หอสมุดกลาง สถาบันวิทยบริการ จุฬาลงกรณ์มหาวิทยาลัย

# บทที่ 5
# ส่วนควบคุมการไหลของข้อมูล

ส่วนควบคุมการไหลของข้อมูลนั้นทำหน้าที่ที่สำคัญ 3 อย่าง คือควบคุมให้การไหลผ่านของข้อมูลเป็นไปอย่างต่อเนื่อง ไม่มีการติดขัดของข้อมูลดิบที่มาจากเครื่องอ่านแผ่นซีดีรอมไปยังตัวประมวลผลหลัก, ควบคุมให้การไหลของข้อมูลที่ถอดรหัสแล้วที่ออกมาจากตัวประมวลผลหลักไปยังตัวแปลงสัญญาณเชิงเลขเป็นสัญญาณแอนะล็อกตามช่วงเวลาที่ถูกต้องขึ้นอยู่กับความเร็วในการสุ่มตัวอย่างของสัญญาณนั้นๆ และทำหน้าที่โหลดโปรแกรมและสั่งงานให้กับตัวประมวลผลหลักทำงานในตอนเริ่มต้น

## 5.1 โครงสร้างภายในของส่วนควบคุมการไหลของข้อมูล

ส่วนควบคุมการไหลของข้อมูลนี้เป็นชิปที่ออกแบบด้วย VHDL แล้วสังเคราะห์ลงบน FPGA โดยมีส่วนประกอบเพิ่มเติมคือ หน่วยความจำอ่านอย่างเดียว (Read Only Memory: ROM), หน่วยความจำเข้าถึงแบบสุ่ม (Random Access Memory: RAM) ส่วนควบคุมการไหลของข้อมูลมีโครงสร้างภายในแสดงดังรูปที่ 5.1

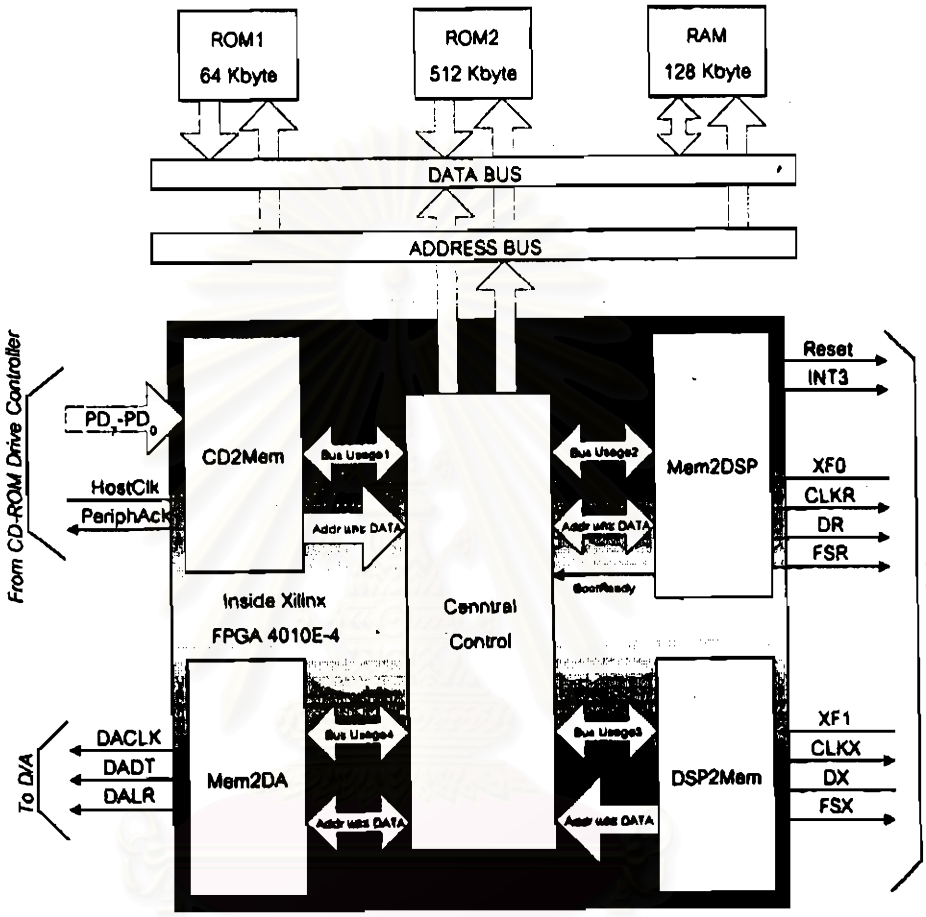
## 5.2 หน่วยความจำเข้าถึงแบบสุ่ม และหน่วยความจำอ่านอย่างเดียว

ส่วนควบคุมการไหลของข้อมูลใช้หน่วยความจำทั้งหน่วยความจำเข้าถึงแบบสุ่ม และหน่วยความจำอ่านอย่างเดียว สำหรับการเก็บข้อมูลโปรแกรมการทำงานของตัวประมวลผลหลัก. ข้อมูลการโปรแกรมชิป FPGA และเป็นที่พักข้อมูล (buffer) เพื่อให้การไหลของข้อมูลเป็นไปอย่างต่อเนื่อง ซึ่งมีรายละเอียดดังนี้

