขึ้น อาทิเช่น การตั้ง Tab เป็นต้น ถ้าหากผู้ใช้ไม่กำหนดชื่อแฟ้มข้อมูลหรือ/หรือ Extension ให้โปรแกรม EDIT/CMD จะ default ให้เป็น "EDIT/OVI"

3.1.2 การเตรียมข้อมูล (Data Entry) ข้อมูลที่ผู้ใช้ต้องการจะป้อนคือของบน บรรทัดสุดท้ายของจอสำหรับแสดงข้อมูลซึ่งบรรทัดสุดท้ายนี้จะเรียกว่า "บรรทัดคำสั่ง (Command line)" เมื่อผู้ใช้คีย์ข้อมูลเรียบร้อยแล้วให้กด ENTER คีย์ ข้อมูลที่ปรากฏอยู่บนจอทั้งหมดจะถูกเลื่อนขึ้น (Roll up) ไป 1 บรรทัด ซึ่งจะทำให้บรรทัดคำสั่งว่างและ ตัวเคอร์เซอร์ (Cursor) จะวิ่งไปยังตำแหน่งบรรทัดคำสั่ง และพร้อมที่จะรับข้อมูลเรคคอร์ดต่อไป การแก้ไขข้อมูลที่ปรากฏอยู่บนบรรทัดคำสั่งสามารถทำได้โดยใช้ BACK SPACE คีย์ หรือ CANCEL คีย์ ส่วนการแก้ไขข้อมูลที่ปรากฏอยู่บนจอนอกเหนือจากบรรทัดคำสั่งก็สามารถทำได้โดยจะกล่าวถึงในหัวข้อ 3.1.4

3.1.3 การเรียกข้อมูล (Data Retrieval) เมื่อผู้ใช้ต้องการเรียกข้อมูลจากแฟ้มข้อมูล <F1> มาใช้สามารถทำได้โดยกด KEYBOARD คีย์และ DISPLAY คีย์พร้อม ๆ กันทั้ง 2 คีย์ จะทำให้ข้อมูลถูกอ่านจากแฟ้มข้อมูล <F1> และนำมาแสดงบนบรรทัดคำสั่ง จากนั้นจะทำการเลื่อนขึ้นไป 1 บรรทัด การอ่านและแสดงบนบรรทัดคำสั่งจะหยุดเมื่อผู้ใช้เลือก KEYBOARD คีย์และ DISPLAY คีย์หรือข้อมูลในแฟ้มข้อมูล <F1> ถูกอ่านจนหมดแล้ว ในกรณีที่ต้องการเรียกข้อมูลเพียง 1 เรคคอร์ด ให้กด SHIFT คีย์และ DEL คีย์

3.1.4 รูปแบบของคำสั่งที่ใช้ในโปรแกรม EDIT/CMD (EDITOR COMMAND FORMAT) ข้อมูลต่าง ๆ ที่ปรากฏอยู่บนจอนอกเหนือจากบรรทัดคำสั่ง คือตั้งแต่บรรทัดที่ 1 ถึงบรรทัดที่ 11 สามารถที่จะทำการ Edited ได้โดยอาศัยคำสั่งและตัวพอยต์เคอร์ ตัวพอยต์เคอร์มีลักษณะเป็น ">" ตัวพอยต์เคอร์นี้จะอยู่ทางด้านซ้ายมือของจอ จุดประสงค์ของการมีพอยต์เคอร์ ก็เพื่อใช้ในการขยับบรรทัดที่ต้องการแก้ไขหรือเปลี่ยนแปลง การเลื่อนพอยต์เคอร์ขึ้นหรือลง

สามารถทำได้โดยอาศัย KEYBOARD คีย์ และ DISPLAY คีย์ เมื่อกด KEYBOARD คีย์หรือ DISPLAY คีย์ แต่ละครั้งจะทำให้ตัวพอยท์เตอร์เลื่อนขึ้นหรือลง 1 บรรทัด เมื่อต้องการเลื่อนตัวพอยท์เตอร์ขึ้นให้กด KEYBOARD คีย์, ต้องการเลื่อนตัวพอยท์เตอร์ลงให้กด DISPLAY คีย์ ในการคีย์คำสั่งจะต้องเริ่มต้นด้วยตัว colon (:) ในตำแหน่งที่หนึ่งของบรรทัดคำสั่ง ตามด้วยตัวอักษรที่ใช้แทนคำสั่งซึ่งมีความยาวได้ไม่เกิน 3 ตัวอักษร และตามด้วยตัวพารามิเตอร์ต่าง ๆ สำหรับบางคำสั่งที่ต้องการใช้ตัวพารามิเตอร์ โดยสรุปรูปแบบของคำสั่งสามารถแสดงได้ ดังนี้

: XXX parameters

3.1.5 คำสั่งที่ใช้ในโปรแกรม EDIT/CMD (EDITOR Commands) คำสั่งต่าง ๆ ที่ใช้ในโปรแกรม EDIT/CMD ผู้ใช้สามารถทำความเข้าใจได้ง่ายและสะดวกต่อการใ้สิ่งที่ผู้ใช้ต้องคำนึงก็คือ คำสั่งต่าง ๆ ที่จะกล่าวถึงต่อไปนี้ ต้องใช้ร่วมกับตัวพอยท์เตอร์ " > " เสมอ คำสั่งที่ไว้หัว ๆ ไป มีดังต่อไปนี้

:D - DELETE เป็นคำสั่งที่ใช้ในการลบหรือลบจนจบบรรทัดซึ่งถูกตัวพอยท์เตอร์ชี้อยู่ทั้งหนึ่งบรรทัด ตัวเคอร์เซอร์จะวิ่งไปปรากฏอยู่ที่ตำแหน่งที่หนึ่งของบรรทัดนั้น เพื่อให้ผู้ใช้ทำการคีย์หรือเอาใหม่เข้าไป เมื่อคีย์หรือเอาใหม่เรียบร้อยแล้ว ให้กด ENTER คีย์ ตัวเคอร์เซอร์จะวิ่งกลับมายังตำแหน่งที่หนึ่งของบรรทัดคำสั่ง ถ้าหากผู้ใช้ไม่ต้องการคีย์หรือเอาใหม่เข้าไปแทนหรือเอาเก่าซึ่งถูกลบทิ้งไป ให้กด ENTER คีย์ ตัวเคอร์เซอร์จะวิ่งกลับมายังตำแหน่งที่หนึ่งของบรรทัดคำสั่ง

:I - INSERT เป็นคำสั่งที่ใช้ในการแทรกหรือเอาเข้าไปใหม่ทั้งบรรทัด หรืออยู่ในบรรทัดที่อยู่เหนือจากบรรทัดที่ถูกชี้โดยตัวพอยท์เตอร์รวมทั้งหรืออยู่ในบรรทัดที่ถูกชี้โดยตัวพอยท์เตอร์จะถูกเลื่อนขึ้น (roll up) 1 บรรทัด ตัวเคอร์เซอร์จะวิ่งไปยังตำแหน่งที่หนึ่งของบรรทัดที่ถูกชี้โดยตัวพอยท์เตอร์ เพื่อให้ผู้ใช้ทำการคีย์หรือเอาที่ตรงการแทรกออกไป เมื่อผู้ใช้กด ENTER คีย์ ตัวเคอร์เซอร์จะวิ่งกลับมายังตำแหน่งที่หนึ่งของบรรทัดคำสั่ง

:A - APPEND คำสั่งนี้จะทำให้หรือเอาทั้งบรรทัดซึ่งถูกตัวพอยท์เตอร์ชี้อยู่ ถูกลอก (Copies) ลงมาไว้ยังบรรทัดคำสั่งและหรือเอาจนจบจะถูกเลื่อนขึ้นไปหนึ่งบรรทัดหรืออยู่ในบรรทัดที่ถูกลอกจะไม่เปลี่ยนแปลงหลังจากทำคำสั่งนี้ ตัวเคอร์เซอร์จะวิ่งกลับมายังตำแหน่งที่หนึ่งของบรรทัดคำสั่ง

:C - COPY ทำให้รอมทั้งบรรทัดซึ่งถูกตัวหยุดเคอร์เซอร์อยู่ถูกออกมาไว้ยังบรรทัดคำสั่ง จากนั้นจะทำการลบรอมบรรทัดซึ่งถูกตัวหยุดเคอร์เซอร์อยู่ทั้งบรรทัดและทำการเลื่อนรอมบรรทัดขึ้นไปอีกหนึ่งบรรทัด ตัวเคอร์เซอร์จะวิ่งไปยังตำแหน่งที่หนึ่งของบรรทัดซึ่งถูกชี้โดยตัวหยุดเคอร์เซอร์ เพื่อให้ผู้ใช้ทำการคีย์รอมชุดใหม่เข้าไปแทน เมื่อกด ENTER คีย์ ตัวหยุดเคอร์เซอร์จะเลื่อนลงมาที่บรรทัดถัดไป และตัวเคอร์เซอร์จะวิ่งกลับมาถึงตำแหน่งที่หนึ่งของบรรทัดคำสั่ง ถ้าไม่ต้องการคีย์รอมชุดใหม่เข้าไปแทน ให้กด ENTER คีย์

:T - TAB set คำสั่งนี้ใช้ในการตั้ง Tab เพื่อสะดวกในการเตรียมรอม เมื่อไรคำสั่งนี้จะปรากฏแถวของตัวเคอร์เซอร์ในหน้าตำแหน่งแต่ละตำแหน่งตั้งแต่ตำแหน่งที่ 1 ถึงตำแหน่งที่ 79 ขึ้นบนบรรทัดคำสั่ง เพื่อไร้หน้าตำแหน่งต่าง ๆ ที่ผู้ใช้ต้องการตั้ง Tab และตัวเคอร์เซอร์จะอยู่ที่ตำแหน่งที่หนึ่งบนบรรทัดคำสั่ง เพื่อรอให้ผู้ใช้ตั้ง Tab วิธีการตั้งก็โดยการที่ผู้ใช้เลื่อนตัวเคอร์เซอร์ไปยังตำแหน่งที่ต้องการตั้ง Tab จากนั้นจึงคีย์ตัวอักษรใดก็ได้ลงบนตำแหน่งนั้น การเลื่อนเคอร์เซอร์สามารถทำได้โดยการกด SPACE คีย์ไปจนถึงตำแหน่งที่ต้องการตั้ง Tab การตั้ง Tab สามารถทำได้มากที่สุด 10 ตำแหน่ง

:E - EKD คำสั่งนี้จะทำให้รอมส่วนที่ยังคงเหลืออยู่ในพื้นที่รอม $\langle r_1 \rangle$ ถูกออก (Appended) ไปไว้ยังพื้นที่รอม $\langle r_2 \rangle$ และจากนั้นรอมที่อยู่ในพื้นที่รอม r_2 จะถูกออก (Copied) กลับไปยังพื้นที่รอม $\langle r_1 \rangle$ อีกครั้ง ในกรณีที่ผู้ใช้ออกคำสั่งนี้ไปแล้วและต้องการจะยกเลิกคำสั่งนี้ ให้กด KEYBOARD คีย์และ DISPLAY คีย์พร้อม ๆ กัน

:EO - EOF with display เป็นคำสั่งที่ใช้ในการสั่งให้นำรอมที่มีอยู่ทั้งหมดจากพื้นที่รอม $\langle r_1 \rangle$ มาแสดงบนจอภาพ ผู้ใช้รอจต้องการยกเลิกการทำงานของคำสั่งนี้ เมื่อใดก็ได้โดยการกด KEYBOARD คีย์และ DISPLAY คีย์พร้อม ๆ กัน

:E* - EOF without display เป็นคำสั่งที่ใช้ในการสั่งให้นำรอม 11 เรคคอร์ดสุดท้ายของพื้นที่รอม $\langle r_1 \rangle$ มาแสดงบนจอภาพ เมื่อต้องการยกเลิกการทำงานของคำสั่งนี้ ให้กด KEYBOARD คีย์และ DISPLAY คีย์พร้อม ๆ กัน

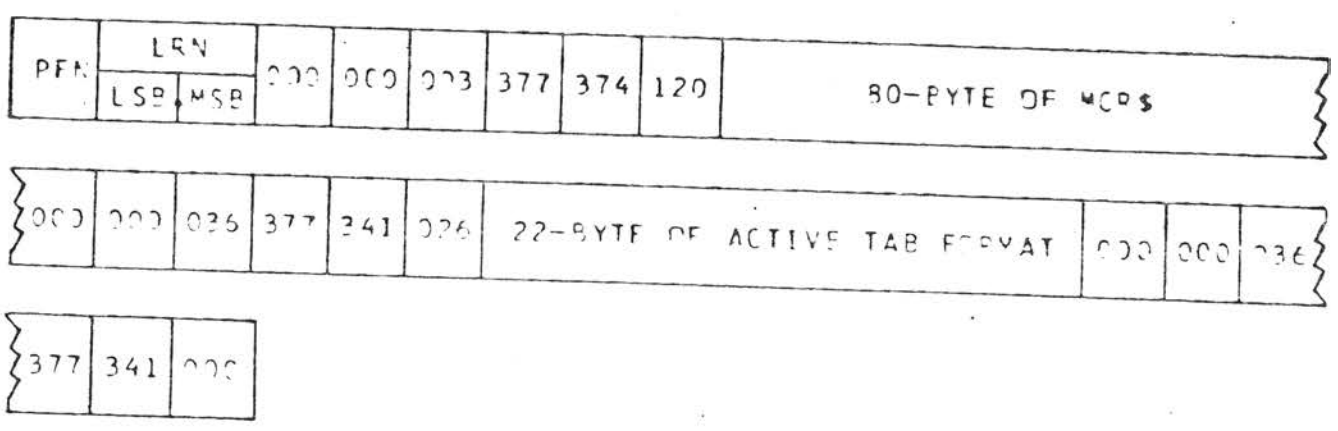
: M (old text) < (new text) - MODIFY เมื่อใช้คำสั่งนี้จะทำให้ข้อมูลที่เป็น old text ถูกแทนด้วยข้อมูลใหม่ที่เป็น new text

: SC - SCRATCH above เป็นคำสั่งที่ใช้ในการลบข้อมูลจากบรรทัดบนสุดของจอภาพจนถึงบรรทัดที่ตัวหยุดเคอร์เซอร์อยู่ (รวมทั้งบรรทัดที่ตัวหยุดเคอร์เซอร์อยู่ด้วย) ตัวเคอร์เซอร์จะวิ่งไปอยู่ที่ตำแหน่งบรรทัดที่ตัวหยุดเคอร์เซอร์ เพื่อให้ผู้ใช้กรอกข้อมูลใหม่เข้าไปแทน จากนั้นให้กด ENTER คีย์ หรือถ้าไม่ต้องการกรอกข้อมูลใหม่เข้าไปแทนให้กด ENTER คีย์ ในตำแหน่งบรรทัดที่มีตัวหยุดเคอร์เซอร์จะทำให้ตัวเคอร์เซอร์วิ่งกลับไปยังตำแหน่งบรรทัดคำสั่ง

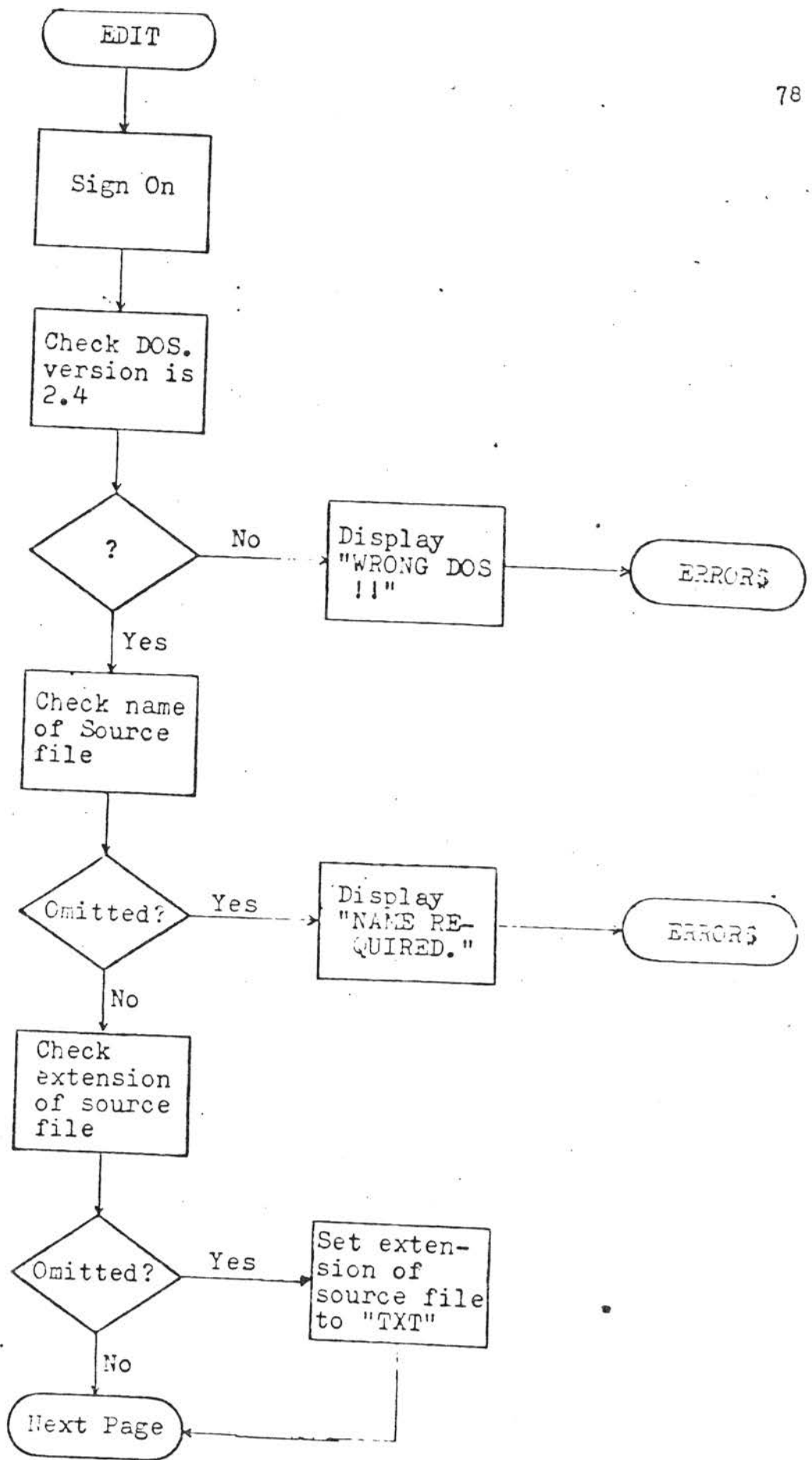
: SB - SCRATCH below ใช้ในการลบข้อมูลจากบรรทัดที่ตัวหยุดเคอร์เซอร์ขึ้นถึงบรรทัดที่ 11 (รวมทั้งข้อมูลที่อยู่ในบรรทัดที่ 11 จะถูกขจัดด้วย) ตัวเคอร์เซอร์จะวิ่งไปอยู่ที่ตำแหน่งบรรทัดที่ตัวหยุดเคอร์เซอร์อยู่ เพื่อให้ผู้ใช้กรอกข้อมูลใหม่เข้าไป ซึ่งมีวิธีการเหมือนกับที่กล่าวไว้ในคำสั่ง : SC

: < tab key > ระบุให้โปรแกรม EDIT/CMD ทราบว่าผู้ใช้ต้องการใช้ตัวอักษรใดเป็น Tab ตัวอักษรที่จะใช้เป็น Tab ต้องไม่ใช่ตัว A - Z, 0-9, : หรือ ENTER คีย์ หรือ Special คีย์ต่าง ๆ

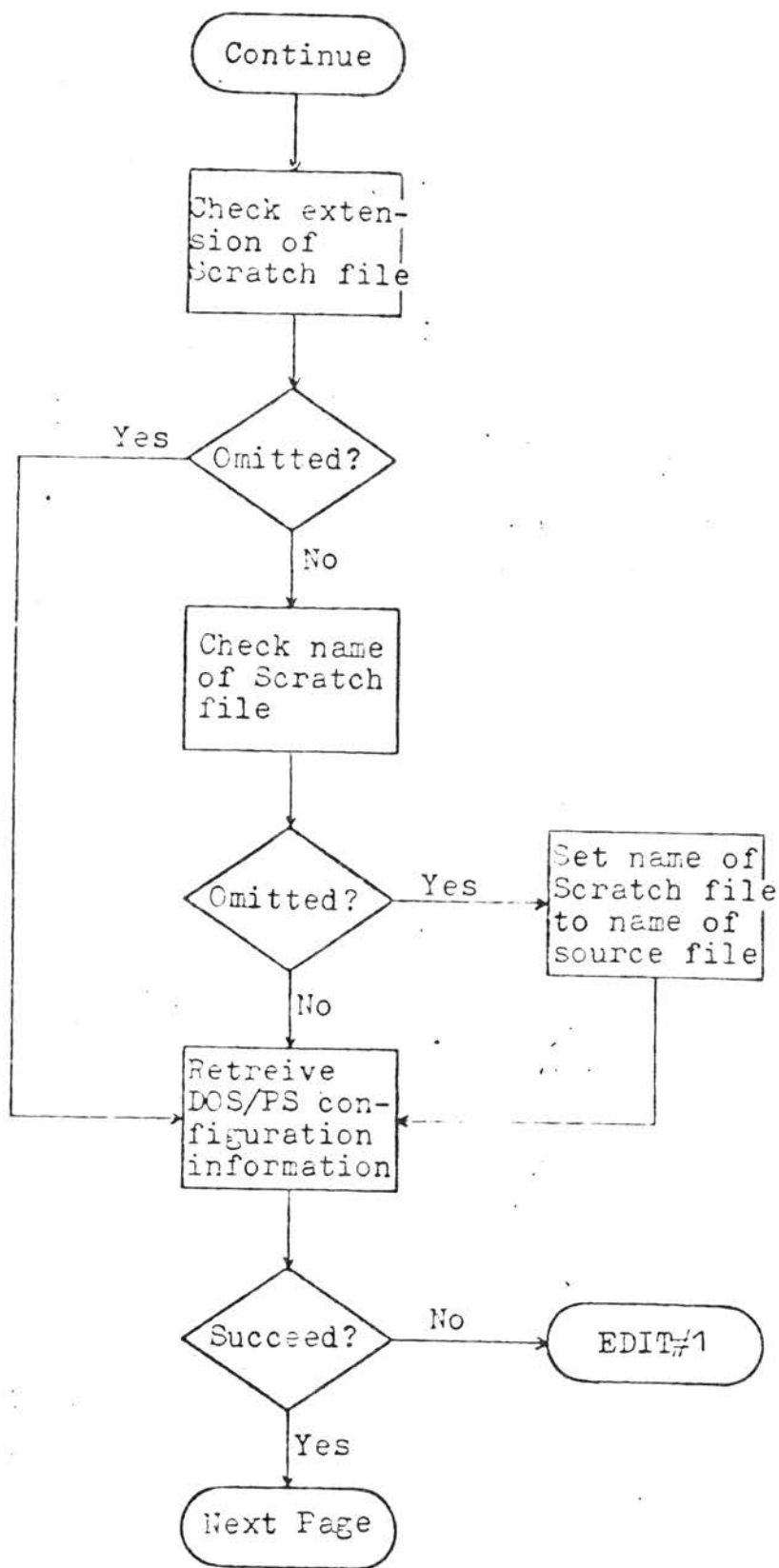
3.1.6 ขั้นตอนการทำงาน ขั้นตอนการทำงานของโปรแกรม EDIT/CMD ใกล้เคียงไว้ในรูปที่ 3.2



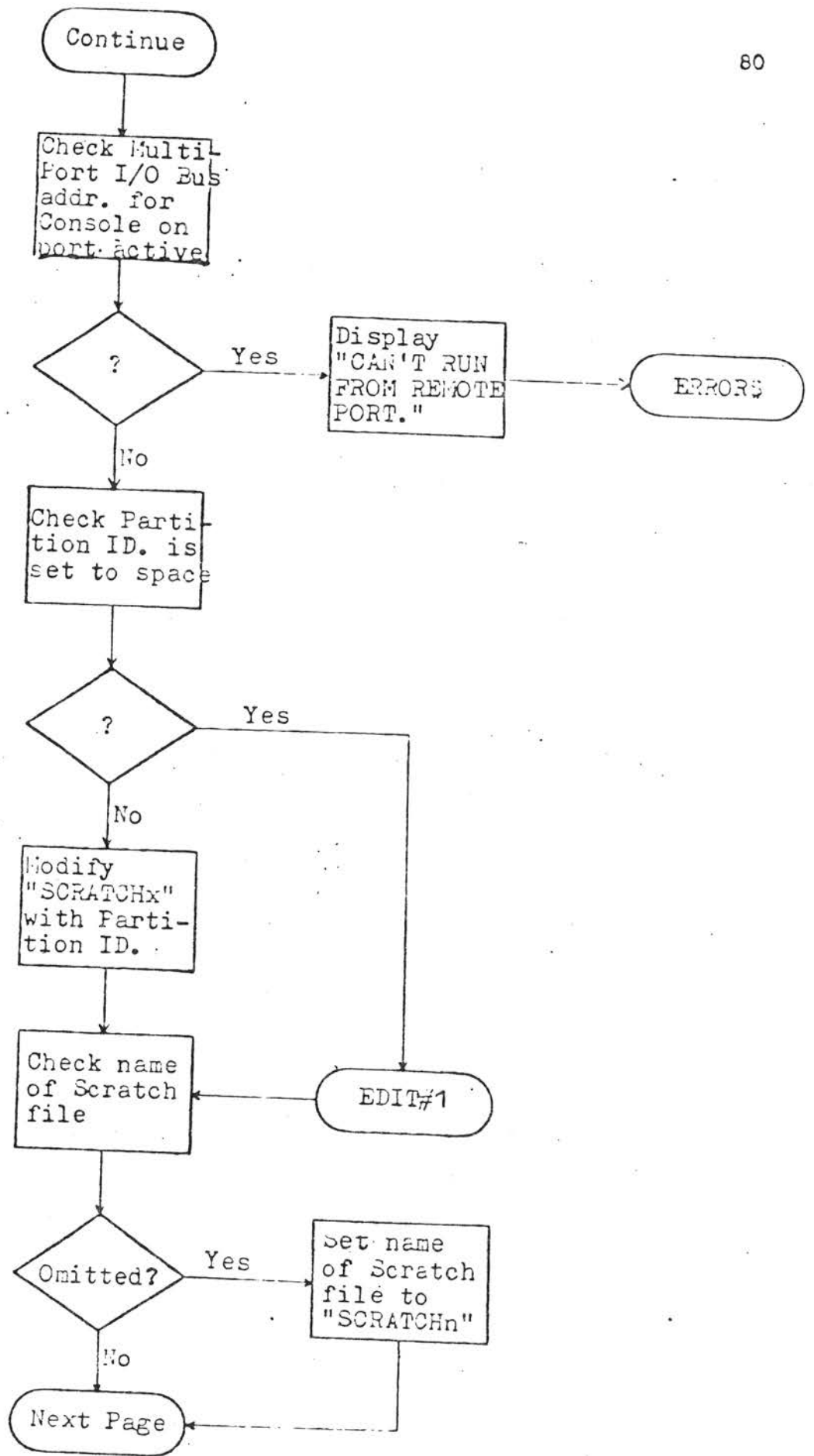
รูปที่ 3.1 แสดงรายละเอียดของแฟ้มข้อมูลที่ใช้เป็น CONFIGURATION FILE



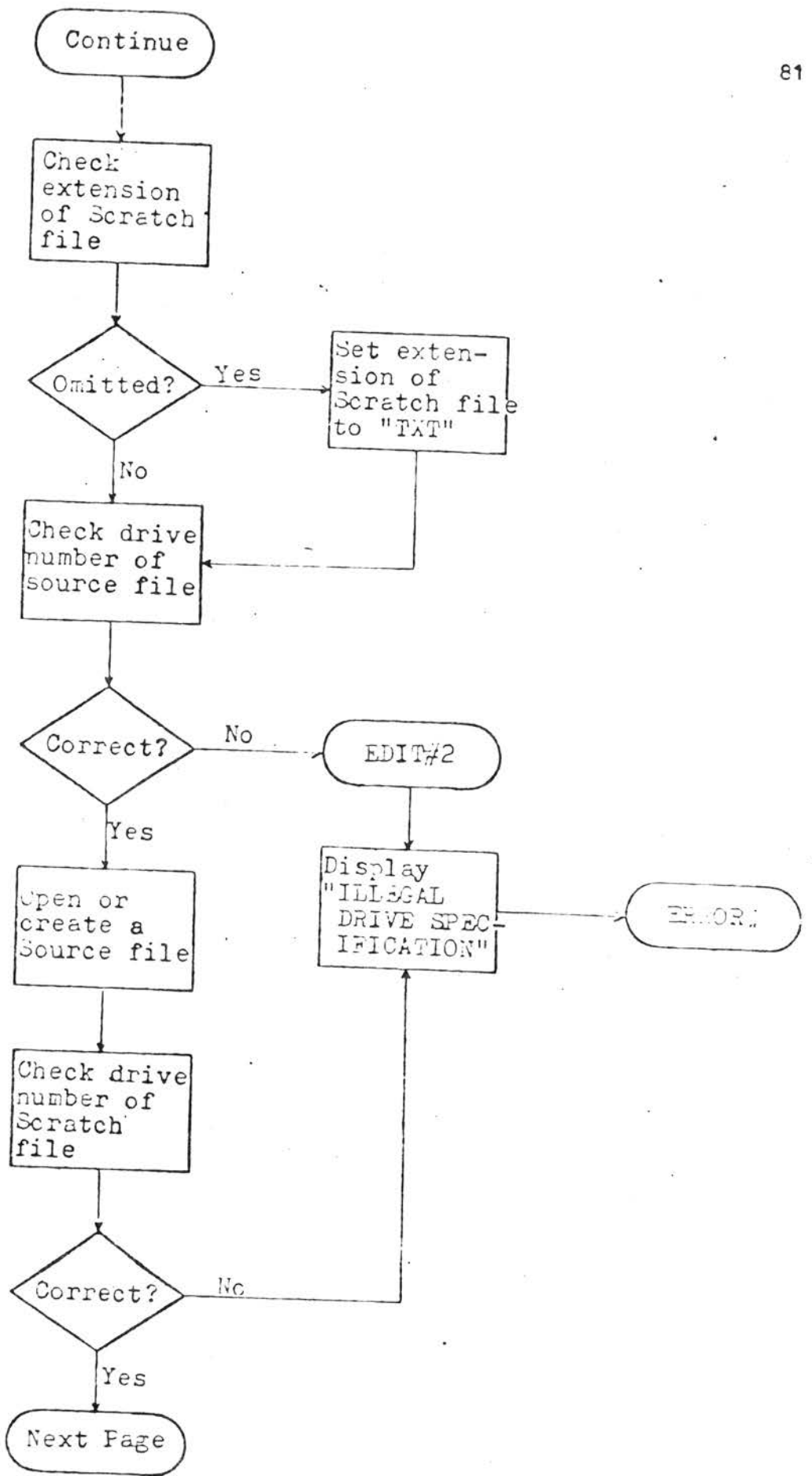
รูปที่ 3.2 แสดงขั้นตอนการทำงานของโปรแกรม EDIT/CHD



