

Message Structure



Valid messages are composed of header, payload and checksum parts. The header's size is 6 bytes containing 2 sync bytes, message ID, message sub-ID and payload length. The checksum shall be computed over the payload part using the algorithm below. Note that all multi-byte entries are written in little-endian format.

Checksum computation algorithm

Please use the function below to do the parity-check and make sure the message is valid

```
checksum_A = checksum_B = 0;

for (i = 0; i < payload_length; ++i)
{
    checksum_A += payload[i];
    checksum_B += checksum_A;
}
```

2.1 Position Message

Field	Type	Description	Content
1	uint8_t	Sync 1	0xAA
2	uint8_t	Sync 2	0x28
3	uint8_t	Message type	0x02
4	uint8_t	Message sub-ID	0x01
5	uint16_t	Payload length	36
6	double	Latitude	radians
7	double	Longitude	radians
8	double	Height	m
9	float[3]	ECEF Coordinate (x, y, z)	km
10	uint8_t	Checksum A	
11	uint8_t	Checksum B	

2.2 System Configuration Message

Field	Type	Description	Content
1	uint8_t	Sync 1	0xAA
2	uint8_t	Sync 2	0x28
3	uint8_t	Message type	0x02
4	uint8_t	Message sub-ID	0x02
5	uint16_t	Payload length	7
6	int16_t	Product ID	
7	bit field	see Table1	
8	float	Antenna Separation	m
9	uint8_t	Checksum A	
10	uint8_t	Checksum B	

Table 1 : Bit field definition

bit:	7	6	5	4	3	2	1	0
Content:	Satellite Type		GNSS Flag	Device Type			Output Type	

2.3 Measurement Message

Field	Type	Description	Content
1	uint8_t	Sync 1	0xAA
2	uint8_t	Sync 2	0x28
3	uint8_t	Message type	0x02
4	uint8_t	Message sub-ID	0x03
5	uint16_t	Payload length	19
6	int32_t	Count	
7	char[5]	Text	
8	short	Width	mm
9	double	Length	m
10	uint8_t	Checksum A	
11	uint8_t	Checksum B	

Question: Please decode the binary file "data.dat" based on the message format 2.1, 2.2 and 2.3. Note that the binary data contains some junk messages. Write the decoded valid content to three txt files.