|  |
| --- |
| **Algorithm: Trade-off for DEC** |
| **Input:** R // Given remainder, mod by A |
| N // error data’s bits counts |
| **Parameter:** R1, R2, // SEC’s remainders |
| AWE1, AWE2 // Single AWE solved by SEC\_LUT |
| **Output:** AWE// Final computed value |
|  |
| **Procedure:** |
| 1. Initialize R1 ← 0, R2 ← 0, AWE1 ← 0, AWE2 ← 0; |
| 2. **for** p = 0 to N **do** |
| 3. **for** o in {0,1} **do** // o = 0: positive , o = 1: negative |
| 4. **if** o = 0 **then** |
| 5. AWE1 ← +2^p |
| 6. **else** |
| 7. AWE1 ← -2^p |
| 8. **end if** |
| 9. R1 ← SEC\_LUT[AWE1] |
| 10. R2 ← R - R1 |
| 11. **if** R2 < 0 **then** |
| 12. R2 ← A + R2 |
| 13. **end if** |
| 14. AWE2 ← SEC\_LUT[R2] |
| 15. **if** AWE2 is valid **then** // Check for a valid AWE2 |
| 16. break |
| 17. **end if** |
| 18. **end for** |
| 19. **end for** |
| 20. **return** AWE ← AWE1+AWE2 |