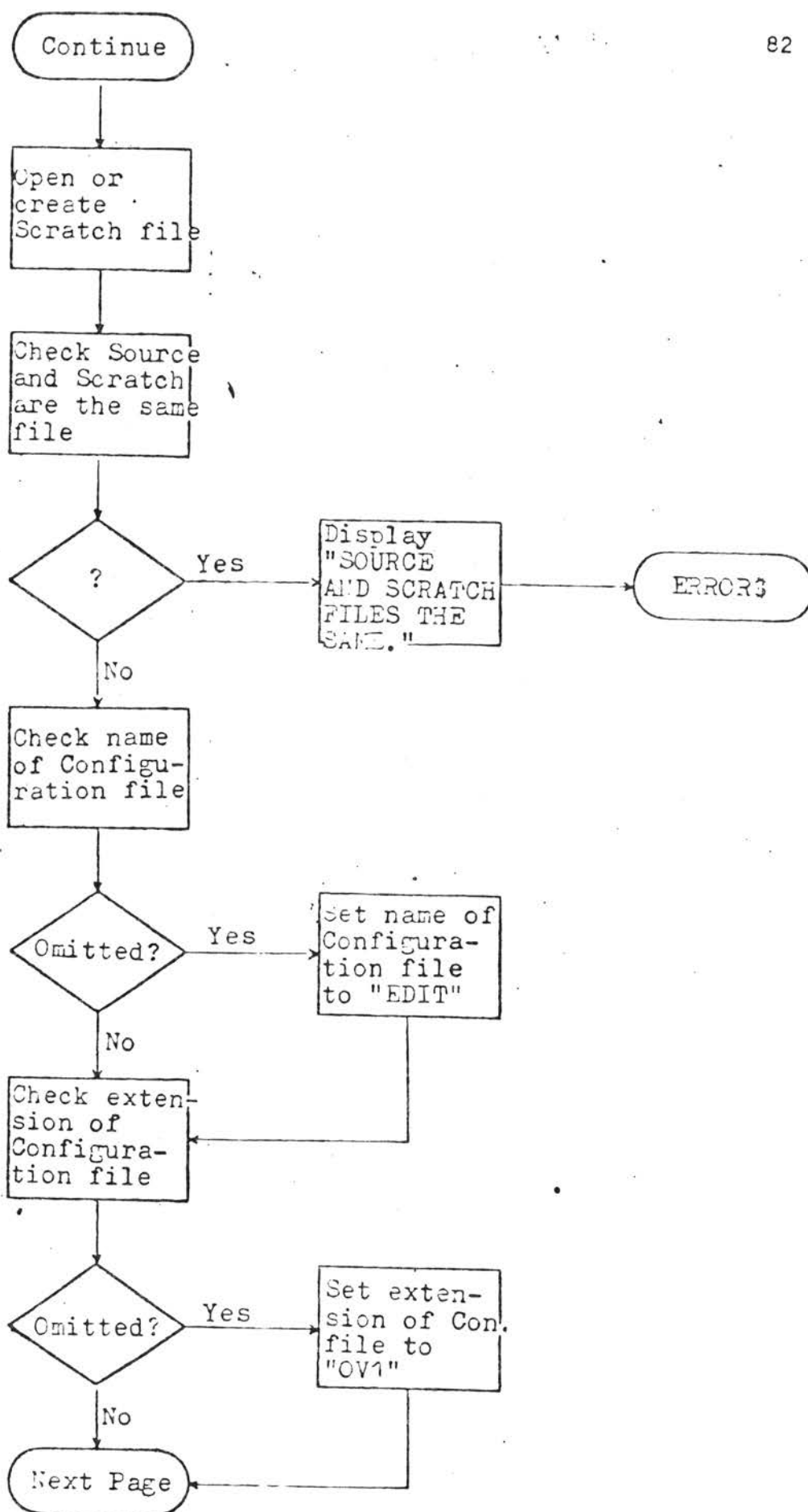
รูปที่ 3.2 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม EDIT/CHD



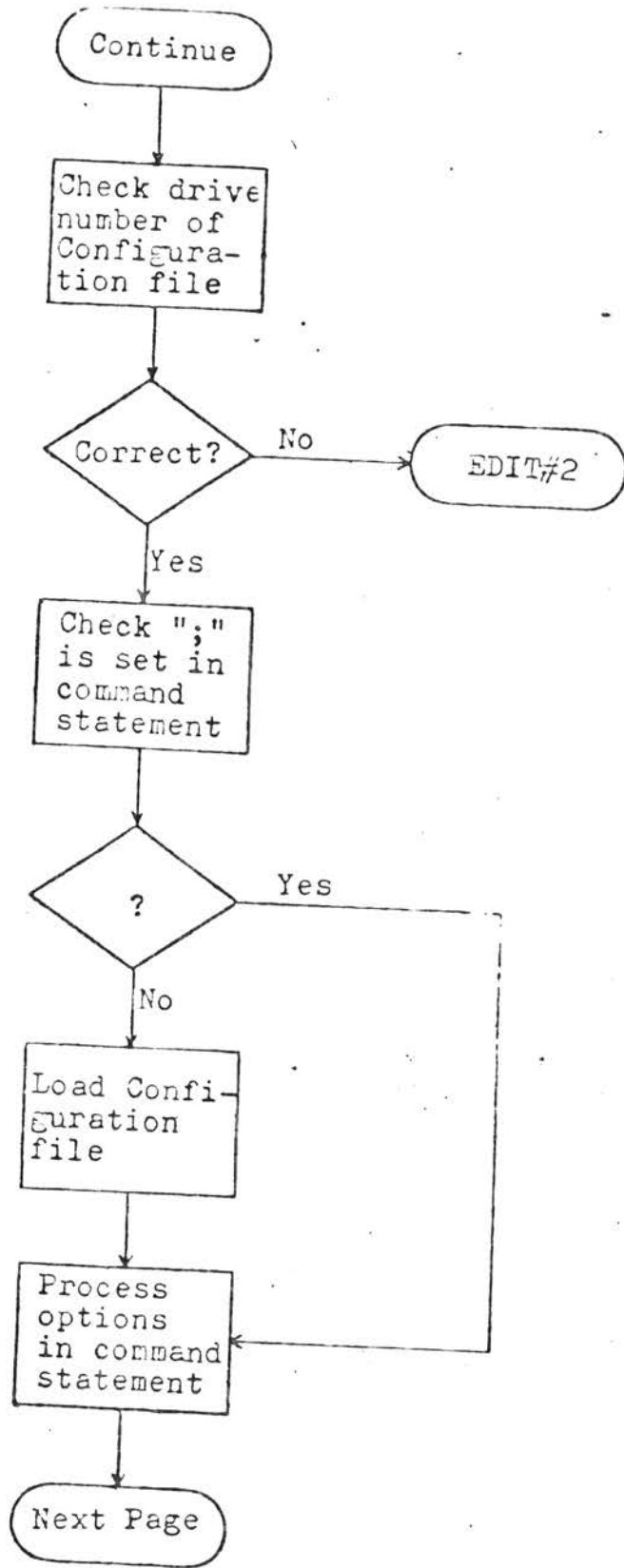
รูปที่ 3.2 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม EDIT/CMD



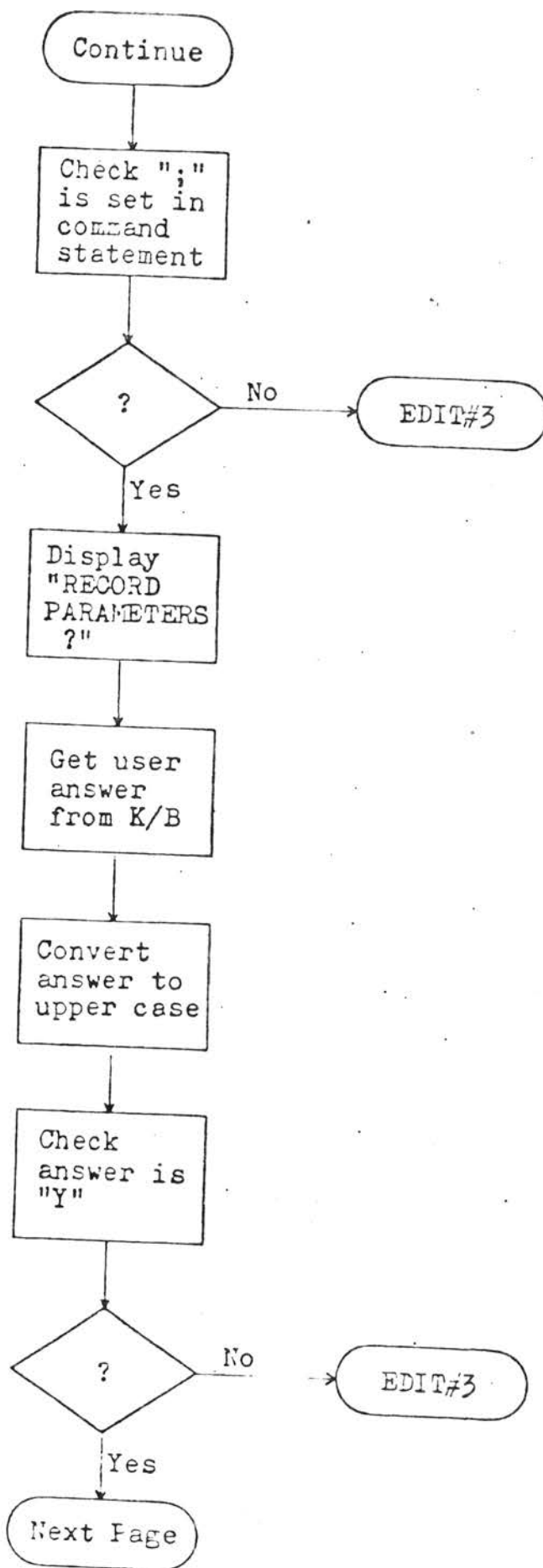
รูปที่ 3.2 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม EDIT/CD



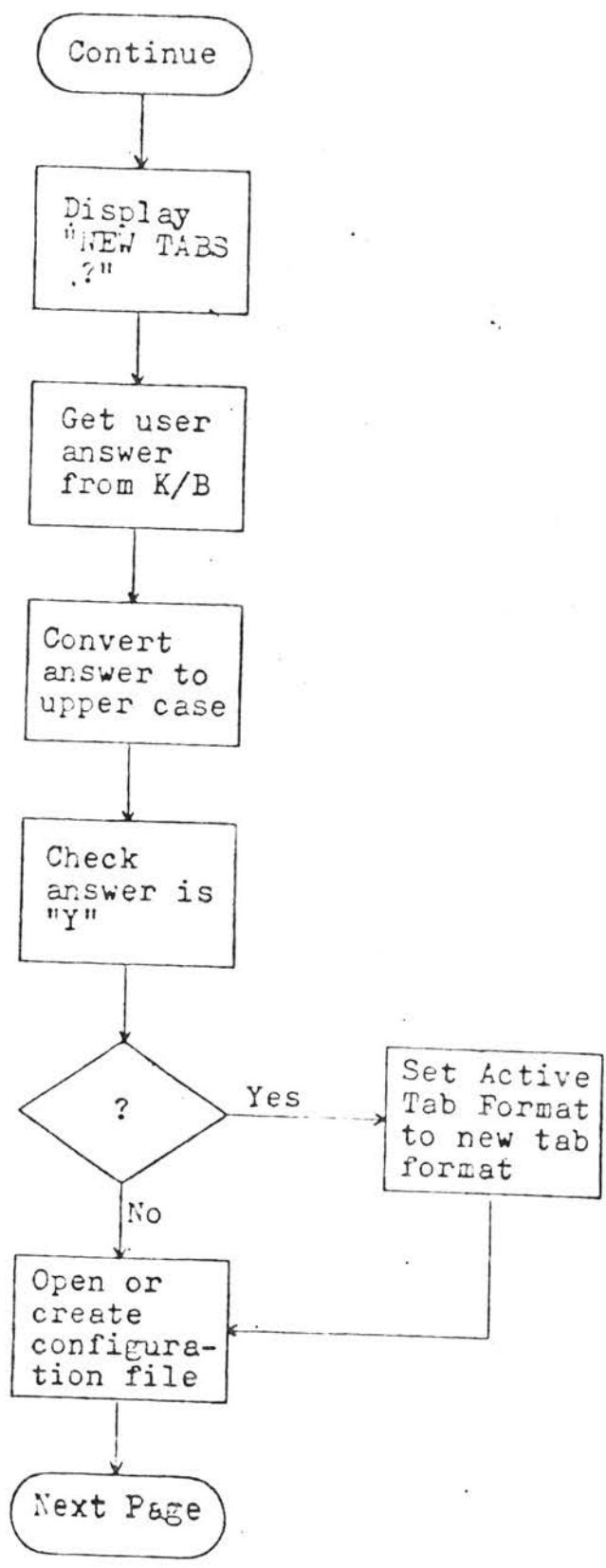
รูปที่ 3.2 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม EDIT/CHD



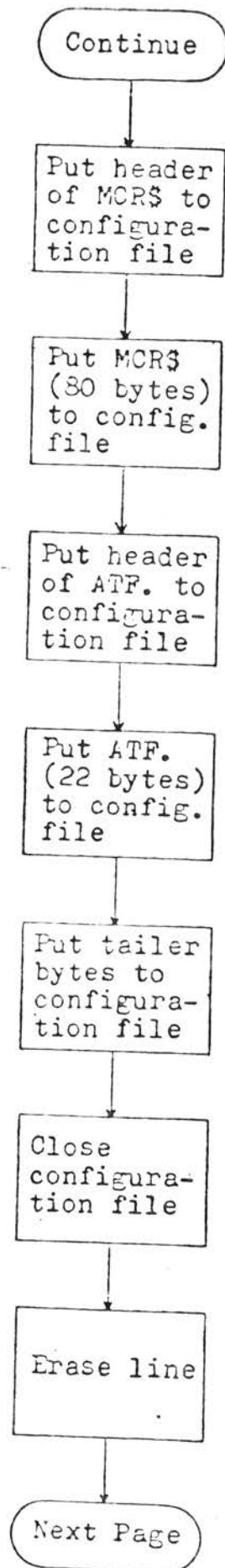
รูปที่ 3.2 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรมEDIT/CMD



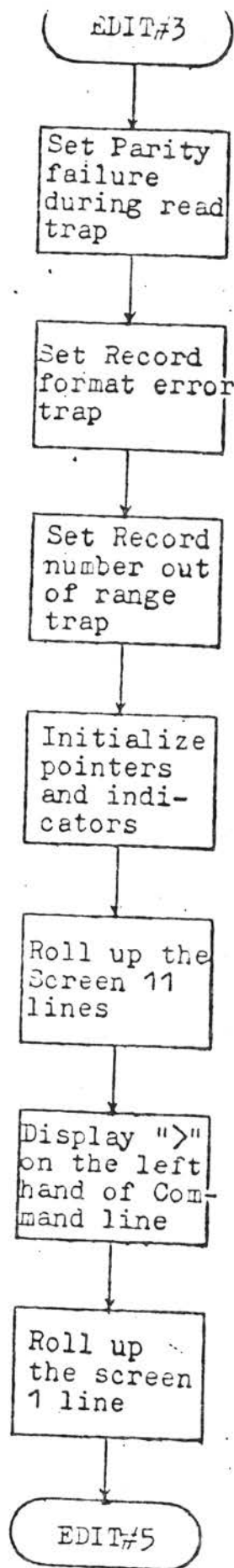
รูปที่ 3.2 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม EDIT/CMD



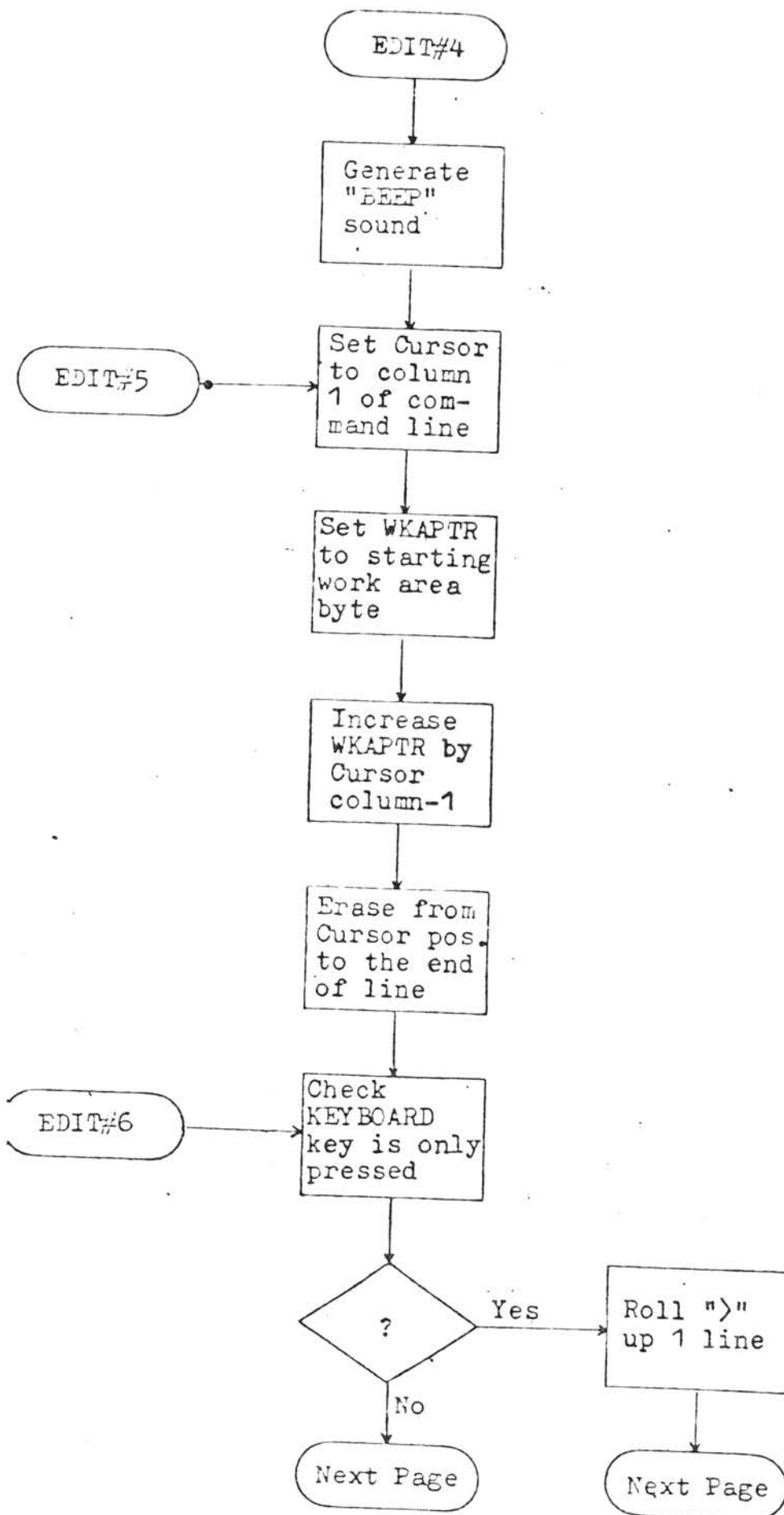
รูปที่ 3.2 (ค) แสดงขั้นตอนการทำงานของโปรแกรม EDIC/CI/D



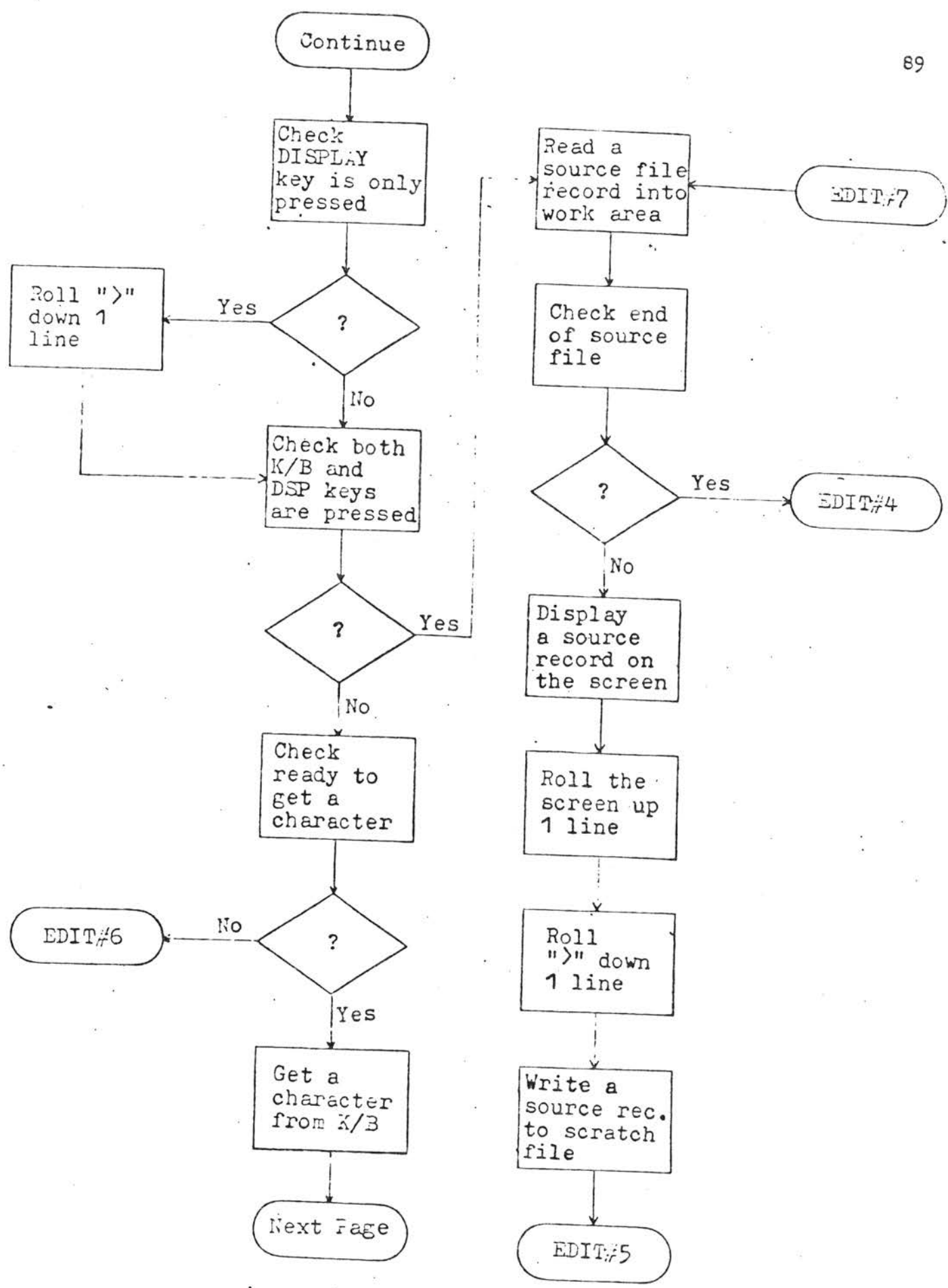
รูปที่ 3.2 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม EDIT/CMD



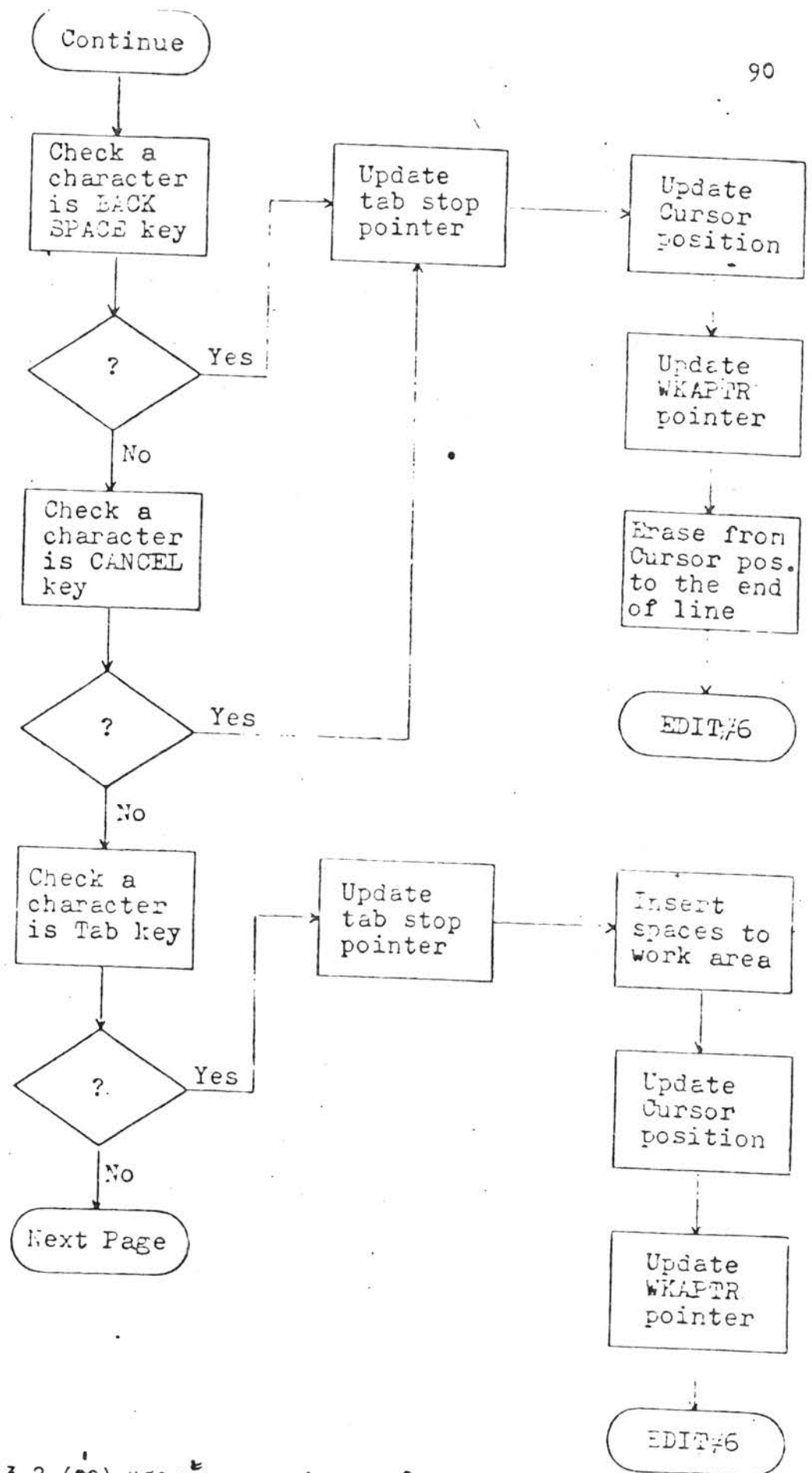
รูปที่ 3.2 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม EDIT/CHD



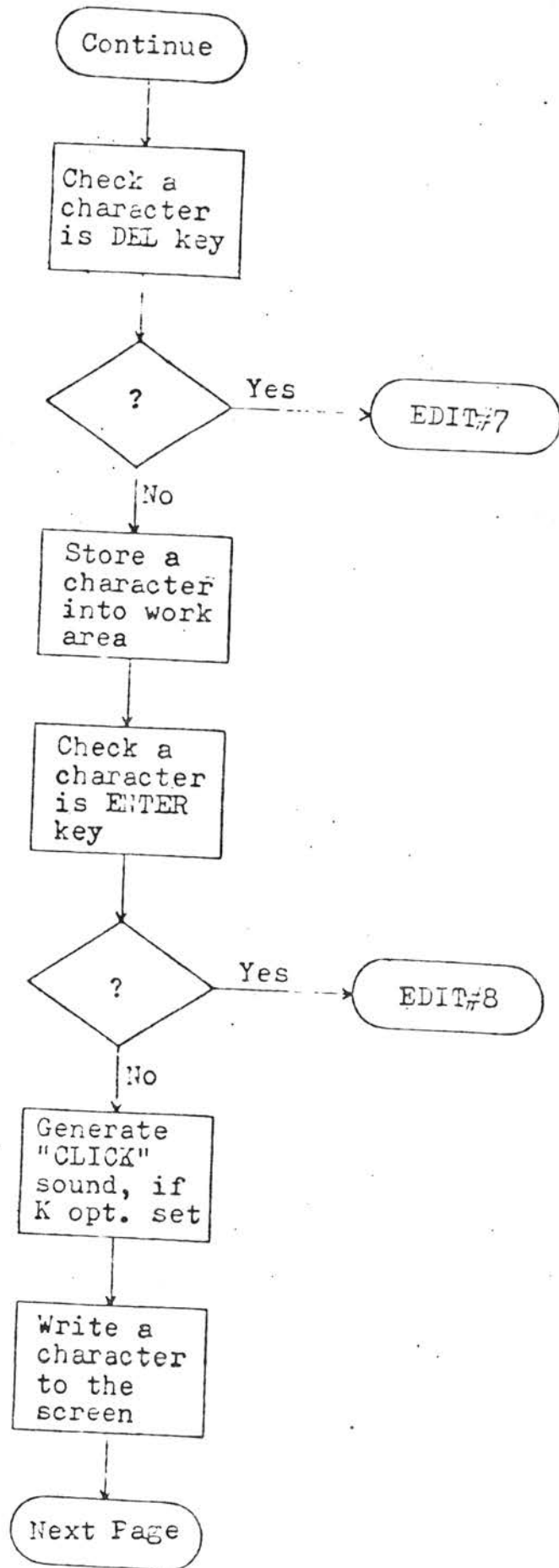
รูปที่ 3.2 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม EDIT/CMD



