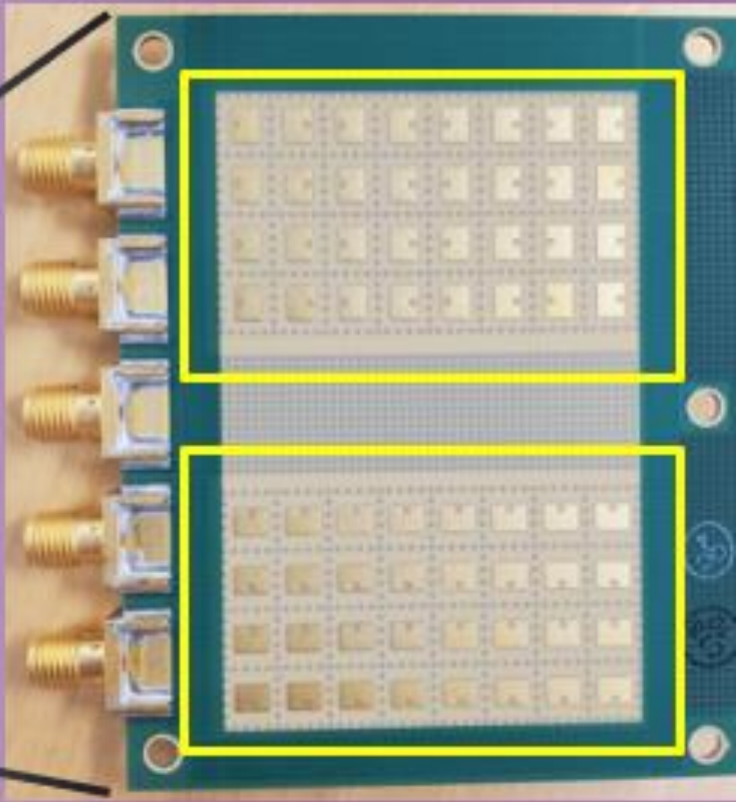
UC San Diego
JACOBS SCHOOL OF ENGINEERING
Electrical and Computer Engineering

WCSNG
Wireless Communications Sensing and Networking



mMobile: A Compact and Real-Time Millimeter-wave Experiment Framework with True Mobility Capabilities


Ish Jain


Raghav Subbaraman


Dinesh Bharadia

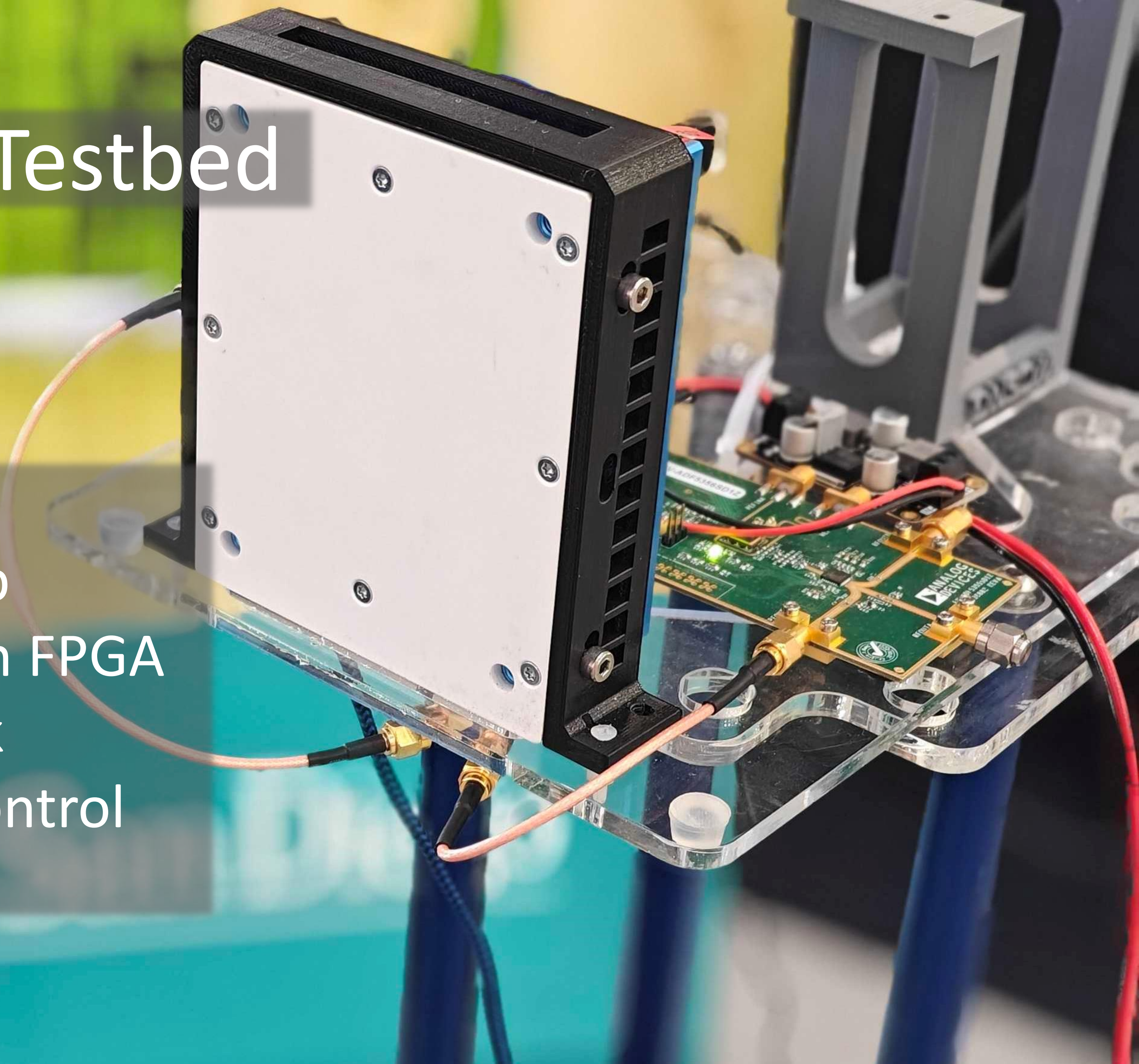
Testbeds – Predominantly 60 GHz

2014	2016	2017	2019	2020
OpenMili 60 GHz	MIRA 24 GHz	X60 60 GHz	Millimetera 60 GHz	Mm-Flex 60 GHz

60 GHz mmWave	28 GHz mmWave
Limited adoption	Deployments Ramping up
IEEE 802.11ad	3GPP 5G-NR
High attenuation (O2 absorption)	Better propagation