1. <u>ข้อมูลการโปรแกรมชิป FPGA</u> เนื่องจากชิป FPGA จำเป็นต้องมีการโปรแกรมการทำงานใหม่ทุกครั้งที่ใช้งาน ซึ่งชิป FPGA ที่ใช้เป็นของบริษัท Xilinx เบอร์ 4010E ตามคู่มือการใช้งาน ระบุไว้ว่าขนาดของข้อมูลที่ใช้โปรแกรมชิปนี้ คือ 178,136 บิต เท่ากับ 22,267 ไบต์ ซึ่งเก็บไว้ในหน่วยความจำอ่านอย่างเดียว

2. <u>ข้อมูลการโปรแกรมชิปประมวลผลสัญญาณเชิงเลข (DSP)</u> ข้อมูลส่วนนี้จะโหลดให้กับชิป DSP ตอนเริ่มแรกการทำงานผ่านทางพอร์ตอนุกรม เมื่อโหลดเสร็จสิ้นแล้ว ชิป DSP ก็จะเริ่ม

ถอดรหัส โดยขนาดของข้อมูลที่ใช้เก็บโปรแกรมชิป DSP นี้คือ 512 กิโลไบต์ ซึ่งเก็บไว้ในหน่วยความจำอ่านอย่างเดียวเช่นกัน



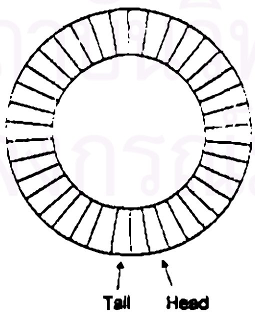

รูปที่ 5.1 โครงสร้างภายในของส่วนควบคุมการไหลของข้อมูล

3. ที่พักข้อมูล (buffer) จะใช้เป็นที่เก็บข้อมูลชั่วคราว โดยแบ่งเป็นสองส่วนคือ ส่วนที่ใช้เก็บข้อมูลดิบที่ได้รับจากเครื่องอ่านแผ่นซีดีรอมเพื่อส่งต่อให้กับชิปประมวลผลสัญญาณเชิงเลข ที่พักข้อมูลในส่วนนี้มีขนาด 64 กิโลไบต์ และส่วนที่รับข้อมูลที่ถอดรหัสเรียบร้อยแล้วจากชิปประมวลผลสัญญาณเชิงเลขเพื่อส่งต่อให้กับชิปแปลงสัญญาณเชิงเลขเป็นสัญญาณแอนะล็อก ก็มีขนาดของที่พักข้อมูล 64 กิโลไบต์เช่นเดียวกัน ในการออกแบบจะรวมที่พักข้อมูลทั้งสองส่วนไว้ในหน่วยความจำเข้าถึงแบบสุ่มตัวเดียวกันขนาด 128 กิโลไบต์

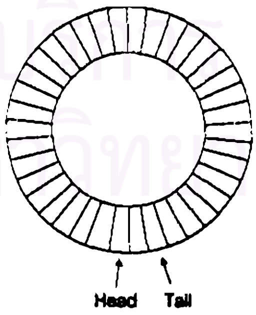
## 5.3 การจัดหน่วยความจำในที่พักข้อมูล

ที่พักข้อมูลจะใช้หน่วยความจำเข้าถึงแบบสุ่มขนาด 128 กิโลไบต์ โดยแบ่งออกเป็นสองส่วนคือส่วนที่ใช้เก็บข้อมูลดิบที่ได้รับจากเครื่องอ่านแผ่นซีดีรอมเพื่อส่งต่อให้กับชิปประมวลผลสัญญาณเชิงเลข (ส่วนที่ 1) และส่วนที่รับข้อมูลที่ถอดรหัสเรียบร้อยแล้วจากชิปประมวลผลสัญญาณเชิงเลขเพื่อส่งต่อให้กับชิปแปลงสัญญาณเชิงเลขเป็นสัญญาณแอนะล็อก(ส่วนที่ 2) ส่วนละ 64 กิโลไบต์ ส่วนที่ 1 จะอยู่ในช่วงแอดเดรส 00000h-0FFFFh และส่วนที่ 2 จะอยู่ในช่วงแอดเดรส 10000h-1FFFFh หน่วยความจำทั้งหมดถูกจัดเรียงแบบคิววงกลม (Circular queue) บล็อกที่ทำหน้าที่เขียนและอ่านข้อมูลในส่วนที่ 1 คือบล็อก CD2Mem และบล็อก Mem2DSP ตามลำดับ ในขณะที่บล็อกที่เขียนและอ่านข้อมูลในส่วนที่ 2 คือบล็อก DSP2Mem และ บล็อก Mem2DA ตามลำดับ

<u>1. ขั้นตอนการเขียนและอ่านข้อมูลในหน่วยความจำส่วนที่ 1</u> หน่วยความจำในส่วนนี้จะถูกเขียนโดยบล็อก CD2Mem ที่แอดเดรสที่ชี้โดย head ในขณะที่ถูกอ่านโดยบล็อก Mem2DSP ที่แอดเดรสที่ชี้โดย Tail หากการเขียนเข้าเร็วกว่าการอ่านออกจะทำให้บัฟเฟอร์เต็มได้ เราจะกำหนดว่าหาก Head+2=Tail จะหมายถึงบัฟเฟอร์เต็ม ดังแสดงในรูปที่ 5.2(ก) บล็อก Central Control ซึ่งทำหน้าที่ควบคุมการใช้บัสก็จะไม่อนุญาตให้มีการเขียนข้อมูลเพิ่มเติมอีก มิฉะนั้นก็จะไปทับกับข้อมูลเดิมที่ยังไม่มีการอ่านออกไปใช้ และในทำนองเดียวกัน หากการอ่านออกเร็วกว่าการเขียนเข้าจะทำให้บัฟเฟอร์ว่างได้เช่นกัน โดยเราจะกำหนดว่าหาก Tail+2=Head ดังแสดงในรูปที่ 5.2(ข) จะหมายถึงบัฟเฟอร์ว่าง บล็อก Central Control ก็จะไม่อนุญาตให้มีการอ่านข้อมูลออกไป



