การออกแบบและสร้างโปรแกรมคำนวนค่า CRC โดยใช้หน่วย CRC ภายในไมโครคอนโทรลเลอร์ STM32L152

DESIGN AND IMPLEMENTATION OF CRC CALCULATION PROGRAM USING CRC MODULE IN STM32L152 MICROCONTROLLER

นายสรวิศ สถาพรชัยสิทธิ์

โครงงานสหกิจศึกษานี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
ปริญญาวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีไทย – ญี่ปุ่น
พ.ศ. 2555

การออกแบบและสร้างโปรแกรมคำนวนค่า CRC โดยใช้
หน่วย CRC ภายในไมโครคอนโทรลเลอร์ STM32L152
เสนอ บริษัท เอ็นดีอาร์ โซลูชั่น (ประเทศไทย) จำกัด

DESIGN AND IMPLEMENTATION OF CRC CALCULATION
PROGRAM USING CRC MODULE IN
STM32L152 MICROCONTROLLER
PRESENTED TO NDR SOLUTION (THAILAND) CO., LTD.

นายสรวิศ สถาพรชัยสิทธิ์

โครงงานสหกิจศึกษานี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
ปริญญาวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีไทย – ญี่ปุ่น
พ.ศ. 2555

คณะกรรมการสอบ

….....ประธานกรรมการสอบ
(อาจารย์ ดร.วิมล แสนอุ้ม)

….....กรรมการสอบ
(อาจารย์ปริวัตร คงกำเนิด)

….....อาจารย์ที่ปรึกษา
(อาจารย์ต่อเกียรติ ใต้ธงชัย)

….....ประธานสหกิจศึกษาสาขาวิชา
(อาจารย์ ดร.วรากร ศรีเชวงทรัพย์)

ลิขสิทธิ์ของสถาบันเทคโนโลยีไทย - ญี่ปุ่น

ชื่อโครงงาน	การออกแบบและสร้างโปรแกรมคำนวนค่า CRC โดยใช้หน่วย CRC ภายในไมโครคอนโทรลเลอร์ STM32L152 เสนอ บรษัท เอ็นดีอาร์ โซลูชั่น (ประเทศไทย) จำกัด DESIGN AND IMPLEMENTATION OF CRC CALCULATION PROGRAM USING CRC MODULE IN STM32L152 MICROCONTROLLER PRESENTED TO NDR SOLUTION (THAILAND) CO., LTD.
ผู้เขียน	นายสรวิศ สถาพรชัยสิทธิ์
คณะวิชา	วิศวกรรมศาสตร์ สาขาวิชา วิศวกรรมคอมพิวเตอร์
อาจารย์ที่ปรึกษา	อาจารย์ ต่อเกียรติ ใต้ธงชัย
พนักงานที่ปรึกษา	นายกฤษฎา อัจฉริยพัฒน์
ชื่อบริษัท	เอ็นดีอาร์ โซลูชั่น (ประเทศไทย) จำกัด
ประเภทธุรกิจ / สินค้า	ให้บริการทางด้านรับออกแบบพัฒนา ด้านวิศวกรรมอิเล็กทรอนิกส์และคอมพิวเตอร์ รับจ้างออกแบบพัฒนาฮาร์ดแวร์และระบบสมองกลฝังตัวให้กับลูกค้าที่อยู่ทั้งในและต่างประเทศ

บทสรุป

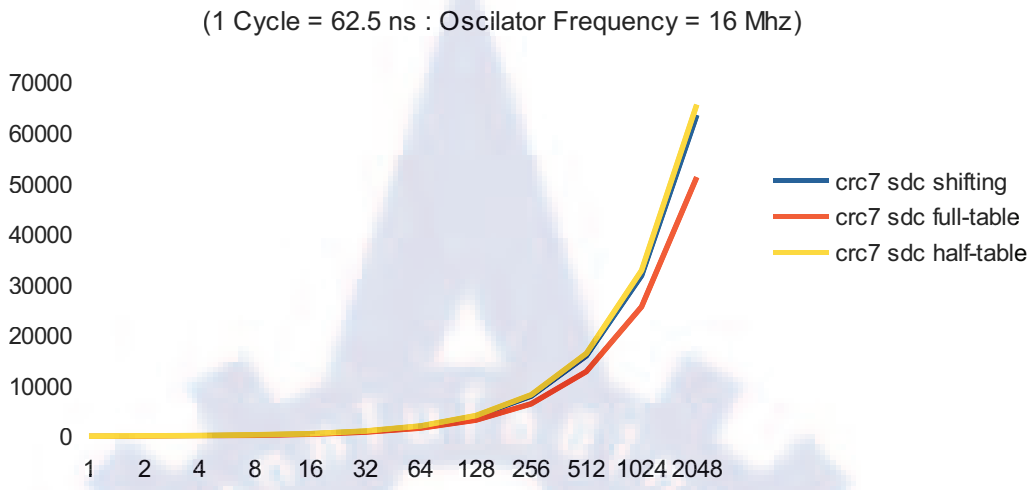
งานที่ปฏิบัติ

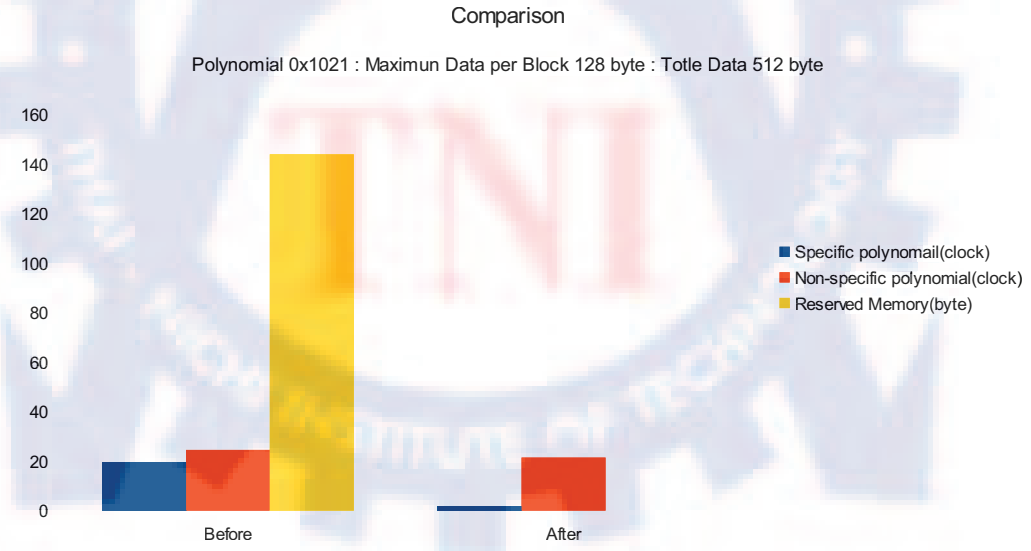
ส่วนแรก คือ การดำเนินงานภายในสถานประกอบการ ซึ่งประกอบไปด้วยการออกแบบและสร้างโปรแกรมต่างๆ เช่น การออกแบบและเขียนระบบทำงานสำหรับควบคุมการทำงานของอุปกรณ์อิเล็กทรอนิกส์ต่างๆ

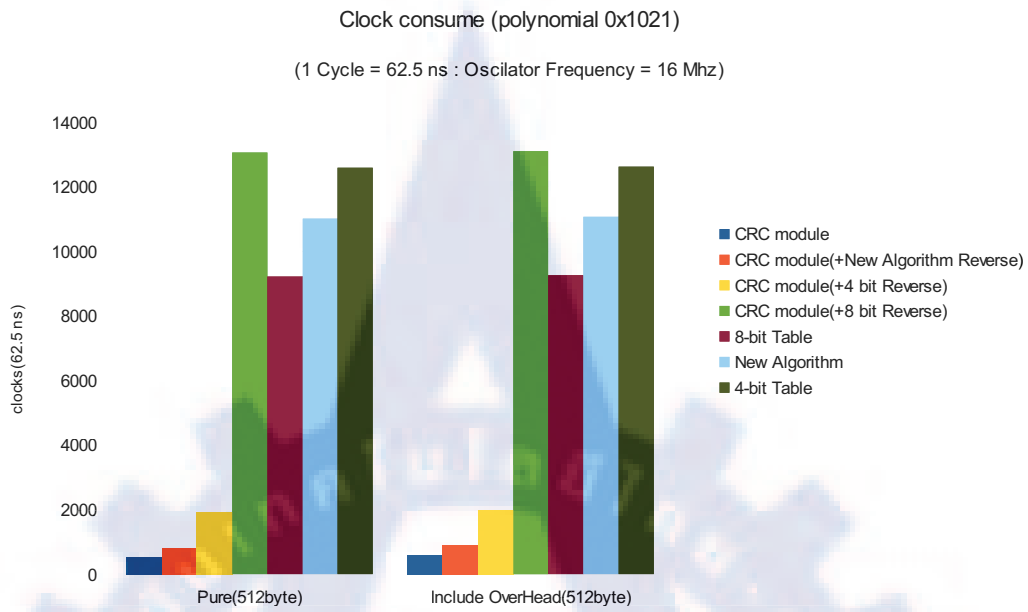
ส่วนที่สอง คือ การออกแบบและดำเนินงานในส่วนของโครงการที่ต้องการเสนอต่อสถานประกอบการ โดยมีจุดมุ่งหมาย เพื่อเพิ่มศักยภาพและความสามารถของฮาร์แวร์ที่กำลังถูกใช้งานภายในสถานประกอบการ งานที่ได้รับมอบหมายคือ การคำนวนค่า CRC โดยใช้งาน CRC Module ภายในไมโครคอนโทรลเลอร์ให้มีประสิทธิภาพสูงสุด โดยลดปริมาณหน่วยความจำและเวลาที่ใช้ให้เหลือน้อยที่สุด

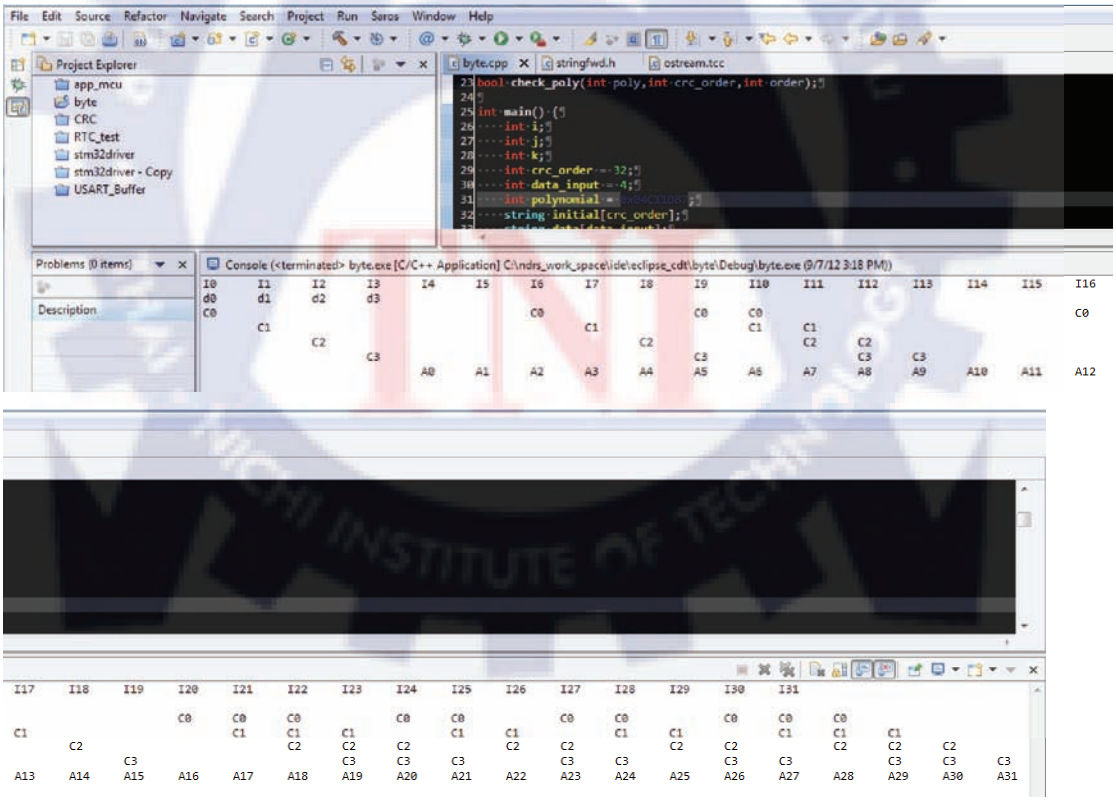
ผลที่ได้รับจากการดำเนินงานและประโยชน์ที่ได้รับ

จากการปฏิบัติงานในสหกิจศึกษา ทำให้ข้าพเจ้าได้รับความรู้ ความเข้าใจในด้านการออกแบบและการสร้างระบบสมองกลฝังตัว โดยศึกษาจากโปรแกรมที่พัฒนาขึ้นในสถานประกอบ อีกทั้งยังได้เรียนรู้เครื่องมือสำหรับการสร้างซอฟต์แวร์ ที่ช่วยอำนวยความสะดวกทั้งในด้านการออกแบบและสร้างโปรแกรมต่างๆขึ้น รวมไปถึงได้ศึกษาเรียนรู้ในลักษณะการดำเนินงานต่างๆทั้งในด้านการออกแบบ และการพัฒนาโปรแกรม โดยใช้หลักการวิศวกรรมซอฟต์แวร์


(1 Cycle = 62.5 ns : Oscilator Frequency = 16 Mhz)
70000
60000
50000
40000
30000
20000
10000
0
1
2
4
8
16
32
64
128
256
512
1024
2048
crc7 sdc shifting
crc7 sdc full-table
crc7 sdc half-table


Comparison
Polynomial 0x1021 : Maximun Data per Block 128 byte : Totle Data 512 byte
160
140
120
100
80
60
40
20
0
Before
After
Specific polynomail(clock)
Non-specific polynomial(clock)
Reserved Memory(byte)


Clock consume (polynomial 0x1021)
(1 Cycle = 62.5 ns : Oscilator Frequency = 16 Mhz)
clocks(62.5 ns)
14000
12000
10000
8000
6000
4000
2000
0
Pure(512byte)
Include OverHead(512byte)
CRC module
CRC module(+New Algorithm Reverse)
CRC module(+4 bit Reverse)
CRC module(+8 bit Reverse)
8-bit Table
New Algorithm
4-bit Table


File Edit Source Refactor Navigate Search Project Run Sanos Window Help
Project Explorer
app_mcu
byte
CRC
RTC_test
stm32driver
stm32driver - Copy
USART_Buffer
byte.cpp
stringfwd.h
ostream.tcc
bool check_poly(int poly,int crc_order,int order);
int main() {
int i;
int j;
int k;
int crc_order = 32;
int data_input = 4;
int polynomial =
string initial[crc_order];
Problems (0 items)
Description
Console (<terminated> byte.exe [C/C++ Application] C:\ndrs_work_space\ide\eclipse_cdt\byte\Debug\byte.exe (9/7/12 3:18 PM))

กิตติกรรมประกาศ

ขอแสดงความขอบคุณเป็นอย่างสูง ต่อ บริษัท เอ็นดีอาร์ โซลูชั่น (ประเทศไทย) จำกัด ที่ได้ให้ความสนับสนุน และไว้วางใจในการดำเนินงานของข้าพเจ้าให้สำเร็จลุล่วงไปด้วยดี ข้าพเจ้าขอขอบพระคุณ พี่พิศิษฏ์ สว่างวงศ์อนัน ในความไว้วางใจ และคอยให้คำเสนอแนะในการดำเนินงานต่างๆ รวมถึงตัวอย่างโปรแกรมที่ช่วยให้ข้าพเจ้าสามารถบรรลุเป้าหมายที่ทางสถานประกอบการได้มอบหมาย และข้าพเจ้าขอขอบพระคุณ พี่กฤษฎา อัจฉริยพัฒน์ ที่ได้ช่วยดูแลและติดตามการดำเนินงานต่างๆของข้าพเจ้าในฐานะพนักงานที่ปรึกษา และทำหน้าที่เป็นพี่เลี้ยงคอยดูแลข้าพเจ้าตลอดการดำเนินงานครั้งนี้ และข้าพเจ้าขอขอบพระคุณ พี่นภัทร ใจชื้น ที่ได้ช่วยดูแลและสั่งสอนสิ่งใหม่ๆ ตลอดถึงวิธีการทำงานและวิธีการเขียนโปรแกรม อีกทั้งคอยให้คำดูแลปรึกษาทุกครั้งที่เกิดปัญหาขึ้น และยังได้ให้ความไว้วางใจมอบหมายงานเพื่อให้ข้าพเจ้ามีความรู้และประสบการณ์ให้เกิดขึ้นอย่างเป็นประจำ ซึ่งเป็นการเสริมประสิทธิภาพและความสามารถในการดำเนินงานต่างๆในแต่ล่ะงานที่มอบหมายให้สำเร็จลุล่วงไปด้วยดีเสมอมา

สารบัญ

หน้า

บทสรุป ข

กิตติกรรมประกาศ ฉ

สารบัญ ช

รายการตาราง ฎ

รายการภาพประกอบ ฐ

บทที่

1. บทนำ **1**

1.1 ชื่อและที่ตั้งของสถานประกอบการ 1

1.2 ลักษณะธุรกิจของสถานประกอบการ หรือการให้บริการหลักขององค์กร 2

1.3 รูปแบบการจัดองค์กรและการบริหารองค์กร 2

1.3.1 แผนก Embedded System 2

1.3.2 แผนก Information Technology 2

1.4 ตำแหน่งและหน้าที่งานที่นักศึกษาได้รับมอบหมาย 2

1.5 พนักงานที่ปรึกษาและตำแหน่งของพนักงานที่ปรึกษา 3

1.6 ระยะเวลาที่ปฏิบัติงาน 3

1.7 วัตถุประสงค์ของการปฏิบัติงานที่ได้รับมอบหมายและโครงงาน 3

1.7.1 งานที่ได้รับมอบหมาย 3

1.7.2 โครงงานที่ข้าพเจ้าตั้งใจนำเสนอ 4

1.8 ผลที่คาดว่าจะได้รับจากการปฏิบัติงานและโครงงาน 4

สารบัญ (ต่อ)

หน้า

2. หลักการและเทคโนโลยีที่ใช้ในการปฏิบัติงาน **5**

2.1 การพัฒนารหัสโปรแกรมด้วยนักพัฒนาหลายคน และควบคุมด้วย Subversion 5

2.1.1 แนวคิดและการบริหารจัดการพัฒนารหัสโปรแกรมร่วมกันโดยใช้ Subversion 6

2.2 การสร้างและพัฒนารหัสโปรแกรมด้วย Eclipse IDE C/C++ บน Windows 7

2.2.1 โปรแกรม Eclipse 7

2.3 การทดสอบการทำงานของโปรแกรมบนบอร์ดด้วย IAR Embedded Workbench 7

2.3.1 การเชื่อมต่ออุปกรณ์ที่ใช้ดาวน์โหลดเฟิร์มแวร์และทดสอบบอร์ดด้วย ST-LINK 7

2.4 สังเกตุการณ์ข้อมูลที่เปลี่ยนแปลงภายในไมโครคอนโทรลเลอร์โดยใช้ STM Studio 10

3. แผนงานปฏิบัติงาน **11**

3.1 การปฏิบัติงานในสถานประกอบการ 11

3.1.1 แผนงานปฏิบัติงาน 11

3.1.2 รายละเอียดงานที่นักศึกษาปฏิบัติในสถานประกอบการ 11

3.1.3 ขั้นตอนการดำเนินงานที่นักศึกษาปฏิบัติงาน 13

3.2 โครงงานที่นำเสนอ 14

3.2.1 แผนงานปฏิบัติงาน 14

3.2.2 รายละเอียดโครงงาน 14

3.2.3 ขั้นตอนการดำเนินงานในโครงงาน 14

สารบัญ (ต่อ)

หน้า

4. ผลการดำเนินงาน การวิเคราะห์และสรุปผลต่างๆ **18**

4.1 การปฏิบัติงานในสถานประกอบการ 18

4.1.1 การออกแบบ USART API 18

4.1.2 การทดสอบ RTC API 18

4.1.3 การแก้ไข CLI API 18

4.1.4 การสร้าง CRC API 18

4.1.5 การเขียนเอกสาร STM Studio Guideline 19

4.1.6 การทดสอบ Memory API test 19

4.2 โครงงานที่นำเสนอ 20

4.2.1 ขั้นตอนและผลการดำเนินงาน 20

5. บทสรุปและข้อเสนอแนะ **21**

5.1 การปฏิบัติงานในสถานประกอบการ 21

5.1.1 สรุปผลการดำเนินงาน 21

5.2 โครงงานที่นำเสนอ 21

5.2.1 สรุปผลการดำเนินงาน 21

เอกสารอ้างอิง **22**

สารบัญ (ต่อ)

