

```

/* Include the standard I/O header file */
#include <stdio.h>

int inGlobal;    /* inGlobal is a global variable because */
                /* it is declared outside of all blocks */

int main()
{
    int inLocal; /* inLocal, outLocalA, outLocalB are all */
    int outLocalA; /* local to main */
    int outLocalB;

    /* Initialize */
    inLocal = 5;
    inGlobal = 3;

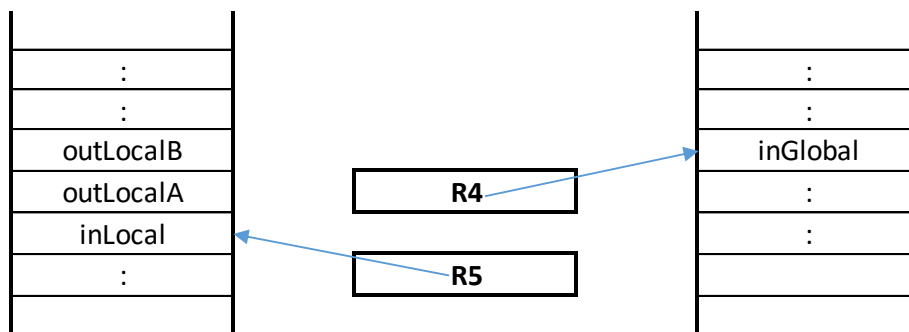
    /* Perform calculations */
    outLocalA = inLocal & ~inGlobal;
    outLocalB = (inLocal + inGlobal) - (inLocal - inGlobal);

    /* Print out results */
    printf("outLocalA = %d, outLocalB = %d\n", outLocalA, outLocalB);
}

```

Patt & Patel Figure 12.8

SYMBOL	SCOPE	ADDRESSING	
		Base Reg.	Offset
inGlobal	Global	R4	0
inLocal	main()	R5	0
outLocalA	main()	R5	-1
outLocalB	main()	R5	-2



```

;inLocal = 5
    AND  R0, R0, #0
    ADD  R0, R0, #5
    STR  R0, R5, #0

;inGlobal = 3
    AND  R0, R0, #0
    ADD  R0, R0, #3
    STR  R0, R4, #0

;outLocalA = inLocal & ~inGlobal
    LDR  R0, R5, #0    ;inLocal
    LDR  R1, R4, #0    ;inGlobal
    NOT  R1, R1
    AND  R2, R0, R1
    STR  R2, R5, #-1

;outLocalB = (inLocal + inGlobal) - (inLocal - inGlobal)
    LDR  R0, R5, #0
    LDR  R1, R4, #0
    ADD  R0, R0, R1    ;R0 = (inLocal + inGlobal)
    LDR  R2, R5, #0
    LDR  R3, R4, #0
    NOT  R3, R3
    ADD  R3, R3, #1
    ADD  R2, R2, R3    ;R2 = (inLocal - inGlobal)

    NOT  R2, R2
    ADD  R2, R2, #1
    ADD  R0, R0, R2
    STR  R0, R5, #-2    ;outLocalB = R0 - R2

```

Adaptation of Patt & Patel Figure 12.10