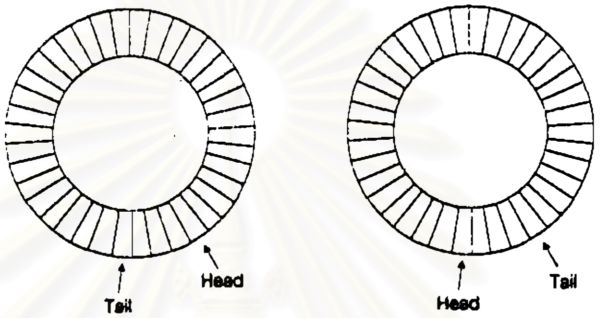

(ก)




(ข)

รูปที่ 5.2 เงื่อนไขที่การที่บัฟเฟอร์ส่วนที่หนึ่งเต็ม (ก) และว่าง (ข) ตามลำดับ

<u>2. ขั้นตอนการเขียนและอ่านข้อมูลในหน่วยความนำส่วนที่ 2</u> ก็จะมีวิธีการคล้ายกับขั้นตอนการเขียนและอ่านข้อมูลในหน่วยความจำส่วนที่หนึ่ง โดยที่หน่วยความจำในส่วนนี้จะถูกเขียนโดยบล็อก DSP2Mem และอ่านโดยบล็อก Mem2DA เนื่องจากในหน่วยความจำในส่วนที่ 2 นี้มีการอ่านและเขียนครั้งละ 4 ไบต์ ดังนั้นเงื่อนไขที่บัฟเฟอร์เต็มคือ Head+4=Tail ดังแสดงในรูปที่ 5.3 (ก) และ บัฟเฟอร์ว่างคือ Tail+4=Head ดังแสดงในรูปที่ 5.3 (ข)




(ก) (ข)

รูปที่ 5.3 เงื่อนไขที่การที่บัฟเฟอร์ส่วนที่สองเต็ม (ก) และว่าง (ข) ตามลำดับ

## 5.4 การทำงานของส่วนควบคุมการไหลของข้อมูล

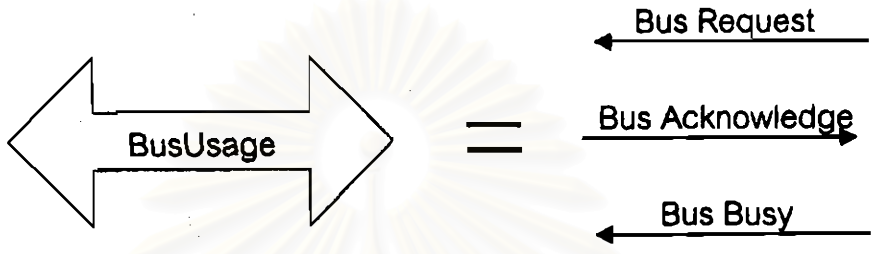
ส่วนควบคุมการไหลของข้อมูลจะแบ่งออกเป็นส่วนต่างๆ ทั้งสิ้น 5 ส่วนดังต่อไปนี้

<u>5.4.1 Central Control</u> จะทำหน้าที่เป็นตัวกลางในการควบคุมการเข้าใช้บัสข้อมูลและบัสแอดเดรสของ 4 บล็อกที่เหลือ โดยการอนุญาตให้บล็อกต่างเข้าใช้บัสได้นั้นจะขึ้นอยู่กับลำดับความสำคัญของบล็อกนั้น ซึ่งเรียงตามลำดับความสำคัญมากไปหาน้อยคือ Mem2DA, DSP2Mem, Mem2DSP และ CD2Mem นอกจากเงื่อนไขของลำดับความสำคัญแล้วยังมีเงื่อนไขเกี่ยวกับสถานะของบัฟเฟอร์ในขณะนั้นด้วย นั่นคือหากบัฟเฟอร์เต็มอยู่จะไม่อนุญาตให้บล็อกที่เขียนข้อมูลลงหน่วยความจำแบบสุ่มอันได้แก่ CD2Mem และ DSP2Mem ได้บัสไปใช้ หรือหากบัฟเฟอร์ว่างอยู่ก็จะไม่อนุญาตให้บล็อกที่อ่านข้อมูลจากหน่วยความจำเข้าถึงแบบสุ่มอันได้แก่ Mem2DA และ Mem2DSP ได้บัสไปใช้ สัญญาณในการขอและอนุญาตให้ใช้บัสคือสัญญาณ BusUsage ซึ่งมีทั้งสิ้น 4 ชุดสำหรับแต่ละบล็อก และในแต่ละชุดจะมีทั้งสิ้น 3 เส้นดังแสดงในรูปที่ 5.4 แผนภูมิสเตตการทำงาน (state diagram) ของบล็อกนี้แสดงดังรูปที่ 5.5

