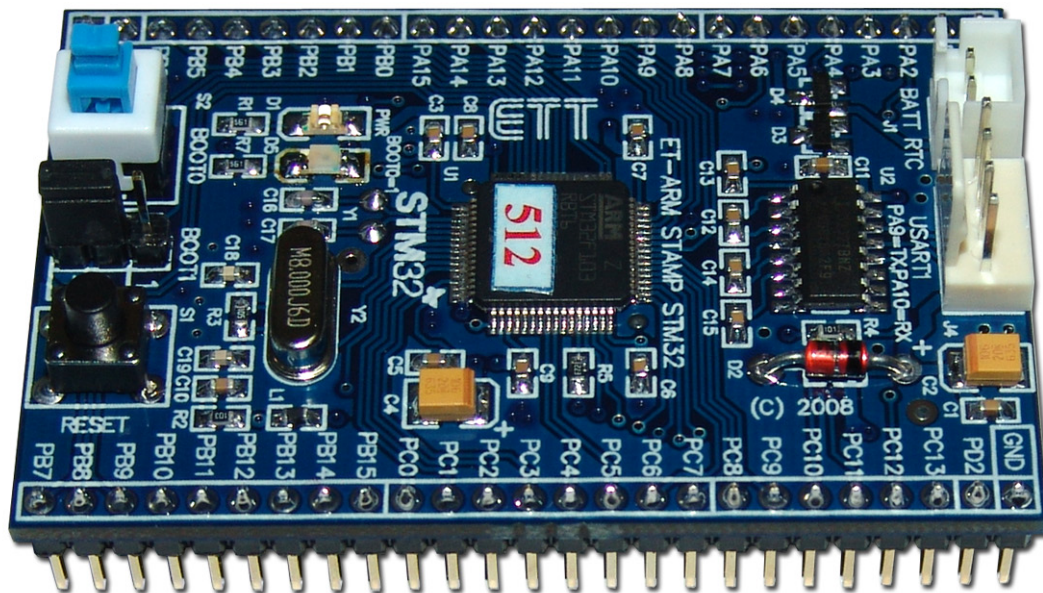


คู่มือการใช้งาน User's manual

STM32

ET-ARM STAMP STM32F103/128
ET-ARM STAMP STM32F103/512



บริษัท อีทีที จำกัด ETT CO., LTD.

1112/96-98 ถนนสุขุมวิท แขวงพระโขนง เขตคลองเตย กรุงเทพฯ 10110 <http://www.etteam.com>

