REPORT

**제목 : 9조 6주차 실험결과 보고서**

**수강과목 : 임베디드 시스템 설계 및 실험**

**조 원 : 201524414 기호영**

**20152437 김요한**

**201624410 권선근**

**201824461 남지원**

**제출일자 : 2020.10.19**

1. **실험목표**
   1. Clock Tree의 이해 및 사용자 Clock 설정
   2. UART 통신의 원리를 배우고, 실제 설정 방법 파악
2. **배경지식**
3. Clock Tree

STM32의 내부 Clock의 흐름

HSI, HSE, PPL의 출력중 하나를 사용하여 System Clock을 출력

PLL은 HSI와 HSE를 곱하거나 나누어서 원하는 주파수를 출력

1. UART( Universal Asynchronous Receiver/Transmitter)  
   비동기 통신 프로토콜

Rx(데이터 수신)와 Tx(데이터 송신)의 교차 연결

서로의 Ground를 연결

baud rate를 일치시켜야 통신 가능

1. **실험방법**

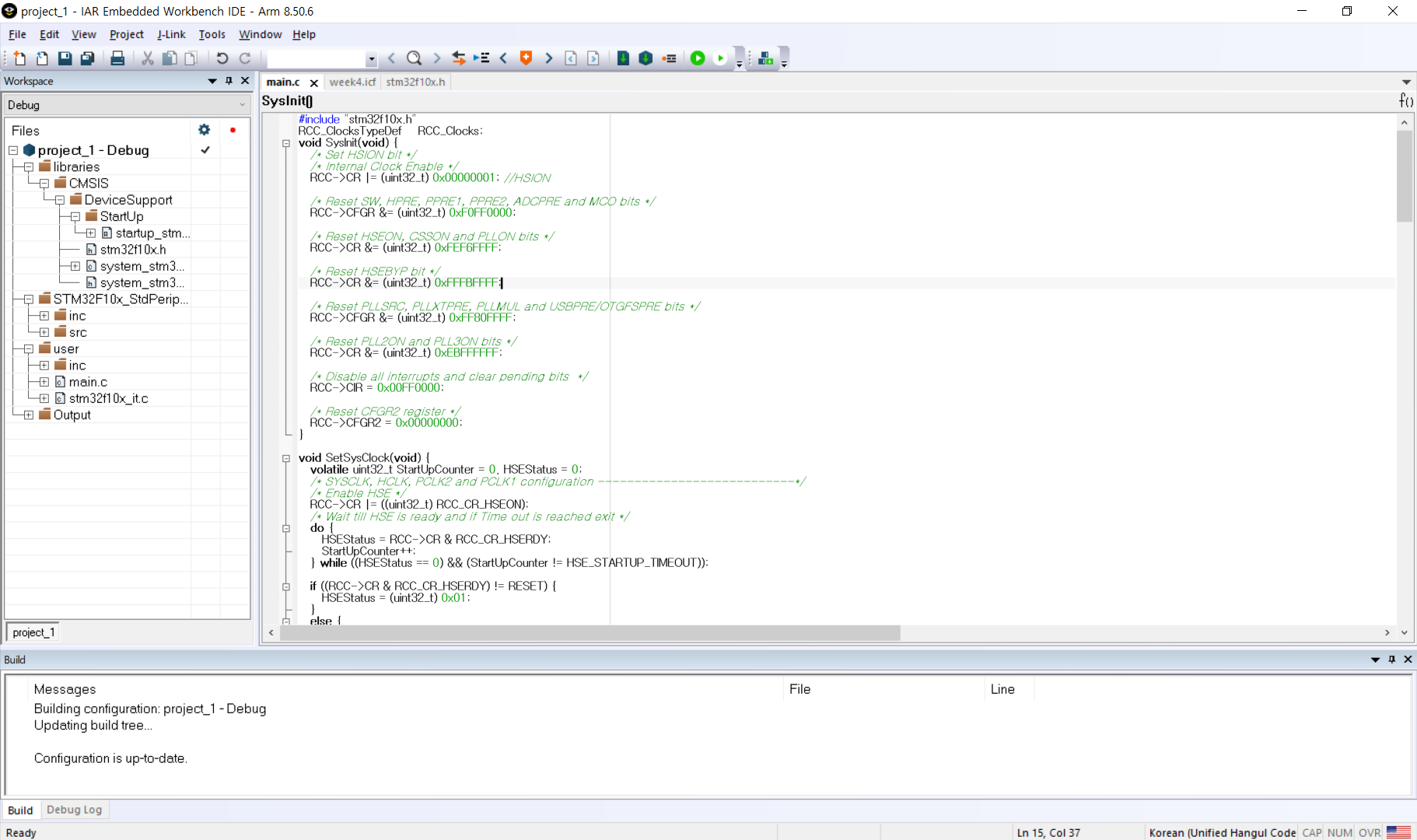
**Clock 을 이용한 시리얼 통신**

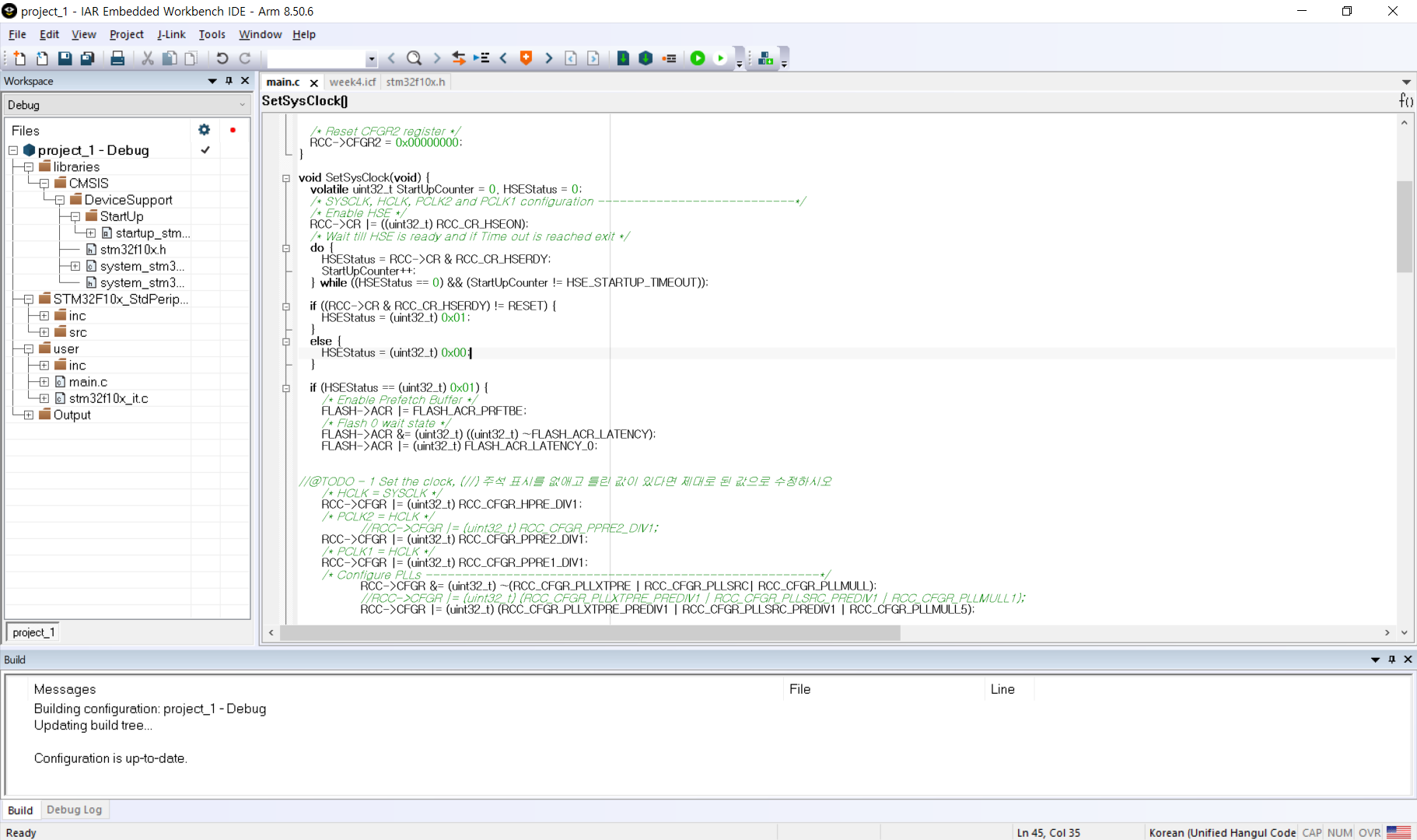
* PuTTY

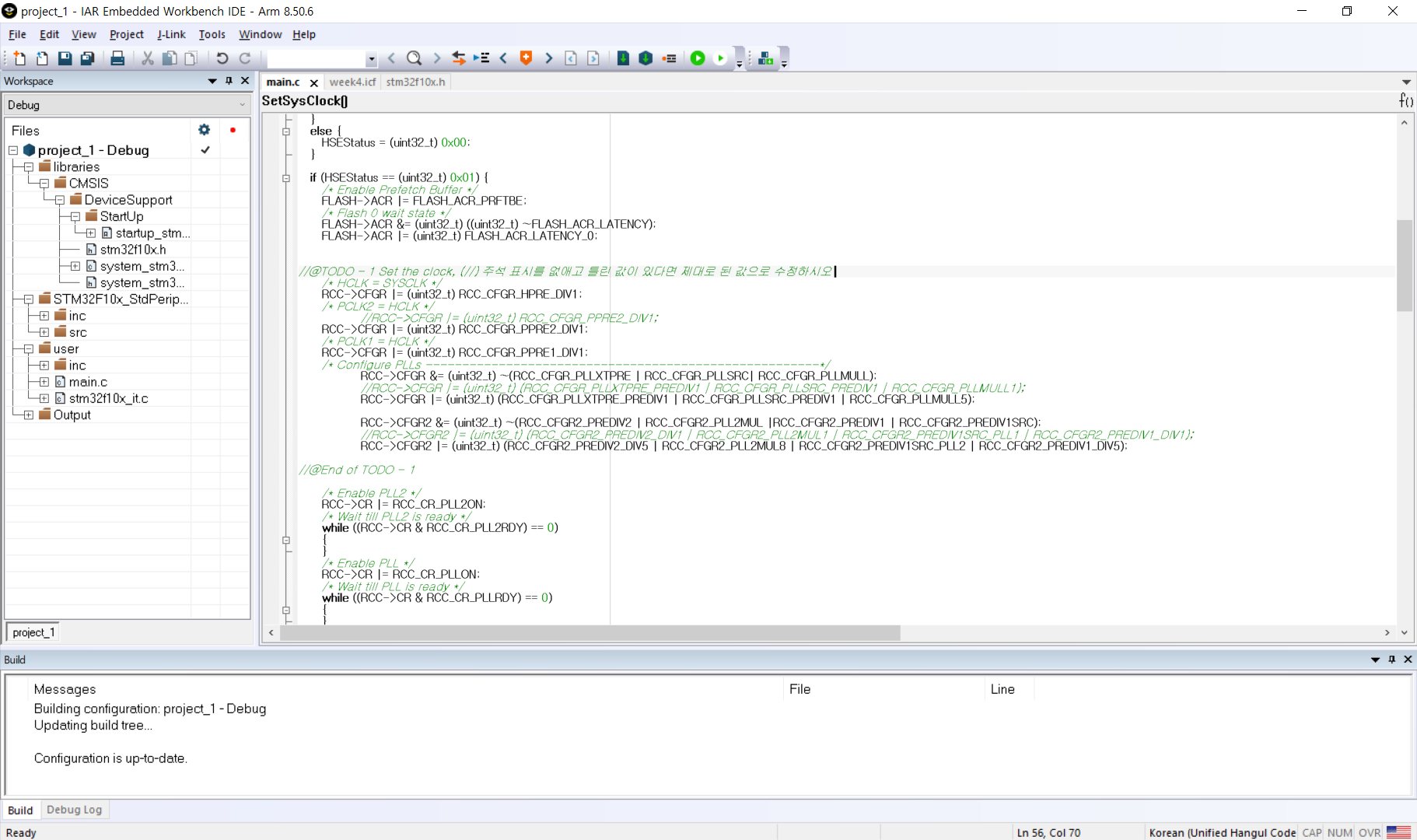
PuTTY는 SSH, 텔넷, rlogin, raw TCP를 위한 클라이언트로 동작하는 자유 및 오픈 소스 단말 에뮬레이터 응용 프로그램이다. PuTTY라는 이름에는 특별한 뜻이 없으나 tty는 유닉스 전통의 터미널의 이름을 가리키며 teletype를 짧게 줄인 것이다.

