Preprocessor Use this feature of C to define registers and locations.

Example — define run mode clock gating registor as:

```
#define RCGCGPIO *((unsigned int *)0x400FE608u)
```

↓ a Macro

RCGCGPIO now uses 0x400FE608 for data manipulation. RCGCGPIO now means dereferencing of a pointer.

These macros are already declared by the vendors and can be used. These macros use "volatile". Difference between self created macros in our code and header files is in how we define them:

```
#define RCGC_GPIO *((unsigned int *)0x400FE608u)
```
→ how I defined it

```
#define SYSCTL_RCGCGPIO_R  (*((volatile unsigned long *)0x400FE608))
```
→ in header file

↓ tells the compiler that the object pointed to by address can change even though no statements in program change it.

Impact of volatile

when code optimization is set to "high" in IDE we see that the delay loops in the blink LED program do not work. Single stepping shows LED to change color but while loops are no longer present.

```
RCGC_GPIO = 0x20u;        //u to i

//Set bits/pins 1,2,3 as output i
GPIO_DIR = 0x0Eu;

GPIO_DEN = 0x0Eu;       //Digital

int count = 0;

while(1)
{
  count = 0;

  //For red LED
  GPIODATA_F = 0x02u;

  //provides delay of 1s
  while(count < 1000000)
  {
    count++;
  }

  count = 0;

  //For turning off all LED
  GPIODATA_F = 0x00u;

  //provides delay of 1s
  while(count < 1000000)
  {
    count++;
```

→ while loop not executed

```
RCGC_GPIO = 0x20u;        //u to indicate

//Set bits/pins 1,2,3 as output in GPIOD
GPIO_DIR = 0x0Eu;

GPIO_DEN = 0x0Eu;       //Digital functi

int count = 0;

while(1)
{
  count = 0;

  //For red LED
  GPIODATA_F = 0x02u;

  //provides delay of 1s
  while(count < 1000000)
  {
    count++;
  }

  count = 0;

  //For turning off all LED
  GPIODATA_F = 0x00u;

  //provides delay of 1s
  while(count < 1000000)
  {
    count++;
  }
```

but, if we declare that counter variable as volatile, the CPU does not remove it during optimization.

```c
int volatile count = 0;

while(1)
{
  count = 0;

  //For red LED
  GPIODATA_F = 0x02u;

  //provides delay of 1s
  while(count < 1000000)
  {
    count++;
  }

  count = 0;

  //For turning off all LED
  GPIODATA_F = 0x00u;

  //provides delay of 1s
  while(count < 1000000)
  {
    count++;
  }

  count = 0;

  //For green LED
  GPIODATA_F = 0x08u;
```

Count remains in a highly optimized code during single stepping.