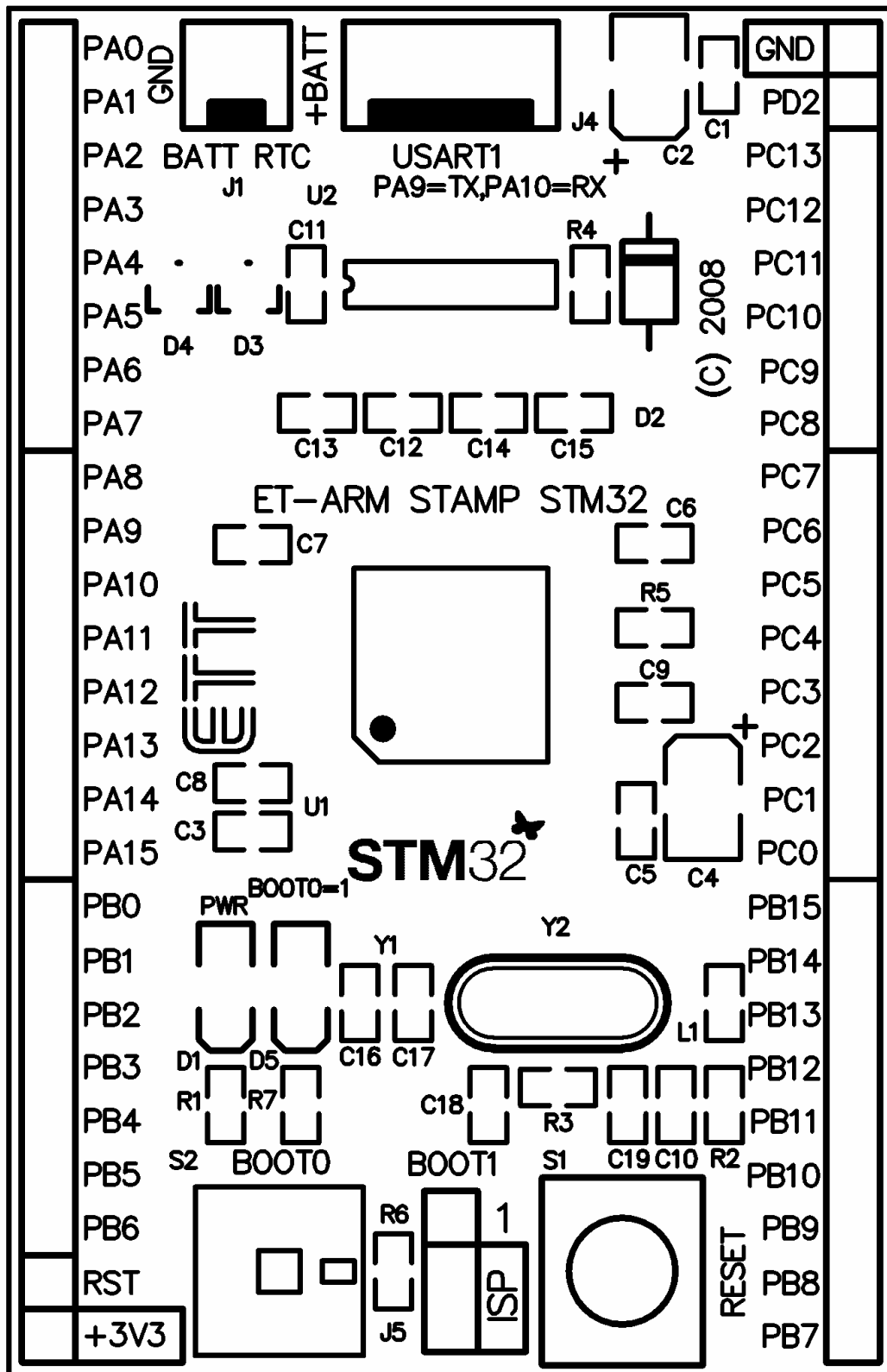
1112/96-98 Sukhumvit Rd., Phrakonong Klongtoey Bangkok 10110

<http://www.ett.co.th>

www.etteam.com Tel : 02-7121120 Fax : 02-3917216

email : sale@etteam.com

ET-ARM STAMP STM32



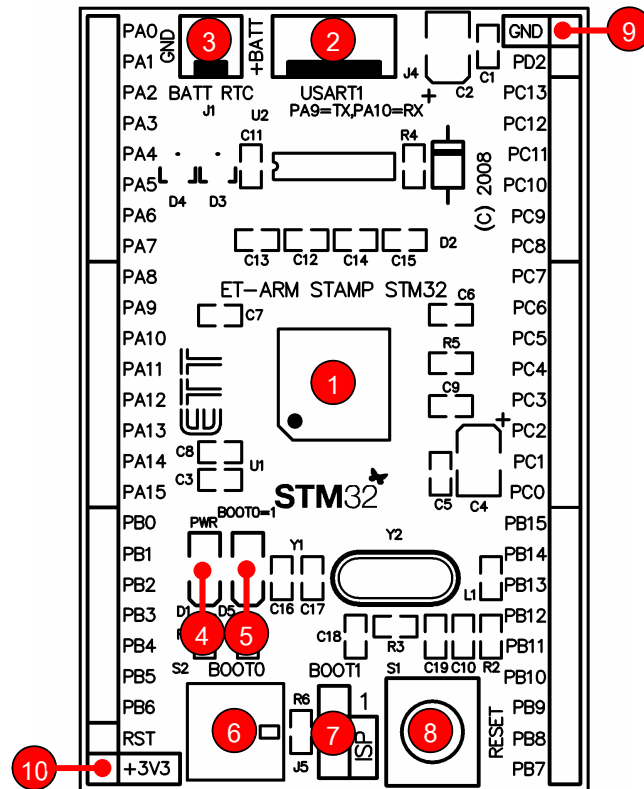
รูปแสดง โครงสร้างของบอร์ด ET-ARM STAMP STM32