หน้า

ภาคผนวก **23**

ก. การใช้งาน Eclipse ขั้นพื้นฐาน 24

ก.1 การติดตั้ง : ดาวน์โหลดเครื่องมือมาจากเว็บไซต์ http://www.eclipse.org/ ดังรูปที่ ก.1 24

ก.2 การใช้งานร่วมกับส่วนเสริมของ Eclipse IDE C/C++ 30

ข. การใช้งาน IAR ขั้นพื้นฐาน 32

ข.1 การติดตั้ง 32

ข.2 หน้าต่างที่แสดงบน IAR Embedded Workbench 33

ค. การใช้งาน IAR ขั้นพื้นฐาน 35

5.2.1 วิธีการใช้งาน STM Studio 36

5.2.2 ตัวเลือกต่างๆใน STM Studio 36

5.2.3 การตั้งค่าเพื่อให้ STM Studio ใช้งานได้ 38

ง. ส่วนประกอบของการคำนวน CRC 41

จ. หลักการที่นำมาใช้ในการหา CRC 43

หลักการที่ 1 การคำนวน CRC ด้วยหลักการ Polynomial Long Division 43

หลักการที่ 2 หลักการคำนวน CRC โดยใช้ Table 44

หลักการที่ 3 หลักการคำนวน CRC โดยใช้ หลักการใหม่ 45

ฉ. การคำนวน CRC 48

ช. พิสูจณ์หลักการหา Reverse CRC โดยใช้สมการ 50

ซ. ตัวอย่างโปรแกรมที่ใช้คำนวน CRC แบบต่างๆ 51

สารบัญ (ต่อ)

หน้า

ฌ. แบบฟอร์มแจ้งรายละเอียดงาน ตำแหน่งงาน พนักงานที่ปรึกษา และที่ตั้งหน่วยงาน 53

ญ. แบบฟอร์มรายงานประจำสัปดาห์ 54

ฏ. แบบแจ้งแผนการปฏิบัติสหกิจศึกษา 70

ฏ. แบบแจ้งโครงร่าง / รายงานปฏิบัติงาน 71

ประวัติผู้แต่ง **73**

รายการตาราง

ตารางที่		หน้า
3.1	แผนงานโดยพนักงานที่ปรึกษา	11
3.2	แผนงานสำหรับโครงงานที่นำเสนอโดยข้าพเจ้า	14
4.1	เปรียบเทียบเวลาที่ใช้ของแต่ละวิธีในการคำนวนหา CRC	20
จ.1	แสดงการหา CRC CCITT ด้วยตัวแปร	45

รายการภาพประกอบ

รูปที่		หน้า
1.1	แสดงสัญลักษณ์ของสถานประกอบการ	1
1.2	แสดงที่ตั้งขอสถานประกอบการ	1
2.1	แสดงตัวอย่างโครงสร้างของการเก็บข้อมูล	6
2.2	แสดงตัวอย่างการเลือกชื่อของโครงการที่อยู่ในพื้นที่การทำงาน	8
2.3	หน้าต่างการปรับแต่งโครงการภายใน IAR Embedded Workbench	8
2.4	หน้าต่างตัวเลือกอุปกรณ์ที่ใช้เชื่อมต่อบอร์ดเป้าหมายใน IAR Embedded Workbench	9
2.5	แสดงการตั้งค่าใช้งานอุปกรณ์ที่เชื่อมต่อเพิ่มเติมใน IAR Embedded Workbench	9
2.6	หน้าต่างโปรแกรม STM Studio ที่กำลังทำงานอยู่	10
3.1	ความต้องการของบริษัท(ซ้าย) หลักการแก้ไขปัญหา(ขวา)	15
3.2	โครงสร้างสำหรับ CRC Calculation ที่ได้ออกแบบไว้	16
3.3	โครงสร้างสำหรับ CRC Calculation ที่ได้ปรับแก้แล้ว	16
ก.1	ตัวอย่างหน้าเว็บไซต์ของ Eclipse [2012, September 19]	24
ก.2	แสดงข้อมูลที่อยู่ภายในเอกสารอัดบีบชนิด ZIP ของ Eclipse	25
ก.3	หน้าต่างแสดงการเริ่มต้นใช้งาน Eclipse	25
ก.4	หน้าต่างแสดงที่ตั้งโฟลเดอร์ของโครงการ โดยโปรแกรม Eclipse	26
ก.5	หน้าต่างต้อนรับของโปรแกรม Eclipse	26
ก.6	หน้าต่างตั้งชื่อและกำหนดที่ตั้งโครงการของ Eclipse	27
ก.7	แสดงหน้าต่างภายหลังการสร้างโครงการใน Eclipse	27
ก.8	แสดงหน้าต่างการแปลงโครงการใน Eclipse	29
ก.9	แสดงหน้าต่างการเพิ่มโฟลเดอร์สำหรับจัดเก็บรหัสโปรแกรมใน Eclipse	29
ก.10	แสดงตัวเลือกในการเพิ่มเอกสารของ Eclipse	30
ก.11	ตัวอย่างหน้าต่างคู่มือการใช้งาน Eclipse	31
ข.1	แสดงส่วนหนึ่งในหน้าดาวน์โหลดของเว็บ IAR	32
ข.2	แสดงหน้าต่างแจ้งผู้ใช้เกี่ยวกับกฎการใช้งานโปรแกรม IAR	33
ข.3	แสดงลิ้งค์เชื่อมโยงโปรแกรม IAR Embedded Workbench	34

รายการภาพประกอบ (ต่อ)

รูปที่		หน้า
ข.4	แสดงหน้าต่างของโปรแกรม IAR Embedded Workbench เมื่อเปิดใช้งานครั้งแรก	34
ค.1	การแสดงผลแบบ Curve	35
ค.2	การแสดงผลแบบ Bar Graph	35
ค.3	การแสดงผลแบบ Table	36
ค.4	หน้าต่าง Setting ของ STM Studio	36
ค.5	ส่วนที่ใช้ในการตั้งค่าเพื่อใช้งาน STM Studio	38
ค.6	การเปลี่ยนเป็น Direct mode	39
ค.7	การตั้งค่า Settings	39
ง.1	ค่าต่างๆที่มีใน CRC Calculator ส่วนใหญ่	41
จ.1	แสดงคำตอบของ CRC CCITT	47
ฉ.1	ตัวอย่างความผิดพลาดที่เกิดขึ้นและไม่สามารถตรวจสอบได้	48
ฉ.2	วิธีการคำนวน CRC โดยวิธีการ Polynomial Long Division	49

บทที่ 1
บทนำ

1.1 ชื่อและที่ตั้งของสถานประกอบการ

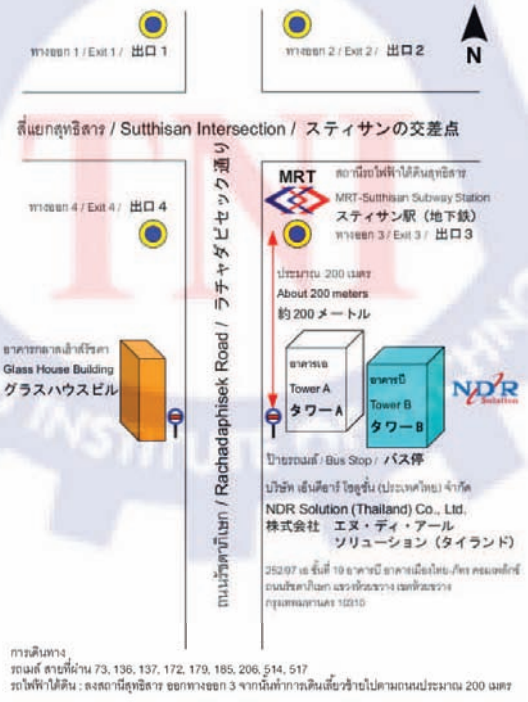
รูปที่ 1.1 แสดงสัญลักษณ์ของสถานประกอบการ

ชื่อ (ภาษาไทย) บริษัท เอ็นดีอาร์ โซลูชั่น (ประเทศไทย) จำกัด

ชื่อ (ภาษาอังกฤษ) NDR Solution (Thailand) Co., Ltd.

ชื่อ (ภาษาญี่ปุ่น) **株式会社 エヌ・ディ・アール ソリューション（タイランド）**

ที่ตั้ง 252/97 (A) ชั้น 19 อาคารเมืองไทย คอมเพล็กซ์ ถนนรัชดาภิเษก ห้วยขวาง กรุงเทพ 10310



รูปที่ 1.2 แสดงที่ตั้งสถานประกอบการ

1.2 ลักษณะธุรกิจของสถานประกอบการ หรือการให้บริการหลักขององค์กร

บริษัท เอ็นดีอาร์ โซลูชั่น (ประเทศไทย) จำกัด ก่อตั้งเป็นบริษัทร่วมทุนกับทางบริษัท NDR Co., Ltd. Japan ในการให้บริการทางด้านรับจ้างออกแบบพัฒนาด้านวิศวกรรมอิเล็กโทรนิกส์และคอมพิวเตอร์ รับจ้างออกแบบพัฒนาฮาร์ดแวร์ และระบบสมองกลฝังตัว (Embedded System) ให้แก่ลูกค้าที่อยู่ทั้งในและต่างประเทศ บริการทางด้านระบบเทคโนโลยีสารสนเทศให้กับกล่มธุรกิจทางด้านพลังงาน สร้างความร่วมมือกับสถานศึกษาในการส่งเสริมพัฒนาระบบสมองกลฝังตัวในไทย

1.3 รูปแบบการจัดองค์กรและการบริหารองค์กร

บริษัท เอ็นดีอาร์ โซลูชั่น (ประเทศไทย) จำกัด จัดการและบริหารการทำงานโดยแบ่งแผนกออกเป็น 2 แผนก ดังนี้

1.3.1 แผนก Embedded System

ทางแผนกจะคอยให้บริการแก่ลูกค้าในด้านการออกแบบแอพพลิเคชั่นบนไมโครคอนโทรลเลอร์ตามที่ลูกค้าต้องการ โดยการออกแบบนั้นทางแผนกจะเริ่มต้นจากการออกแบบกระบวนการทางความคิด ลำดับชั้นการทำงานของระบบ การสร้างเฟิรม์แวร์ การสร้างฮาร์ดแวร์ต้นแบบ ตลอดจนถึงการออกแบบเป็นผลิตภัณฑ์ขั้นสุดท้าย

1.3.2 แผนก Information Technology

ทางแผนกจะคอยให้บริการแก่ลูกค้าในด้านการออกแบบ สร้างและดูแลระบบรวมถึงการนำเสนอข้อมูลสารสนเทศผ่านทางอินเทอร์เน็ตหรือเครือข่ายตามที่ลูกค้าต้องการ โดยจะให้บริการลักษณะการออกแบบเป็นเว็บแอพพลิเคชั่นบนเซิร์ฟเวอร์ของลูกค้าได้

1.4 ตำแหน่งและหน้าที่งานที่นักศึกษาได้รับมอบหมาย

ข้าพเจ้าได้รับมอบหมายให้เป็นผู้ช่วยและสนับสนุนในโครงการ SMART METER ภายในแผนก Embedded System ตำแหน่ง Embedded Software Developer โดยมีหน้าที่คอยสนับสนุนภายในแผนก เช่น การพัฒนา USART API และ CRC API สำหรับไมโครคอลโทรลเลอร์ การตรวจ

สอบ แก้ไข ทดสอบโครงสร้างของรหัสโปรแกรมของนักพัฒนาท่านอื่น การทำเอกสารสำหรับช่วยเหลือในการทำงานและวิเคราะห์ข้อมูลจากไมโครคอลโทรลเลอร์

1.5 พนักงานที่ปรึกษาและตำแหน่งของพนักงานที่ปรึกษา

พนักงานที่ปรึกษาคือ นาย กฤษฎา อัจฉริยพัฒน์ ดำรงตำแหน่ง Supervisor Embedded Software Developer ซึ่งเป็นผู้คอยมอบหมายหัวข้องานย่อยๆภายในแผนก โดยงานที่ได้รับมอบหมายนั้น มักจะเป็นงานที่เน้นการศึกษาโดยจะมีเวลาจำกัดหรือไม่ก็แล้วแต่กรณี แต่ทุกงานจะคำนึงถึงความสามารถของข้าพเจ้าเป็นหลัก และเพื่อเป็นการแบ่งเบาภาระงานภายในแผนกอีกด้วย

1.6 ระยะเวลาที่ปฏิบัติงาน

ข้าพเจ้าเริ่มการปฏิบัติงานตั้งแต่ 18 มิถุนายน 2555 จนถึงวันที่ 5 ตุลาคม 2555 เป็นระยะเวลา 16 อาทิตย์

1.7 วัตถุประสงค์ของการปฏบัติงานที่ได้รับมอบหมายและโครงงาน

การปฏิบัติงานของข้าพเจ้าในสถานประกอบการนี้สามารถจำแนกจุดประสงค์ของการดำเนินงานออกเป็น 2 ส่วนได้ดังนี้

1.7.1 งานที่ได้รับมอบหมาย

ข้าพเจ้ามีความตั้งใจที่จะหาความรู้พร้อมทั้งช่วยแบ่งเบาภาระงานภายในแผนกและสถานประกอบการให้มากที่สุดเท่าที่ความสามารถของข้าพเจ้าเอื้ออำนวย โดยยึดการทำงานของสถานประกอบการและข้อปฏิบัติที่ทางบริษัทกำหนดไว้ เพื่อให้งานที่ข้าพเจ้าได้ปฏิบัตินั้นเป็นประโยชน์สูงสุดแก่สถานประกอบการ และเพื่อป้องกันปัญหาที่ข้าพเจ้าอาจจะก่อขึ้น ข้าพเจ้าจึงมีจุดมุ่งหมายในการปฏิบัติงาน ดังนี้

1) ช่วยแบ่งเบาภาระงานของแผนกให้มากที่สุดเท่าที่จะทำได้

2) พยายามหาความรู้จากงานที่ได้รับให้มากที่สุดเท่าที่เป็นไปได้

3) ศึกษาการทำงานเป็นทีม เพื่อลดปัญหาที่ข้าพเจ้าก่อให้เกิดขึ้นกับแผนก

4) ศึกษาเพิ่มเติมจากแหล่งข้อมูลภายนอกสถานประกอบการ เพื่อเพิ่มพูนความรู้ความสามารถในส่วนที่ข้าพเจ้ายังขาดตกบกพร่อง ซึ่งอาจจะเป็นอุปสรรคต่อการดำเนินงานบางอย่างในอนาคต

5) มีความรับผิดชอบต่องานที่ได้รับมอบหมาย โดยทำงานให้เสร็จทันเวลาและพยายามทำให้เกิดปัญหาน้อยที่สุดเท่าที่เป็นไปได้ และน้อมรับข้อตำหนิ ติเตียน หรือข้อเสนอแนะต่างๆที่มีต่อผลงานของข้าพเจ้า เพื่อนำมาใช้พัฒนาตนเองต่อไป

1.7.2 โครงงานที่ข้าพเจ้าตั้งใจนำเสนอ

ข้าพเจ้ามีความตั้งใจที่จะถ่ายทอดความรู้จากการปฏิบัติงาน ณ สถานประกอบการแห่งนี้ โดยหวังให้นักศึกษารุ่นน้อง หรือผู้ที่มีความสนใจทั่วไป นำไปใช้ประโยชน์ต่อไปได้ โดยโครงงานจะพยายามนำหลักการที่ข้าพเจ้าใช้ในการค้นคว้า มานำเสนอเพื่อให้ผู้อ่านได้รู้ถึงหลักการและวิธีการที่ใช้ และหวังว่าผู้อ่านจะนำหลักการดังกล่าวไปใช้ประโยชน์อย่างสร้างสรรค์สืบต่อไป

1.8 ผลที่คาดว่าจะได้รับจากการปฏิบัติงานและโครงงาน

ข้าพเจ้ามีความคาดหวังว่าหลักการที่ข้าพเจ้าคิดขึ้นจะสามารถนำไปใช้งานในการคำนวนหาค่า CRC ทั้งในกรณีที่ไม่มี Hardware สำหรับคำนวน CRC และในกรณีที่มี Hardware แต่ไม่ได้ออกแบบมาให้สามารถโหลดค่า Initial ได้

นำไปใช้พัฒนาต่อเป็น Hardware Module สำหรับการคำนวนค่า CRC โดยเฉพาะเพื่อรองรับการรับส่งข้อมูลผ่านสายรับส่งสัญญาณความเร็วสูง นอกจากจะทำงานได้เร็วแล้ว ยังสามารถแก้ไข Polynomial เป็นค่าใดก็ได้ตามต้องการ เพื่อเพิ่มความสามารถในการเข้ารหัสของข้อมูล

เพื่อเพิ่มความสามารถในการปรับเปลี่ยนและแก้ไขโครงสร้างของ Protocol ที่ใช้งาน เพื่อลดปริมาณหน่วยความจำที่ต้องสงวนไว้สำหรับการคำนวนค่า CRC และไม่จำเป็นต้องเก็บข้อมูลทั้งหมดที่ได้รับให้อยู่ใน RAM ก่อนนำไปตรวจสอบความถูกต้องของข้อมูล

บทที่ 2
หลักการและเทคโนโลยีที่ใช้ในการปฏิบัติงาน

เนื่องจากข้าพเจ้าได้รับมอบหมายงานส่วนใหญ่เป็นการออกแบบและสร้างซอฟต์แวร์เป็นหลัก ดังนั้น เทคโนโลยีที่ใช้ส่วนใหญ่นั้นจะเป็นซอฟต์แวร์ที่นำมาช่วยสร้าง และพัฒนาซอฟต์แวร์ขึ้น โดยทางสถานประกอบการได้สนับสนุนการใช้งานซอฟต์แวร์ที่ถูกต้องตามลิขสิทธิ์ อีกทั้งสถานประกอบการยังสนับสนุนในการใช้งานโอเพ่นซอร์จ ซึ่งบางอย่างก็มีความสะดวกสะบายมากกว่าซอฟต์แวร์ที่ใช้กันตามปกติอีกด้วย

2.1 การพัฒนารหัสโปรแกรมด้วยนักพัฒนาหลายคน และควบคุมด้วย Subversion

ทางสถานประกอบการมีนโยบายในการบริหารและจัดเก็บรหัสโปรแกรมโดยใช้ Subversion ซึ่ง Subversion เป็นซอฟต์แวร์ที่ใช้เป็นตัวกลางในการจัดการบริหารไฟล์ข้อมูลและรหัสโปรแกรมต่างๆจากพื้นที่ทำงานเดียวโดย Subversion นี้มีความสามารถหลายอย่าง และข้อดีของ Subversion คือ ช่วยป้องกันการแก้ไขรหัสโปรแกรมในขณะเดียวกัน อีกทั้งยังสามารถแสดงสถานะให้แก่นักพัฒนาทุกคนทราบได้ว่า ในขณะนั้นมีซอฟต์แวร์ใดที่กำลังพัฒนาอยู่และได้พัฒนาไปถึงไหนแล้ว มีผู้ใดเป็นผู้แก้ไขหรือทำเพิ่มเติมส่วนใดบ้าง อีกทั้งยังสามารถย้อนกลับไปดูข้อมูลย้อนหลังได้ และสามารถกู้ไฟล์กลับไป ณ เวลาที่ทำการเก็บข้อมูลได้ทุกเมื่อ มีประโยชน์อย่างยิ่งในเวลาที่เกิดปัญหาในการเพิ่มเติมหรือแก้ไขรหัสโปรแกรม

ทางสถานประกอบการได้เลือกใช้โปรแกรม Bazaar ซึ่งเป็นโปรแกรมที่ใช้บริหารจัดการเอกสาร จัดเก็บความเปลี่ยนแปลงของเอกสาร รวมถึงการติดตามโครงการและการพัฒนารหัสโปรแกรมร่วมกันระหว่างทีมงานในลักษณะเดียวกันกับแนวคิดข้างต้น โปรแกรม Bazaar เป็นส่วนหนึ่งของโครงการ GNU ซึ่งพัฒนาโดยใช้ Subversion เป็นโครงสร้างหลัก โดยโปรแกรมนี้มีลักษณะเป็นโอเพ่นซอร์จ จึงไม่มีค่าธรรมเนียมใดๆในการใช้งาน และโปรแกรมนี้มีการสนับสนุนการพัฒนาโดยทีมงาน Canonical