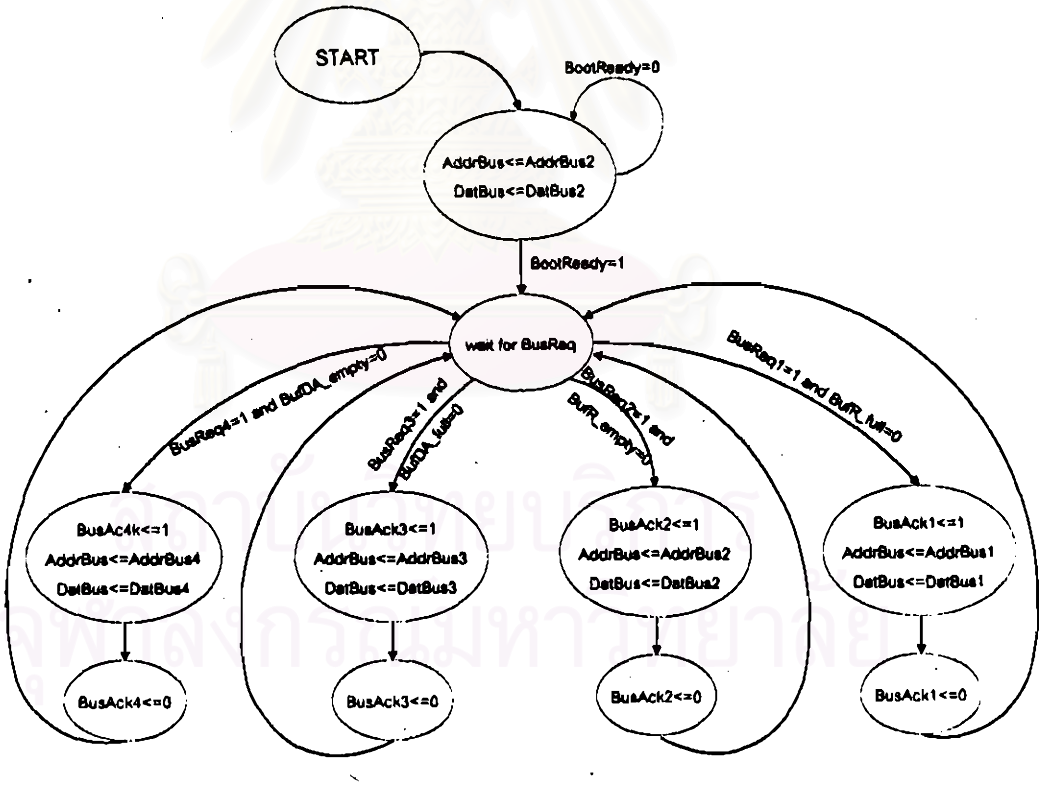
<u>5.4.2 CD2Mem</u> จะทำหน้าที่รับข้อมูลจาก CD-ROM Controller ครั้งละ 8 บิต โดยข้อมูลจะผ่านเข้ามาทางบัส $PD_7$-$PD_0$ และมีสัญญาณแฮนด์เชกคือ HostClk และ PeriphAck เมื่อรับข้อ

I ๑๘๙๓๓๓๓๖

มูลเรียบร้อยแล้วก็จะขอใช้บัสหน่วยความจำและจะรอจนกว่าจะได้รับอนุญาตให้เข้าใช้ได้จึงจะเขียนข้อมูลลงไป จากนั้นก็จะไปรอรับข้อมูลไบต์ถัดไปจาก CD-ROM Controller แผนภูมิสเตตการทำงานของบล็อกนี้แสดงในรูปที่ 5.6 โดยแอดเดรสที่จะเขียนข้อมูลเข้าไปเป็นแอดเดรสแรกเมื่อมีการกดปุ่มรีเซตคือแอดเดรส 0000h

