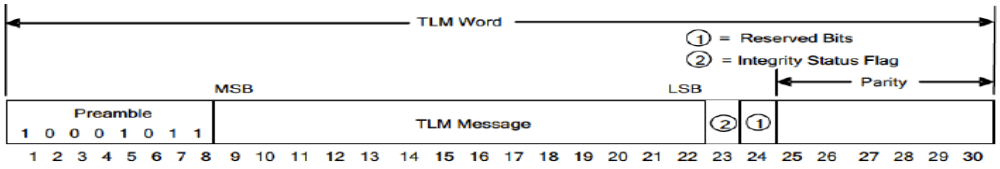
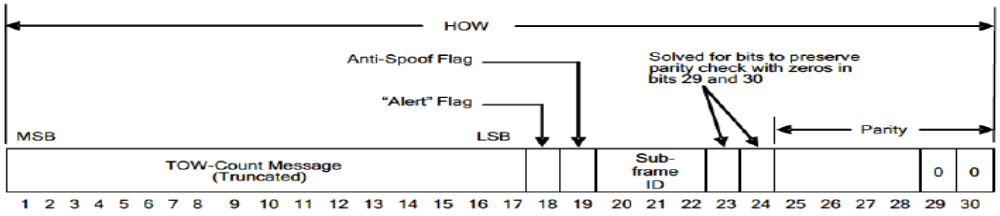
Satellite Navigation

To extract the information we need we have to determine which sub-frame we have to use. The GPS navigation message contains the info. In all the Sub-frames 1, 2 & 3 we notice TLM & HOW all of them have this. In side this we could see the information about the sub-frame.

TLM and HOW:





[1]

The sub-frame id from the second graph start from 20 and end at 22 and from the one above end at 30. Thus, the sub-frame start from the row number 50 in the Sub-frame matrix and end at 52. Which is 3 bits.

subframe <300x5 double>

	1	2	3	4	5
43	0	1	1	1	1
44	1	0	0	0	0
45	1	0	0	0	0
46	1	0	0	1	1
47	1	0	1	0	1
48	0	0	0	0	0
49	1	1	1	1	1
50	0	0	1	1	0
51	1	1	0	0	0
52	0	1	0	1	1

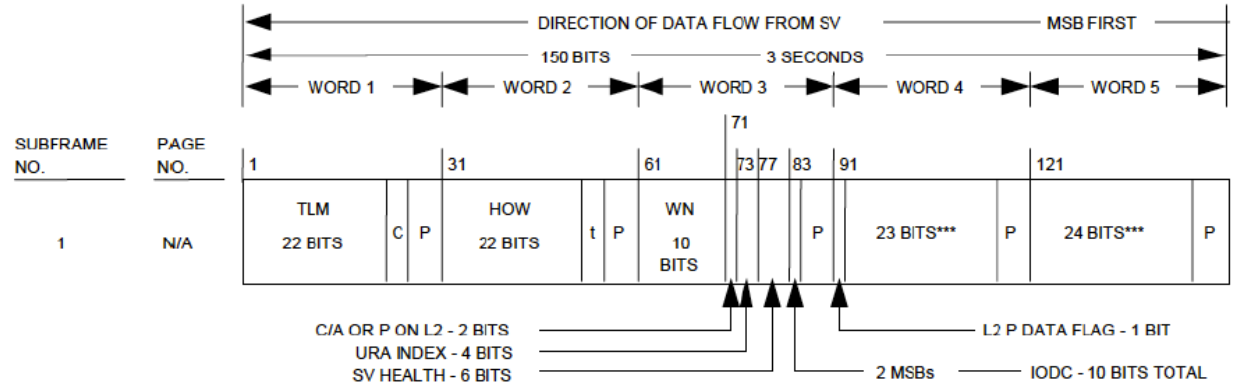
We compare this with binary number of 3 bits

Subframe	ID Code
1	001
2	010
3	011
4	100
5	101

[1]

So we get the first column belongs to the second sub-frame and so on. And the fifth column belong to the first sub-frame.

GPS Week



From the subframe number 1, we observe that the parameter “GPS Week” is situated in the interval 61-70 (10 bits), every bits having a different value (0 or 1).

We used the “flipud” command in order to return the WK with the order of elements flipped upside down along the first dimension. We will obtain the followings value, for each bit.

	1
1	0
2	0
3	1
4	1
5	0
6	0
7	1
8	0
9	0
10	1

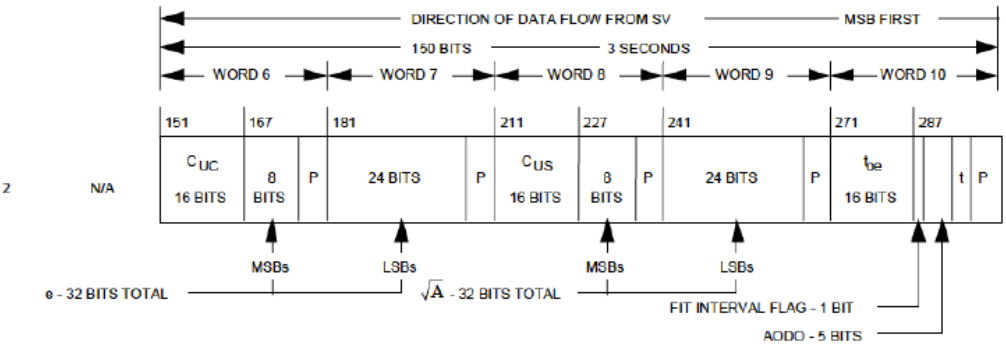
Than we wrote a loop, which will multiply the value “1” with different power of 2. The power of 2, will be the number before of the bit corresponding to the respective value 1. For example, in the figure above, the 3rd bit has value one.

```
% Week Gps (10 bits) subframe(1)
WK = subframe(61:70,5);
WK = flipud(WK);
t = zeros(length(WK),1)';
for i =1:length(WK)
    if WK(i)==1
        t(i) = WK(i)*2^(i-1);
    else
        t(i) = WK(i);
    end
end
WeekGPS = sum(t)+1024;
```