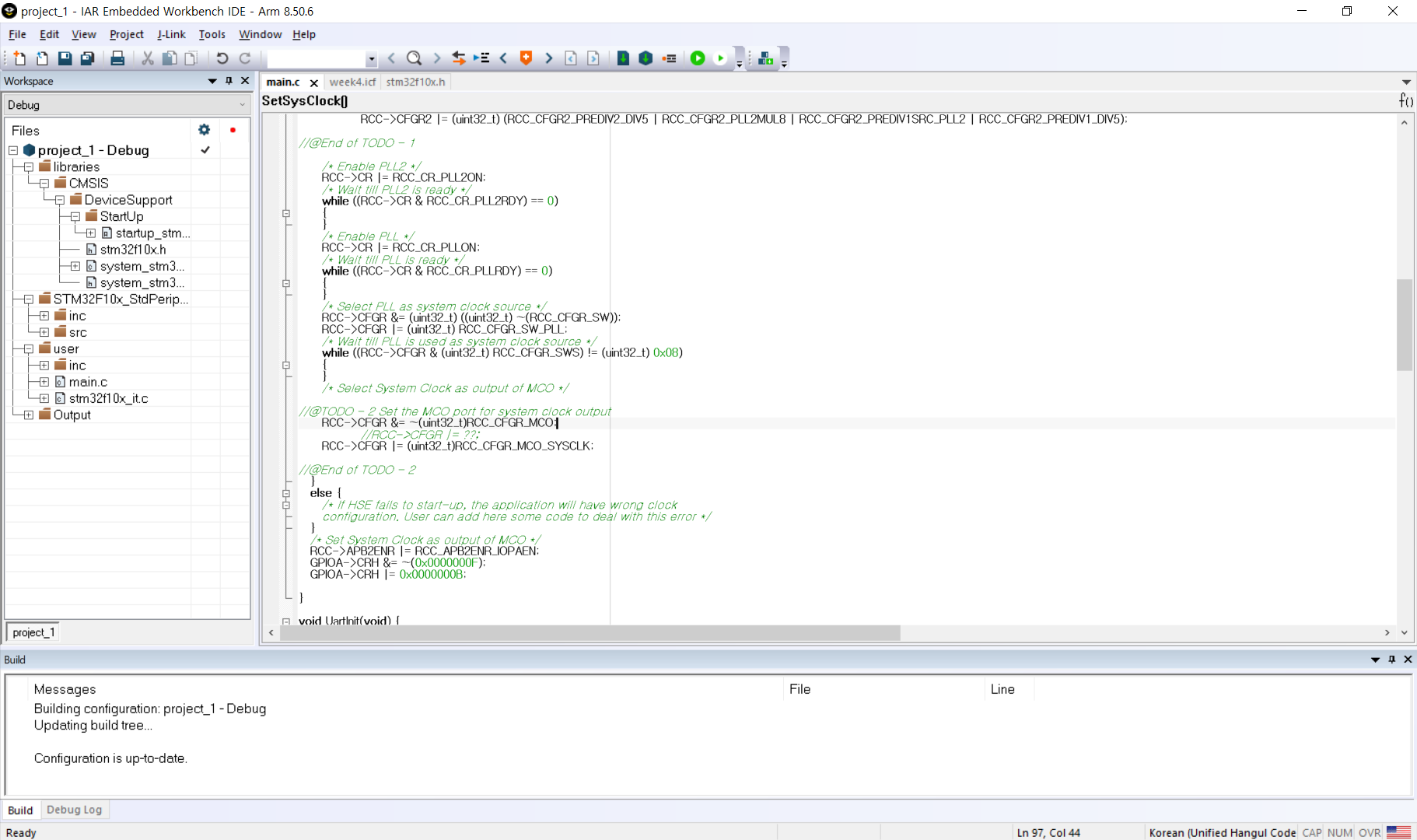
1. 주어진 Template.c 코드 작성

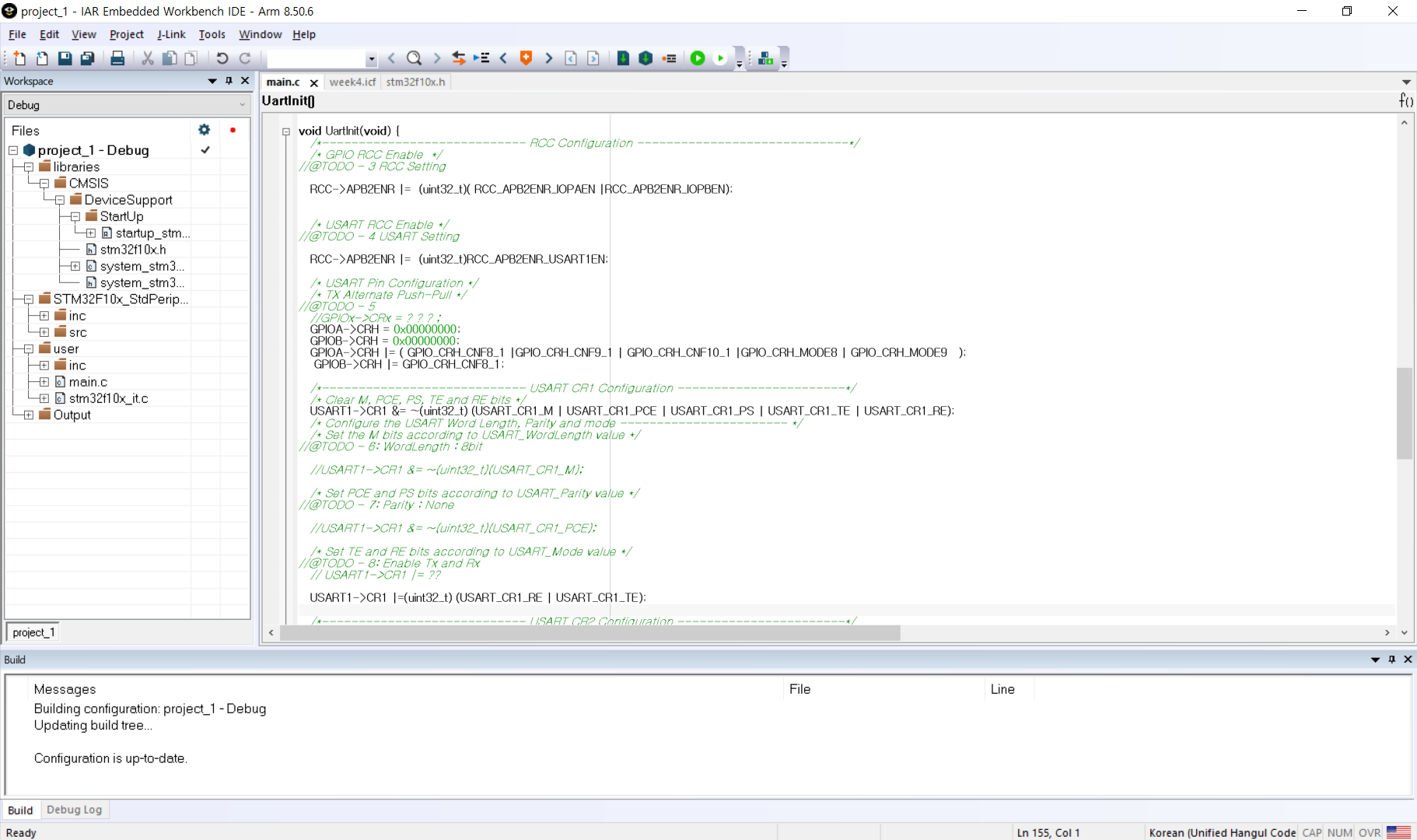