Need testbeds at 28 GHz to keep up with current trends

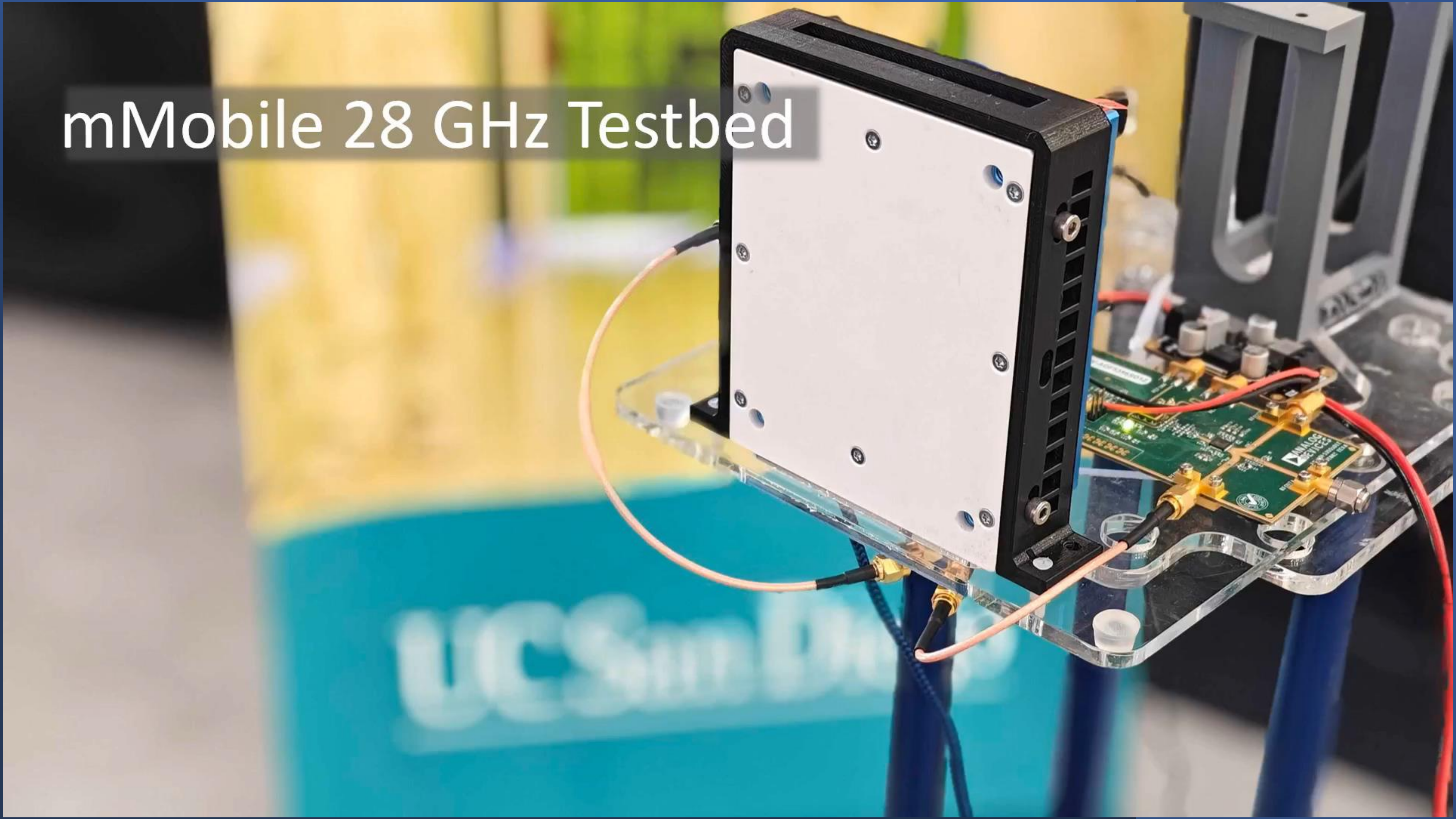
mMobile 28 GHz Testbed



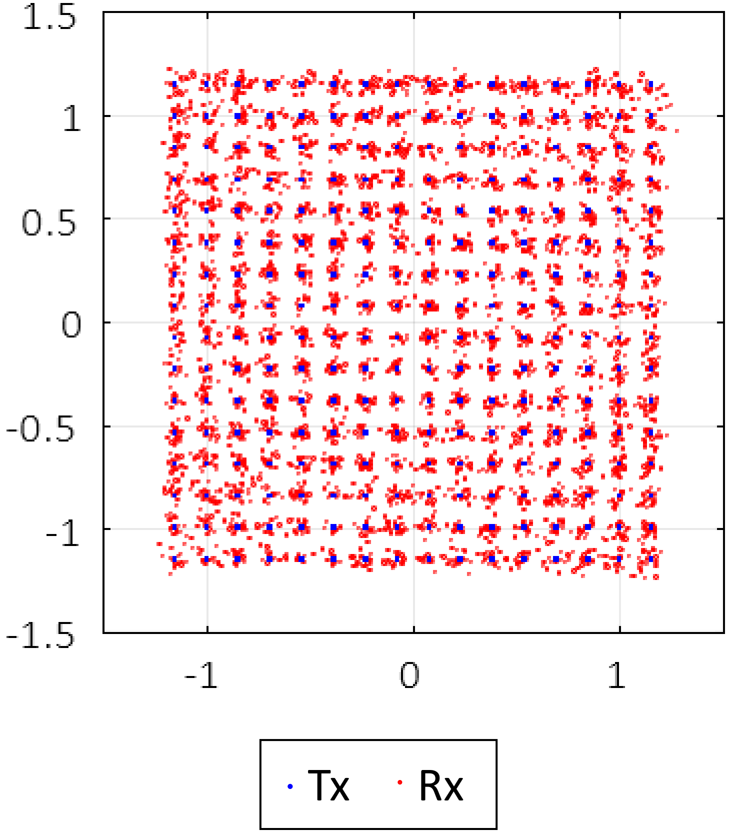
Features

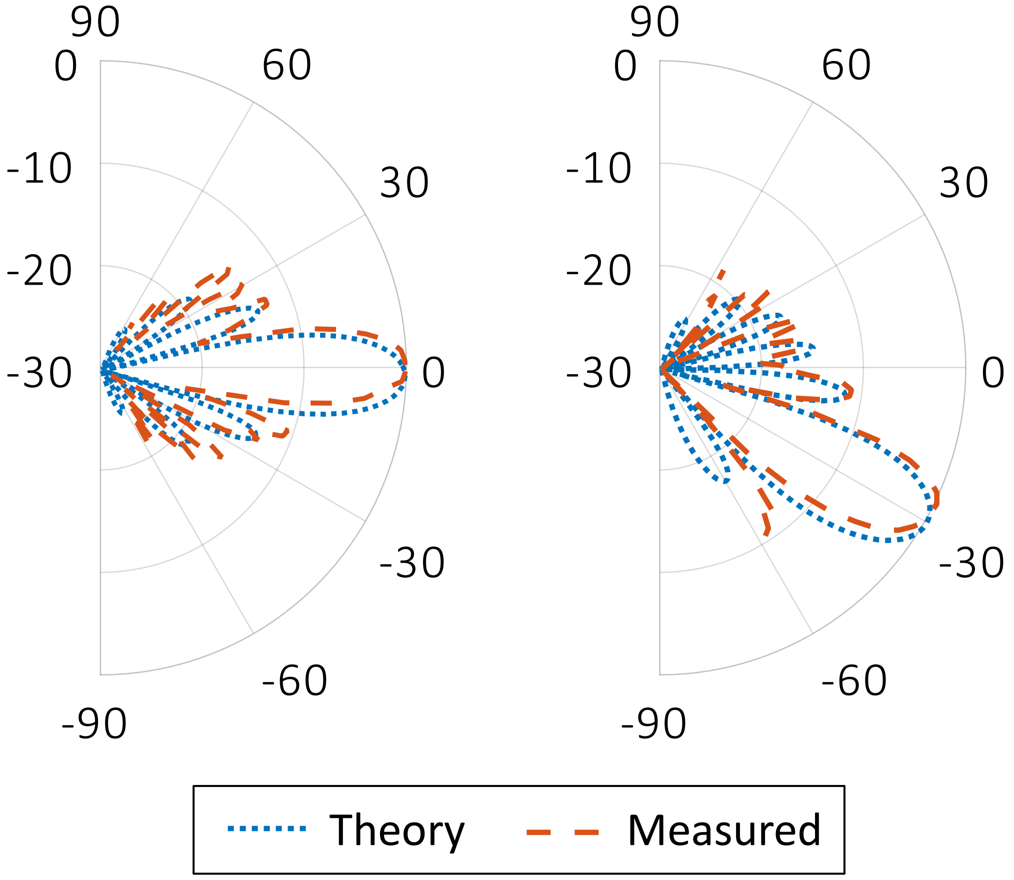
- Compact hand-held setup
- Fast scan: $2\mu s$/beam with FPGA
- Large 1024 size codebook
- Precise phase and gain control
- 5G NR ready