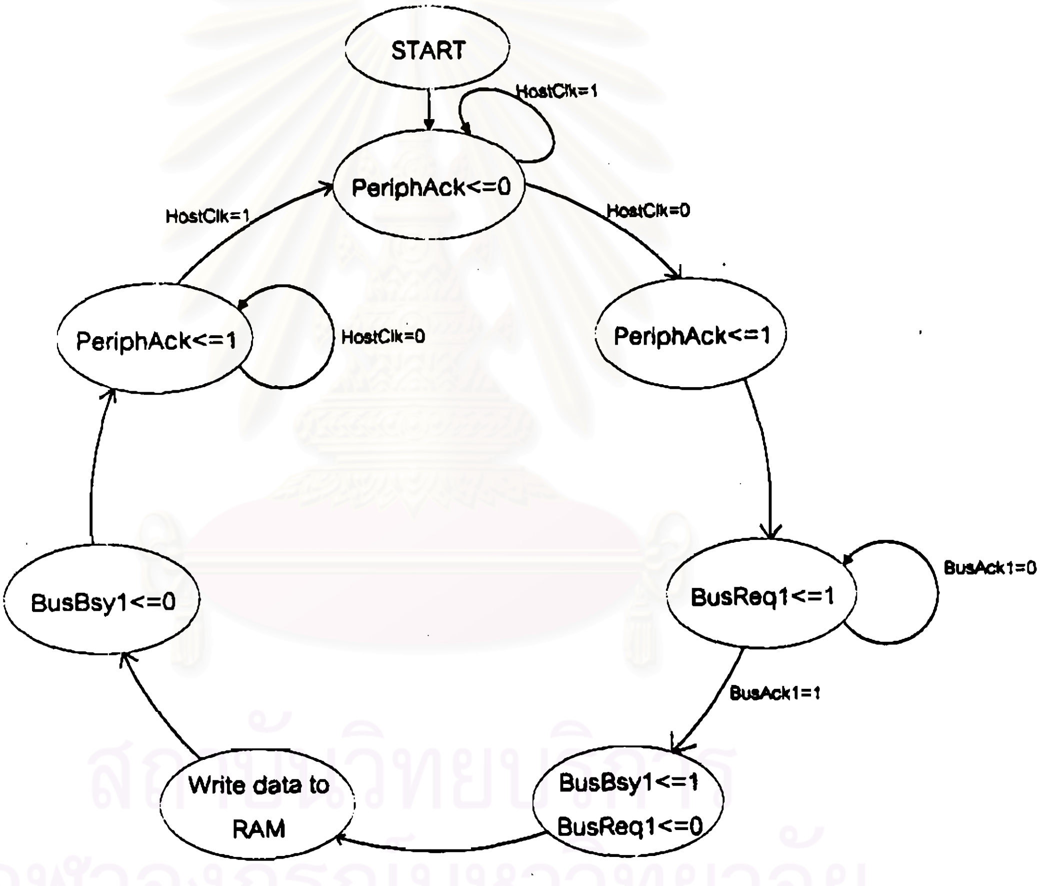



รูปที่ 5.4 สัญญาณภายในของบัส BusUsage




รูปที่ 5.5 แผนภูมิสเตตการทำงานของบล็อก Central Control

5.4.3 Mem2DSP จะทำหน้าที่สามอย่าง คือในตอนเริ่มต้นหรือกดปุ่มรีเซตบล็อกนี้จะส่งสัญญาณเพื่อบูตชิป DSP ด้วยโปรแกรมถอดรหัส MPEG-1 ลำดับชั้น 3 ที่เก็บอยู่ในหน่วยความจำอ่านอย่างเดียวตัวที่ 2 หลังจากที่บูตชิป DSP เรียบร้อยแล้วก็จะเริ่มต้นทำงานในหน้าที่ที่สอง หน้าที่ที่สองคือเคลียร์ข้อมูลในหน่วยความนำเข้าถึงแบบสุ่มทั้ง 128 กิโลไบต์ หลังจากนั้นก็จะเข้าสู่การทำงานในหน้าที่ที่สามจนกว่าจะมีการกดปุ่มรีเซตอีกครั้ง หน้าที่ที่หนึ่งและสองนี้จะทำเพียงครั้งเดียวตอนผู้ใช้กดปุ่มรีเซต ส่วนหน้าที่ที่สามคืออ่านข้อมูลจากหน่วยเข้าถึงแบบสุ่มส่วนที่ 1 เพื่อส่งต่อให้กับ DSP

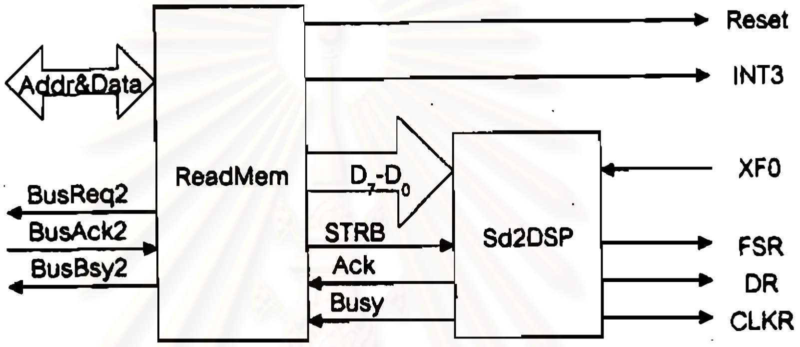



รูปที่ 5.6 แผนภูมิสเตตการทำงานของบล็อก CD2Mem

ภายในบล็อกนี้จะแบ่งเป็นบล็อกย่อย 2 บล็อก คือบล็อก ReadMem และ Sd2DSP ดังแสดงในรูปที่ 5.7 บล็อก ReadMem จะทำหน้าที่อ่านข้อมูลจากหน่วยความจำเข้าถึงแบบสุ่มส่วนที่หนึ่งเพื่อส่งต่อให้กับบล็อก Sd2DSP และบล็อก Sd2DSP ก็จะส่งต่อให้แก่ชิป DSP อีกต่อหนึ่ง

แผนภูมิสเตตการทำงานของบล็อกทั้งสองแสดงดังรูปที่ 5.8 และ รูปที่ 5.9 ตามลำดับ สำหรับการส่งข้อมูลในกับ DSP นั้นแสดงในรูปที่ 5.10

