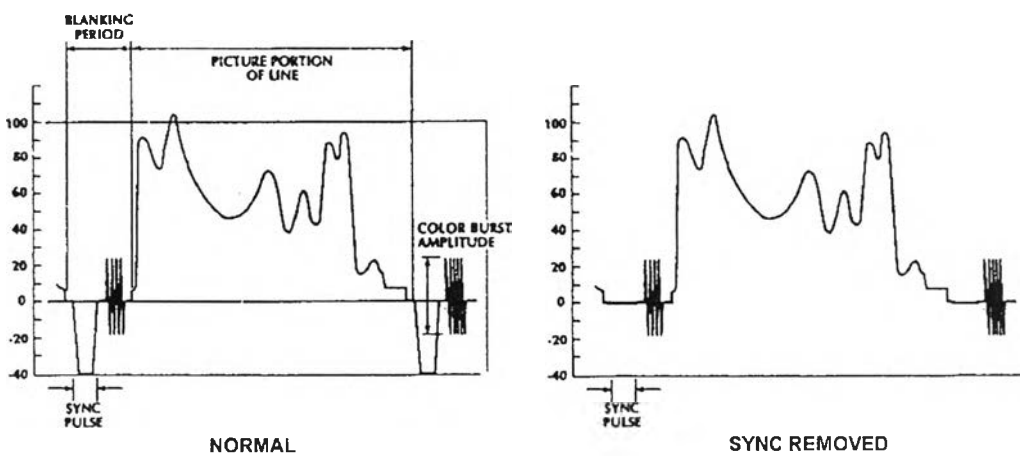
บทที่ 3
ข้อกำหนดของระบบสแครมเบิล

ระบบสแครมเบิลที่เสนอในวิทยานิพนธ์นี้ประกอบด้วย 2 ส่วนสำคัญคือ ส่วนที่หนึ่งเกี่ยวกับการสแครมเบิล คือการทำให้ภาพไม่สามารถดูได้ด้วยเครื่องรับโทรทัศน์ปกติในด้านส่ง และการทำให้ภาพนั้นกลับมาเป็นภาพปกติในด้านรับ และส่วนที่สองเกี่ยวกับการเข้าถึงอย่างมีเงื่อนไข คือการควบคุมวงจรดีสแครมเบิลในทางด้านรับว่าจะให้ทำงานหรือไม่ตามเงื่อนไขว่าสมาชิกได้ชำระเงินเพื่อรับชมรายการนี้หรือไม่ เป็นต้น

3.1 วิธีการสแครมเบิล

วิธีการสแครมเบิลที่ใช้ในวิทยานพนธ์นี้ประกอบด้วยวิธีการหลัก 2 วิธีคือการกลับสัญญาณภาพแบบสุ่มกับการตัดสัญญาณซิงโครไนซ์ และได้เสริมวิธีการเปลี่ยนระดับแรงดันในช่วงไร้ภาพทางแนวราบเพื่อเพิ่มคุณภาพการสแครมเบิล ดังมีรายละเอียดต่อไปนี้

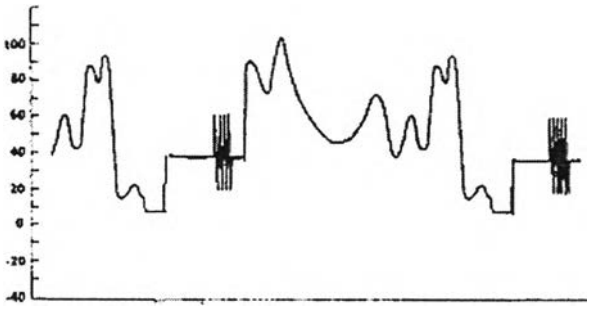
3.1.1 การตัดสัญญาณซิงโครไนซ์



รูปที่ 3.1 การตัดสัญญาณซิงโครไนซ์

การตัดสัญญาณซิงโครไนซ์ทำโดยการตัดพัลส์ที่ใช้ในการซิงโครไนซ์ออกทั้งสัญญาณซิงโครไนซ์ทางแนวตั้งและสัญญาณซิงโครไนซ์ทางแนวราบ จะได้ลักษณะของสัญญาณวีดิทัศน์ดังรูปที่ 3.1 ด้านขวามือ แต่การตัดสัญญาณซิงโครไนซ์ออกไปเพียงเท่านี้ยังให้คุณภาพการสแครมเบิลที่ไม่ดีพอ คือยังคงทำให้เครื่องรับโทรทัศน์สามารถกวาดเส้นภาพที่ตำแหน่งด้านซ้ายของจออยู่ได้บ้างเนื่องจากแรงดันไฟตรงในช่วงไร้ภาพทางแนวราบมีระดับตรงกับระดับสีดำซึ่งเป็นระดับต่ำสุดของสัญญาณภาพ จึงทำให้ช่วงไร้ภาพทางแนวราบนี้ยังอยู่ต่ำกว่าส่วนอื่นๆ โดยรวม เสมือนเกิดขอบขาลงที่ตอนต้นของช่วงไร้ภาพที่ถูกตัดซิงก์ออกไป เครื่องรับโทรทัศน์จะเข้าใจว่าตรงนี้เป็นสัญญาณซิงโครไนซ์ทางแนวราบ จึงเริ่มต้นกวาดเส้นภาพที่ตำแหน่งนี้ ภาพส่วนใหญ่จึงเริ่มต้นที่เดียวกันภาพที่ได้จึงดูเหมือนไม่ถูกดึงไปทางซ้ายหรือขวาตามที่ควรจะเป็น เพื่อแก้ปัญหานี้เราจึงยกระดับแรงดันไฟตรง

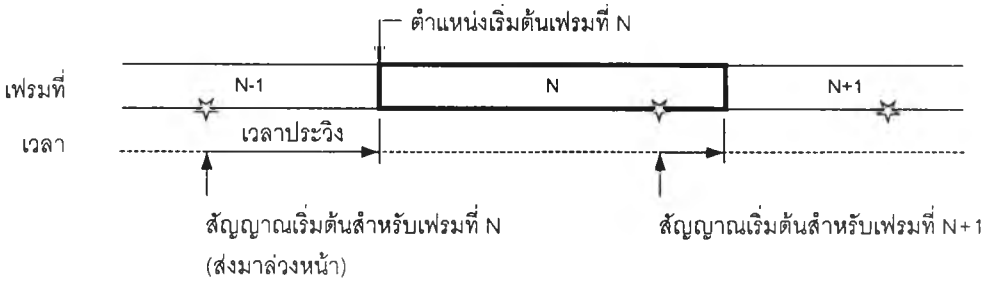
ในช่วงไร้ภาพทางแนวราบนี้ให้สูงขึ้น หลบเข้าไปอยู่ในระดับสีเทาดังรูปที่ 3.2 ภาพที่ได้ก็จะดูเหมือนถูกดึงมากขึ้นตามต้องการ ให้คุณภาพการสแครมเบิลที่ดีขึ้น



รูปที่ 3.2 สัญญาณวีดิทัศน์เมื่อตัดซิงก์และยกระดับแรงดันไฟตรงในช่วงไร้ภาพทางแนวราบ

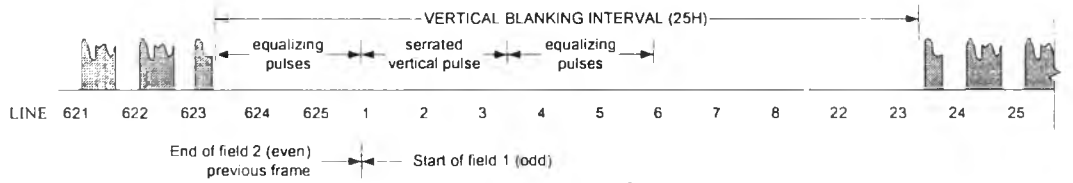
ผลของการตัดสัญญาณซิงโครไนซ์จะทำให้เครื่องรับโทรทัศน์ไม่สามารถจับตำแหน่งเริ่มต้นของเส้นภาพได้ ภาพที่แสดงบนจอจะมีตำแหน่งเริ่มต้นไม่ถูกต้องและไม่ตรงกันในแต่ละเส้นภาพ นอกจากนั้นยังทำให้เครื่องรับไม่สามารถจับตำแหน่งของเบิรสต์สีที่ถูกต้องได้ ทำให้ภาพที่แสดงไม่มีสีหรือมีสีที่ผิดเพี้ยนไป

การดีสแครมเบิลทำได้โดยการเติมสัญญาณซิงโครไนซ์กลับเข้าไปตามเดิม สำหรับตำแหน่งที่จะเติมสัญญาณซิงโครไนซ์นั้นเครื่องสแครมเบิลที่ต้นทางจะส่งมาให้โดยแทรกมากับสัญญาณภาพรวม แต่อย่างไรก็ตามถ้าส่งตำแหน่งเริ่มต้นเฟรมมาให้ตรงๆ จะถูกลักลอบดีสแครมเบิลได้โดยง่าย จึงส่งมาก่อนล่วงหน้าเป็นระยะเวลาค่าหนึ่งซึ่งจะเปลี่ยนแปลงไปตลอดเวลาดังรูปที่ 3.3



รูปที่ 3.3 การส่งสัญญาณเริ่มต้นเฟรมมาล่วงหน้า

เวลาประวิง (delay time) นี้ใช้ความถี่ 1 MHz ในการจับเวลา ดังนั้นในระบบ PAL ซึ่งใช้เวลา 40 mS ในการส่งภาพ 1 เฟรม จึงใช้ข้อมูลขนาด 16 บิตในการกำหนดเวลาประวิง

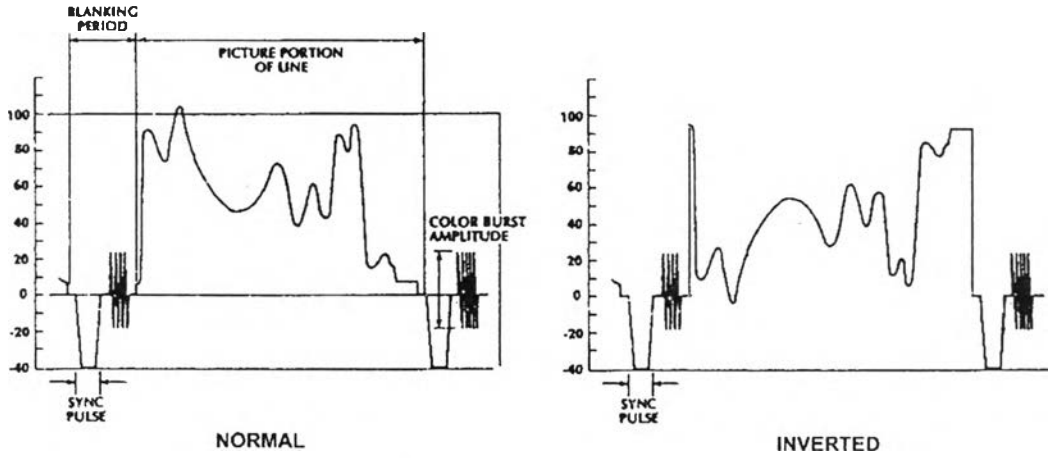


รูปที่ 3.4 สัญญาณวีดิทัศน์ในช่วงไร้ภาพทางแนวตั้งหลังจากตัดสัญญาณซิงโครไนซ์

สำหรับในช่วงไร้ภาพทางแนวตั้ง สัญญาณซิงโครไนซ์จะถูกตัดออกเช่นกัน ลักษณะรูปสัญญาณจะเป็นดังรูปที่ 3.4