todo 부분의 코드를 수정

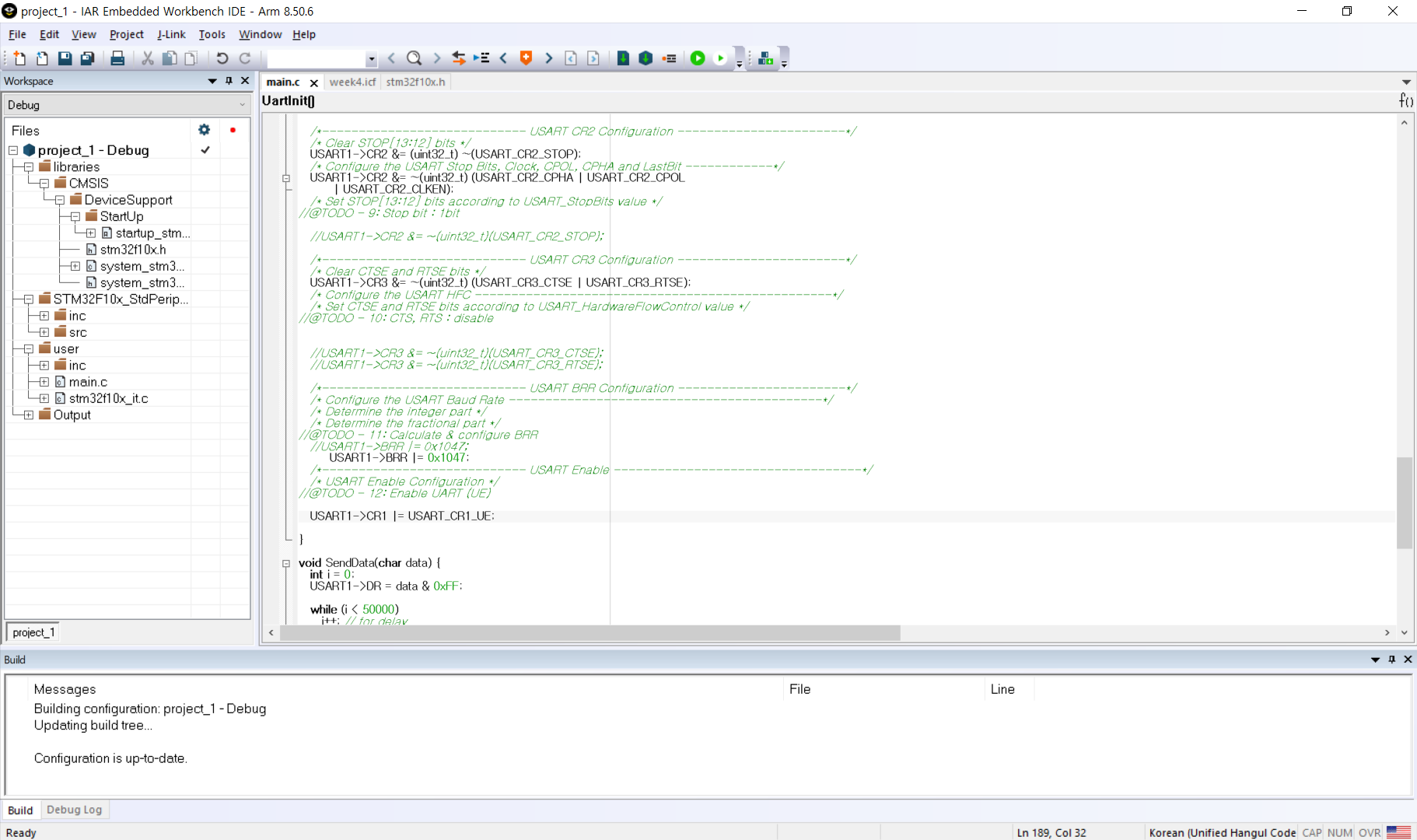


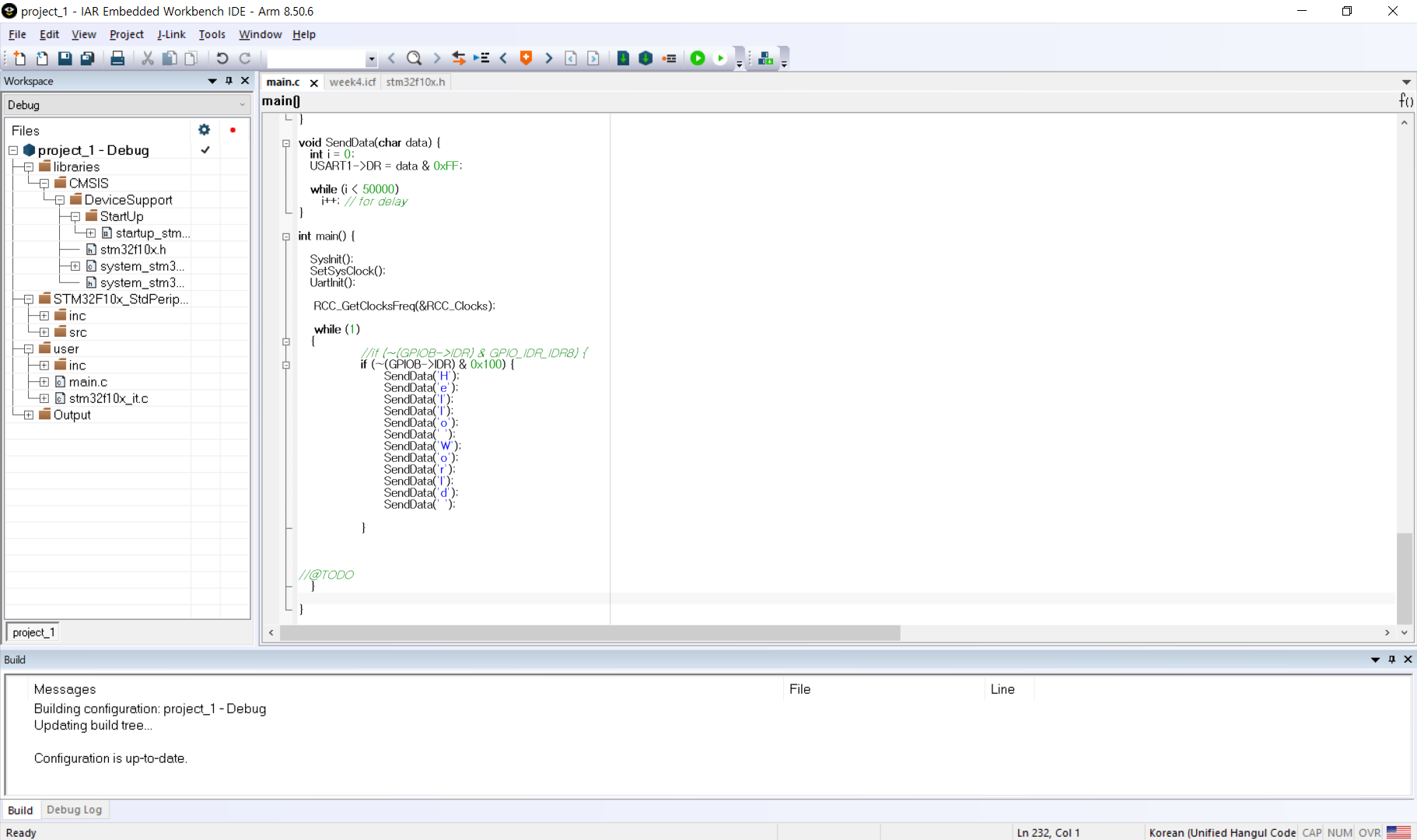












1. PuTTY 실행

pc 장치 관리자에 보드와 연결된 serial port를 확인하고 Connection Type을 serial로 선택하고 port와 Baud rate를 입력 후 Open을 클릭