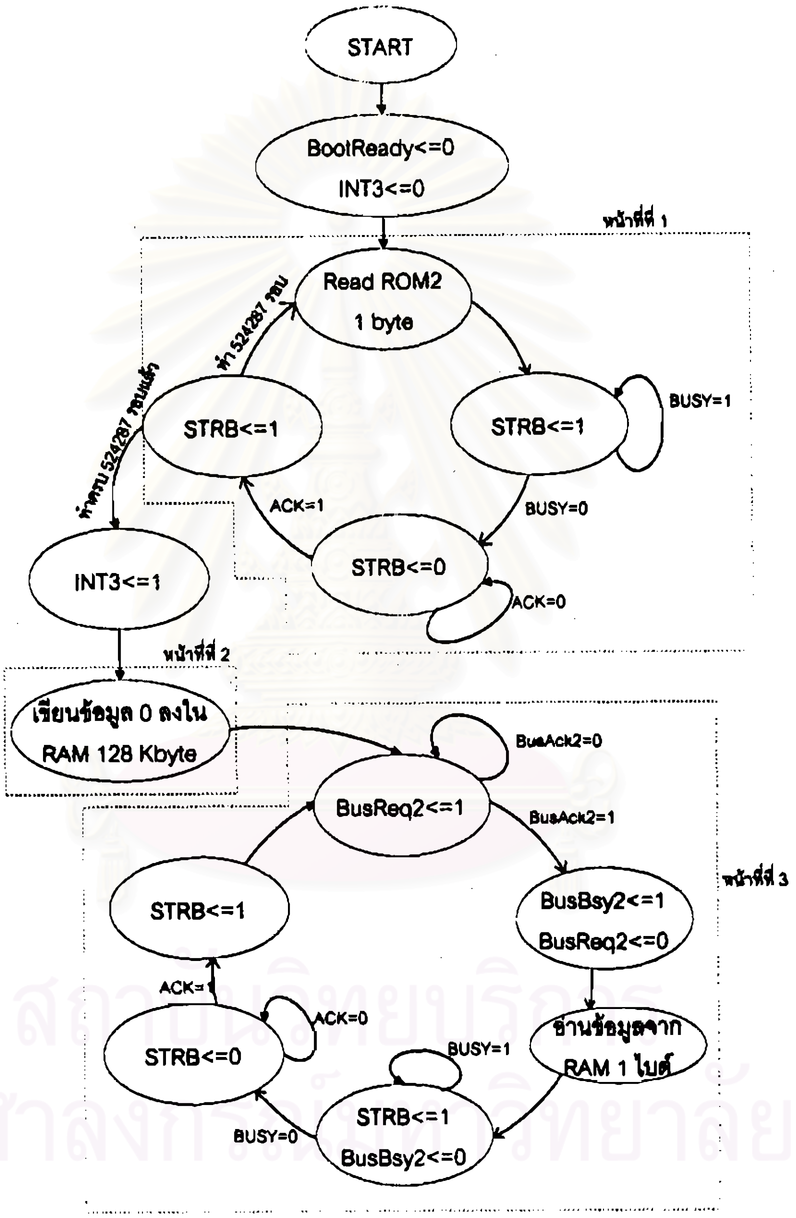
5.4.4. DSP2Mem ทำหน้าที่ในการรับข้อมูลที่ถอดรหัสเรียบร้อยแล้วขนาด 32 บิตจาก DSP แล้วเขียนลงหน่วยความจำเข้าถึงแบบสุ่มส่วนที่สอง ภายในบล็อกนี้จะถูกแบ่งเป็นบล็อกย่อยๆสองบล็อกดังแสดงในรูปที่ 5.11 คือบล็อก ReadSerial ทำหน้าที่รับข้อมูลจากชิป DSP แบบอนุกรมและบล็อก WriteMem ทำหน้าที่เขียนข้อมูลที่ได้รับจากชิป DSP ลงในหน่วยความจำเข้าถึงแบบสุ่มส่วนที่สอง แผนภูมิสเตตการทำงานของบล็อก ReadSerial กับแผนภาพสัญญาณการรับ