mMobile 28 GHz Testbed



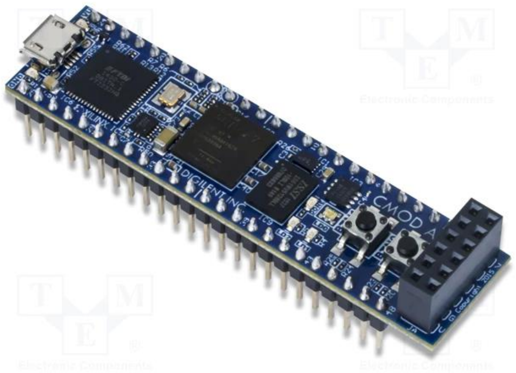
256 QAM modulation and perfect beam patterns

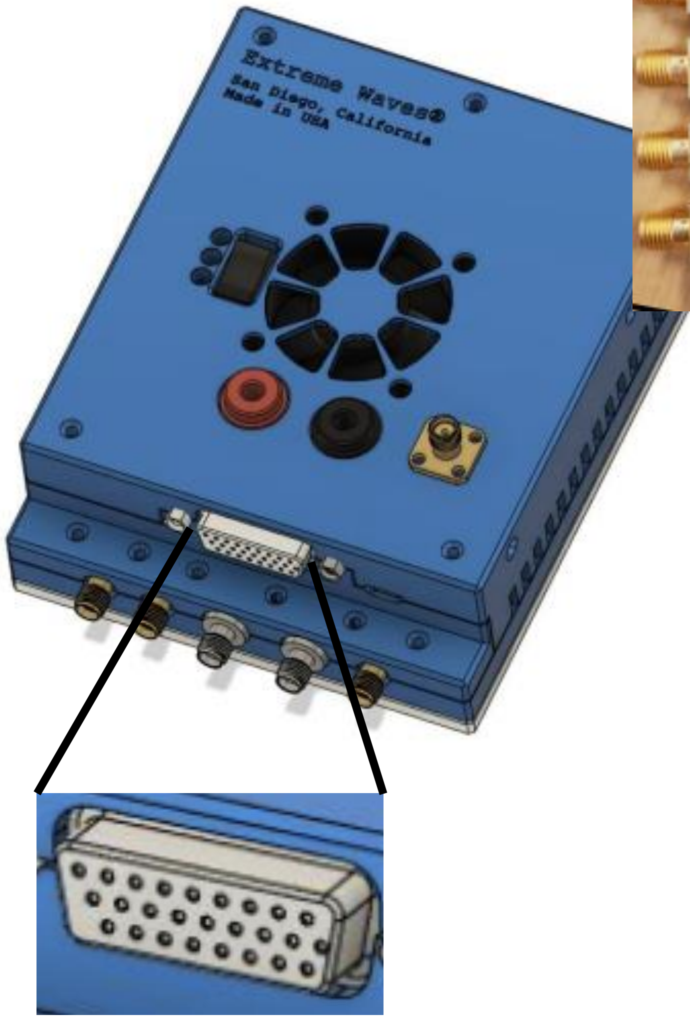


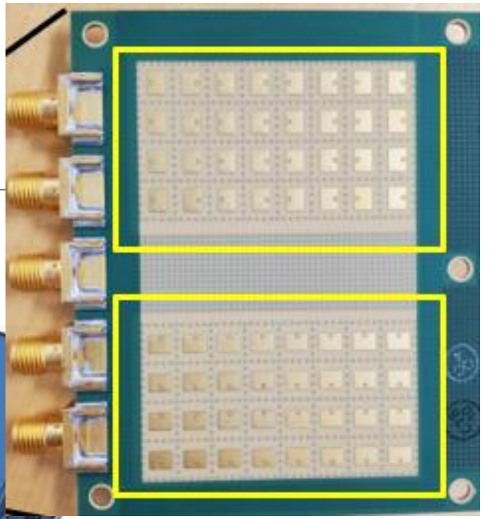


Fast Programming phased array

- Goal – Control the beam pattern of the phased array.
  - How? – Registers in the phased array contain the phase and gain values corresponding to a beam pattern.
- Objective – Program the registers with the required phase and gain values.
  - How? – Registers can be programmed using microcontrollers through **SPI** interface.