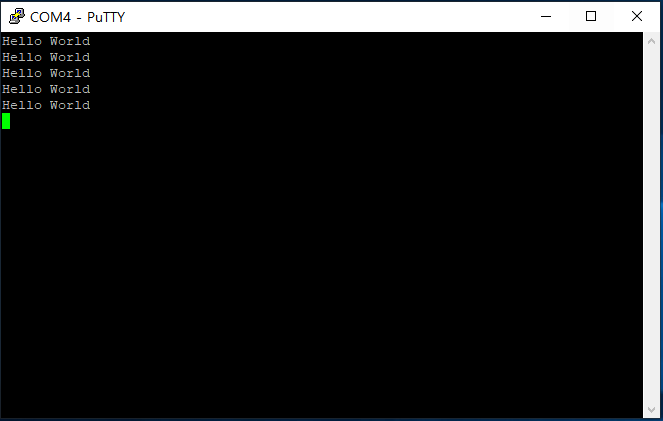
****

1. **실험 결과**

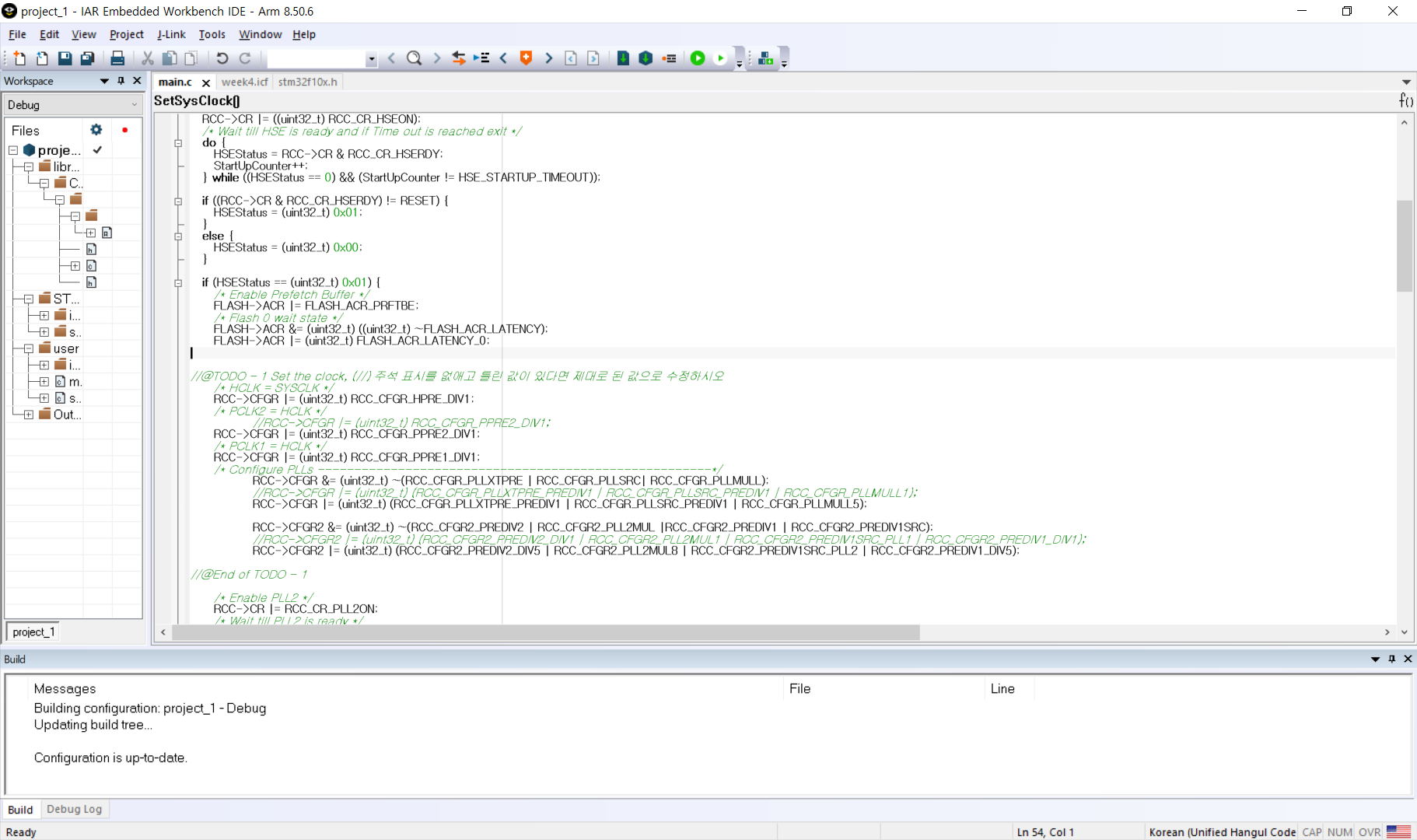
**A. 결과**

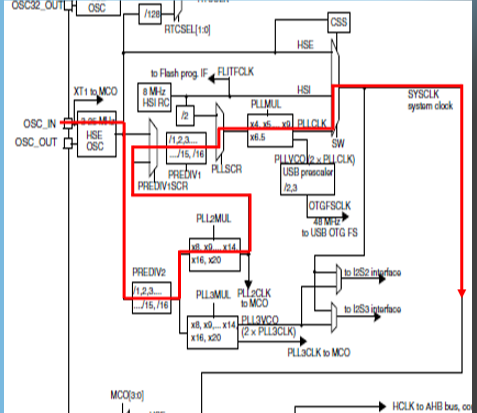
버튼을 누를 때마다 Putty를 통해 Hello world을 출력.

전체 화면



**B. 코드 분석**



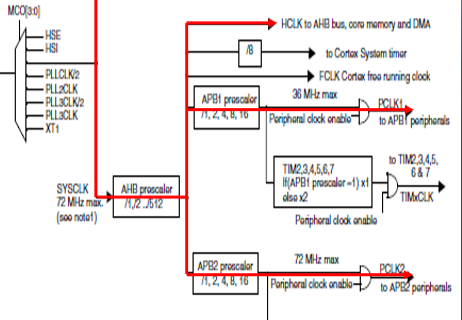


설정하고자 하는 SYSCLK이 40MHz이고 HSE OSC에서 25MHz가 나오므로 PREDIV2에서 /5를 PLL2MUL에서 \*8을 PREDIV1에서 /5를 계산해 주어야 하므로 주어진 template.c의

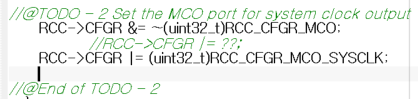
RCC->CFGR2 |= (uint32\_t)(RCC\_CFGR2\_PREDIV2\_DIV1 | RCC\_CFGR2\_PLL2MUL1 |

RCC\_CFGR2\_PREDIV1SRC\_PLL2 | RCC\_CFGR2\_PREDIV1\_DIV1);를

RCC->CFGR2 |= (uint32\_t) (RCC\_CFGR2\_PREDIV2\_DIV5 | RCC\_CFGR2\_PLL2MUL8 | RCC\_CFGR2\_PREDIV1SRC\_PLL2 | RCC\_CFGR2\_PREDIV1\_DIV5);로 수정합니다.

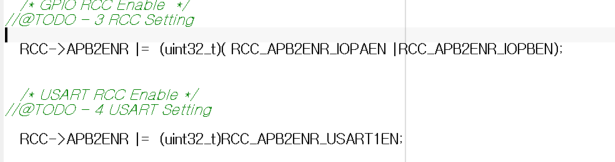


PCLK2로 40MHz를 내보내야 합니다.



MCO port를 SYSCLK의 output으로 set하기 위해

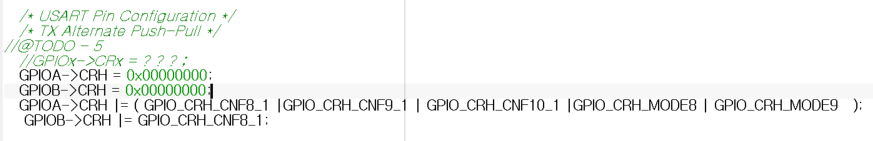
RCC->CFGR |= RCC\_CFGR\_MCO\_SYSCLK; 를 추가합니다.

