Algorithm: Bits counter

**Input:** EPC Sequence: *S = (S1, S2, …, Sn)*

**Output:** Number of bit 0: *zero*

Number of bit 1: *one*

*zero* 🡨 0, *one* 🡨 0

Convert hexadecimal to binary bits (without discarding 0 bits in front of the binary bits)

**or** *bits* = hex2bin(*S*)

*len* 🡨 length(*bits*), *i* 🡨 0

**while** *i* < *len* **do**

**if** *i* = 0 **then**

*prev* 🡨 0, *shared\_bit* 🡨 *false*

**else**

*prev* 🡨 *bits*(*i* - 1)

*curr* 🡨 *bits*(*i*)

**if** *curr* = 0 **then**

*zero* 🡨 *zero* + 4

**if** *prev* = 1 **and** *shared\_bit* = *true* **then**

*shared\_bit* 🡨 *true*

**else**

*shared\_bit* 🡨 *false*

**end if**

**else**

*one* 🡨 *one* + 1, *zero* 🡨 *zero* + 2

**if** *shared\_bit* = *true* **then**

*zero* 🡨 *zero* + 1

**end if**

*shared\_bit* 🡨 **not** *shared\_bit*

**end if**

*i* 🡨 *i +* 1

**end while**