3.1.2 การกลับสัญญาณภาพแบบสุ่ม

การกลับสัญญาณภาพคือการกลับระดับของสัญญาณภาพรอบระดับอ้างอิงระดับหนึ่งซึ่งกำหนดให้เป็นระดับกึ่งกลางระหว่างระดับภาพสีดำและระดับภาพสีขาว ผลที่ได้จะทำให้ภาพกลับจากดำเป็นขาวจากขาวเป็นดำ และยังทำให้เฟสของสัญญาณสีเลื่อนเฟสไป 180 องศาด้วย ลักษณะของสัญญาณภาพก่อนและหลังการกลับสัญญาณภาพแสดงดังรูปที่ 3.5



รูปที่ 3.5 การกลับสัญญาณภาพ

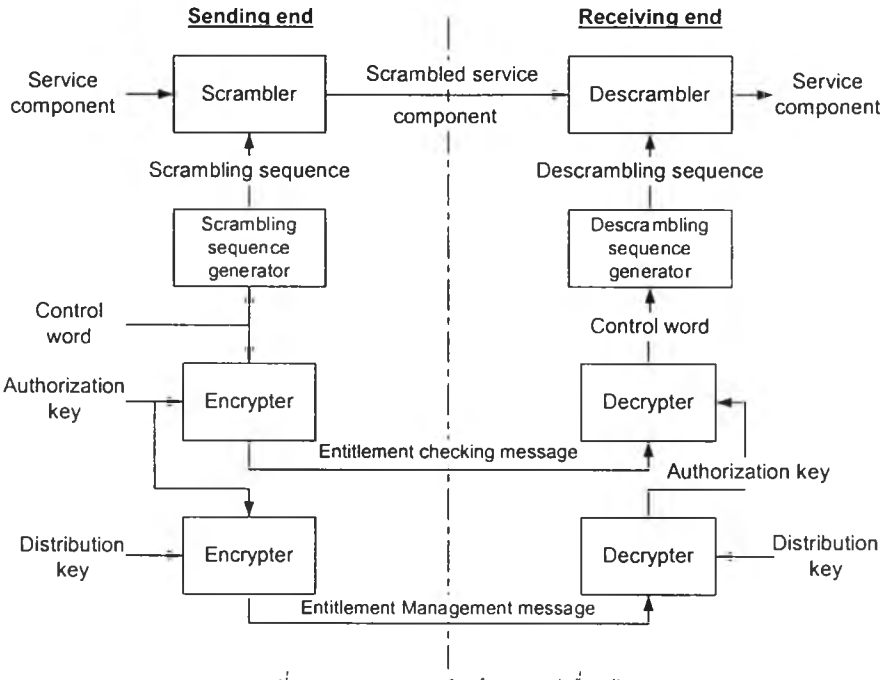
การดีสแครมเบิลทำได้โดยการกลับสัญญาณภาพอีกครั้งหนึ่ง จะได้สัญญาณภาพเดิมกลับคืนมา การสแครมเบิลวิธีนี้ใช้ข้อมูล 1 บิตในการบอกสถานะของสัญญาณภาพว่าเป็นปกติหรือถูกกลับ

3.2 การเข้าถึงอย่างมีเงื่อนไข

3.2.1 การทำงาน

การเข้าถึงแบบมีเงื่อนไขคือการควบคุมการสแครมเบิลให้เป็นไปอย่างมีประสิทธิภาพ รวมถึงการส่งและรับกุญแจที่ใช้ในการสแครมเบิลเพื่อป้องกันไม่ให้มีการเข้าถึงกุญแจได้โดยง่าย แผนภาพของระบบการเข้าถึงแบบมีเงื่อนไขแสดงดังรูปที่ 3.6 ซึ่งเป็นไปตามคำแนะนำของ ITU [6] โดยได้มีการดัดแปลงบางส่วนเพื่อให้เหมาะสมกับระบบสแครมเบิลที่ได้ออกแบบไว้

การทำงานเริ่มจากทางด้านส่งเลือกตัวเลขจำนวนหนึ่งขึ้นมาเป็นคำควบคุม (control word : CW) ตัวกำเนิดลำดับสำหรับการสแครมเบิล (scrambling sequence generator) จะใช้ CW นี้กำเนิดลำดับเลขสุ่มขึ้นมาป้อนให้แก่วงจรสแครมเบิลทำการสแครมเบิลสัญญาณวีดีทัศน์รวมที่เข้ามาโดยมีพารามิเตอร์ของการสแครมเบิลเป็นไปตามเลขสุ่มที่ได้รับมา



รูปที่ 3.6 ระบบการเข้าถึงแบบมีเงื่อนไข

เพื่อให้การดีสแครมเบิลเป็นไปอย่างถูกต้อง ทางด้านส่งจะต้องส่ง CW นี้ให้กับทางด้านรับด้วย CW ตัวเดียวกันนี้จะถูกเข้ารหัสลับด้วยกุญแจ Authorization Key แล้วส่งไปกับกลุ่มข้อมูล (packet) ชื่อ ECM (Entitlement Checking Message) *(รายละเอียดของกลุ่มข้อมูลนี้ดูได้ในหัวข้อ 3.2.2 โครงสร้างของกลุ่มข้อมูล หน้า 21)* เนื่องจาก Authorization Key มีหลายอันจึงต้องส่งหมายเลขของ Authorization Key ที่ใช้ หรือเรียกว่า Authorization Pointer ไปกับกลุ่มข้อมูล ECM ด้วย

ทางด้านรับ เมื่อได้ Authorization Pointer และ CW ที่ถูกเข้ารหัสลับมาแล้ว จะนำ Authorization Pointer ไปเปิดตาราง Authorization Table เพื่อหา Authorization Key ที่คู่กันมาถอดรหัสให้ได้ CW กลับคืนมา ถ้าทางด้านรับไม่ได้รับอนุญาตให้รับชมรายการนี้ ทางด้านส่งจะไม่ส่งคู่ Authorization Pointer และ Authorization Key นี้มาให้ ทางด้านรับก็จะไม่มีคู่ Pointer และ Key นี้ ทำให้ไม่สามารถรับชมรายการนี้ได้

CW ที่ได้มาจะใช้เป็นตัวเลขเริ่มต้นให้กับตัวกำเนิดลำดับสุ่มเทียมในทางด้านรับ ซึ่งมีการทำงานเหมือนกับทางด้านส่ง ดังนั้นลำดับสุ่มเทียมที่กำเนิดทางด้านรับจะเหมือนกับด้านส่ง วงจรดีสแครมเบิลจึงใช้ลำดับสุ่มเทียมที่ได้นี้เพื่อดีสแครมเบิลสัญญาณวีดีทัศน์กลับคืนมาได้

การส่งคู่ Authorization Pointer และ Authorization Key ให้แก่ด้านรับเครื่องใด ด้านส่งจะเข้ารหัสลับข้อมูลเหล่านี้ด้วยกุญแจ Distribution Key ซึ่งจะมีอยู่เฉพาะที่เครื่องรับที่ต้องการส่งถึงนั้นเท่านั้น ข้อมูลที่เข้ารหัสลับแล้วจะส่งไปด้วยกลุ่มข้อมูล EMM (Entitlement Management Message) *(รายละเอียดของกลุ่มข้อมูลนี้ดูได้ในหัวข้อ 3.2.2 โครงสร้างของกลุ่มข้อมูล หน้า 21)* กลุ่มข้อมูลนี้จะส่งถึงเครื่องรับเฉพาะเครื่องโดยระบุหมายเลขเครื่อง (address) ของผู้รับ

Control Word (CW) คือกุญแจในการสแครมเบิลและดีสแครมเบิลโดยทำหน้าที่เป็นตัวเลขเริ่มต้นให้แก่ตัวกำเนิดลำดับของเลขสุ่มเทียม

Entitlement Checking Message (ECM) คือข่าวสารที่ใช้ส่ง CW ให้แก่เครื่องรับปลายทางโดยปลอดภัย ประกอบด้วย 2 ส่วนคือ

1. Authorization Pointer เป็นตัวชี้ว่าจะใช้ Authorization Key ตัวใดในการถอดรหัส CW ที่ส่งมา
2. Encrypted Control Word

Entitlement Management Message (EMM) คือข่าวสารที่ใช้ส่งตาราง Authorization Table ตารางประกอบด้วยคู่ของ Authorization Pointer และ Authorization Key ดังตัวอย่างในตารางที่ 3.1

ตารางที่ 3.1 ตาราง Authorization Table

Authorization Pointer	Authorization Key
01	C7 9E A1 26 58 EC 22 D1
2C	B8 A3 9C A8 8F D2 BC 39
...	.....