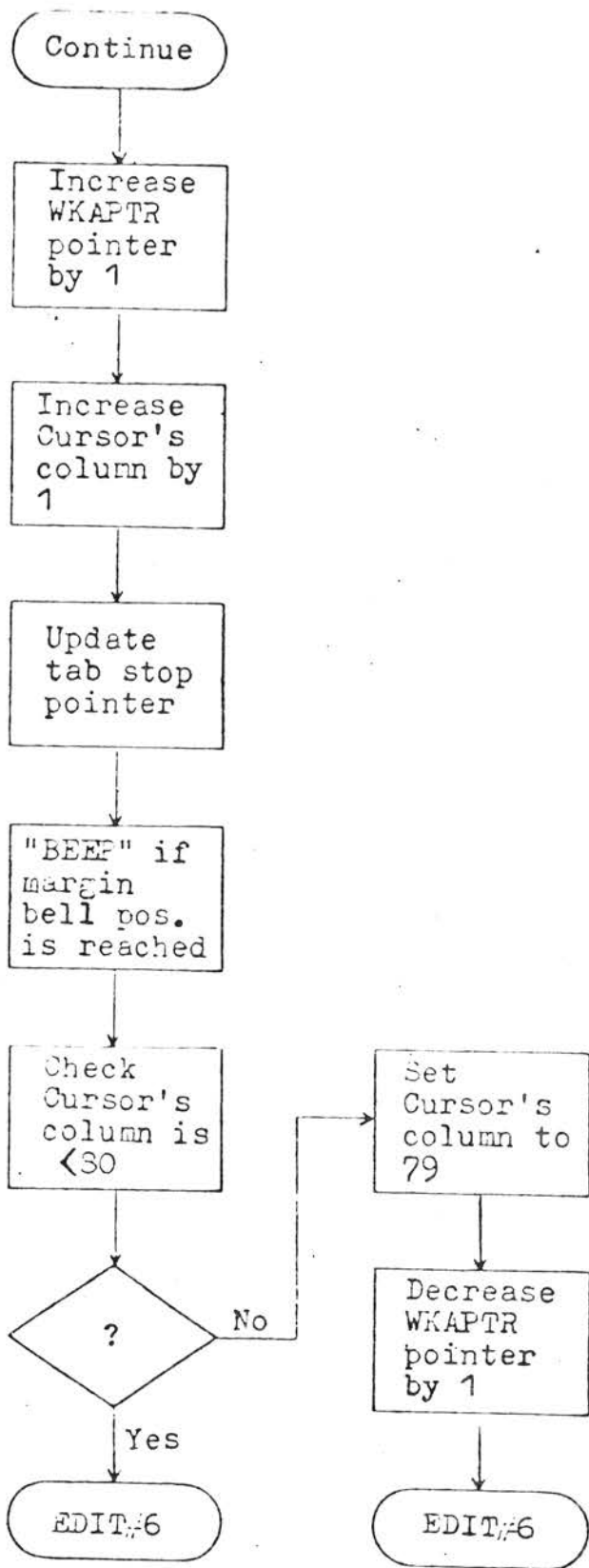
รูปที่ 3.2 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม EDIT/CMO



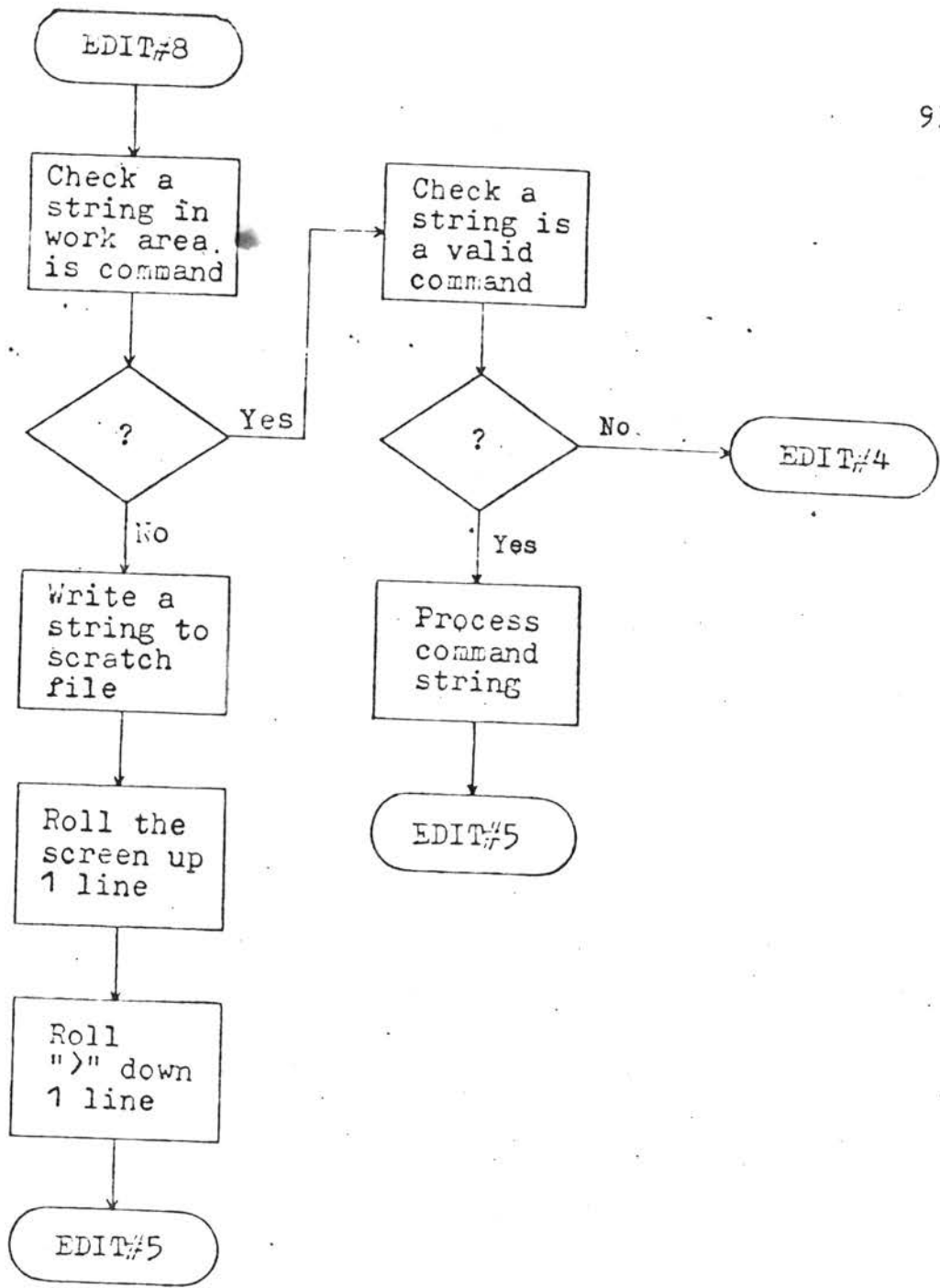
รูปที่ 3.2 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม EDIT/6



รูปที่ 3.2 (ค) แสดงขั้นตอนการทำงานของโปรแกรม EDIT/CMD



รูปที่ 3.2 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม EDIT/6



รูปที่ 3.2 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม EDIT/CMD

3.2 โปรแกรมที่ใช้ในการแสดงเนื้อหาของแฟ้มข้อมูลแบบนอกรีต (LIST/CMD) ถูกเขียนมาเพื่อใช้ในการแสดงเนื้อหาบางส่วนหรือทั้งหมดของแฟ้มข้อมูลแบบนอกรีตที่ถูกบันทึกอยู่ในจานแม่เหล็ก การแสดงเนื้อหาสามารถแสดงได้บนจอภาพหรือเครื่องพิมพ์ การแสดงเนื้อหาบนจอภาพจะแสดงเพียง 72 คิวบิตของคีย์เรคคอร์ด ส่วนการแสดงเนื้อหาบนเครื่องพิมพ์จะแสดงเนื้อหาเฉพาะ 124 คิวบิตของคีย์เรคคอร์ด ที่ว่างที่เหลืออีก 8 ค่าหนึ่งบนจอภาพหรือบนเครื่องพิมพ์จะถูกใช้สำหรับเป็นคีย์ควบคุมสำคัญของเรคคอร์ด

3.2.1 การเรียกใช้โปรแกรม LIST/CMD มีรูปแบบการเรียกใช้ดังนี้

LIST < file spec > [<spec2>] [<filespec2>] [; options]

< file spec > เป็นชื่อแฟ้มข้อมูลแบบนอกรีตที่ใช้ต้องการดูเนื้อหา ตัวอย่างเช่น PROGRAM/TXT, DATA/TXT, PAYROLL/ISI เป็นต้น ถ้าผู้ใช้ไม่กำหนดค่าของ Extension ให้โปรแกรม LIST/CMD จะกำหนดค่าของ Extension ให้โดยพิจารณาจากค่าของ options ที่ผู้ใช้กำหนด

< Spec2 > เป็นตัวกำหนดให้โปรแกรม LIST/CMD เริ่มต้นแสดงเนื้อหาของแฟ้มข้อมูลตามค่าหนึ่งที่ใช้ต้องการ ถ้าค่าของ < Spec2 > ไม่ได้ถูกกำหนด โปรแกรม LIST/CMD จะกำหนดให้มีค่าเป็น เรคคอร์ดแรกและเช็คเตอร์แรกของแฟ้มข้อมูล การกำหนดค่าของ < Spec2 > ผู้ใช้สามารถกำหนดเป็นหมายเลขเรคคอร์ดหรือหมายเลขเช็คเตอร์ที่อยู่ในแฟ้มข้อมูลที่ใช้ต้องการจะดูเนื้อหา

Lnnnn เป็นการกำหนดเรคคอร์ดเริ่มต้นให้โปรแกรม LIST/CMD

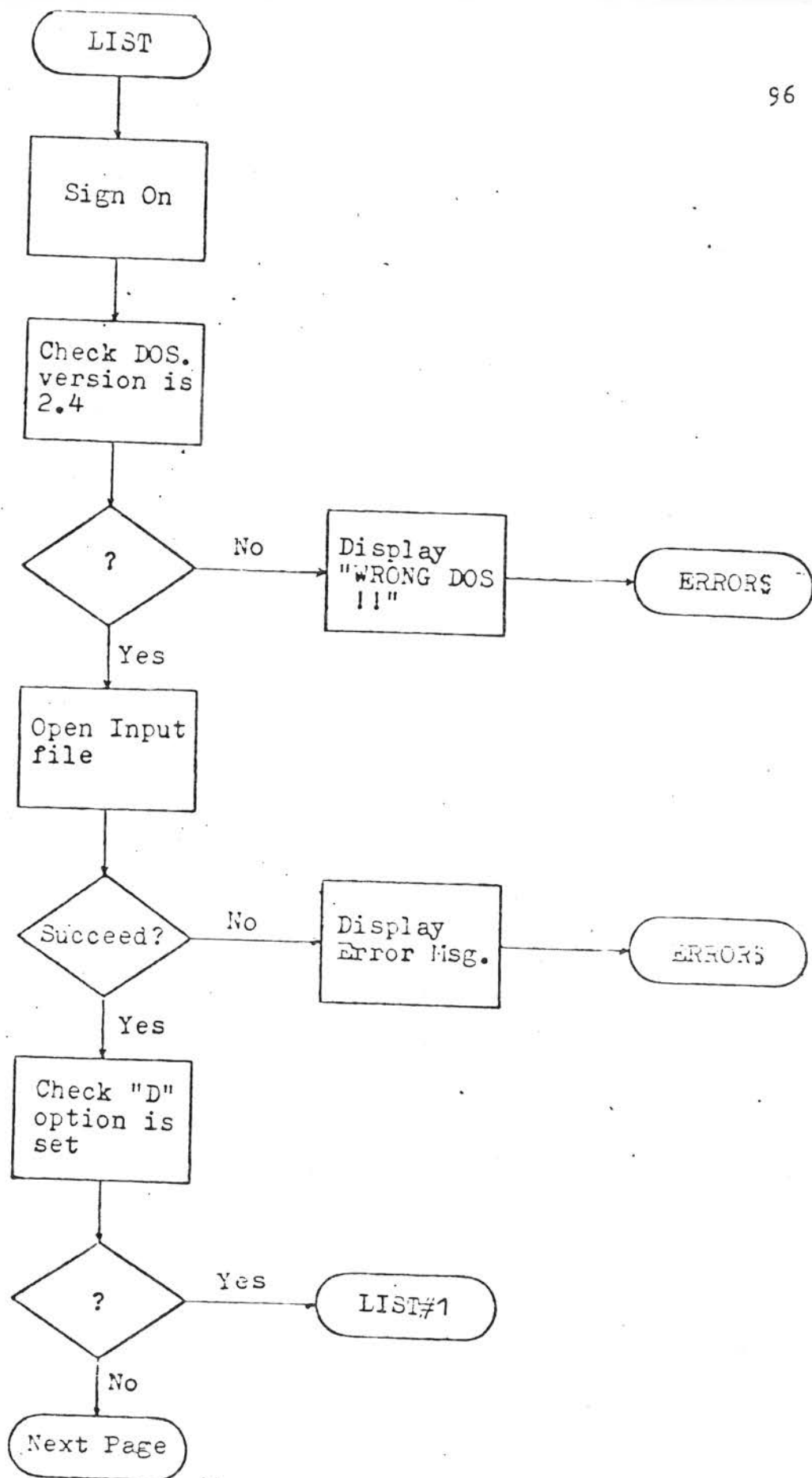
Rnnnn เป็นการกำหนดเช็คเตอร์เริ่มต้นให้โปรแกรม LIST/CMD

< filespec2 > เป็นชื่อของแฟ้มข้อมูลที่จะถูกบันทึกอยู่ในจานแม่เหล็กในเมื่อผู้ใช้กำหนดค่า options เป็น "P" หรือ "Q" ถ้าหากผู้ใช้ไม่กำหนดค่า < file-spec2 > โปรแกรม LIST/CMD จะกำหนดให้มีชื่อเหมือนกับชื่อแฟ้มข้อมูลตาม < file-spec > และมีค่า extension เป็น "PRT"

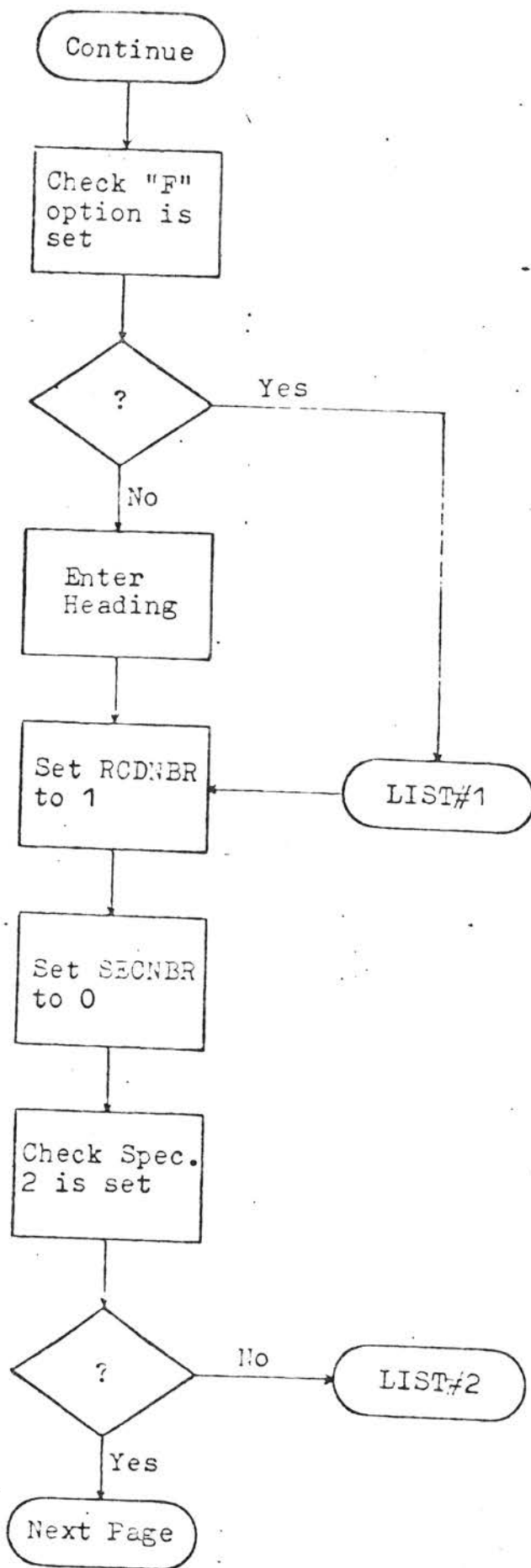
< Options > มีค่าต่าง ๆ เพื่อให้ผู้ใช้เลือกใช้ตามความต้องการดังต่อไปนี้

- L - ให้แสดงเนื้อหาของเครื่องพิมพ์แบบ Local
 - S - แสดงเนื้อหาของเครื่องพิมพ์แบบ Servo
 - D - แสดงเนื้อหาของพิมพ์รูปบนจอภาพ
 - X - แสดงเนื้อหาโดยไม่มีหมายเลขลำดับเรคคอร์ด
 - F - แสดงเนื้อหาของพิมพ์รูปแบบ Formatted Print
 - P - ให้ทำการบันทึกเนื้อหาของพิมพ์รูปแบบ Formatted Print
 - Q - ให้ทำการบันทึกเนื้อหาของพิมพ์รูปแบบ Formatted Print โดยทำการบันทึกต่อจากเนื้อหาเดิมที่มีอยู่
 - Mn - เป็นการกำหนดจำนวนเรคคอร์ดที่ปรากฏหน้ากระดาษพิมพ์
 - I - ทำการแสดงผลเนื้อหาของพิมพ์รูปโดยเรียงตามลำดับที่มี
- 3.2.2 ขั้นตอนการทำงานของโปรแกรม LIST/CMD ได้แสดงไว้ตามรูปที่

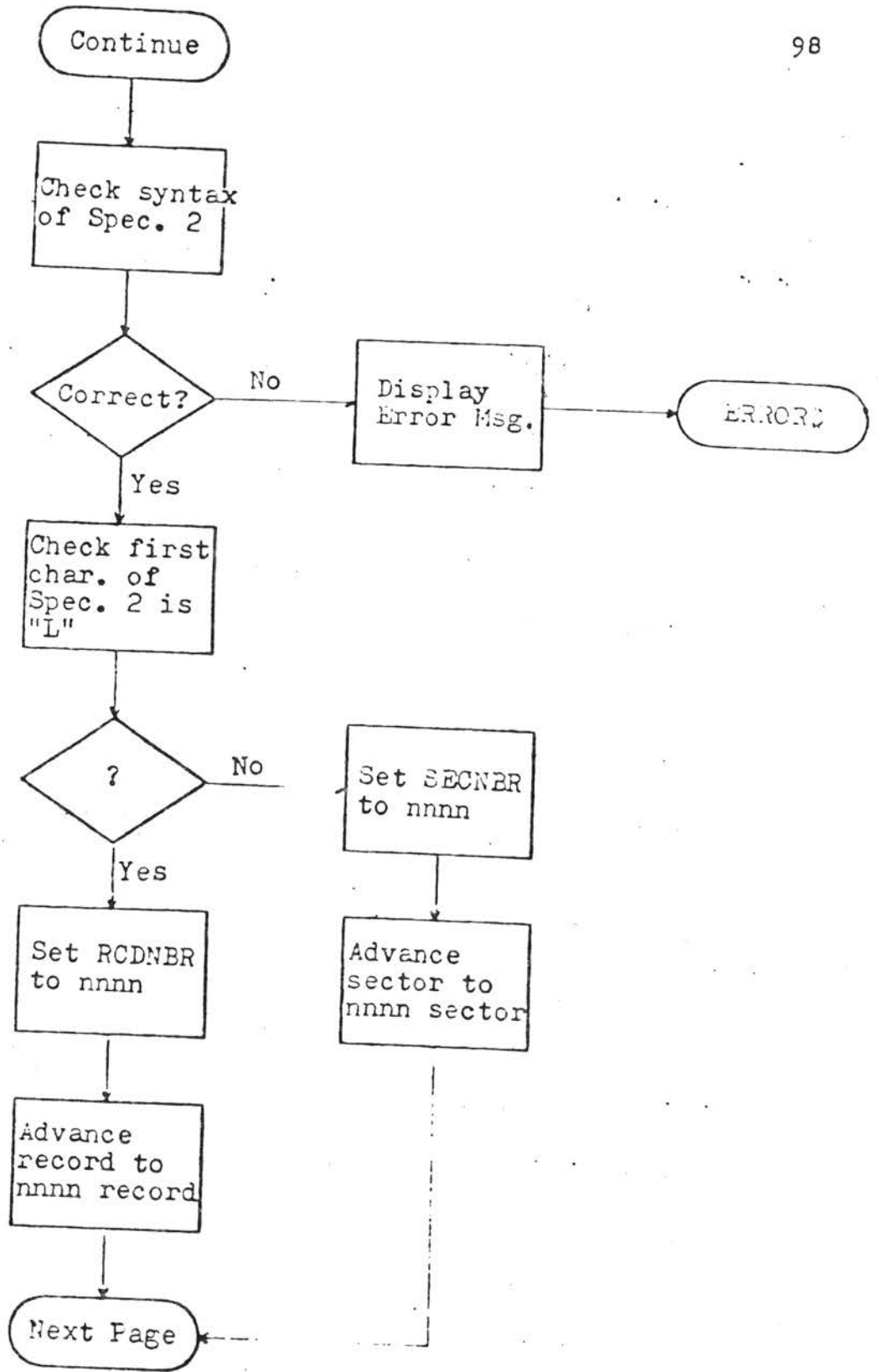
3.3



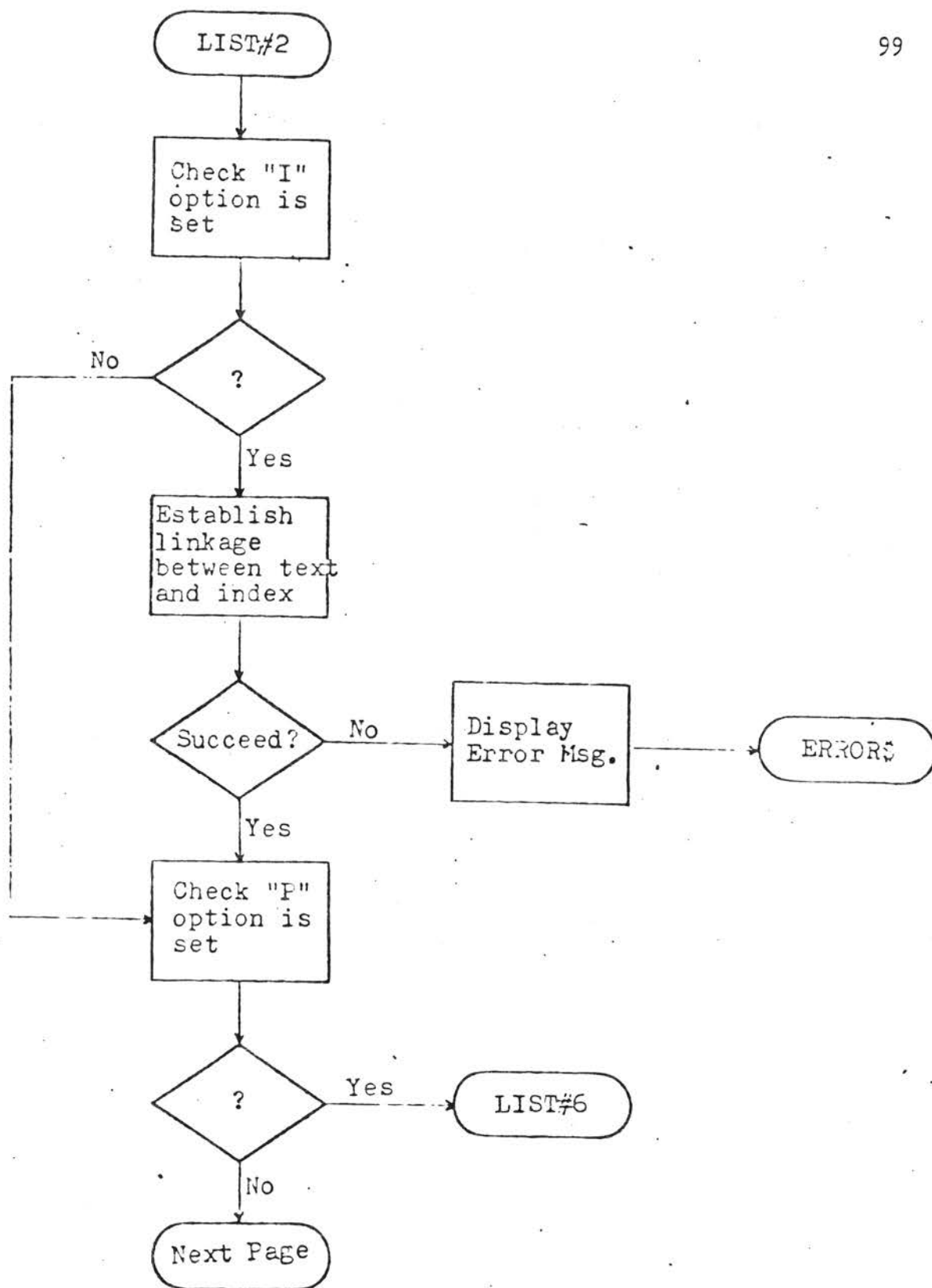
รูปที่ 3.3 แสดงขั้นตอนการทำงานของโปรแกรม LIST/CMD



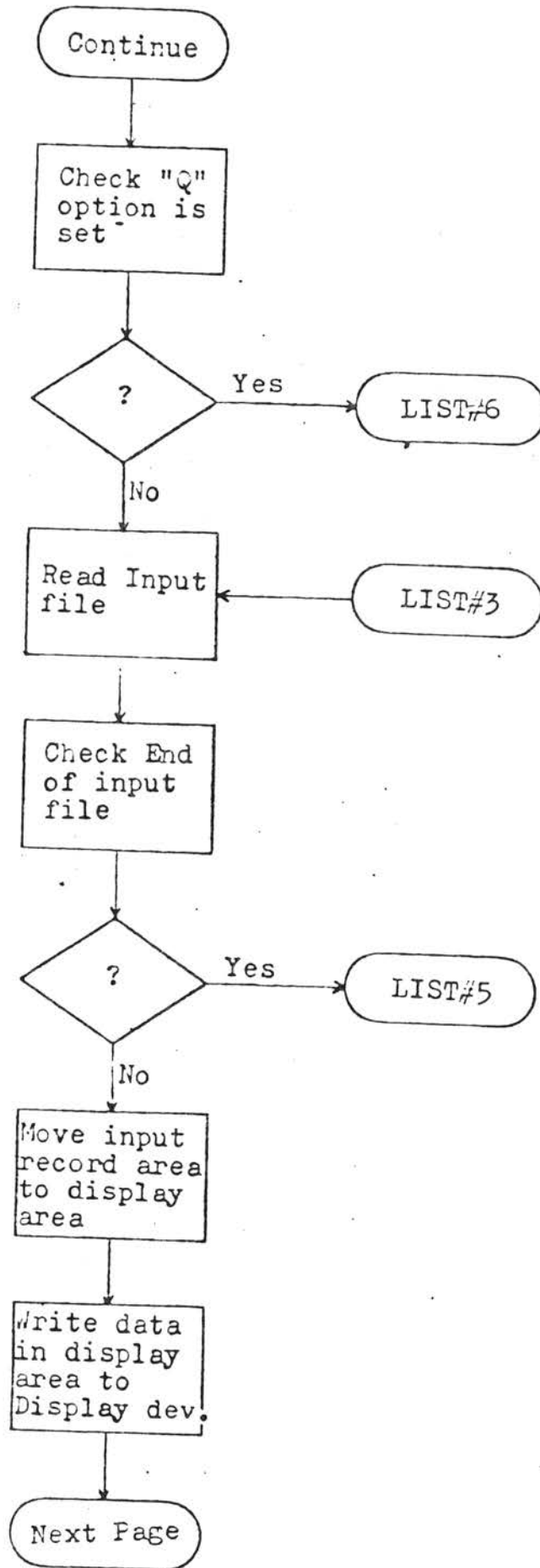
รูปที่ 3.3 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม LIST/CMD



