วิชา Data Communication Laboratory
ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

การทดลองที่ 5 Cyclic Redundancy Check

วัตถุประสงค์

1. เพื่อศึกษาวิธีการตรวจสอบความผิดพลาดในการรับ-ส่งข้อมูล
2. เพื่อสามารถสร้างวงจรในการตรวจสอบความผิดพลาดของข้อมูลแบบ CRC ได้

ทฤษฎี

CRC (Cyclic Redundancy Check) เป็นวิธีการที่ใช้ในสำหรับตรวจสอบความผิดพลาดของข้อมูลวิธีการหนึ่ง โดยเทคนิคนี้ใช้หลักการหาร โพลิโนเมียลของ เริ่มด้วยการแทนบิตข้อมูลด้วย โพลิโนเมียลซึ่งมีค่าสัมประสิทธิ์ 0 และ 1 โดยบิตข้อมูลที่มีความยาว k บิต จะแทนด้วยโพลีโนเมียลยาว k เทอมตั้งแต่ x^{k-1} ถึง x^0 เช่น บิตข้อมูลซึ่งมีค่า 110001 จะแทนด้วยโพลีโนเมียล $1\times x^5+1\times x^4+0\times x^3+0\times x^2+0\times x^1+1\times x^0 = 1\times x^5+1\times x^4+1$ เทคนิค CRC จะใช้โพลิโนเมียลก่อกำเนิด (Generator Polynomial) $G(x)$ เป็นตัวหารเพื่อสร้างผลหาร $R(x)$ โดย $G(x)$ เป็นโพลิโนเมียลที่มีกำลังเป็น g ซึ่งจะมีกำลังน้อยกว่ากำลังของโพลิโนเมียลของข้อมูล $D(x)$

หลักการของการส่งข้อมูล ที่มีการการตรวจสอบข้อมูลที่ผิดพลาดโดยใช้ Cyclic Redundancy Check

1. คูณข้อมูล $D(x)$ ด้วย x^g (เป็นการเลื่อนบิตข้อมูลไป g บิต)
2. จากนั้นหารผลคูณของ $D(x)\,x^g$ ด้วย $G(x)$ ผลหารที่ได้คือ $Q(x)$ และส่วนที่เหลือที่เหลือจากการหารคือ $R(x)$ ตามสมการที่ (1)

$$\frac{x^g D(x)}{G(x)} = Q(x) \oplus \frac{R(x)}{G(x)} \tag{1}$$

โดยที่ $R(x)$ จะมีค่าน้อยกว่า $G(x)$ เสมอ

2. $R(x)$ จะถูกบวกเข้ากับข้อมูลที่มีการเลื่อนบิต เพื่อสร้างเฟรมที่ใช้ในการส่งคือ $C(x)$ ดังนี้

$$C(x) = x^g D(x) \oplus R(x) \tag{2}$$

หลักการตรวจสอบความผิดพลาดขอข้อมูลที่ได้รับมาเป็นดังนี้

เมื่อด้านรับได้รับเฟรมข้อมูล $C(x)$ จะทำการหารด้วย $G(x)$ ดังสมการที่ (3)

$$\frac{C(x)}{G(x)} = \frac{x^g D(x) \oplus R(x)}{G(x)} \tag{3}$$

แทนเทอม $x^g \times \frac{D(x)}{G(x)}$ ด้วย $Q(x) \oplus \frac{R(x)}{G(x)}$ จะได้

$$\frac{C(x)}{G(x)}=Q(x)\oplus\frac{R(x)}{G(x)}\oplus\frac{R(x)}{G(x)} \tag{4}$$

ถ้าไม่มีข้อมูลผิดพลาดส่วนที่เหลือจากการหาร $\frac{R(x)}{G(x)}\oplus\frac{R(x)}{G(x)}$ จะเป็น 0

สมมติว่าต้องการส่งข้อมูล 101101001 โดยมีโพลิโนเมียลก่อกำเนิด $G(x)$ คือ 101001 (สามารถเขียนให้อยู่ในรูปของโพลีโนเมียลได้เป็น x^5+x^3+1) ทำการหาค่า $R(x)$ และ $C(x)$ ได้ตามขั้นตอนที่กล่าวมาดังนี้

1. $G(x)$ มี $g=5$ ซึ่งหมายความว่า $C(x)$ ถูกเลื่อนไปทางซ้าย 5 บิต ได้ผลลัพธ์เป็น

$$C(x)\,x^5=10110100100000$$

2. จากขั้นตอนที่ 1 นำผลลัพธ์มาหารด้วย $G(x)$ ได้ผลดังนี้

```
                1 0 0 1 0 1 0 1 0 (Quotient)
101001 ) 1 0 1 1 0 1 0 0 1 0 0 0 0 0
       ⊕ 1 0 1 0 0 1
         -----------
               1 0 0 0 0 1
             ⊕ 1 0 1 0 0 1
               -----------
                   1 0 0 0 0 0
                 ⊕ 1 0 1 0 0 1
                   -----------
                       1 0 0 1 0 0
                     ⊕ 1 0 1 0 0 1
                       -----------
                         1 1 0 1 0 (Remainder)
```

3. เศษ $R(x)$ ที่ได้จากการหารมีค่าเป็น 11010
4. ทำการบวก $R(x)$ เข้ากับข้อมูล $C(x)\,x^g$ เพื่อให้ได้ข้อมูลที่พร้อมทำการส่งคือ $C(x)$

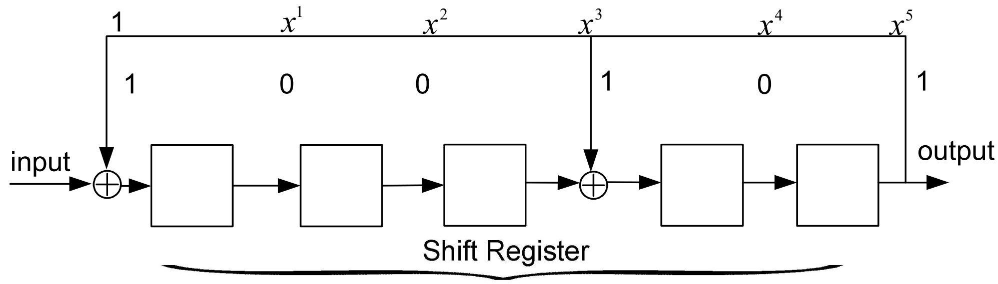
```
10110100100000
         11010 +
--------------
10110100111010
==============
```

ดังนั้นข้อมูลที่จะถูกส่งออกไปคือ 101101001 11010

5. ทางด้านรับจะทำการหารข้อมูลที่รับเข้ามาหารด้วย $G(x)$ ดังนี้

```
                1 0 0 1 0 1 0 1 0 (Quotient)
101001 ) 1 0 1 1 0 1 0 0 1 1 1 0 1 0
       ⊕ 1 0 1 0 0 1
         -----------
               1 0 0 0 0 1
             ⊕ 1 0 1 0 0 1
               -----------
                   1 0 0 0 1 1
                 ⊕ 1 0 1 0 0 1
                   -----------
                       1 0 1 0 0 1
                     ⊕ 1 0 1 0 0 1
                       -----------
                                 0 (Remainder)
```

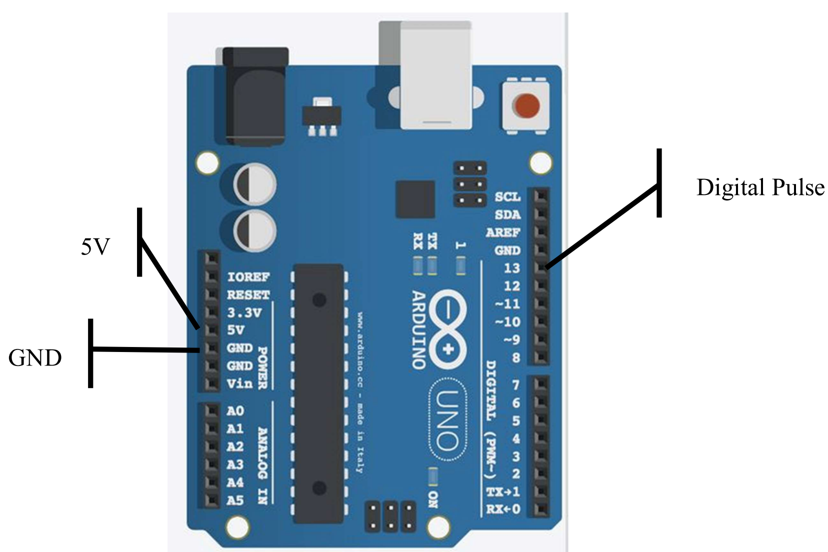
6. จากโพลิโนเมียลก่อกำเนิด x^5+x^3+1 สามารถต่อวงจรเพื่อหาผลหาร $R(x)$ นี้ได้ดังรูปที่ 5.1