กุญแจทั้งหมดที่ใช้ในโครงงานมีรายละเอียดดังตารางที่ 3.2

ตารางที่ 3.2 ขนาดของกุญแจ

กุญแจ	จำนวนบิต
Control Word	64
Authorization Pointer	8
Authorization Key	64
Distribution Key	64
Subscriber Address	16

3.2.2 โครงสร้างของกลุ่มข้อมูล

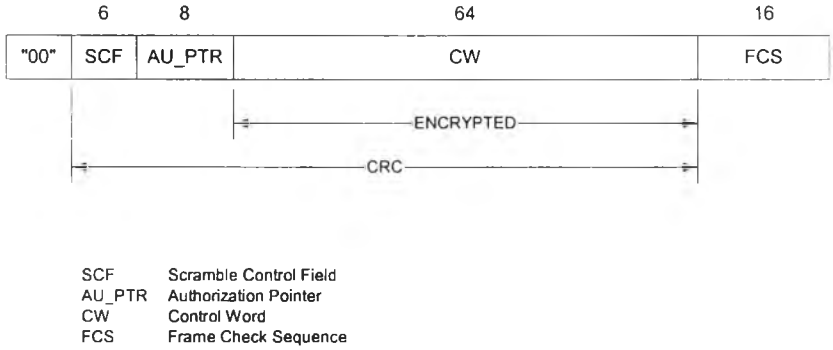
กลุ่มของข้อมูลแบ่งเป็น 2 ระดับคือ ระดับบน ประกอบด้วยกลุ่มข้อมูล ECM และ EMM กลุ่มข้อมูลทั้งสองกลุ่มนี้เวลาส่งจริงจะถูกแบ่งเป็นท่อนๆ แล้วส่งไปด้วยกลุ่มข้อมูลระดับล่าง ที่ชื่อ Data Packet อีกทีหนึ่ง ส่วนประกอบของกลุ่มข้อมูลทั้งสองระดับแสดงดังต่อไปนี้

3.2.2.1 ระดับบน

กลุ่มข้อมูลระดับนี้ออกแบบเพื่อให้ส่งข้อมูลการเข้าถึงอย่างมีเงื่อนไขทั้ง 2 ประเภทคือ ECM และ EMM กลุ่มข้อมูลระดับนี้จัดอยู่ในระดับ 3-4 ของโมเดลโอเอสไอ (Layer 3-4 of Open System Interconnection Reference Model) ทำหน้าที่กำหนดผู้รับปลายทาง ให้ความเชื่อถือได้ของข้อมูลที่ส่ง ควบคุมการไหลของข้อมูล และรวบรวมข้อมูลที่ถูกแบ่งเป็นส่วนย่อยๆ ในระหว่างการส่งกลับมาเป็นกลุ่มข้อมูลตามเดิม

ในระดับนี้ประกอบด้วยกลุ่มข้อมูล 2 ชนิด คือ

1) ECM



รูปที่ 3.7 โครงสร้างของกลุ่มข้อมูล ECM

กลุ่มข้อมูลชนิดนี้ใช้ส่ง CW เป็นหลัก โครงสร้างข้อมูลแสดงในรูปที่ 3.7 ในแต่ละฟิลด์กำหนดให้บิตที่มีนัยสำคัญสูงอยู่ทางซ้าย กลุ่มข้อมูลประกอบด้วย

- "00" คือ รหัสบอกชนิดของกลุ่มข้อมูลว่าเป็น ECM
- SCF หรือ Scramble Control Field ใช้บอกวิธีการสแครมเบิลที่ใช้อยู่ มีจำนวน 6 บิตสำหรับ 2 วิธีดังแสดงในตารางที่ 3.3 บิตใดเป็น "1" แสดงว่าวิธีการสแครมเบิลนั้นกำลังใช้งาน บิตใดเป็น "0" วิธีการสแครมเบิลนั้นไม่ได้ใช้งาน บิตที่ 1 ในตารางหมายถึงบิตที่มีนัยสำคัญน้อยสุดซึ่งอยู่ทางด้านขวา

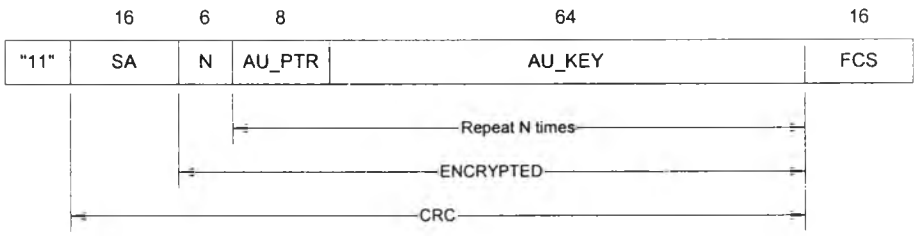
ตารางที่ 3.3 การกำหนดความหมายของแต่ละบิตใน SCF

บิตที่	ความหมาย
1	ตัดสัญญาณซิงโครไนซ์
2	กลับสัญญาณภาพ
3	ยังไม่กำหนด สงวนไว้ใช้ในอนาคต
4	ยังไม่กำหนด สงวนไว้ใช้ในอนาคต
5	ยังไม่กำหนด สงวนไว้ใช้ในอนาคต
6	ยังไม่กำหนด สงวนไว้ใช้ในอนาคต

- AU_PTR หรือ Authorization Pointer คือหมายเลขของกุญแจที่ใช้เข้ารหัสลับ CW ที่ส่งมา มีขนาด 8 บิต
- CW หรือ Control Word คือตัวเลขเริ่มต้นสำหรับส่งให้ตัวกำเนิดลำดับสุ่มเทียม มีขนาด 64 บิต ข้อมูลในส่วนนี้ได้รับการเข้ารหัสลับด้วยกุญแจ Authorization Key ที่สอดคล้องกับ Authorization Pointer ที่ส่งมาก่อนหน้า
- FCS หรือ Frame Check Sequence ใช้สำหรับตรวจสอบความถูกต้องของข้อมูล SCF, AU_PRT และ CW ที่เข้ารหัสลับ มีขนาด 16 บิต ส่วนนี้ใส่เข้ามาไม่ได้มีจุดประสงค์เพื่อแก้ไขความผิดพลาด

ของข้อมูล การแก้ไขความผิดพลาดได้กระทำแล้วในกลุ่มข้อมูลระดับล่าง แต่มีไว้เพื่อเพิ่มความมั่นใจในความถูกต้องของข้อมูลที่จะนำไปใช้งาน ถ้าผลการตรวจสอบออกมาว่าข้อมูลมีความผิดพลาดจะไม่นำข้อมูลนี้มาใช้งาน การตรวจสอบใช้วิธี CRC-16 *(รายละเอียดให้ดูในหัวข้อ 3.2.5 การควบคุมและแก้ไขข้อผิดพลาดในการส่งข้อมูล หน้า 32)*

2) EMM



SA	Subscriber Address
N	Number of control word pair
AU_PTR	Authorization Pointer
AU_KEY	Authorization Key
FCS	Frame Check Sequence

รูปที่ 3.8 โครงสร้างของกลุ่มข้อมูล EMM

กลุ่มข้อมูลชนิดนี้ใช้ส่งตาราง Authorization Table ซึ่งประกอบด้วยคู่ของ Authorization Pointer และ Authorization Key โครงสร้างข้อมูลเปิดโอกาสให้ส่งได้ได้มากกว่า 1 คู่ในแต่ละครั้ง โดยส่งได้สูงสุด 64 คู่ แต่ในทางปฏิบัติจะไม่สามารถส่งได้ถึงเนื่องจากติดที่จะต้องส่ง ECM มาเป็นระยะๆ ทำให้ขนาดของ EMM มีได้จำกัด

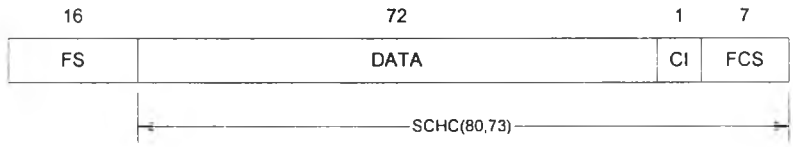
โครงสร้างของกลุ่มข้อมูลแสดงในรูปที่ 3.8 ประกอบด้วยส่วนต่างๆ ดังนี้

- "11" คือรหัสบอกชนิดของกลุ่มข้อมูลว่าเป็น EMM
- SA หรือ Subscriber Address คือหมายเลขเครื่องรับของสมาชิก มีขนาด 16 บิต
- N บอกจำนวนคู่ของ AU_PTR และ AU_KEY ที่จะส่งตามมา มีขนาด 6 บิต ส่งคู่ Authorization ได้ 64 คู่/ครั้ง
- AU_PTR หรือ Authorization Pointer คือหมายเลขที่ใช้อ้างถึงกุญแจ Authorization Key ที่ส่งตามมา มีขนาด 8 บิต จึงทำให้มีกุญแจได้ทั้งหมด 256 อัน
- AU_KEY หรือ Authorization Key คือกุญแจที่ใช้ถอดรหัส CW มีขนาด 64 บิต
- N, AU_PTR และ AU_KEY ทั้งหมด 6+N(8+64) บิต ได้รับการเข้ารหัสลับด้วยกุญแจ Distribution Key ซึ่งมีอยู่ที่เครื่องรับเท่านั้น

- FCS ใช้สำหรับตรวจสอบความถูกต้องของข้อมูล SA ที่ไม่ได้เข้ารหัสลับ และ N, AU_PRT, AU_KEY ที่เข้ารหัสลับ มีขนาด 16 บิต การตรวจสอบใช้วิธี CRC-16 *(รายละเอียดให้ดูในหัวข้อ 3.2.5 การควบคุมและแก้ไขข้อผิดพลาดในการส่งข้อมูล หน้า 32)*

3.2.2.2 <u>ระดับล่าง</u>

กลุ่มข้อมูลระดับล่างหรือกลุ่มข้อมูล Data Packet จัดอยู่ในระดับ 2 ของโมเดลโอเอสไอ มีหน้าที่ส่งข้อมูลแทรกไปในสัญญาณวีดิทัศน์โดยไม่ให้มีข้อผิดพลาด แต่เนื่องจากในกรณีนี้เป็นการส่งแบบกระจายทางเดียว (broadcast) จึงไม่สามารถที่จะร้องขอให้มีการส่งข้อมูลอีกครั้งได้ จึงทำได้เพียงตรวจสอบความผิดพลาดและแก้ไขเท่านั้น



FS Frame Sync
CI Continuity Index
DATA ECM packet or EMM packet
FCS Frame Check Sequence
SCHC Shortened Cyclic Hamming Code

รูปที่ 3.9 โครงสร้างของกลุ่มข้อมูล Data Packet

โครงสร้างของ Data Packet แสดงในรูปที่ 3.9 เป็นโครงสร้างที่มีขนาดแน่นอนคือ 96 บิต หรือ 12 ไบต์ โดยออกแบบให้มีขนาดที่สามารถแทรกได้พอดีกับส่วนที่เป็นสัญญาณภาพของสัญญาณวีดิทัศน์ เพื่อให้สามารถแทรกไปในช่วงไร้ภาพทางแนวตั้งได้โดยไม่รบกวนเบิรสต์สี

จุดประสงค์ของการออกแบบกลุ่มข้อมูลนี้เพื่อที่จะให้รองรับกลุ่มข้อมูลระดับบนซึ่งมีขนาดไม่แน่นอนให้สามารถส่งโดยแทรกไปกับสัญญาณวีดิทัศน์ได้โดยสะดวก ไม่รบกวนการทำงานของสัญญาณวีดิทัศน์ ส่วนประกอบของกลุ่มข้อมูลมีดังนี้

