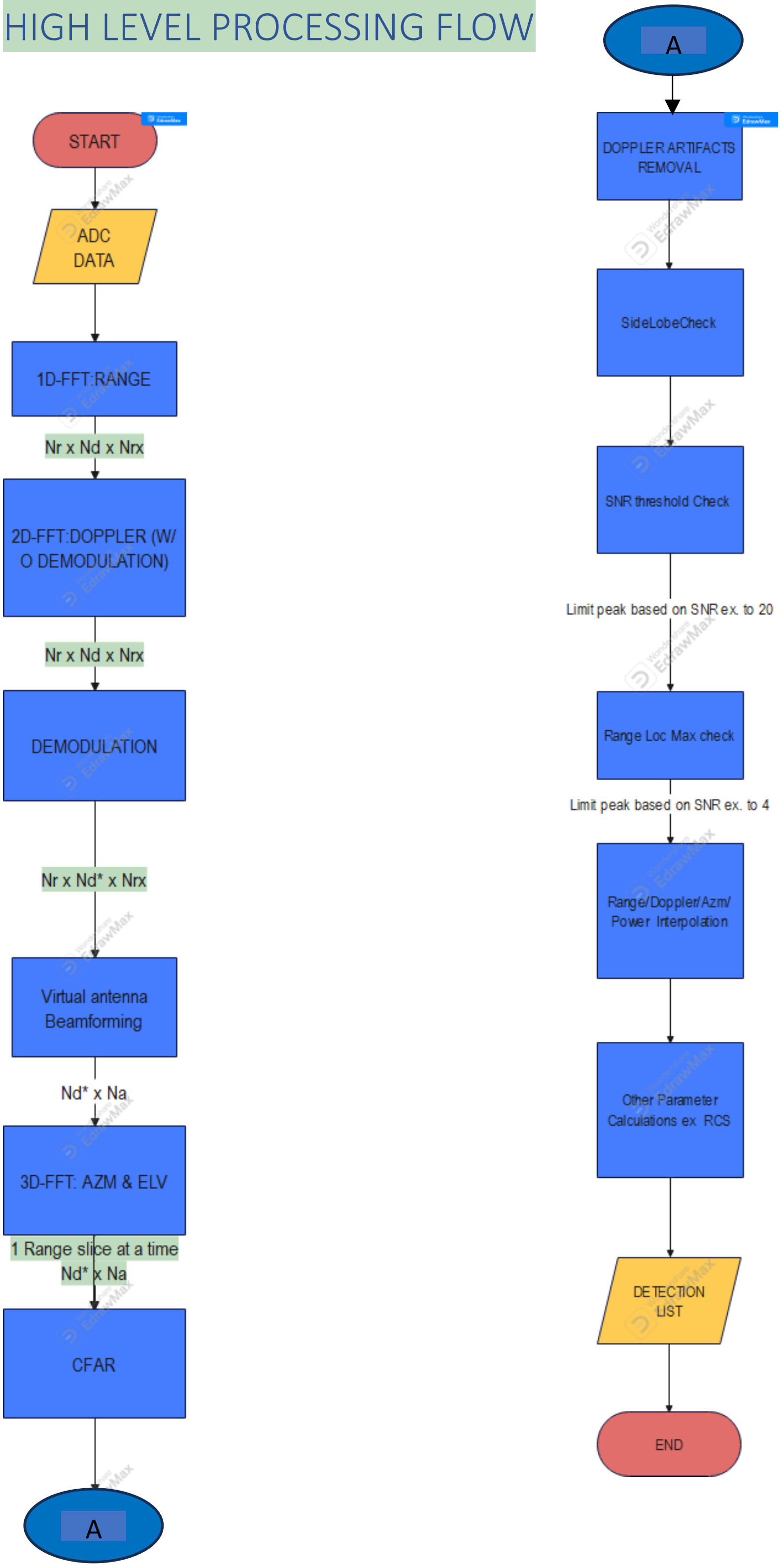
Work Profile

Upon joining the company, I took on the role of the sole developer within the team. In the initial phase, I independently constructed the entire foundational radar processing chain using Matlab. This involved various processes such as DDMA, synthetic data generation, and 1D, 2D, 3D, and 4D FFT data processing for range, Doppler, azimuth, and elevation. Additionally, I implemented interpolation and CFAR for detecting the antenna design parameters provided by TI.

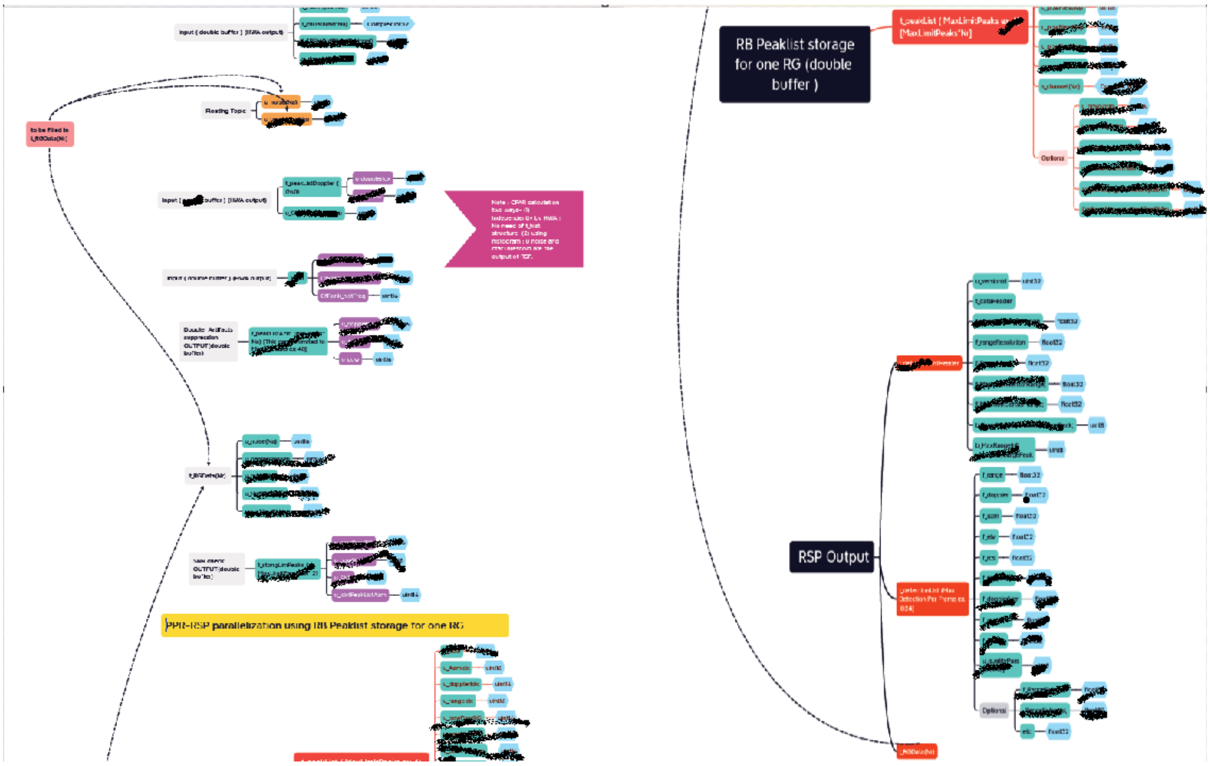
In terms of my contributions, I played a significant role in system design, actively shaping the software architecture, and participating in decision-making processes. Specifically, I contributed to the creation of comprehensive flowcharts, defined structures, and optimized system efficiency to enhance the overall performance. These are my major skills:

- **Project Management**: Contribution in system design, shaping software architecture and decision-making processes.
- **Software Architecture**: Flowchart creation, defining structures and optimizing system efficiency.
- **Algorithm Design**: Designing, implementing and testing algorithms for radar signal processing.
- **Radar Signal Processing**: Deep understanding of underlying principles of radar signal processing, HRT elevation techniques and MIMO antenna beamforming.
- **Communication and Leadership**: Effective communication, teaching experience and leading modules.