2.1.1 แนวคิดและการบริหารจัดการพัฒนารหัสโปรแกรมร่วมกันโดยใช้ Subversion

จากแนวคิดที่กล่าวไปในก่อนหน้านี้ การบริหารจัดการรหัสโปรแกรมจะต้องกระทำโดยรหัสโปรแกรมที่อยู่ในพื้นที่การทำงานเดียวกัน ดังนั้นก่อนการเริ่มโครงการพัฒนารหัสโปรแกรมใดๆ นั้น นักพัฒนาจะต้องสร้างพื้นที่การทำงานขึ้นมาก่อน โครงสร้างที่เหมาะสมจะแสดงดังรูปที่ 2.1

รูปที่ 2.1 แสดงตัวอย่างโครงสร้างของการเก็บข้อมูล

1) พื้นที่เก็บรหัสโปรแกรมส่วนรวม คือ ชุดข้อมูลรหัสโปรแกรมที่สมบูรณ์ที่สุดจากการรวมการรหัสโปรแกรมที่พัฒนาแล้วโดยนักพัฒนาท่านอื่น ในพื้นที่เก็บรหัสโปรแกรมส่วนบุคคล

2) พื้นที่เก็บรหัสโปรแกรมส่วนบุคคล คือ พื้นที่เก็บชุดข้อมูลรหัสโปรแกรมที่มีโครงสร้างหน้าตาในลักษณะเดียวกันกับพื้นที่เก็บรหัสโปรแกรมหลัก และนักพัฒนาไม่ควรปรับเปลี่ยนโครงสร้างของการเก็บข้อมูลให้แตกต่างกับพื้นที่เก็บรหัสโปรแกรมส่วนร่วมมากเกินไป เพราะอาจจะก่อให้เกิดผลกระทบต่อโครงสร้างของพื้นที่เก็บรหัสโปรแกรมส่วนรวม แล้วจะส่งผลต่อนักพัฒนาผู้อื่น ดังนั้นก่อนการเริ่มต้นโครงการควรมีการวางแผนเรื่องโครงสร้างไว้ก่อน

3) นอกจากพื้นที่ทั้งสองแล้วยังควรจะมีพื้นที่สำหรับรหัสโปรแกรมที่พัฒนาอย่างสมบูรณ์แล้วในระดับหนึ่ง ถึงแม้ว่าจะสามารถเก็บไว้ในพื้นที่เก็บรหัสโปรแกรมส่วนบุคคลได้ แต่เนื่องจากจำนวน Version ที่มากเกินไปอาจจะทำให้ผู้รับผิดชอบโครงงานที่มีหน้าที่ในการรวบรวมรหัสโปรแกรม เกิดปัญหาในการแยก Version ที่ใช้งานได้แล้วออกจาก Version ที่กำลังพัฒนาอยู่ ซึ่งจะทำให้เสียเวลาในการทำงาน

2.2 การสร้างและพัฒนารหัสโปรแกรมด้วย Eclipse IDE C/C++ บน Windows

2.2.1 โปรแกรม Eclipse

Eclipse เป็น IDE ที่ใช้ช่วยทำงานภายในสถานประกอบการ ด้วยคุณสมบัติที่สามารถวิเคราะห์โปรแกรม และแสดงผลชนิดตัวแปรและส่วนประกอบของโปรแกรมด้วยสีที่ต่างกัน ช่วยในการพัฒนาโปรแกรมได้เป็นอย่างดี รายละเอียดเพิ่มเติมสามารถศึกษาได้ที่ [1] และวิธีการใช้งานสามารถศึกษาได้จากภาคผนวก ก

2.3 การทดสอบการทำงานของโปรแกรมบนบอร์ดด้วย IAR Embedded Workbench

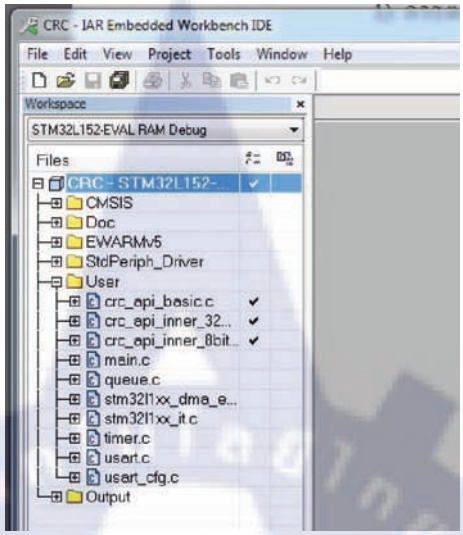
โปรแกรม IAR Embedded Workbench เป็นโปรแกรมที่ใช้พัฒนา และสร้างเฟิร์มแวร์ให้แก่ไมโครคอนโทรลเลอร์ อีกทั้งยังเป็นเครื่องมือสำหรับทดสอบการทำงานของโปรแกรมในแต่ละเฟิร์มแวร์ที่สร้างขึ้นได้ ซึ่งโปรแกรมนี้ได้สร้างและพัฒนาขึ้นโดยบริษัท IAR System โปรแกรมเป็นซอฟต์แวร์แบบมีลิขสิทธิ์ โดยวิธีการใช้งานโปรแกรมสามารถศึกษาได้จากภาคผนวก ข

2.3.1 การเชื่อมต่ออุปกรณ์ที่ใช้ดาวน์โหลดเฟิร์มแวร์และทดสอบบอร์ดด้วย ST-LINK

ST-LINK คือ ระบบที่ออกแบบขึ้นโดยบริษัท ST Microelectronics ทำหน้าที่ติดต่อระหว่างคอมพิวเตอร์และไมโครคอนโทรลเลอร์ โดยจะช่วยทำหน้าที่ในการดาวโหลดเฟิร์มแวร์ หรือควบคุมการทำงานของไมโครคอนโทรลเลอร์ในโหมด Debug ให้แก่ไมโครคอนโทรลเลอร์ที่มีสถาปัตยกรรมซึ่งเข้ากันกับระบบ ในสถานปัตยกรรมที่เราใช้งานอยู่หรือ ARM ซึ่งผลิตขั้นโดยบริษัท ST Microelectronics นั้นได้มีการออกแบบให้รอบรับการทำงานของ ST-LINK อยู่แล้ว อีกทั้งภายในบอร์ด STM32L-Discovery ก็มีวงจร ST-LINK บรรจุไว้อยู่แล้ว[2] การใช้งานจึงเป็นเพียงการเชื่อมต่อคอมพิวเตอร์ผ่าน USB Port

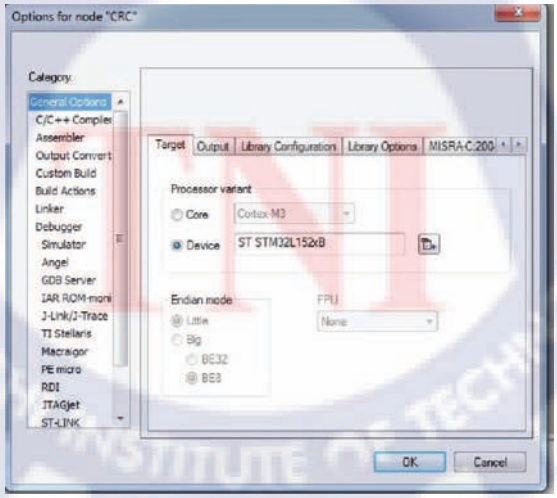
สำหรับโปรแกรม IAR Embedded Workbench ได้มีการผนวกไดร์ฟเวอร์ของ ST-LINK ไว้ตั้งแต่การติดตั้งโปรแกรม IAR Embedded Workbench ในครั้งแรกแล้ว เพราะฉะนั้นการใช้งานจึงง่ายเพียงแค่ทำตามขั้นตอนดังต่อไปนี้

1) การตั้งค่าโครงการในพื้นที่ทำงานบน IAR Embedded Workbench ให้ใช้งานระบบการงานของ ST-LINK โดยเลือกไปที่ชื่อของโครงการที่มีอยู่ในพื้นที่การทำงาน ดังตัวอย่างที่แสดงในรูปที่ 2.2 แล้วทำการเลือกคำสั่ง Project → Option



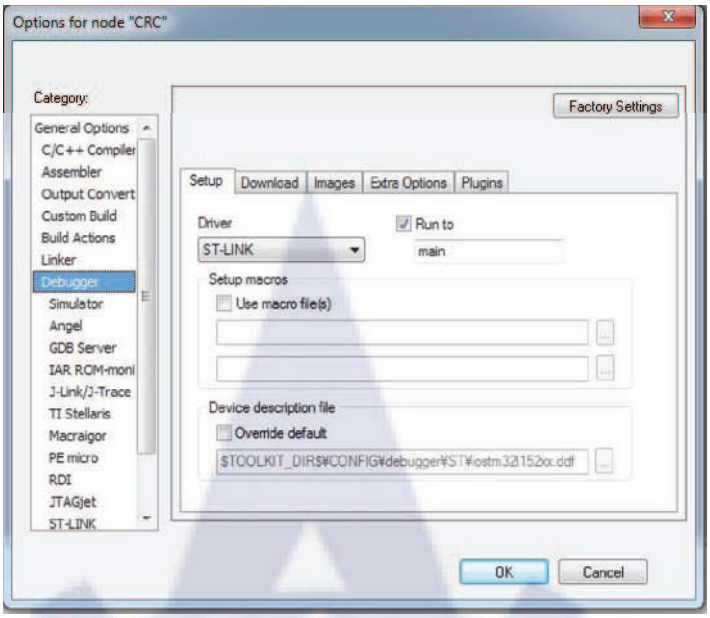
รูปที่ 2.2 แสดงตัวอย่างการเลือกชื่อของโครงการที่อยู่ในพื้นที่การทำงาน

2) หลังจากนั้นจะปรากฏหน้าต่าง ดังรูปที่ 2.3



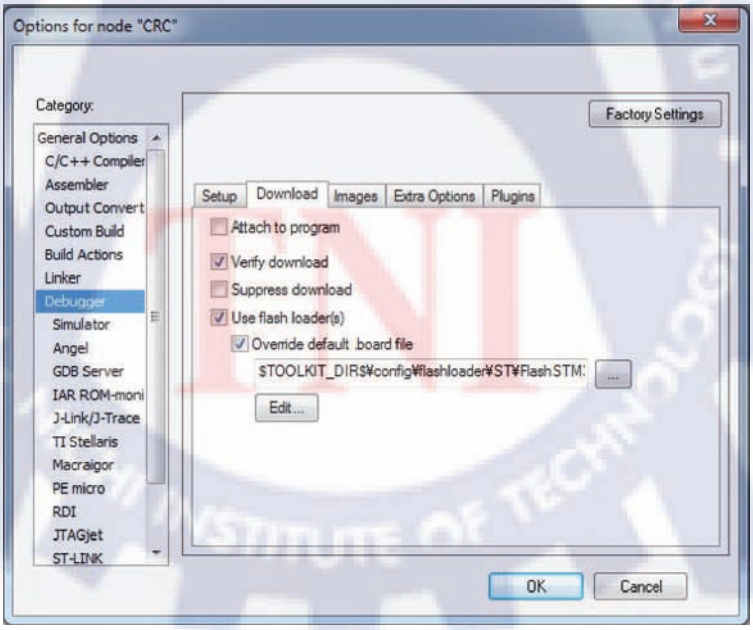
รูปที่ 2.3 หน้าต่างการปรับแต่งโครงการภายใน IAR Embedded Workbench

3) ในส่วนของ Category ให้เลือก Debugger แล้วจะปรากฏ ดังรูปที่ 2.4 แล้วให้ทำการตั้งค่าตามภาพดังกล่าว



รูปที่ 2.4 หน้าต่างตัวเลือกอุปกรณ์ที่ใช้เชื่อมต่อบอร์ดเป้าหมายใน IAR Embedded Workbench

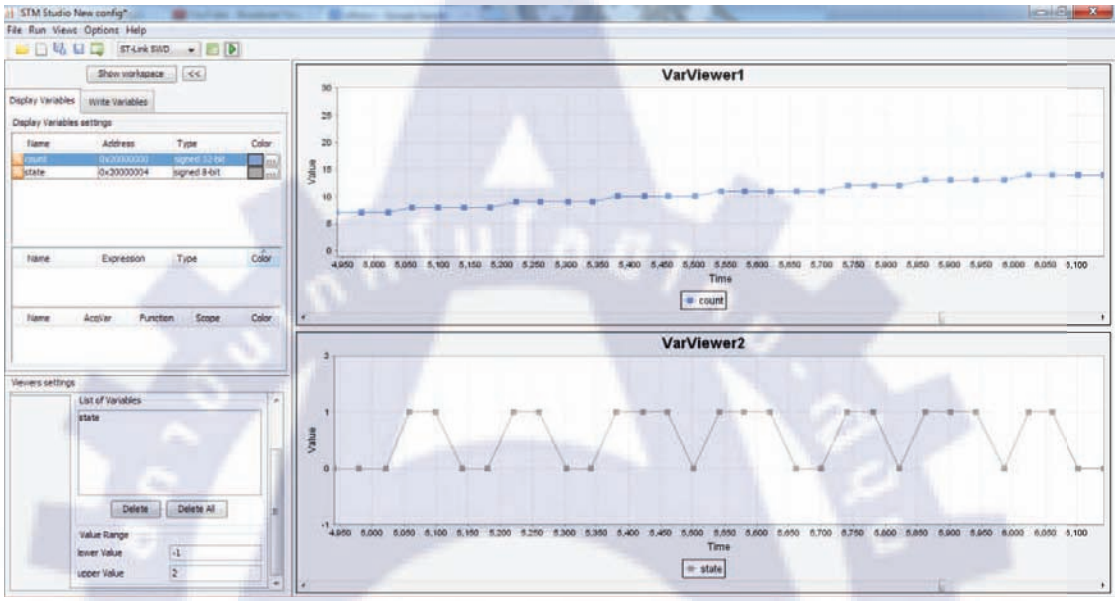
4) ต่อมาให้เปลี่ยนไปที่แท็บ Download แล้วทำการตั้งค่า ดังรูปที่ 2.5 เป็นการเสร็จสิ้นการตั้งค่า และสามารถเริ่มทำการดาวน์โหลดเฟิร์มแวร์ หรือดีบัคอุปกรณ์ได้



รูปที่ 2.5 แสดงการตั้งค่าใช้งานอุปกรณ์ที่เชื่อมต่อเพิ่มเติมใน IAR Embedded Workbench

2.4 สังเกตุการณ์ข้อมูลที่เปลี่ยนแปลงภายในไมโครคอนโทรลเลอร์โดยใช้ STM Studio

ในการตรวจสอบสถานะของ Module และค่าตัวแปรต่างๆภายในไมโครคอนโทรลเลอร์ขณะกำลังทำงานอยู่ดังรูปที่ 2.6 ทางสถานประกอบการได้แนะนำให้ใช้ STM Studio โดยมีข้อดีก็คือสามารถเก็บข้อมูลจากไมโครคอนโทรลเลอร์ได้ตลอดเวลาและไม่จำเป็นต้องเขียนโปรแกรมเพิ่มเติมใดๆ รายละเอียดเพิ่มเติมสามารถศึกษาได้จาก [3] หรือภาคผนวก ค



รูปที่ 2.6 หน้าต่างโปรแกรม STM Studio ที่กำลังทำงานอยู่

บทที่ 3
แผนงานปฏิบัติงาน

3.1 การปฏิบัติงานในสถานประกอบการ

3.1.1 แผนงานปฏิบัติงาน

ตารางที่ 3.1 แผนงานโดยพนักงานที่ปรึกษา

หัวข้องาน	เดือนที่ 1				เดือนที่ 2				เดือนที่ 3				เดือนที่ 4			
USART API desire	x	x														
USART API debug and test		x	x	x												
RTC and Calendar module study					x											
Command Line Interface study					x											
Command Line Interface studydebug and test						x	x									
CRC API							x	x	x	x	x	x				
STM-studio study												x				
Flash memory debug													x			

3.1.2 รายละเอียดงานที่นักศึกษาปฏิบัติในสถานประกอบการ

งานที่ข้าพเจ้าได้รับมอบหมายนั้นเป็นส่วนหนึ่งของโครงการ ซึ่งทางสถานประกอบการได้ดำเนินการมาในก่อนหน้านี้แล้ว แต่ยังขาดรหัสโปรแกรมบางส่วนสำหรับเป็นไดร์ฟเวอร์ให้กับเฟิร์มแวร์ของบอร์ดในโครงการนั้นๆ ซ่งส่วนของไดร์ฟเวอร์ที่อยู่ในความรับผิดชอบของข้าพเจ้ามีดังนี้

3.1.2.1 สร้างไดร์ฟเวอร์สำหรับ USART API

เพื่อใช้ควบคุมการรับส่งข้อมูลของไมโครคอนโทรลเลอร์ผ่านทางพอร์ตอนุกรม USART เพื่อไม่ให้เกิดการสูญเสียข้อมูลและปกปิดความซับซ้อนในการทำงานของ USART พร้อมกันนั้นออกแบบให้สามารถรองรับวิธีการทำงานทุกวิธีเท่าที่สามารถทำได้

3.1.2.2 RTC API test ทดสอบการทำงานของ RTC API ที่พนักงานท่านอื่นสร้างขึ้น

เพื่อทดสอบการทำงานของ RTC API ที่พนักงานบริษัทได้สร้างขึ้น ว่าสามารถทำงานได้อย่างถูกต้องหรือไม่กับอุปกรณ์จริง

3.1.2.3 CLI API แก้ไขและทดสอบให้ทำงานได้โดยเขียนต่อจากโครงสร้างที่มีอยู่แล้ว

เพื่อแก้ไขการทำงานของไดร์ฟเวอร์ CLI หรือ Command Line Interface ซึ่งเป็นเส้นทางหลักที่ติดต่อระหว่างไมโครคอนโทรลเลอร์ที่ทำงานอยู่กับคอมพิวเตอร์ที่เป็น User Interface ซึ่งใช้ลักษณะการติดต่อแบบ USART แบบสาย Serial ซึ่งจะเปลี่ยนข้อมูลจาก USB เป็น Serial Port เพื่อให้ USART สามารถรับข้อมูลที่คอมพิวเตอร์เป็นผู้ส่งได้

3.1.2.4 CRC API มีการสร้างขึ้นใหม่

เพื่อนำ CRC Calculator ภายในไมโครคอนโทรลเลอร์ STM32L152RB มาช่วยในการคำนวน CRC ของการส่งข้อมูลผ่านทาง Ethernet เนื่องจาก CRC Calculator ไม่สามารถกำหนดค่า Initial Value ได้ จึงทำให้เกิดปัญหาอย่างมากถ้าต้องการนำมาใช้คำนวนค่า CRC เพราะต้องรอให้ข้อมูลทั้งหมดมาถึงตัวไมโครคอนโทรลเลอร์ถึงจะสามารถคำนวนค่า CRC ที่ถูกต้องได้

3.1.2.5 STM Studio Guideline เอกสารการใช้งาน STM Studio

เพื่ออธิบายวิธีการใช้งานโปรแกรม STM Studio อย่างง่าย โดยเน้นที่อธิบายวิธีการเชื่อมต่อรับข้อมูลจากไมโครคอนโทรลเลอร์ผ่าน ST-LINK นำมาแสดงผลบน STM Studio และนำมาพิจารณาถึงข้อดีและข้อเสียในการใช้งานโปรแกรมดังกล่าว

3.1.2.6 Memory API test ทดสอบการทำงานของ Memory API

เพื่อแก้ไขการทำงานของไดร์ฟเวอร์ Memory API ซึ่งจะเป็นเส้นทางหลักในการติดต่อกับ EEPROM กับ Flash Memory หรือ Memory ส่วนที่ไม่ได้อยู่ในไมโครคอนโทรลเลอร์ โดยใช้ Peripheral System Controller ในการควบคุมการเข้าถึงข้อมูล อีกทั้ง API ดังกล่าวยังสามารถเพิ่มพื้นที่เก็บข้อมูลได้โดยการต่อ Memory ภายนอกแล้วสั่ง Register ขณะ Run Time ได้อีกด้วย