คุณสมบัติของบอร์ด รุ่น ET-ARM STAMP STM32

1. ใช้ MCU 32Bit ตระกูล ARM Cortex-M3 ขนาด 64Pin(LQFP) ของ “STmicroelectronics”
2. ใช้ Crystal 8.00 MHz + Phase-Locked Loop (PLL) Run ความถี่ 72 MHz ประมวลผลด้วยความเร็ว 1.25 DMIPS/MHz เทียบเท่ากับความเร็วการประมวลผลเท่ากับ 90MIPS
3. มีวงจร RTC(Real Time Clock) พร้อม XTAL ค่า 32.768KHz และ Battery Backup
4. รองรับการโปรแกรมแบบ In-System Programming (ISP) และ In-Application Programming (IAP) ผ่านทาง On-Chip Boot-Loader Software ทางพอร์ต USART1 (RS232)
5. มีวงจรสื่อสาร RS232 โดยใช้ขั้วต่อแบบ 4-PIN มาตรฐาน ETT จำนวน 1 ช่อง
6. มี 48 Bit GPIO พร้อม 5V-Tolerant สำหรับประยุกต์ต่างๆโดยอิสระ
7. ใช้กับแหล่งจ่ายไฟขนาด +3.3V
8. ขั้วต่อใช้งานวางตัวบน Pin Header ระยะห่าง 2.54mm(100mil) ขนาด 50 Pin (ด้านละ 25Pin) ระยะห่าง 1.5 นิ้ว(1500mil/38.1mm)ง่ายต่อการนำไปต่อประยุกต์ใช้งาน และ ขยายวงจร I/O สามารถใช้กับ Project Board และ PCB

Specification Peripheral	ET-ARM STAMP STM32F103/128	ET-ARM STAMP STM32F103/512
MCU	STM32F103RBT6	STM32F103RET6
Flash	128K	512K
RAM	20K	64K
SPI	2	3 (I2S x 2)
I2C	2	2
USART	3	5
USB	1	1
CAN	1	1
SDIO	-	1
ADC 12 Bit	16	16
DAC 12 Bit	-	2

โครงสร้างบอร์ด ET-ARM STAMP STM32



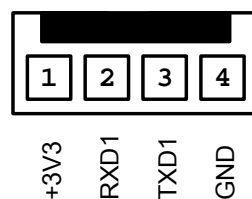
รูปแสดง ตำแหน่งของอุปกรณ์ต่างๆในบอร์ด ET-ARM STAMP STM32

- หมายเลข 1 คือ MCU ประจําบอร์ด
- หมายเลข 2 คือ ขั้วต่อ USART1(RS232) สำหรับใช้งาน และ Download
- หมายเลข 3 คือ ขั้วต่อ Battery Backup-RTC ขนาด +3V
- หมายเลข 4 คือ LED Power
- หมายเลข 5 คือ LED BOOT0 (BOOT0=1)
- หมายเลข 6 คือ สวิตช์ BOOT0 ใช้ร่วมกับ Jumper J1 สำหรับเลือกโหมดการทำงานของบอร์ดแบบ Download จาก USART1 หรือ Run ปรกติ โดย ON เพื่อ Download และ OFF เพื่อ Run
- หมายเลข 7 คือ Jumper(BOOT1) ใช้ร่วมกับสวิตช์ BOOT0 สำหรับเลือกโหมดการทำงานของบอร์ดแบบ Download จาก USART1 หรือ Run ปรกติ ซึ่งต้องกำหนดไว้ที่ ISP เสมอ
- หมายเลข 8 คือ สวิตช์ RESET สำหรับ RESET การทำงานของ MCU
- หมายเลข 9 คือ จุดต่อ GND
- หมายเลข 10 คือ ไฟเลี้ยงวงจร +3.3V