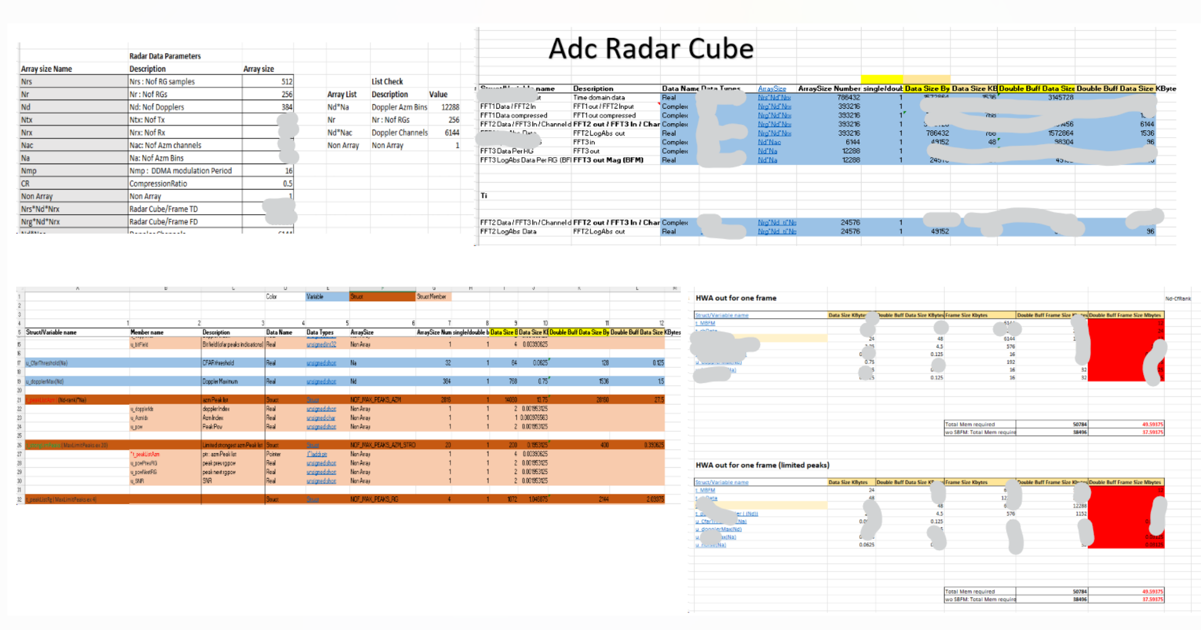
HIGH LEVEL PROCESSING FLOW



Structure definitions and data type decisions



Structure definitions and memories estimation



	Radar Data Parameters	
Array size Name	Description	Array size
Nrs	Nrs : Nof RG samples	512
Nr	Nr : Nof RGs	256
Nd	Nd: Nof Dopplers	384
Ntx	Ntx: Nof Tx	[illegible]
Nrx	Nrx: Nof Rx	[illegible]
Nac	Nac: Nof Azm channels	[illegible]
Na	Na: Nof Azm Bins	[illegible]
Nmp	Nmp : DDMA modulation Period	16
CR	CompressionRatio	0.5
Non Array	Non Array	1
Nrs*Nd*Nrx	Radar Cube/Frame TD	[illegible]
Nrg*Nd*Nrx	Radar Cube/Frame FD	[illegible]

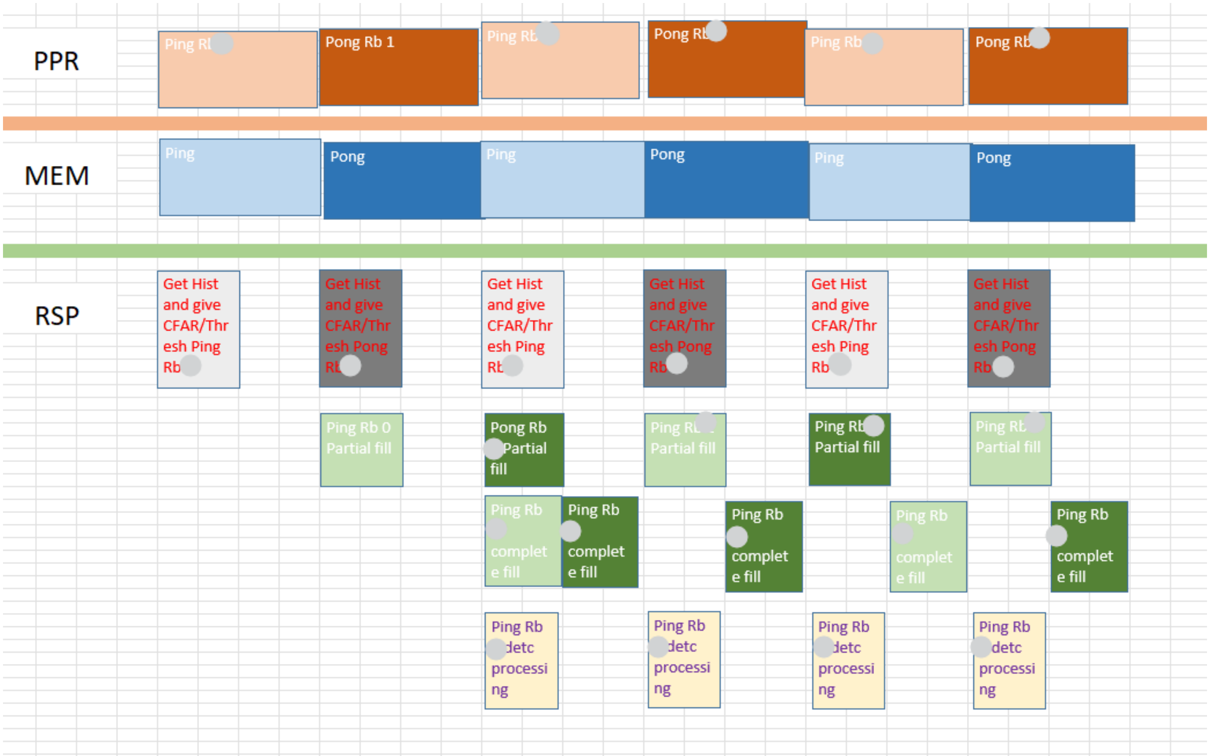
	List Check	
Array List	Description	Value
Nd*Na	Doppler Azm Bins	12288
Nr	Nr : Nof RGs	256
Nd*Nac	Doppler Channels	6144
Non Array	Non Array	1

Data flow and scaling estimation to overflow/underflow protection

			max adc out
adc vtg	[illegible]		
adc bits	[illegible]		
adc res	7.62939E-05		
adc out min	0	0	Volt
adc out max	[illegible]	4.999923706	Volt
HWA i/o bits Max		23	
HWA window fixed down scale bits	[illegible]		