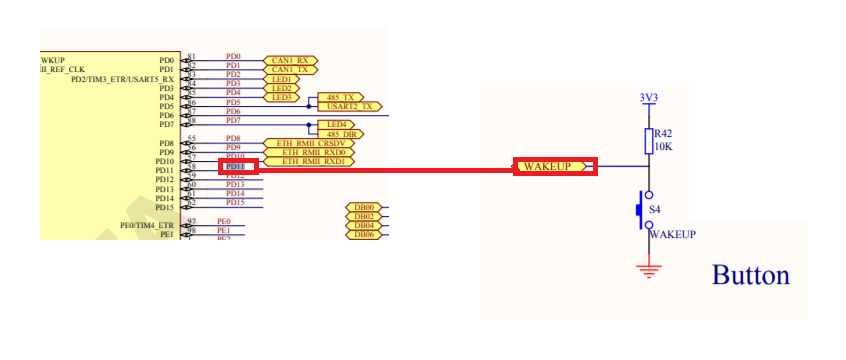
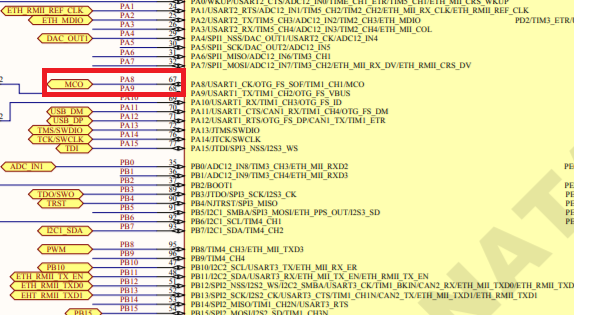


SYSCLK를 출력할 Port A와 버튼 입력을 받을 Port D 를 통신을 위해 USART 1을 enable 시키기 위해서

RCC->APB2ENR |= (RCC\_APB2ENR\_IOPAEN | RCC\_APB2ENR\_IOPDEN);

RCC->APB2ENR |= RCC\_APB2ENR\_USART1EN; 를 추가합니다.





S1버튼의 input 값을 받아드리고 MCO의 output 값을 내보내기 위해서

A와 D PORT를 reset하는

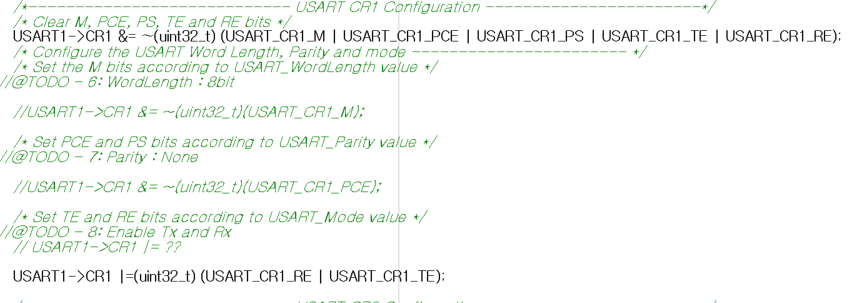
GPIOA->CRH = 0x00000000;

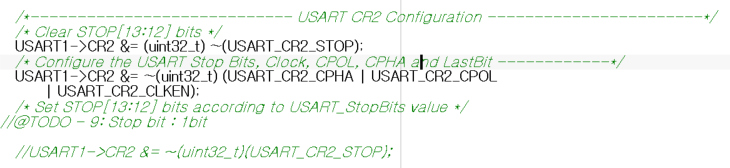
GPIOB->CRH = 0x00000000;

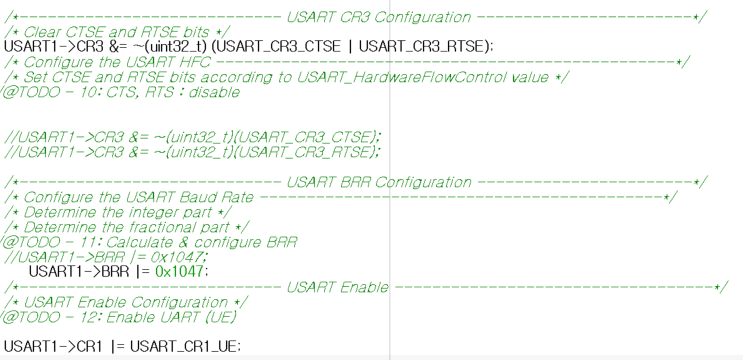
GPIOA->CRH |= ( GPIO\_CRH\_CNF8\_1 |GPIO\_CRH\_CNF9\_1 | GPIO\_CRH\_CNF10\_1 |GPIO\_CRH\_MODE8 | GPIO\_CRH\_MODE9 );

GPIOB->CRH |= GPIO\_CRH\_CNF8\_1;

코드를 추가 합니다.



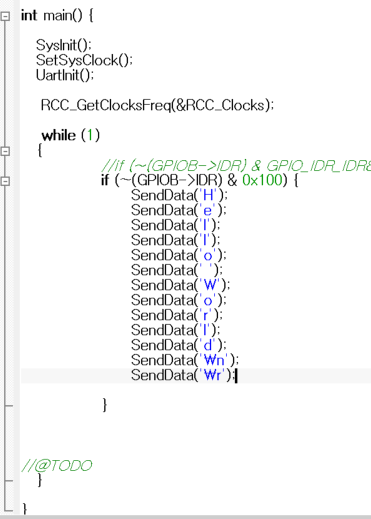




이므로 USART1->BRR |= 0x1047;를 추가합니다.

UART를 enable하기 위해서

USART1->CR1 |= USART\_CR1\_UE;를 추가합니다.



버튼을 눌러주는 동안 “Hello World”을 보내주어야 하므로 위와 같이 while문을 채웠습니다.

1. **결론**

Clock tree의 그림을 보고 이를 이용하여 SYSCLK과 PCLK2를 직접 설정하여 주고 Baud Rate 식을 이용하여 UART통신을 하기 위한 코드를 직접 짜봄으로써 Clock과 UART에 대한 이해와 지식이 더욱 늘어났습니다.