- FS หรือ Frame Sync คือกลุ่มข้อมูลจำนวน 16 บิตที่บิตที่มีนัยสำคัญสูงสุดมีค่าเป็น "1" และมีจำนวนบิตที่เป็น "0" และเป็น "1" ใกล้เคียงกัน *(รายละเอียดให้ดูในหัวข้อ 3.2.3 การแทรกข้อมูลในสัญญาณวีดิทัศน์ หน้า 25)* ใช้เพื่อบอกตำแหน่งเริ่มต้นของกลุ่มข้อมูล Data Packet ในเบื้องต้นนี้กำหนดให้ FS มีค่าเป็น "1001 1010 1001 1011" (0x9A9B)

- CI หรือ Continuity Index คือบิตบอกความต่อเนื่องของข้อมูล ในการส่งกลุ่มข้อมูลระดับบน 1 กลุ่มข้อมูลนั้นอาจต้องใช้กลุ่มข้อมูล Data Packet หลายกลุ่มข้อมูลต่อเนื่องกัน ถ้าบิตนี้เป็น "1" แสดงว่ายังไม่จบชุดของข้อมูล ถ้ากลุ่มข้อมูล Data Packet นี้เป็นกลุ่มข้อมูลสุดท้ายในชุด บิตนี้จะมีค่าเป็น "0"

- DATA คือส่วนที่เป็นสัมภาระของกลุ่มข้อมูล กลุ่มข้อมูลระดับบนจะถูกแบ่งเป็นส่วนย่อย ส่วนละ 72 บิต แล้วนำมาบรรจุลงในส่วนนี้ ในกรณีที่ข้อมูลบรรจุไม่เต็ม 72 บิต ให้ใส่ข้อมูล 0x40 ให้ลงตัวในไบต์ต่อๆ มาจนเต็ม 9 ไบต์ (72 บิต) เพื่อให้ระบบยังคงรักษาสัญญาณนาฬิกาที่ใช้สำหรับการชักตัวอย่างได้อยู่

- FCS หรือ Frame Check Sequence คือส่วนตรวจสอบและแก้ไขความผิดพลาดของข้อมูลในส่วน CI และ DATA การตรวจสอบใช้รหัสวงเวียนแฮมมิ่งลดขนาด SCHC(80,73) สามารถตรวจพบและแก้ไขบิตผิดพลาดไม่เกิน 1 บิต *(รายละเอียดให้ดูในหัวข้อ 3.2.5 การควบคุมและแก้ไขข้อผิดพลาดในการส่งข้อมูล หน้า 32)*

3.2.3 การแทรกข้อมูลในสัญญาณวีดิทัศน์

ข้อมูลที่จะแทรกลงในสัญญาณวีดิทัศน์มี 2 ประเภทคือ กลุ่มข้อมูล BVF ใช้สำหรับบอกตำแหน่งเริ่มต้นเฟรมซึ่งจะส่งมาล่วงหน้าดังที่กล่าวแล้วในหัวข้อ *3.1.1 การตัดสัญญาณซิงโครไนซ์ หน้า 17* และกลุ่มข้อมูล Data Packet สำหรับส่งข้อมูลการเข้าถึงอย่างมีเงื่อนไข

ลักษณะของข้อมูลที่แทรกเป็นแบบ NRZ บิต "0" อยู่ที่ระดับแรงดันสีดำของสัญญาณวีดิทัศน์ บิต "1" อยู่ที่ระดับแรงดัน 130% ของระดับแรงดันสีขาว ระดับแรงดันที่สูงกว่าปกตินี้เพื่อให้เครื่องรับสามารถแยกความแตกต่างระหว่างข้อมูลและสัญญาณวีดิทัศน์ได้อย่างชัดเจน อัตราการส่งข้อมูลอยู่ที่ 2 MBit/s อัตราการส่งข้อมูลนี้อาจทำให้วงจรภาครับมีความซับซ้อนมากขึ้นแต่ก็จำเป็นเนื่องจากต้องการให้สามารถบรรจุกลุ่มข้อมูล BVF ลงในช่วงซิงก์ทางแนวราบได้ การส่งข้อมูลที่ความเร็วสูงขึ้นนี้ยังมีประโยชน์คือ สามารถลดระยะเวลาที่ใช้ส่งข้อมูลในกรณีที่ต้องส่งข้อมูลให้แก่กลุ่มสมาชิกจำนวนมาก

3.2.3.1 <u>การส่งกลุ่มข้อมูล</u> BVF

โครงสร้างของกลุ่มข้อมูล BVF ประกอบด้วยกลุ่มของบิตจำนวน 8 บิต ใช้สำหรับเป็นสัญญาณบอกให้เครื่องรับทราบตำแหน่งเริ่มต้นของเฟรม เงื่อนไขการกำหนดรหัสของกลุ่มข้อมูล BVF คือ บิตที่มีนัยสำคัญสูงสุดมีค่าเป็น "1" และอีก 7 บิตที่เหลือไม่ควรเป็น "0" หรือ "1" ทั้งหมด เนื่องจากอาจเกิดความผิดพลาดทำให้คิดว่าสัญญาณรบกวนเป็นกลุ่มข้อมูล BVF ได้ รหัสในส่วน 7 บิตนี้จะเปลี่ยนแปลงทุกครั้งเมื่อได้รับ CW ใหม่ โดยรหัสจะได้จากลำดับสุ่มเทียม *(ดูรายละเอียดในหัวข้อ 3.2.4.2 การนำลำดับสุ่มเทียมไปเป็นพารามิเตอร์ของการสแครมเบิล หน้า 30)* ถ้าลำดับสุ่มเทียมสุ่มให้เป็น "0" หรือ "1" ทั้งหมด จะเปลี่ยนไปใช้รหัส "0010011" แทน

กลุ่มข้อมูล BVF ได้ถูกออกแบบให้มีขนาดเล็ก เพื่อให้สามารถแทรกได้ในบริเวณที่เป็นซิงก์ทางแนวราบเดิม ไม่เช่นนั้นจะแทรกได้เฉพาะในช่วงไร้ภาพทางแนวตั้งซึ่งเป็นส่วนน้อยของเฟรมคิดเป็นเพียง 6.72% เท่านั้น ทำให้ไม่เกิดการกระจายของการแทรก ลดความแข็งแรงของระบบ

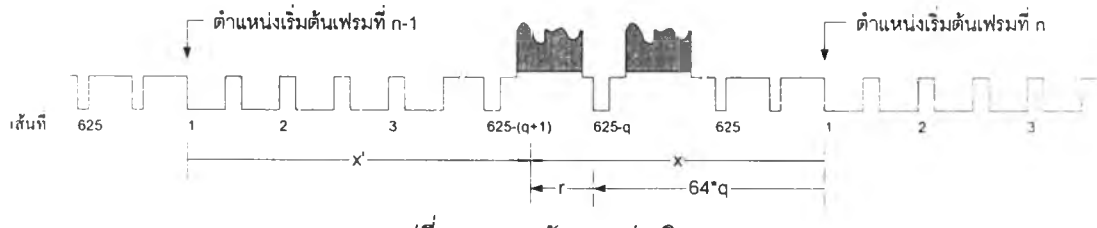
เนื่องจากในบางช่วงของเฟรม เช่น ส่วนที่เป็นเบิรสต์สี หรือส่วนที่เป็นสัญญาณภาพ ไม่สามารถจะแทรกกลุ่มข้อมูล BVF ได้ ดังนั้นถ้าตัวเลขสุ่มที่ได้มาตกอยู่ในช่วงดังกล่าวจะต้องดัดแปลงค่าที่ได้เล็กน้อย วิธีการดัด

แปลงแบ่งเป็น 2 ส่วนคือ ส่วนของเส้นภาพที่มีแต่เบิรสต์สี และส่วนของเส้นภาพที่มีทั้งเบิรสต์สีและสัญญาณภาพ ดังนี้

1) **<u>เส้นที่ 6-22 และ 319-335</u>**

เส้นภาพในช่วงนี้เป็นเส้นภาพที่อยู่ในช่วงไร้ภาพทางแนวตั้ง ไม่มีสัญญาณภาพแต่มีเบิรสต์สีส่งมาแล้ว จึงต้องหลีกเลี่ยง ไม่ส่งกลุ่มข้อมูลมาในช่วงที่มีเบิรสต์สี

ให้ x เป็นตัวเลขสุ่มที่ได้ x จะเป็นระยะเวลาจากบิตสุดท้ายของกลุ่มข้อมูล BVF ไปจนถึงตำแหน่งเริ่มต้นเฟรมจริง x เป็นเลขจำนวนเต็ม มีหน่วยเป็นไมโครวินาที (uS) เนื่องจากกำหนดให้ใช้สัญญาณนาฬิกาความถี่ 1 MHz ในการนับ การนับค่า x จะนับถอยหลัง โดยให้ 0 อยู่ที่ตำแหน่งเริ่มต้นเฟรมจริงดังรูปที่ 3.10



รูปที่ 3.10 การนับเวลาประวิง x

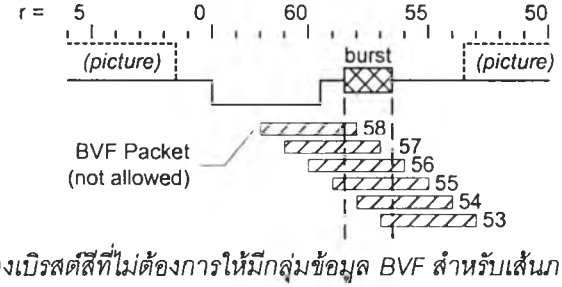
เนื่องจาก 1 เส้นภาพมีขนาด 64 uS จึงสามารถเขียน x ให้อยู่ในรูป

$$x = 64 \cdot q + r \tag{3.1}$$

จะได้ $$q = \left\lfloor \frac{x}{64} \right\rfloor \tag{3.2}$$

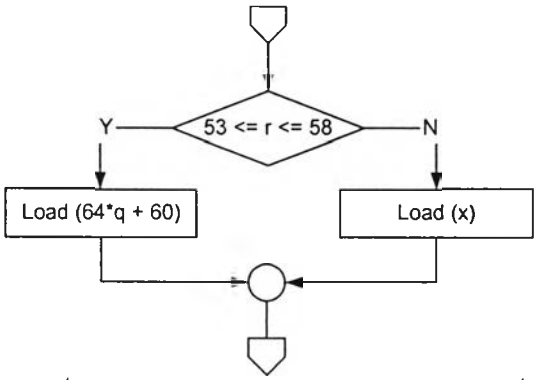
และ $$r = x \bmod 64 \tag{3.3}$$

โดยที่ q คือจำนวนเส้นภาพและ r คือเศษของเส้นภาพหน่วยเป็น uS



รูปที่ 3.11 ส่วนขยายของเบิรสต์สีที่ไม่ต้องการให้มีกลุ่มข้อมูล BVF สำหรับเส้นภาพที่ 6-22 และ 319-335