The final value for “GPS Week” parameter will be obtained by summing all values described before, adding 1024 (the first cycle of measurements, from 1980), which is the maximum value for 10 bits.

Toe - ephemeris reference epoch

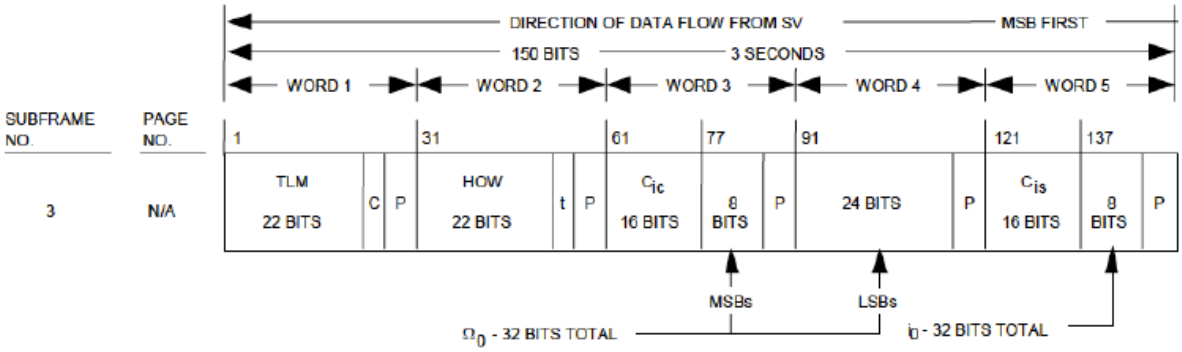
The next value that has to be computed, Toe (ephemeris reference epoch) is situated in the second subframe, between 271 and 286 bits (16 bits). The same procedure is applied, like described above.




```
% Time ref. (16 bits) subframe(2)
toe = subframe(271:286,1);
toe = flipud(toe);
t = zeros(length(toe),1)';
for i =1:length(toe)
    if toe(i)==1
        t(i) = toe(i)*2^(i-1);
    else
        t(i) = toe(i);
    end
end
toe = sum(t)*(2^4);
```

The final value of Toe is computed by multiplying the sum of all bits’ values with the scale factor described in slide number 19 (Navigation Message.pdf), 2^4.

Cic - Amplitude for cos-correction of inclination




```
% Cic (16* bits) subframe(3)
Cic = subframe(61:76,2);
Cic = flipud(Cic);
for i=1:length(Cic)
    if Cic(i) == 1
        Cic(i) = 0;
    else Cic(i) = 1;
    end
end
t = zeros(length(Cic),1)';
for i =1:length(Cic)
    if Cic(i)==1
        t(i) = Cic(i)*2^(i-1);
    else
        t(i) = Cic(i);
    end
end
Cic = (-sum(t)-1)*(2^-29);
```

Again, for calculating Cic parameter (Amplitude for cos-correction of inclination) the same algorithm (loop) is applied. But, here, because Cic parameter has 16 bits, it will be two's complement, with the sign bit + or -, occupying the MSB. Therefore, we applied this rule. The scale factor is (2^-29) (see the figure above).

21	10	12	2	10	0	0.0	-.106480438262D-03	-.272848410532D-11	.000000000000D+00
.970000000000D+02				.378750000000D+02			.479162816197D-08	-.266536534417D+01	
.168569386005D-05				.169980023056D-01			.120662152767D-04	.515361678696D+04	
.381600000000D+06				.931322574615D-07			.227667222079D+01	-.284984707832D-06	
.931992888993D+00				.128187500000D+03			-.242944885637D+01	-.787139930395D-08	
-.390016245742D-09				.100000000000D+01			.161200000000D+04	.000000000000D+00	
.100000000000D+01				.000000000000D+00			-.116415321827D-07	.970000000000D+02	
.374418000000D+06				.100000000000D+01					

# Satellite	Epoche	af0	af1	af2
	IODE	Crs	Delta n	M0
	Cuc	e	Cus	Sqrt(A)
	Toe	Cic	OMEGA0	Cis
	i0	Crc	omega	OMEGA DOT
	IDOT	Codes on L2	GPS Week	L2 P data flag
	SV Accuracy	SV health	TGD	IODC
	Transmission Time	Fit interval	-	-

Analysing the RINEX file, we can extract the values for GPS Week, Toe– ephemeris reference epoch and Cic-Amplitude for cos-correction of inclination.

Extracted values:
GPS Week: 1612
Toe: 381600 sec
Cic: $0.931323 \cdot 10^{-7}$ rad

Computed values:
GPS Week: 1612
Toe: 381600 sec
Cic: $0.931323 \cdot 10^{-7}$ rad