**How to Set/Unset a Bit in a Flag**

Let us say we have a **flags** variable where each bit represents the following:

**flags = read bit, write bit, exec bit.**

So for read and execute, **flags** would look like:

**flags = 00000101**

Let’s define these bits:

|  |  |
| --- | --- |
| **READ\_BIT =** | **00000100** |
| **WRITE\_BIT =** | **00000010** |
| **EXEC\_BIT =** | **00000001** |

So **flags** is equal to:

|  |  |
| --- | --- |
| **flags =** | **READ\_BIT | EXEC\_BIT** |
| **=** | **00000100 | 00000001** |
| **=** | **00000101** |

Let’s say I want to unset the read bit. I take bitwise inverse of **READ\_BIT** and bitwise-AND it with **flags**:

|  |  |
| --- | --- |
| **flags =** | **00000101** |
| **READ\_BIT=** | **00000100** |
| **~READ\_BIT=** | **11111011** |
| **Flags & ~READ\_BIT=** | **00000101 &**  **11111011** |
| **flags=** | **00000001** |

The **READ\_BIT** has now been unset in **flags**.

Let’s say I want to set the read bit again, I would take **READ\_BIT** and bitwise-OR it with **flags**:

|  |  |
| --- | --- |
| **flags =** | **00000001** |
| **READ\_BIT=** | **00000100** |
| **Flags | READ\_BIT=** | **00000001 |**  **00000100** |
| **flags=** | **00000101** |

**flags** is back to where it was.