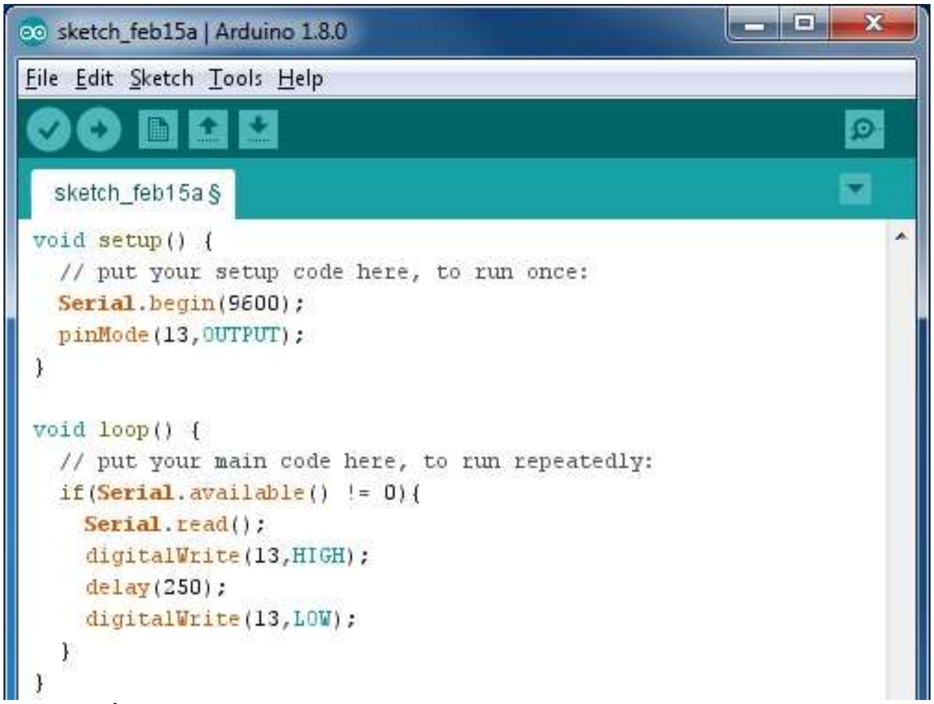
รูปที่ 5.1 CRC generating circuit $\left(x^5+x^3+1\right)$

การทดลองที่ 5.1 การสร้างค่าที่จะส่งด้วยวิธี Cyclic Redundancy Check

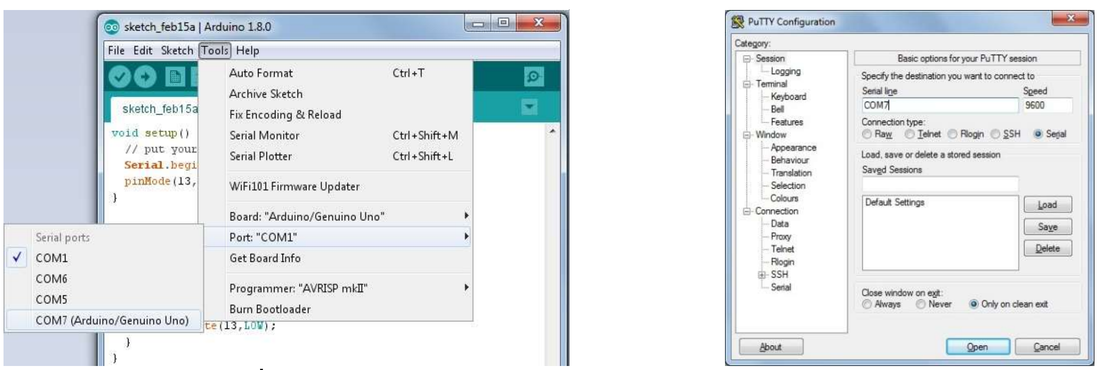
1. ใช้ Arduino UNO R3 เป็น Power supply และ Digital Pulse โดยที่
   1.1. ใช้ 5V และ GND เป็น Power supply และ Port 13 เป็น Digital Pulse ดังรูปที่ 5.2
   1.2. เปิดโปรแกรม Arduino แล้วพิมพ์โปรแกรมตามรูปที่ 5.3 แล้วเลือก Verify
   1.3. ต่อสาย Upload
   1.4. เลือก Com Port ที่เป็น Arduino/Genuino Uno ดังรูปที่ 5.4 แล้วเลือก Upload
   1.5. ปิดโปรแกรม Arduino
   1.6. เปิดโปรแกรม puty แล้วเลือก Com Port ตามข้อ 1.4
   1.7. ใช้ Enter ในการสร้าง Digital Pulse ออกที่ Port 13