3.1.3 ขั้นตอนการดำเนินงานที่นักศึกษาปฏิบัติงาน

เนื่องจากทางสถานประกอบการมีนโยบายในการสร้างและพัฒนารหัสโปรแกรมให้เป็นสากล โดยจะมีเอกสารและเครื่องมือที่ต้องใช้งาน จึงทำให้ในช่วงแรกของการทำงานจะเป็นการศึกษาวิธีการออกแบบโครงเขียนโปรแกรม โดยเฉพาะอย่างยิ่งเนื่องจากงานที่ข้าพเจ้าได้รับมอบหมายจะเป็นงานประเภทศึกษาและแก้ไขรหัสโปรแกรมของผู้อื่น เพราะฉะนั้นความสามารถในการเข้าใจการทำงานรหัสโปรแกรมของพนักงานคนอื่น ในบริษัทเป็นเรื่องที่จำเป็นมาก

การดำเนินงานในส่วนของงานที่ได้รับมอบหมายนั้น ข้าพเจ้ามีแผนงานดังต่อไปนี้

1) ศึกษาหัวข้องานและจุดมุ่งหมายของการทำงาน
2) ค้นหาโปรแกรมตัวอย่าง และทำความเข้าใจ[4]
3) ออกแบบรหัสโปรแกรมให้ทำงานเข้ากับโครงสร้างที่ถูกกำหนดไว้แล้ว
4) พัฒนารหัสโปรแกรมตามการออกแบบ
5) ทดสอบกับบอร์ดอีกครั้ง เพื่อตรวจสอบการทำงาน
6) แก้ไขก่อนส่งให้แก่พนักงานผู้มอบหมาย

3.2 โครงงานที่นำเสนอ

3.2.1 แผนงานปฏิบัติงาน

ตารางที่ 3.2 แผนงานสำหรับโครงงานที่นำเสนอโดยข้าพเจ้า

หัวข้องาน	เดือนที่ 1				เดือนที่ 2				เดือนที่ 3				เดือนที่ 4			
Study CRC calculation study							x	x								
CRC API desire								x								
Reverse CRC desire(1 prototype)									x							
Study patching method									x	x						
Reverse CRC desire(2 prototype)											x					
Study 4-bit table method											x					
Reverse CRC desire(3 prototype)											x	x				
New algorithm analysis												x				
CRC and Reverse CRC calculation by new algorithm												x	x			

3.2.2 รายละเอียดโครงงาน

โครงงานนี้นำเสนอถึงการคำนวนหาค่า CRC และการกระทำทางคณิตศาสตร์ต่างๆ เพื่อช่วยเหลือการทำงานของ CRC Module ภายใน STM32L152 ให้สามารถนำมาใช้ในการทำงานจริงของสถานประกอบการได้ โดยศึกษาวิธีการที่จะเพิ่มความเร็วในการคำนวน หรือลดเวลาที่ใช้คำนวนให้เหลือน้อยที่สุดเท่าที่จะเป็นได้

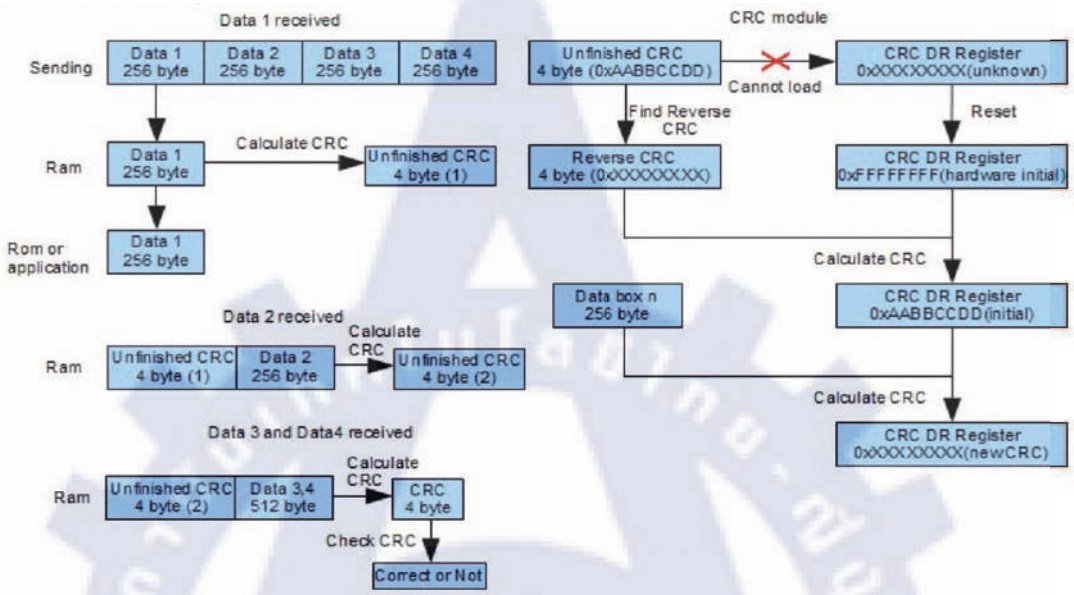
3.2.3 ขั้นตอนการดำเนินงานในโครงงาน

ในเบื้องต้นของการทำงานที่สถานประกอบการนั้น ข้าพเจ้าใช้เวลาส่วนใหญ่ในการช่วยเหลือสนับสนุนโครงการ SMART Meter ของสถานประกอบการเป็นหลัก อีกทั้งจำเป็นต้องศึกษาวิธีและหลักการดำเนินงานของสถานประกอบการทำให้ต้องเลื่อนการดำเนินการของโครงงานที่ข้าพเจ้าต้องการนำเสนอออกไป

3.2.3.1 ศึกษาหาหัวข้อวิธีการคำนวนค่า CRC ที่มีความเร็วและใช้ปริมาณหน่วยความจำลดลง

ส่วนประกอบของการคำนวน CRC สามารถศึกษาได้จากภาคผนวก จ และ [5]

ข้าพเจ้าได้รับมอบหมายให้สร้าง CRC API ขึ้นมาแล้ว โดยลักษณะโครงงานเป็นการคิดหา Algorithm วิธีการคำนวน CRC หลายแบบตามภาคผนวก จ และ [6] เพื่อนำมาเขียนเป็นหลักการให้กับบริษัทใช้ในการทำงานต่อไป โดยข้าพเจ้าเริ่มจากความต้องการของบริษัท และหลักการแก้ไขปัญหาที่ทำให้ CRC Module สามารถทำงานได้ตามที่ต้องการ ดังรูปที่ 3.1



รูปที่ 3.1 ความต้องการของบริษัท(ซ้าย) หลักการแก้ไขปัญหา(ขวา)

- โครงสร้างการคำนวน CRC ที่ทางสถานประกอบการได้ออกแบบไว้คือ สามารถตั้งค่า Initial เป็นค่าใดก็ได้ เพื่อให้สามารถคำนวนข้อมูลหายชุดในเวลาเดียวกัน และเน้นการใช้งาน CRC Module

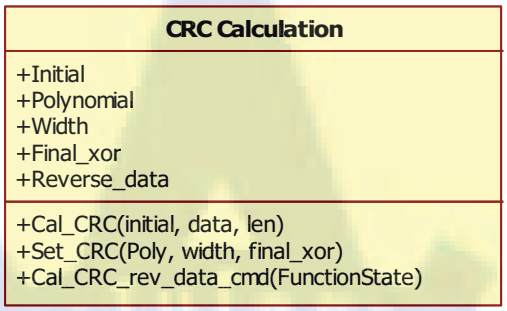
- ปัญหาคือ CRC Module ภายในไมโครคอนโทรลเลอร์ STM32L152RB ไม่สามารถเปลี่ยนค่า Initial ให้เป็นค่าอื่นนอกจาก 0xFFFFFFFF [2] จึงไม่สามารถใช้งาน CRC Module ได้หากการหาค่า CRC ยังไม่เสร็จสิ้น(ข้อมูล 1 ชุด)

- วิธีแก้ปัญหาที่ข้าพเจ้าใช้คือ การหา Reverse CRC หรือ ค่าข้อมูลที่เมื่อผ่าน CRC Module แล้วจะได้ผลลัพธ์ CRC เป็นค่าที่เราต้องการ เพื่อใช้กำหนด Initial

หลักการที่ใช้ในการคำนวน Reverse CRC สามารถศึกษาได้จาก ภาคผนวก ช และ [7]

3.2.3.2 ช่วงของการออกแบบ สามารถแบ่งออกเป็น 2 ส่วนดังนี้

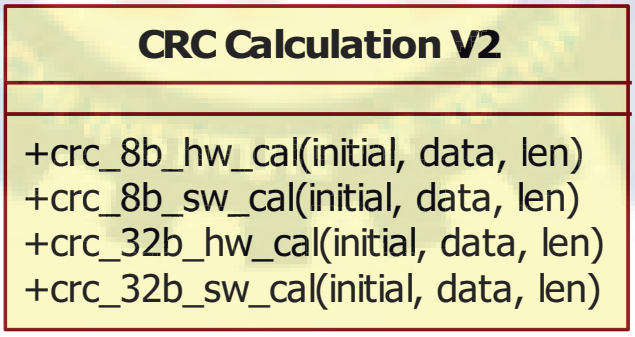
ในการออกแบบครั้งแรก ข้าพเจ้าตั้งใจจะออกแบบให้ทำงานครบทุกด้าน คือสามารถคำนวน CRC ได้ครบทุก polynomial และเลือกทางที่เร็วที่สุดให้เสมอ โดยการออกแบบดังรูปที่ 3.2



รูปที่ 3.2 โครงสร้างสำหรับ CRC Calculation ที่ได้ออกแบบไว้

แต่เนื่องจากการทำงานในลักษณะนี้จะทำให้ต้องใช้ Clock ของไมโครคอนโทรลเลอร์มากขึ้นในการเลือกสถานะที่สมควรแก่การใช้งาน โดยเฉพาะที่เมื่อต้องการเปลี่ยน polynomial ทุกครั้งก็ต้องเรียกใช้โปรแกรม Set_CRC เสมอ ทำให้สิ้นเปลืองสัญญาณนาฬิกามากยิ่งขึ้น พนักงานผู้มอบหมายงานจึงแนะนำให้ลดเหลือเพียงการคำนวน CRC32 อย่างเดียว แต่ให้เปลี่ยน Initial ให้ได้ด้วย

จึงทำการออกแบบใหม่เป็นดังรูปที่ 3.3



รูปที่ 3.3 โครงสร้างสำหรับ CRC Calculation ที่ได้ปรับแก้แล้ว

3.2.3.3 ช่วงของการเขียนโปรแกรมและพัฒนาต้นแบบ

ในช่วงนี้ข้าพเจ้าไดทำการเขียนโปรแกรมเป็นโปรแกรมสำหรับการคำนวน CRC8 CCITT, CRC16 CCITT, CRC7 SCD และ CRC32 Ethernet โดยแต่ล่ะ CRC จะมีคุณสมบัติดังนี้

(1) ในกรณีคำนวนด้วย Software สามารถเลือกการคำนวนได้ว่าต้องการจะใช้ตารางใด เช่น 8-bit และ 4 bit ซึ่งจะใช้ขนาดหน่วยความจำต่างกัน 16 เท่า แต่ถ้าเลือกไม่ใช้ตาราง ระบบจะเลือกวิธีการคำนวนด้วย หลักการใหม่ ให้ ซึ่งอาจจะช้าหรือเร็วกว่าเล็กน้อย

(2) มีโปรแกรมสำหรับคำนวน Reverse CRC ให้โดยใช้หลักการของหลักการใหม่เนื่องจากเร็วกว่าวิธีเก่า

(3) มีโปรแกรมสำหรับคำนวน Reflect CRC ด้วยเช่นกัน โดยระบบจะทำการ Reflect ทุกส่วนที่จำเป็นแล้วจะคำนวนโดยใช้วิธีปกติเมื่อได้รับคำตอบแล้วจะทำการ Reflect กลับมาให้โดยอัตโนมัติ อาจจะใช้เวลามากกว่าแบบธรรมดาเล็กน้อย ศึกษาเพิ่มเติมได้จากภาคผนวก ฉ

ตัวอย่างโปรแกรมสามารถดูได้จาก ภาคผนวก ซ

3.2.3.4 ช่วงของการทดสอบและแก้ไข

ในช่วงนี้ข้าพเจ้าจะทำการทดลองคำนวนด้วยโปรแกรมที่ข้าพเจ้าได้เขียนไว้แล้วเปรียบเทียบกับ CRC Calculator ที่สามารถหาพบได้บน Internet เพื่อยืนยันความถูกต้อง แล้วเปรียบเทียบประสิทธิภาพโดยดูในส่วนของความเร็วและปริมาณ Memory ที่ใช้ โดยตารางด้านล่างจะแสดงบางส่วนของผลการทำงานของรหัสโปรแกรมของข้าพเจ้า

บทที่ 4
ผลการดำเนินงาน การวิเคราะห์และสรุปผลต่างๆ

การดำเนินงานภายในสถานประกอบการของข้าพเจ้าได้ทำการแบ่งออกเป็น 2 ส่วนดังนี้

4.1 การปฏิบัติงานในสถานประกอบการ

งานที่ได้รับมอบหมายจากสถานประกอบการมานั้น ส่วนใหญ่จะเป็นการสร้างและออกแบบไดร์ฟเวอร์ของบอร์ดในโครงการที่ทางสถานประกอบการกำลังดำเนินอยู่เป็นหลัก ซึ่งผลการดำเนินงานที่ได้รับมอบหมายในส่วนที่ข้าพเจ้ารับผิดชอบมีดังนี้

4.1.1 การออกแบบ USART API

ผลลัพธ์ การทำงานเป็นไปตามแผนที่วางไว้และ USART API สามารถทำงานได้ตามที่สถานประกอบการต้องการ

4.1.2 การทดสอบ RTC API

ผลลัพธ์ การทำงานเป็นไปตามแผนที่วางไว้และ RTC API มีข้อผิดพลาดเล็กน้อยในส่วนของการเปลี่ยนแปลงเวลา ซึ่งเกิดจาก RTC API ไม่ได้เข้าไปแก้ไขในส่วนของ System Controller ที่อยู่ใน System API จึงได้แจ้งต่อพนักงานผู้ดูแล

4.1.3 การแก้ไข CLI API

ผลลัพธ์ การทำงานเป็นไปตามแผนที่วางไว้และ การทำงานของ CLI API ที่ผ่านการแก้ไขแล้ว สามารถติดต่อสื่อสารระหว่างไมโครคอนโทรลเลอร์ STM32L152RB และ Computer ได้

4.1.4 การสร้าง CRC API

ผลลัพธ์ การทำงานเป็นไปตามแผนที่วางไว้และสามารถสร้าง CRC API ที่ตรงต่อความต้องการของสถานประกอบการได้

4.1.5 การเขียนเอกสาร STM Studio Guideline

ผลลัพธ์ การทำงานเป็นไปตามแผนที่วางไว้

4.1.6 การทดสอบ Memory API test

ผลลัพธ์ การทำงานเป็นไปตามแผนที่วางไว้และพบปัญหาหลายอย่าง ซึ่งได้แจ้งต่อพนักงานผู้ดูแลแล้ว

4.2 โครงงานที่นำเสนอ

4.2.1 ขั้นตอนและผลการดำเนินงาน

ผลลัพธ์ การทำงานทุกอย่างสำเร็จลุล่วงและเป็นไปตามแผน โดยเปรียบเทียบประสิทธิภายได้ตามตารางที่ 4.1

ตารางที่ 4.1 เปรียบเทียบเวลาที่ใช้ของแต่ล่ะวิธีในการคำนวนหา CRC

Data 512 byte	clocks used	memory used(byte)
CRC module(Propose Algor	น้อยมาก	0
CRC module(4 bit Reverse)	ปานกลาง	64
CRC module(8 bit Reverse)	มาก	1024
Propose Algorithm	มาก	0
4-bit Table	มาก	64
8-bit Table	ค่อนข้างมาก	1024

4.2.1.1 ผลการวิเคราะห์ข้อมูล

ข้อได้เปรียบของ Algorithm ที่นำเสนอมีดังนี้

- ไม่จำเป็นต้องใช้หน่วยความจำ
- สามารถทำงานได้เร็วกว่าวิธีการคำนวนทางคณิตศาสตร์
- ไม่ขึ้นอยู่กับ Hardware
- สามารถหา Reverse CRC ได้เร็วกว่าวิธีใช้ตาราง

เมื่อได้ข้อสรุปดังกล่าว ทางสถานประกอบการจึงมอบหมายให้ข้าพเจ้าสร้าง CRC API โดยนำวิธีการหา Reverse CRC ด้วย Algorithm ใหม่และนำไปทดสอบความถูกต้อง ซึ่งจากการทดสอบก็ได้พบว่า CRC ทั้งหมดที่คำนวนได้มีความถูกต้อง และ Reverse CRC ทุกค่าก็สามารถทำงานได้อย่างถูกต้อง สามารถศึกษา Algorithm ใหม่ได้จาก ภาคผนวก จ และ [7]

บทที่ 5
บทสรุปและข้อเสนอแนะ

5.1 การปฏิบัติงานในสถานประกอบการ

5.1.1 สรุปผลการดำเนินงาน

จากการดำเนินงานที่ได้รับมอบหมายจากสถานประกอบการ พบว่าการดำเนินงานต่างๆนั้น ข้าพเจ้าจะต้องมีความรู้ความเข้าใจในระบบโดยรวมก่อนเสมอ ก่อนที่นักพัฒนาแต่ล่ะคนจะสามารถเริ่มดำเนินการสร้างโมดูล หรือโปรแกรมต่างๆขึ้นมาได้นั้น จำเป็นต้องมีแหล่งอ้างอิงในการสร้างชุดโมดูลดังกล่าวเสมอ เพื่อเป็นการสร้างความน่าเชื่อถือ อีกทั้งในการออกแบบโมดูลหรือไดร์ฟเวอร์ต่างๆนั้นจะต้องกระทำอย่างชาญฉลาด การออกแบบและสร้างโมดูลควรจะมีคำอธิบายการทำงานภายในโครงสร้างของโปรแกรม

5.2 โครงงานที่นำเสนอ

5.2.1 สรุปผลการดำเนินงาน

การดำเนินงานในโครงงานนั้น มักจะมีปัญหาอยู่ในส่วนของความสามารถทางด้านคณิตศาสตร์ของข้าพเจ้า เนื่องจากข้าพเจ้าค่อนข้างจะด้อยในด้านนี้ แต่หลักการคำนวนทั้งหมดนั้นใช้คณิตศาสตร์ในการอธิบายทั้งสิ้น อีกทั้งบทความที่อธิบายถึงเรื่องที่ข้าพเจ้าศึกษาก็หาได้น้อยเต็มมาก โดยเฉพาะอย่างยิ่งในหลักการที่ข้าพเจ้าได้เขียนไว้ในเอกสารฉบับนี้ ซึ่งข้าพเจ้าไม่สามารถหาข้อมูลเกี่ยวกับหลักการนี้จาก Internet ได้เลยข้าพเจ้าจึงจำเป็นต้องคิดตั้งแต่หลักการพื้นฐาน ในที่สุดจึงสามารถพิสูจน์ทางคณิตศาสตร์ได้ว่าสมการดังกล่าวเป็นจริง แล้วได้นำหลักการดังกล่าวมาช่วยในการคำนวนหา Reverse CRC ที่เป็นเป้าหมายหลักที่สถานประกอบการมอบหมายให้แก่ข้าพเจ้า

เอกสารอ้างอิง

1. The Eclipse Foundation, **About the Eclipse Foundation**, Available : http://www.eclipse.org/org [2012, September 18]

2. STMicroelectronics, 2011, **STM32L151xx/STM32L152xx/STM32L162xx** [online], Available : http://www.st.com/internet/com/TECHNICAL_RESOURCES/TECHNICAL_LITERATURE/REFERENCE_MANUAL/CD00240193.pdf [2012, June 18]

3. STMicroelectronics, 2011, **STM Studio**, Available : http://www.st.com/internet/com/TECHNICAL_RESOURCES/TECHNICAL_LITERATURE/USER_MANUAL/CD00291015.pdf

4. STMicroelectronics, 2011, STM32L-Discovery firmware package [online], Available : http://www.st.com/jp/com/SOFTWARE_RESOURCES/SW_COMPONENT/FIRMWARE/stm32l-discovery_fw_pack.zip [2012, June 30]