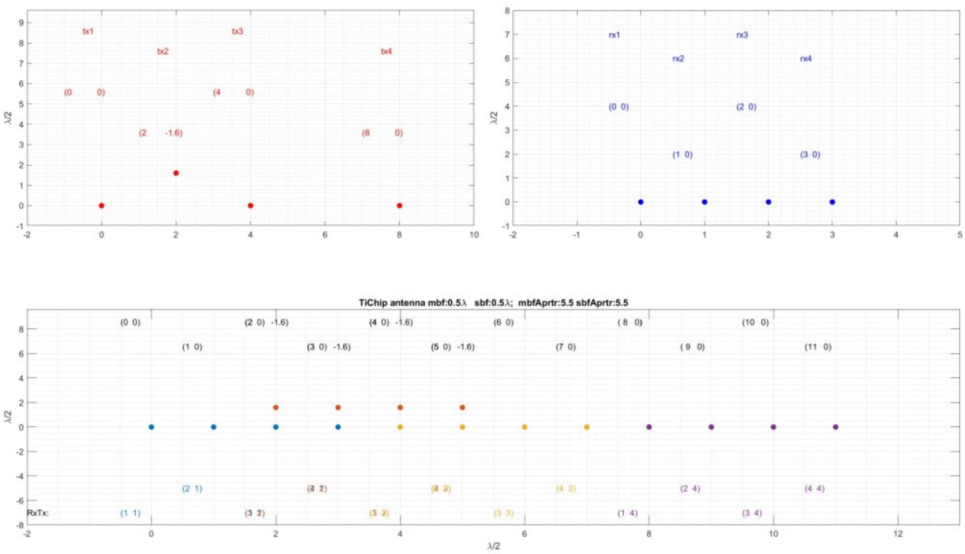
	ADC		HWA DC removal & interferance Mitigation		Windowing		R FFT (non complex input)		Windowing		D FFT		Demodulation/DBF preparation		Windowi
					Selected window WIN_HANN	Gain Bits 16			Selected window WIN_HANN	Gain Bits 16					Selected window
	Bits/data type	Max values	Bits/data type	Max values	Bits/data type	Max values	Bits/data type	Max values	Bits/data type	Max values	Bits/data type	Max values	Bits/data type	Max values	Bits/data type
Input type	[illegible]		[illegible]		[illegible]		[illegible]		[illegible]		Complex		Complex		Real
Input data type	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	int32
Input bits	16	32767	16	32767	16	32767	[illegible]	[illegible]	16	32767	16	32767	[illegible]	[illegible]	24
Input Scale	0		0		0		0		0		0		0		0
process scale	0		0		[illegible]		6.99859043		[illegible]		7.58355293		[illegible]		17
Output Scale	0		0		-17		[illegible]		-17		[illegible]		0		[illegible]
output bits	16	32767	16	32767	16	32767	16	32767	16	32767	[illegible]	[illegible]	[illegible]	[illegible]	24
Out type	Real		Real		Real		Complex		[illegible]		Complex		Complex		Real
Out data type	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	int32
Out fits in type	0	yes	0	yes	0	yes	0	yes	0	yes	[illegible]	yes	[illegible]	yes	[illegible]

Tasks parallelization using planning tasks on multiple cores



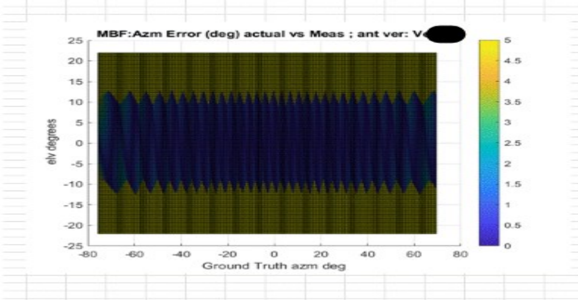
Ti Antenna Layout (Available on "TI" Website)

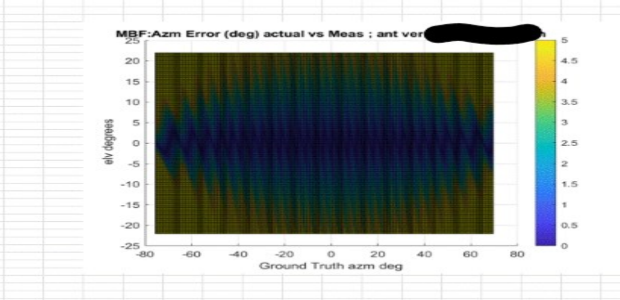
- Matlab script for the :
  - virtual layout and other parameters generation from the given Tx/Rx positions

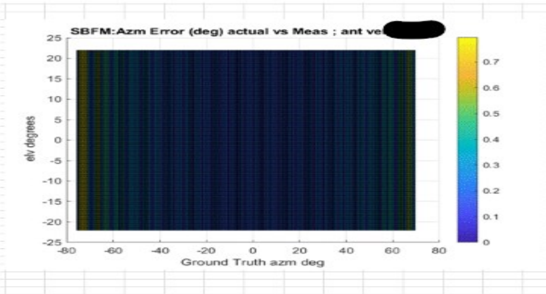


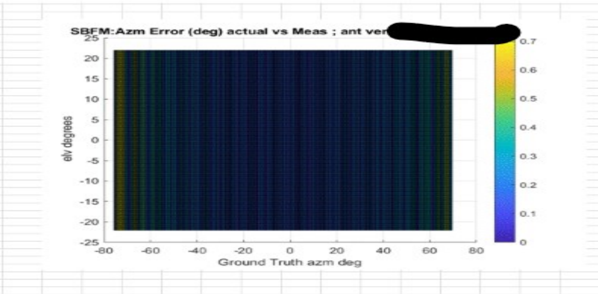
tx1
tx2
tx3
tx4
(0 0)
(4 0)
(2 -1.6)
(8 0)
λ/2
rx1
rx2
rx3
rx4
(0 0)
(2 0)
(1 0)
(3 0)
λ/2
TiChip antenna mbf:0.5λ sbf:0.5λ; mbfAprtr:5.5 sbfAprtr:5.5
(0 0)
(2 0) -1.6)
(4 0) -1.6)
(6 0)
(8 0)
(10 0)
(1 0)
(3 0) -1.6)
(5 0) -1.6)
(7 0)
(9 0)
(11 0)
(2 1)
(4 3)
(2 4)
(4 4)
RxTx:
(1 1)
(3 3)
(1 4)
(3 4)
λ/2
λ/2

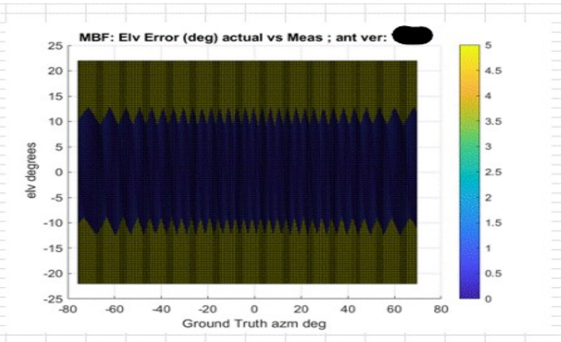
- Generation of the report for the basic antenna angle performance assessment