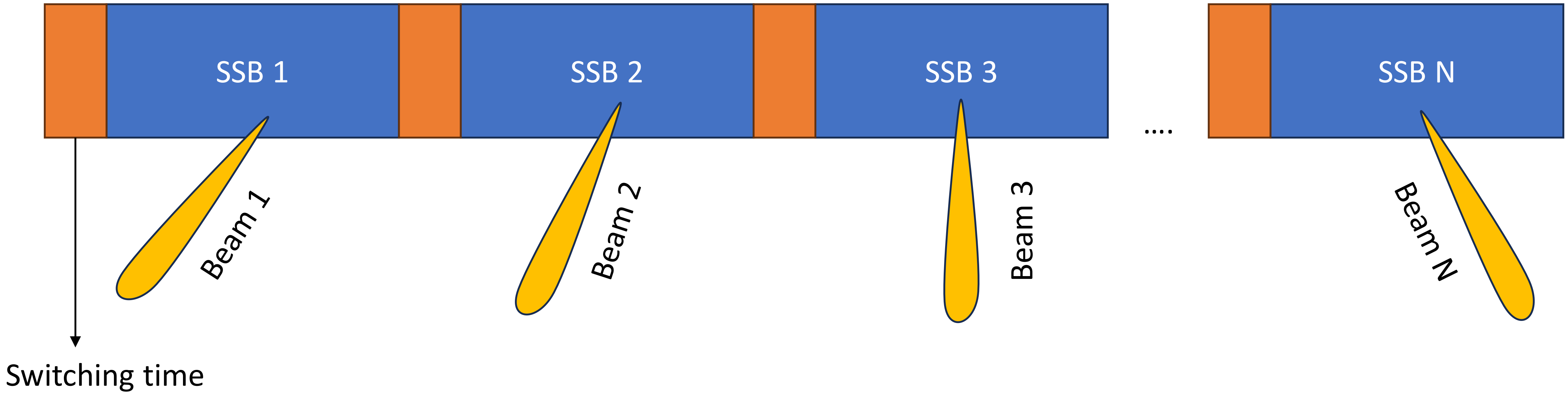


Front end

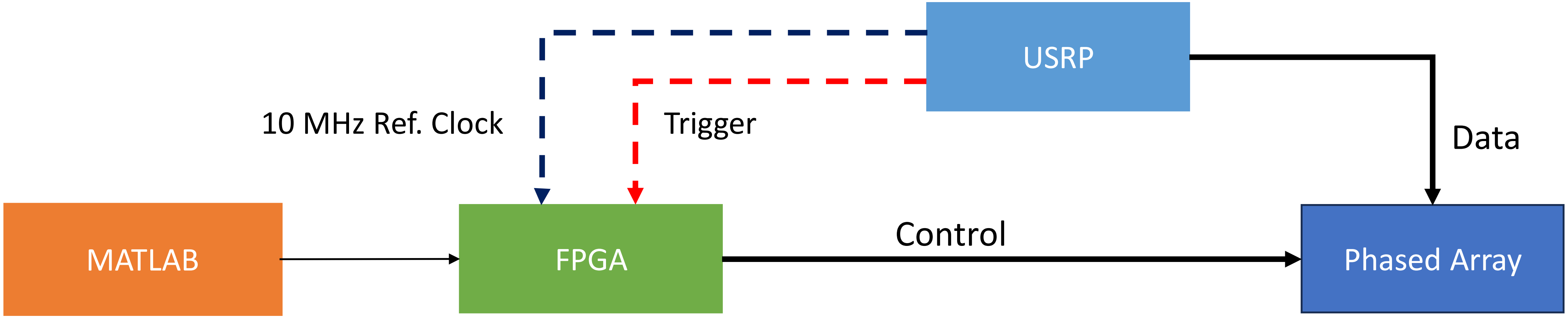
Support for 5G NR waveforms

5G NR waveform support:

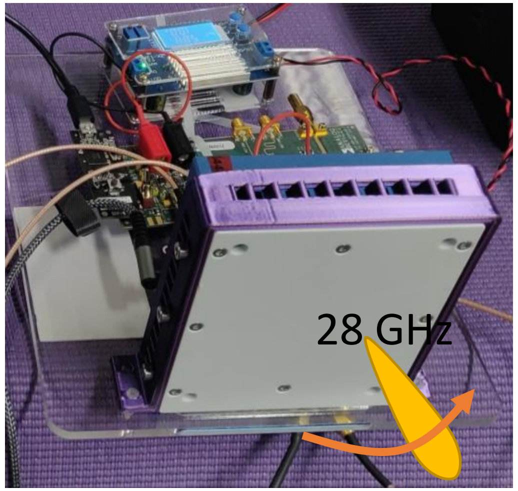
- 5G NR waveform with SSB length of 12.5us is taken.
- Next beam is switched within 3us.



USRP – FPGA interface for synchronized beam scan



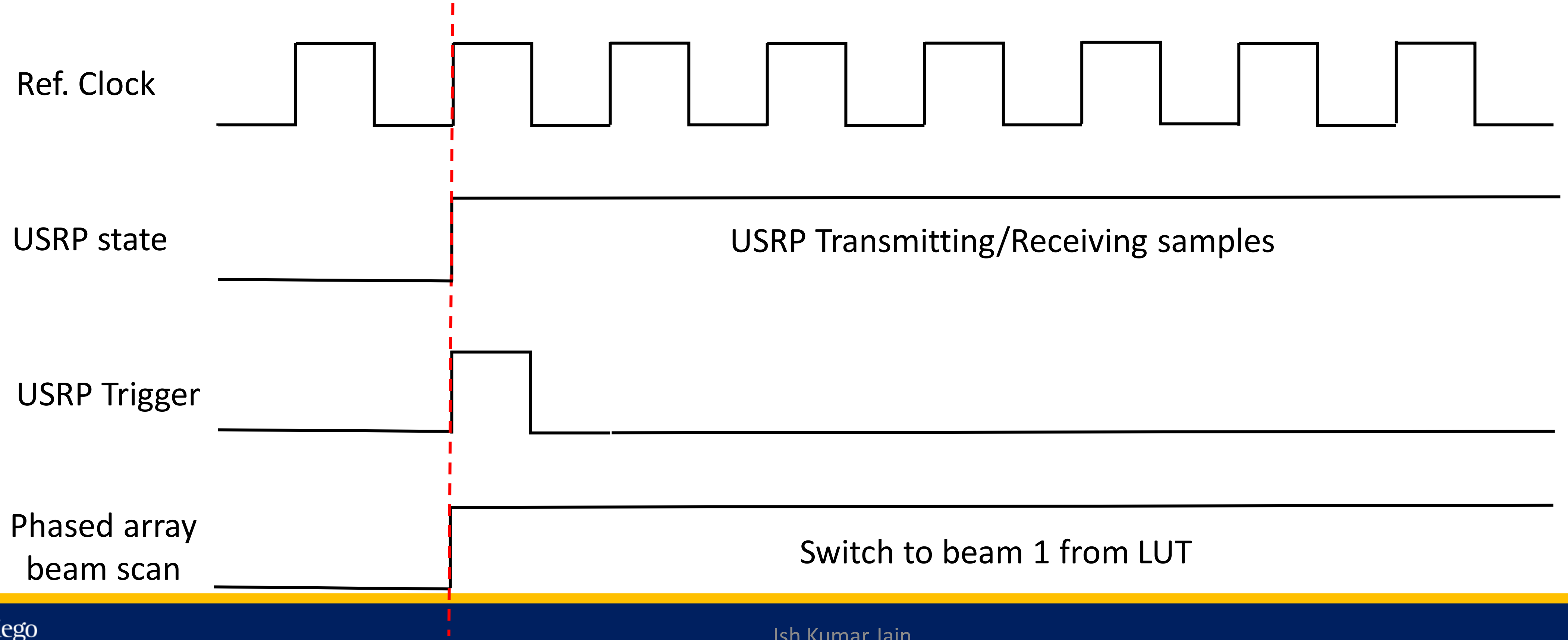
Powerful pre/post processing tools	Flexible and Fast I/O interface	GNURadio support for development
Quick Prototyping	Sample level synchronization	Provides Clock and Trigger



How we achieve synchronization?

Synchronization with USRP:

- Both USRP and FPGA have the same reference clock
- Trigger from USRP upon beginning of transmission or reception
- Enables sample level synchronization between USRP and Phased Array.