5. CRC Calculator, 2003, by **Sven Reifegerste,** Available : http://www.zorc.breitbandkatze.de/crc.html [2012, July 30]

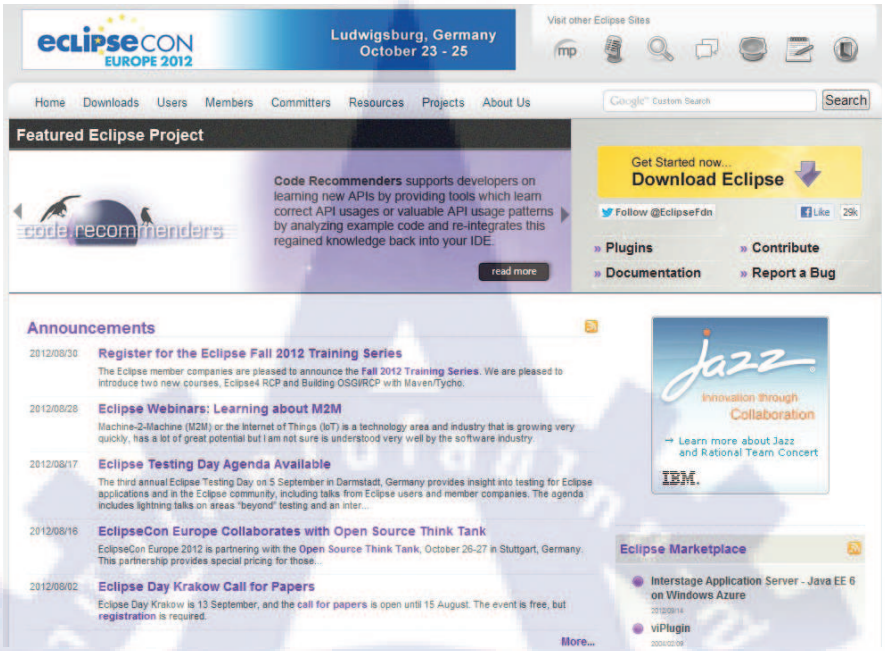
6. Mathematics of CRC, Wiki, Available : http://en.wikipedia.org/wiki/Mathematics_of_CRC [2012, September 18]

7. Calculating Reverse CRC, Daniel Vik, Available : http://www.danielvik.com/2010/10/calculating-reverse-crc.html [2012, September 18]

ภาคผนวก

ก. การใช้งาน Eclipse ขั้นพื้นฐาน

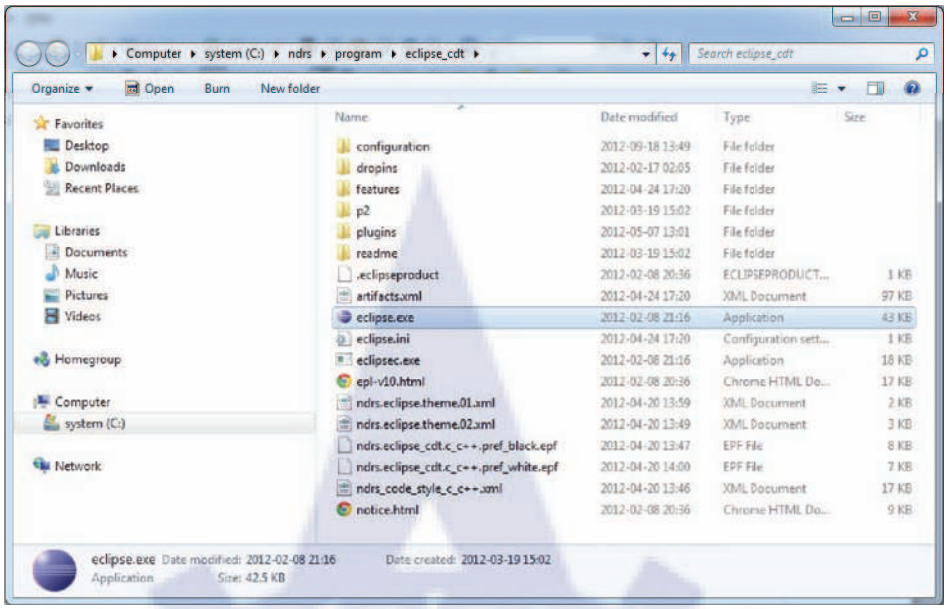
ก.1 การติดตั้ง : ดาวน์โหลดเครื่องมือมาจากเว็บไซต์ http://www.eclipse.org/ ดังรูปที่ ก.1



รูปที่ ก.1 ตัวอย่างหน้าเว็บไซต์ของ Eclipse [2012, September 19]

ชุดเครื่องมือที่ Eclipse มีอยู่นั้นจะแบ่งเป็นการพัฒนาซอฟต์แวร์ตามภาษาของโปรแกรม หรือวิธีการเขียนซอฟต์แวร์ที่แตกต่างกัน เช่น ภาษา C/C++, ภาษา Java หรือ การสร้างเว็บแอพพลิเคชั่น เป็นต้น ซึ่งในเอกสารฉบับนี้ รวมถึงเครื่องคอมพิวเตอร์ของสถานประกอบการในแผนก Embedded System นั้นใช้คือ Eclipse IDE for C/C++ Developers เพื่อใช้เป็นเครื่องมือสำหรับสร้างเฟิร์มแวร์ในภาษา C และ C++ เป็นหลัก อีกทั้งเครื่องมือดังกล่าวสามารถเรียกดู Text file ชนิดอื่น ได้ เช่น รหัสโปรแกรมในภาษาแอสเซมบลี เป็นต้น

ดังนั้นเครื่องมือที่ต้องดาวน์โหลดมา คือ Eclipse IDE for C/C++ Developers ซึ่งแสดงอยู่ที่ http://www.eclipse.org/downloads/ สามารถดาวน์โหลดแล้วจัดเก็บเป็นไฟล์เอกสารบีบอัดชนิด ZIP ซึ่งภายในแสดงโครงสร้างของโฟลเดอร์ได้ ดังรูปที่ ก.2 ให้นักพัฒนาแต่ละคนทำการ Extract all files ไปยังโฟลเดอร์ปลายทางใดก็ได้ เนื่องจากเครื่องมือชนิดนี้สามารถเรียกใช้งานผ่าน eclipse.exe ได้โดยไม่ต้องทำการติดตั้งอย่างสมบูรณ์ ซึ่ง eclipse.exe จะแสดงอยู่ภายในโฟลเดอร์ที่ติดตั้งแล้ว



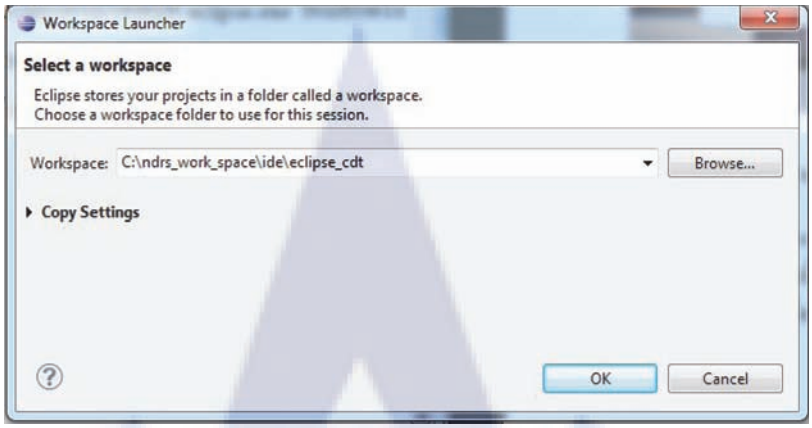
รูปที่ ก.2 แสดงข้อมูลที่อยู่ภายในเอกสารอัดบีบชนิด ZIP ของ Eclipse

เมื่อทำการเปิด eclipse.exe เพื่อเริ่มต้นการทำงานในการสร้างโครงสร้างสำหรับการพัฒนาซอฟต์แวร์ภาษาซีแล้วจะมีการแสดงหน้าต่างขึ้นมา ดังรูปที่ ก.3 เพื่อแจ้งให้ผู้ใช้ทราบถึงปริมาณทรัพยากรภายในเครื่อง

รูปที่ ก.3 หน้าต่างแสดงการเริ่มต้นใช้งาน Eclipse

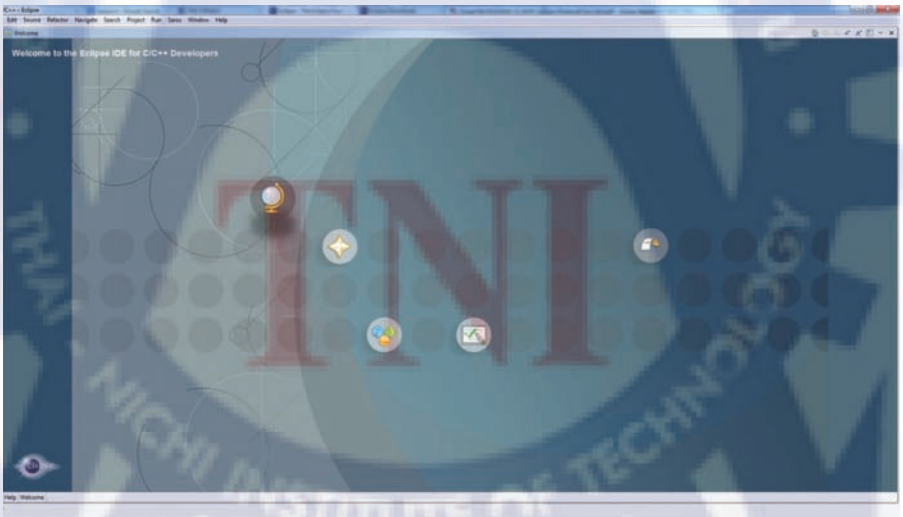
หากเกิดปัญหาขัดข้องขึ้นขณะเปิดใช้งาน ในกรณีที่เครื่องที่เปิดใช้งานนั้นยังไม่ได้ทำการติดตั้ง Java Virture Machine ให้นักพัฒนาแต่ละคนทำการติดตั้งจาก http://www.java.com/en/ โดยเลือก Java Virture Machine ให้ตรงกับสถาปัตยกรรมของ Operating System ที่ใช้งานอยู่ ตัวอย่างหากเราใช้งาน Windows 7 แบบ 64 bit ก็ควรจะโหลด Java Virture Machine ที่ใช้ทำงานกับ Windows 7 แบบ 64 bit จากนั้นให้เริ่มโปรแกรมโดยเปิด eclipse.exe ใหม่อีกครั้ง ซึ่งจะปรากฏ

หน้าต่างขึ้นดังรูปที่ ก.4 เพื่อให้นักพัฒนาแต่ล่ะคนได้เลือกตำแหน่งของโครงการหลักสำหรับอ้างอิงหรือจัดเก็บข้อมูลรหัสโปรแกรมไว้เป็นแต่ล่ะหน่วยย่อย



รูปที่ ก.4 หน้าต่างแสดงที่ตั้งโฟลเดอร์ของโครงการ โดยโปรแกรม Eclipse

ต่อมาโปรแกรม Eclipse จะแสดงหน้าต่างดังรูปที่ ก.5 เพื่อแสดงข้อความต้อนรับการเข้าสู่การใช้งานครั้งแรก ซึ่งหน้าต่างนี้จะประกอบไปด้วยการเชื่อมโยงเอกสารไปยังคู่มือประกอบการใช้งาน ส่วนแสดงการใช้เครื่องมือที่ Eclipse ได้จัดเตรียมไว้ เป็นต้น

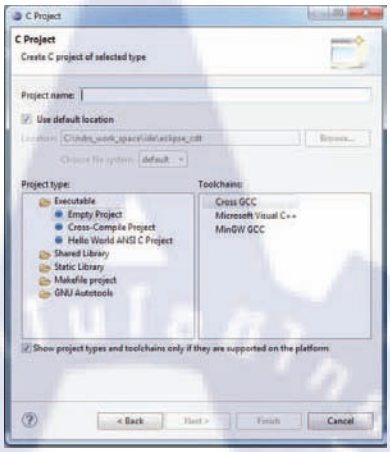


รูปที่ ก.5 หน้าต่างต้อนรับของโปรแกรม Eclipse

5) การสร้างโครงการย่อยหรือการนำเข้าโครงการย่อยที่มีอยู่แล้ว

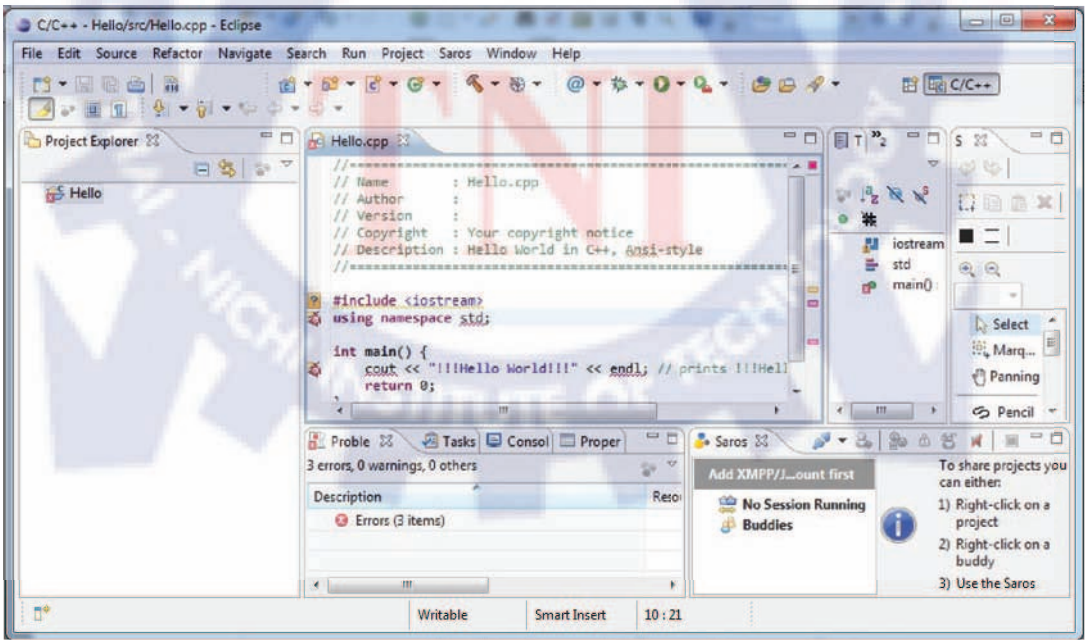
(1) กรณีที่นักพัฒนาต้องการสร้างโครงการย่อยใหม่ภายใต้ที่ตั้งของโครงการหลักที่ได้กำหนดไว้เบื้องต้น ดังนี้

เริ่มต้นจากการสร้างโครงการใหม่โดยเลือก File → Project... จะปรากฏหน้าต่าง New Project ให้เลือกลักษณะของโครงการที่จะสร้างขึ้นใน Eclipse ในที่นี้ให้นักพัฒนาเลือกไปที่ General → Project แล้วเลือก Next เพื่อทำการตั้งชื่อของโครงการนี้ และเลือกที่ตั้งของโครงนี้ดังที่แสดงใน รูปที่ ก.6 ในกรณีตัวอย่างนี้จะเลือกที่ตั้งของโครงการเป็นแบบ default location โดยการเลือกตัวเลือก Use default location ใต้ชื่อโครงการ



รูปที่ ก.6 หน้าต่างตั้งชื่อและกำหนดที่ตั้งโครงการของ Eclipse

ขั้นตอนต่อไป ให้นักพัฒนาเลือก Window → Show View → C/C++ Project หน้าต่างของ Eclipse จะแสดงดังรูปที่ ก.7 ซึ่งจะเป็นการเสร็จสิ้นการสร้างโครงการขั้นต้น



รูปที่ ก.7 แสดงหน้าต่างภายหลังการสร้างโครงการใน Eclipse

(2) กรณีที่นักพัฒนาต้องการนำโครงการย่อยที่มีอยู่เข้าในโครงการหลัก ดังนี้

ภายหลังนักพัฒนาได้กำหนดที่ตั้งของโครงการหลักแล้ว ในนักพัฒนาเลือก File → Import แล้วเลือกไปยังที่ตั้งของโครงการย่อยที่มีอยู่ โดยต้องเป็นโครงการที่เคยถูกสร้างขึ้นโดยใช้ Eclipse ไว้ก่อนหน้าแล้ว

จากนั้นหน้าต่าง Import จะแสดงขึ้นมาให้นักพัฒนาเลือก General → Existing Project into Workspace แล้วทำการเลือกที่ตั้งปลายทางที่ต้องการได้

6) การสร้างและพัฒนารหัสโปรแกรม

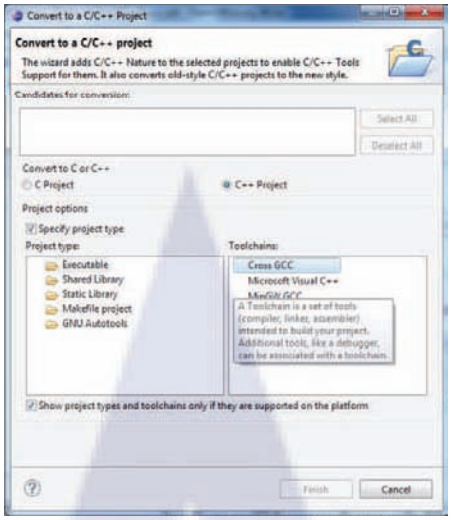
เนื่องจากโครงการที่ออกแบบนั้นใช้ภาษาซีในการพัฒนาเป็นหลัก ในเบื้องต้นนักพัฒนาที่มีโครงการย่อยภายในโครงการหลักที่ยังไม่ได้แปลงโครงการย่อยแบบทั่วไปเป็นโครงการภาษาซี สามารถกระทำได้ดังต่อไปนี้

(1) การแปลงโครงการทั่วไปให้เป็นโครงการภาษาซี

ให้นักพัฒนาเลือก File → New → Convert to a C/C++ Project แล้วจะแสดงหน้าต่างดังรูปที่ ก.8 ให้นักพัฒนาเลือกโครงการย่อยที่ต้องการซึ่งแสดงอยู่ในหัวข้อ Candidates for conversion แล้วจึงเลือก Finish

ในส่วนของ Toolchains จะแสดงชุดคอมไพล์เลอร์ที่ติดตั้งไว้ในเครื่องคอมพิวเตอร์ของนักพัฒนา และสามารถนำมาใช้งานร่วมกับ Eclipse ได้ซึ่งนักพัฒนาสามารถเลือกใช้งานคอมไพล์เลอร์ที่แสดงอยู่ก็ได้ตามความต้องการ แต่ในที่นี้จะใช้งานชุดคอมไพร์เลอร์ MinGW เป็นหลัก

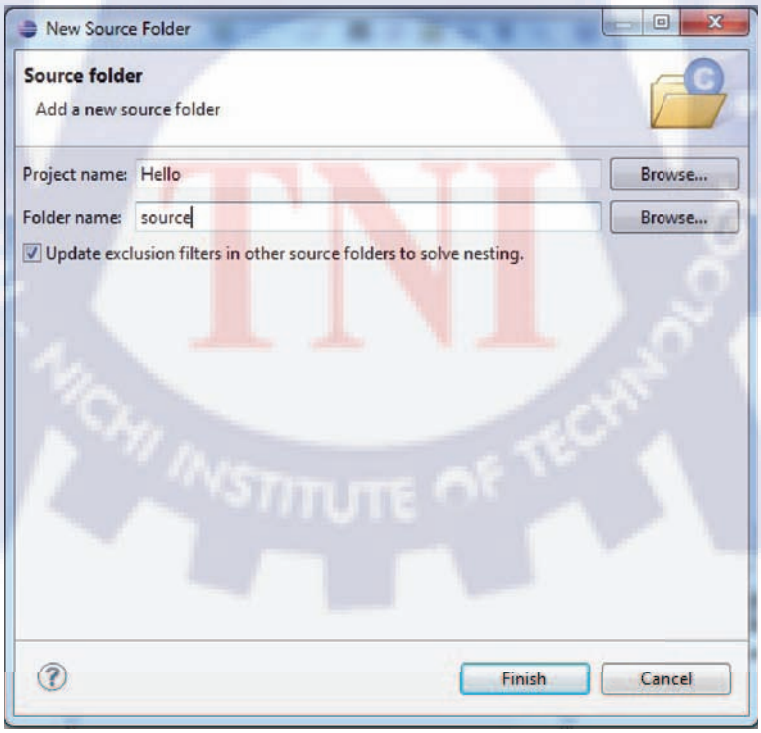
ในกรณีที่เครื่องคอมพิวเตอร์ยังไม่ได้ทำการติดตั้งคอมไพล์เลอร์ ให้นักพัฒนาทำการติดตั้งคอมไพล์เลอร์ที่สามาใช้งานร่วมกับ Eclipse ได้ โดยในที่นี้จะแนะนำให้ใช้ MinGW ซึ่งสามารถค้นหาตัวติดตั้ง พร้อมทั้งคู่มือการติดตั้งได้ที่เว็บไซต์ http://www.mingw.org/