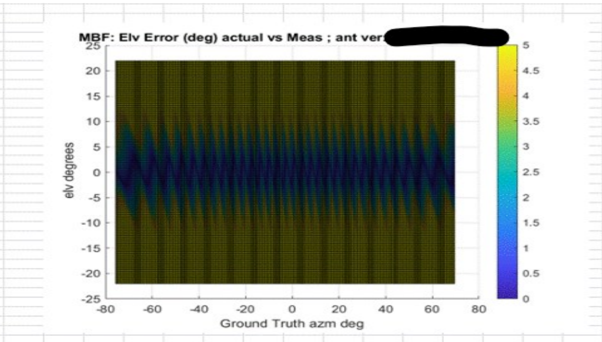


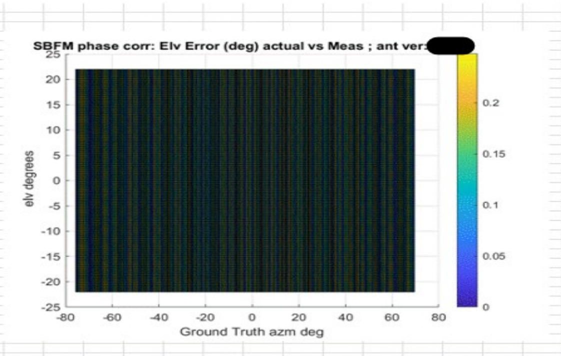


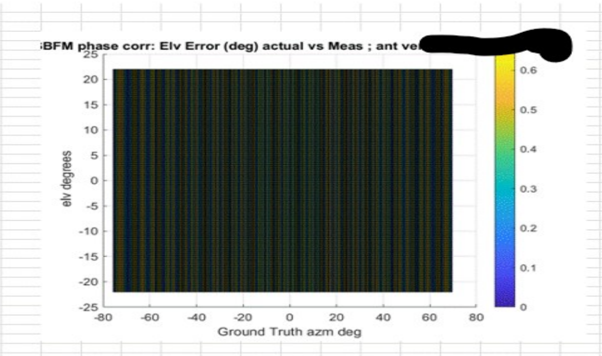












Range and doppler processing

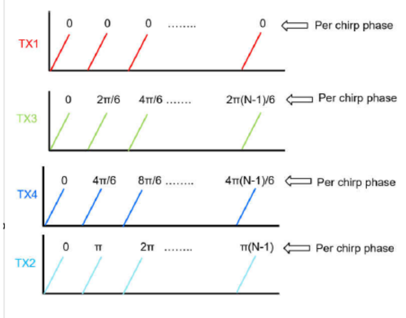


Figure : DDMA Modulation

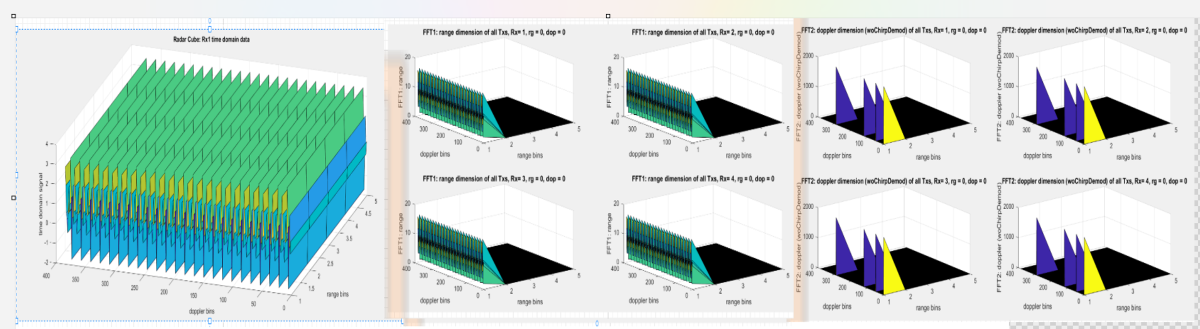


Figure : ADC, range and doppler processing

Doppler Dimension has Tx modulated information.

(1) Range :-

$$f_b = f_r = \left(\frac{f_{s\_r}}{NFFT\_r}\right) * (f_{r\_bin}) \quad \text{--- (1)}$$

sampling freq.

f_r: freq at bin i.e ~~bit freq~~

f_{r_bin}: bin no (r) (Note ⇒ This is not a freq.)

due to linear freq. modul^n

we know that ⇒ $f_r = 2 * \text{slope} * \text{travel time of sig to/from target}$

$$f_b = f_r = \left(\frac{B}{T_c}\right)\left(\frac{2R}{c}\right) \quad \text{--- (2)}$$

T_c: entire chirp time

$$f_{r\text{-}resol^n} = \frac{f_{s\text{-}r}}{NFFT_r}$$

freq. resol^n in terms of range ⇒

$$\boxed{\therefore R_{resol^n}\ (\Delta R) = \left(\frac{f_{s\_r}}{NFFT_r}\right)\left(\frac{c}{2B}\right) T_c} \quad \text{--- (3)}$$

or Range gate resol^n

∴ ~~But is const~~ ADC sampling time follows this equality ⇒ $\boxed{f_{s\_r} = NFFT/T_c}$ ... (4)

∴ for eq^n $\boxed{R_{resol^n} = R_g = \frac{c}{2B}}$

$$\boxed{\therefore \text{Range } (R) = f_{r\text{-}bin} * R_g}$$

Continental

R_g ⇒ Range Gate Length

(2) Velocity :-

f_v or f_d → freq for doppler or velocity sig.

(1) $f_d = \left(\frac{f_{s\text{-}d}}{NFFT\_d}\right)(f_{d\_bin})$ ---- (1)

=

(2) path_diff = ~~2A~~ $2 * (v\, T_c)$

v: relative velocity betn ego vehicle & target

T_c: chirp time

∴ phase diff induced =

$$P_{diff} = \left(\frac{2\pi}{\lambda}\right) * \text{path\_diff}$$

$$= \frac{2\pi}{\lambda} * 2 v T_c * n$$

$$= 2\pi\left(\frac{2v}{\lambda}\right) T_c\, n \quad \cdots (2)$$

n: samples

compare to $2\pi f t$

(*) ∴ $f_d = \frac{2v}{\lambda}$ ~~...~~ inter chirp period (3)

(*) freq. resol^n

$$f_{d\text{-}resol^n}\ (\Delta V) = \frac{f_{s\text{-}d}}{NFFT\_d} = \frac{1/T_c}{NFFT_d} \quad \text{- (4)}$$

Continental

From eq^n (3) & (4)

$$\boxed{V_{resol^n} = \frac{\lambda}{2}\cdot\frac{1}{T_c\, NFFT\_d}} \quad \cdots (5)$$

$$\boxed{\therefore \text{velocity} = V = \left(\frac{\lambda}{2\, T_c\, NFFT_d}\right) * f_{d\_bin}}$$

f_{d_bin}: freq bin no.