การใช้งาน RS232

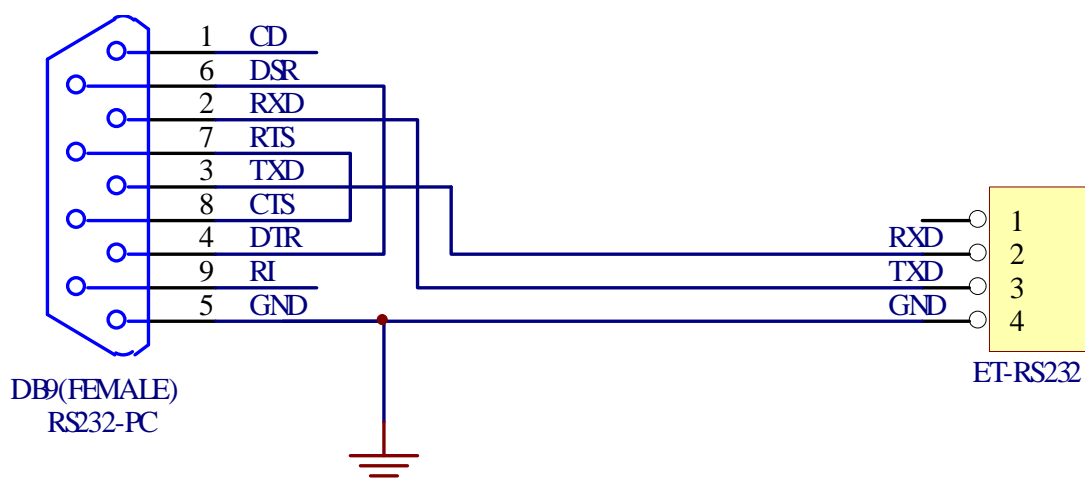
พอร์ต RS232 เป็นสัญญาณ RS232 ซึ่งผ่านวงจรแปลงระดับสัญญาณ MAX3232 เรียบร้อยแล้ว คือ USART1 โดยสามารถใช้เชื่อมต่อกับสัญญาณ RS232 เพื่อรับส่งข้อมูลได้ นอกจากนี้แล้ว USART1 ยังสามารถใช้งานเป็น ISP Download สำหรับทำการ Download Hex File ให้กับ MCU ได้ด้วย โดยในกรณีนี้ ต้องใช้งานร่วมกับ Jumper(BOOT1) SW(BOOT0) และ SW(RESET) เพื่อ Reset ให้ CPU เริ่มต้นทำงาน ใน Boot-Loader Mode เพื่อทำการ Download Hex File ให้กับ CPU ได้ด้วย(ดูรายละเอียดเพิ่มเติมเรื่อง “การ Download Hex File ให้กับ MCU ของบอร์ด”)

USART1(Download)



- USART1 ใช้ขาสัญญาณจาก PA9(TXD1) และ PA10(RXD1)

สำหรับ Cable ที่จะใช้ในการเชื่อมต่อ RS232 ระหว่าง Comport ของเครื่องคอมพิวเตอร์ PC เข้ากับขั้วต่อ USART-1 และ USART-2 ของบอร์ด ET-ARM STAMP STM32 นั้น เป็นดังนี้



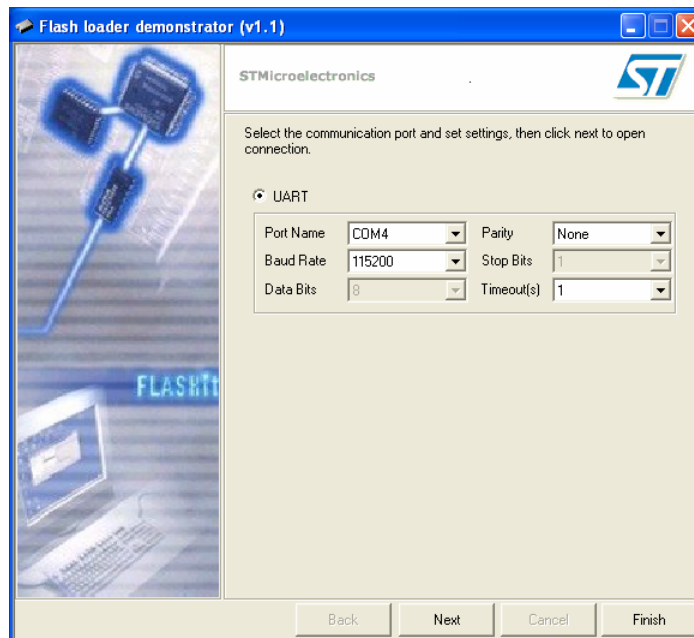
รูป แสดงวงจรสาย Cable สำหรับ RS232

การ Download Hex file ให้กับ MCU ของบอร์ด

การ Download Hex File ให้กับหน่วยความจำ Flash ของ MCU ในบอร์ดนั้น จะใช้โปรแกรมชื่อ Flash Loader ของ “STMicroelectronics” ซึ่งจะติดต่อกับ MCU ผ่าน Serial Port ของคอมพิวเตอร์ PC โดยโปรแกรมดังกล่าวสามารถดาวน์โหลดฟรีได้ที่ www.st.com