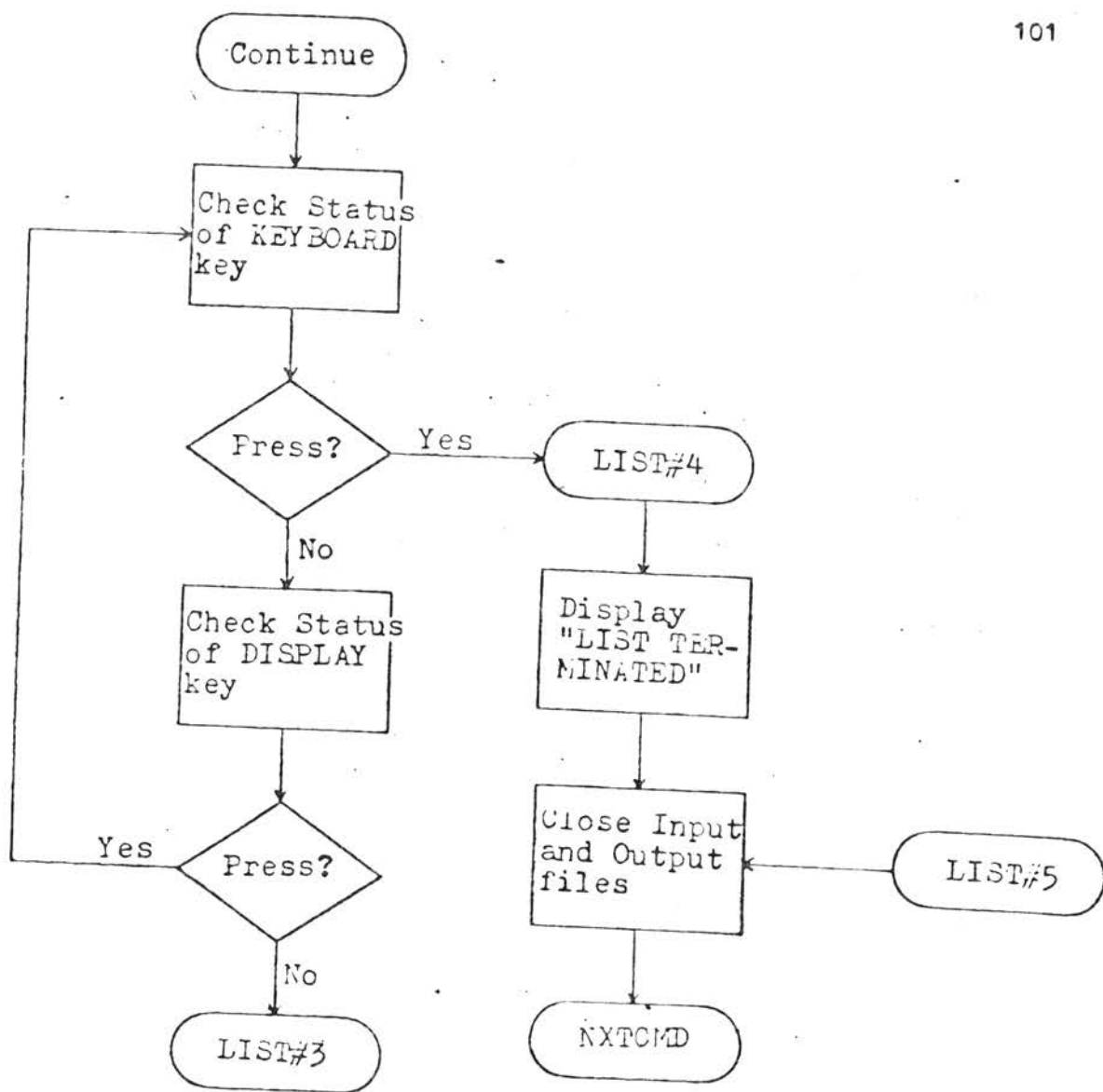
รูปที่ 3.3 (ค) แสดงขั้นตอนการทำงานของโปรแกรมLIST/CHD



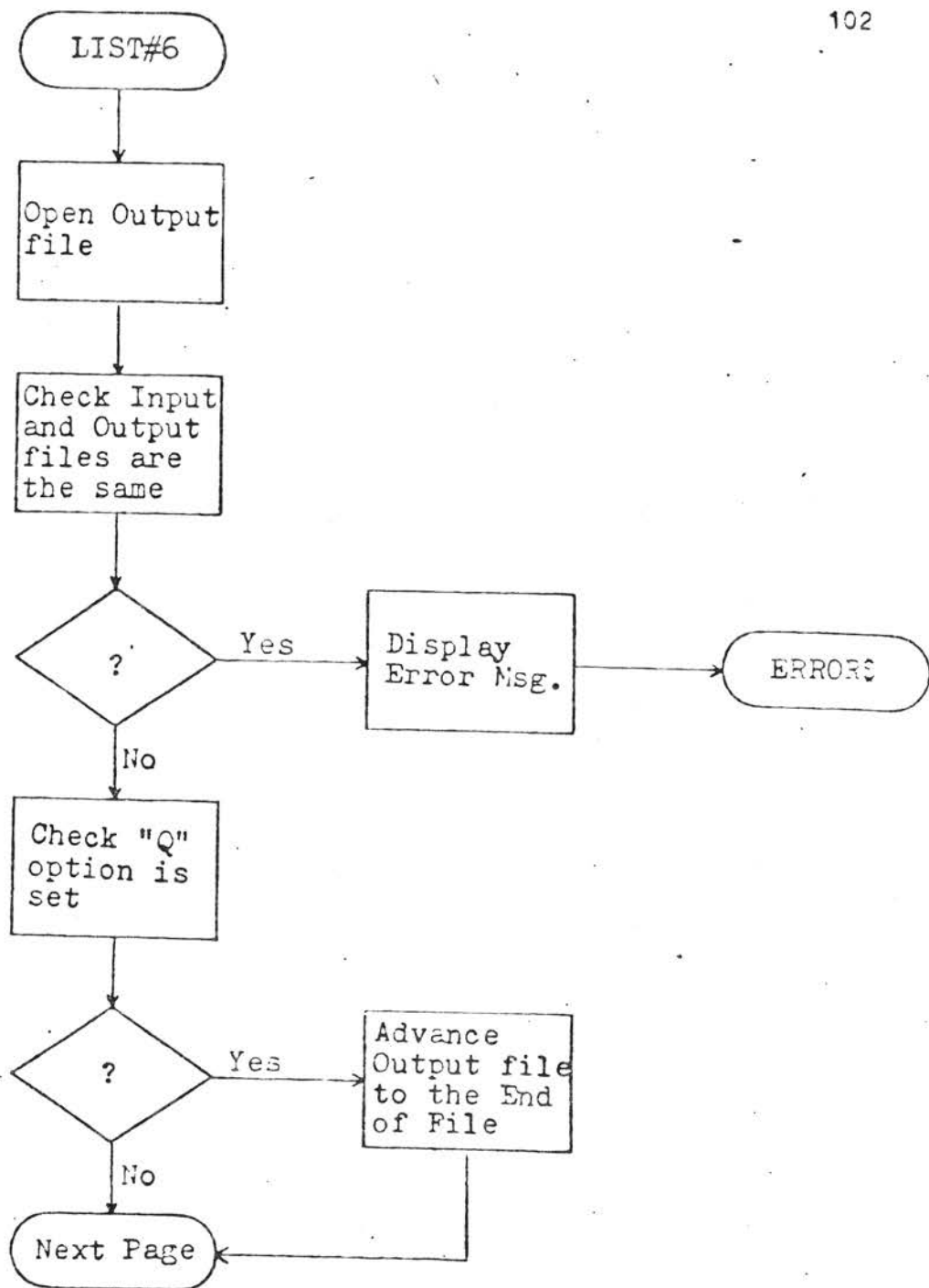
รูปที่ 3.3 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม LIST/CMD



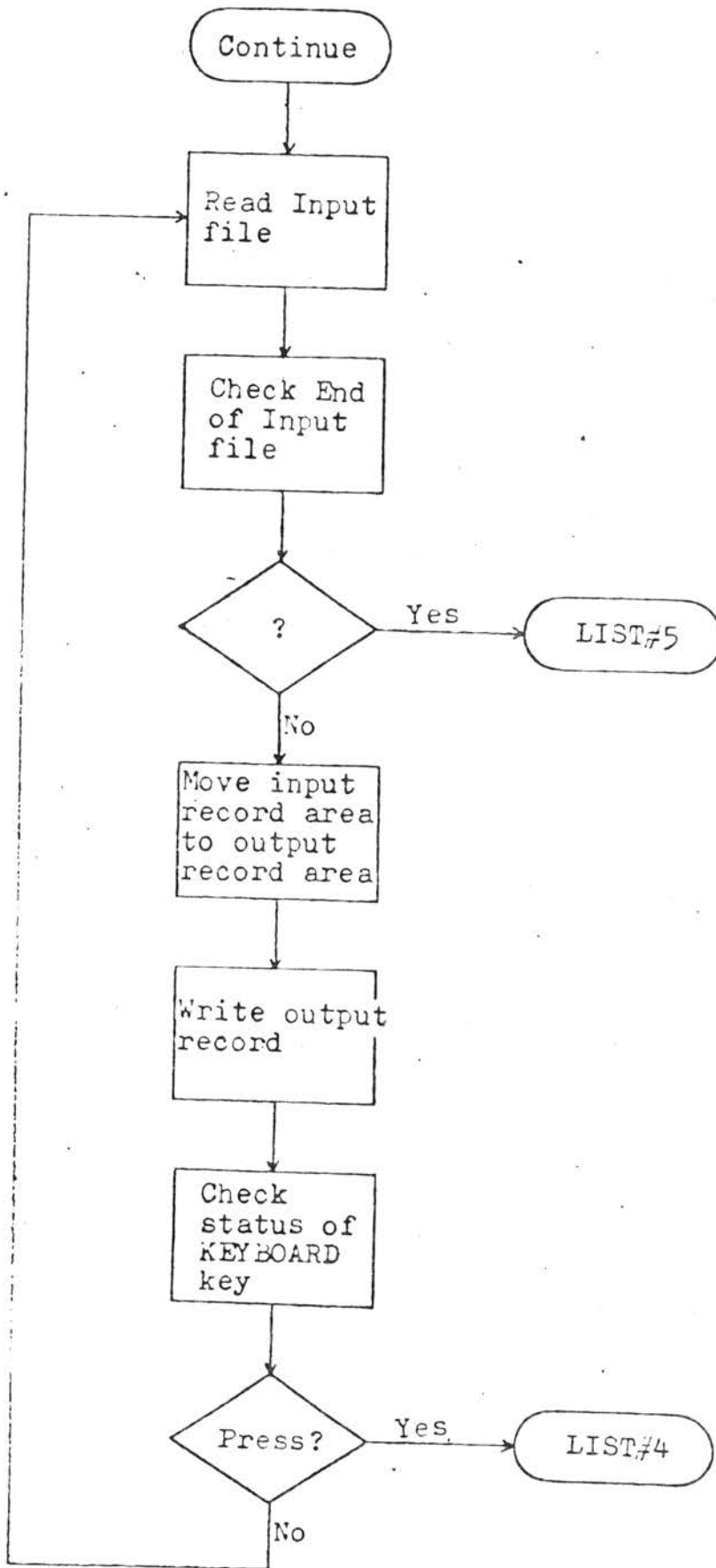
รูปที่ 3.3 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม LIST/CMD



รูปที่ 3.3 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม LIST/CMD



รูปที่ 3.3 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม LIST/CMD



รูปที่ 3.3 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม LIST/C/D

3.3 โปรแกรมที่ใช้ในการจัดรูปแบบของแฟ้มข้อมูลแบบฮัก (REFORMAT/CMD)

REFORMAT/CMD เป็นโปรแกรมที่ใช้ในการเปลี่ยนแปลงรูปแบบของการบันทึกข้อมูลภายในจานแม่เหล็กของแฟ้มข้อมูลแบบฮักให้มีรูปแบบตามที่ผู้ใช้ต้องการ

3.3.1 การเรียกใช้โปรแกรม REFORMAT/CMD มีรูปแบบการเรียกใช้ดังนี้

```
REFORMAT <filespec1> [><filespec2>] [;parameters]
```

<filespec1> เป็นชื่อของแฟ้มข้อมูลแบบฮักที่ต้องการจะเปลี่ยนแปลงรูปแบบการบันทึกข้อมูล ถ้าหากผู้ใช้ไม่กำหนดค่าของ Extension โปรแกรม REFORMAT/CMD จะกำหนดให้มีค่าเป็น "TXT"

<filespec2> เป็นชื่อของแฟ้มข้อมูลที่เป็น Output ของการเปลี่ยนแปลงรูปแบบการบันทึกข้อมูลจะถูกบันทึกลงในแฟ้มข้อมูลนี้ ถ้าผู้ใช้ไม่กำหนด <filespec2> ของการเปลี่ยนแปลงจะถูกบันทึกลงในแฟ้มข้อมูลเดิมคือ <filespec1> ดังนั้นถ้าหากผู้ใช้ต้องการให้ผลการเปลี่ยนแปลงถูกบันทึกลงในแฟ้มข้อมูลเดิมก็สามารถทำได้โดยไม่ต้องกำหนดค่าของ <filespec2>

<Parameters> พารามิเตอร์ที่ไร้เครื่องหมายดอกจันที่ว่างหรือเครื่องหมาย " ," รายละเอียดของพารามิเตอร์นี้คือะควมี้ดังนี้

| Parameter | Description |
|---|--|
| B<n> | The output file will be blocked. This implies no space or record compression, with <n> logical records per physical sector. |
| C | The output file will be space and record compressed. The number of logical records per physical sector will be indeterminate. |
| R | The output file will be record compressed, but no space compression will be done. In general, the number of logical records per physical sector will be indeterminate. |
| L<n> | The length of each logical record will be adjusted to <n> characters. Note that if the logical records are space compressed, this will not make the physical length of the records <n> characters. If the logical record is shorter than <n> characters, it will be padded with blanks to the proper length. If the logical record is longer than <n> characters, the action taken depends on the T and S parameter. |
| T | (Only valid if L parameter is given) Truncate the logical record if it is longer than <n> characters. |
| S | (Only valid if L parameter is given) If the length of the logical record is greater than <n> characters, segment it into (q) logical records each of length <n> padding if necessary. The number (q) is defined as input length divided by <n> rounded upward to the next integer. |
| If neither S or T is specified, and an input record of length greater than <n> is found, a message is issued and REFORMAT gives up. | |
| D | If reformatting is done in place and this parameter is specified, any disk space vacated by the reformatting process will be returned to the operating system for re-use. |

3.3.2 รูปแบบของแฟ้มข้อมูลที่เป็น Output มี 3 รูปแบบดังนี้

3.3.2.1 Blocked File ลักษณะของแฟ้มข้อมูลแบบนี้จะไม่มีการจัดเรียงที่เป็นที่ว่างแต่ละบล็อกจะมีจำนวนเรคคอร์ดเท่ากัน

3.3.2.2 Record Compressed File ลักษณะของแฟ้มข้อมูลแบบนี้จะไม่มีการจัดเรียงที่เป็นที่ว่าง

3.3.2.3 Record and Space Compressed File ลักษณะของแฟ้มข้อมูลแบบนี้จะมีการจัดเรียงที่เป็นที่ว่างเข้าด้วยกัน

3.3.3 เหตุผลในการเปลี่ยนแปลงรูปแบบของแฟ้มข้อมูลแบบอื่น

3.3.3.1 ใช้ในการสื่อสารทางไกล (Telecommunications) ซึ่งขนาดของเรคคอร์ดจะคงมีความยาวคงที่เท่ากับทุก ๆ เรคคอร์ด

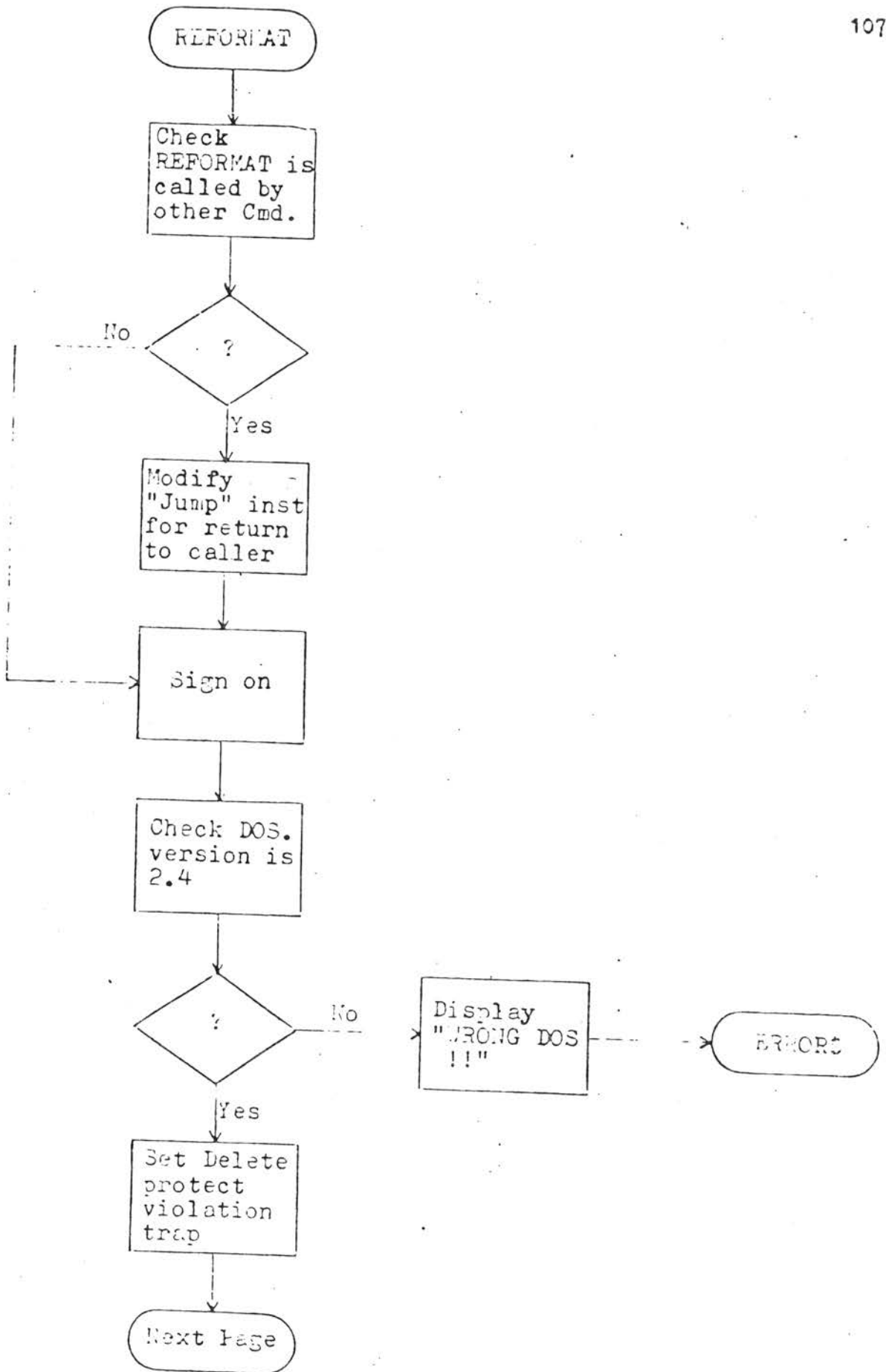
3.3.3.2 ใช้ในการเปลี่ยนรูปแบบของแฟ้มข้อมูลซึ่งใช้โปรแกรม EDIT/CMD

ในการเตรียมข้อมูลให้มีลักษณะเป็นแฟ้มข้อมูลแบบบล็อก (Blocked File)

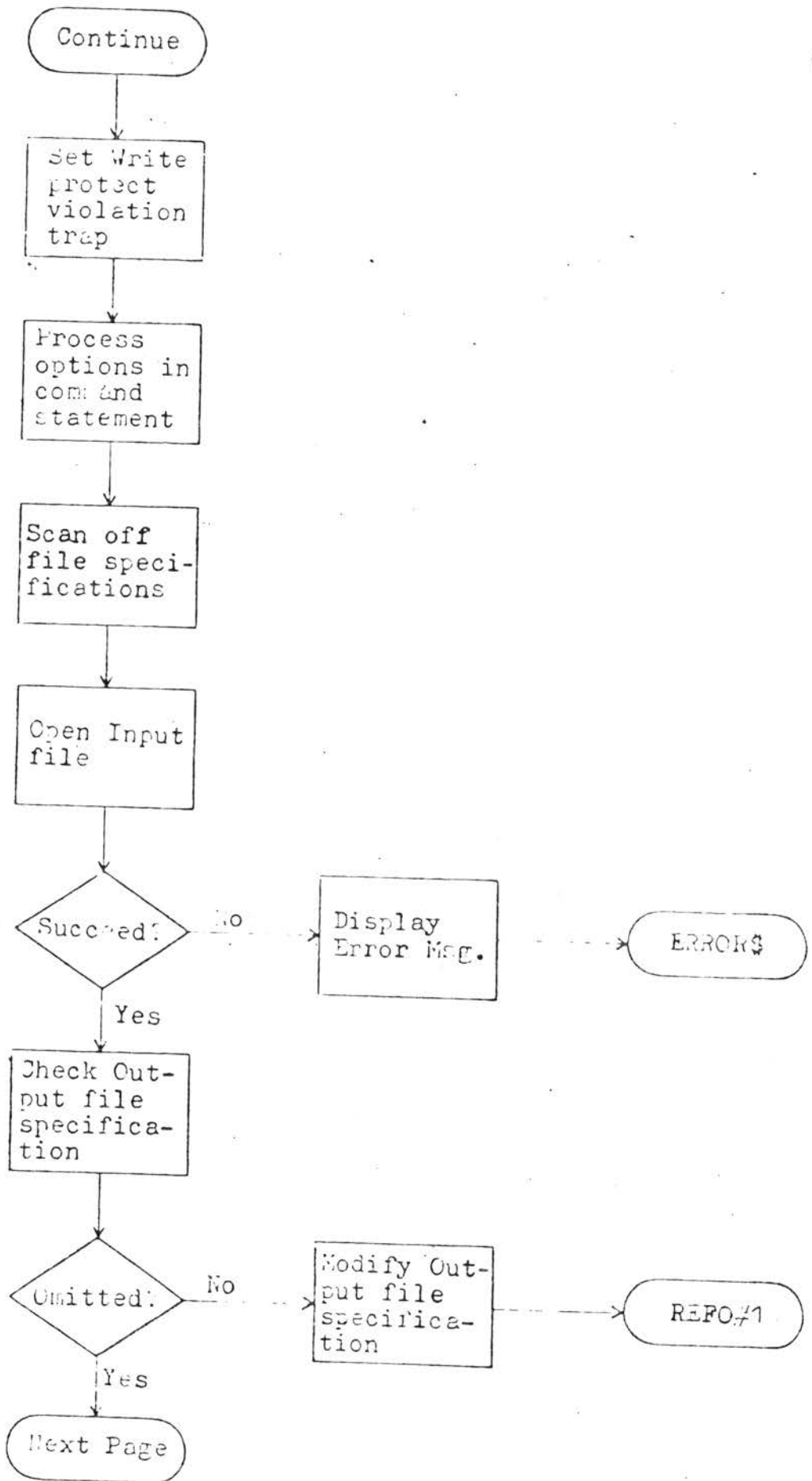
เพื่อให้สามารถเข้าถึงแบบสุ่ม (Random access) เช่น Database

Programs หรือ Databus programs

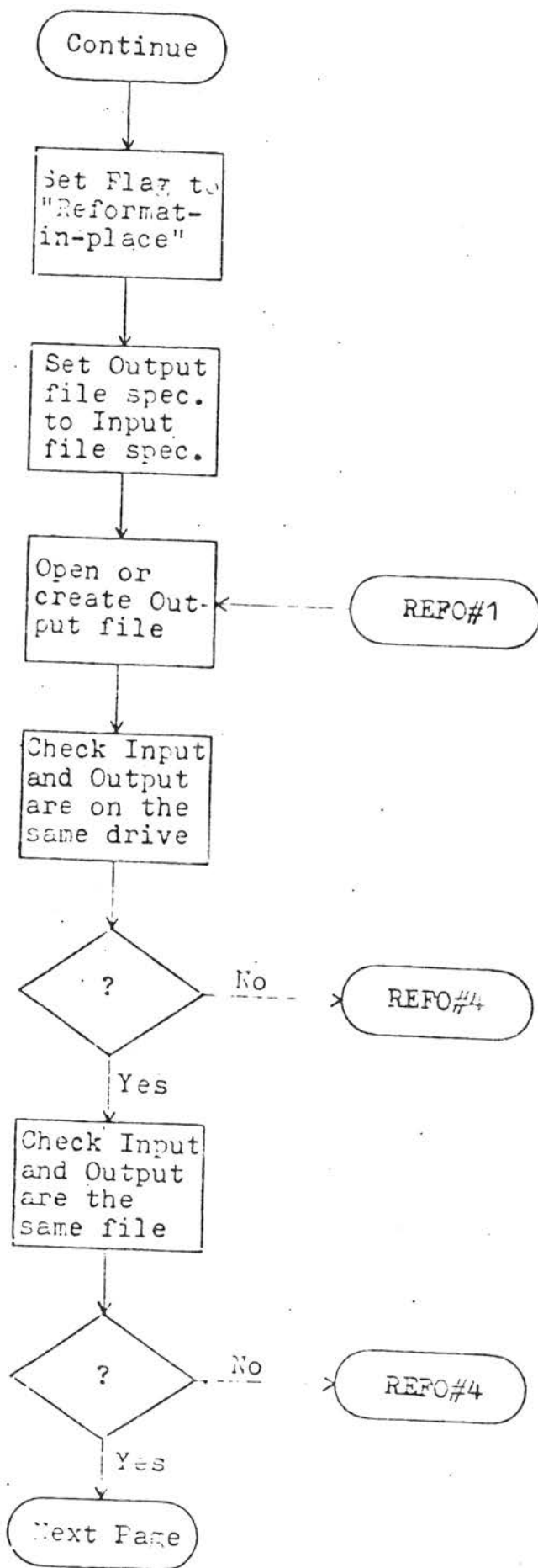
3.3.3.3 ใช้ในการรวบรวมเรคคอร์ดที่ไม่ได้ใช้งานที่มีอยู่ในแฟ้มข้อมูลแบบอื่นออกจากแฟ้มข้อมูลเพื่อให้สามารถนำไปจัดสรรให้กับแฟ้มข้อมูลอื่น ๆ ต่อไป



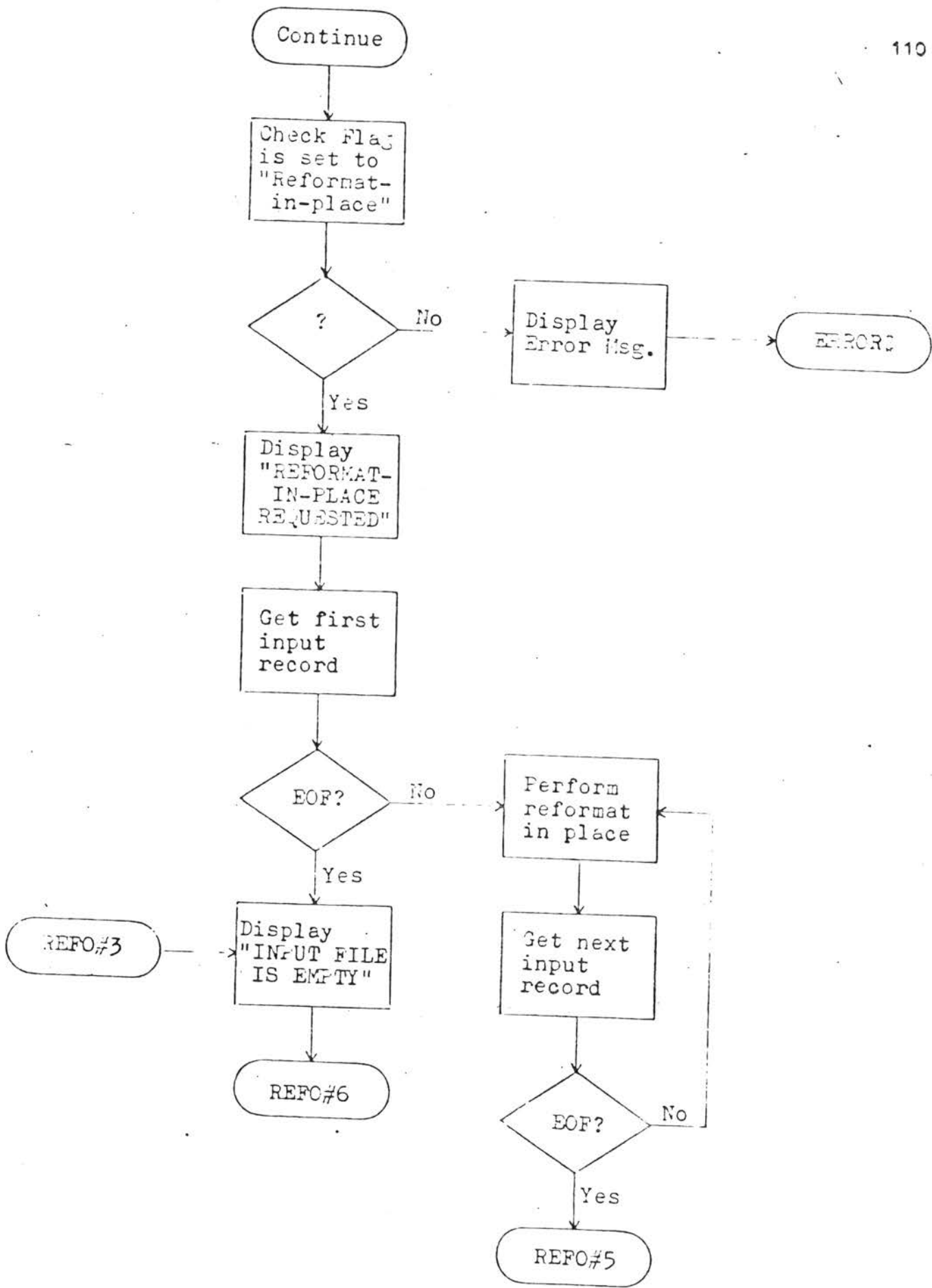
รูปที่ 3.4 แสดงขั้นตอนการทำงานของโปรแกรม REFORMAT/CMC