It is clear from above eq^n. Max^m velocity unambigously captured is obtained by substituting max ~~fd bin~~ $\boxed{f_{d\text{-}bin\text{-}max} = \pm \frac{NFFT_d}{2}}$ (freq ⇒ since freq. resol^n $\pm(\frac{f_s}{2})/NFFT$ ~~3...~~) ∴ max freq can be $\pm f_s/2$

$$\boxed{\therefore V_{max}\ (\text{unambigious}) = \pm \frac{\lambda}{4 T_c}} \quad \cdots (6)$$

or simply put $\pm f_s/2 = \pm \frac{1}{T}$ ~~...~~ in eq^n (3) you get

max^m unambigous velocity can measure

Encoding (DDMA: Doppler division multiple access)

- Doppler division multiple access (DDMA) is a MIMO scheme that allows simultaneous transmission of all TX channels
- Rotation speed scheme should be in a such way that doppler artifacts power should be as minimum as possible:
- Ex. (for 4x4 Txs) Period : 16, code : 0, 3, 10 , 14, Txs contribution to target and doppler artifacts peaks and empty band : **4**,0,1,1,1,1,1,1,0,1,1,1,1,1,1,0

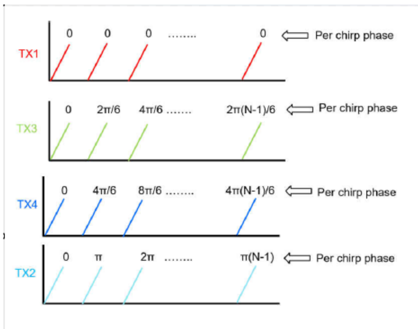


Figure : DDMA Modulation

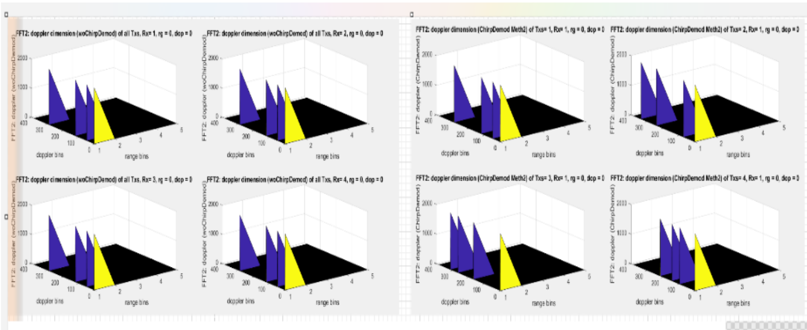
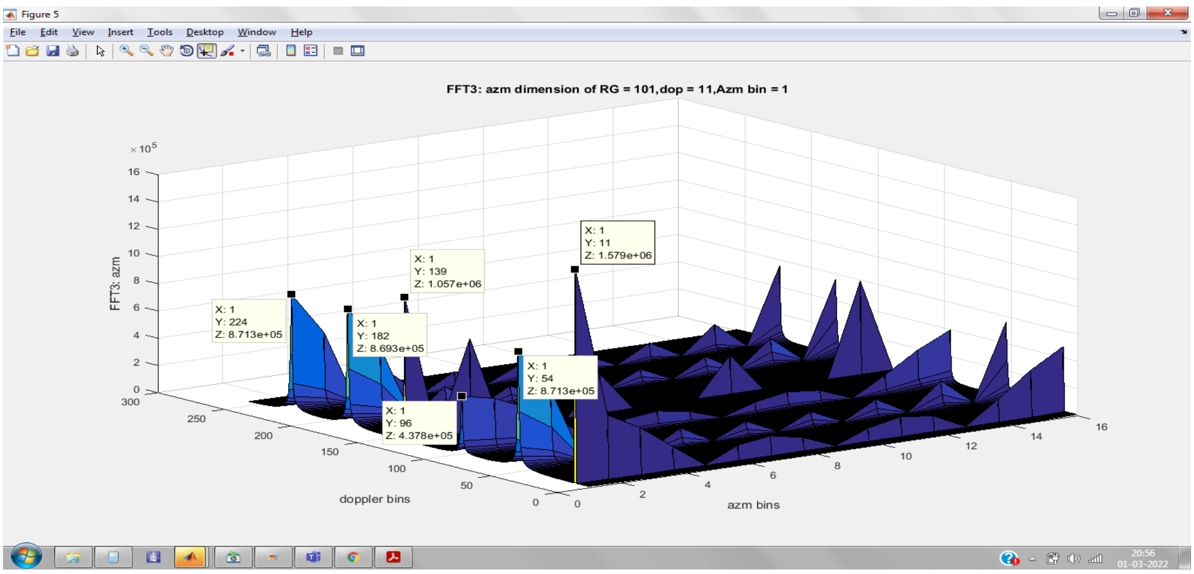
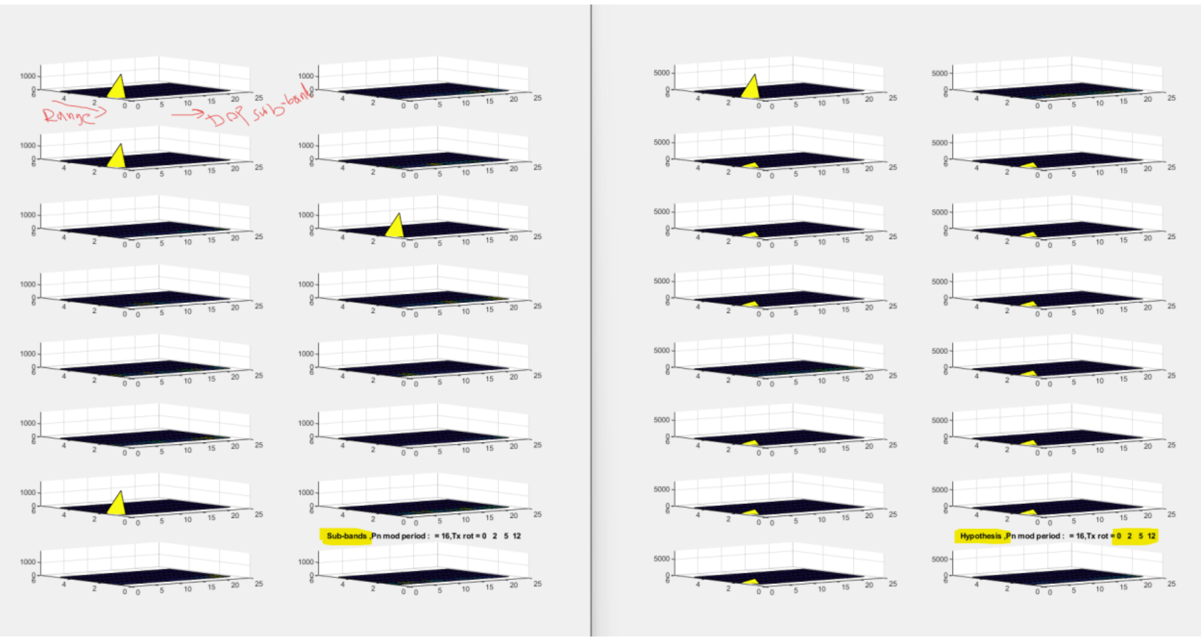


Figure : Before and after demodulation

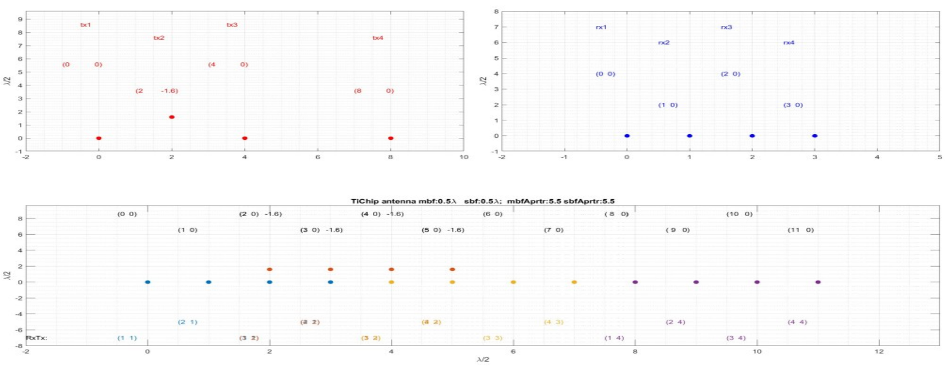
- **Traditional doppler demodulation methods:**
  - Way 1: After fft1 demodulate the phases using opposite phase of modulations, then fft2 will be applied.
  - Way 2: Do fft1 and fft2 without demodulation, and rearrange the doppler dimension data according to our DDMA codes (phase (in terms of the dopplers)
  - Both methods produce same results. It has the “doppler artifacts peaks” which need to be removed in the further processing.

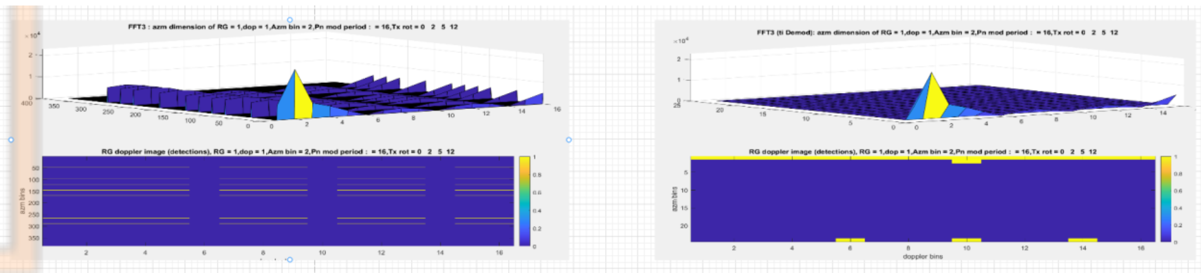


- **Ti doppler demodulation methods:**
  - In this method , no doppler artifacts will be generated but it will have only Nd/DDMA periods dopplers (so artifacts positions will be lost already)
  - According to DDMA rotation speed in terms of doppler positions, we create doppler sub-bands and their hypothesis. Hypotheses are forming using the subband addition according to DDMA codes ex. Here 0,2,5,12. Based on hypothesis we select the corresponding demodulated signals corresponding to Txs.



Azm channels sequencing according to Virtual antenna layout and azm calculations





Date: __/__/__

(13) Azimuth Angle

normalized freq means normalization Fs to 1 $F_{sa} = 1$

corresponding to angle bin $\rightarrow$ $$f_a = \left(\frac{f_{s\text{-}a}}{NFFT_{\text{-}a}}\right) \cdot f_{a\text{-}bin}$$

(Fs spatially)

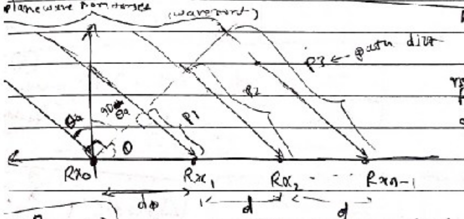
* sig obtained in angle dimension is a spacially sampled sig unlike time sampled sig in range & doppler dimension. We don't have the value of $f_{s\text{-}a}$. Thus we have to compute angle value from normalised freq. (f_{ang} given by $\frac{1}{NFFT}$)

$$\boxed{f_{ang} = \frac{f_a}{f_{s\text{-}a}} = \frac{f_{a\text{-}bin}}{NFFT_{\text{-}a}}} \quad \ldots (0)$$

this will go from $-\frac{1}{2}$ to $\frac{1}{2}$ as NFFT = 16 ...

normalizing Fs = 1. so FFT resoln is in bins instead of Hz. but mapping SI unaltered!

planewave from target (wavefront)



$a = \theta$
$= 90 - (90 - \theta_a)$

Fig: Encoding of Angle

* path diff. $P_i = d \sin(\theta)\, n$ $\ldots (1)$

* The induced phase diffn

$$P_{diff}(n) = \left(\frac{2\pi}{\lambda}\right) d \sin(\theta)\, n \quad \ldots (2)$$

Continental

Date: __/__/__

It is spatially sampled sig of following form

phase $\omega = 2\pi \left(\frac{d \sin(\theta)}{\lambda}\right) n$

$\omega = 2\pi f$ so f

f normalised $= \frac{1}{NFFT}$

$$\boxed{\therefore f_{ang} = \frac{d \sin\theta}{\lambda}} \quad \ldots (3)$$

$$\frac{1}{NFFT} = \frac{d \sin\theta}{\lambda}$$

now put normalize Fs = 1 (so instead ~~of freq to~~ we will get bins to θ relation)

$\therefore$ angle of target

$$\theta = \sin^{-1}\left(f_{ang}\, \lambda/d\right)$$

from eqn (0)

$$\boxed{\theta = \sin^{-1}\left[\left(\frac{\lambda}{d \cdot NFFT_{\text{-}a}}\right) * f_{a\text{-}bin}\right]} \quad \ldots (4)$$

distance betn Rx antennas

Resolns $\boxed{\theta_{max\,(unambiguous)} = \sin^{-1}\left(\frac{\lambda}{d \cdot 2}\right)}$ since ($f_{a\text{-}bin\,max} = \frac{NFFT}{2}$)

R,D,A resolutions

distance betn Rx antennas

Resolns $\boxed{\theta_{max\,(unambiguous)} = \sin^{-1}\left(\frac{\lambda}{d \cdot 2}\right)}$ since ($f_{a\text{-}bin\,max} = \frac{NFFT}{2}$)

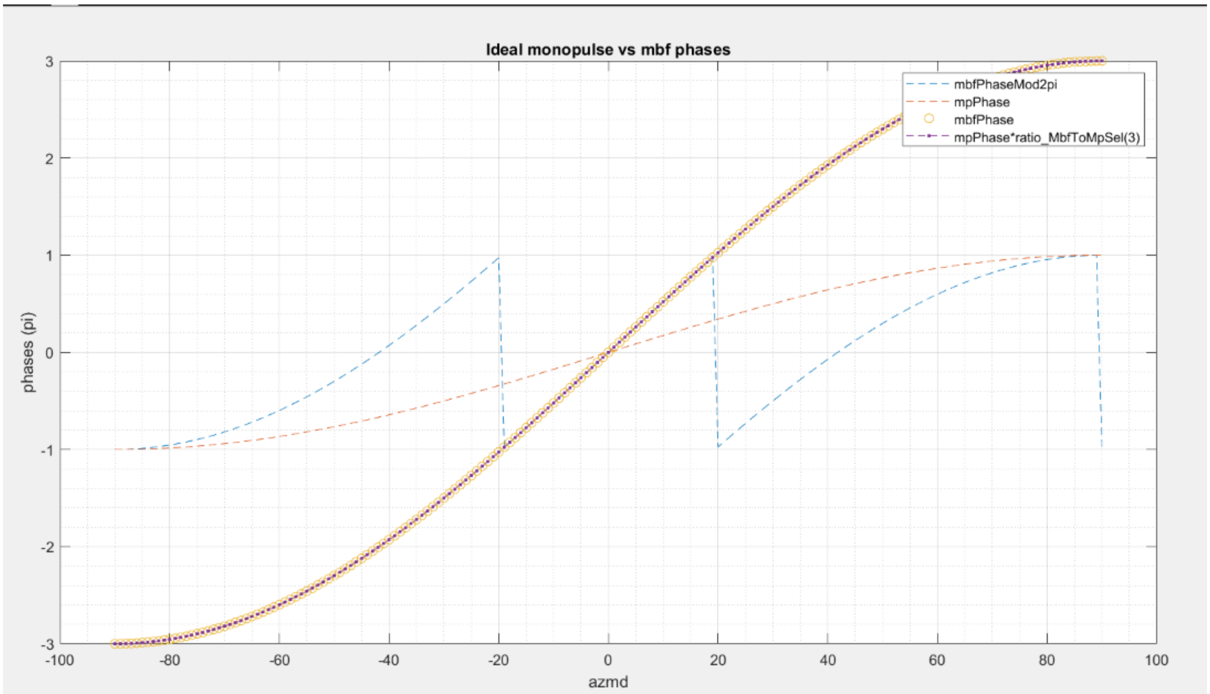
$$\boxed{\Delta_R = \frac{c}{2 B_c}} \quad \boxed{\Delta_D = \frac{\lambda}{2 T_c\, NFFT_D}} \quad \boxed{\Delta_\theta = \sin^{-1}\left(\frac{\lambda}{d\, NFFT_\theta} \cdot 2\right)}$$