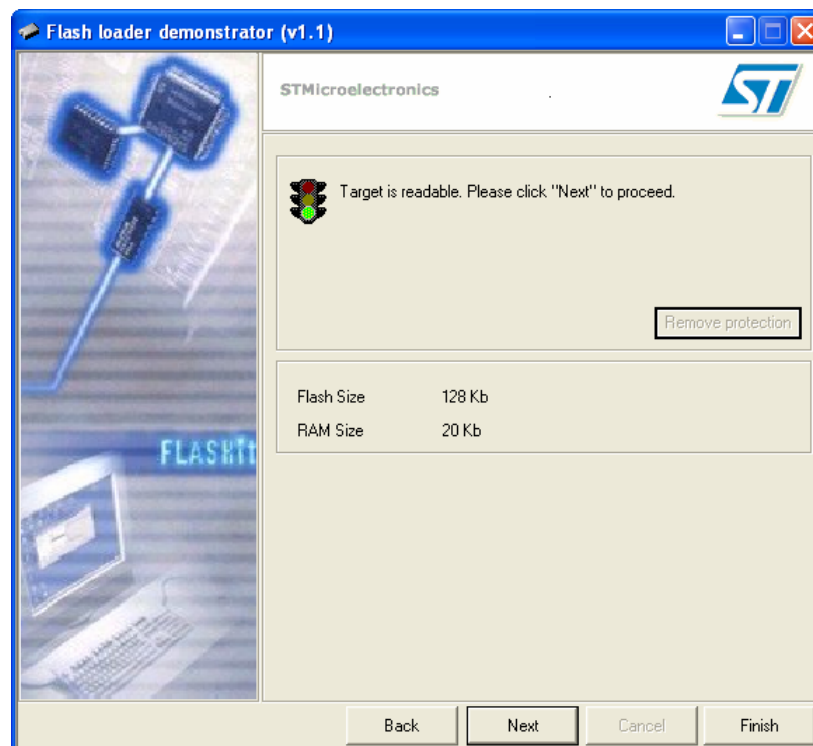
ขั้นตอนการ Download HEX File ให้กับ MCU

1. ต่อสายสัญญาณ RS232 ระหว่างพอร์ตสื่อสารอนุกรม RS232 ของ PC และบอร์ด USART1
2. จ่ายไฟเลี้ยงวงจรให้กับบอร์ด ซึ่งจะสังเกตเห็น LED PWR ติดสว่างให้เห็น
3. สั่ง Run โปรแกรม Flash loader ซึ่งถ้าเป็นรุ่น demo 1.1 จะได้ผลดังรูป