รูปที่ ก.8 แสดงหน้าต่างการแปลงโครงการใน Eclipse

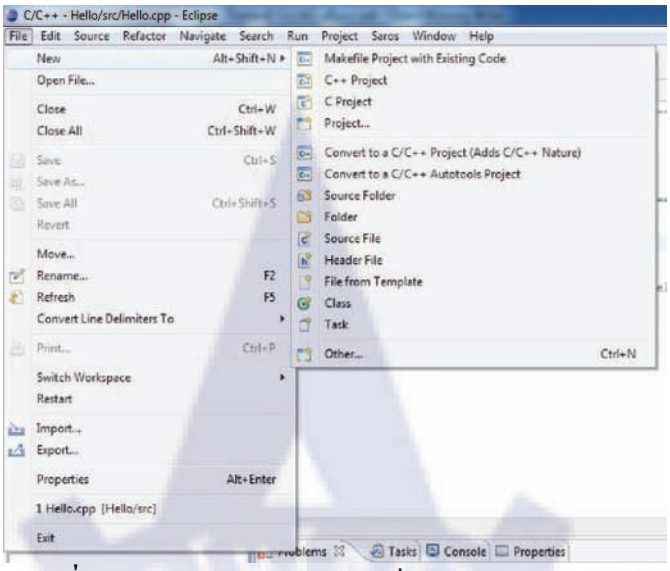
(2) การเพิ่มรหัสโปรแกรม และโฟลเดอร์ภายในโครงการ

นักพัฒนาสามารถเพิ่มลดโฟลเดอร์สำหรับจัดเก็บรหัสโปรแกรมภาษาซีโดยเลือก File → New → Source Folder แล้วจะแสดงหน้าต่างดังรูปที่ ก.9 เพื่อระบุโครงการที่จะเพิ่มโฟลเดอร์ และตั้งชื่อให้แก่โฟลเดอร์



รูปที่ ก.9 แสดงหน้าต่างการเพิ่มโฟลเดอร์สำหรับจัดเก็บรหัสโปรแกรมใน Eclipse

ทั้งนี้นักพัฒนาสามารถเพิ่มเอกสารที่แสดง ดังรูปที่ ก.10 โดยเลือก File → New



รูปที่ ก.10 แสดงตัวเลือกในการเพิ่มเอกสารของ Eclipse

ก.2 การใช้งานร่วมกับส่วนเสริมของ Eclipse IDE C/C++

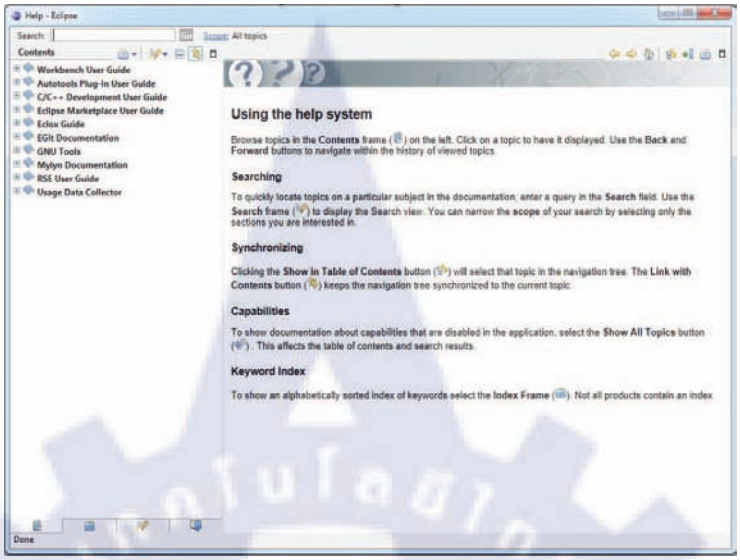
1) ส่วนเสริม Saros

เป็นส่วนเสริมที่ใช้สำหรับการพัฒนารหัสโปรแกรมร่วมกับทีมงานในเวลาเดียวกัน โดยนักพัฒนาสามารถทำงานร่วมกันในโปรเจคเดียวกันได้แบบระยะไกล และสามารถส่งข้อความ สนทนาภายในทีมได้อีกด้วย ทั้งนี้ส่วนเสริม Saros มีจุดประสงค์เพื่อให้นักพัฒนาทำการสร้างและพัฒนารหัสโปรแกรมร่วมกันในเวลาเดียวกัน สามารถหาข้อมูลเกี่ยวกับส่วนเสริม Saros และตัวติดตั้งได้ที่เว็ปไซต์ http://www.saros-project.org/

2) ส่วนเสริม Eclox

เป็นส่วนเสริมที่ช่วยสร้างเอกสารจากเนื้อหาคำอธิบายในรหัสโปรแกรมขณะสร้างหรือพัฒนาขึ้นได้ โดยคำอธิบาย ที่ระบุอยู่ในรหัสโปรแกรมนั้นต้องมีรูปแบบโครงสร้างที่ตรงกับข้อกำหนดของ Doxygen ซึ่งสามารถศึกษาวิธีการหรือรูปแบบการเขียนได้ที่ http://www.stack.nl/~dimitri/doxygen/ ส่วนเสริม Eclox สารถค้นหาตัวติดตั้งและข้อมูลได้ที่ http://home.gna.org/eclox/

รายละเอียด ซึ่งเกี่ยวกับการใช้งาน Eclipse นั้นนักพัฒนาสามารถเปิดคู่มือการใช้งานได้จาก Help → Help Contents ซึ่งคู่มือจะมีลักษณะ ดังรูปที่ ก.11



รูปที่ ก.11 ตัวอย่างหน้าต่างคู่มือการใช้งาน Eclipse

ข. การใช้งาน IAR ขั้นพื้นฐาน

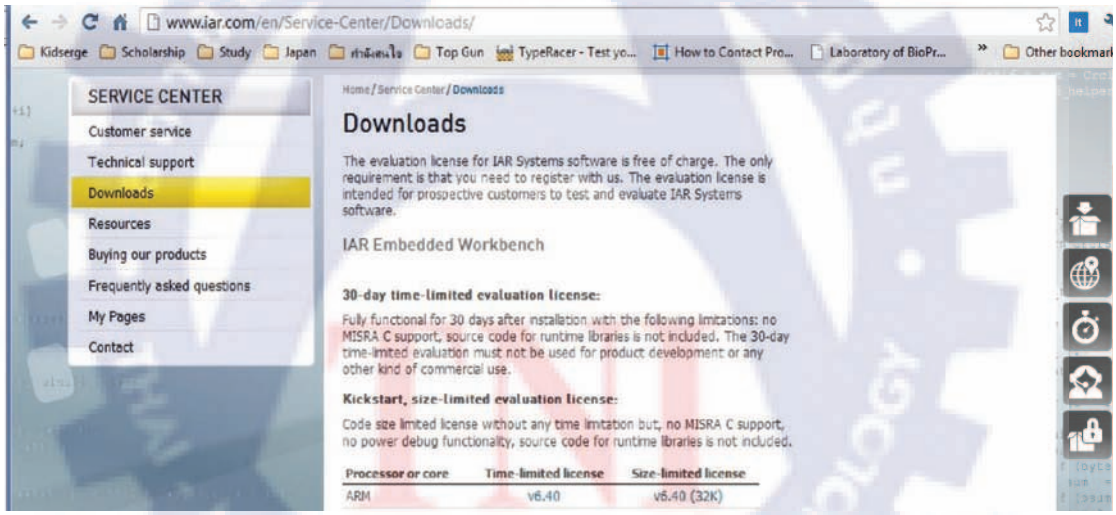
ข.1 การติดตั้ง

ถึงแม้ว่าโปรแกรมนี้จะเป็นโปรแกรมในทางธุรกรรม แต่ทางบริษัท IAR ได้ออกเงื่อนไขในการนำโปรแกรมนี้มาใช้งานโดยไม่ต้องเสียค่าใช้จ่าย ได้สองกรณี คือ

1) การใช้งานแบบจำกัดขนาดของการพัฒนาโปรแกรมที่จะทำการดาวน์โหลด

2) การใช้งานแบบจำกัดเวลาโดยไม่จำกัดขนาดของโปรแกรมที่จะทำการดาวน์โหลด

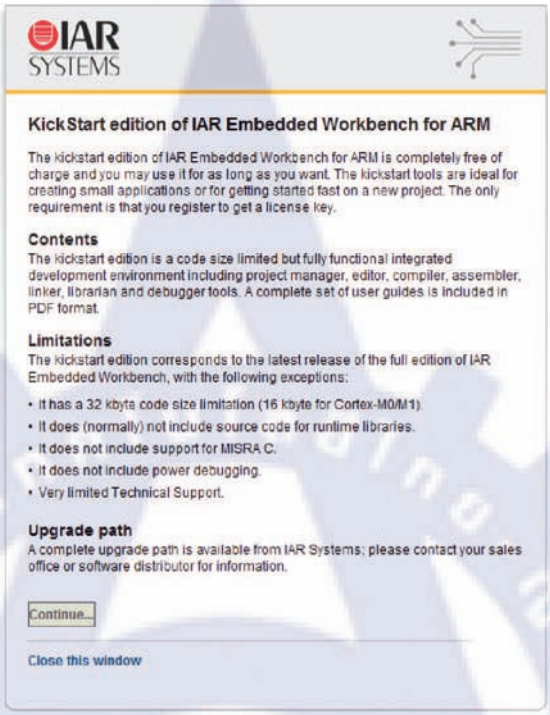
โปรแกรม IAR Embedded Workbench โหลดได้จากเว็ปไซต์ http://www.iar.com/ ในส่วนของหน้าดาวน์โหลด ดังที่แสดงตัวอย่างหน้าเว็ปดาวน์โหลดในรูปที่ ข.1



รูปที่ ข.1 แสดงส่วนหนึ่งในหน้าดาวน์โหลดของเว็บ IAR

เนื่องจากข้าพเจ้าได้ดำเนินการสร้างและออกแบบไดร์ฟเวอร์และใช้สถาปัตยกรรมของไมโครคอนโทรลเลอร์ของ ARM เช่น Core M3 และ M4 เป็นหลัก ซึ่งไดร์ฟเวอร์ ที่สร้างขึ้นนั้นมีขนาดไม่ใหญ่มากเกินข้อกำหนดที่ทางบริษัท IAR กำหนดไว้ในโปรแกรมรุ่นจำกัดขนาด (Kickstart edition) ในกรณีที่นักพัฒนาต้องการนำโปรแกรมนี้มาใช้งานโดยไม่เสียค่าใช้จ่ายและไม่มีเวลาที่จำกัดการใช้งานด้วย ดังนั้นข้าพเจ้าจำทำการอธิบายในส่วนของวิธีดาวน์โหลดโปรแกรมนี้มาใช้งานดังต่อไปนี้

1) เลือกเวอร์ชั่นของโปรแกรมในบรรทัดของสถาปัตยกรรมของไมโครคอนโทรลเลอร์ที่จะทำการพัฒนา และเลือกในคอลัมน์ Kickstart edition หลังจากนั้นจะปรากฏหน้าต่างถัดไป ดังรูปที่ ข.2 ให้เลือก Continue...



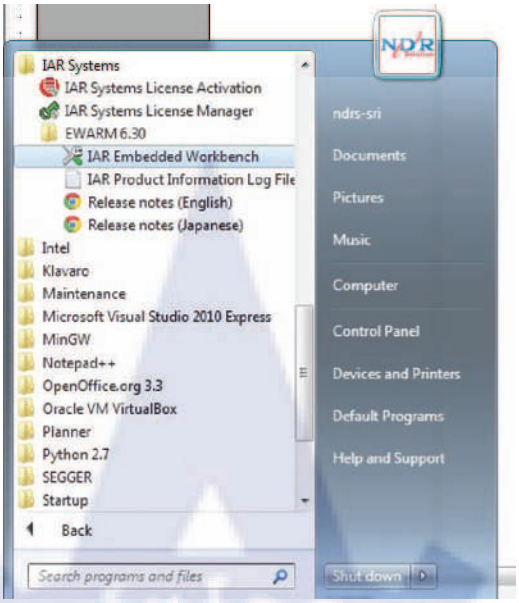
รูปที่ ข.2 แสดงหน้าต่างแจ้งผู้ใช้เกี่ยวกับกฎการใช้งานโปรแกรม IAR

2) ให้นักพัฒนาทำการกรอกข้อมูลตามที่บริษัท IAR ต้องการ และใส่อีเมลที่ใช้งานจริงเพื่อรับลิ้งค์สำหรับเชื่อมโยงไปยังหน้าดาวน์โหลด ซึ่งหน้าดาวน์โหลดนั้นจะแสดงภาษาของโปรแกรมที่รองรับ จากนั้นให้นักพัฒนาทำการดาวน์โหลดตัวติดตั้ง และบันทึกข้อมูลที่จำเป็นต่อการติดตั้ง ซึ่งแสดงอยู่ในหน้าต่างนี้ด้วย

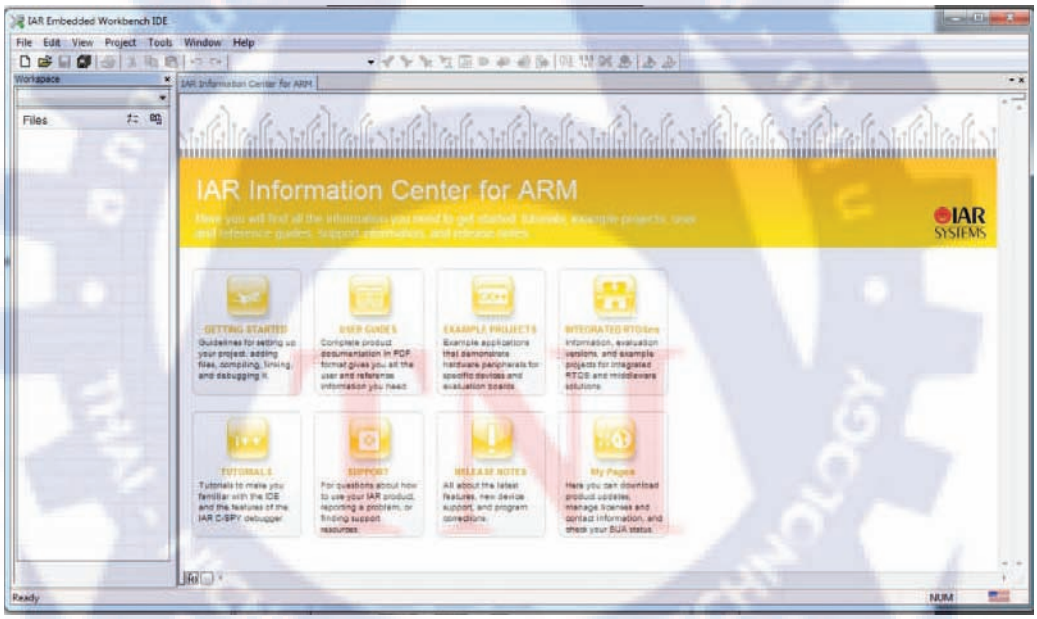
3) เมื่อทำการดาวน์โหลดเสร็จสิ้นให้นักพัฒนาเริ่มการติดตั้งและนำเอาข้อมูลที่ได้บันทึกไว้ในข้างต้นมาใช้ประกอบการติดตั้งด้วย

ข.2 หน้าต่างที่แสดงบน IAR Embedded Workbench

ภายหลังที่นักพัฒนาได้ทำการติดตั้งโปรแกรมเสร็จสิ้น ให้นักพัฒนาเริ่มต้นการทำงานของโปรแกรมนี้ดังรูปที่ ข.3



รูปที่ ข.3 แสดงลิ้งค์เชื่อมโยงโปรแกรม IAR Embedded Workbench



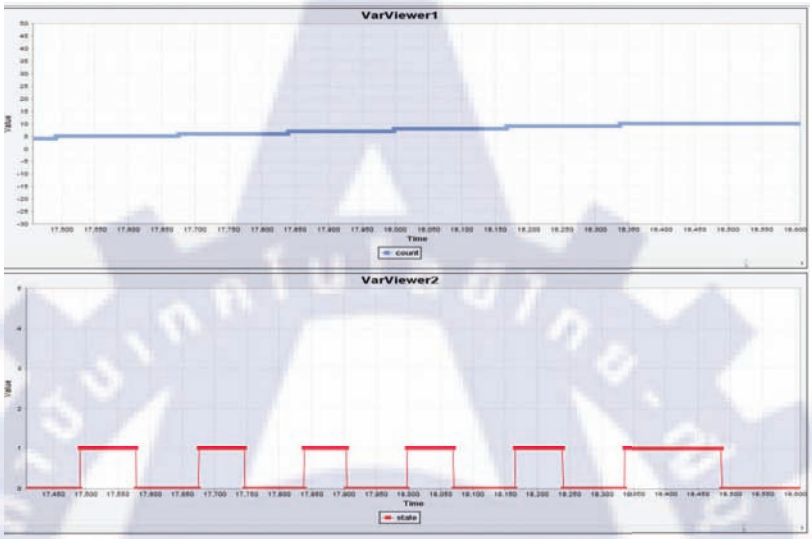
รูปที่ ข.4 แสดงหน้าต่างของโปรแกรม IAR Embedded Workbench เมื่อเปิดใช้งานครั้งแรก

จากรูปที่ ข.4 เป็นหน้าแสดงการต้อนรับต่อผู้ใช้งานโดยโปรแกรม IAR Embedded Workbench ซึ่งนักพัฒนาสามารถเลือกศึกษาการใช้งาน หรือเลือกโครงการที่เป็นตัวอย่างได้จากหน้าต่างนี้ได้โดยง่ายดาย แต่ถ้าหากหน้าต่างนี้ไม่ปรากฏขึ้น ให้นักพัฒนาเลือกคำสั่ง Help → Informarion Center ทั้งนี้วิธีการใช้งานเบื้องต้น ข้าพเจ้าขอไม่อธิบายในรายงานฉบับนี้ เนื่องจากผู้ใช้งานสามารถศึกษาได้จากคำสั่ง Help

ค. การใช้งาน IAR ขั้นพื้นฐาน

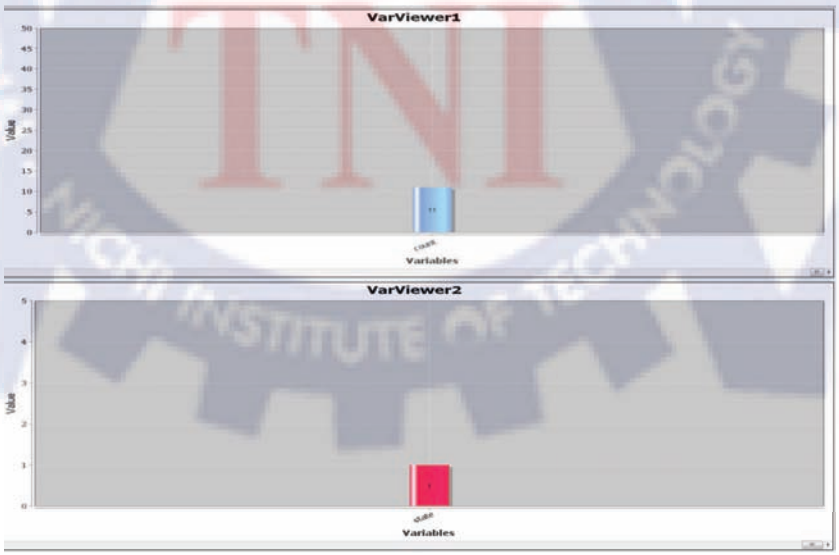
STM Studio เป็นโปรแกรมของทาง STM สามารถนำมาใช้ช่วยในการแสดงตัวแปรที่อยู่ในไมโครคอนโทรลเลอร์ซึ่งเชื่อมต่ออยู่กับคอมพิวเตอร์ผ่าน ST-LINK หรือ RLINK โดยตัวแปรที่สามารถนำมาแสดงได้ต้องเป็น public variable ของโปรแกรม การแสดงผลจะได้ 3 รูปแบบคือ