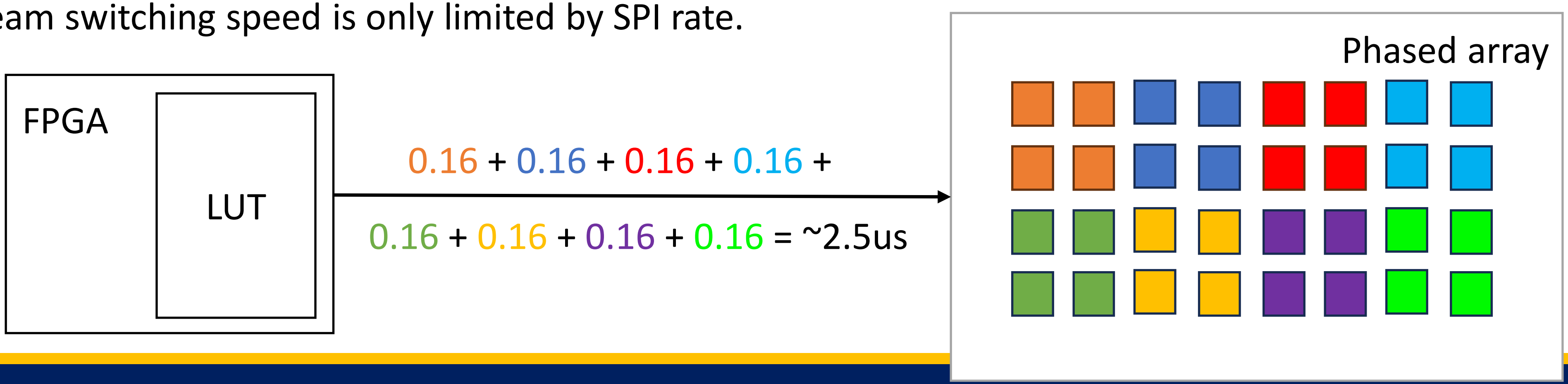
How we get fast Beam switching to 3 us?

Fast beam switching:

- SPI operates up to 50MHz = 50 Mbps
- One SPI byte level transfer = 1/50 * 8 = 0.02 * 8 = 0.16us
- Loading beam configuration to each beamformer chip requires a 2-byte SPI command
- For a 32-element array, with one beamformer chip for every 4 elements, beam switching can be done within 2 bytes * 0.16us / byte * 32 / 4 = ~2.5us.
- Thus, beam switching can be done under 3us.

LUT:

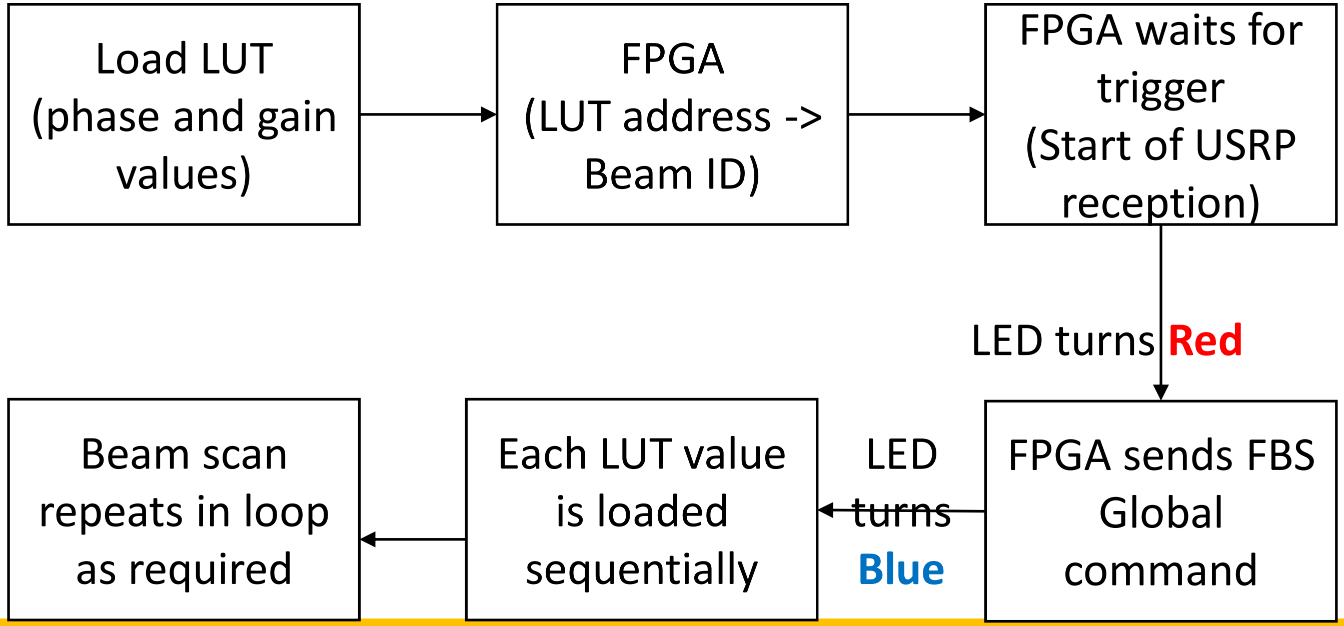
- LUTs are used to store the beam configurations within the FPGA.
- Mitigates the need to calculate beam configuration values every time before sending SPI command.
- Thus, beam switching speed is only limited by SPI rate.

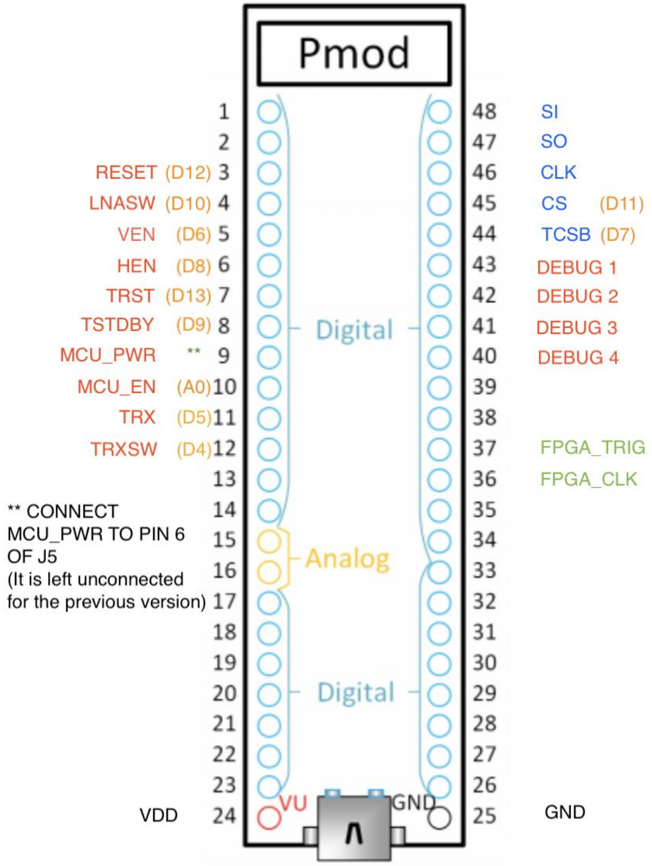


FPGA Programming

- Synchronized with USRP clock, either by using the USRP clock as reference or feeding a common reference clock for both FPGA and USRP.
- Receives trigger from USRP, indicating the start of reception of signal.
- Receives commands from MATLAB through UART.
- Decode and extract the SPI command embedded within the UART message, according to the mapping shown in previous slide.

