### 1. Convert 8-bit Binary to 3-byte Packed BCD

**Reference:** <https://en.wikipedia.org/wiki/Double_dabble>

**Example**. Convert 101101012 to 3-byte **packed BCD** using the *double dabble* algorithm

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **BCD Scratch Space** | | | **Original** | **Comment** |
| **Hundreds** | **Tens** | **Ones** |  |  |
| 0000 | 0000 | 0000 | 111100112 | Initialize |
| 0000 | 0000 | 0001 | 111001102 | Shift |
| 0000 | 0000 | 0011 | 110011002 | Shift |
| 0000 | 0000 | 0111 | 100110002 | Shift |
| 0000 | 0000 | 1010 | 100110002 | Add 3 |
| 0000 | 0001 | 0101 | 001100002 | Shift |
| 0000 | 0001 | 1000 | 001100002 | Add 3 |
| 0000 | 0011 | 0000 | 011000002 | Shift |
| 0000 | 0110 | 0000 | 110000002 | Shift |
| 0000 | 1001 | 0000 | 110000002 | Add 3 |
| 0001 | 0010 | 0001 | 100000002 | Shift |
| **0010** | **0100** | **0011** | **000000002** | **Shift** |
| **2** | **4** | **3** |  |  |

### 2. Double Dabble. Your Turn.

Convert 15910 to 3-byte **packed BCD** using the *double dabble* algorithm

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **BCD Scratch Space** | | | **Original** | **Comment** |
| **Hundreds** | **Tens** | **Ones** |  |  |
| 0000 | 0000 | 0000 | 1001 11112 | Initialize |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
| **1** | **5** | **9** |  |  |

### 3. 16-Bit to 5-Digit Packed BCD. Try It.

Convert 102310 to 5-byte **packed BCD** using the *double dabble* algorithm

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **BCD Scratch Space** | | | | | **Original** | | **Comment** |
| **10000s** | **1000s** | **100s** | **10s** | **1s** | **High** | **Low** |  |
| 0000 | 0000 | 0000 | 0000 | 0000 | 0000 00112 | 1111 11112 | Initialize |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| **0** | **1** | **0** | **2** | **3** |  | |  |