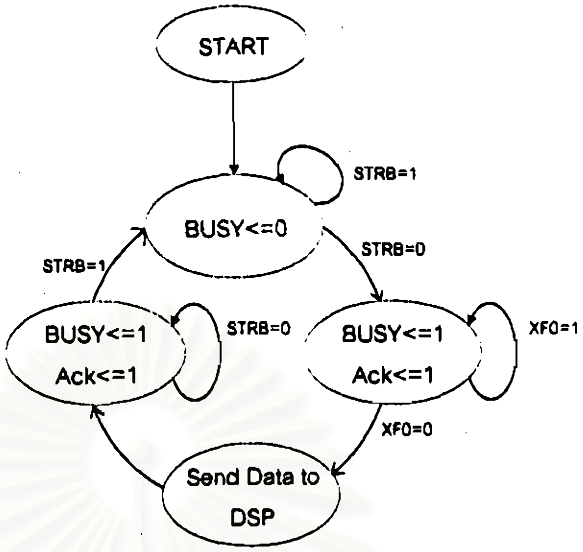

รูปที่ 5.7 โครงสร้างภายในของบล็อก Mem2DSP

5.4.5 Mem2DA ทำหน้าที่อ่านข้อมูลจากหน่วยความจำเข้าถึงแบบสุ่มส่วนที่สองออกมาทีครั้งละ 4 ไบต์ (32 บิต) เพื่อส่งให้กับชิปแปลงสัญญาณเชิงเลขเป็นสัญญาณแอนะล็อก (D/A) บล็อกนี้จะแบ่งออกเป็นบล็อกย่อยๆ สองบล็อกคือ ReadMem และ Sd2DA ดังแสดงในรูปที่ 5.15 แผนภาพรูปสัญญาณในการส่งข้อมูลให้แก่ชิป D/A แสดงดังรูปที่ 5.16 จะเห็นได้ว่าการส่งข้อมูล 1 ชุดจะใช้ DAClk ทั้งสิ้น 32 ลูก เนื่องจากบล็อกนี้ต้องส่งข้อมูลให้แก่ D/A ในช่วงเวลาที่กำหนดตายตัวตลอดเวลา เราจึงกำหนดให้บล็อกนี้มีความสำคัญสูงสุดในการขอเข้าใช้บัส และเพื่อให้การส่งข้อมูลให้แก่ D/A เป็นไปอย่างถูกต้องเราจำเป็นต้องมีข้อกำหนดระหว่างบล็อก ReadMem และ Sd2DA ว่า หากบล็อก ReadMem ไม่สามารถอ่านข้อมูลจากหน่วยความจำเข้าถึงแบบสุ่มส่วนที่สองได้ในช่วงเวลาที่กำหนดไม่ว่าด้วยเหตุผลใดก็ตาม บล็อก ReadMem จะส่งข้อมูลศูนย์จำนวน 32 บิตมาให้แทน เราจะแบ่งช่วงเวลาในการส่งข้อมูลหนึ่งชุด 32 บิตออกเป็น 32 ช่วงแต่ละช่วงห่างกันเท่ากับคาบเวลาของ DAClk โดยเราจะส่งสเตตจากบล็อก Sd2DA ไปให้กับบล็อก ReadMem ผ่านทาง iState ซึ่งมีจำนวน 5 บิต คือตั้งแต่ 0-31 บล็อก ReadMem จะเริ่มขอเข้าใช้บัสตั้งแต่ iState=2 หากไม่สามารถเข้าใช้บัสได้ภายใน iState=30 ก็จะยกเลิกการขอเข้าใช้บัสและวางข้อมูล

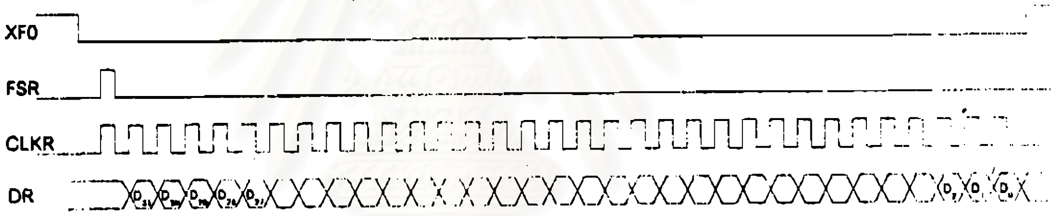
0 ลงบน Dout และรอที่จะเข้าใช้บัสที่ iState=2 ต่อไป หากการขอเข้าใช้บัสเป็นผลสำเร็จก็จะทำการอ่านข้อมูลจำนวน 4 ไบต์และวางข้อมูลนั้นลงบน Dout เพื่อให้บล็อก Sd2DA นำไปขยับ (shift) ออกให้กับ D/A ในรอบต่อไป แผนภูมิสเตตการทำงานของบล็อก ReadMem แสดงดังรูปที่ 5.17




รูปที่ 5.8 แผนภูมิสเตตการทำงานของบล็อก Mem2DSP

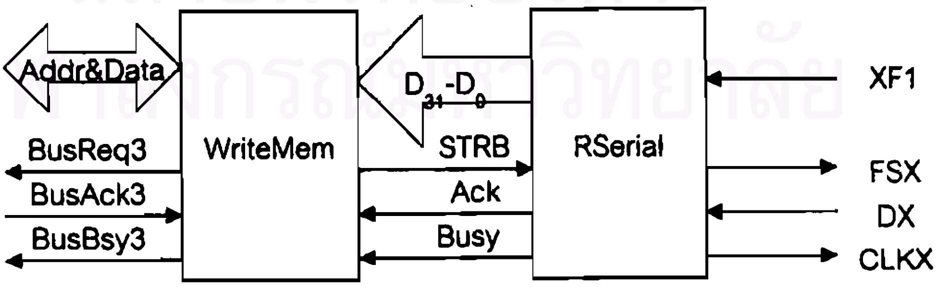



รูปที่ 5.9 แผนภูมิสเตตการทำงานของบล็อก Sd2DSP



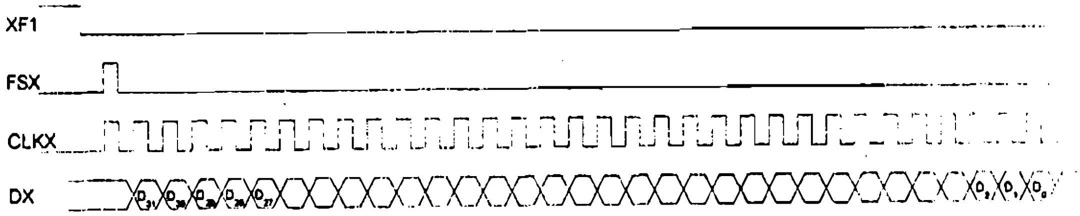

รูปที่ 5.10 แผนผังสัญญาณการเขียนข้อมูลเข้าทางพอร์ตอนุกรมของ DSP

ข้อมูลแบบอนุกรมจากชิป DSP แสดงในรูปที่ 5.12 และรูปที่ 5.13 ตามลำดับ รูปที่ 5.14 แสดงแผนภูมิสเตตของบล็อก WriteMem