Fast Beam Scan:





FPGA Programming details

Load LUT:

- The LUT address is calculated in the FPGA and appended into the SPI message extracted from UART (data[1] and data[2]).
- The mapping with Beam ID is stored (trackerLUT).

```
int chipAddress = (data[0] & 60) >> 2;
int LUTAddress = floor(trackerLUT[chipAddress] / 4);

data[1] = LUTAddress >> 3;
data[2] = (data[2] | (LUTAddress << 5));

trackerLUT[chipAddress] = trackerLUT[chipAddress] + 1;
```

FPGA waits for trigger:

```
do{FreezeSignal =XGpio_DiscreteRead(&Gpio, GPIO_CHANNEL);// Continuously reading the GPIO pins
}while((FreezeSignal&1)!=0);
```

Fast Beam Scan:

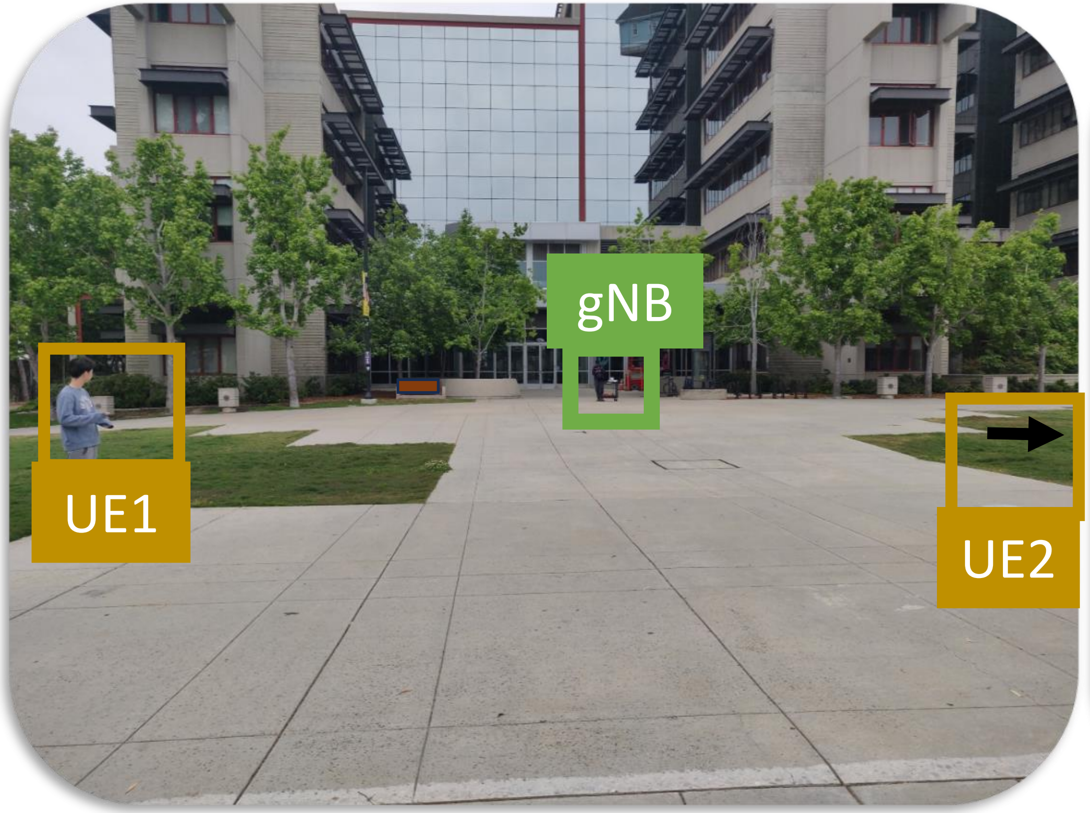
- Preload the SPI messages in array to avoid calculation during beam scan (loadedData).
- Send the preloaded SPI messages sequentially.
- Wait in the beam configuration for the switch period interval.
- Repeat the beam scan for the number of codebook repetitions (numCBrep).

```
for (int addressLUT = 0; addressLUT < CBSize; addressLUT++){
        loadedData[addressLUT * 2] = (1 << 7) | (pol << 5) | (mode << 4) | (addressLUT >> 7);
        loadedData[addressLUT * 2 + 1] = (addressLUT << 1) | latchEN;
    }
}
```

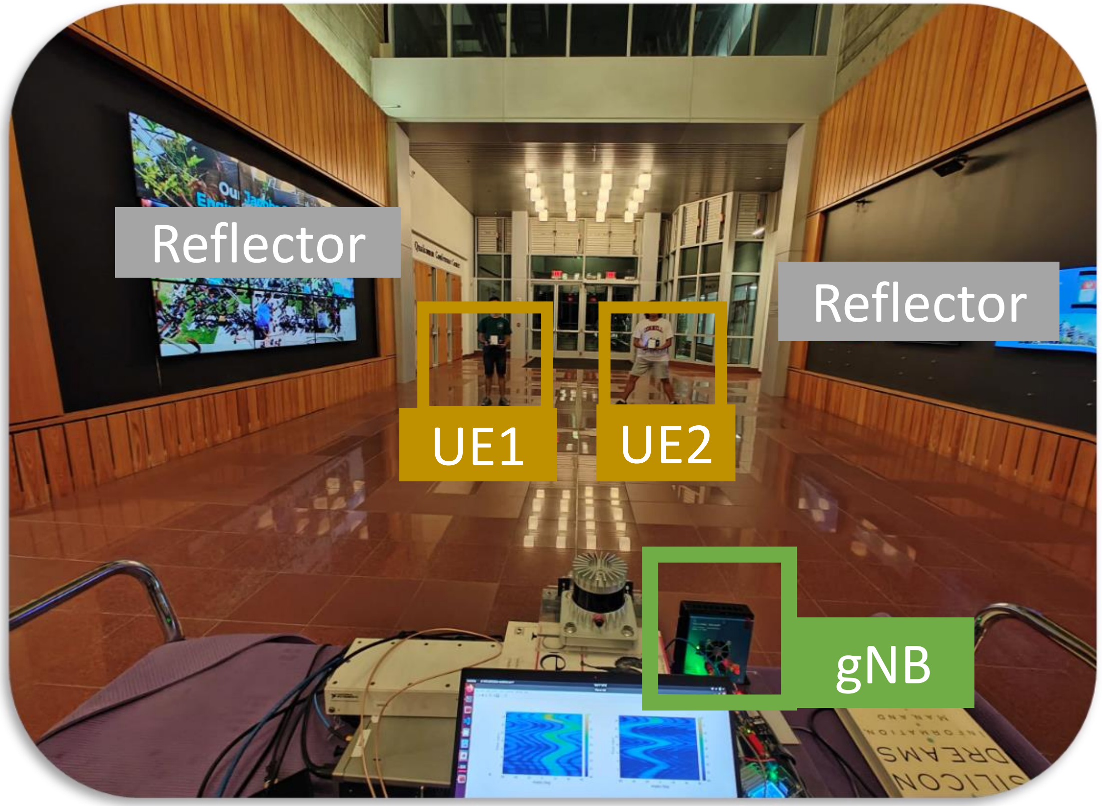
```
for(int rep = 0; rep < numCBrep; rep++){
    for (int addressLUT = 0; addressLUT < CBSize; addressLUT++){
        XSpi_Transfer(SpiInstancePtr, &loadedData[addressLUT * 2], ret, 2);
        do{ TimerValue=XTmrCtr_GetTimerCounterReg(TMRCTR_BASEADDR, TIMER_COUNTER_0);
        }while(TimerValue>=100);

    }
```