เส้นภาพเส้นที่ 6-22 และ 319-335 นี้ถ้าจัดให้อยู่ในรูปสมการที่ 3.1 ซึ่งเป็นการนับถอยหลัง จะได้ค่า q เป็น 619-603 และ 306-290 ตามลำดับ พิจารณารูปที่ 3.11 แสดงส่วนขยายของเบิรสต์สีที่ไม่ต้องการให้กลุ่มข้อมูล BVF ไปทับ จะเห็นว่ามีกลุ่มข้อมูล BVF ที่มีบิตสุดท้ายตกอยู่ในช่วงค่า r ตั้งแต่ 53 ถึง 58 เป็นช่วงที่ต้องดัดแปลงค่า โดยให้เปลี่ยนค่า r เป็น 60 ซึ่งสามารถแสดงให้เห็นเป็นผังงานในรูปที่ 3.12 คำสั่ง "Load" หมายถึงการป้อนค่าเวลาประวิงให้แก่หน่วยจับเวลา



รูปที่ 3.12 ผังงานแสดงเงื่อนไขการดัดแปลงค่าเวลาประวิงสำหรับเส้นที่ 6-22 และ 319-335

ในทางด้านส่ง การแทรกกลุ่มข้อมูล BVF ของเฟรมที่ n มาล่วงหน้า คือการนับระยะเวลา x' จากตำแหน่งเริ่มต้นเฟรมที่ n-1 (รูปที่ 3.10) โดยที่

$$x' = (40\ 000\ \text{uS}) - x$$

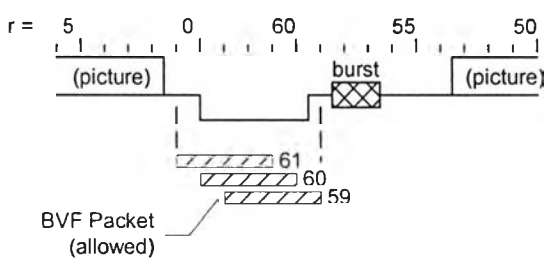
$$- (\text{ขนาดของกลุ่มข้อมูล BVF และออฟเซ็ตที่เกิดจากการทำงานของส่วนควบคุม}) \qquad (3.4)$$

2) <u>เส้นที่ 23-310 และ 336-623</u>

เส้นภาพในช่วงนี้เป็นเส้นภาพที่ปรากฏบนจอของเครื่องรับโทรทัศน์ มีทั้งเบิรสต์สีและสัญญาณภาพที่จะต้องหลีกเลี่ยงไม่ส่งกลุ่มข้อมูล BVF มาในช่วงนี้ จึงเหลือเพียงบริเวณซิงก์เท่านั้นที่สามารถแทรกกลุ่มข้อมูลได้ดังแสดงในรูปที่ 3.13 จะเห็นว่ามีค่า r เพียง 3 ค่าเท่านั้นที่แทรกได้คือ 59, 60 และ 61 จึงให้จัดการแทรกเสียใหม่โดยกำหนดให้

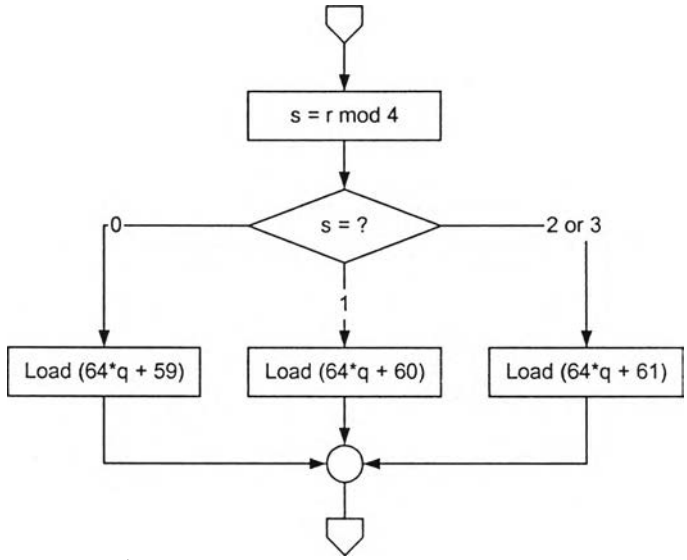
$$s = r \bmod 4 \qquad (3.5)$$

แล้วดัดแปลงค่า x ตามผังงานในรูปที่ 3.14



รูปที่ 3.13 ส่วนขยายในช่วงซิงก์แนวราบแสดงบริเวณที่สามารถแทรกกลุ่มข้อมูล BVF ได้

สำหรับเส้นภาพที่ 23-310 และ 336-623



รูปที่ 3.14 ผังงานแสดงเงื่อนไขการดัดแปลงค่าเวลาประวิงสำหรับเส้นที่ 23-310 และ 336-623

3.2.3.2 <u>การส่งกลุ่มข้อมูล Data Packet</u>

กลุ่มข้อมูล Data Packet มีขนาดคงที่ 96 บิต ที่อัตราการส่งข้อมูล 2 MBit/s จะใช้เวลา 48 uS ในการส่ง เนื่องจากสัญญาณซิงโครไนซ์ไม่มีแล้ว การแทรกกลุ่มข้อมูล Data Packet จึงสามารถแทรก ณ ตำแหน่งใดก็ได้ที่ไม่เกิดการซ้อนทับกับเบิรสต์สีและสัญญาณภาพ จากรูปที่ 2.1 และมาตรฐานการส่งสัญญาณวีดิทัศน์ระบบ PAL ที่กำหนดให้ส่งเบิรสต์สีในเส้นที่ 6-22 และ 319-335 ในช่วงไร้ภาพทางแนวตั้งด้วย [7,8] จึงสรุปช่วงที่สามารถแทรกกลุ่มข้อมูล Data Packet ได้ดังตารางที่ 3.4

ตารางที่ 3.4 ตำแหน่งที่สามารถแทรกกลุ่มข้อมูล Data Packet ได้

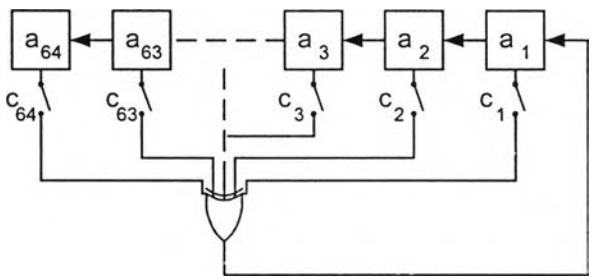
เส้นที่	วิธีการแทรก
1-5	แทรกได้ตลอดทั้งช่วง
6-22	แทรกได้ 1 เส้นต่อ 1 กลุ่มข้อมูล โดยหลีกเลี่ยงการแทรกทับตำแหน่งเบิรสต์สี
311-318	แทรกได้ตลอดทั้งช่วง
319-335	แทรกได้ 1 เส้นต่อ 1 กลุ่มข้อมูล โดยหลีกเลี่ยงการแทรกทับตำแหน่งเบิรสต์สี
623.5-625	แทรกได้ตลอดทั้งช่วง

จากตารางที่ 3.4 แถวสุดท้าย สังเกตว่าในเส้นที่ 623 จะมีเศษจุดห้าด้วย ทั้งนี้เนื่องจากในครึ่งหลังของเส้นนี้จะว่าง สามารถแทรกข้อมูลได้ ดูรูปที่ 3.4 กรณีที่มีภาพเพียงครึ่งเส้นนี้จะเกิดขึ้นอีกครั้งในเส้นที่ 23 แต่เนื่องจากในตอนต้นของเส้นที่ 23 นี้มีเบิรสต์สีอยู่ จึงไม่สามารถแทรกกลุ่มข้อมูล Data Packet ซึ่งมีขนาดใหญ่กว่าช่องว่างดังกล่าวได้ ช่วงที่แทรกได้ตามแถวที่สองของตารางที่ 3.4 จึงเป็น 6-22 แทนที่จะเป็น 6-23.5

3.2.4 ลำดับสุ่มเทียม

3.2.4.1 วิธีการกำเนิดลำดับสุ่มเทียม

วิธีการกำเนิดลำดับสุ่มเทียมมีด้วยกันหลายวิธี บางวิธีให้ผลทางสถิติที่ดี บางวิธีใช้การคำนวณที่ซับซ้อน บางวิธีให้วงรอบของลำดับขนาดใหญ่ก่อนที่จะกลับมาซ้ำเป็นวงรอบ (cycle) วิธีที่เลือกใช้เป็นวิธีที่คำนวณง่าย เพื่อความสะดวกในการใช้งาน และมีวงรอบที่ใหญ่โดยเพิ่มจำนวนบิตในรีจิสเตอร์ ค่าทางสถิติที่ได้เป็นสิ่งที่ไม่ต้องคำนึงถึงมากนัก สิ่งที่สนใจคือการกำเนิดลำดับให้มีความหลากหลายตาม CW ที่ได้รับมา จึงเลือกใช้วิธีรีจิสเตอร์เลื่อนแบบป้อนกลับเชิงเส้น (linear feedback shift register) ใช้รีจิสเตอร์ขนาด 64 บิตเพื่อเพิ่มขนาดวงรอบของการซ้ำ



รูปที่ 3.15 รีจิสเตอร์เลื่อนแบบป้อนกลับเชิงเส้น

รีจิสเตอร์เลื่อนแบบป้อนกลับเชิงเส้นประกอบด้วยรีจิสเตอร์ 64 ตัวดังรูปที่ 3.15 แต่ละตัวบรรจุข้อมูล 1 บิต เมื่อมีสัญญาณนาฬิกา ค่าในรีจิสเตอร์ถูกเลื่อนไปทางซ้าย ค่าในรีจิสเตอร์ตัวขวาสุด a_1 จะได้จากการคำนวณเอกซ์คลูซีฟออร์ (xor) ของรีจิสเตอร์อื่นๆ นั่นคือ a_1 ที่เวลาถัดไปสามารถหาได้จากสมการ 3.6

$$a_1(t+1) = \left[\sum_{i=1}^{64} c_i \cdot a_i(t) \right] \bmod 2 \qquad (3.6)$$

