# GPS

U-blox CAM-M8 GNSS Reciever

<https://www.u-blox.com/sites/default/files/CAM-M8-FW3_DataSheet_%28UBX-15031574%29.pdf>

<https://www.u-blox.com/sites/default/files/products/documents/u-blox8-M8_ReceiverDescrProtSpec_%28UBX-13003221%29_Public.pdf>

<https://www.youtube.com/watch?v=TwhCX0c8Xe0>

## DDC

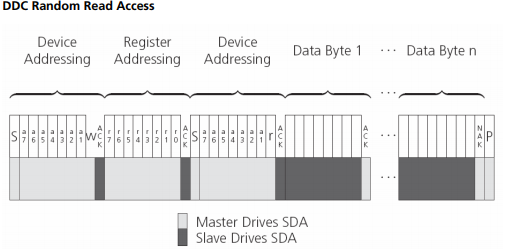
PG. 49 of link 2 describes the protocol. It is consistent with I2C I have used.

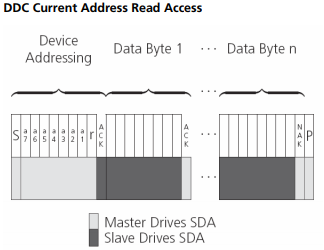
**Address**

#define GPS\_ADDR 0x42

**No Message Waiting**

0xFF means no message data is waiting



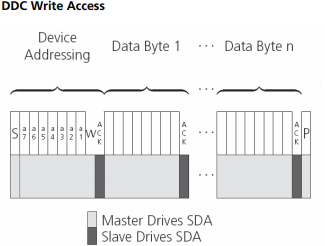


Only three addresses are readable:

0xFD, Number of bytes available (high byte)

0xFE; Number of bytes available (low byte)

0xFF; Data Stream



Register set above is not writable, instead use UBX messages.

## UBX Protocol (pg. 150)

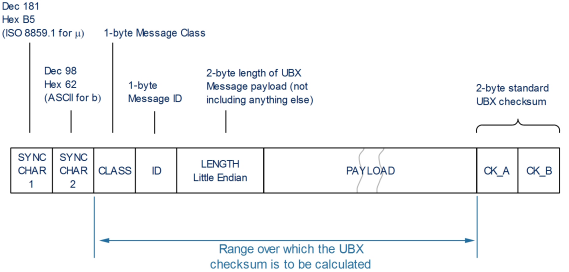
u-blox receivers support a u-blox proprietary protocol to communicate with a host computer. This protocol has the following key features:

• Compact - uses 8-bit Binary Data.

• Checksum Protected - uses a low-overhead checksum algorithm

• Modular - uses a 2-stage message identifier (Class and Message ID)

**Frame Structure**



• Every Frame starts with a 2-byte Preamble consisting of two synchronization characters: 0xB5 0x62.

• A 1-byte Message Class field follows. A Class is a group of messages that are related to each other.

• A 1-byte Message ID field defines the message that is to follow.

• A 2-byte Length field follows. The length is defined as being that of the payload only. It does not include the Preamble, Message Class, Message ID, Length, or CRC fields. The number format of the length field is a Little-Endian unsigned 16-bit integer.

• The Payload field contains a variable number of bytes.

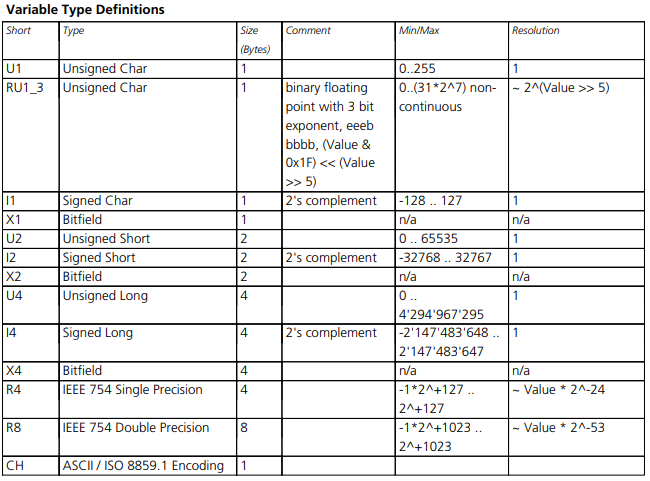
• The two 1-byte CK\_A and CK\_B fields hold a 16-bit checksum whose calculation is defined on page 152. This concludes the Frame.

**UBX Polling**

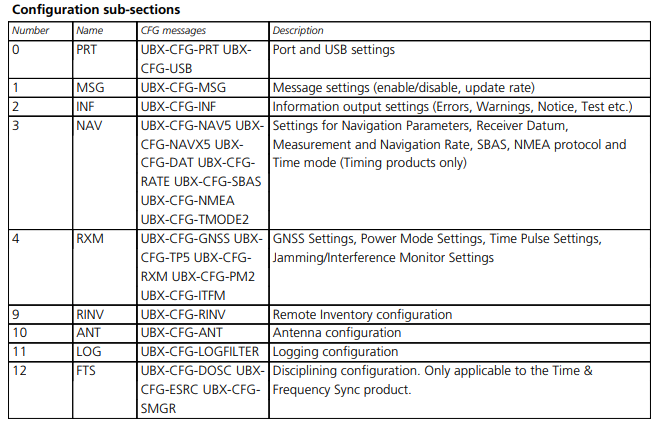
All messages that are output by the receiver in a periodic manner (i.e. messages in classes MON, NAV and RXM) and Get/Set type messages, such as the configuration messages in the CFG class, can also be polled. The UBX protocol is designed so that messages can be polled by sending the message required to the receiver but without a payload (or with just a single parameter that identifies the poll request). The receiver then responds with the same message with the payload populated.

**UBX Checksum**

<https://portal.u-blox.com/s/question/0D52p00008HKCchCAH/evkm8l-ubx-checksum>



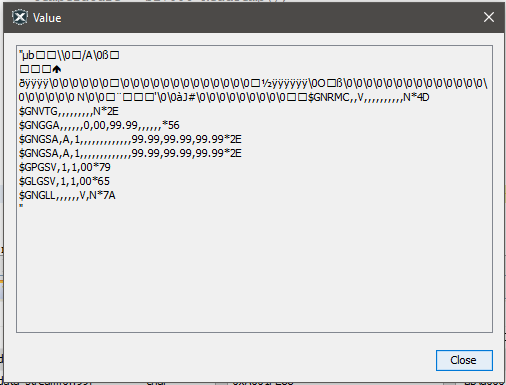
## Configuration



# Thing I will be using it for:

## Real Time Clock

* UTC 9.6 (pg 41) give time format
* 33.17.3 UBX-NAV-CLOCK (0x01 0x22)
* Time Pulse generation
* Reading Latitude and Longitude
  + 32.1.5 (pg 124) latitude and longitude format



NAV-PVT info indoors