Experimental scenario



Outdoor Scenario

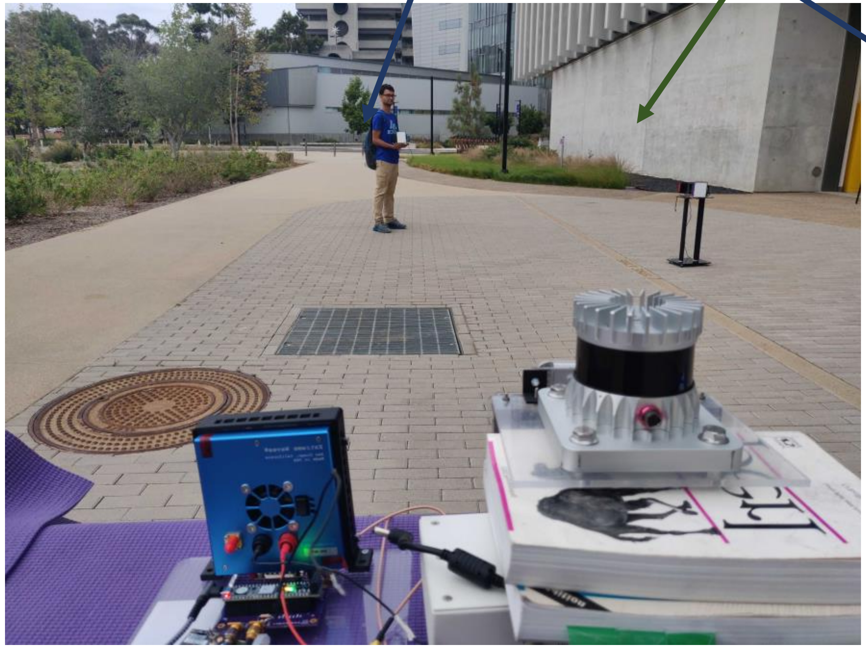


Indoor Scenario

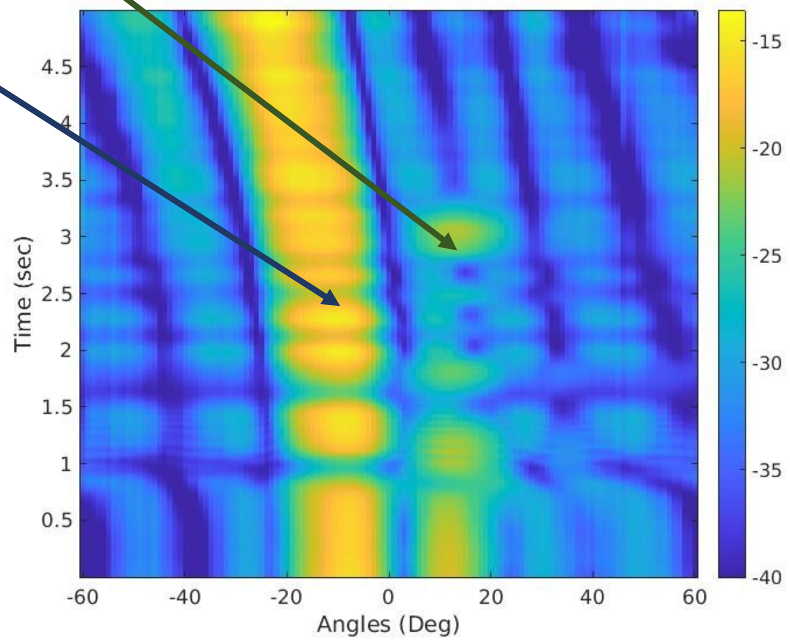
Users and Reflector measurement

User

Reflector



Experiment Scenario

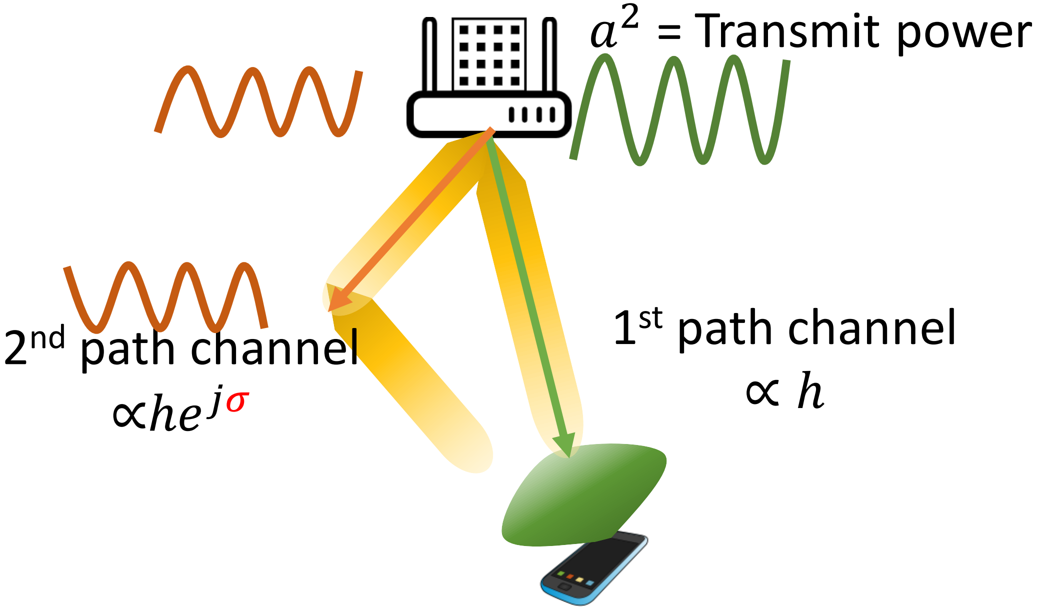


Beam scan measurement

mmReliable: Two beams are better than one

Destructive Multi-Beam

Constructive Multi-Beam

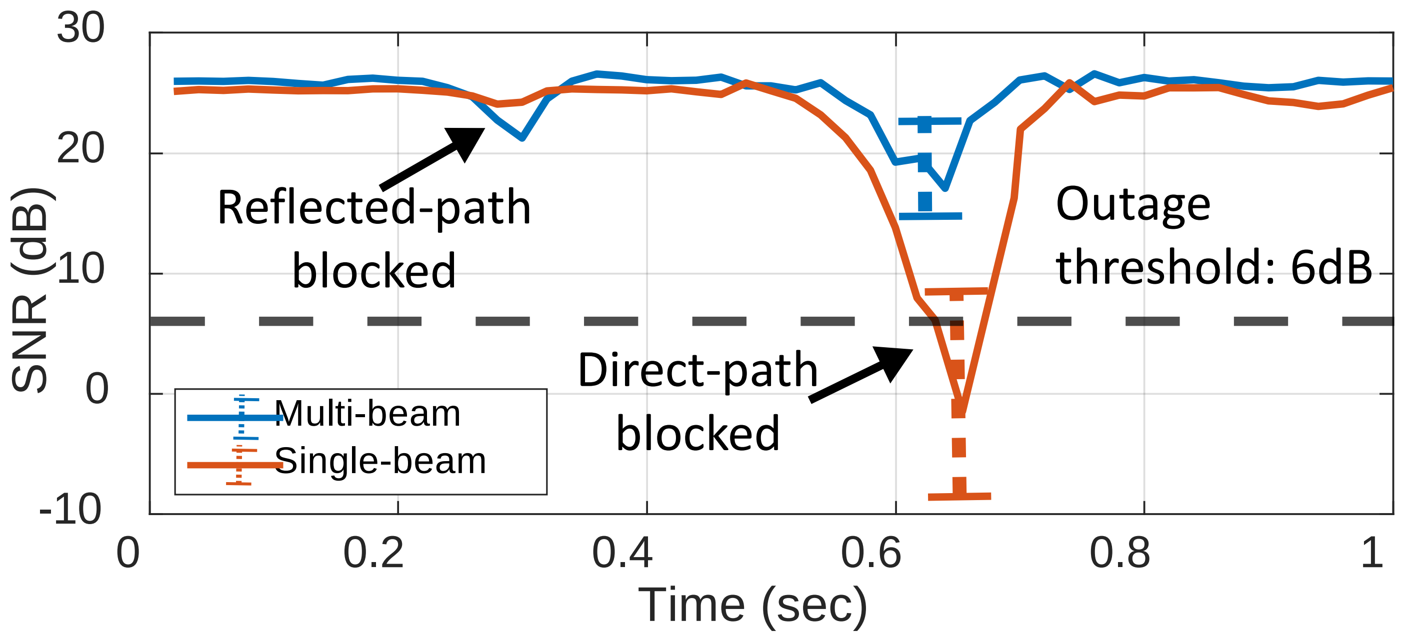


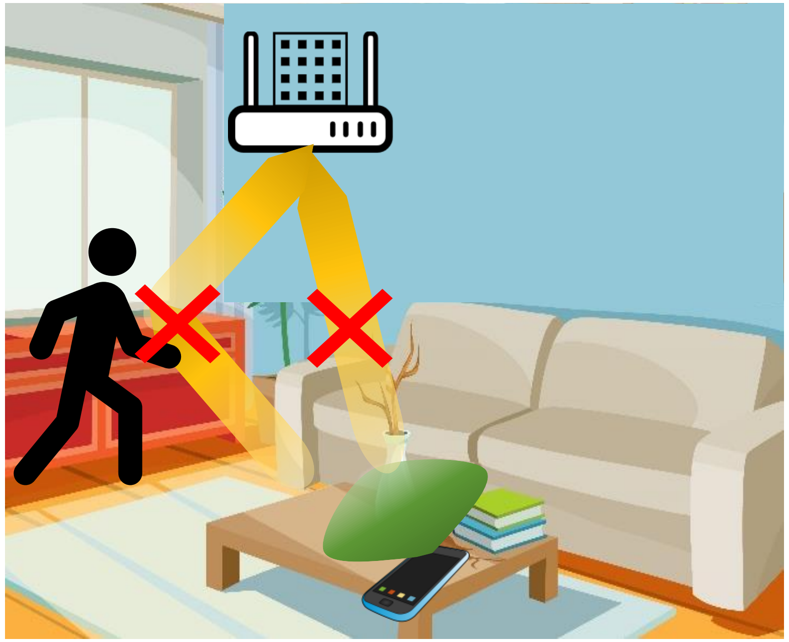
Multi-Beam

CMB SNR	Single-beam SNR
$\propto \lvert\sqrt{2}ah\rvert^2 = 2a^2\,\lvert h\rvert^2$	$\propto a^2\,\lvert h\rvert^2$

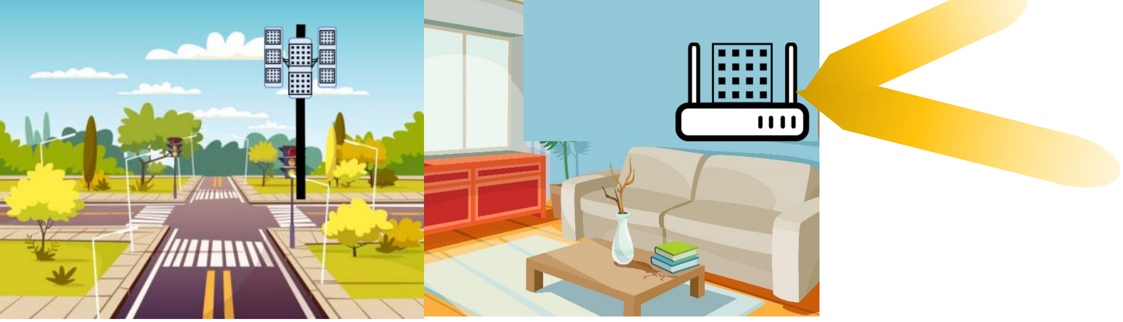
Constructive multi-beams (CMB) requires **phase and power control** to achieve 2x better SNR and higher throughput than single beam