ค่าคงที่ C ขนาด 64 บิต ($c_1..c_{64}$) กำหนดลักษณะสมบัติของลำดับสุ่มเทียม และมีผลต่อขนาดของวงรอบการซ้ำ ซึ่งจะมีขนาดใหญ่ที่สุดเท่าที่เป็นไปได้คือ 2^{64}-1 เมื่อเลือกค่า C เป็นสัมประสิทธิ์ของพหุนามพริมิทีฟ (primitive polynomial) [9-11] พหุนามพริมิทีฟคือพหุนามลดทอนไม่ได้ (irreducible polynomial) p(X) ที่มีอันดับ m ที่สามารถหารพหุนาม X^n+1 ลงตัว โดยที่ n เป็นจำนวนเต็มบวกน้อยที่สุดที่ไม่ต่ำกว่า 2^m-1 นั่นคือถ้า n มีค่าน้อยกว่า 2^m-1 พหุนาม p(X) จะต้องหาร X^n+1 ไม่ลงตัว [12] ในที่นี้เลือก C มีค่าเป็น (80 00 00 00 00 00 00 0D)$_{16}$

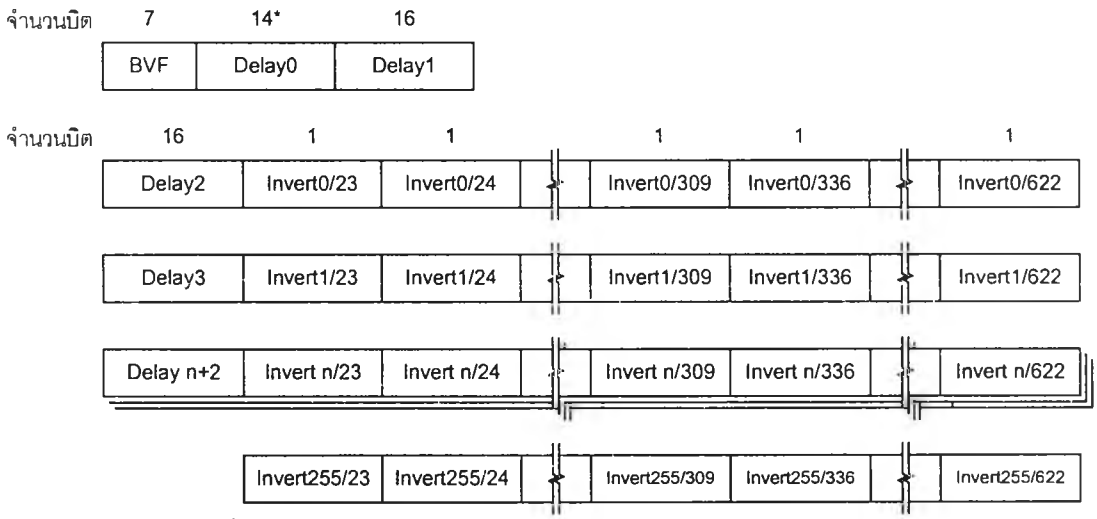
การคำนวณหาพหุนามพริมิทีฟอันดับ 64 ต้องใช้การคำนวณจำนวนมาก ต้องการเครื่องจักรคำนวณประสิทธิภาพสูงและใช้เวลานาน ผู้วิจัยจึงขอยกตัวอย่างโดยนำสัมประสิทธิ์ของพหุนามพริมิทีฟจาก [10] ซึ่งมีแสดงไว้บางค่ามาใช้

พหุนามพริมิทีฟอันดับ 64 มีเป็นจำนวนมาก ประกอบกับหมายเลขเริ่มต้นมีถึง 2^{64}-1 หรือประมาณ 1.8×10^{19} รูปแบบ ทำให้โอกาสที่จะตรวจพบมีได้น้อยมาก

วงจรที่แสดงในรูปที่ 3.15 เป็นการแสดงในทางฮาร์ดแวร์ การนำไปใช้งานจะแปลงเป็นโปรแกรมเพื่อให้ไมโครคอนโทรลเลอร์ทำหน้าที่กำเนิดลำดับสุ่มนี้

3.2.4.2 <u>การนำลำดับสุ่มเทียมไปเป็นพารามิเตอร์ของการสแครมเบิล</u>

ลำดับสุ่มเทียมที่ได้จากตัวกำเนิดจะถูกนำไปกำหนดพารามิเตอร์ของการสแครมเบิลได้แก่ กำหนดรหัสของกลุ่มข้อมูล BVF, กำหนดระยะเวลาประวิงจากกลุ่มข้อมูล BVF ถึงตำแหน่งเริ่มต้นเฟรมแต่ละเฟรม และกำหนดการกลับสัญญาณภาพของเส้นภาพแต่ละเส้น แต่ละบิตที่กำเนิดออกมาจะนำไปใช้กำหนดพารามิเตอร์ต่างๆ ดังแสดงในรูปที่ 3.16



รูปที่ 3.16 การนำลำดับสุ่มเทียมไปเป็นพารามิเตอร์ของการสแครมเบิล

เลขสุ่มที่ได้จำนวน 7 บิตแรกจะนำไปเป็นรหัสของกลุ่มข้อมูล BVF แต่จริงๆ แล้วกลุ่มข้อมูล BVF มีขนาด 8 บิต โดยบิตแรกกำหนดให้เป็น "1" และบิตต่อมาอีก 7 บิตคือเลขสุ่มที่ได้ แต่ถ้าเกิดกรณีที่ 7 บิตที่ได้เป็น "0" หรือ "1" ทั้งหมด ให้ใช้ "10010011" (0x93) เป็นรหัสกลุ่มข้อมูล BVF แทน

เลขสุ่มจำนวน 14 บิตต่อมาใช้กำหนดเวลาประวิงจากกลุ่มข้อมูล BVF ถึงตำแหน่งเริ่มต้นเฟรมที่ 0 ค่าประวิงเวลาของเฟรมที่ 0 นี้จะมีขนาดน้อยกว่าเฟรมอื่นๆ คือใช้แค่ 14 บิต เพื่อให้ค่าประวิงเวลามีค่ามากที่สุดไม่เกินเส้นที่ 336 จะได้ใช้เวลาในช่วงไร้ภาพทางแนวตั้งทั้งสองช่วงส่งกลุ่มข้อมูล ECM มาได้

เลขสุ่มจำนวน 16 บิตต่อมาใช้กำหนดเวลาประวิงสำหรับเฟรมที่ 1

ต่อจากนี้จะมองเลขสุ่มที่ได้เป็นกลุ่ม กลุ่มละ 590 บิต 16 บิตแรกใช้สำหรับกำหนดเวลาประวิงของเฟรมที่ 2 อีก 574 บิตที่เหลือใช้สำหรับกำหนดการกลับสัญญาณภาพของเส้นที่ 23-309 และเส้นที่ 336-622 ในเฟรมที่ 0 สังเกตว่า 16 บิตแรกจะใช้สำหรับกำหนดเวลาประวิงของเฟรมล่วงหน้าไป 2 เฟรม

สำหรับเฟรมสุดท้าย เฟรมที่ 255 จะลดเหลือกลุ่มละ 574 บิตสำหรับกำหนดการกลับสัญญาณภาพเท่านั้น ไม่กำเนิด 16 บิตสำหรับเวลาประวิงของเฟรมที่ 257 ซึ่งไม่มีและเพื่อความสะดวกในการจัดการเรื่องการซิงโครไนซ์ลำดับสุ่ม และในเฟรมก่อนเฟรมสุดท้าย เฟรมที่ 254 จะเห็นว่ามีการกำเนิดเลขสุ่ม 16 บิตสำหรับเวลา

ประวิงของเฟรมที่ 256 ซึ่งไม่มีเช่นกัน เวลาประวิงนี้มิได้นำไปใช้แต่กำเนิดขึ้นมาเพื่อความสะดวกในการจัดการเรื่องการซิงโครไนซ์ลำดับสุ่มเช่นกัน

ค่าเวลาประวิงที่สุ่มได้จะต้องอยู่ในขอบเขตบนและขอบเขตล่างที่กำหนด ขอบเขตล่างกำหนดให้เท่ากับ 128 เพื่อเผื่อเวลาให้คอนโทรลเลอร์ทำงานก่อนที่จะหมดเฟรม ถ้าค่าที่สุ่มได้ต่ำกว่า 128 ให้กำหนดให้บิตที่ 13 ของเลขที่สุ่มได้มีค่าเป็นหนึ่ง หรือกล่าวอีกนัยหนึ่งว่าให้บวกค่า 0x2000 เข้ากับค่าที่สุ่มได้

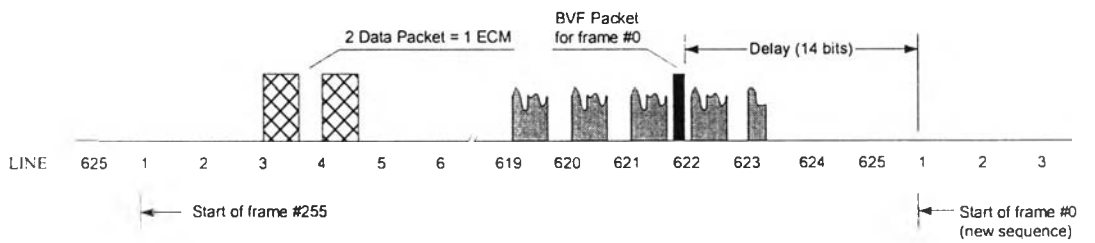
ค่าขอบเขตบนคิดจาก

ขอบเขตบน = 40 000 – ออฟเซ็ตจากการทำงานของระบบ – ค่าเผื่อเวลาทำงาน
= 40 000 – 36 – 128
= 39 836

ถ้าค่าที่สุ่มได้มีขนาดมากกว่าค่าขอบเขตบน ให้กำหนดให้บิตที่มีนัยสำคัญมากที่สุดของเลขที่สุ่มได้เป็นศูนย์ หรือกล่าวอีกนัยหนึ่งมีให้ลบค่า 0x8000 ออกจากค่าที่สุ่มมาได้ และให้บิตที่ 13 มีค่าเป็นหนึ่งถ้าหลังจากลบแล้วค่าที่ได้ต่ำกว่าขอบเขตล่าง