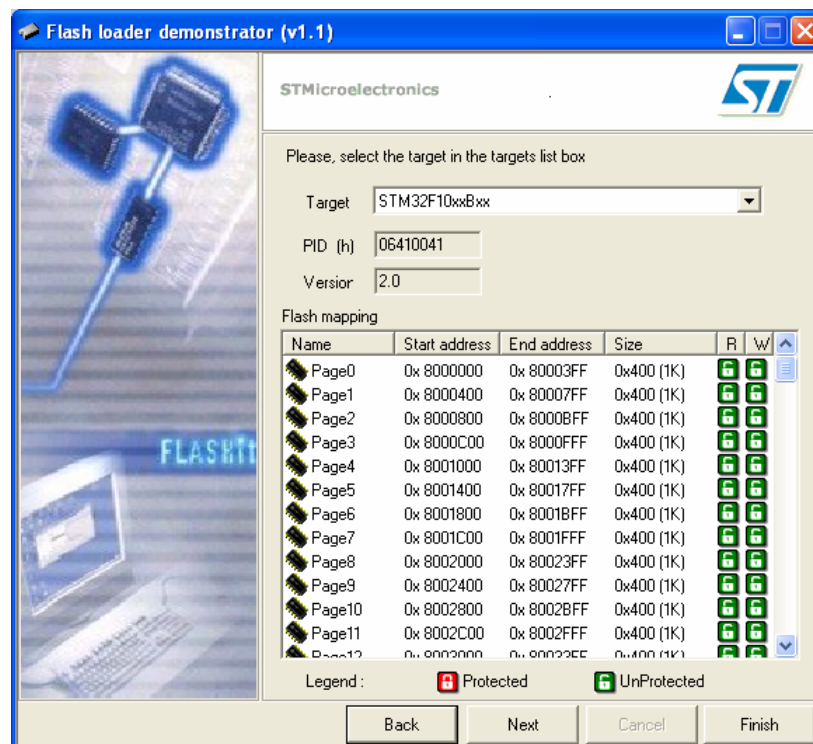


4. เริ่มต้นกำหนดค่าตัวเลือกต่างๆ ให้กับโปรแกรม ซึ่งในกรณีใช้กับบอร์ด ET-ARM STAMP STM32 ของ อีทีที ให้เลือกกำหนดค่าต่างๆ ให้โปรแกรมดังนี้
 - 4.1 เลือก COM Port ให้ตรงกับหมายเลข COM Port ที่ใช้งานจริง (ในตัวอย่างใช้ COM4)
 - 4.2 ตั้งค่า Baud Rate อยู่ระหว่าง 1200 - 115200 ซึ่งถ้าคอมพิวเตอร์ที่ใช้มีปัญหาเมื่อเลือกใช้ค่า Baud rate สูงๆ แล้วเกิด Error ให้ลดค่า Baud rate ให้ต่ำลง ตัวอย่างใช้ 115200
 - 4.3 กำหนดค่า Parity Bit ของพอร์ตอนุกรม แนะนำเป็น None
 - 4.4 กำหนดค่า Timeout ของการสื่อสารจากตัวอย่างเลือกเป็น 1 วินาที
 - 4.5 ทำการ RESET ให้ MCU ทำงานใน Boot Loader ตามขั้นตอนดังต่อไปนี้
 - เลือก Jumper BOOT1 ไว้ทางด้าน ISP (สัญญาณ BOOT1 = 0)
 - ON สวิตช์ BOOT0 จะเห็น LED BOOT0 ติดสว่าง (สัญญาณ BOOT0 = 1)
 - กดสวิตช์ RESET แล้วปล่อยเพื่อให้ CPU เริ่มทำงานใน Boot Mode

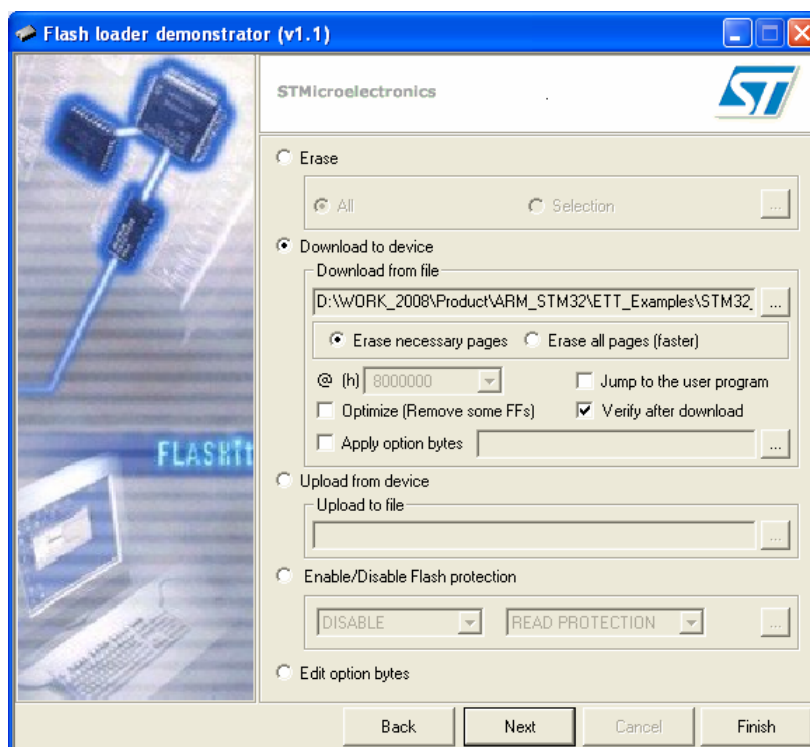
5. ให้คลิกเมาส์ที่ “Next” เพื่อเริ่มทำการ Download ซึ่งถ้าทุกอย่างถูกต้องโปรแกรมจะทำการอ่านค่าสถานะจาก CPU และรายงานผลให้ทราบ ซึ่งกรณีของ STM32F103RBT6 จะได้ดังรูป