T_c: Inter chirp periode (sample periode)

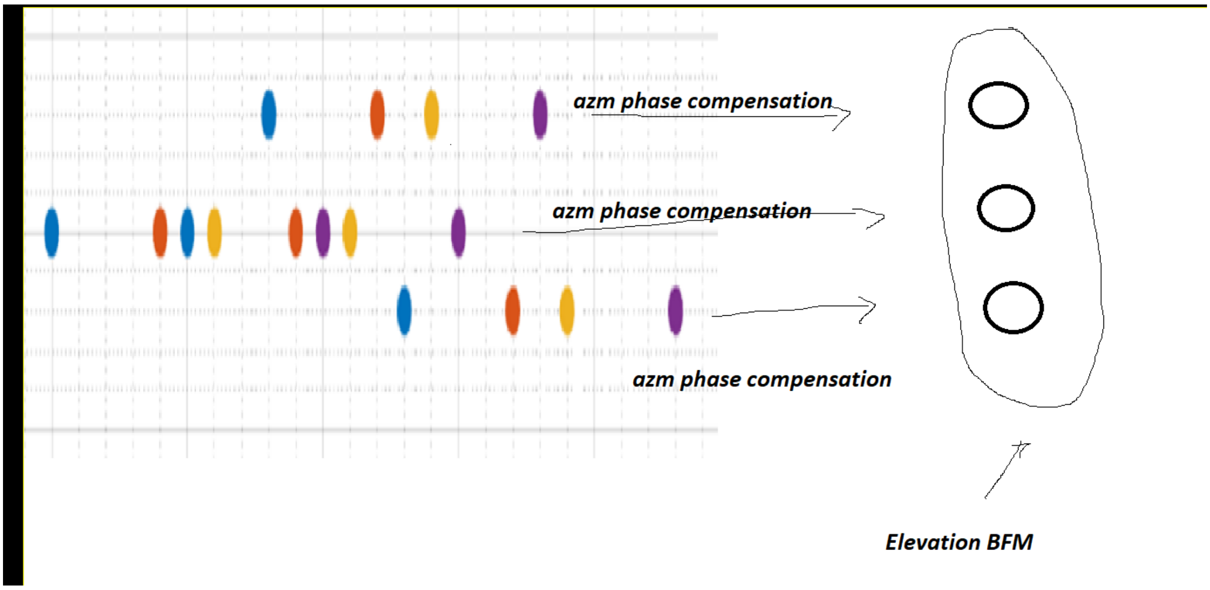
d: sample periode (spatially)

Monopulse (when antenna spacing > 0.5 lambda)

Ambiguity solved using the monopulse pair channels i.e. channels with the 0.5 lambda distance.



Elevation

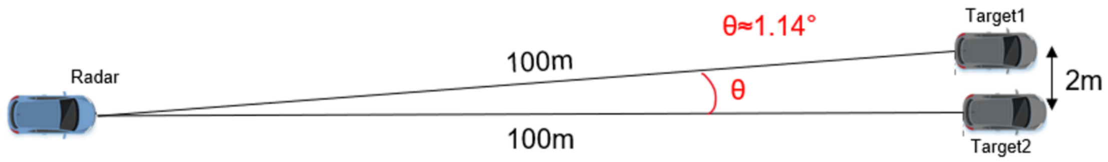


Super resolution techniques (SRA) for angles

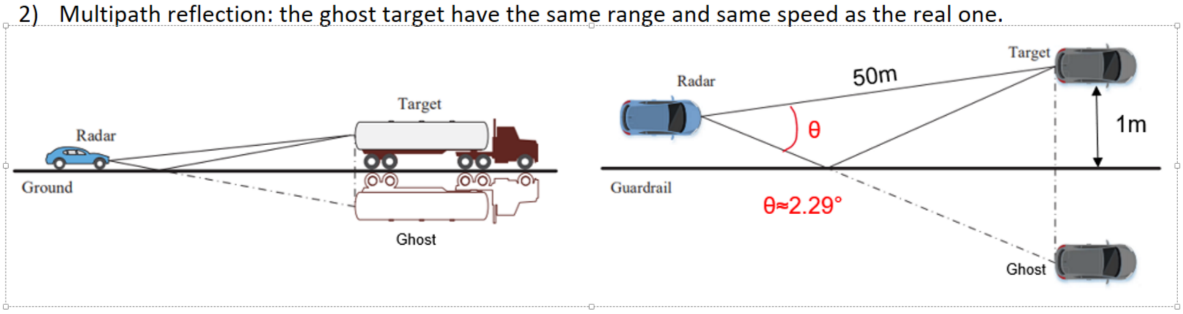
$$\theta_{res} = \frac{\lambda}{Nd\cos\theta}$$

However, the resolution is not enough in some tricky cases:

1) Two targets that are close to each other have the same range and same speed (or stationary).



2) Multipath reflection: the ghost target have the same range and same speed as the real one.



We build an objective function with target angles as parameters, and does optimization w.r.t. parameters i.e. angles of targets.

* signal models *optimization functions * GLRT (generalized likelihood ratio test)

*time consuming process *still not worked for elevation *alpha filtering in tracker

