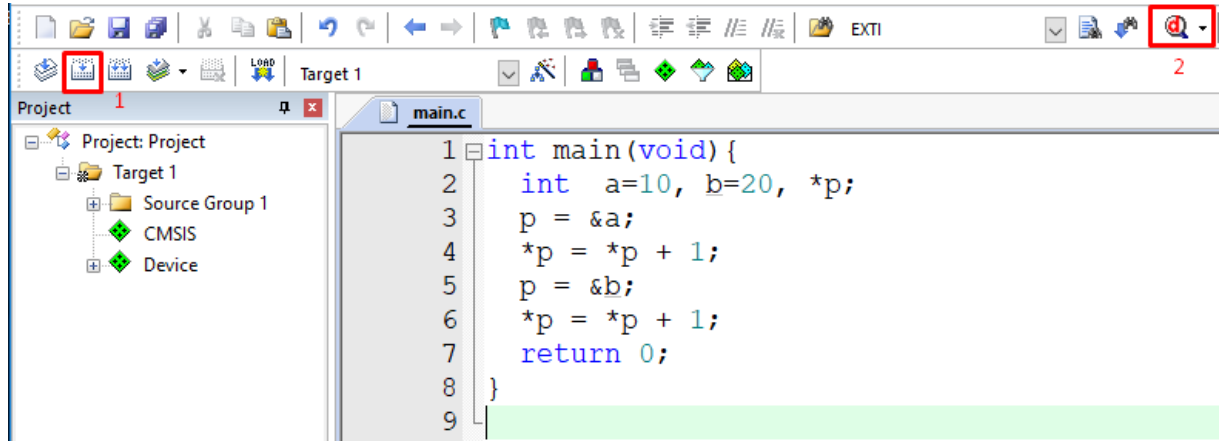
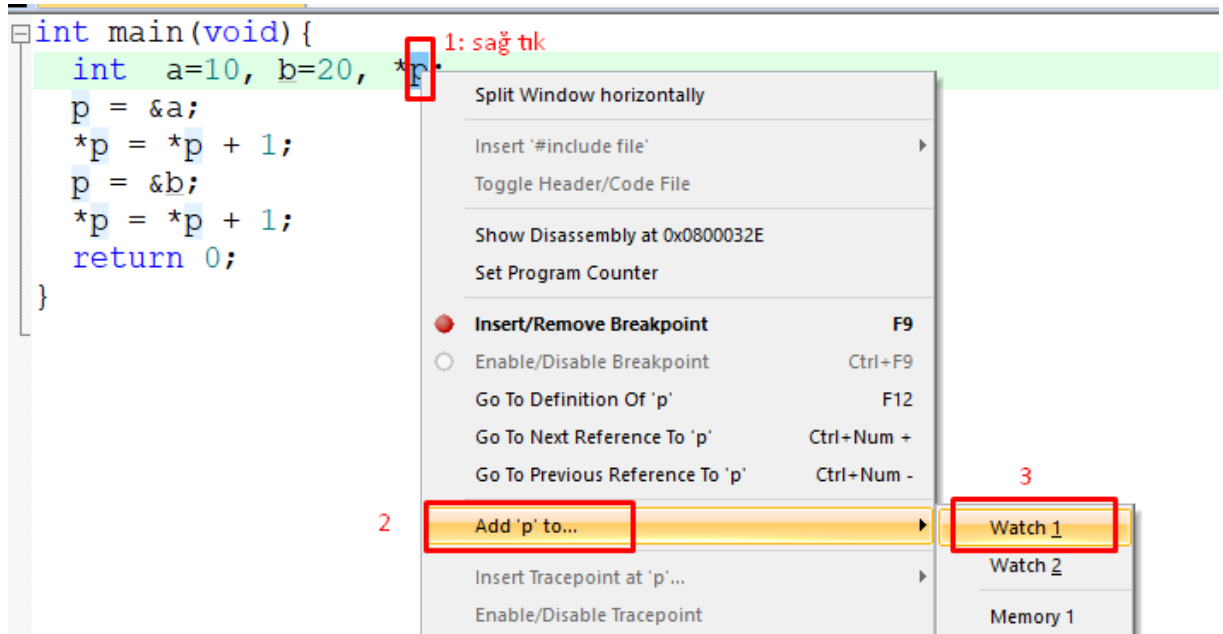