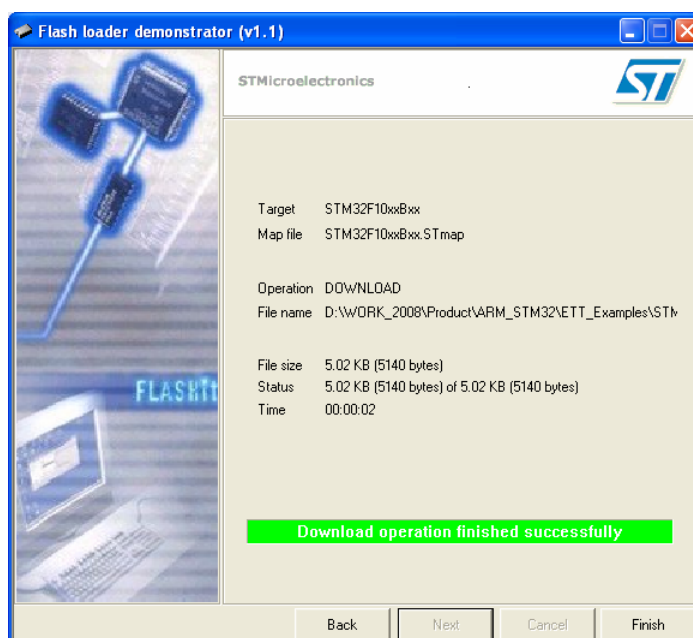
6. ให้ทำการคลิกเมาส์ที่ “Next” ซึ่งโปรแกรม Flash loader จะเริ่มแสดงค่าสถานะต่างๆที่อ่านได้จากตัว MCU ให้ทราบ ซึ่งในกรณีของ STM32F103RBT6 จะได้ดังรูป



7. ให้เลือก “Next” เพื่อไปยังขั้นตอนถัดไป



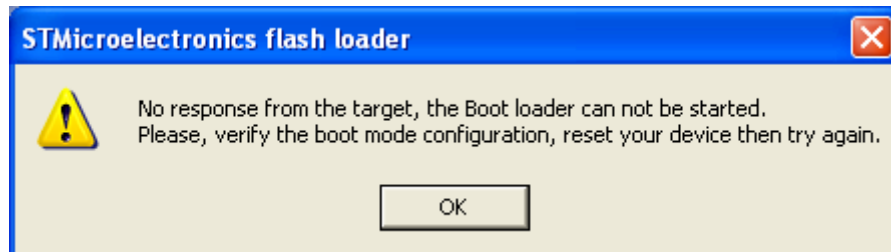
8. เลือก “Next” ซึ่งโปรแกรมจะเริ่มดำเนินการ Download ข้อมูลให้กับ MCU ทันที โดยสังเกตการทำงานที่ Status bar โดยในขั้นตอนนี้ให้รอจนกว่าการทำงานของโปรแกรมจะเสร็จสมบูรณ์ดังรูป



9. เมื่อทำงานของโปรแกรมเสร็จเรียบร้อยแล้ว ให้เลือก “Finish” พร้อมกับ OFF สวิตช์ BOOT0 ซึ่งจะสังเกตเห็น LED BOOT0 จะต้องดับ จากนั้นให้กดสวิตช์ Reset ที่บอร์ด ซึ่ง MCU จะเริ่มต้นทำงานตามโปรแกรมที่สั่ง Download ให้ทันที

ปัญหา Error ตอนสั่ง Download ด้วยโปรแกรม Flash loader V1.1

ในบางครั้งเมื่อจะสั่ง Download Hex File ให้กับ MCU ผ่านทาง Boot Loader ด้วย RS232 แล้ว โปรแกรมแจ้งข้อผิดพลาด ว่าไม่สามารถสื่อสารกับ Boot Loader ได้ ดังตัวอย่าง



ความผิดพลาดนี้เกิดจากการที่ โปรแกรม Flash Loader ไม่สามารถติดต่อสื่อสารกับ MCU ผ่านทาง Boot Loader ด้วย RS232 ได้ ซึ่งปัญหาจะเกิดจากหลายสาเหตุ ดังเช่น