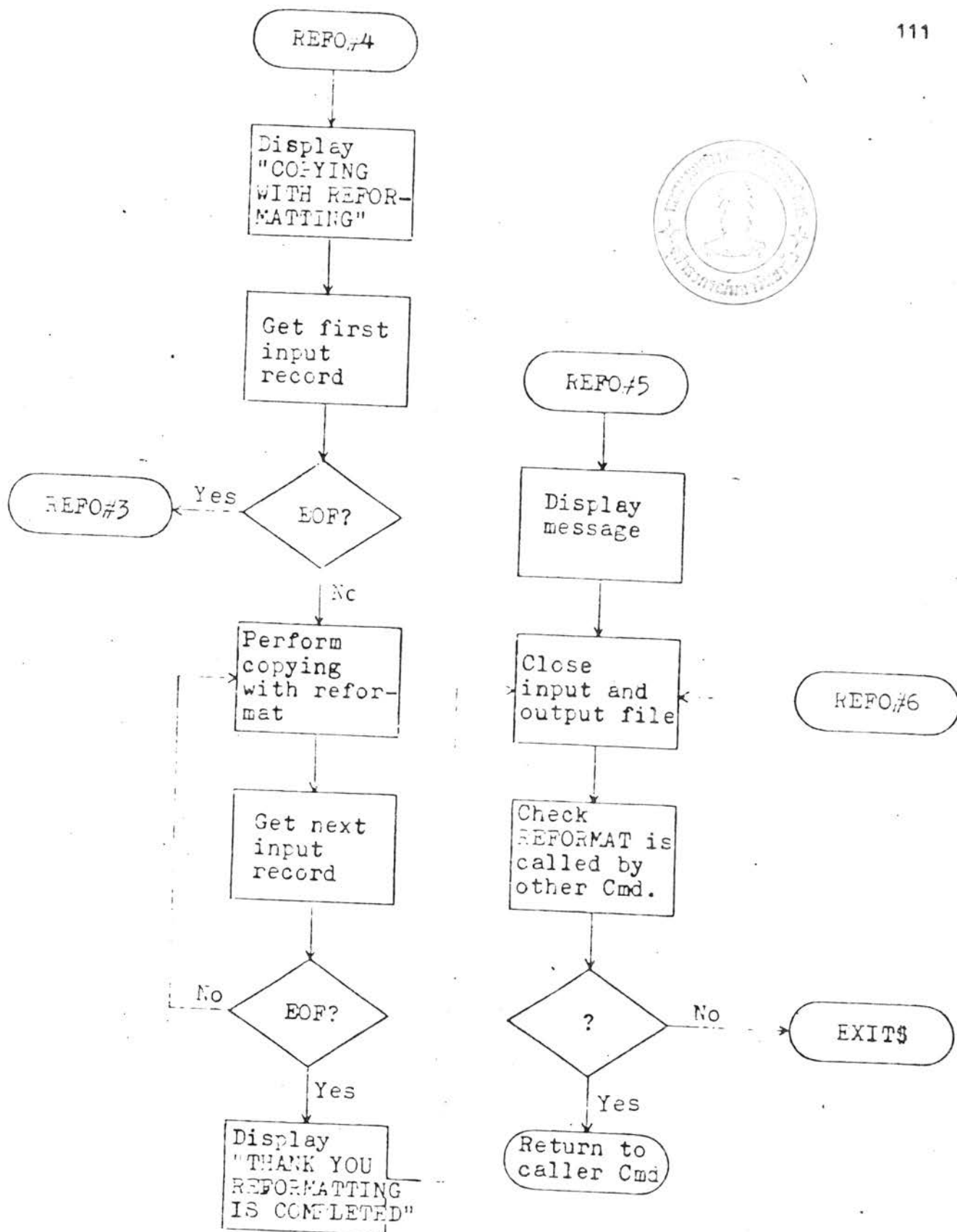
รูปที่ 3.4 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม REFORMAT/0.10



รูปที่ 3.4 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม REFORMAT/IN-PLACE



รูปที่ 3.4 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม REFORMAT/IN-PLACE



รูปที่ 3.4 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม REFORMAT/CMD

3.4 โปรแกรมที่ใช้ในการถ่ายเทข้อมูลระหว่างเครื่องงานแม่เหล็กและเครื่องเทปแม่เหล็ก (TAPE/CMD) TAPE/CMD เป็นโปรแกรมที่ถูกสร้างขึ้นมาเพื่อใช้ในการถ่ายเทข้อมูลระหว่างเครื่องเทปแม่เหล็กและเครื่องงานแม่เหล็ก และ/หรือเครื่องพิมพ์ ขนาดของบอออกที่ใช้ในการบันทึกข้อมูลของบนเทปแม่เหล็กมีค่าได้ไม่เกิน 1057 ตัวอักษร ในการอ่านข้อมูลจากเทปแม่เหล็กออกมาบันทึกยังงานแม่เหล็ก ข้อมูลส่วนที่เป็นที่ว่างจะถูกทำการลัดเข้าด้วยกันเพื่อประหยัดเนื้อที่ในงานแม่เหล็ก ลักษณะการบันทึกข้อมูลของบนเทปแม่เหล็กจะเป็นแบบ Fixed Block หรือ Fixed Unblock เท่านั้น

3.4.1 การเรียกใช้โปรแกรม TAPE/CMD มีรูปแบบการเรียกใช้ดังนี้

TAPE [<filespec>] [<label code>/<Code Set>]

<filespec> เป็นชื่อของพื้นที่ข้อมูลที่ต้องการนำข้อมูลขึ้นไปบันทึกของบนเทปแม่เหล็ก หรือเป็นชื่อของพื้นที่ข้อมูลที่ต้องการสร้างเริ่มหรือมีอยู่แล้วในงานแม่เหล็ก ในกรณีที่ต้องการนำข้อมูลจากเทปแม่เหล็กออกมาบันทึกยังงานแม่เหล็ก

<label code> เพื่อบอกให้ทราบว่าบนเทปมีเอเบอแบบใด หรือเพื่อบอกให้โปรแกรม TAPE/CMD ทำการบันทึกเอเบอที่ผู้ใช้ต้องการในกรณีที่ต้องการบันทึกข้อมูลของบนเทปแม่เหล็ก การกำหนดเอเบอสามารถทำได้โดยใส่ตัวอักษรต่อไปนี้

- N - ไม่มีเอเบอ (No Label)
- I - เอเบอตามแบบ IBM (IBM Standard Label)
- A - เอเบอตามแบบมาตรฐาน (ASCII Standard Label)

ถ้าหากไม่กำหนดจะถือว่า label code มีค่าเป็น "A"

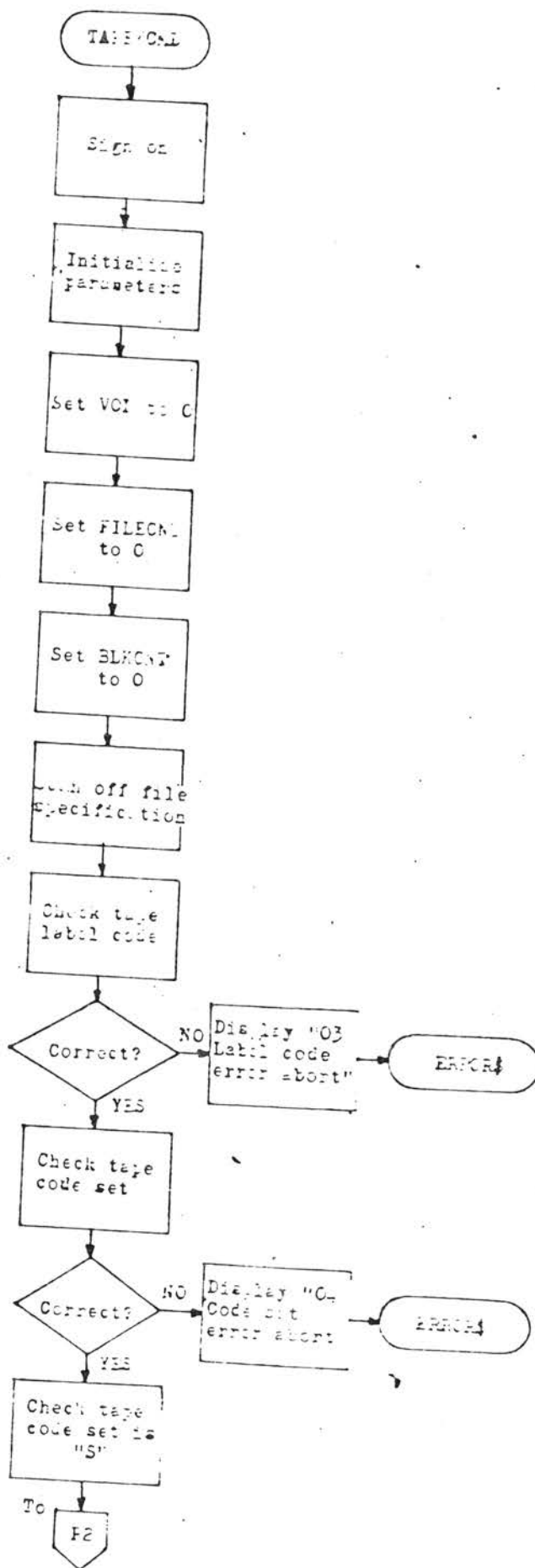
<Code Set> เพื่อบอกให้ทราบว่าข้อมูลที่ต้องการบันทึกของบนเทปแม่เหล็กหรือข้อมูลที่บันทึกอยู่บนเทปแม่เหล็กได้ใช้รหัสแบบใดในการบันทึก กำหนดได้โดยใส่ตัวอักษรดังต่อไปนี้

- A - ใช้รหัสแบบ ASCII
- E - ใช้รหัสแบบ EBCDIC
- S - ใช้รหัสแบบ ASCII โดยมีขนาดความยาวของบอออกเป็น 1012 ตัวอักษร และมีขนาดความยาวของเรคคอร์ดเป็น 253 ตัวอักษร

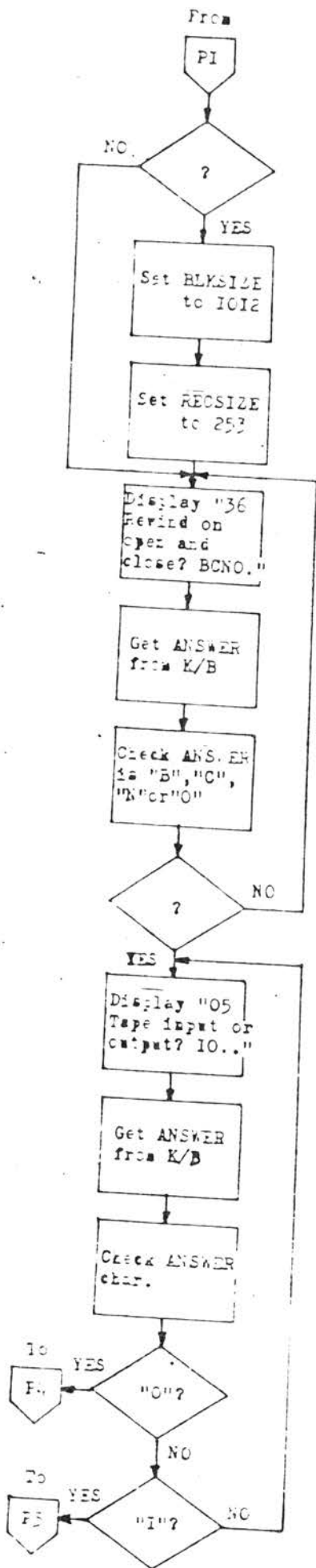
ถ้าหากไม่กำหนดจะถือว่า Code Set มีค่าเป็น "S"

01 DFS MAGNETIC TAPE UTILITY ROUTINE RELEASE A.A 04/25/76
 02 DRIVE NUMBER ERROR. ABORT.
 03 LABEL CODE ERROR. ABORT.
 04 CODE SET ERROR. ABORT.
 05 TAPE INPUT OR OUTPUT ID ...
 06 SHOULD ALL FILES ON DRX BE KILLED NY ...
 07 FILE 12345678/123 IS PROTECTED.
 08 NO 1234 LABEL BLOCK. ABORT.
 09 ENTER 6-CHARACTER NEW TAPE ID NUMBER ...
 09 ENTER OWNER'S 14 CHARACTER ID ...
 10 NONBLANK TAPE. HDRI_XXXXXXXXXXXXXXXXXXXXX CPX...
 11 ENTER C WHEN NEW PEEL MOUNTED ...
 12 ENTER C WHEN NEXT VOLUME MOUNTED ...
 13 NONBLANK TAPE. CPX ...
 14 NO FILES ON THIS TAPE. ABORT.
 15 BAD LABEL STRUCTURE. ABORT.
 16 FILENAME ...
 17 NO SUCH NAME.
 18 FILE 12345678/123 COPYING FROM DRX.
 19 CONTINUE TO NEW VOLUME NY ...
 20 FILE SEQUENCE ERROR. ACD ...
 21 PROCESS THIS FILE. 12345678/123 NY...
 22 UNACCEPTABLE BLOCKING FORMAT. AS ...
 23 ENTER BLOCKING FORMAT IN DECIMAL (M = CHAR/REC, N = REC/BLOCK) M MXN ...
 24 BAD MXN SPECS.
 25 PRINT OR DISK. BDP ...
 26 IS COLUMN 1 A CONTROL COLUMN NY ...
 27 FIRST 80 BYTES.
 28 FILE 12345678/123 ALREADY EXISTS ON DRX. ACPS ...
 29 FILE 12345678/123 IS WRITE PROTECTED. ACPS ...
 30 FILE 12345678/123 COPYING TO DRX.
 31 BLOCK COUNT ERROR. 123456 COUNTED 123456 ON TAPE LABEL. AD ...
 32 END OF COPY WITH 1234 VOLUMES AND 1234 FILES ON LAST VOLUME.
 33 END OF TAPE ENCOUNTERED.
 34 PARITY ERROR DURING X. ACR ...
 35 END OF VOLUME. MOUNT NEW NY ...
 36 REWIND ON OPEN AND CLOSE BCND ...
 NO WRITE RING --- ABORT
 PARITY ERROR DURING DISK READ

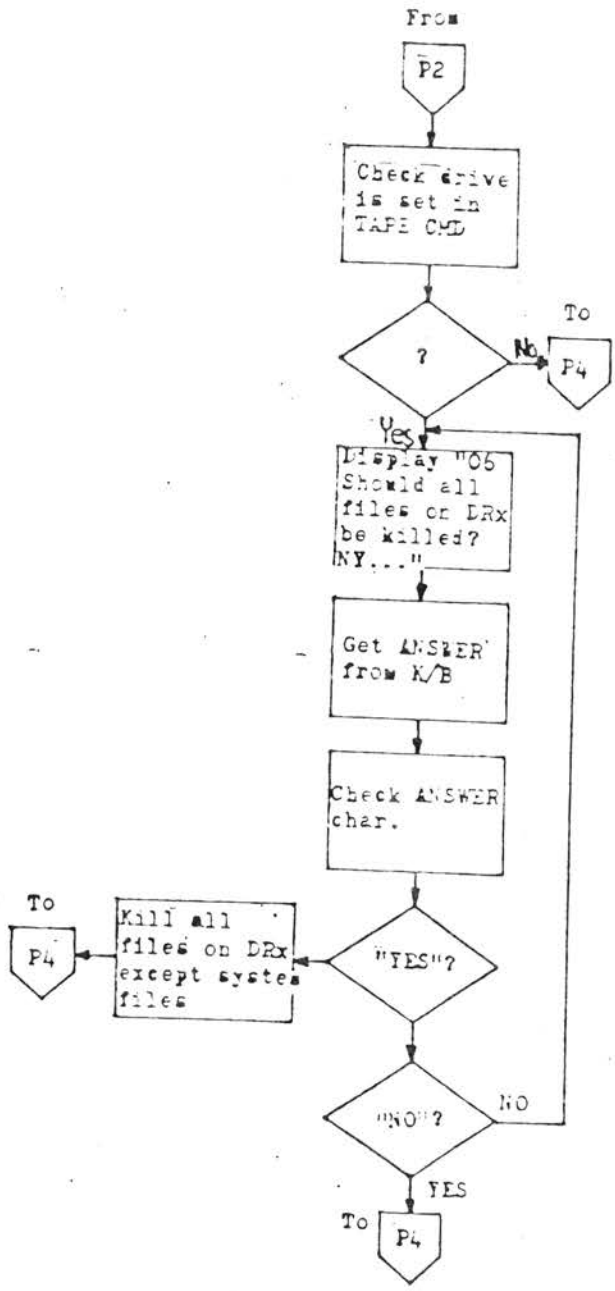
3.5 MESSAGE MESSAGES FOR TAPE/CMD



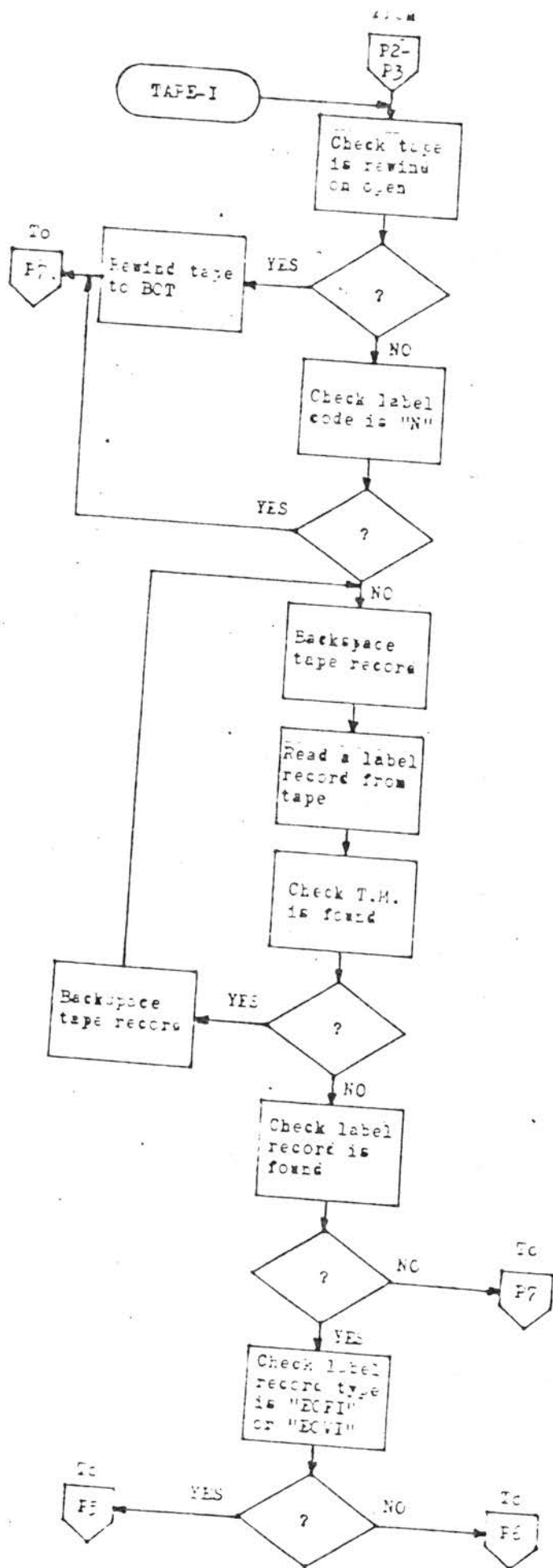
รูปที่ 3.6 แสดงขั้นตอนการทำงานของโปรแกรม TAPE/CMD



รูปที่ 3.6 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม TAPE/CMO



รูปที่ 3.6 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม TAPE/CMD

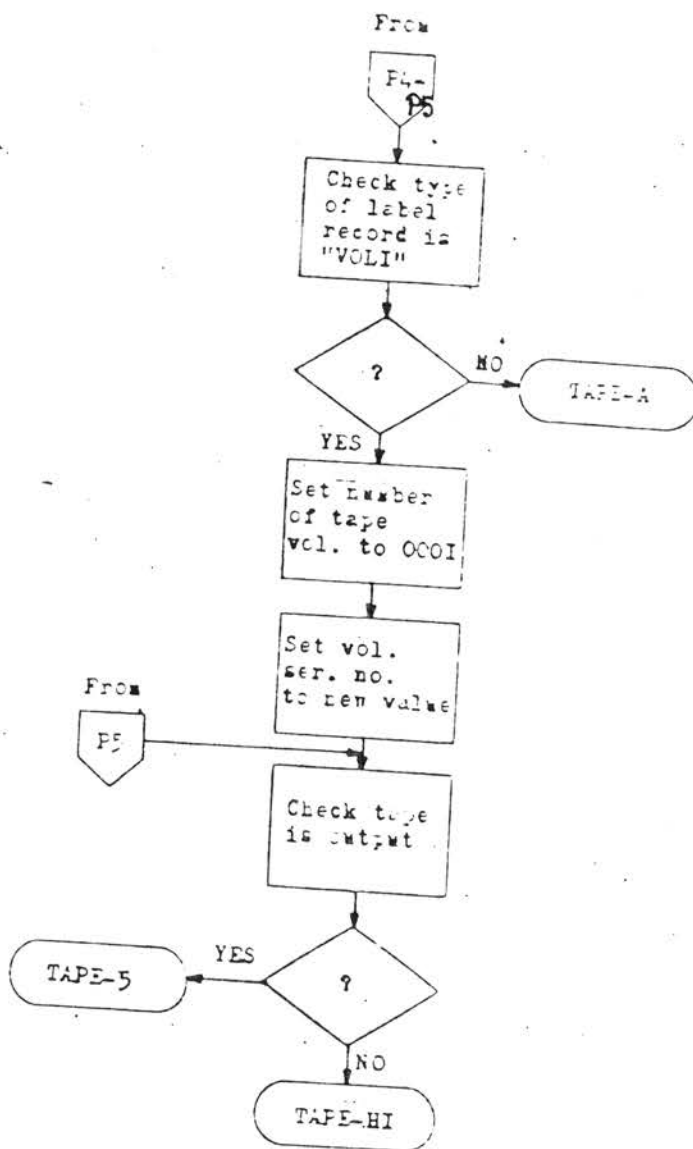


รูปที่ 3.6 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม TAPE/CMD

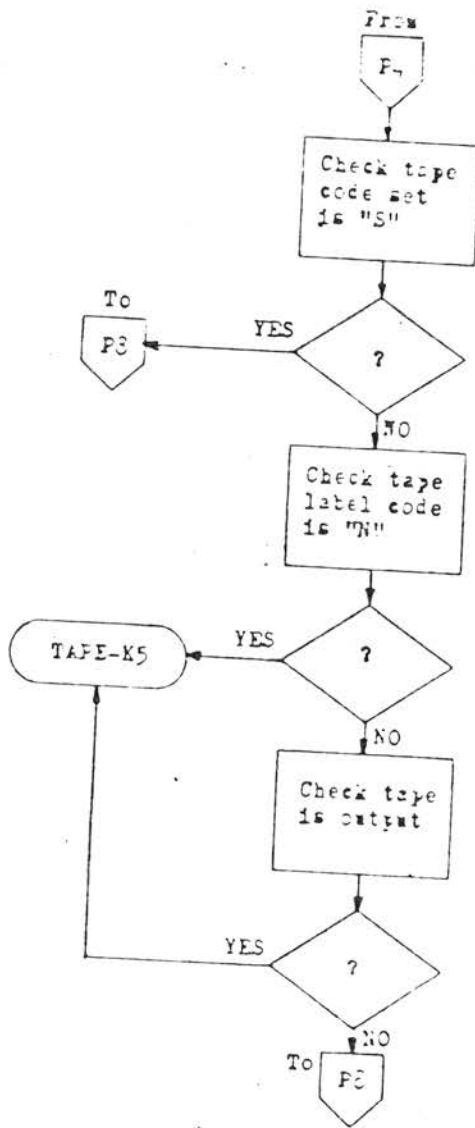
P5



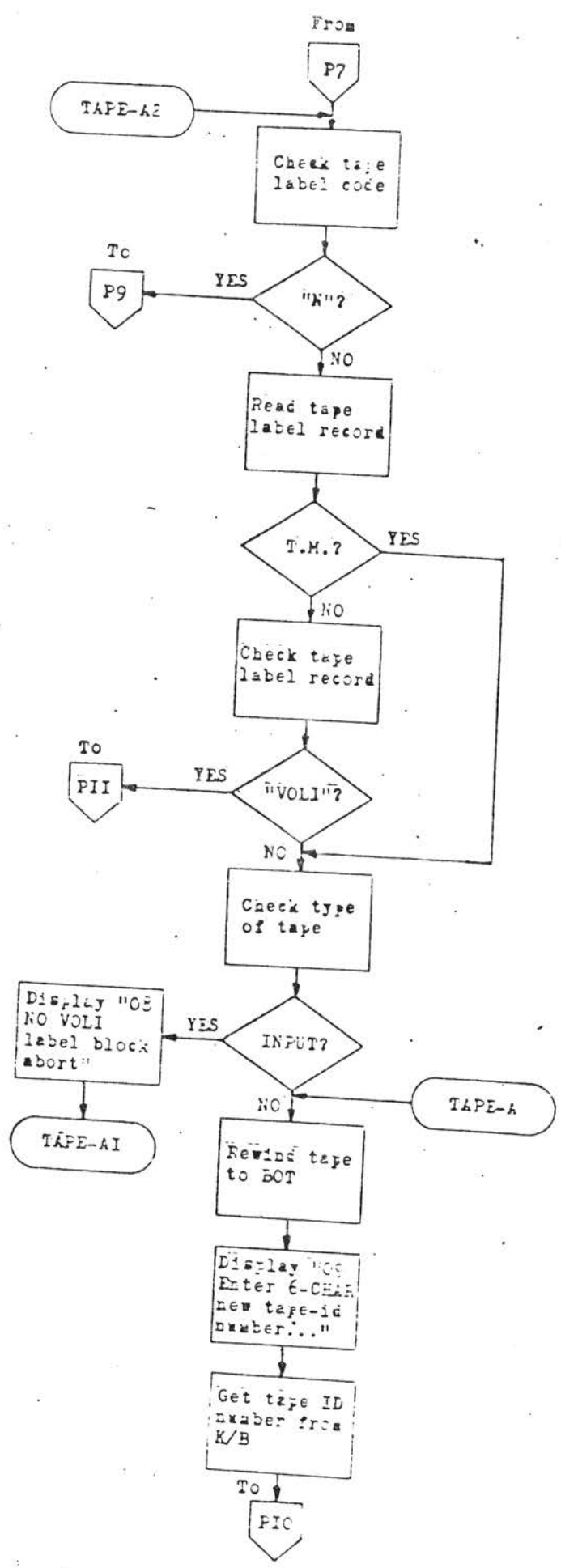
รูปที่ 3.6 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม TAPE/CMD



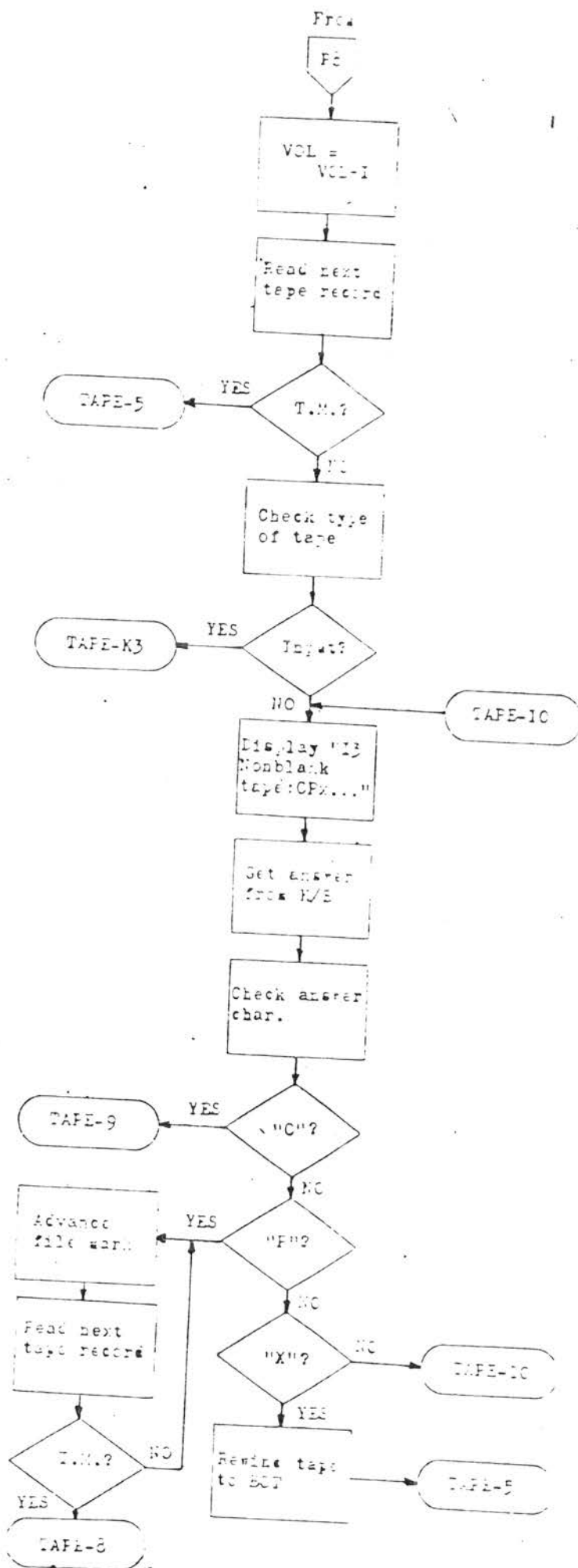
รูปที่ 3.6 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม TAPE/CMO