1) Curve นำค่าที่ได้มาพล็อตตามเวลาจากซ้ายไปขวาดังรูปที่ ค.1



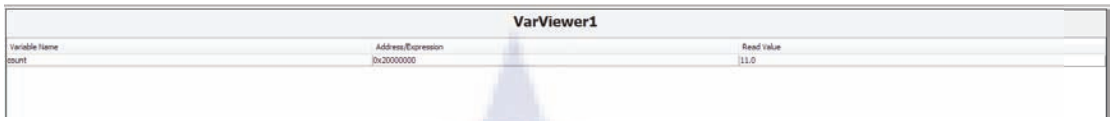
รูปที่ ค.1 การแสดงผลแบบ Curve

2) Bar Graph นำข้อมูลมาทำเป็นแผนภูมิแท่งตามเวลาจริงดังรูปที่ ค.2



รูปที่ ค.2 การแสดงผลแบบ Bar Graph

3) Table เป็นการแสดงข้อมูลเป็นบรรทัด โดยแต่ละบรรทัดจะมีชื่อข้อมูล ตำแหน่งและค่าของข้อมูล ดังรูปที่ ค.3



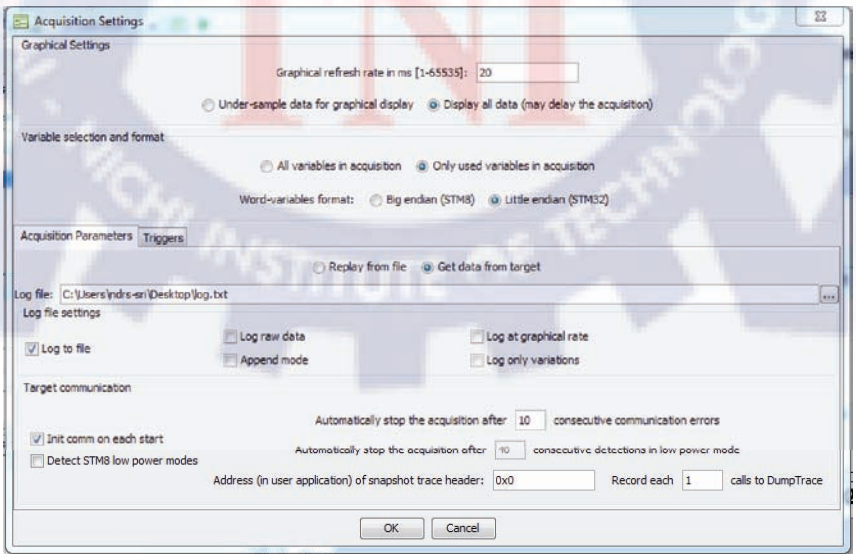
รูปที่ ค.3 การแสดงผลแบบ Table

5.2.1 วิธีการใช้งาน STM Studio

การใช้งาน STM studio ต้องมีคุณสมบัติดังต่อไปนี้ก่อน

- ต้องมีไฟล์ executable ของโปรแกรมที่ต้องการจะเก็บข้อมูล หรือรู้ address ที่เก็บข้อมูล และข้อมูลนั้นๆต้องเป็นข้อมูลแบบ Public หรือ Global Variable
- โปรแกรมที่ต้องการจะเก็บข้อมูล ต้องถูกโหลดลงในไมโครคอนโทรลเลอร์เรียบร้อยแล้ว พร้อมจะทำงาน
- ตัวไมโครคอนโทรลเลอร์ต้องเชื่อมต่อกับคอมพิวเตอร์ผ่าน ST-LINK หรือ RLINK โดยใช้ Protocal รูปแบบใดรูปแบบหนึ่งที่สนับสนุนกับ STM studio

5.2.2 ตัวเลือกต่างๆใน STM Studio



รูปที่ ค.4 หน้าต่าง Setting ของ STM Studio

Graphical settings ดังรูปที่ ค.4 มีรายละเอียดดังต่อไปนี้

- Graphical refresh rate in ms หมายถึงระยะเวลาแสดงผลข้อมูล 1 ครั้ง ให้ตั้งค่าตามที่ต้องการ เช่น 100 หมายความว่าทุกๆ 100 ms โปรแกรมจะแสดงผล 1 ครั้ง

- Under-sample คือการใช้วิธีการแสดงผลข้อมูลด้วยความถี่ที่น้อยกว่าของจริงเพื่อให้แสดงผลได้ทัน

- Display all data คือการรับข้อมูลทั้งหมดให้ครบก่อนแล้วค่อยแสดงผล จึงทำงานได้ช้า

Variable selection and format เลือกวิธีการจัดการกับข้อมูล ประกอบด้วย

- All variables in acquisition คือการรับข้อมูลทั้งหมดที่มีอยู่ในไมโครคอนโทรลเลอร์ ไม่แนะนำให้ใช้ เพราะจะทำให้เกิดความผิดพลาดในการรับข้อมูลบางส่วนได้ และเมื่อเกิดความผิดพลาด ตัวโปรแกรมจะหยุดทำงาน

- Only used variables in acquisition คือการรับข้อมูลเฉพาะส่วนที่ต้องแสดงผลจากไมโครคอนโทรลเลอร์

- Word-variables format คือขนาดของข้อมูลที่จะรับ ให้เลือกตามโครงสร้างของไมโครคอนโทรลเลอร์ที่ใช้ โดยสามารถตัดสินจากชื่อได้ดังนี้

  (1) Big endian (STM8) สำหรับผู้ใช้ STM8

  (2) Little endian (STM32) สำหรับผู้ใช้ STM32

Acquisition Parameters เลือกข้อมูลที่นำมาแสดง

- Replay from file ในกรณีที่เราได้เก็บ Log file มาไว้แล้ว เราสามารถนำมาแสดงผลใหม่ได้โดยการเลือกตัวเลือกนี้ แล้วตัวเลือกที่เหลือก็ไม่เป็น

- Get data from target เป็นการเก็บข้อมูลจากไมโครคอนโทรลเลอร์ที่เราเลือกและได้

เชื่อมต่อเข้ากับคอมพิวเตอร์ ถ้าหากเลือกตัวนี้เลือกจะต้องเลือกตัวเลือกอื่นด้วย

- Log file คือพื้นที่เก็บ Log file โดยสามารถเปลี่ยนชื่อได่

- Log file settings

(1) Log to file จะเก็บข้อมูลที่แสดงผลลงใน Log file

(2) Log raw data จะเก็บข้อมูลดิบที่ได้รับจากไมโครคอนโทรลเลอร์ลงใน Log file ด้วย

(3) Log at graphical rate จะเก็บข้อมูลลงใน Log file ด้วยความถี่ของ Graphical refresh rate in ms ที่ได้ตั้งไว้

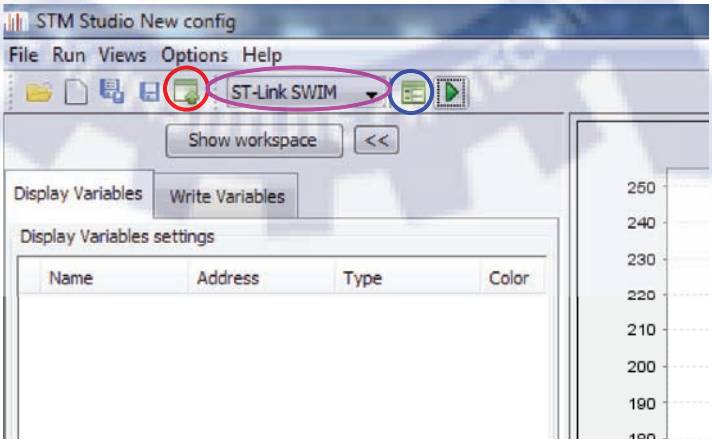
(4) Append mode จะต้อง disable ทุกครั้งก่อนเขียนข้อมูลลงใน Log file

(5) Log only variations จะเก็บข้อมูลเฉพาะตัวแปรที่รับมา

Target Communication และ Triggers ในส่วนนี้ไม่จำเป็นต้องเปลี่ยนแปลงใดๆทั้งสิ้น จึงขอไม่อธิบาย

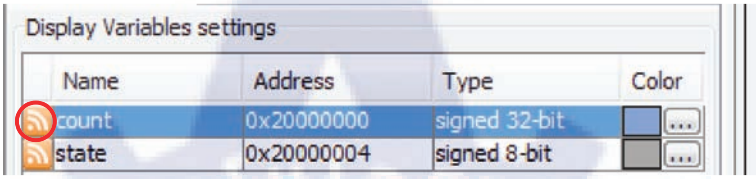
5.2.3 การตั้งค่าเพื่อให้ STM Studio ใช้งานได้

เปิดโปรแกรม STM Studio ขึ้นมาจะเห็นเครื่องมือดังรูปที่ ค.5



รูปที่ ค.5 ส่วนที่ใช้ในการตั้งค่า

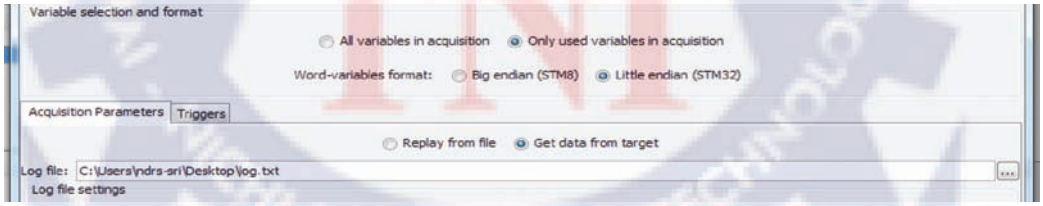
1) กดปุ่มที่วงกลมสีแดง ตามรูปที่ ค.5 แล้วเลือก executable file ปกติจะอยู่ในโฟรเดอร์ Project\STM32L152-EVAL RAM Debug\Exe สกุลไฟล์คือ .out เมื่อเลือกแล้วจะมีตัวแปรขึ้นมา ก็ให้ import เฉพาะตัวแปรที่ต้องการเก็บข้อมูล แล้วให้เปลี่ยนการรับข้อมูลเป็นแบบ Direct mode ให้หมดทุกตัวแปร โดยการกดที่ด้านหน้าของชื่อตัวแปรตามวงกลมสีแดงของรูปที่ ค.6



รูปที่ ค.6 การเปลี่ยนเป็น Direct mode

2) กดปุ่มที่วงกลมสีชมพู่ ตามรูปที่ ค.5 แล้วเลือก ST-LINK SWD หรือวิธีการเชื่อมต่อที่ใช้ให้ถูกต้อง

3) กดปุ่มที่วงกลมสีฟ้า ตามรูปที่ ค.5 แล้วเลือกใช้งาน Only used variables in acquisition, Little endian (STM32) และเปลี่ยนที่อยู่ Log file เป็นสถานที่อื่นนอกจากที่กำหนดมาให้ดังรูปที่ ค.7



รูปที่ ค.7 การตั้งค่า Settings

ต่อไปจะเป็นการนำตัวแปรที่มีมาแสดงผลใน VarViewer เนื่องจากเรามีตัวแปร 2 ตัวที่ต้องการแสดงผลในรูปแบบ Curve แยกจากกัน จึงต้องเพิ่ม Varviewer ขึ้นอีก 1 ชุดโดยคลิกที่ General ในส่วนของ Viewers settings แล้วเลือก Add Viewers จะมีแทป VarViewer 2 แสดงขึ้นมาทางด้านซ้าย

ใส่ตัวแปรโดยคลิกขวาที่ตัวแปรแล้วเลือก Sent to ในกรณีทดลองนี้ เราจะให้ count แสดงที่ VarViewer1 และ state แสดงที่ VarViewer2

เปลี่ยน Range ของ VarViewer แต่ล่ะตัว โดย VarViewer 1 ให้ดูตั้งแต่ -5 ถึง 30 และ VarViewer 2 ให้ดูตั้งแต่ -1 ถึง 2 โดยเข้าไปเปลี่ยนแปลง lower Value และ upper Value ใน VarViewer แต่ล่ะตัว

หลังจากนั้นคลิก Start แล้วโปรแกรมจะเริ่มเก็บข้อมูล

ง. ส่วนประกอบของการคำนวน CRC

ในโปรแกรมคำนวน CRC มักจะมีตัวเลือกอื่นๆที่ไม่เกี่ยวกับ Initial Polynomial และ Data อยู่ ซึ่งแสดงอยู่ดังรูปที่ ง.1 มีรายละเอียดดังต่อไปนี้

	CRC-CCITT	CRC-16	CRC-32
Width	16 bits	16 bits	32 bits
(Truncated) Polynomial	0x1021	0x8005	0x04C11DB7
Initial Remainder	0xFFFF	0x0000	0xFFFFFFFF
Final XOR Value	0x0000	0x0000	0xFFFFFFFF
Reflect Data?	No	Yes	Yes
Reflect Remainder?	No	Yes	Yes
Check Value	0x29B1	0xBB3D	0xCBF43926

รูปที่ ง.1 ค่าต่างๆที่มีใน CRC Calculator ส่วนใหญ่

1) Width

เป็นค่ากำหนดขนาดของการ CRC เช่น ในกรณีของ CRC8 ก็หมายความว่าเมื่อการคำนวน CRC ทั้งหมดเสร็จสิ้นแล้วจะมีจำนวน bit ของ CRC เท่ากับ 8 โดยปกติจะเป็นตัวกำหนด bit ของค่า Initial และจำนวน bit ของ Polynomial

2) Polynomial

เปรียบเสมือน Key ที่ใช้ในการ Decode ข้อมูลเพียงแค่ในกรณีนี้เราใช้ Key เพื่อตรวจสอบความถูกต้องในการส่งข้อมูล โดยขนาดของคีจะมีค่าเท่ากับ Width + 1 แต่เนื่องจากเรารู้ bit หน้าสุดอยู่แล้วว่าต้องเป็น 1 จึงไม่จำเป็นต้องใส่ และ bit สุดท้ายของ Polynomial ควรจะมีค่าเป็น 1 เสมอ ในปัจจุบัน Polynomial ที่ใช้กันมีหลายชนิดดังตารางที่ 1.2.1

3) Initial Remainder

คือค่าเริ่มต้นของการหา CRC ในกรณีนี้หมายถึงค่า CRC ของการคำนวนข้อมูล Box ที่แล้ว โดยส่วนมากจะมีค่าไม่เป็น 0 ทุกตัวก็ 1 ทุกตัวตามมาตราฐานของ CRC

4) Initial Final XOR Value

ใช้ในการเข้ารหัส ในกรณีที่ต้องการเพิ่มความปลอดภัยให้กับ CRC เพราะโดยทั่วไปแล้วผู้ที่ต้องการแก้ไขข้อมูลสามารถคาดเดา Polynomial จากค่า CRC และ Data ได้ โดยจะนำค่าที่ใส่ไป XOR กับผลลัพธ์สุดท้ายที่ได้จากการคำนวน ในกรณีที่เราต้องการจำลองการคำนวนของ CRC Calculation ภายใน Microcontroller STM32L152RB แล้วไม่จำเป็นต้องมี เพราะว่าตัว Module จะคำนวนค่า CRC ให้เราเท่านั้น ถ้าเราต้องการจะเข้ารหัสจำเป็นต้องเขียนรหัสโปรแกรมเอง

5) Reflect Data/Reverse Data

เป็นคำสั่งเพื่อบอกว่าจะทำการกลับบิทของข้อมูลก่อนจะเริ่มการหาค่า CRC หรือไม่ ในกรณีของ CRC Calculation ของไมโครคอนโทรลเลอร์จะไม่มีการกลับบิทโดย Hardware แต่สามารถเขียนโปรแกรมเพื่อทำการกลับบิทก่อนนำไปคำนวนได้

6) Reflect Remainder/Reverse CRC result

เป็นคำสั่งเพื่อบอกว่าจะทำการกลับบิทของผลลัพธ์จากการคำนวน CRC หรือไม่ ในกรณีของ CRC Calculation ของไมโครคอนโทรลเลอร์จะไม่มีการกลับบิทโดย Hardware แต่สามารถเขียนโปรแกรมเพื่อทำการกลับบิทก่อนนำไปคำนวนได้ และการกลับบิทนี้จะเกิดขึ้นก่อนที่จะนำไป XOR กับค่า Initial Final XOR Value เสมอ

7) Check Value

คือค่าที่เอาไว้เช็คการทำงานของ Module CRC Calculation โดยให้ใส่ข้อมูล String “123456789” ลงไป ถือเป็นมาตราฐานของ CRC ทุกตัว

จ. หลักการที่นำมาใช้ในการหา CRC

หลักการที่ 1 การคำนวน CRC ด้วยหลักการ Polynomial Long Division

กำหนดให้	Width	=	8
	Polynomial	=	0x03 (0x103)
	Initial Remainder	=	0xFF
	Data	=	0x30 (String “0”)
	Initial Final XOR Value	=	0x00
	Reflect Data/Reverse Data	=	no
	Reflect Remainder/Reverse CRC result	=	no

(1) เริ่มจากโหลดค่า Initial Remainder เข้ามาเก็บไว้ใน Register คำตอบ

Answer = 0xFF;

(2) แล้วทำการ Shift ไปทางซ้าย(MSB)

Answer = 0xFE โดย bit ที่ Shift ออกไปมีค่าเท่ากับ L

(3) ทำการตรวจสอบว่า bit ที่ Shift ออกไป มีค่าเท่ากับ Data bit ที่ L – n หรือไม่ (กำหนดให้ L = Width)

ในกรณีนี้ Data = 0b00110000 bit ที่ L – n ก็คือบิทซ้ายสุด มีค่าเท่ากับ 0 ถ้าหากมีค่าไม่เท่ากัน ให้ XOR คำตอบด้วยค่า Polynomial ถ้าเท่ากัน กลับไปทำข้อ 2

Answer = 0xFE XOR 0x03 = 0xFD n=1

(4) กลับไปทำข้อ 2 ใหม่ จนกว่า n = Width

ผลการคำนวนในกรณีนี้

n = 1 : Answer = 0xFD	Data bit (L – n) = 0	bit just shift out = 1
n = 2 : Answer = 0xF9	Data bit (L – n) = 0	bit just shift out = 1
n = 3 : Answer = 0xF2	Data bit (L – n) = 1	bit just shift out = 1
n = 4 : Answer = 0xE4	Data bit (L – n) = 1	bit just shift out = 1
n = 5 : Answer = 0xCB	Data bit (L – n) = 0	bit just shift out = 1
n = 6 : Answer = 0x95	Data bit (L – n) = 0	bit just shift out = 1
n = 7 : Answer = 0x29	Data bit (L – n) = 0	bit just shift out = 1
n = 8 : Answer = 0x52	Data bit (L – n) = 0	bit just shift out = 0

เพราะฉะนั้น ค่าของ CRC จะมีค่าเท่ากับ 0x52

หลักการที่ 2 หลักการคำนวน CRC โดยใช้ Table

ถ้าเรามีข้อมูลที่อยู่ใน Register เป็น 0x98 โดย biy ที่พึ่ง shift ออกมีค่าเป็น 0x3 และกำหนดให้ polynomial มีค่าเท่ากับ 0x0A หรือ 0x103

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

จะเห็นได้ว่า ข้อมูลของ 8 bit หลังจะขึ้นอยู่กับ 4 bit หน้าโดยตรง เพราะงั้นถ้าหากเรารู้ polynomial เราสามารถสร้างองค์ประกอบของข้อมูล 8 bit หลัง แล้วใช้คุณสมบัติของ Array เพื่อช่วยให้คำนวนข้อมูล 8 bit หลังได้เร็วขึ้น เราเรียกวิธีการนี้ว่าการทำ CRC Table

หลักการที่ 3 หลักการคำนวน CRC โดยใช้ หลักการใหม่

เนื่องจากเราสามารถพิสูจณ์ทางคณิตศาสตร์ได้ว่า CRC เป็นเพียงผลรวมระหว่างข้อมูลและค่า Initial ที่มา xor กันเท่านั้น และลำดับการ xor ค่อนข้างจะง่ายในการใช้วิธีการ shift bit เพื่อแก้สมการ ดังตัวอย่างต่อไปนี้

สมมุติให้เราต้องการคำนวนหา CRC CCITT ซึ่งมี width 16 bit และ polynomial เท่ากับ 0x1021 เราสามารถแทนค่าทุกค่าด้วยตัวแปรได้ตามตารางต่อไปนี้

ตารางที่ จ.1 แสดงการหา CRC CCITT ด้วยตัวแปร

	I0	I1	I2	I3	I4	I5	I6	I7	I8	I9	I10	I11	I12	I13	I14	I15							
	d0	d1	d2	d3	d4	d5	d6	d7															
C0	p0	p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	p15	p16						
C1		p0	p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	p15	p16					
C2			p0	p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	p15	p16				
C3				p0	p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	p15	p16			
C4					p0	p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	p15	p16		
C5						p0	p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	p15	p16	
C6							p0	p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	p15	p16
C7								p0	p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	p15	p16
	0	0	0	0	0	0	0	0	A0	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	A12	A13	A14	A15