- ผู้ใช้ยังไม่ได้สั่ง Reset MCU ให้เข้าทำงานใน Boot loader ก่อนจะเลือก “Next” ในขั้นตอนแรกของการสั่งงานโปรแกรม Flash Loader ซึ่งตรงนี้สามารถตรวจสอบได้ คือ Jumper BOOT1 ใต้ทางด้าน ISP และ SW-BOOT0 ต้องอยู่ในตำแหน่ง ON(LED BOOT0 ติดสว่าง)
- กำหนดหมายเลข Comport ไม่ตรงกับที่ต่อไว้จริง

แต่ถ้าผู้ใช้ปฏิบัติตามขั้นตอนต่างๆถูกต้องหมดแล้ว แต่ยังเกิดอาการ Error ดังกล่าวอยู่ แสดงว่าปัญหาอยู่ที่ความผิดพลาดของโปรแกรม Flash Loader เอง ซึ่งจากข้อสังเกต ที่ได้ทดลองใช้งานจริงๆแล้วพบว่า ปัญหาเกิดจาก มีข้อมูลขยะค้างอยู่ใน Buffer ของพอร์ต RS232 จึงทำให้โปรแกรม Flash Loader ทำงานผิดพลาด ซึ่งเกิดจากองค์ประกอบหลายอย่างดังนี้

- มีการใช้งานพอร์ตสื่อสาร RS232 สำหรับ Download และใช้งาน ช่องเดียวกัน แต่มีการกำหนดค่าพารามิเตอร์ที่แตกต่างกัน เช่น ตอนสั่ง Download ด้วยโปรแกรม Flash Loader ใช้ Baudrate เป็น 115200 พอ Download เสร็จแล้ว ในโปรแกรมการทำงานของ MCU มีการใช้งานรับส่งข้อมูลกับ RS232 ด้วยค่า Baudrate ที่ 9600 ซึ่งเป็นค่าที่แตกต่างกัน หรือมีการเสียบสาย Download ค้างไว้ตลอดเวลา พอทดลองโปรแกรมเสร็จแล้วจะกลับมา Download ใหม่ แล้วพบปัญหาดังกล่าวข้างต้น ซึ่งขอแนะนำในการแก้ปัญหาและลดความผิดพลาดความผิดพลาดจากสาเหตุนี้ ขอแนะนำให้ผู้ใช้งานแก้ปัญหาเบื้องต้นดังนี้
 - อย่ากำหนดค่า Baudrate ในการ Download และ ใช้งาน ด้วยค่าพารามิเตอร์ที่แตกต่างกัน ควรกำหนดให้ใช้ค่าเดียวกันจะดีที่สุด

- ถ้าการทดลองการทำงานของโปรแกรมที่เขียนขึ้นไม่มีการใช้งาน RS232 พอทำการ Download Code เสร็จแล้ว ให้ถอดสาย RS232 ออกจากบอร์ด อย่าเสียบสายค้างไว้ เพื่อลดการเกิดข้อมูลขยะใน Buffer ของ RS232
- ถ้า Code คำสั่งในโปรแกรมของ MCU มีการทำงานในลักษณะของการส่งพิมพ์ค่าข้อมูลออกมาทางพอร์ต RS232 ตลอดเวลา ก่อนที่จะเปิดโปรแกรม Flash Loader เพื่อสั่ง Download Code ในครั้งใหม่ ให้ทำการ Reset MCU ให้หยุดการส่งข้อมูลมายังพอร์ต RS232 เสียก่อน โดยการรีเซ็ตให้ทำงานใน Boot Loader รอไว้ก่อน แล้วจึงสั่งเปิดโปรแกรม Flash Loader
- ถ้าเป็น Converter แปลงพอร์ต USB/RS232 ก่อนสั่งเปิดโปรแกรม Flash Loader ให้ถอดสาย USB ออกแล้วเสียบกลับเข้าไปใหม่ เพื่อล้างข้อมูลต่างๆใน Buffer ทิ้งก่อน แล้วรอสักครู่ก่อน แล้วจึงสั่งเปิดโปรแกรม Flash Loader ขึ้นมาใหม่
- ถ้าเครื่องคอมพิวเตอร์ที่ใช้งานมีพอร์ต RS232 ให้ใช้งานมากกว่า 1 ชุด แนะนำให้แยกพอร์ต RS232 สำหรับ Download และใช้งาน ออกจากกันไปเลย