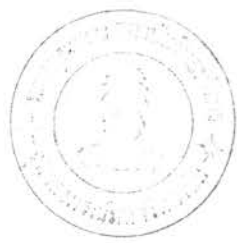
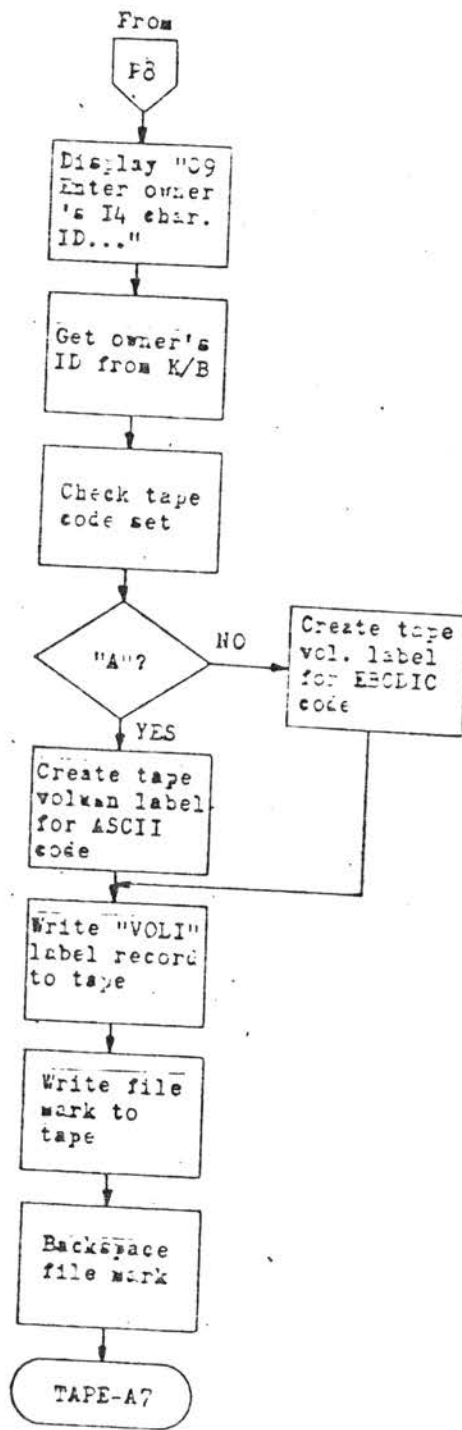
รูปที่ 3.6 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม TAPE/CMD



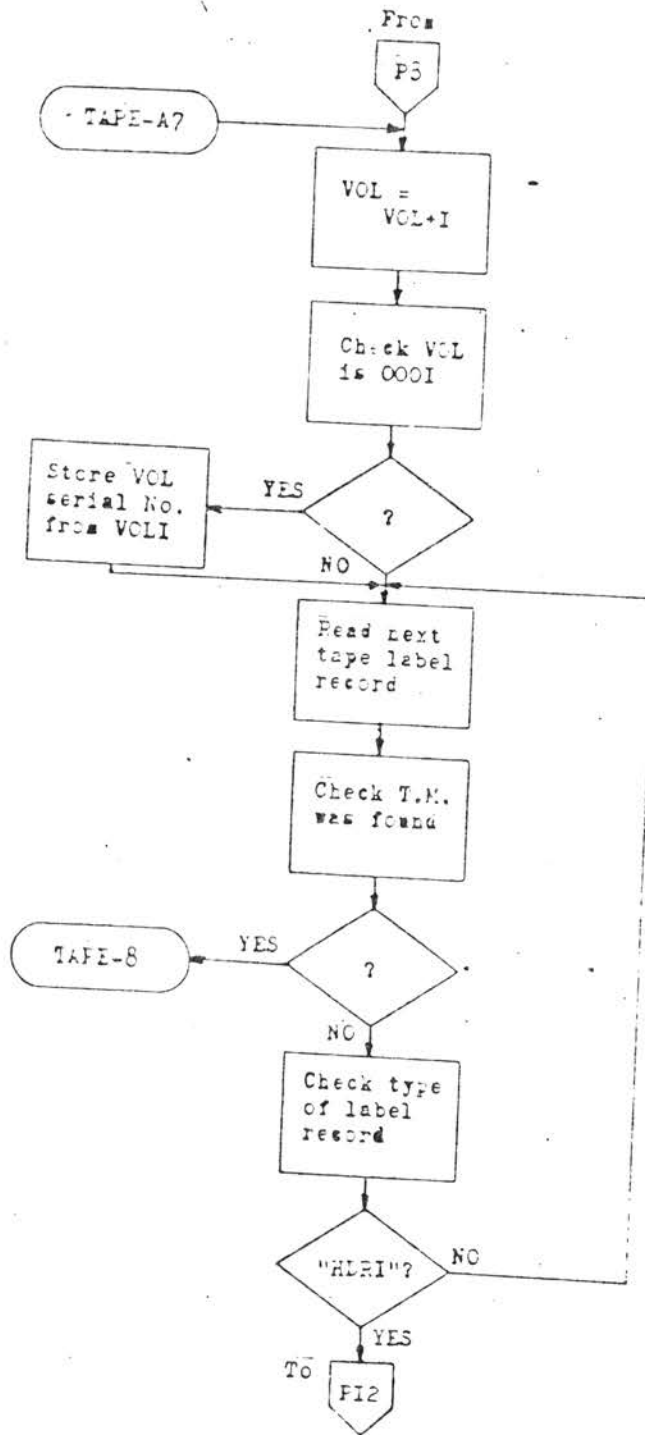
รูปที่ 3.6 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม TAPE/CMD



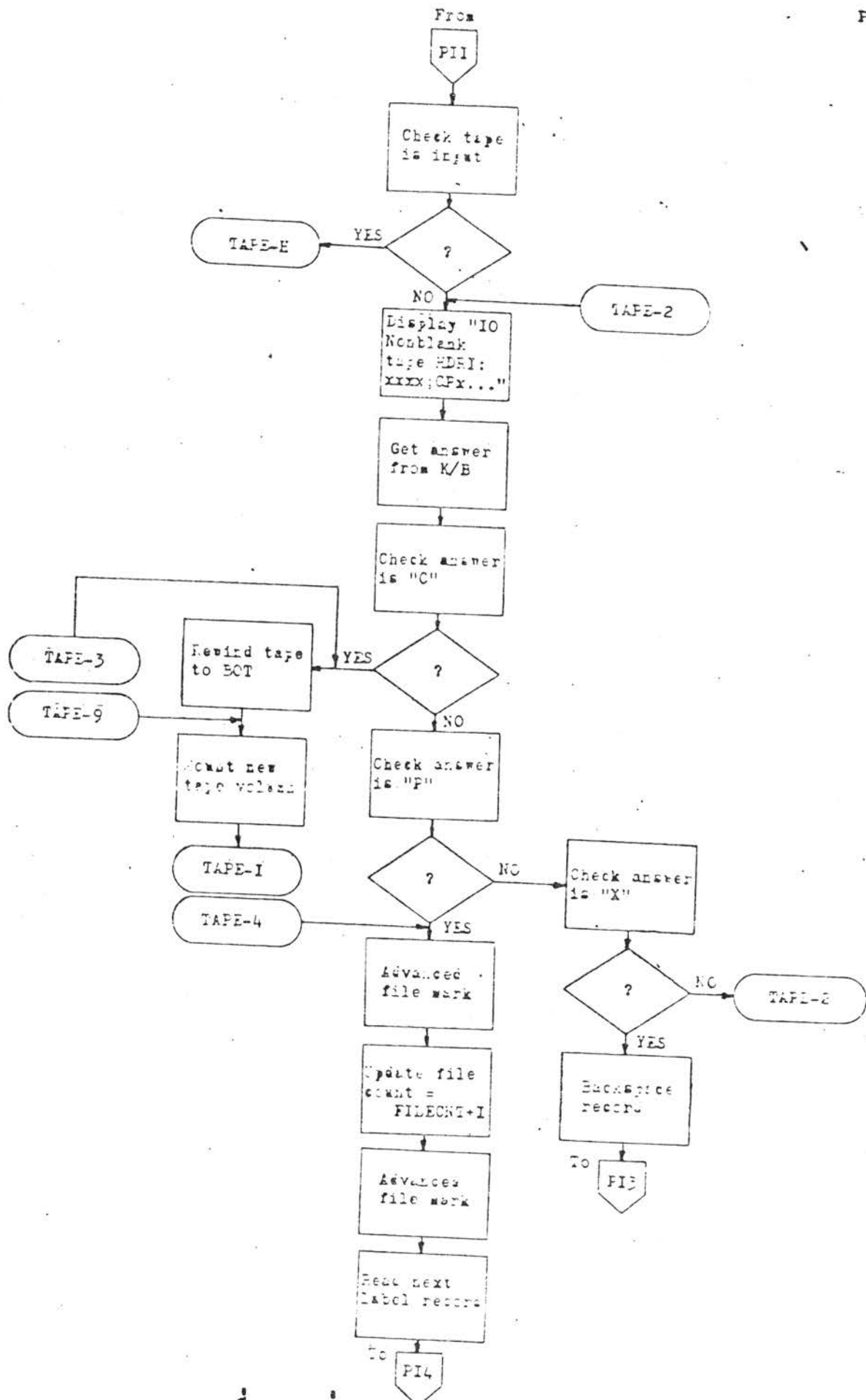
รูปที่ 3.6 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม TAPE/CMD



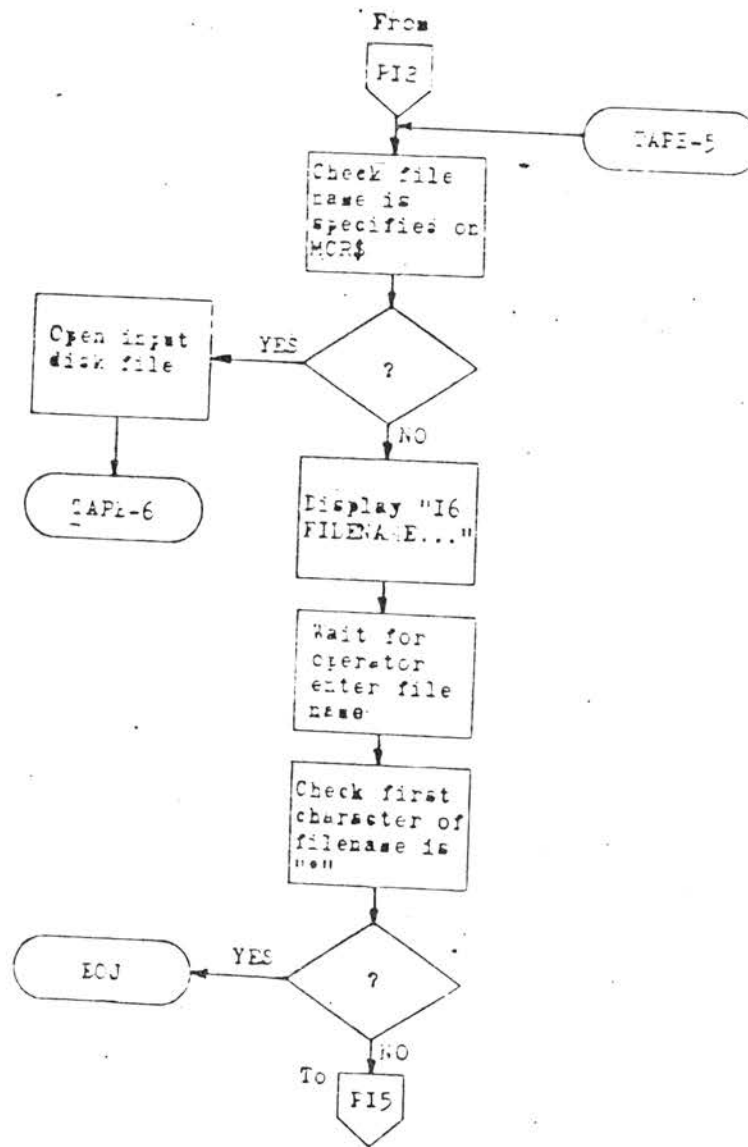
รูปที่ 3.6 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม TAPE/CMD



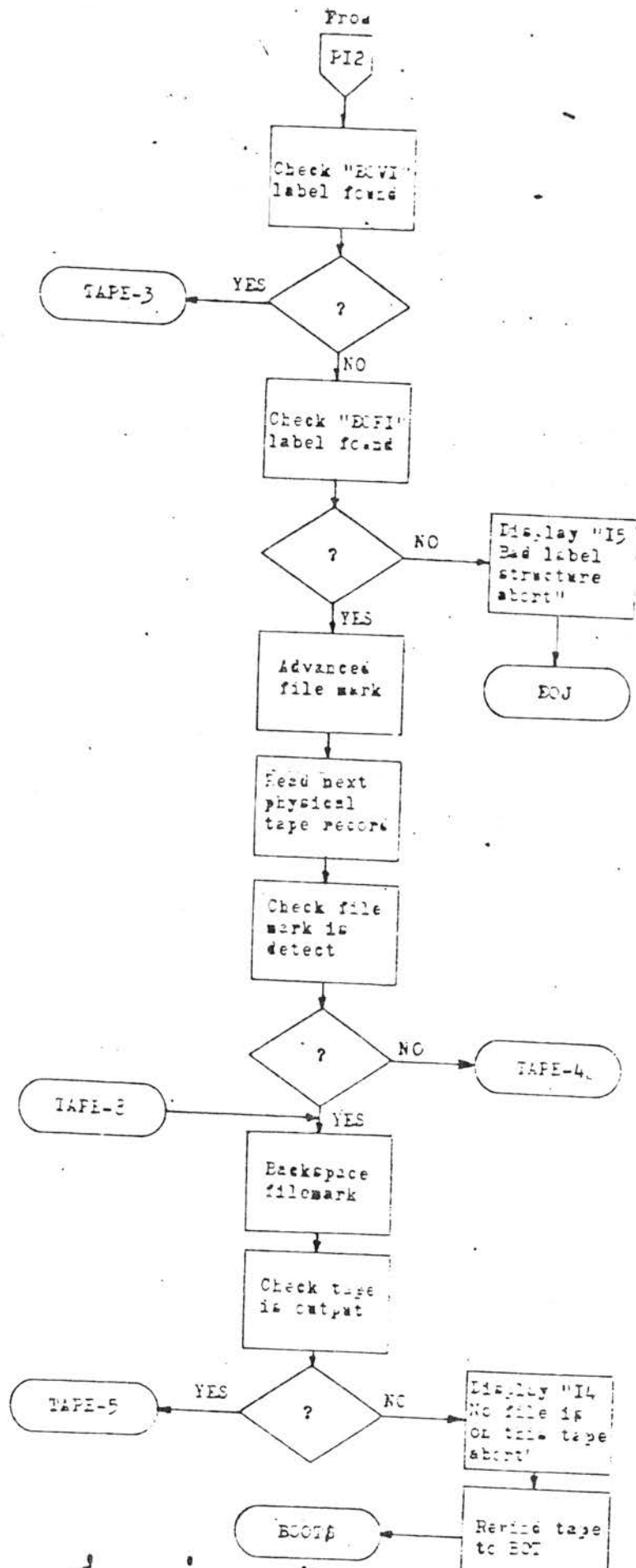
รูปที่ 3.6 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม TAPE/CMD



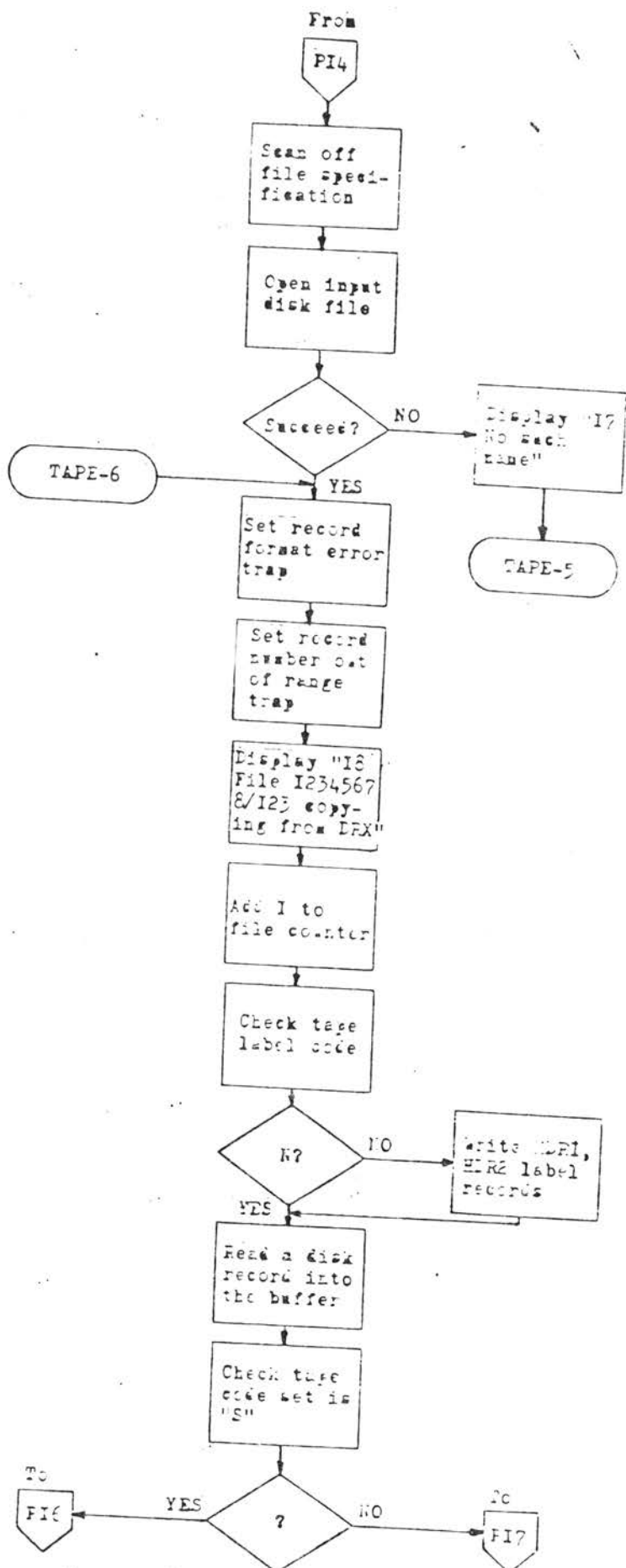
รูปที่ 3.6 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม TAPE/CP



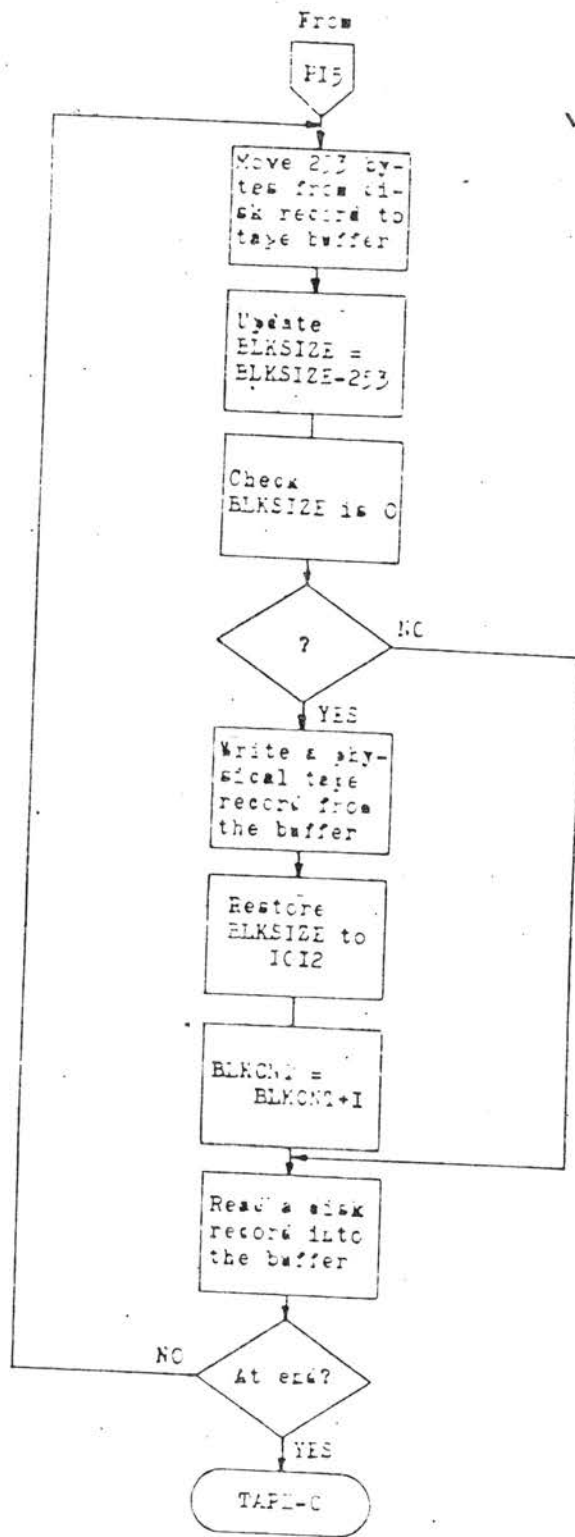
รูปที่ 3.6 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม TAPF/CMD



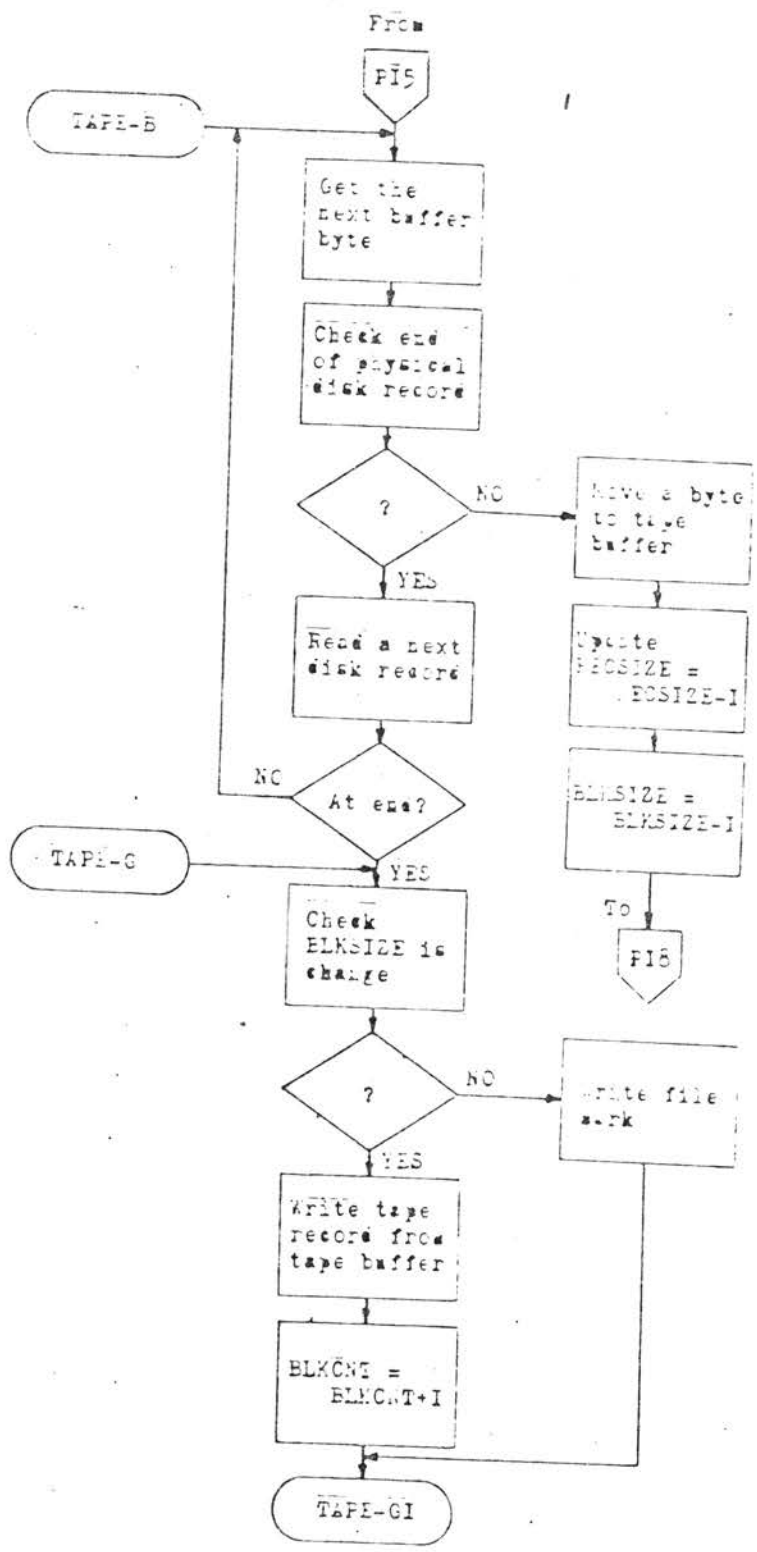
รูปที่ 3.6 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม TAPE/CMD



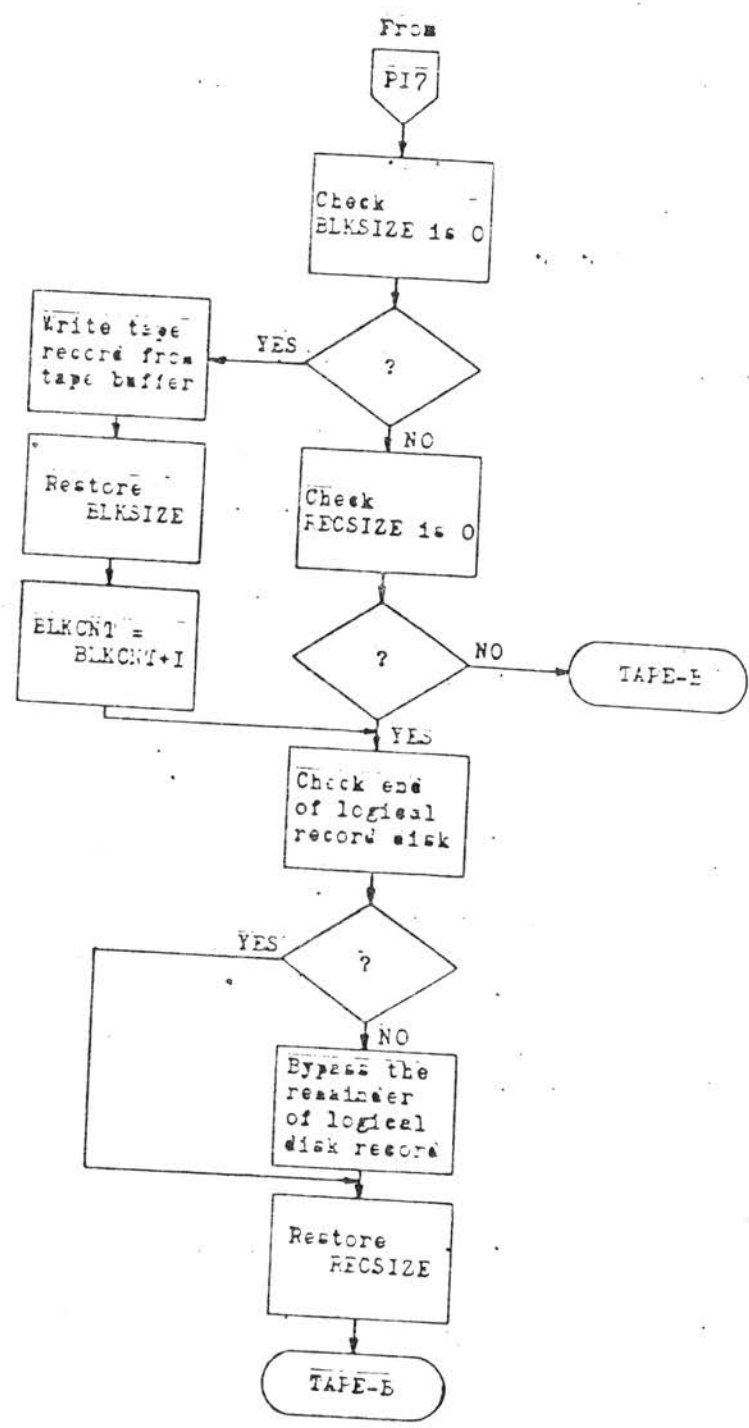
รูปที่ 3.6 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม TAPE/CMD



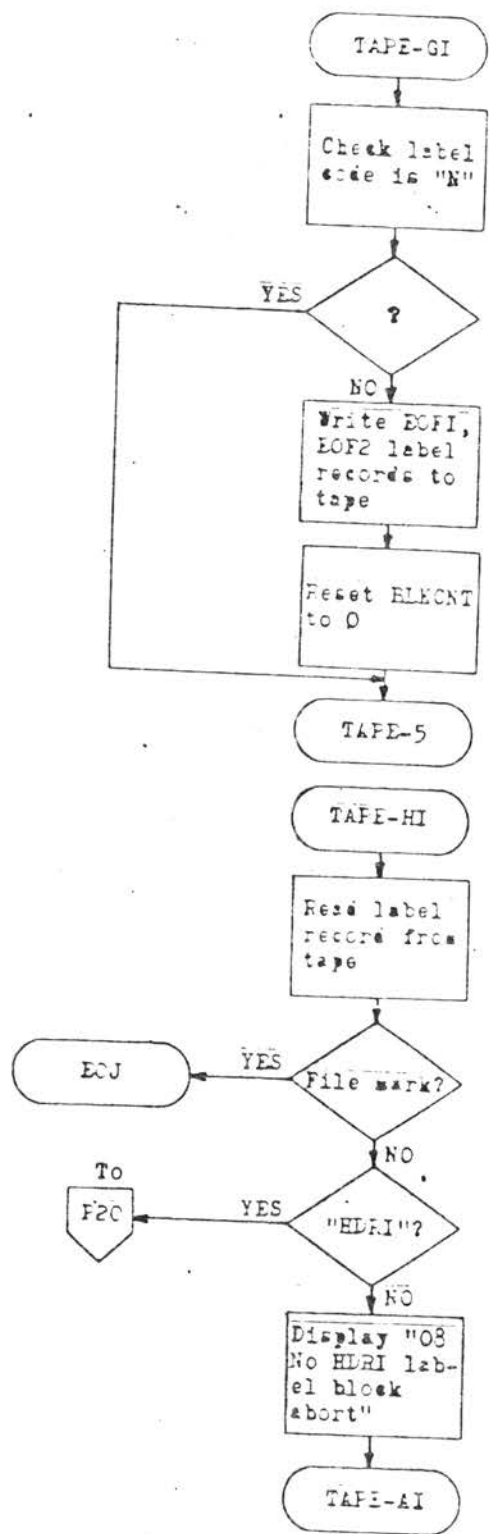
รูปที่ 3.6 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม TAPF/CMD