โดย I0 ถึง I15 คือค่า initial bit ที่ 0 ถึง 15

d0 ถึง d7 คือค่า data bit ที่ 0 ถึง 7

p0 ถึง p16 คือค่า polynomial ของ bit ที่ 0 ถึง 16

C0 ถึง C7 คือค่าคงที่ที่ควบคุมว่า polynomial ชุดนั้นจำเป็นหรือไม่

เขียนสมการออกมาได้ดังนี้

0 = I0 ^ d0 ^ C0p0
0 = I1 ^ d1 ^ C0p1 ^ C1p0
0 = I2 ^ d2 ^ C0p2 ^ C1p1 ^ C2p0
0 = I3 ^ d3 ^ C0p3 ^ C1p2 ^ C2p1 ^ C3p0
0 = I4 ^ d4 ^ C0p4 ^ C1p3 ^ C2p2 ^ C3p1 ^ C4p0
0 = I5 ^ d5 ^ C0p5 ^ C1p4 ^ C2p3 ^ C3p2 ^ C4p1 ^ C5p0
0 = I6 ^ d6 ^ C0p6 ^ C1p5 ^ C2p4 ^ C3p3 ^ C4p2 ^ C5p1 ^ C6p0
0 = I7 ^ d7 ^ C0p7 ^ C1p6 ^ C2p5 ^ C3p4 ^ C4p3 ^ C5p2 ^ C6p1 ^ C7p0

แทนด้วย polynomial 0x1021

0 = I0 ^ d0 ^ C0	C0 = I0 ^ d0
0 = I1 ^ d1 ^ C1	C1 = I1 ^ d1
0 = I2 ^ d2 ^ C2	C2 = I2 ^ d2
0 = I3 ^ d3 ^ C3	C3 = I3 ^ d3
0 = I4 ^ d4 ^ C0 ^ C4	C4 = I4 ^ d4 ^ C0
0 = I5 ^ d5 ^ C1 ^ C5	C5 = I5 ^ d5 ^ C1
0 = I6 ^ d6 ^ C2 ^ C6	C6 = I6 ^ d6 ^ C2
0 = I7 ^ d7 ^ C3 ^ C7	C7 = I7 ^ d7 ^ C3

จากสมการจะพบว่าเราสามารถหา C0 ถึง C7 ได้ในรูปแบบติดค่า Initial และ data ที่ใส่เข้ามา

A0 = I8 ^ C7p1 ^ C6p2 ^ C5p3 ^ C4p4 ^ C3p5 ^ C2p6 ^ C1p7 ^ C0p8
A1 = I9 ^ C7p2 ^ C6p3 ^ C5p4 ^ C4p5 ^ C3p6 ^ C2p7 ^ C1p8 ^ C0p9
A2 = I10 ^ C7p3 ^ C6p4 ^ C5p5 ^ C4p6 ^ C3p7 ^ C2p8 ^ C1p9 ^ C0p10
A3 = I11 ^ C7p4 ^ C6p5 ^ C5p6 ^ C4p7 ^ C3p8 ^ C2p9 ^ C1p10 ^ C0p11
A4 = I12 ^ C7p5 ^ C6p6 ^ C5p7 ^ C4p8 ^ C3p9 ^ C2p10 ^ C1p11 ^ C0p12
A5 = I13 ^ C7p6 ^ C6p7 ^ C5p8 ^ C4p9 ^ C3p10 ^ C2p11 ^ C1p12 ^ C0p13
A6 = I14 ^ C7p7 ^ C6p8 ^ C5p9 ^ C4p10 ^ C3p11 ^ C2p12 ^ C1p13 ^ C0p14
A7 = I15 ^ C7p8 ^ C6p9 ^ C5p10 ^ C4p11 ^ C3p12 ^ C2p13 ^ C1p14 ^ C0p15
A8 = C7p9 ^ C6p10 ^ C5p11 ^ C4p12 ^ C3p13 ^ C2p14 ^ C1p15 ^ C0p16
A9 = C7p10 ^ C6p11 ^ C5p12 ^ C4p13 ^ C3p14 ^ C2p15 ^ C1p16
A10 = C7p11 ^ C6p12 ^ C5p13 ^ C4p14 ^ C3p15 ^ C2p16
A11 = C7p12 ^ C6p13 ^ C5p14 ^ C4p15 ^ C3p16
A12 = C7p13 ^ C6p14 ^ C5p15 ^ C4p16
A13 = C7p14 ^ C6p15 ^ C5p16
A14 = C7p15 ^ C6p16
A15 = C7p16

คิดหาสมการของคำตอบ จะได้ว่า

แทนด้วย polynomial 0x1021 และค่า C0 ถึง C7 ลงในสมการ จะได้คำตอบดังนี้

A0 = I8^C4	A0 = I8 ^ I4 ^ d4 ^ I0 ^ d0
A1 = I9^C5	A1 = I9 ^ I5 ^ d5 ^ I1 ^ d1
A2 = I10^C6	A2 = I10 ^ I6 ^ d6 ^ I2 ^ d2
A3 = I11^C7 ^ C0	A3 = I11 ^ I7 ^ d7 ^ I3 ^ d3
A4 = I12^C1	A4 = I12 ^ I1 ^ d1
A5 = I13^C2	A5 = I13 ^ I2 ^ d2
A6 = I14^C3	A6 = I14 ^ I3 ^ d3
A7 = I15^C4	A7 = I15 ^ I4^ d4 ^ I0 ^ d0
A8 = C5 ^ C0	A8 = I0 ^ d0 ^ I1 ^ d1 ^ I5 ^ d5
A9 = C6 ^ C1	A9 = I1 ^ d1 ^ I2 ^ d2 ^ I6 ^ d6
A10 = C7 ^ C2	A10 = I2 ^ d2 ^ I3 ^ d3 ^ I7 ^ d7
A11 = C3	A11 = I3 ^ d3
A12 = C4	A12 = I0 ^ d0 ^ I4 ^ d4
A13 = C5	A13 = I1 ^ d1 ^ I5 ^ d5
A14 = C6	A14 = I2 ^ d2 ^ I6 ^ d6
A15 = C7	A15 = I3 ^ d3 ^ I7 ^ d7

สามารถเปลี่ยนให้ดูง่ายโดย เขียนดังนี้

I8	I9	I10	I11	I12	I13	I14	I15	I0	I1	I2	I3	I4	I5	I6	I7
								d0	d1	d2	d3	d4	d5	d6	d7
I4	I5	I6	I7				I4	I5	I6	I7		I0	I1	I2	I3
d4	d5	d6	d7				d4	d5	d6	d7		d0	d1	d2	d3
I0	I1	I2	I3				I0	I1	I2	I3					
d0	d1	d2	d3				d0	d1	d2	d3					
			I0	I1	I2	I3									
			d0	d1	d2	d3									
A0	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	A12	A13	A14	A15

รูปที่ จ.1 แสดงคำตอบของ CRC CCITT

สามารถเขียนเป็นโปรแกรมได้ดังต่อไปนี้

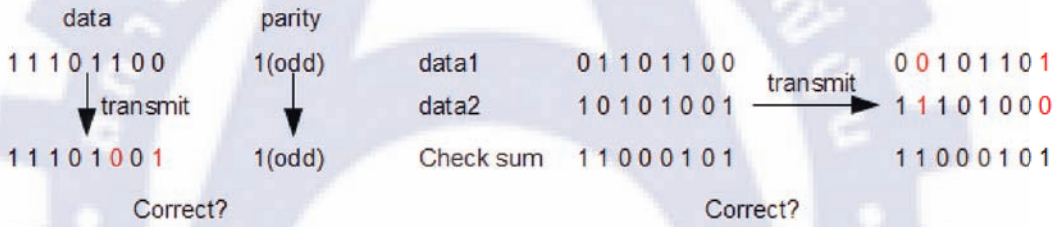
```
uint16_t crc16_ccitt_acc(uint_fast16_t crc_acc, uint_fast8_t dat_inp) {
    crc_acc  = SWAP_16(crc_acc);
    crc_acc ^= dat_inp;
    crc_acc ^= (crc_acc & 0x00FFU) >> 4;
    crc_acc ^= (crc_acc << 12);
    crc_acc ^= ((crc_acc & 0x00FFU) << 5);

    return (uint16_t)(crc_acc);
}
```

ฉ. การคำนวน CRC

CRC หรือ Cyclic redundancy checking แปลเป็นภาษาไทยได้ว่า วิธีการเช็คโดยการคำนวนเป็นรอบ เป้าหมายในการคำนวน CRC คือการตรวจสอบว่าข้อมูลที่ทำการส่งไปนั้นถูกต้องหรือไม่ เนื่องจากในการสื่อสารระหว่างไมโครคอนโทรลเลอร์หรือ computer มีโอกาสที่ข้อมูลจะเปลี่ยนแปลงไปเนื่องจากสภาวะที่มีสัญญาณรบกวนสูง ปกติแล้ววิธีการทั่วๆไปที่ใช้กันก็จะมีอย่างเช่น parity bit หรือ check sum ซึ่งจะมีข้อเสียดังต่อไปนี้

parity bit ถ้าหากข้อมูลมีการเปลี่ยนแปลงไป 2 ค่าจะทำให้ไม่ทราบถึงความเปลี่ยนแปลง แสดงดังรูปที่ ฉ.1 (ซ้าย)

check sum ถ้าหากข้อมูลที่อยู่ในแถวเดียวกันมีการเปลี่ยนแปลงไป 2 ค่าจะทำให้ไม่ทราบถึงความเปลี่ยนแปลงแสดงดังรูปที่ ฉ.1 (ขวา)

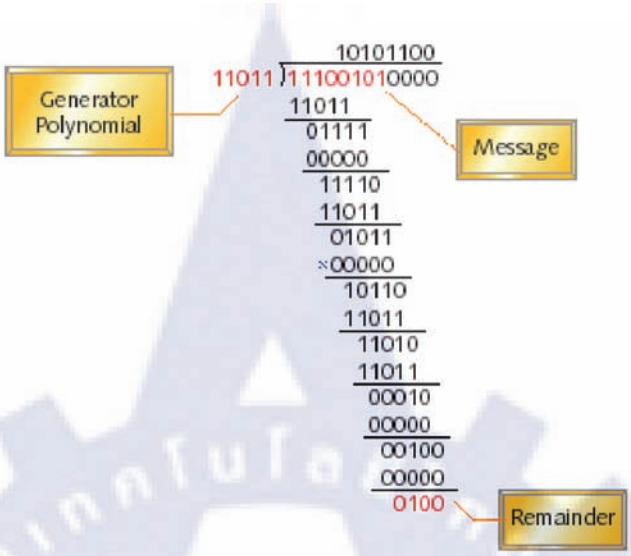


รูปที่ ฉ.1 ตัวอย่างความผิดพลาดที่เกิดขึ้นและไม่สามารถตรวจสอบได้

หลักการการคำนวน CRC คือ การใช้คุณสมบัติของการหาร เพื่อตรวจสอบข้อมูลทุก bit และทุกลำดับ เพราะฉะนั้นถ้าหากมี bit ไหนเปลี่ยนแปลง จะทำให้ค่าของ CRC เปลี่ยนแปลงในทันที ด้วยวิธีนี้มีความแม่นยำอย่างมาก การคำนวน CRC มีตัวแปรที่สำคัญดังต่อไปนี้

- Generator Polynomial หรือ Polynomial ที่เราใช้เข้ารหัส
- Message หรือ Data ที่ xor กับ Initial
- Remainder หรือ CR

ในการคำนวน CRC ที่มีข้อมูลหลายชุด ค่า CRC ของข้อมูลก่อนหน้าจะต้องนำมาใช้เป็นค่า Initial สำหรับการคำนวน CRC ชุดถัดไป



รูปที่ ฉ.2 วิธีการคำนวน CRC โดยวิธีการ Polynomial Long Division

การคำนวน CRC ดังรูปที่ ฉ.2 อาจคล้ายกับหลักการตั้งหารยาว แต่ที่จริงแล้วจะมีความแตกต่างกันอยู่เล็กน้อยคือ ในการตั้งหารยาวจะใช้วิธีการลบ แต่การทำ Polynomial Long Division ใช้วิธีการ xor หรือ ถ้าหากตัวตั้งต่างกัน ผลลัพธ์จะเป็น 1 ถ้าเหมือนกันจะเป็น 0

ตัวอย่าง

11100 xor 11011 จะได้ 00111 / 11100 ลบ 11011 จะได้ 00011

Reflect CRC โดยทั่วไปจะถูกเรียกว่า Reverse CRC

การคำนวน Reflect CRC คือการใช้ Polynomial ที่ผ่านการ Reflect (สลับ MSB และ LSB) มาคำนวนหา CRC โดยจะมีผลลัพธ์เท่ากับการนำ Initial และ Data ที่ผ่านการ Reflect มาคำนวนกับ Polynomial ที่ยังไม่ผ่านการ Reflect

Reverse CRC คือ Data ที่ทำให้ได้ค่า CRC ที่ต้องการ

ช. พิสูจณ์หลักการหา Reverse CRC โดยใช้สมการ

นำหลักการที่ทำการพิสูจณ์แล้วมาทดลองการหาค่า Reverse CRC โดยใช้หลักการ Inverse mod (2^32) ตามสมการ X^32*M(x) = Q(x)G(x) + R(x), แต่หลังจากใช้ความพยายามในการแก้สมการอยู่หลายวันก็พบว่าเราไม่สามารถแก้สมการดังกล่าวได้ เนื่องจากเราไม่ทราบ Q(x) ซึ่งจะเป็นค่าที่จะบอกค่า M(x) หรือ data ที่เราต้องการนั้นเอง แต่ข้าพได้พบกับอีกสมการ ก็คือ สมการดังต่อไปนี้

(((BB1 ^ 0) >> 8) ^ CRC_Table[i3]) >> 16 = BB

BB1 = (CRC_Table[i2] ^ initial(0)) >> 24

(((CC1 ^ 0) >> 8) ^ CRC_Table[i3]) >> 8 = CC

CC1 = (CC2 ^ CRC_Table[i2]) >> 16

CC2 = (CRC_Table[i1] ^ initial(0)) >> 24

(((DD1 ^ 0) >> 8) ^ CRC_Table[i3]) = DD

DD1 = (DD2 ^ CRC_Table[i2]) >> 8

DD2 = (DD3 ^ CRC_Table[i1]) >> 16

DD3 = (CRC_Table[i0] ^ initial(0)) >> 24

ซึ่งสามารถพิสูจณ์ทางคณิตศาสตร์ได้ มีหลักการคือสร้างตารางที่ bypass Q(x) และเมื่อเรารู้ R(x) เราจะรู้ M(x) ทันที

ซ. ตัวอย่างโปรแกรมที่ใช้คำนวน CRC แบบต่างๆ

CRC32 using table 8 bit

```
uint32_t    crc_tbl_cal(uint32_t init,uint8_t value){
  uint32_t crc;
  uint32_t tem32_1;

  tem32_1 = (uint32_t)(init >> 24) ^ (value);
  crc = (init << 8) ^ CRC_Table[tem32_1];

  return(crc);
}


uint32_t    crc_tbl_cal_data_box(uint32_t init,uint8_t* pBuffer,uint32_t size){
  uint32_t crc;
  uint32_t i;

  crc = init;
  for(i=0;i<size;i++){
    crc = crc_tbl_cal(crc,pBuffer[i]);
  }
  return(crc);
}
```

Reverse CRC32 using table 8 bit

```
uint32_t   crc_inv(uint32_t crc32){                 //!< Get CRC inversion.
  uint32_t result;
  find_indx(crc32);
  result = patch_value(CRC32_XOR_INIT);
  return(result);
}


void        find_indx(uint32_t value){
  for(int i = 3; i >= 0 ; i--){
   TableIdx[i] = lookup_table(value);
   value = (value ^ CRC_Table[TableIdx[i]]) >> 8;
  }
}


uint32_t    patch_value(uint32_t init){
  uint32_t patch[4];
  uint32_t crc = init;
  for(int i = 0; i < 4;i++){
   patch[i] = ((crc ^ (TableIdx[i] << 24)) & 0xFF000000U) >> 24;
   crc = (crc << 8) ^ CRC_Table[patch[i]^((crc & 0xFF000000U) >> 24)];
  }
  return(patch[0]<<24 | patch[1]<<16 | patch[2] << 8 | patch[3]);
}
```

CRC32 using table 4 bit

```
uint32_t crc32_tbl_cal(const uint8_t* msg_ptr, uint16_t msg_size) {
  uint32_t crc_acc;
  uint16_t tmp16_1;

  crc_acc = CRC32_INIT;
  while (msg_size > 0U) {
    msg_size--;

    #if (USE_FULL_CRC_TABLE == 1)
    tmp16_1 = (crc_acc >> 24) ^ (*msg_ptr);
    crc_acc = crc32_tbl[tmp16_1] ^ (crc_acc << 8);
    #else
    // First compute high nibble
    tmp16_1 = (uint16_t)(crc_acc >> 28) ^ (uint16_t)(*msg_ptr >> 4);
    crc_acc = crc32_tbl[tmp16_1] ^ (crc_acc << 4);

    // Next compute low nibble
    tmp16_1 = (uint16_t)(crc_acc >> 28) ^ (uint16_t)(*msg_ptr & 0x0FU);
    crc_acc = crc32_tbl[tmp16_1] ^ (crc_acc << 4);
    #endif

    msg_ptr++;
  }

  return (crc_acc ^ CRC32_FINAL_XOR);
}
```

Reverse CRC32 using byte width shifting algorithm 4 bit version

```
uint32_t crc32_inv(uint32_t crc_acc, uint32_t crc_wat){
  uint8_t     patch[8];
  uint8_t     indx_tbl[8];
  uint32_t    crc_reverse;
  int    i,j;


  for (s8_1 = 7; s8_1 >= 0; s8_1--) {
    constant_1 = (uint8_t)(((crc_wat) ^ (crc_wat << 1) ^ (crc_wat << 3)) & 0xFU);
    indx_tbl[s8_1] = constant_1;
    constant_2 = (constant_1 ^ (uint8_t)(constant_1 << 1));
    constant_3 = (constant_2 ^ (uint8_t)(constant_1 << 2));
    crc_wat = crc_wat ^ ((uint32_t)constant_1 << 26);
    crc_wat = crc_wat ^ ((uint32_t)constant_1 << 16);
    crc_wat = crc_wat ^ ((uint32_t)constant_2 << 22);
    crc_wat = crc_wat ^ ((uint32_t)constant_2 << 7U);
    crc_wat = crc_wat ^ ((uint32_t)constant_2 << 4U);
    crc_wat = crc_wat ^ ((uint32_t)constant_3 << 10);
    crc_wat = crc_wat ^ (constant_3);

    crc_wat >>= 4;
  }

  crc_reverse = 0;

  for(i=0;i<8;i++){
    patch[i] = ((crc_acc ^ (indx_tbl[i] << 28)) & 0xF0000000U) >> 28;
    crc_acc = (crc_acc << 4) ^ crc32_tbl[patch[i]^((crc_acc & 0xF0000000U) >> 28)];
    crc_reverse = (crc_reverse << 4) | patch[i];
  }
  return(crc_reverse);

}
```

ประวัติผู้แต่ง

ชื่อ – สกุล	นายสรวิศ สถาพรชัยสิทธิ์
วัน เดือน ปีเกิด	4 เมษายน 2534
ประวัติการศึกษา	
ระดับประถมศึกษา	ประถมศึกษาตอนต้น – ตอนปลาย พ.ศ. 2540 โรงเรียน วัดเวตวันธรรมาวาส
ระดับมัธยมศึกษา	มัธยมศึกษาตอนต้น พ.ศ. 2546 โรงเรียน ศีลาจารพิพัฒน์ (วัดเวตวันธรรมาวาส มัธยมศึกษา) มัธยมศึกษาตอนปลาย พ.ศ. 2549 โรงเรียน สามเสนวิทยาลัย
ระดับอุดมศึกษา	คณะวิศวกรรมศาสตร์บัณฑิต สาขาวิศวกรรมคอมพิวเตอร์ พ.ศ. 2552 สถาบันเทคโนโลยีไทย – ญี่ปุ่น
ทุนการศึกษา	ทุนการศึกษาของสถาบันเทคโนโลยีไทย – ญี่ปุ่น ประเภทที่ 3
ประวัติการฝึกอบรม	1. ศึกษาเรื่อง “ILQ Control” ในโครงการแลกเปลี่ยน ณ Shibaura Institute of Technology เมืองโตเกียว ประเทศญี่ปุ่น
ผลงานที่ได้รับตีพิมพ์	- ไม่มี -