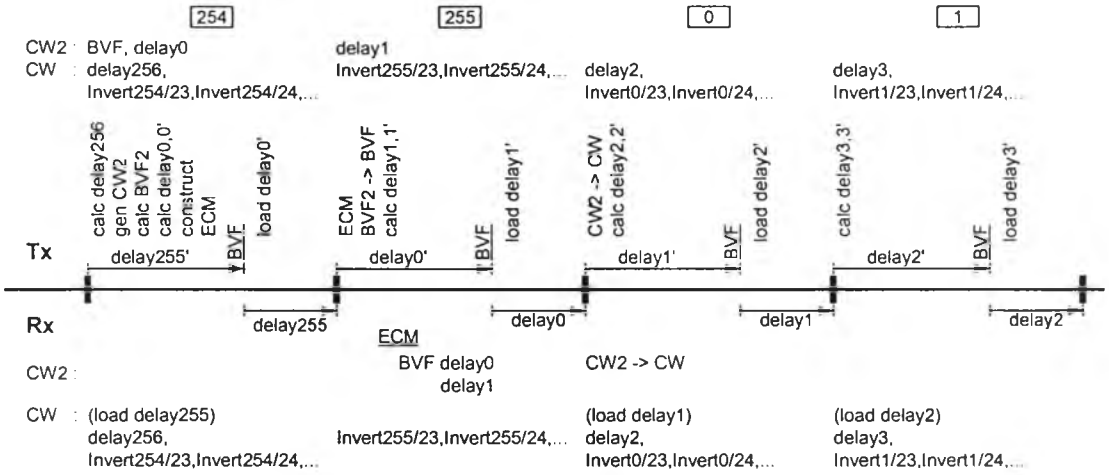
3.2.4.3 **<u>การซิงโครไนซ์ลำดับสุ่มเทียม</u>**

หลักการเบื้องต้นของการซิงโครไนซ์คือจะเกิดการเริ่มใหม่ของลำดับทุกครั้งที่ส่ง CW มาให้ในกลุ่มข้อมูล ECM และการเริ่มใหม่นี้กระทำทุกๆ 256 เฟรม รูปที่ 3.17 แสดงการซิงโครไนซ์ในช่วงเฟรมสุดท้ายของรอบการทำงาน เครื่องส่งจะส่งกลุ่มข้อมูล ECM มาในเฟรมที่ 255 ก่อนเส้นที่ 337 เครื่องรับเมื่อได้รับแล้วจะคำนวณหาเวลาประวิงสำหรับเฟรมที่ 0 และรอกลุ่มข้อมูล BVF เริ่มนับเวลาประวิงที่คำนวณไว้เพื่อจะได้ตำแหน่งเริ่มต้นจริงของเฟรมที่ 0



รูปที่ 3.17 การซิงโครไนซ์ลำดับสุ่มเทียม

ถ้าจะพิจารณาในรายละเอียดการทำงานที่มากขึ้นอีกระดับหนึ่ง แผนภาพการซิงโครไนซ์แบบละเอียดแสดงไว้ในรูปที่ 3.18 ในทางด้านส่ง การเตรียมการเริ่มต้นที่เฟรมที่ 254 เครื่องส่งจะกำเนิดตัวเลขเริ่มต้นสำหรับลำดับชุดถัดไป CW2 จากนั้นจะใช้ CW2 นี้กำเนิดเลขสุ่มสำหรับแบบรูปของกลุ่มข้อมูล BVF2 7 บิต กำเนิดเวลาประวิงสำหรับเฟรมที่ 0 (delay0), คำนวณส่วนเติมเต็ม (complement) delay0' จากนั้นสร้างกลุ่มข้อมูล ECM เตรียมไว้รอส่งในเฟรมที่ 255



รูปที่ 3.18 การซิงโครไนซ์ลำดับสุ่มเทียมแบบละเอียด

เมื่อถึงเฟรมที่ 255 จะส่งกลุ่มข้อมูล ECM, กำเนิดเวลาประวิง delay1 และคำนวณส่วนเติมเต็ม delay1'

เมื่อถึงเฟรมที่ 0 จะกำเนิดเวลาประวิง delay2 และคำนวณส่วนเติมเต็ม delay2' หลังจากนั้นจะเริ่มกำเนิดลำดับสุ่มสำหรับกำหนดการกลับสัญญาณภาพจำนวน 574 บิตสำหรับเฟรมที่ 0 และเป็นเช่นนี้เรื่อยไปจนถึงเฟรมที่ 254 ในเฟรมที่ 254 จะมีการกำเนิด delay256 ซึ่งไม่ได้นำมาใช้ และในเฟรมที่ 255 จะไม่มีการกำเนิดค่าประวิงเวลา delay257 จะกำเนิดแต่เลขสุ่มสำหรับกำหนดการกลับสัญญาณภาพ

ในช่วงรอยต่อเฟรมที่ 254-255 นี้ระบบจะต้องรองรับการกำเนิดลำดับสุ่มพร้อมๆ กัน 2 ลำดับ คือขณะที่มีการใช้ลำดับใหม่ที่เกิดจาก CW2 จะต้องรักษาลำดับเดิมที่เกิดจาก CW ต่อจนหมดถึงเฟรมที่ 255 จึงจะเปลี่ยนไปใช้ CW2 อย่างเดียวในเฟรมที่ 0

ในทางด้านรับ เมื่อได้รับ CW2 จากกลุ่มข้อมูล ECM ในเฟรมที่ 255 แล้วนำมากำเนิดแบบรูปของกลุ่มข้อมูล BVF เพื่อเตรียมรับแบบรูปใหม่ของกลุ่มข้อมูล BVF ที่กำลังจะมาถึง จากนั้นกำเนิดเวลาประวิง delay0 และ delay1 เมื่อถึงเฟรมที่ 0 จะนำค่า delay1 ไปใช้พร้อมกับกำเนิดค่าประวิงเวลา delay2 และบิตกำหนดการกลับสัญญาณภาพ เมื่อถึงจุดนี้จะเห็นได้ว่าทางด้านส่งและด้านรับใช้ลำดับสุ่มเทียมได้อย่างสอดประสานกัน

3.2.5 การควบคุมและแก้ไขข้อผิดพลาดในการส่งข้อมูล

การควบคุมและแก้ไขข้อผิดพลาดในการส่งข้อมูลนี้ ผู้วิจัยได้ศึกษาและได้นำเสนอไว้เป็นแนวทางเพื่อเพิ่มคุณภาพของการส่งข้อมูล แต่ยังมิได้มีการนำไปประยุกต์ใช้กับต้นแบบที่ได้สร้างขึ้นแต่อย่างใด

3.2.5.1 การตรวจสอบความถูกต้องด้วยรหัสซีอาร์ซี (CRC-16)

การตรวจสอบนี้ทำเพื่อสร้างความมั่นใจอีกชั้นหนึ่งว่าข้อมูลที่จะนำไปใช้นั้นถูกต้อง ถ้าตรวจสอบแล้วพบว่าไม่ถูกต้องจำเป็นที่จะต้องทิ้งข้อมูลนั้นไปเสีย การนำข้อมูลที่ผิดพลาดไปใช้อาจสร้างความเสียหายได้มากกว่าข้อมูลที่ทิ้งไปอาจส่งผลให้สมาชิกไม่สามารถรับชมรายการนั้นๆ ได้ เมื่อเกิดเหตุการณ์เช่นนี้ขึ้น สมาชิกจะติดต่อ

ศูนย์บริการและร้องเรียน ผู้ให้บริการเพียงแต่ส่งข้อมูลไปอีกครั้งหนึ่งเท่านั้น การตรวจสอบความถูกต้องด้วยวิธีซีอาร์ซีกระทำที่กลุ่มข้อมูลระดับบนคือกลุ่มข้อมูล ECM และกลุ่มข้อมูล EMM

พหุนามตัวกำเนิด (generator polynomial) ใช้แบบ CRC-16 ซึ่งเป็นที่นิยมใช้กันโดยทั่วไปมี $g(x)=1+x^2+x^{15}+x^{16}$ ซึ่งสามารถแยกตัวประกอบได้เป็น $g(x)=(1+x)(1+x+x^{15})$ ทำให้มีเลขจำนวนเต็ม m ที่น้อยที่สุดที่ g(x) สามารถหาร $(1+x^m)$ ลงตัวคือ 32,767 ทำให้พหุนาม CRC-16 สามารถตรวจพบบิตผิดพลาดจำนวน 1 บิต, 2 บิต, 3 บิต และบิตผิดพลาดเป็นจำนวนคี่จากคำรหัสขนาด n=32,767 (หรือคิดเฉพาะขนาดข้อมูล k=32,751) นอกจากนั้นยังสามารถตรวจพบบิตผิดพลาดแบบเบิรสต์ขนาด 16 บิตหรือน้อยกว่าได้ทั้งหมด, ตรวจพบ 99.997% ของบิตผิดพลาดแบบเบิรสต์ขนาด 17 บิต และตรวจพบ 99.998% ของบิตผิดพลาดแบบเบิรสต์ขนาด 18 บิตหรือมากกว่า [13]

3.2.5.2 <u>การตรวจสอบและแก้ไขข้อผิดพลาดด้วยรหัสวงเวียน</u>

การตรวจสอบและแก้ไขนี้ใช้กับกลุ่มข้อมูล Data Packet เพื่อรักษาความถูกต้องของข้อมูลไว้ในช่องสื่อสารที่มีความผิดพลาดไม่เกิน 1.25% ในการออกแบบ ต้องการขนาดของคำรหัสขนาด 80 บิตซึ่งเมื่อรวมกับ FS ขนาด 16 บิตแล้วจะได้กลุ่มข้อมูล Data Packet ขนาด 96 บิต สามารถส่งไปในช่วงสัญญาณภาพได้พอดี

จากรหัสวงเวียนแฮมมิ่ง (cyclic hamming code) [12] ให้ n เป็นขนาดของคำรหัส(หน่วยเป็นบิต), k เป็นขนาดของบิตข้อมูล และ m เป็นเลขจำนวนเต็ม เลือก n ที่มากกว่าหรือเท่ากับ 80

$$n = 2^m - 1 = 127$$

ได้ $m = 7$

จำนวนบิตข้อมูล $k = 2^m - m - 1 = 120$

ทำให้ได้รหัสวงเวียนแฮมมิ่ง (127,120) ซึ่งมี t=1 และ $d_{min}=3$ สามารถแก้ไขบิตผิดพลาดได้จำนวน 1 บิต เลือกพหุนามตัวกำเนิดเป็นพหุนามพริมิทีฟอันดับ m [12]