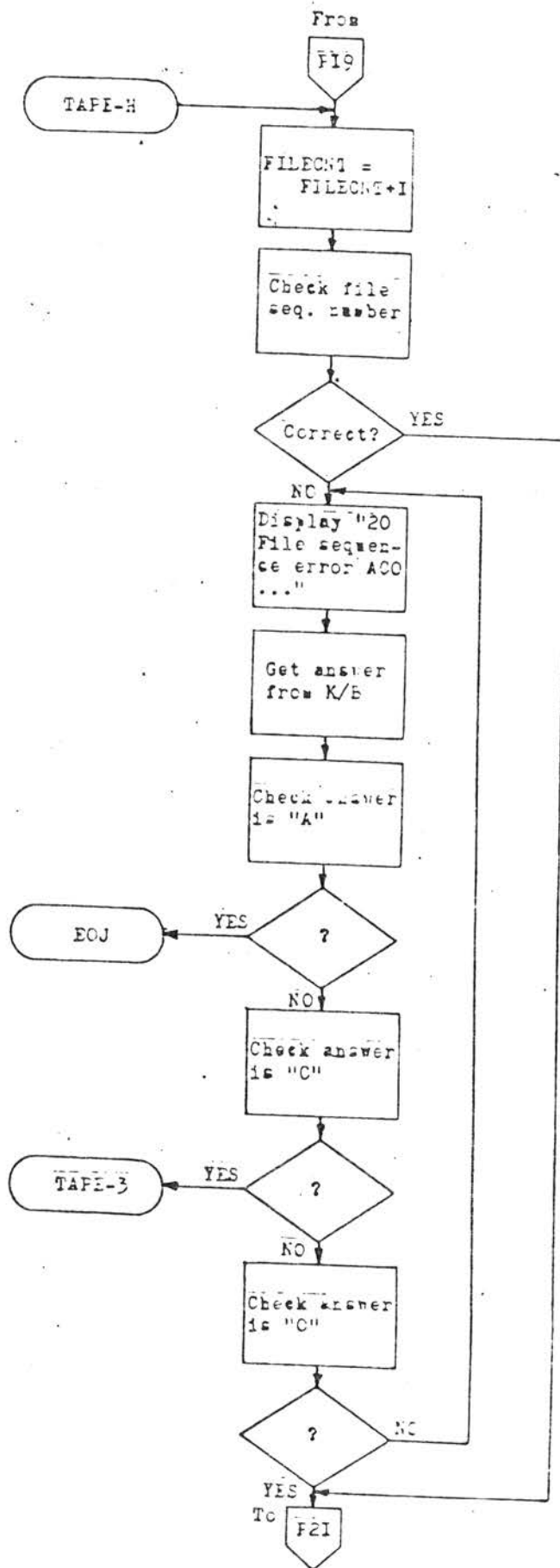
รูปที่ 3.6 (ก) แสดงขั้นตอนการทำงานของโปรแกรม TAPE/CMD



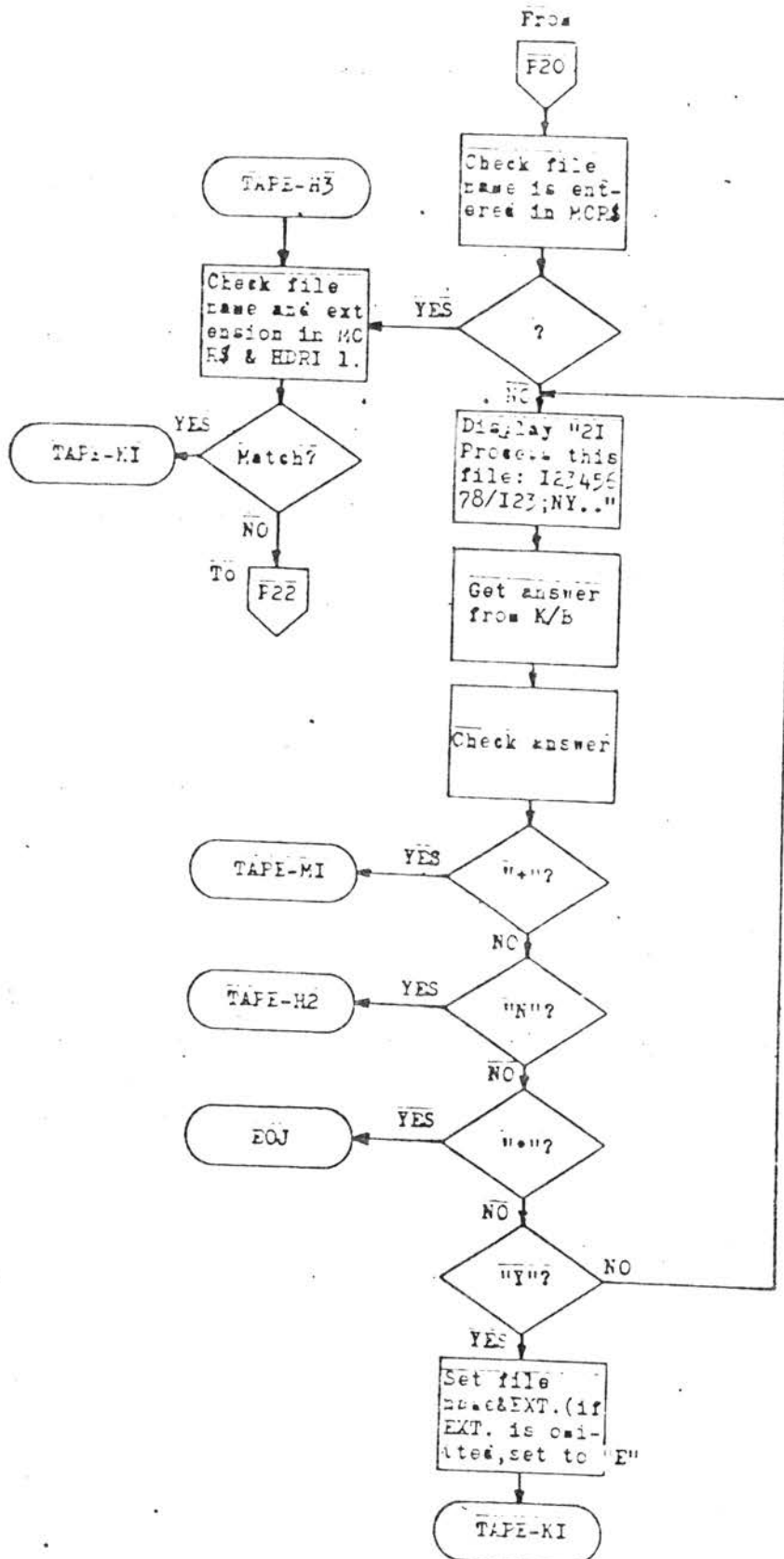
รูปที่ 3.6 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม TAPE/CMD



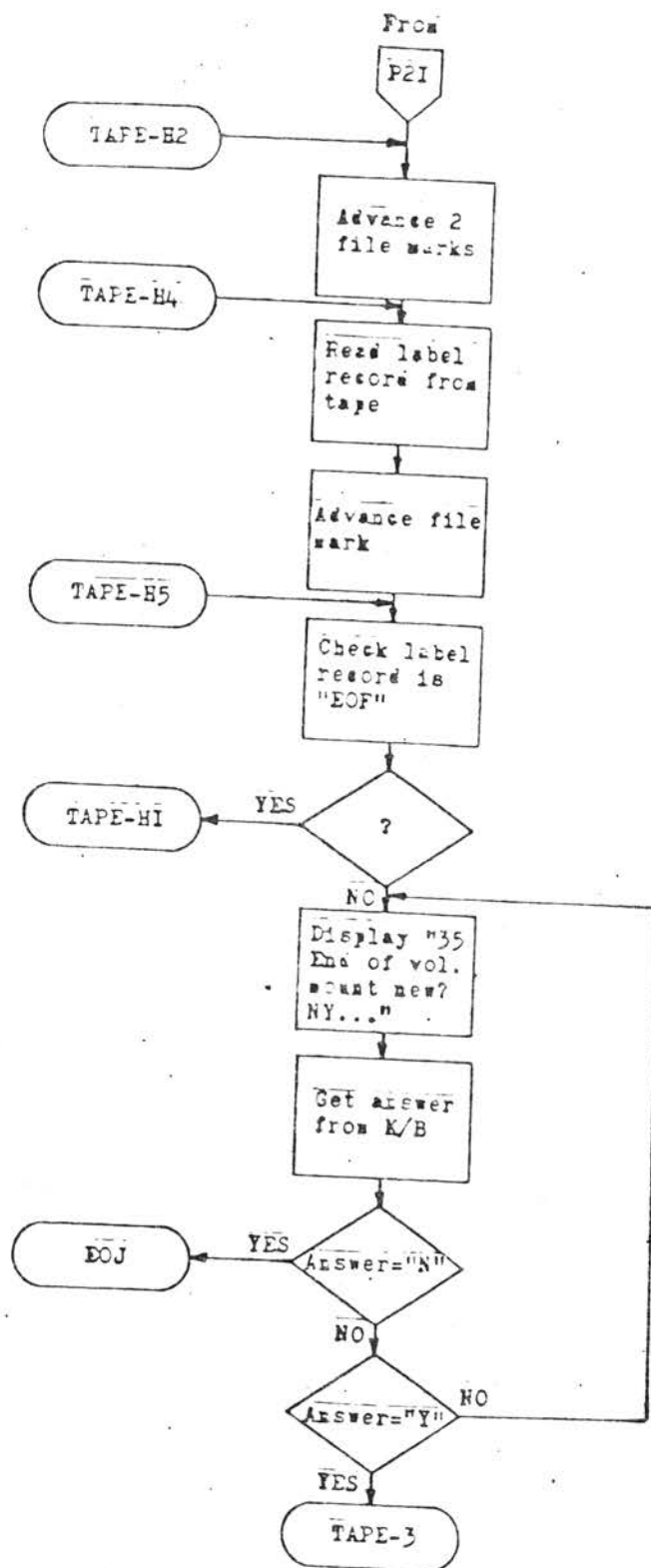
รูปที่ 3.6 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม TAPE/CMC



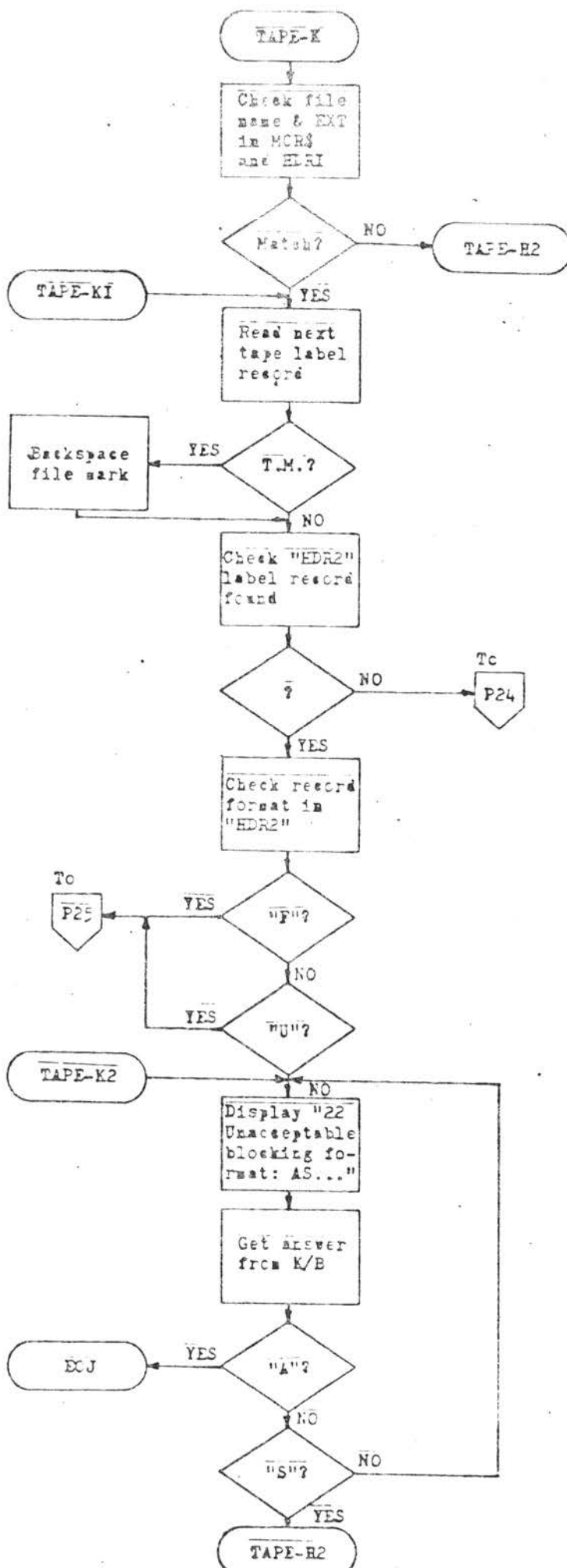
รูปที่ 3.6 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม TAPE/CMD

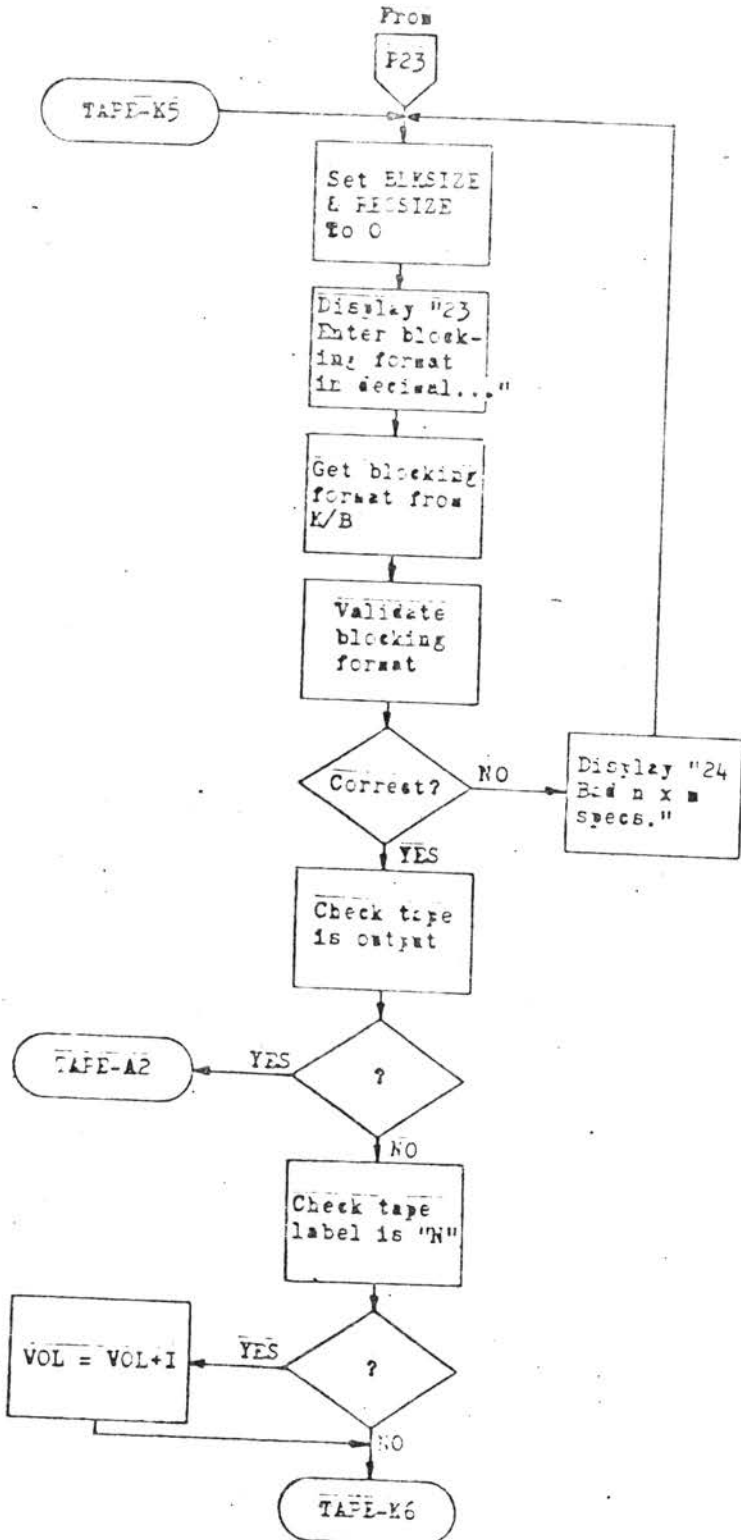


รูปที่ 3.6 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม TAPE/CMD

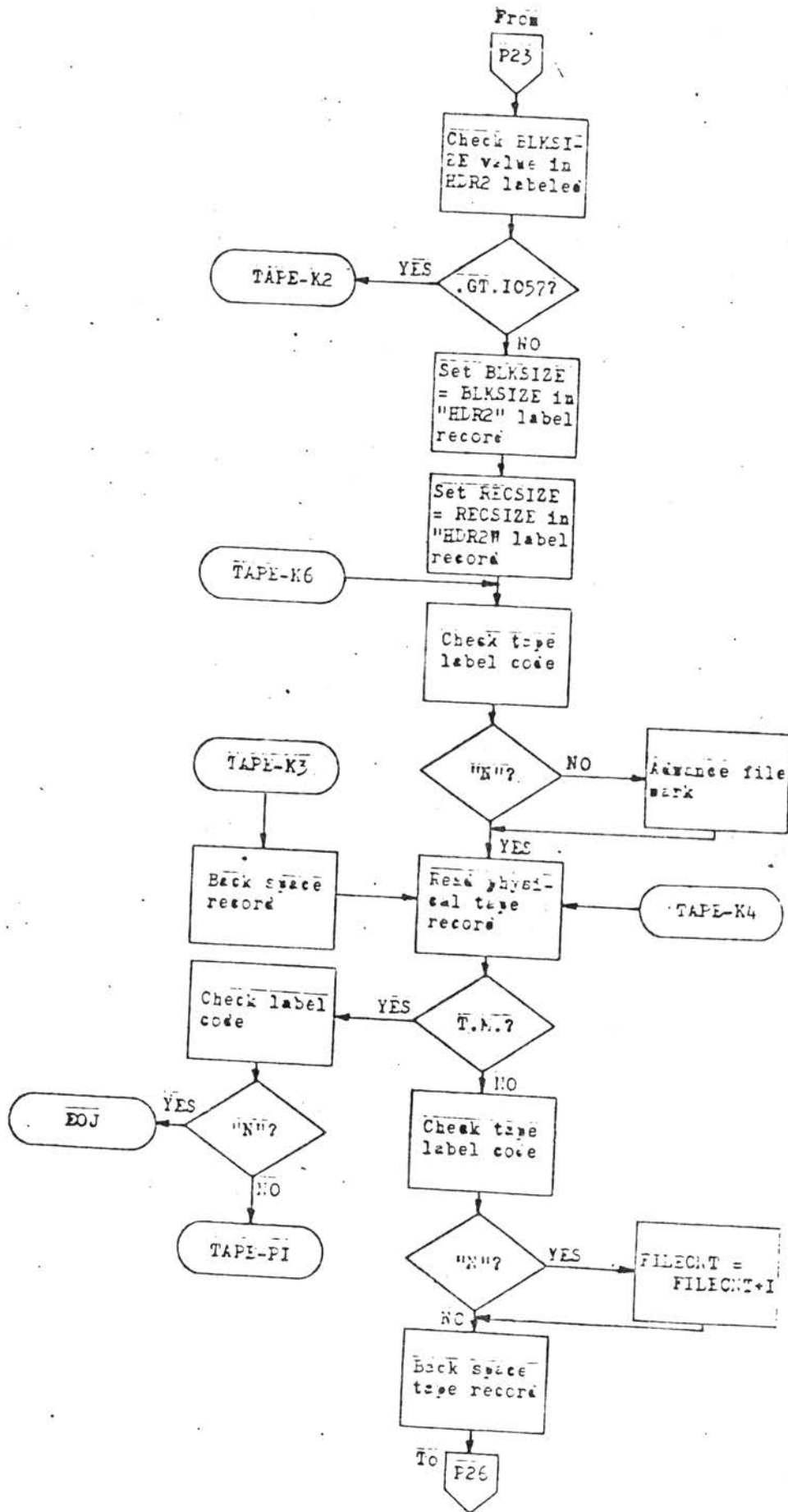


รูปที่ 3.6 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม TAPE/CMD

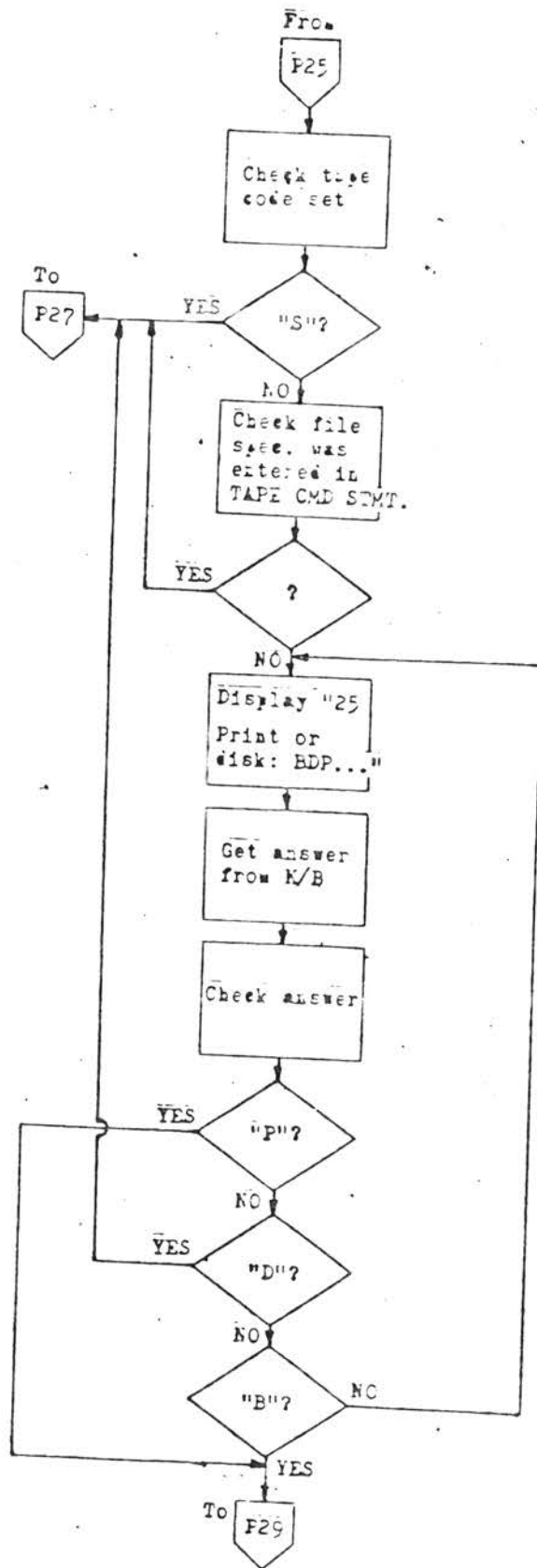




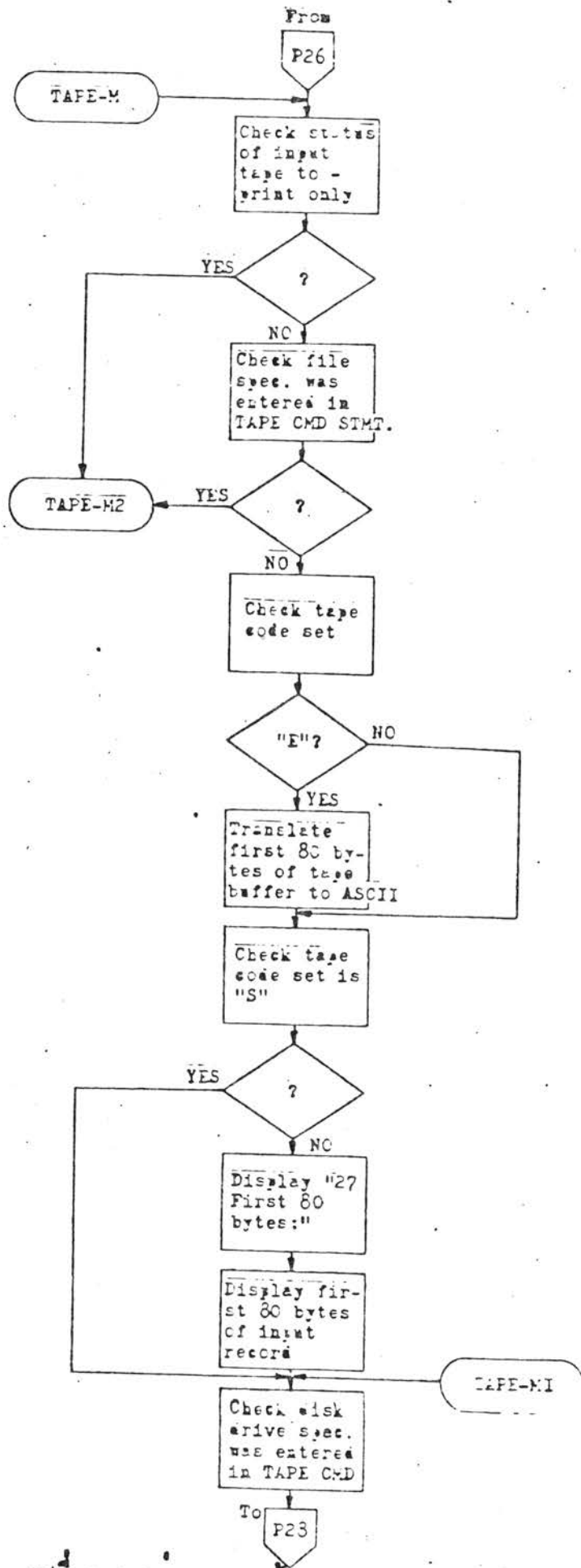
รูปที่ 3.6 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม TAPE/CMD



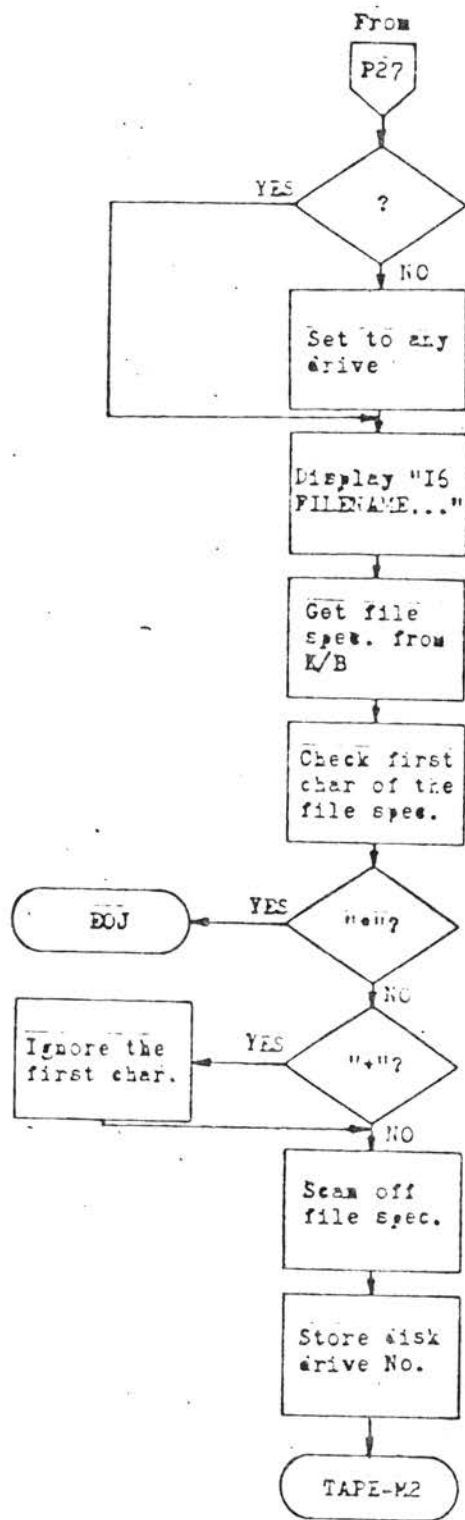
รูปที่ 3.6 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม TAPE/CMO