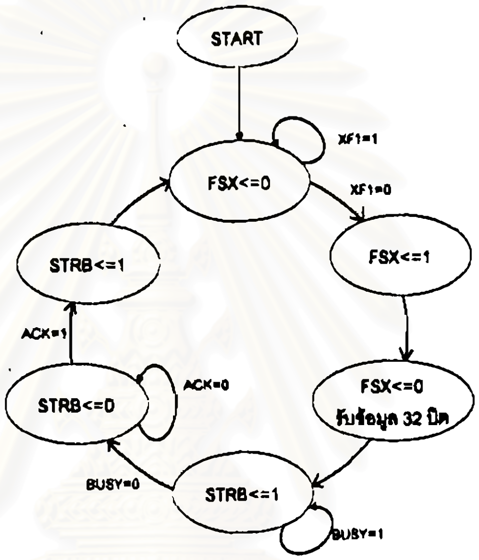



รูปที่ 5.11 โครงสร้างภายในของบล็อก DSP2Mem

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

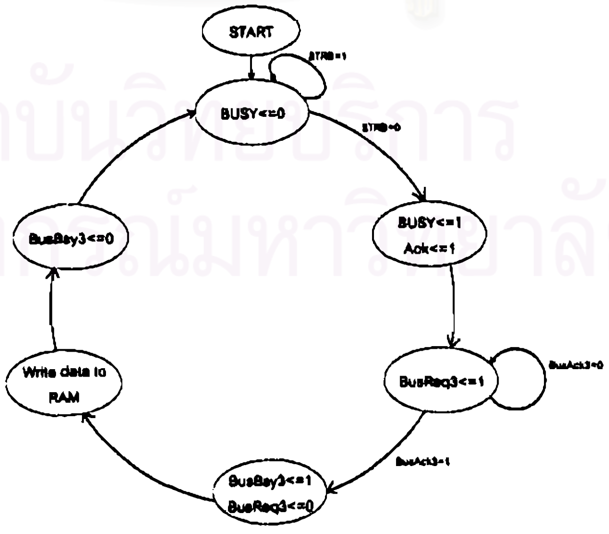



รูปที่ 5.12 แผนผังสัญญาณการอ่านข้อมูลทางพอร์ตอนุกรมของ DSP

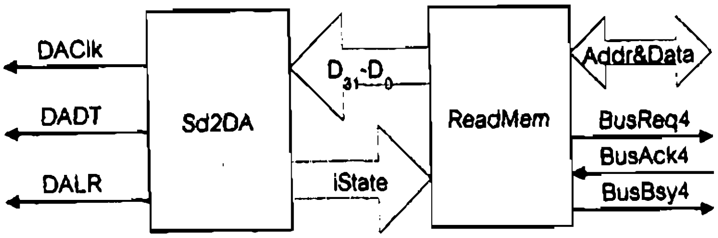



รูปที่ 5.13 แผนภูมิสเตตการทำงานของบล็อก Rserial

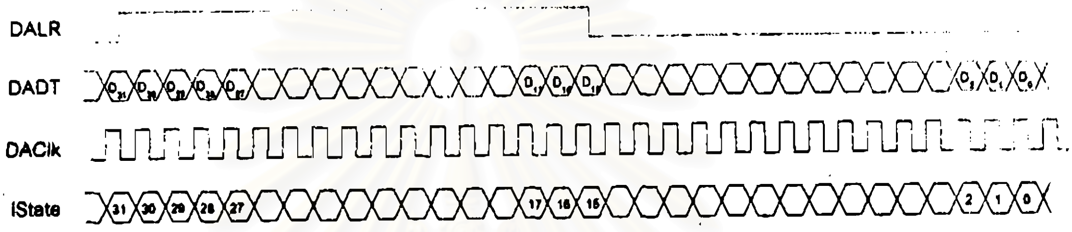
สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



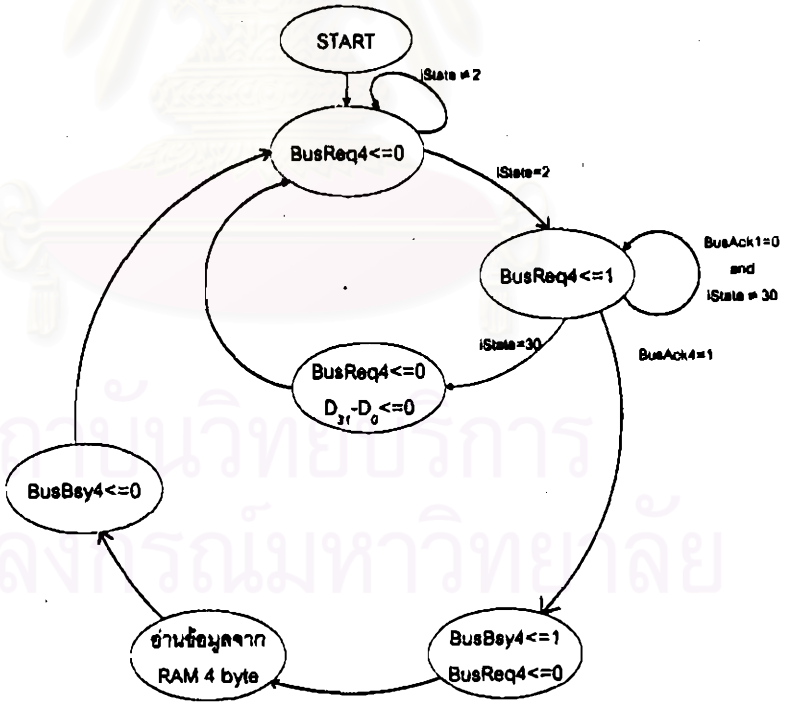

รูปที่ 5.14 แผนภูมิสเตตการทำงานของบล็อก WriteMem




รูปที่ 5.15 โครงสร้างภายในของบล็อก Mem2DA




รูปที่ 5.16 แผนผังสัญญาณที่ออกจากบล็อก Sd2DA




รูปที่ 5.17 แผนภูมิสเตตการทำงานของบล็อก ReadMem