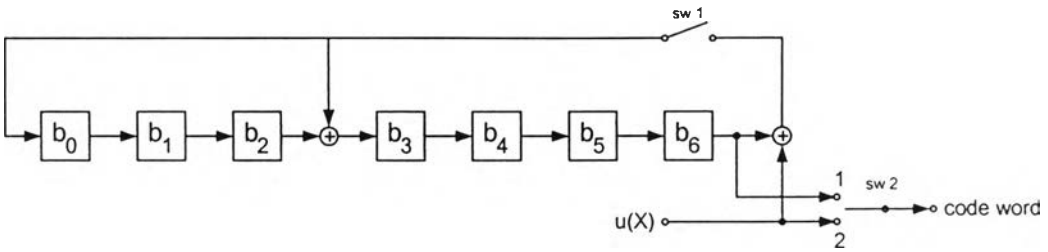
$$g(X) = 1 + X^3 + X^7$$

แต่ต้องการใช้คำรหัสขนาด 80 บิต จึงใช้วิธีการลดขนาดรหัสวงเวียน (shortened cyclic code) [12]

$$n - l = 80$$

$$l = n - 80 = 127 - 80 = 47$$

จึงได้รหัสวงเวียนแฮมมิ่งลดขนาด $(n-l, k-l) = (80,73)$



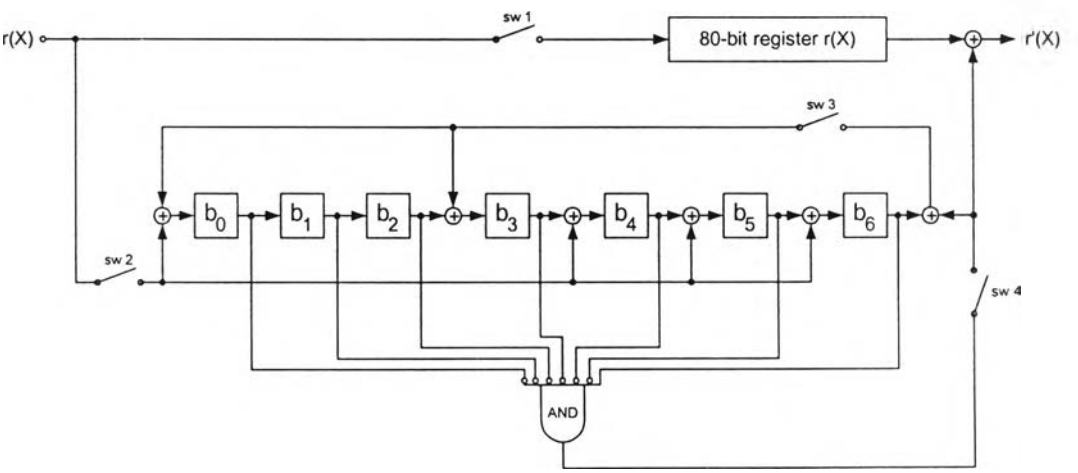
รูปที่ 3.19 วงจรสร้างรหัสวงเวียนแฮมมิ่งลดขนาด (80,73) ที่กำเนิดโดย g(X) = 1+X³+X⁷

รูปที่ 3.19 แสดงวงจรสร้างรหัสวงเวียนในรูปของฮาร์ดแวร์ ในการใช้งานจะแปลงให้เป็นโปรแกรมเพื่อให้ไมโครคอนโทรลเลอร์ทำหน้าที่นี้

การทำงานเริ่มที่กำหนดให้รีจิสเตอร์ทุกตัวมีค่าเป็น "0" สวิตช์ sw1 ปิดวงจร สวิตช์ sw2 ต่ออยู่ที่ตำแหน่ง 2 บิตข้อมูล u(X) จำนวน 73 บิตถูกเลื่อนเข้ามาโดยบิตที่มีนัยสำคัญมากเข้ามาก่อน ข้อมูลจะถูกส่งเข้าไปคำนวนเศษและพร้อมกันนั้นก็ส่งออกเป็นรหัสวงเวียนด้วย เมื่อครบ 73 บิตจะได้เศษจากการหารเก็บอยู่ใน $b_0..b_6$ ขั้นตอนต่อไปเป็นการเลื่อนเศษจากการหารนี้ออกไปโดยเปิดวงจรสวิตช์ sw1 และเลื่อนสวิตช์ sw2 ไปอยู่ที่ตำแหน่ง 1 จากนั้นเลื่อนข้อมูลในรีจิสเตอร์อีก 7 ครั้ง

ในการถอดรหัสวงเวียนแฮมมิ่งที่มีการลดขนาด จะต้องหา $\rho(X)$ เพื่อกำหนดจุดป้อนข้อมูล r(X) ให้แก่รีจิสเตอร์ในวงจรถอดรหัส (รูปที่ 3.20) โดย $\rho(X)$ คือเศษจากการหาร X^{n-k+l} ด้วย g(X) ผลจากการหารได้

$$\rho(X) = 1 + X^4 + X^5 + X^6$$



รูปที่ 3.20 วงจรตรวจสอบและแก้ไขความผิดพลาดของรหัสวงเวียนแฮมมิ่งลดขนาด (80,73) ที่กำเนิดโดย g(X) = 1+X³+X⁷

รูปที่ 3.20 แสดงวงจรถอดรหัสในรูปแบบของฮาร์ดแวร์ ในการนำไปใช้งาน จะแปลงให้อยู่ในรูปของโปรแกรมเพื่อให้ไมโครคอนโทรลเลอร์เป็นผู้ทำหน้าที่ตรวจสอบและแก้ไขข้อผิดพลาดของข้อมูล การทำงานของวงจรคือการหารพหุนามข้อมูล d(X) ขนาด 73 บิตด้วย g(X) จะได้เศษจากการหารอยู่ใน $b_0..b_6$

การทำงานของวงจรเริ่มต้นจากรีจิสเตอร์ทุกตัวในวงจรเป็น "0" ข้อมูลที่จะตรวจสอบเข้ามาทาง r(X) โดยบิตที่มีนัยสำคัญมากเข้ามาก่อน ในขณะที่ข้อมูลจำนวน 80 บิตเข้ามา สวิตช์ sw1, sw2 และ sw3 ปิดวงจร และสวิตช์ sw4 เปิดวงจร การทำงานในช่วง 80 บิตนี้เป็นการคำนวณหาซินโดรม s(X) เมื่อข้อมูลเลื่อนเข้ามาครบ 80 บิต ซินโดรม s(X) จะอยู่ในรีจิสเตอร์ $b_0..b_6$

หลังจากนี้จะเป็นการแก้ไขความผิดพลาดของข้อมูล(ถ้ามี) สวิตช์ sw1 และ sw2 เปิดวงจร สวิตช์ sw3 และ sw4 ปิดวงจร รีจิสเตอร์ทุกตัวจะถูกเลื่อนไปอีก 80 ครั้ง แต่ละครั้งที่เลื่อน จะให้ข้อมูลที่ถูกต้องออกมาที่ r'(X) การแก้ไขจะเกิดขึ้นเมื่อซินโดรม s(X) ที่ถูกเลื่อนไปมีค่าเป็น X^6 ($B=1000000_2$) ผลลัพธ์จากเกตแอนด์จะให้ค่าเป็น "1" ไปเอ็กซ์คลูซีฟออร์กับบิตข้อมูลที่กำลังออกมาจากบัฟเฟอร์ (80-bit register) ซึ่งคาดว่าเป็นบิตที่ผิด ให้กลับเป็นตรงกันข้าม ซึ่งคาดว่าจะเป็นข้อมูลที่ถูก นอกจากนั้นผลลัพธ์ "1" จากเกตแอนยังป้อนกลับไปยังซินโดรมทำให้ค่าในรีจิสเตอร์ซินโดรมทุกตัวเป็น "0" และเป็นศูนย์ตลอดการเลื่อนที่เหลือจนครบ 80 บิต ข้อดีของการทำให้ซินโดรมเป็น 0 นี้คือทำให้รีจิสเตอร์ $b_0..b_6$ พร้อมที่จะรับข้อมูลชุดใหม่ได้ทันที

ในกรณีที่ข้อมูลที่ได้รับมาไม่มีบิตผิดพลาด ซินโดรม s(X)=0 และจะเป็นศูนย์ตลอดการเลื่อน เกตแอนด์จะให้ผลลัพธ์ "0" ตลอดเช่นกัน r'(X) ที่ได้จะเท่ากับ r(X) ทุกประการ

3.2.6 การเข้ารหัสลับ

เช่นเดียวกันกับการควบคุมและแก้ไขข้อผิดพลาดในการส่งข้อมูล การเข้ารหัสลับที่เสนอไว้นี้ยังมิได้นำไปประยุกต์ใช้งานกับเครื่องต้นแบบแต่อย่างใด

การเข้ารหัสลับใช้ประโยชน์จากลำดับสุ่มเทียม โดยเรียกลำดับสุ่มเทียมที่ได้เสียใหม่ว่าลำดับกวนข้อมูล (confusion sequence) การเข้ารหัสลับทำโดยนำลำดับกวนข้อมูลนี้ไปเอกซ์คลูซีฟออร์กับข้อมูลที่ต้องการเข้ารหัสลับบิตต่อบิต กุญแจที่ใช้สำหรับการเข้ารหัสลับคือตัวเลขเริ่มต้นของลำดับสุ่มเทียมนั่นเอง

การถอดรหัสทำได้โดยนำกุญแจตัวเดียวกันมากำเนิดลำดับกวนข้อมูลซึ่งจะมีลักษณะเหมือนกันกับลำดับที่สร้างในตอนเข้ารหัสลับทุกประการ นำลำดับที่ได้ไปเอกซ์คลูซีฟออร์กับข้อมูลที่เข้ารหัสลับแล้วจะได้ข้อมูลต้นฉบับกลับมา

เมื่อพิจารณาความแข็งแรงของการเข้ารหัสลับวิธีนี้ด้วยการคิดว่าจะลักลอบถอดรหัสข้อมูลได้อย่างไร ในการถอดรหัสต้องทราบข้อมูล 2 อย่างคือ สัมประสิทธิ์ของรีจิสเตอร์เลื่อนที่ใช้ในการกำเนิดลำดับกวนข้อมูล C *(ดูรายละเอียดในหัวข้อ 3.2.4.1 วิธีการกำเนิดลำดับสุ่มเทียม หน้า 29)* และต้องทราบกุญแจหรือตัวเลขเริ่มต้นซึ่งปกติจะปกปิดไว้ ไม่สามารถทราบได้ทั้ง 2 อย่าง แต่สมมติว่าผู้ลักลอบทราบสัมประสิทธิ์ C และมีข้อมูลที่เข้ารหัสลับแล้ว ผู้ลักลอบจะทราบข้อมูลต้นฉบับได้ก็ต่อเมื่อทราบกุญแจ หรือทราบลำดับ แต่ผู้ลักลอบจะทราบลำดับได้ก็ต่อเมื่อทราบข้อมูลต้นฉบับก่อนซึ่งเป็นสิ่งที่ต้องการหา การที่ผู้ลักลอบจะหาลำดับได้จะต้องใช้วิธี Known Plaintext Attack โดยผู้ลักลอบทดลองใส่ข้อมูลต้นฉบับที่ทราบเพื่อให้ได้ข้อมูลที่เข้ารหัสลับแล้ว แต่ในกรณีนี้ผู้ลักลอบไม่สามารถทดลองใส่ข้อมูลต้นฉบับได้ และในความเป็นจริงก็เป็นการยากที่ผู้ลักลอบจะทราบสัมประสิทธิ์ C การถอดรหัสจึงทำได้ยาก