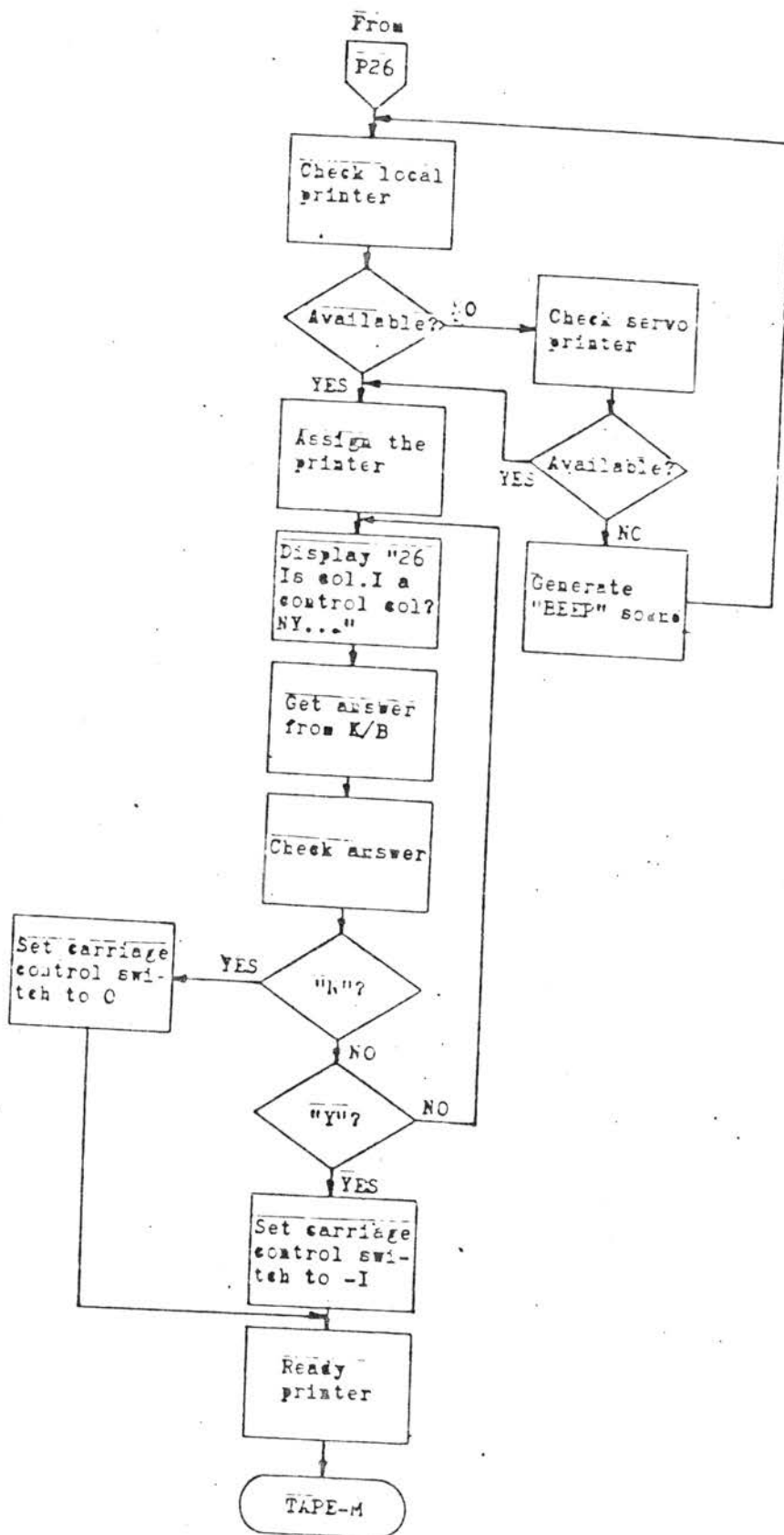
รูปที่ 3.6 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม TAPE/CMD



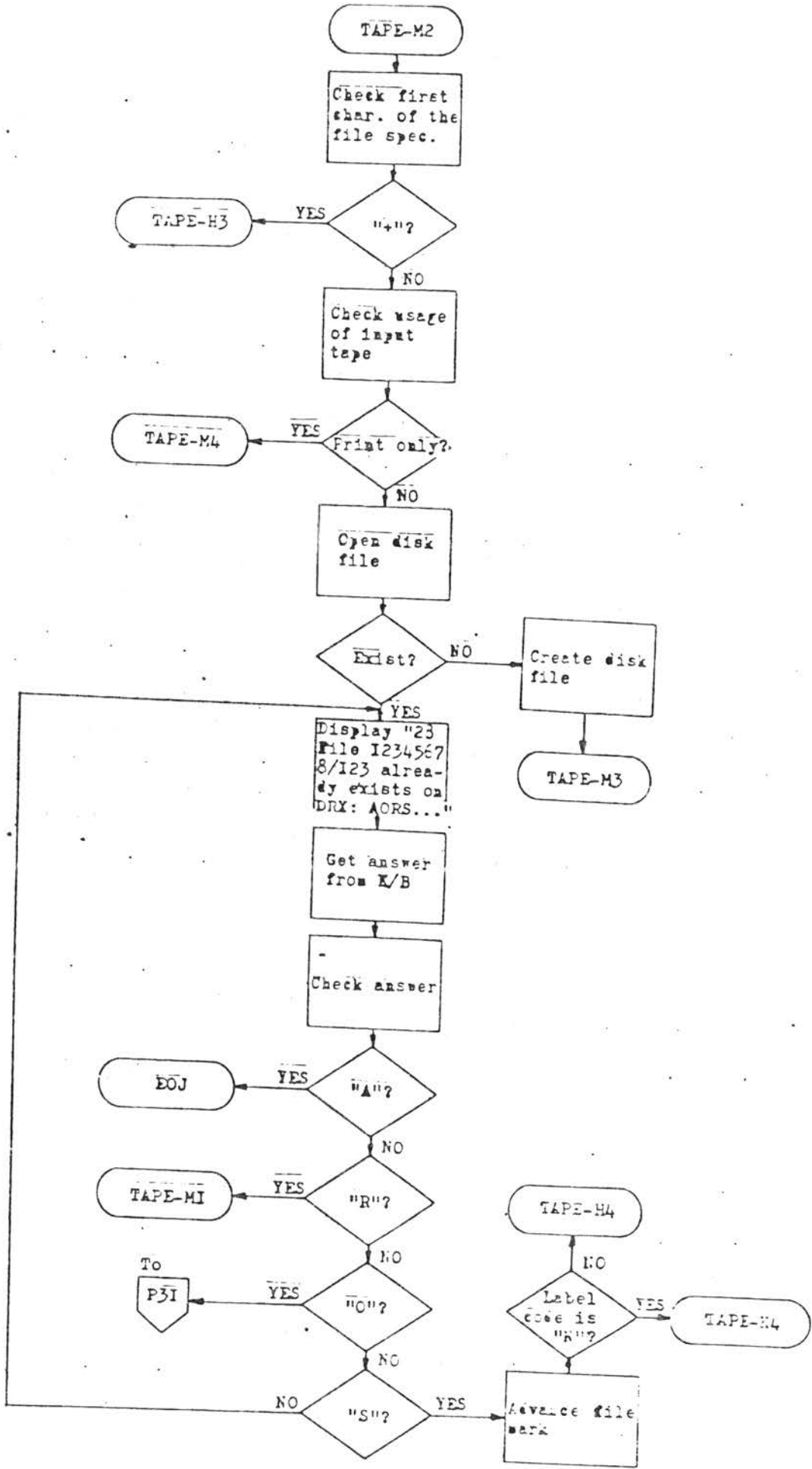
รูปที่ 3.6 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม TAPE/CMD



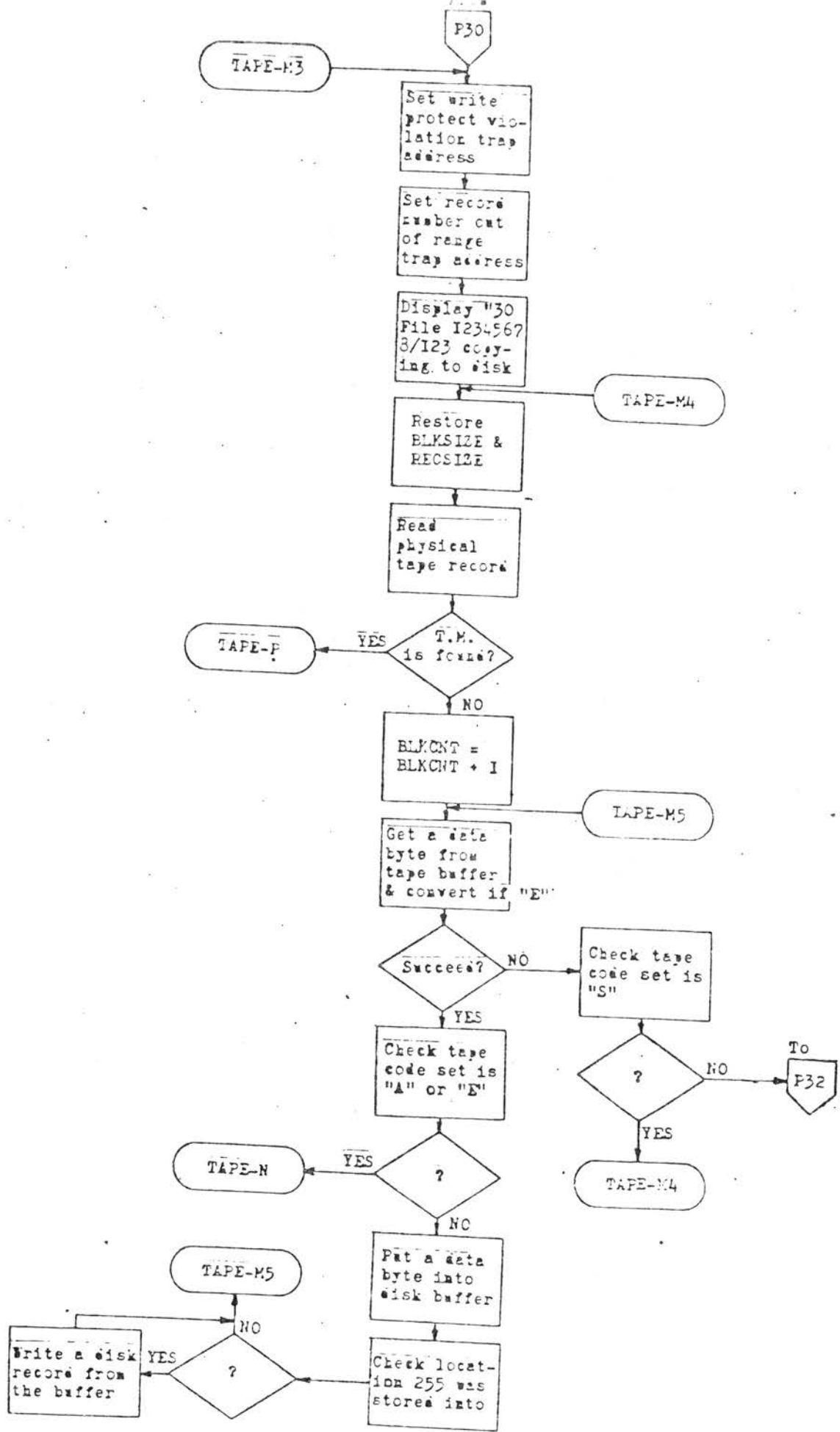
รูปที่ 3.6 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม TAPE/CMO



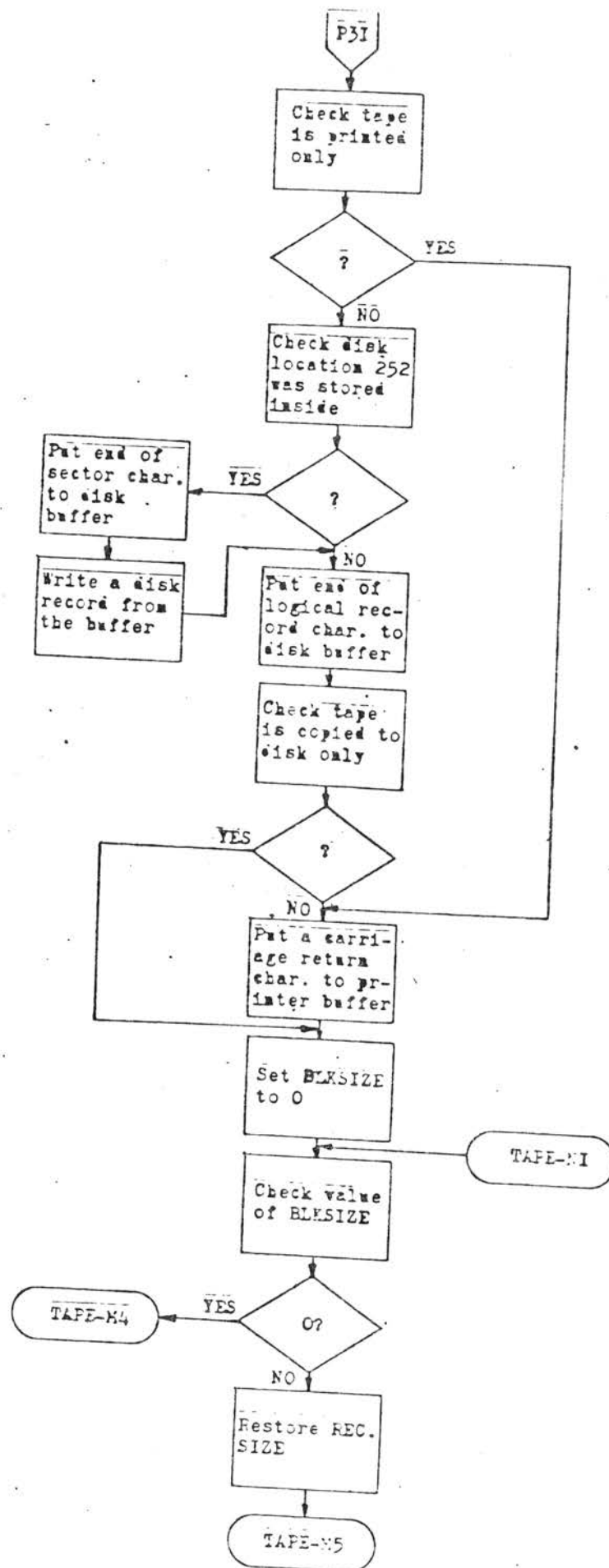
รูปที่ 3.6 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม TAPE/CMD



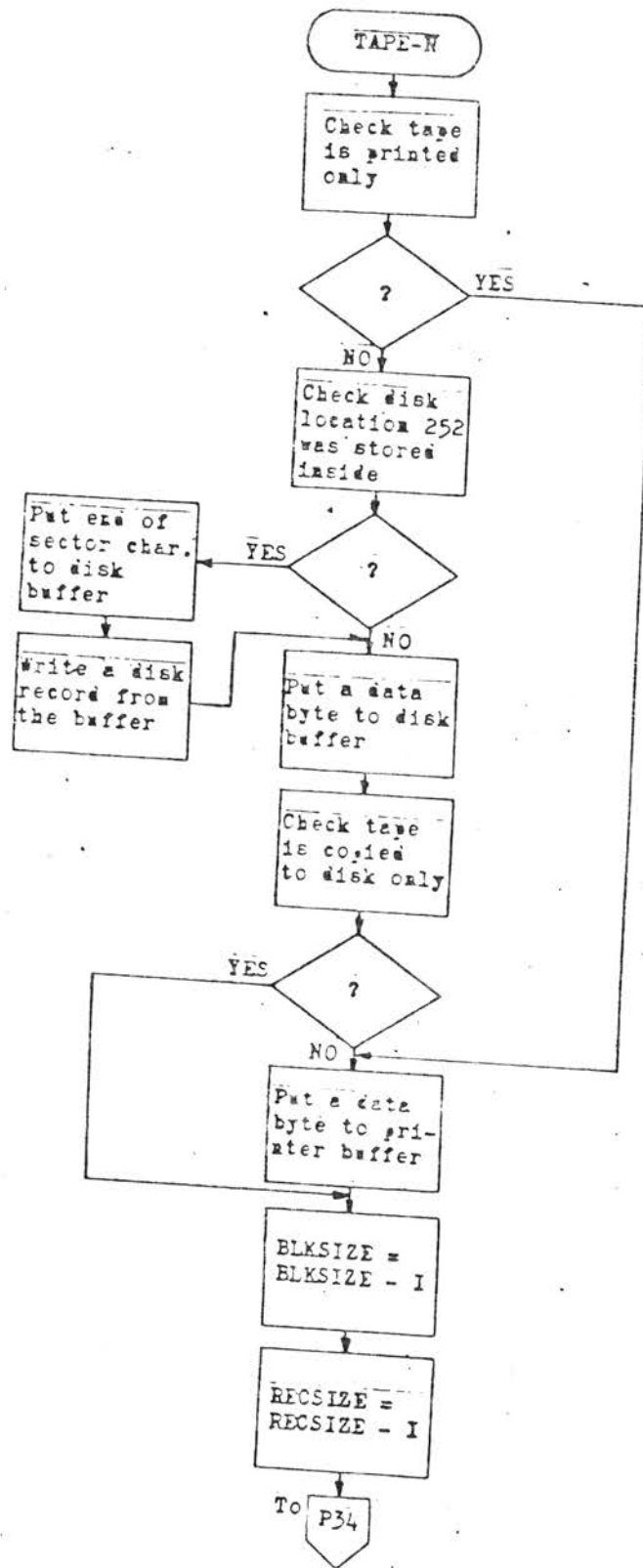
รูปที่ 3.6 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม TAPE/CMD



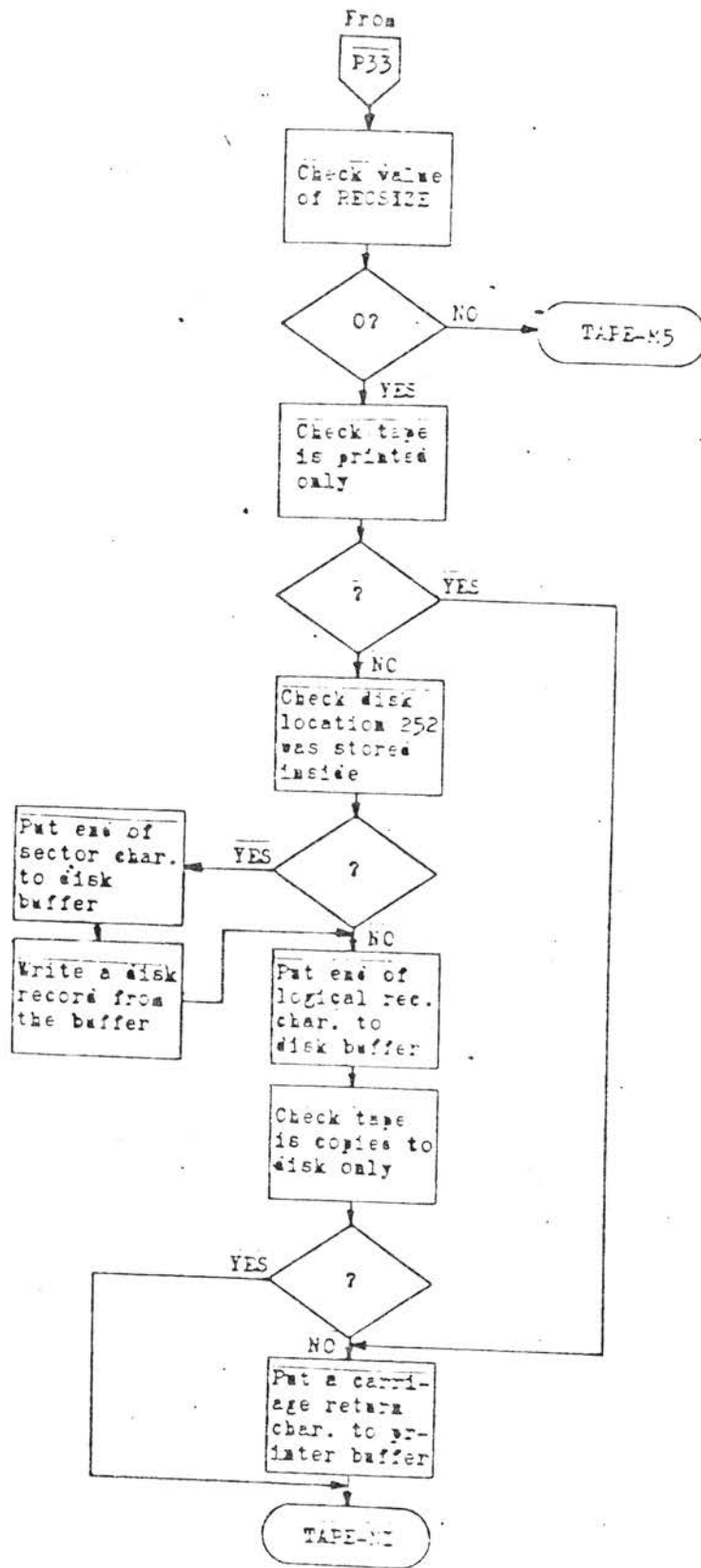
รูปที่ 3.6 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม TAPE/CMD



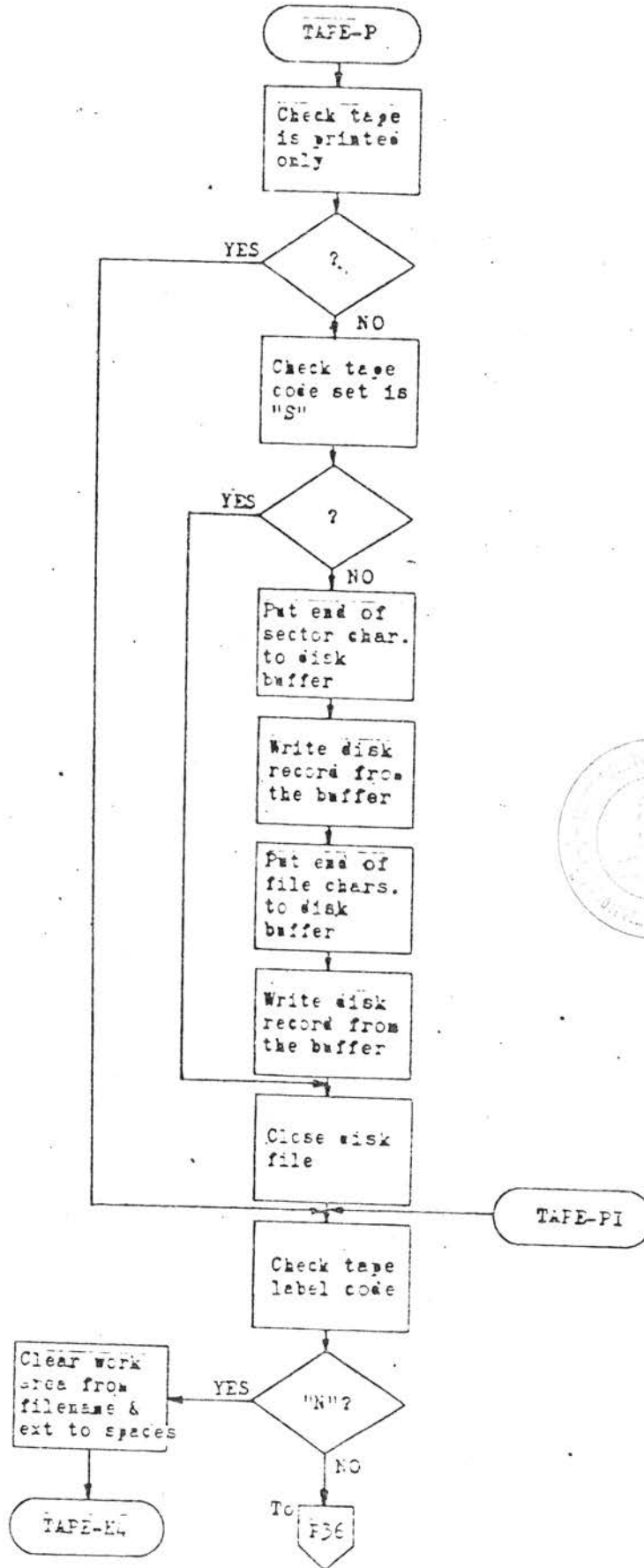
รูปที่ 3.6 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม TAPE/CMO



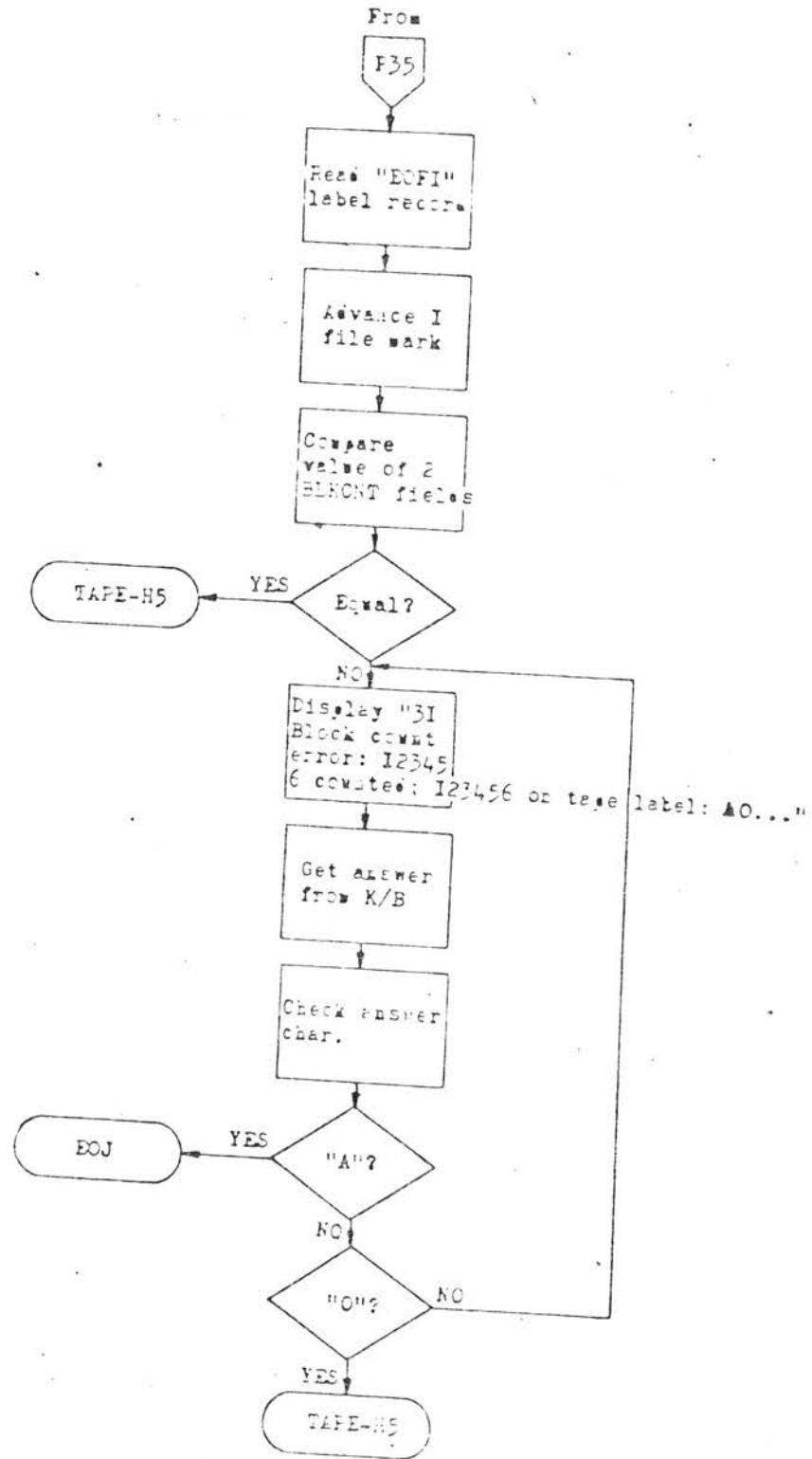
รูปที่ 3.6 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม TAPE/CMC



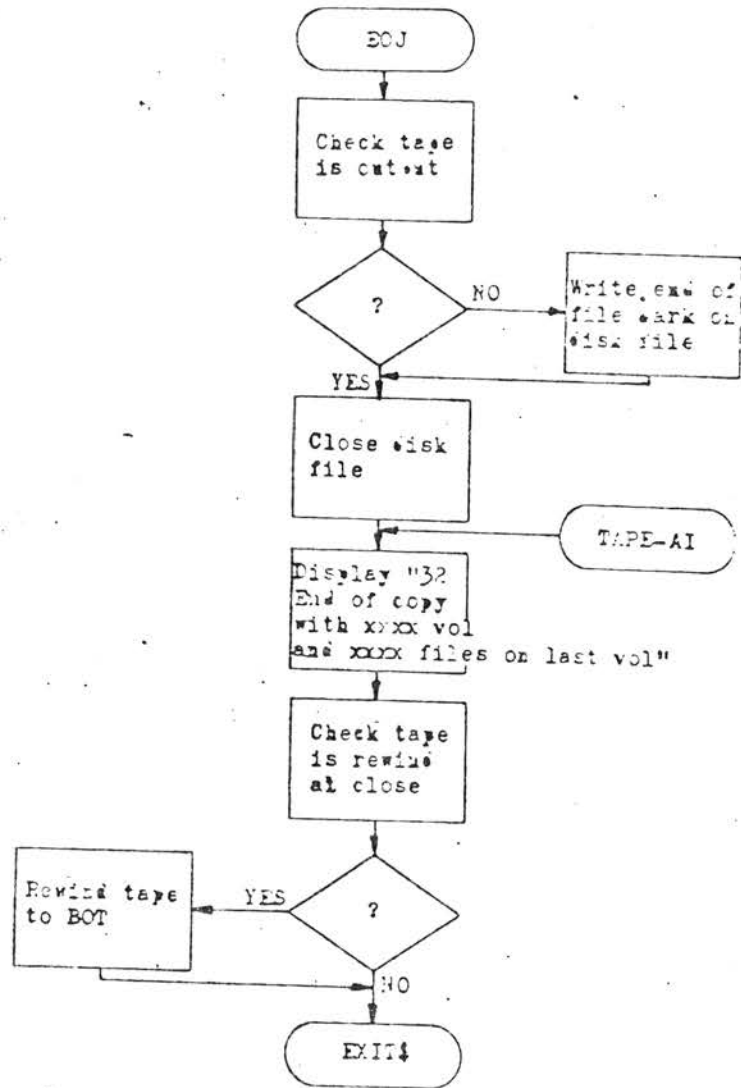
รูปที่ 3.6 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม TAPE/CMO



รูปที่ 3.6 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม TAPE/CMO



รูปที่ 3.6 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม TAPE/CMD



รูปที่ 3.6 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรม TAPE/CMD