Constructive multi-beams are resilient to blockages

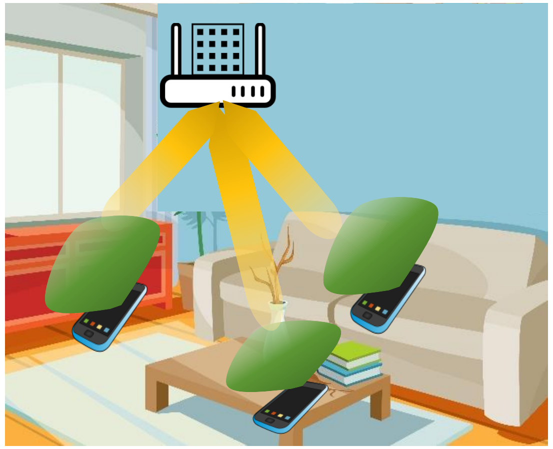


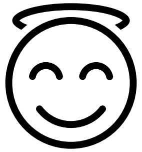


Multi-beam maintain high throughput despite occasional blockages



Publications	[mmReliable] Two beams are better than one: Towards reliable and high throughput mmWave links, ACM SIGCOMM 2021
	mmFlexible: Towards Flexible Directional Multiplexing for Multi-user mmwave Networks, IEEE INFOCOM 2023
	mMobile: Building a mmwave testbed to evaluate and address mobility, ACM mmNets workshop 2020
	mmSpoof: Resilient Spoofing of Automotive Millimeter-wave Radars using Reflect Array, IEEE S&P 2023





Ish Kumar Jain – wcsng.ucsd.edu