5/6/2021

**CO LAB-08**

**NAME: Amina Qadeer**

**SYNDICATE: CE-42-A**

**TEACHER: Rimsha Tariq**

**ROLL # 359607**

**Lab Tasks:**

**What is difference between arithmetic left shift and logical left shift?**

A **shift left logical** of one position moves each bit to the **left** by one. The low-order bit (the right-most bit) is replaced by a zero bit and the high-order bit (the **left**-most bit) is discarded. **Shifting** by two positions is the same as performing a one-position **shift** two times.

**Arithmetic left shifts** are equivalent to multiplication by a (positive, integral) power of the radix (e.g., a multiplication by a power of 2 for binary numbers). Logical **left shifts** are also equivalent, except multiplication and **arithmetic shifts** may trigger **arithmetic** overflow whereas logical **shifts do** not.

**Some differences**:

1. Arithmetic shift preserve sign bit, whereas Logical shift cannot preserve sign bit.
2. Arithmetic shift perform multiplication and division operation, whereas Logical shift perform only multiplication operation.
3. Arithmetic shift is used for signed interpretation, whereas Logical shift is used for unsigned interpretation.
4. With a logical shift, the bits of a word are shifted left or right. On one end, the bit shifted out is lost. On the other end, a 0 is shifted in.
5. The arithmetic shift operation treats the data as a signed integer and does not shift the sign bit. On a right arithmetic shift, the sign bit is replicated into the bit position to its right. On a left arithmetic shift, a logical left shift is performed on all bits but the sign bit, which is retained.

