|  |  |  |  |
| --- | --- | --- | --- |
| a | b | c | D |

|  |  |  |  |
| --- | --- | --- | --- |
| e | f | g | H |

|  |  |  |  |
| --- | --- | --- | --- |
| i | j | k | L |

0 1 2 3

0-31,32-63 0-31,32-63 0-31,32-63 0-31,32-63

Shift << by 40

Reduce 40, to 8, by discarding Limb 0

Limb 0 is discarded

Low 8 bits of Limb 1 is discarded by shifting Limb1 >> 8

The low 8 bits of Limb2 is used to set the upper 8 bits of Limb1

Limb0 = Limb1 >> 8

Limb0 |= (Limb2 << 56) >> 32

Limb1 = Limb2 >> 8

Limb1 |= (Limb3 << 56) >> 32

The left-shift operation discards the high-order bits that are outside the range of the result type and sets the low-order empty bit positions to zero, as the following example shows: