Checksum calculation

Checksum of a block of data is the complement of the one's complement of the 16-bit sum of the block. If checksum is included in the block of data, the new block of data will have its checksum zero. This is how checksum is used for error detection in datagram transmissions.
As an example, consider a block of data:

    0x23fb
    0x34c0
    0xa090
    0xbcaf
    0xfc05

Sum 0x2b1ff. To calculate 16-bit 1's complement sum, the excess digit 2 needs to be added back to the least significant 16 bits:

```
0xb1ff+
     2
----------
0xb201
```

The complement of this is 0x4dfe. This is the checksum of the block of data. Note that checksum(checksum+data)=0. So, if we transmit the block of data including the checksum field, the receiver should see a checksum of 0 if there are no bit errors.

UDP checksum is optional. It is calculated over the header and data, after attaching a pseudo-header consisting of source and destination IPs and the Protocol field (0x11) plus the total length field.

Example:

UDP datagram plus IP header:

```
4500 004a 6581 4000 4011 6eca c0a8 0065
44a8 60a2 8206 0035 0036 XXX 79d0 0100
0001 0000 0000 0000 0331 3135 0331 3031
0331 3938 0331 3332 0769 6e2d 6164 6472
0461 7270 6100 000c 0001
```

Pseudo Header

```
c0a8 0065
44a8 60a2
0011 0036  0x1669E
```

UDP Datagram

```
8206 0035
0036 0000
79d0 0100
```

```
0001 0000
0000 0000  0xFD42
0331 3135
0331 3031
0331 3938
0331 3332
0769 6e2d  0x1502A
6164 6472
0461 7270
6100 000c
0001       0x19DB4
           -------------
           0x551BE
```

So, the 1's complement sum is

```
0x51BE+
      5
-----------
0x51C3 -------> 1's complement of this is
                          0xAE3C, the checksum.
```