รูปที่ 5.2 Arduino UNO R3



รูปที่ 5.3 โปรแกรมสร้าง Digital Pulse ออกที่ Port 13 ของ Arduino UNO R3



รูปที่ 5.4 การเลือก Com Port ของโปรแกรม Arduino และ puty

2. สร้างวงจร CRC generating ที่ใช้โพลีโนเมียล $G(x)$ **ตามที่อาจารย์กำหนดให้หน้าชั้น**
3. Clear ค่าในวงจรแล้ว เริ่มทำการป้อนข้อมูล $D(x)\ x^g$ เพื่อหาผลลัพธ์ $R(x)$
   3.1. ให้นำรหัสนักศึกษา 3 ตัวท้าย (มีเงื่อนไขว่า ถ้าน้อยกว่า 400 ต้องบวกด้วยค่า 321 ก่อน นำไปใช้) มาคิดเป็นข้อมูลใช้เป็นข้อมูล $D(x)$ โดยแปลงเป็นเลขฐานสองแบบ BCD
4. บันทึกผลลัพธ์ $R(x)$ ที่ได้จากการทดลอง

5. แสดงผล $C(x)$ ที่ต้องใช้งาน

6. แสงวิธีการคำนวณหา Remainder $R(x)$ และ ค่า $C(x)$ โดยใช้ค่า $G(x)$ จากข้อ 2 และ $D(x)$ จากข้อ 3 แล้วตรวจสอบว่าผลที่ได้จากการทดลองถูกต้องหรือไม่

รหัส 630100 69
รหัส 3 ตัวท้าย : 069+321 = 390
แปลงเป็น BCD : 0011 1001 0000 0000

CRC : x^4+x^3+1 = 1 1 0 0 1

```
                    1 0 1 1 0 1 0 1 1 0
1 1 0 1 ) 0 0 1 1 1 0 0 1 0 0 0 0 0 0 0 0
              1 1 0 0 1
              ---------
                  1 0 1 1 0
                  1 1 0 0 1
                  ---------
                    1 1 1 1 0
                    1 1 0 0 1
                    ---------
                        1 1 1 0 0
                        1 1 0 0 1
                        ---------
                          1 0 1 0 0
                          1 1 0 0 1
                          ---------
                            1 1 0 1 0
                            1 1 0 0 1
                            ---------
                                  1 1 0
```

∴ R(x) : 0110
C(x) : 0110 1001 0000 0110

การทดลองที่ 5.2 การถอดรหัสกรณีที่ไม่มีความผิดพลาด

1. Clear ค่าในวงจรแล้ว
2. นำค่า $C(x)$ ที่ได้จากการทดลองตอนที่ 1 ป้อนเข้าวงจร
3. บันทึกผลลัพธ์ $R(x)$ ที่ได้

การทดลองที่ 5.3 การถอดรหัสกรณีที่มีความผิดพลาด

1. นำค่า $C(x)$ ที่ได้จากการทดลองที่ 5.1 มาเปลี่ยนแปลงเพื่อให้ข้อมูลเกิดความผิดพลาด โดยสลับค่า 0 – 1 ตั้งแต่ข้อมูลบิตที่ 11-14 ข้อมูลที่เปลี่ยนแล้วเป็น
2. แสงวิธีการคำนวณหา Remainder $R(x)$ โดยนำ $C(x)$ ที่เปลี่ยนแปลงเพื่อให้ผิดพลาดจากข้อ 1 หารด้วย $G(x)$ จากข้อ 1 (การทดลองตอนที่ 5.1)

	C(x)		:	0011		1001		0000		0110									
	C(x)+E(x)			:	0011		1001		0011	1010									
				0	0	0	0	0	0	1	0	1	1	0	1	0	1	0	0
	11001		)	0	0	1	1	1	0	0	1	0	0	1	1	1	0	1	0
						1	1	0	0	1									
								1	0	1	1	0							
								1	1	0	0	1							
									1	1	1	1	0						
									1	1	0	0	1						
											1	1	1	1	1				
											1	1	0	0	1				
													1	1	0	1	0		
													1	1	0	0	1		
																1	1	1	0
∴	เศษเป็น			1110	ซึ่งแปลว่าตรวจพบ							Error							

3. นำค่า $C(x)$ ที่ผิดพลาดจากข้อ 1 ป้อนเข้าวงจร
4. บันทึกผลลัพธ์ $R(x)$ ที่ได้จากการทดลอง แล้วตรวจสอบว่าตรงกับที่คำนวณหรือไม่
5. เชิญอาจารย์ตรวจผลการทดลอง