Quality parameters of the detections

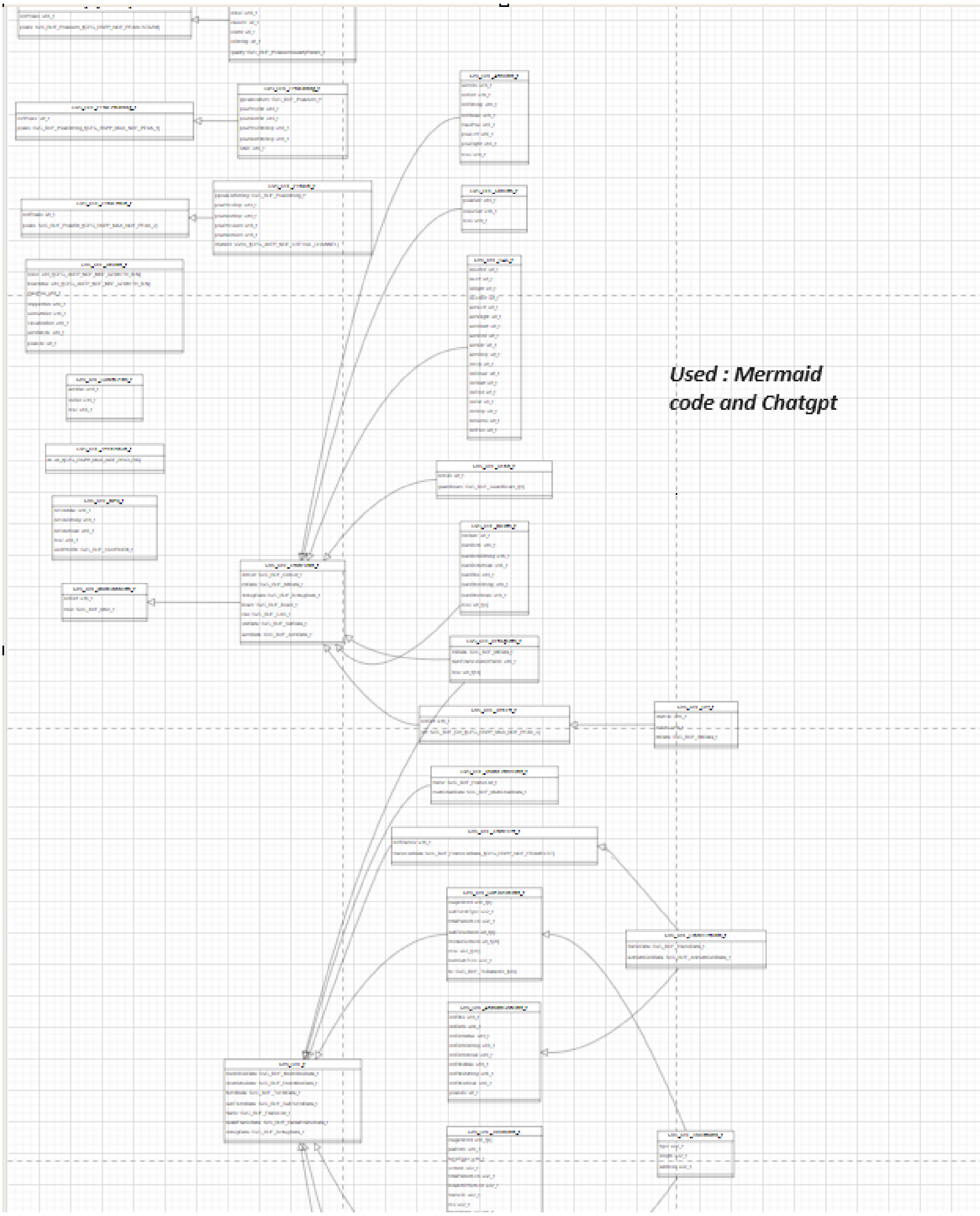
- Monopulse
- Multi target
- Nacom/near range ghosts
- Artifacts

Challenges

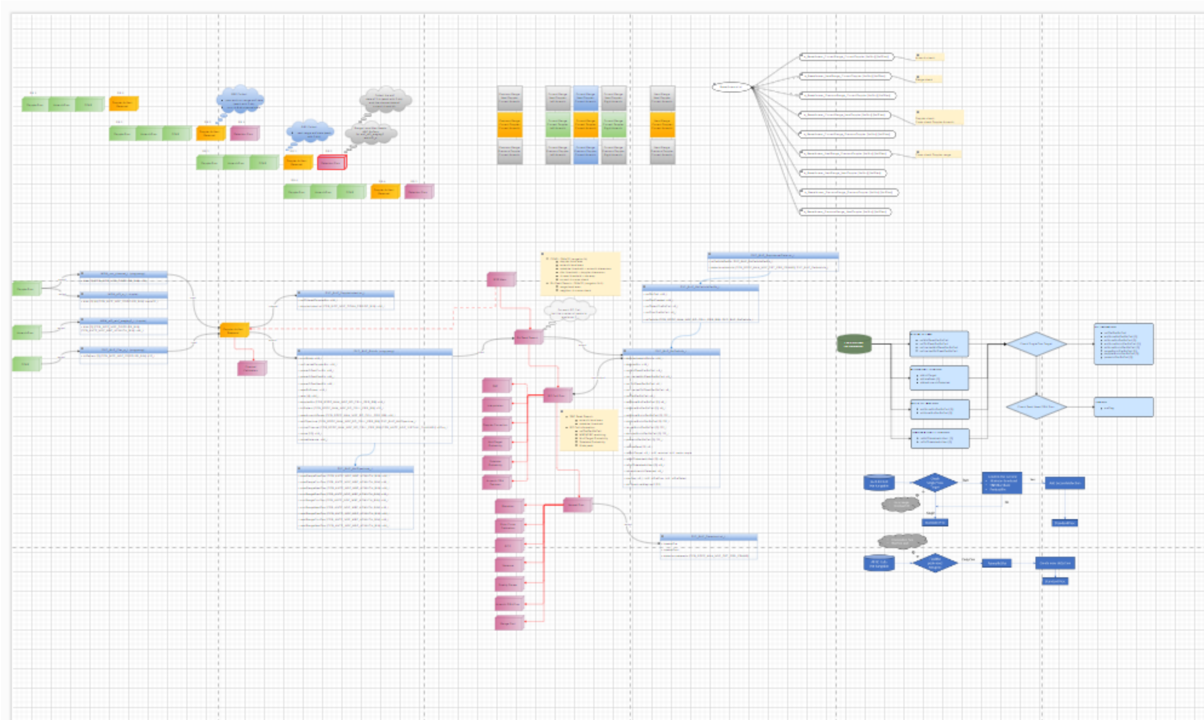
- Near range noise : alpha filter

- Near range filter / NACOM
- Near range detection : change threshold if drastic change in the threshold in near range (use accumulated noise)
- Elevation : use filtering
- Fake moving detections : tag as low power detections based on some thresholds

Architecture restructuring



Used : Mermaid code and Chatgpt



Field Application Engineer (FAE) – Calterah Semiconductor, Munich

In my role as a Field Application Engineer at Calterah Semiconductor, I contributed across customer support, technical debugging, solution development, and product promotion. I worked closely with automotive customers integrating Calterah's CMOS millimeter-wave radar SoCs, including the Alps Std, Alps Pro, and Alps Mini.

Key Contributions:
• Supported customers in radar sensor bring-up, debugging software and hardware issues, and aligning system-level configurations.
• Prepared detailed analysis for issues related to radar signal chain, calibration, communication interfaces (Ethernet, SPI, CAN), boot-up sequences, and SDK-level integrations.
• Conducted product demonstrations, technical workshops, and training sessions for customer engineering teams to help them understand radar SoC features, SDK usage, and performance tuning.
• Collaborated with internal R&D teams by reporting customer findings, validating fixes, and testing new SDK releases.
• Assisted in marketing and exhibition activities, including representing the company at Electronica 2024 to introduce Calterah radar products to global visitors.
• Provided system-level design suggestions to customers regarding antenna layout, MIMO configurations, thermal considerations, and integration constraints.
• Ensured smooth communication between customers and internal design teams, improving feedback loops and project timelines.

This role strengthened my customer-facing skills, deepened my understanding of CMOS radar SoC ecosystems, and improved my ability to bridge engineering, product, and customer workflows.