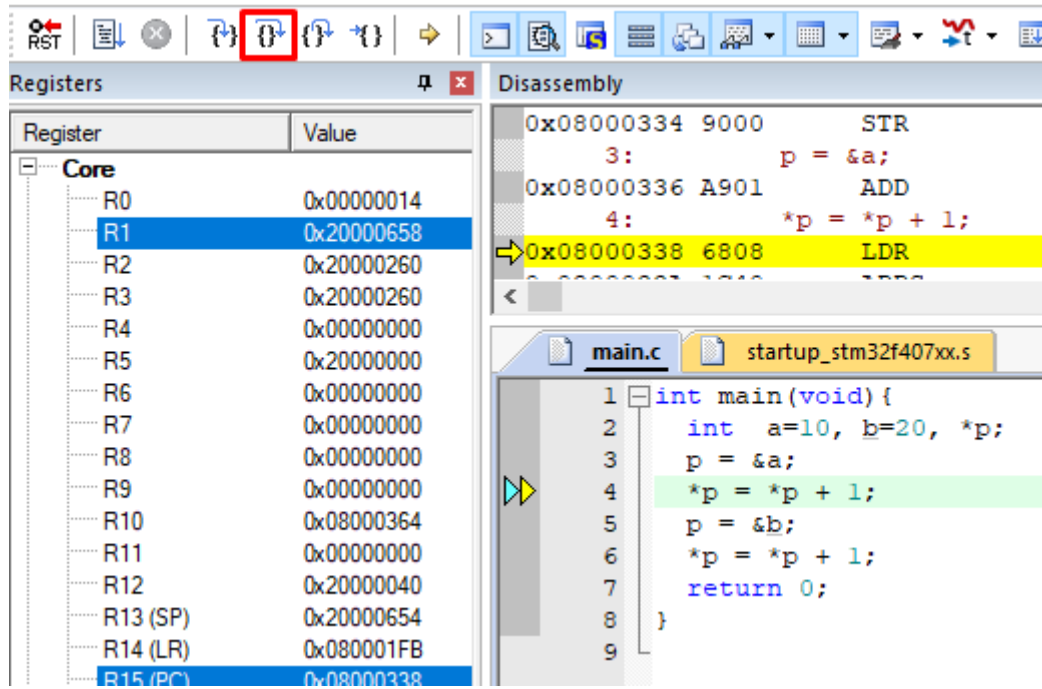
1 ile gösterilen buton (veya F7) ile kod derlendikten sonra Debug modunda (2 nolu buton) çalıştırılır.



a, b ve p değişkenleri seçilip sağ tıklanarak Watch Penceresine eklenir.



Programın 2. ve 3. satırı çalıştırıldığında:



The screenshot shows a debugger interface with two main windows: Registers and Disassembly. The Registers window lists registers R0 through R15, with R1 highlighted and showing a value of 0x20000658. The Disassembly window shows the assembly code for the main function, with the instruction 'LDR R1, [0x08000338, #6808]' highlighted. The C source code is also visible, showing the main function with variables a, b, and p.

Register	Value
R0	0x00000014
R1	0x20000658
R2	0x20000260
R3	0x20000260
R4	0x00000000
R5	0x20000000
R6	0x00000000
R7	0x00000000
R8	0x00000000
R9	0x00000000
R10	0x08000364
R11	0x00000000
R12	0x20000040
R13 (SP)	0x20000654
R14 (LR)	0x080001FB
R15 (PC)	0x08000338

```
0x08000334 9000 STR
3:          p = &a;
0x08000336 A901 ADD
4:          *p = *p + 1;
0x08000338 6808 LDR
<          [0x08000338, #6808]

main.c startup_stm32f407xx.s
1 int main(void) {
2     int a=10, b=20, *p;
3     p = &a;
4     *p = *p + 1;
5     p = &b;
6     *p = *p + 1;
7     return 0;
8 }
9
```

Watch penceresinde:

Watch 1		
Name	Value	Type
a	0x0000000A	int
b	0x00000014	int
p	0x20000658	int *
[0]	0x0000000A	int
< Enter expression >		

a değişkeninin bellekte 0x20000658 adresinde tutulduğu anlaşıyor. Bu adreste ise 0xA veya 10 tabanında 10 sayısı olduğu anlaşıyor. Bu durum Bellek Penceresinden de (Memory Windows) teyit edilebilir. Ulaşmak için:

