M-Cube: A Millimeter-Wave Massive MIMO Software Radio

Renjie Zhao
University of California San Diego
r2zhao@ucsd.edu

Timothy Woodford
University of California San Diego
twoodfor@ucsd.edu

Teng Wei*
University of California San Diego
sjwt2009@gmail.com

Kun Qian
University of California San Diego
kuq002@ucsd.edu

Xinyu Zhang
University of California San Diego
xyzhang@ucsd.edu

ABSTRACT

Millimeter-wave (mmWave) technologies represent a cornerstone for emerging wireless network infrastructure, and for RF sensing systems in security, health, and automotive domains. Through a MIMO array of phased arrays with hundreds of antenna elements, mmWave can boost wireless bit-rates to 100+ Gbps, and potentially achieve near-vision sensing resolution. However, the lack of an experimental platform has been impeding research in this field. This paper fills the gap with M^3 (M-Cube), the first mmWave massive MIMO software radio. M^3 features a fully reconfigurable array of phased arrays, with up to 8 RF chains and 256 antenna elements. Despite the orders of magnitude larger antenna arrays, its cost is orders of magnitude lower, even when compared with state-of-the-art single RF chain mmWave software radios. The key design principle behind M^3 is to hijack a low-cost commodity 802.11ad radio, separate the control path and data path inside, regenerate the phased array control signals, and recreate the data signals using a programmable baseband. Extensive experiments have demonstrated the effectiveness of the M^3 design, and its usefulness for research in mmWave massive MIMO communication and sensing.

CCS CONCEPTS

• **Networks → Programming interfaces**; • **Hardware → Wireless devices**; *Signal processing systems.*

KEYWORDS

60 GHz, Millimeter-wave, MIMO, Software radio, Testbed, Experimental platform

ACM Reference Format:
Renjie Zhao, Timothy Woodford, Teng Wei, Kun Qian, and Xinyu Zhang. 2020. M-Cube: A Millimeter-Wave Massive MIMO Software Radio. In *The 26th Annual International Conference on Mobile Computing and Networking (MobiCom '20), September 21–25, 2020, London, United Kingdom.* ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3372224.3380892

*Teng Wei contributed to this work when he was a visiting student in UC San Diego.

MobiCom '20, September 21–25, 2020, London, United Kingdom

ACM ISBN 978-1-4503-7085-1/20/09…$15.00
https://doi.org/10.1145/3372224.3380892

1 INTRODUCTION

Millimeter-wave (mmWave) networking technologies are widely recognized as the most promising solution to confront the mobile data explosion. However, commercially viable use cases, *e.g.*, 60 GHz 802.11ad and 70 GHz backhaul, have been limited to short-range, static, point-to-point settings. The fundamental reason lies in the use of highly directional beams as the communication medium, which can be easily disturbed by obstacle blockage and device movement. These challenges become most severe when a large phased array is used, with a massive number of antenna elements (and hence a large number of directional beams to manage).

In addition, mmWave devices can serve as RF sensors to achieve high spatial resolution, owing to their intrinsically shorter wavelength, wider bandwidth, and larger antenna aperture [32]. Besides conventional use cases such as vehicular radar ranging and security/medical imaging, mmWave sensing is becoming available on pervasive mobile devices. For example, the 5G NR standard has incorporated mmWave location sensing [59]. Meanwhile, the emerging 802.11ay standard also introduces a WLAN radar mode which repurposes the mmWave radio as a MIMO radar [1].

To fully explore the challenges and opportunities in mmWave technologies, it is critical to have a programmable experimental platform with the following capabilities: *(i)* Equipped with low-cost and large-scale phased arrays which allow real-time beam switching, to accommodate high mobility vehicular networking/sensing scenarios; *(ii)* Supporting the mmWave MIMO architectures to be used in 5G NR and 802.11ay radios [22, 37]; *(iii)* Allowing reconfiguration of beam patterns, communication/sensing algorithms and network stack. Existing mmWave experimental platforms are either too costly (around $200K per link [33, 39]), or lack a reconfigurable phased array antenna with reasonable size [39, 48, 65]. Moreover, such devices are often bulky and can barely support mobile experiments. None of the existing platforms include support for both multiple RF chains and reconfigurable phased arrays, which are critical for research into mmWave MIMO.

In this paper, we describe the design and implementation of M^3, the first mmWave massive MIMO experimental platform to meet the aforementioned requirements. M^3 is a low-cost software-defined radio/radar comprised of up to 256 antenna elements and up to 8 RF chains. The key research thrust in M^3 is to repurpose a commodity 802.11ad phased array as a programmable phased array, and to interface it with an existing baseband processing unit (BPU), such as an FPGA with data converters, or a low-frequency software radio. M^3's software radio/radar design cuts the per-node cost significantly, *e.g.*, down to $3.8K for a narrowband (56 MHz)

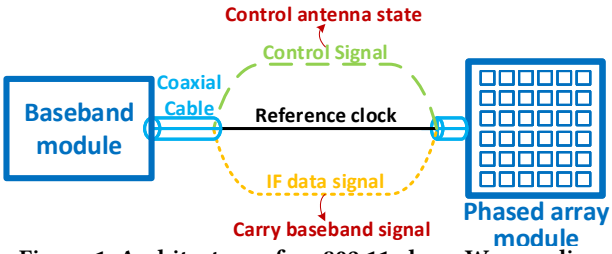

Figure 1: Architecture of an 802.11ad mmWave radio.

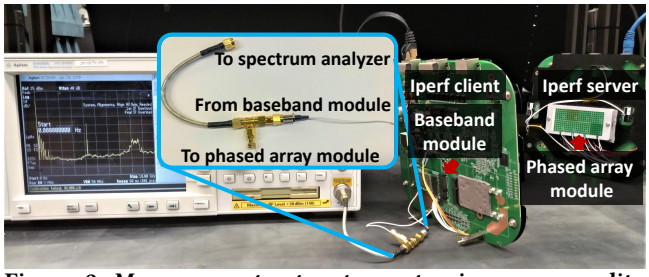

Figure 2: Measurement setup to anatomize a commodity split-IF 802.11ad radio.

2 RF-chain 72-antenna mmWave MIMO, and below $15K for a wideband (4 GHz) 4 RF-chain 128-antenna version.

The key observation behind the M^3 design is that many modern mmWave radios [9, 42] adopt a split-IF (intermediate frequency) architecture as shown in Fig. 1. The baseband-to-IF and IF-to-RF-plus-antenna modules, henceforth referred to as *baseband module* (BM) and *phased array module* (PM), are realized in two separate chips, connected through a single coaxial cable that carries both data and control signals. By reverse engineering the *control channel*, and regenerating the control signals using an external FPGA, we gain full access to the phased array, including reconfiguring its codebook entries (beam patterns), triggering beam scanning, selecting and switching between the beam patterns in real-time, and tuning the individual antenna element gain. For the *data channel*, we replace the original 802.11ad BM with a customized BPU, along with a *bridge board* that interfaces the BPU and the PM. The bridge board is designed such that it can take as input/output either baseband I/Q signals or modulated RF signals below 4 GHz. With this board, the low-cost commodity phased array can be attached to any existing BPU, such as a USRP, WARP, or customized FPGAs.

To extend this architecture to a mmWave MIMO setup, we found that recently emerged multi-phased-array 802.11ad radios [3, 30] provide the same data channel to multiple carrier-synchronized phased arrays, and can switch on one or more of them simultaneously. By interfacing each phased array to a separate bridge board and separate ADC/DAC channel on the BPU, we can construct a hybrid beamforming architecture with up to 8 RF chains, each attached to a 6×6 phased array. Furthermore, by clock-sharing between the transmitter and receiver path, we can convert an M^3 node into a *software-defined mmWave MIMO radar* with a massive number of phased array elements. In addition, the phased arrays can be rearranged into a ring or cube layout, to expand the field-of-view to the entire 3D space.

We have conducted comprehensive measurement and testing to validate the feasibility and effectiveness of the M^3 design. Our key findings include: *(i) Reconfigurability.* After a one-time calibration, the multiple phased arrays on M^3 can be reconfigured separately to generate desired beam widths and directions. M^3 enables real-time mmWave MIMO communication, channel measurement, and radar sensing. *(ii) Control path performance.* M^3 can control the beam switching in real-time with a latency of 412 ns, which is commensurate with commodity 802.11ad radios. *(iii) Data path performance.* The bridge board design in M^3 does not degrade the phase noise and signal to noise ratio (SNR) level. Depending on the sampling frequency of the BPU, it can achieve an end-to-end SNR of 19 dB.

Furthermore, we conducted two case studies to demonstrate the application of M^3 in exploring mmWave MIMO networking and sensing systems. *(i) Context-aware mmWave MIMO hybrid beamforming.* We implement a mmWave MIMO OFDM framework to characterize the single-user MIMO and multi-user MIMO performance in indoor/outdoor environments. Our experiments reveal the need for multipath context-aware MIMO mode adaptation. *(ii) mmWave MIMO radar with uniform and non-uniform array layout.* We implement a multi-phased-array radar with non-uniform array layout, and demonstrate its higher angular resolution in comparison with single array or uniform arrays.

The key contributions of M^3 can be summarized as follows: *(i)* Designing the data path to bridge programmable baseband processors with low-cost commodity phased array modules. *(ii)* Designing the control path to reconfigure and control the phased array with sub-μs latency. *(iii)* Restructuring the commodity 802.11ad radio into a massive MIMO mmWave radio/radar. *(iv)* Experimental verification of the M^3 architecture and performance, along with new measurement insights for mmWave MIMO radio/radar systems. To our knowledge, M^3 represents the first-of-its-kind programmable mmWave MIMO platform. We will follow the WARP project model [54] to make M^3 available to the wireless research community, through open-source hardware and paid fabrication/assembly services. The code, documentation and further information will be released through the project website, http://m3.ucsd.edu/sdr/.

2 ANATOMY OF COMMODITY 802.11AD MMWAVE RADIO

This section presents our reverse engineering work on a commodity 802.11ad radio, which serves as the basis of the M^3 design.

Mainstream 802.11ad network interface cards (NICs) [9, 38, 58] all follow a modular *split-IF architecture* as illustrated in Fig. 1. The NIC comprises two modules connected via a *coaxial cable*: a *baseband module (BM)*, which converts between baseband signals and IF signals; and a *phased array module (PM)*, comprised of the phased array antenna and RF front-end (converting between the IF signals and 60 GHz RF signals). Unlike low-frequency radios, the antenna and RF chain are integrated on the same module rather than connected via a cable because routing mmWave signals across even a few millimeters leads to high losses [42].

To tap into the PM, we use a spectrum analyzer (Keysight E4448A) to monitor the coaxial interface on a commodity 802.11ad radio from Airfide Inc. [3], which realizes the split-IF architecture through a dual-chip solution—A Qualcomm QCA6335 baseband, and QCA6310 RF front-end integrated with one or more 6×6 phased arrays. As

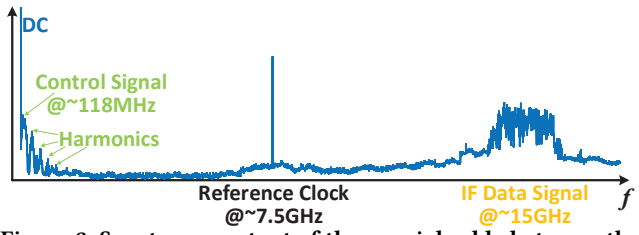

Figure 3: Spectrum content of the coaxial cable between the baseband module and phased-array front-end on a commodity 802.11ad radio.

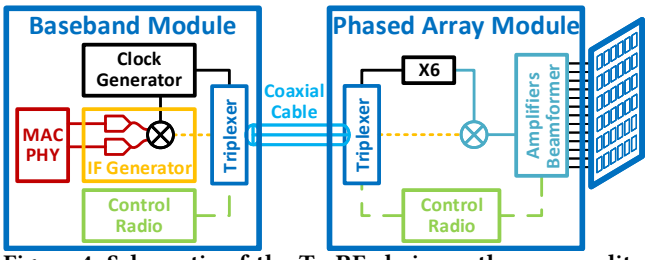

Figure 4: Schematic of the Tx RF chain on the commodity 802.11ad radio (Rx chain is similar).

shown in Fig. 2, we use a 3-port splitter (Pomona #72969), SMA adapters and cables to enable the spectrum analyzer to "eavesdrop" on the coaxial cable between the BM and PM. The radio transmits data continuously through iPerf, while the normal 802.11ad protocols are in operation (*e.g.*, beam scanning). The spectrum content (Fig. 3) shows that the coaxial cable carries 4 types of signals: a DC power supply, an IF reference clock at around 7.5 GHz, control signals around 118 MHz, and IF data signals around 15 GHz. The spectrum composition shares similar principles with a recent split-IF 802.11ad chipset design from Broadcom [9], although with different frequency planning and phased array structure. Based on our measurement and insights from [9], the schematic of the Qualcomm 802.11ad NIC can be reconstructed as in Fig. 4. Below we present an anatomy of the signals passing between the BM and PM.

Control signal: This signal configures key parameters of the PM, customizing the codebook, beam selection, RF gain, power amplifier gain on individual antennas, probing the phased array status, *etc.* The NIC is usually attached to a host PC through PCIe M.2. Parameter configuration is initiated by issuing a wireless module interface (WMI) command [44] from the host PC running the wil6210 802.11ad driver [25]. The firmware on the BM receives the command and forwards it to the PM. Both hardware modules have a digital modem for control commands, which manages the modulation/demodulation, error correction, *etc.*. The modulated control signal resides on a low frequency (118 Msps symbol rate) to avoid interference with the data channel. The control channel allows for two modes of operation: *(i)* Register read-write mode, wherein the BM can control/probe the status of the PM; *(ii)* Streaming mode, wherein a customized codebook matrix can be loaded from the BM to the PM. The PM maintains a simple state-machine that sets the status of the digital modem.

IF data signal: As in typical superheterodyne radios, the commodity 802.11ad transmitter first converts the baseband I/Q signal

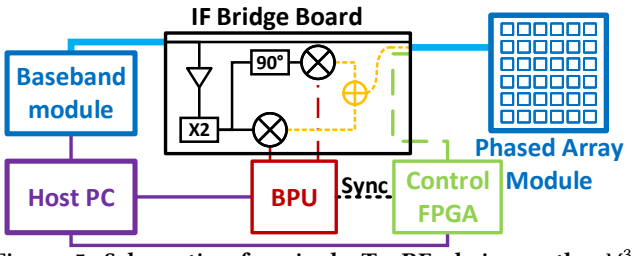

Figure 5: Schematic of a single Tx RF chain on the M^3 mmWave MIMO software-radio (Rx chain is similar).

to an IF analog I/Q signal, at IF carrier frequency 15 GHz (2× the reference clock signal passing through the coaxial interface). Therefore, we can send arbitrary signals through the PM, as long as they are within the passband centered at 15 GHz. The PM integrates IF-to-RF up/downconversion chains, IF amplifiers, power management units, as well as the necessary building blocks for the phased array itself (*e.g.*, phase shifters and RF power amplifiers).

Reference clock: Mainstream mmWave chipsets [9, 13, 14] typically use a sliding IF architecture to achieve mmWave signal upconversion and channelization. In the Qualcomm 802.11ad radio, the BM provides a reference clock around 7.5 GHz (switching between 7.29, 7.56, 7.83 and 8.10 GHz) to the PM, enabling it to switch among the four 802.11ad channels (centered at carrier frequencies 58.32, 60.48, 62.64 and 64.80 GHz). The reference clock signal passes through the coaxial cable, and then a ×6 multiplier is used to generate the local oscillator (LO) for the PM. For example, with $7.56 \times 6 = 45.36$ GHz LO and $7.56 \times 2 = 15.12$ GHz IF signal, the output RF signal is $45.36 + 15.12 = 60.48$ GHz.

3 OVERVIEW: M^3 ARCHITECTURE

In order to transform the commodity 802.11ad radio into a software radio, our basic idea is to *reuse the baseband module as a clock/power generator and boot loader, but regenerate the control signal using an FPGA-based digital controller, and create a customized data channel by using a programmable BPU plus a baseband-to-IF converter (referred to as a bridge board).*

As a fundamental architectural level design choice, M^3 separates the data channel and control channel and makes both reconfigurable. As illustrated in Fig. 5, the Tx bridge board reuses the 802.11ad BM's 7.5 GHz clock signal as a clock source, and converts it to 15 GHz IF. It then takes the BPU's baseband I/Q signal or modulated low-frequency signal as input, and mixes it with the IF clock signal to create data signal at 15 GHz IF. As for the control path, we reverse engineer the control channel waveform, and regenerate the control commands using a low-profile *control FPGA*. The bridge board then combines all three signal paths–the 15 GHz IF data signal, the 118 MHz control signal, and the 7.5 GHz clock and 3.3 V DC power supply from the BM, and injects them into the PM. The Rx path follows the same architecture with a reversed data path direction.

This single RF-chain design can be easily extended to a multi-RF-chain MIMO mmWave architecture as illustrated in Fig. 6. In MIMO mode, the same 7.5 GHz clock source and power source is generated by a single QCA6335 802.11ad BM and shared among multiple PMs, ensuring carrier synchronization at RF frequencies. Each RF-chain

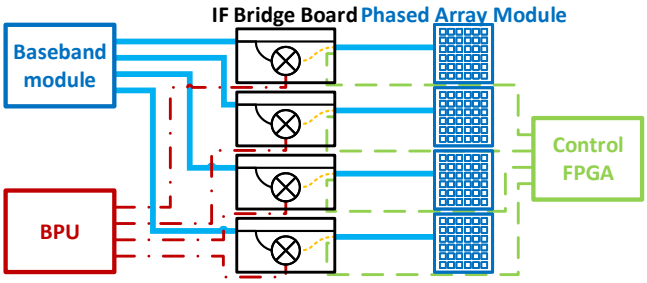

Figure 6: Integrating multiple RF chains to form the MIMO mmWave RF front-end.

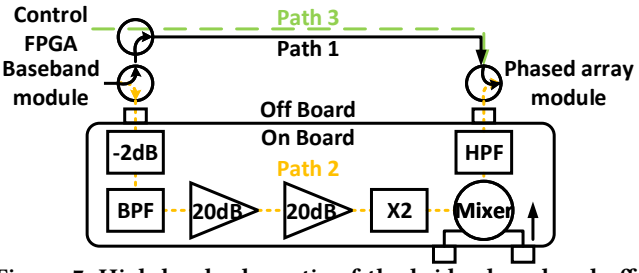

Figure 7: High level schematic of the bridge board and off board connection.

has a separate control channel and data channel, generated by the control FPGA and a multi-channel BPU, respectively. A similar architecture can be used to build a software-defined mmWave MIMO radar.

In terms of cost, we note that the commodity Airfide 802.11ad radio (comprised of 8 phased arrays, 256 antenna elements in total) costs below $700 [3]. Four pairs of Tx and Rx bridge board prototypes cost around $1.8K for the components, $650 for PCB fabrication and $1.3K for assembly (price drops significantly as volume increases). So the necessary RF front-end building blocks to realize 4×4 mmWave MIMO (256 elements) only costs $4.5K in total. A lower profile 2×2 MIMO and single RF chain cost $2.6K and $1.6K, respectively. When counting the BPU cost, a narrow bandwidth 2×2 MIMO (*e.g.*, using USRP B210, a 2-channel 56 MHz BPU at $1.2K) costs only $3.8K. Even when counting the high-profile 4-channel BPU USRP N310 ($10K, 125 MHz bandwidth) or 4-channel Xilinx UltraScale+ RFSoC ($9K, 4 GHz bandwidth), the entire 4×4 mmWave MIMO software radio (with 256 Tx/Rx antenna elements) costs below $15K. The cost is significantly lower even when compared with the state-of-the-art single RF-chain platforms such as X60 [39] ($170K, 12-element phased array), and OpenMili ($15K, 4-element phased array) [65, 66].

4 DATA PATH DESIGN

In this section, we describe the data path design in detail. Without loss of generality, our description focuses on the Tx path. The Rx path simply follows the reverse flow.

4.1 Bridge Board Design

As shown in Fig. 7, the bridge board comprises three paths. ***Path 1*** connects the QCA6335 BM directly to the QCA6310 PM. Through this path, the BM provides the 3.3 V DC power supply, and 7.5 GHz reference clock for the carrier LO generation at the PM. The normal control commands, such as loading a customized codebook, can still be issued from a PC host, routed by the BM through this path, and eventually executed by the PM.

Path 2 is the *bridging path*, which uses the 7.5 GHz reference signal from the BM to generate the IF data signal. The reference clock is generated from the on-chip PLL of QCA6335, optimized for high stability and low phase noise. By avoiding regenerating the clock, we can reduce the complexity and cost of the bridge board substantially. To generate the IF data signal, the reference clock signal first goes through a ×2 frequency multiplier to generate the 15 GHz IF clock, which is then mixed with baseband data signals through a passive image rejection I/Q mixer HMC8191.

Before the mixer, a 6850-7850 MHz band pass filter (BPF) BFCN-7331+ is chosen to reject the signals other than the reference clock. However, BFCN-7331+ has a return loss of 0.13 dB @100 MHz (around 97.3% power reflected) [31], which will cause strong reflections to the 118 MHz control signals, creating an interfering "multi-path" effect. Therefore, a 2 dB 0-25 GHz attenuator HMC652LP2E is placed before the BPF to weaken the reflected signal to prevent it from corrupting the control commands.

We chose the passive I/Q mixer HMC8191, because it interoperates with a wide range of LOs (6 GHz to 26.5 GHz) and can be used for direct I/Q modulation or image reject mixing, needed by the homodyne and heterodyne interfaces to the BPU (Sec. 4.2). The mixer requires a stable 14-20 dBm LO to achieve stable performance, so we use an active ×2 frequency multiplier HMC814 which has flat ~17 dBm output over 2-6 dBm input. Considering the around -30 dBm reference clock input, 2 dB attenuation and 1.5 dB insertion loss of the BPF, two additional 20 dB amplifiers PMA3-83LN+ are used to reach $-30 - 2 - 1.5 + 20 + 20 = 6.5$ dBm power, which matches the input requirement of the ×2 multiplier. Besides the 15 GHz signal, the output of the mixer also contains a signal at the 7.5 GHz fundamental frequency, which acts as a noise to the reference clock to the phased array module. We thus add a 13-19 GHz wideband high pass filter (HPF) XHF-143M+ at the output of this path to filter out this noise. This is a reflectionless filter with around 26 dB return loss at 100 MHz (0.2% power reflection). The reflection of HPF is much lower than BFCN-7331+ and will not destroy the control signal.

Path 3 routes control commands from the control FPGA to the PM. Although the BM can send control commands through path 1, path 3 is still necessary for real-time control, as detailed in Sec. 5. In current version of the bridge board, path 1 and path 3 are implemented with off-board coaxial components to preserve the flexibility of signal strength balancing. These components will be integrated to a PCB in a future version, making the bridge board even more compact.

In the prototype version, we utilize off-board connectors for more convenient debugging and circuit performance evaluation (Fig. 7). Path 1 and 3 pass through the off board connection and the splitters are using Tee connectors which has low isolation. Although a power splitter with higher isolation would solve the reflected signal issue introduced in path 2, the available ultra wideband splitter covering all the signals locating from DC up to 15+ GHz would introduce

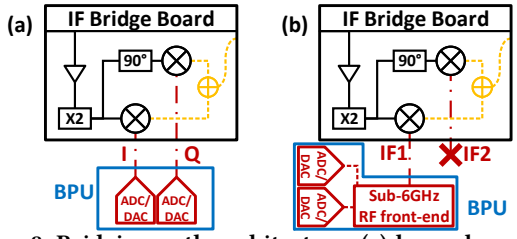

Figure 8: Bridging path architecture: (a) homodyne and (b) heterodyne.

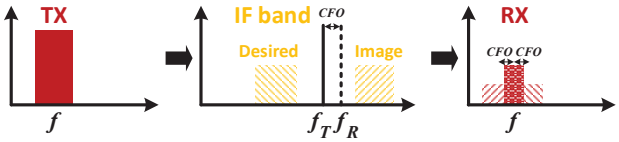

Figure 9: In the heterodyne architecture, upconversion at the Tx bridge board produces two components in IF band. Due to CFO, downconversion at the Rx board imperfectly combines the two components, which severely reduces SNR.

additional insertion loss. Then the reference clock and baseband command power would be too low for the RF module to receive them.

The Rx bridge board shares most of its components with the reverse signal path, except that two 5-18 GHz 13 dB amplifiers (AVA-183A+) are added between the HPF and mixer to ensure proper input power to the BPU.

4.2 Bridging Path Architecture

The single bridge board design mentioned above can fit into two different architectures along *path 2*: homodyne and heterodyne, interfacing with BPUs that generate dual-channel I/Q signals and single-channel low IF signals, respectively.

4.2.1 Homodyne Architecture. In the homodyne architecture (Fig. 8(a)), the baseband I/Q signals are directly upconverted to IF with a quadrature LO, which is generated by the mixer using the 15 GHz reference. This architecture is widely used by modern sub-6 GHz radios because it is simple, has lower cost and no image problem.

We chose the HMC8191 mixer with DC to 5 GHz IF bandwidth, to ensure interoperability with the wideband I/Q input from DAC centered at DC. The upconverted double side band signal centered at 15 GHz will also be able to pass through the HPF. Note that the filter chosen here is for a wideband use case. An additional low pass filter will be needed at baseband to filter out the harmonics caused by the discrete signal generated by DAC.

We implement this homodyne architecture using two different BPUs built by combining an ADC/DAC module with an FPGA. *(i) The FMC150 BPU* uses a Virtex-6 LX240T FPGA with an FMC150 ADC/DAC board supporting a 40 Msps I/Q sampling rate. We developed the FPGA bitstream to be compatible with the WARP v3 [28] PC host driver. *(ii) The FMCDAQ2 BPU* uses a Xilinx KCU105 development board with an FMCDAQ2 1 Gsps ADC/DAC a Kintex Ultrascale XCKU040 FPGA running the open-source FPGA bitstream developed in [65]. Detailed evaluation of M^3 with these BPUs is in Sec. 9. It is possible to achieve a sampling rate compatible with 802.11ad or 802.11ay, *e.g.* using Xilinx Zynq UltraScale+ RFSoC [62] which has 8 4 Gsps ADCs and 8 6.5 Gsps DACs, but this is beyond the scope of our current work.

4.2.2 Heterodyne Architecture. For the heterodyne architecture, a low-frequency software-defined radio (SDR), *e.g.*, USRP, first generates a carrier-modulated first-stage sub-6 GHz IF signal, then the bridge board acts as a second stage IF mixer to upconvert the signal into the desired 15 GHz IF data signal.

Unfortunately, we cannot directly interface the single-output signal from the SDR with the dual-input quadrature mixer on the bridge board as shown in Fig. 8(b) due to an *image problem*. As illustrated in Fig. 9, consider a Tx SDR generating a signal $m(t)sin(2\pi ft)$, with first stage IF f and baseband signal $m(t)$. With Tx LO f_T, the signal at IF band is:

$$m(t)sin(2\pi ft)sin(2\pi f_T t) = \frac{m(t)}{2}[cos(2\pi(f_T - f)t) - cos(2\pi(f_T + f)t)],$$

which has two components at frequency $f_T - f$ and $f_T + f$ called *desired signal* and *image signal*, respectively. When $f < 2$ GHz, the two components are both located in the passband of the filters on bridge board and the phased array (*i.e.*, 13 GHz~17 GHz), so both components will be transmitted through the phased array. This image incurs two problems: *(i)* it will cause a waste on the frequency band. *(ii)* it will introduce "self-interference" at the RX side when carrier frequency offset (CFO) exists. More specifically, at the Rx side, the signal is downconverted from IF using carrier frequency f_R to become:

$$\frac{m(t)}{2}[cos(2\pi(f_T - f)t) - cos(2\pi(f_T + f)t)]sin(2\pi f_R t) = \frac{m(t)}{4}[sin(2\pi(f + (f_R - f_T))t) + sin(2\pi(f - (f_R - f_T))t)].$$

Because there is no way to perfectly eliminate the CFO, the desired and image signal components are separated by $2 \times (f_R - f_T)$. Since the CFO value $(f_R - f_T)$ is typically tens of kHz, the two components will be offset in frequency domain and thus interfere with each other where they overlap. There is no simple post-processing solution to separate them, so the final SNR will be extremely low.

In M^3, we explore two methods to overcome this challenge on the Tx side:

(i) Adding a 90° *hybrid coupler* between the bridge board and the baseband SDR to leverage the image rejection function of the HMC8191 mixer. The 90° hybrid coupler divides the single-channel low-IF signal into two channels with a 90° phase offset, which are then fed into IF1 and IF2 respectively as in Fig. 8(b). The output signal from the bridge board becomes:

$$\frac{m(t)}{2}sin(2\pi ft)sin(2\pi f_T t) + \frac{m(t)}{2}cos(2\pi ft)cos(2\pi f_T t) = \frac{m(t)}{2}cos(2\pi(f_T - f)t).$$

which is the desired single side band signal.

(ii) Taking advantage of a built-in filter to reject the image signal. By sweeping a wide frequency range using a high-frequency signal generator, we found the QCA6310 PM contains a bandpass filter with a passband of 12~17 GHz. Therefore, if the LO frequency is 15.12 GHz, the image signals will be directly filtered if the BPU

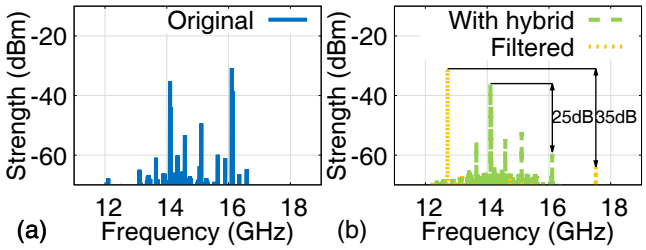

Figure 10: Image rejection of original, higher IF filtered (35 dB) and using 90° hybrid (25 dB).

output is above 17 − 15.12 = 1.88 GHz. This can be easily satisfied on popular BPUs such as USRP and WARP (output up to 5.8 GHz).

To verify the effectiveness of these two methods, we connect a Tx PM (15.12 GHz IF) to a bridge board, and then use a signal generator to generate the low-IF in three setups: *(i)* 1 GHz single tone to one port of the bridge board, *(ii)* 1 GHz single tone to two ports of the bridge board through a 90° hybrid coupler (KRYTAR 3005040), *(iii)* 2.4 GHz single tone to one port of the bridge board. Meanwhile, we connect an Rx PM like the "eavesdrop" setup introduced in Sec. 2 , and plot the result in Fig. 10. Fig. 10(a) shows the double side band received signal at 1 GHz. The image signal has comparable strength with the desired signal. Fig. 10(b) shows the effectiveness of two methods: adding a hybrid coupler leads to around 25 dB image rejection, which is aligned with the HMC8191 performance specification [6]. Using a 2.4 GHz first-stage IF will have around 35 dB image rejection, which will not impede normal data transmissions since the end-to-end SNR is usually much lower. We expect the performance can be further enhanced by combining the two aforementioned methods.

Since adding an additional hybrid coupler or filter will introduce more insertion loss, we need to consider this in the gain budget calculation. For the TX bridge board, the circuit will introduce mixer conversion loss (10 dB), BPF loss (2 dB), and splitting loss (4 dB). To reach the same 15 GHz IF signal strength (~-26 dBm) from the baseband module, the sub-6G IF signals should be larger than (−26 + 4 + 2 + 10 + 1 = −9 dBm), which is easily generated by a power-tunable SDR. Therefore, the gain budget is sufficient for the design and this loss will not degrade the performance.

5 CONTROL PATH DESIGN

To control the phased array, we could reuse the original control channel by issuing WMI commands from the PC host, as in recent research [35, 45]. However, this approach has two fundamental limitations when applied to M^3: *(i) High control latency.* It takes around 20 ms to switch from one beam to another using WMI commands. As a result, sweeping 64 beams following a customized order (needed by many beam management algorithms [23, 60]) will take more than 1.28 s, which will undoubtedly fail to handle link dynamics. *(ii) Lack of support for MIMO and radar mode.* WMI commands can only change all the phased array to the same specific state at a time , even though the QCA6335-based NIC can drive 8 phased arrays. As a result, it is not possible to synchronize phased arrays for MIMO operations, or set different phased arrays to Tx and Rx mode for radar operations.

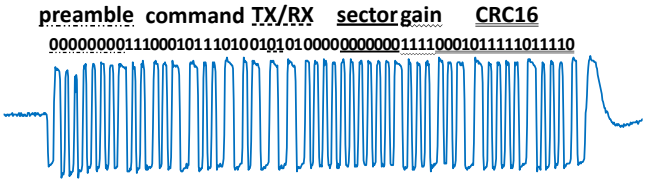

Figure 11: Manchester encoded tx mode control signal for sector 0, gain 15.

5.1 Deciphering and Regenerating Control Signals

To overcome these two barriers, we instead resort to regenerating the control channel using a customized FPGA. To reproduce the control signals, we first used a similar setup as in Fig. 2 to capture the command data with an oscilloscope. *First*, we sent multiple WMI commands through the 802.11ad NIC, changing the Tx/Rx mode, sector index and gain index separately. By identifying the control channel symbols changed for each WMI command, we located the modulated symbols corresponding to each of these three parameters. By comparing the symbols with the known sector indices used in the WMI command, we found the control symbols are consistent with Biphase-L line coding, also known as Manchester coding. The analog waveform and corresponding decoded bits are shown in Fig. 11.

We found that the control signal starts with several zeros acting as a preamble. The actual command begins with a 16-bit command ID, followed by 2 bits indicating whether to activate the Tx (01) or Rx (10) mode, 7 bits for the sector index, and 4 bits for the gain index. In total, there are 128 codebook entry indices and 16 available gain steps (0 disables the RF output entirely, whereas 1 and 15 represent the largest and smallest gain, respectively). The last 16 bits are a cyclic redundancy check (CRC) following the CRC16-CCITT format. The CRC does not count the zero bits in the preamble, so a control signal with a shorter preamble can still be decoded by the phased array module.

In order to regenerate the line coding signals, which operate at a 236 MHz switching frequency (2× 118 MHz due to Biphase modulation), we use the GPIO ports on a control FPGA (Digilent CMOD A7). To match the desired control signal input power of the PM, we need to attenuate the 3.3 V (or 14.5 dBm) GPIO output by around 18 dB to -3.5 dBm. Furthermore, to circumvent the 3.3 V DC power on the coaxial cable, we add a DC blocker to AC couple the bridge board's path 3 with the control FPGA.

5.2 Synchronizing the Control and Data Channel

To synchronize the control FPGA's output with the data samples from the BPU, we leverage the automatic GPIO control functionality built in the BPU, which are synchronized with the start of Tx/Rx samples. We use the GPIO controlled by Automatic Tx/Rx (ATR) on the USRP and trigger output on the WARP. Alternative BPUs such as the FMCDAQ2 and FMC150 (Sec. 4.2) possess high-speed GPIOs, and can even act as the control FPGA themselves.

Certain beam management protocols require rapidly testing a sequence of beam control operations with precise timing and predefined ordering of beams. For example, 802.11ad adopts a hierarchical beam searching protocol, with wide beams to narrow down the

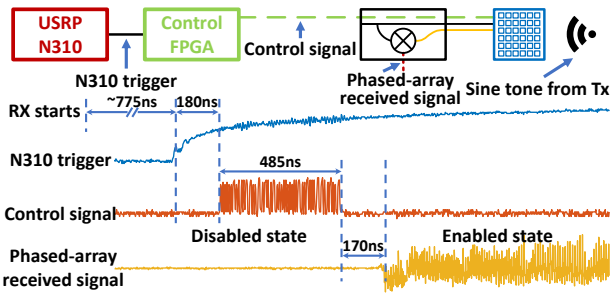

Figure 12: Measuring the control path latencies: from a trigger command to actual transmission.

search space, followed by fine tuning of narrow beams. To simplify the implementation of such beam sweeping sequences, we design a *real-time beam sweeping controller*, which is implemented in RTL on the control FPGA, and allows the PC host of the BPU to inject control commands into the PM at precisely designated timestamps. This controller is split in two clock domains to handle the different demands of the configuration functionality. The *configuration clock domain* receives UART commands from the PC host which define the sequence of configuration commands that the controller will send (*e.g.*, Tx/Rx mode, RF gain, beam index) and their timing. When the external trigger from the BPU is activated, a state machine cycles through the command parameters specified via UART, and selects the appropriate CRC value from a look-up table, to assemble the command data. When the state machine clock indicates that a new command should be sent, a trigger signal is sent to the *output clock domain*, which sends the command to the GPIO output using Biphase-L encoding. The controller can be configured from a Matlab interface that we developed. It allows us to *script a wide variety of beam sweeping configurations in software, without any time-consuming changes to the FPGA code itself.*

Control path latency. Several sources of hardware artifacts can make it difficult to align the BPU-generated data samples with the control signals. *(i) Initial delay* between the data samples and the phased array control signals. *(ii) Response time* of the phased array after receiving the control command, caused by the latency in reconfiguring internal amplifiers and phase shifters. *(iii) Clock drift* between the BPU and the control FPGA, which may accumulate over a long time and eventually break the synchronization.

To measure these artifacts, we connect a Tx PM and Rx PM to the same BPU (USRP N310). The Tx transmits a sine tone, while the Rx switches between two beams. The initial delay then corresponds to the sample index where the Rx signal envelop stabilizes, the components are shown in Fig. 12. The response time is the transition period between two beams where received signal level drops to zero. We found the clock drift by comparing the actual transition times to the expected times if the clocks were perfectly synchronized.

Table 1 summarizes the measurement results. We can compensate for the initial delay by shifting the Tx samples in the time domain as appropriate. However, since different beam configurations require setting different numbers of phase shifters inside the PM, initial delay and response times vary. These uncertainties increase the effective maximum latency, since there is an additional time period where beam switching may still be in progress. The total maximum latency is then $t_{lat} = \Delta t_{ini} + t_{res} + t_{data} \left|\rho_{C_1,C_2}\right|$,

Table 1: Control Latency Measurements

	Mean	Std Dev	Max
Intial Delay	1.61 μs	14.5 ns	1.62 μs
Array Response Time	87 ns	92 ns	192 ns
Clock Drift	7.7 ppm	0.05 ppm	7.8 ppm

where Δt_{ini} is the range of the initial delay, t_{res} is the maximum phased array reconfiguration time, and ρ_{C_1,C_2} is the clock drift. For beam sweeping across the maximum allowable number of beams (64) of a single phased array in 802.11ad, we need a maximum of only 412 ns to switch between two beams, even with a clock drift of up to 200 ppm. Given that 802.11ad [23] defines the time between beam sweep frames as 1 μs, our control timing is sufficiently precise to enable standard-compatible beam sweeping operations. Furthermore, for applications requiring many rapid measurements, we can continuously sweep 4,800 beams in one second using 802.11ad frames. Our measurements show that the variance of the latency metrics is negligible, so a fixed zero padding suffices and no additional calibration is needed when M^3 is replicated.

6 PHASED ARRAY CALIBRATION

Two types of 60 GHz phased array antennas have been integrated with the Qualcomm QCA6310 RFIC to form a monolithic PM. The first is NGFF595A-L-Ant, a rectangular non-uniform 32-element array used in commodity 802.11ad access points and laptops [35, 45, 56]. The second is a 6×6 uniform planar array (UPA), used in 802.11ad outdoor backhaul radios or indoor access points [3, 30]. In M^3, we use the 6×6 UPA, which can be more easily calibrated to generate desired beam patterns.

The set of beam patterns is stored on the PM as a codebook matrix which can be updated and reloaded using WMI commands. Each row in the codebook defines a *codebook entry*, a vector of beamforming weights to be applied to the antenna elements to form a specific beam. Each weight element comprises a 2-bit phase-shifter (for 0, $\frac{\pi}{2}$, π, and $\frac{3\pi}{2}$), and a 1-bit amplitude weight (enabling/disabling the antenna element). The codebook entry corresponding to a specific beam width and direction can be computed through well known theoretical models [8]. However, three hardware artifacts must be calibrated as input to the model.

Mapping between antenna elements and codebook. To determine the index mapping between antenna elements and beamforming weights, we send a test signal, shielding all but one antenna element with metal foil. Meanwhile, we disable all but one amplitude element in the codebook entry. The antenna that actually outputs signals corresponds to the non-zero codebook element. Fig. 13 shows the resulting element map for the 6×6 UPA. We found that 4 antennas in the corners are unused during beamforming.

Calibrating antenna spacing. Ideally, the spacing of adjacent antennas on a UPA is set to half of the wavelength to achieve a low sidelobe beam pattern [32]. However, the relationship between the antenna spacing and the wavelength will be different for different carrier frequencies. Moreover, since the antenna spacing of a 60 GHz antenna array is only several millimeters and the size of antenna element cannot be neglected at this scale, it is not practical to directly measure the antenna spacing.

Instead, to calibrate the exact spacing, we mount a Tx array on a programmable motor, and vary its azimuth/elevation angle relative

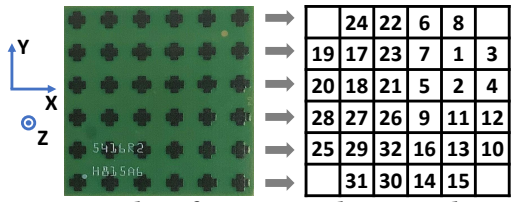

Figure 13: A snapshot of 6×6 UPA and mapping between antennas in the array and elements in the codebook entry.

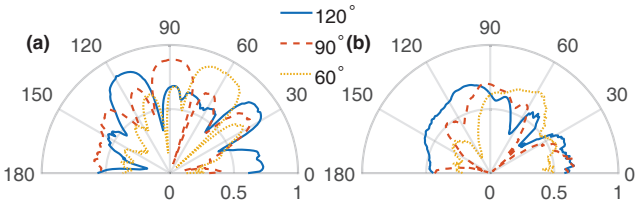

Figure 14: Beam patterns generated by a calibrated 6×6 UPA: (a) narrow beam. (b) wide beam.

to a static Rx. At each angle, we activate each antenna element of the Tx in turn, and collect the channel response at the Rx. The Rx is surrounded with absorbers to block all NLoS paths, so that the theoretical channel responses of the antenna elements of the Tx can be calculated according to the LoS direction between the Tx and Rx. Then, we use a simple optimization method to identify the exact antenna spacing. Specifically, for each possible spacing with certain range (*e.g.*, 0.3 to 0.7 wavelength), we use the theoretical model of the UPA [8] to generate channel responses of the antenna elements of the Tx for all LoS directions. Then, for each antenna element, the magnitude of the correlation between its theoretical and measured channel responses across all LoS directions is calculated. Finally, we sum the absolute correlations of all antenna elements of the Tx, and select the antenna spacing that yields the maximum sum. With this approach, we found that the antenna spacing of the 6×6 UPA is 0.58 wavelength at 60.48 GHz.

Calibrating phase offset between antenna elements. The third hardware artifact lies in the the phase offsets between different antenna elements, possibly due to different length of their transmission lines. To estimate the phase offset, we again use the channel responses of antenna elements of the Tx generated according to the theoretical model of the UPA [8], with the calibrated antenna spacing. For each antenna, we calculate the average phase difference between its theoretical channel responses and measured channel reponses across all LoS directions and use it as the phase offset of that antenna.

These three artifacts are mainly due to the hardware difference, so the calibration procedure only needs to be done once per antenna array. To show the effectiveness of the calibration, we create a set of wide beams and narrow beams by activating two columns and all columns of the 6×6 array, respectively. We use the Matlab phased array toolbox to design the codebook entries, so that the beams point to 60°, 90° and 120° in the azimuth plane. We then load the codebook into the PM and measure the beam pattern. The results (Fig. 14) show that the main lobe of the beam patterns match the desired directions. The horizontal half-power beam widths are approximately 20° and 45°, which match the theoretical model of UPA [21, Ch 2.1]. However, due to the discrete phase of beamforming

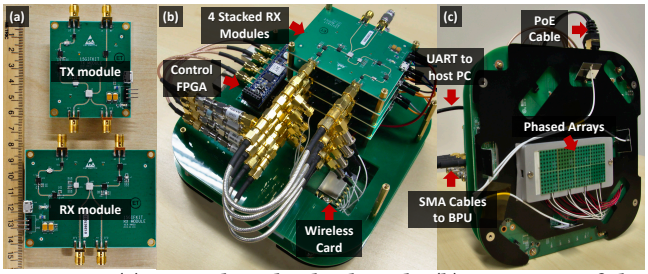

Figure 15: (a) TX and RX bridge boards. (b) Rear view of the 4 RF-chains node. (c) Front view of fully assembled M^3 node with 8 6×6 phased arrays on the same plane.

weight and limited amplitude control, the sidelobe patterns differ from the theoretical model. Narrower beams can be constructed through an array of phased arrays (Sec. 7).

7 MIMO MMWAVE ARCHITECTURE

A MIMO node may be assembled by extending the aforementioned single RF-chain version.

The QCA6335 BM has a single RF chain but 8 RF ports, which receives the same power/clock but separate control signals from the BM. In M^3, we replicate the aforementioned control/data path design, and use the QCA6335 RF ports to drive up to 8 phased arrays to form an *array of phased-arrays* (APA). The layout of this APA can be flexibly adjusted, by repositioning or reorienting the individual arrays, *e.g.*, to expand the field-of-view coverage.

Recall that in the split-IF architecture, the lower-frequency LO is generated on the BM chip while the higher frequency LO is generated on the PM. In the multi-array QCA6310 front-end, the 8 PMs generate their own high frequency LOs based on the same 7.5 GHz reference clock. Therefore, *all the 60 GHz PMs are naturally carrier synchronized and phase coherent.* To realize mmWave MIMO, M^3 can simply add multiple RF chains. More specifically, by connecting several bridge boards to the RF ports of the same QCA6335 BM, and attaching one PM to each bridge board, we can realize a multi-RF-chain, multi-phased-array, mmWave MIMO RF front-end.

Fig. 15 illustrates an M^3 mmWave MIMO node, with 4 RF chains and 8 phased arrays (4 active). In our current M^3 design, the Tx and Rx RF chains employ separate PMs, so up to 4 Tx and 4 Rx RF chains (4×4) can be supported. To realize 8×8 MIMO, it is possible to share the same PM between the Tx and Rx path and use control commands to switch between Tx/Rx mode.

8 SOFTWARE-DEFINED MMWAVE RADAR DESIGN

Radar requires that both the Tx and Rx RF front-ends be activated simultaneously, with carrier frequency and phase coherency. This can be realized in M^3, by forcing one phased array into Tx and one into Rx mode. Coherency is guaranteed as long as the two are driven by the same BM reference clock. By activating multiple (up to 4) pairs of Tx/Rx chains, we can implement a MIMO phased array radar, with fine angular resolution owing to the massive number of antenna elements. Since the phased array module does not include a self-interference cancellation design, the Tx antenna can leak signals to the co-located Rx. We found the leakage is acceptable

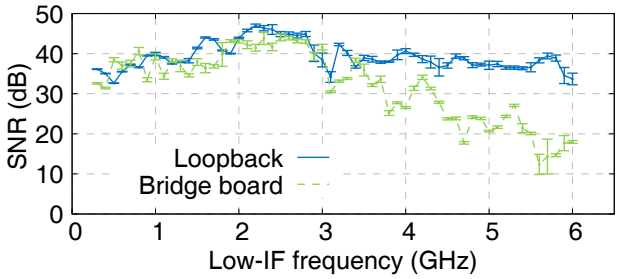

Figure 16: SNR over frequency for SDR loopback connection and through bridge board connection.

for short-range sensing, as long as the Tx and Rx planes are placed side-by-side (within the low-gain direction of each other), and a small metal blocker is placed in between.

Commercial mmWave radar hard-codes the signal processing flow on chip. In contrast, M^3 realizes a *software-defined mmWave radar*. The waveform generation, beam control, and post processing can all be customized on the BPU. We have implemented the classical FMCW radar ranging algorithm [32] on M^3, along with a simple angle estimation method, to locate multiple targets. Although recent systems explored mmWave sensing applications, *e.g.*, gesture/vital sign sensing [27, 36, 64], environment reconstruction [57] and imaging [70], they all use horn antennas or a small antenna array. We believe M^3's massive MIMO radar can bring the RF sensing resolution to the next level.

9 SYSTEM EVALUATION

We have quantified M^3's control path performance in Sec. 5. In this section, we evaluate its data path performance, and showcase its capabilities in real-time communication and radar sensing by integrating both paths.

Effectiveness of the bridge board design. The data path is mostly built from commodity 802.11ad NIC, except for the bridge board. We thus focus on how the bridge board affects the SNR and phase noise performance. All our SNR measurements are done by running reference OFDM code (adapted from [53]) on our BPU. Note that imperfections in the digital baseband (*e.g.*, quantization error, channel estimation miss-match) may also degrade SNR. Therefore, we mainly examine the relative SNR compared with baselines, rather than the absolute values.

We first isolate the PM, and only compare the SNR between two loopback settings: directly connecting a low-IF BPU's Tx port with its Rx; attaching the bridge board, and directly connecting the Tx bridge board output (*i.e.*, 15 GHz IF) with the Rx input. Here the Tx and Rx share the same LO to avoid the impact of CFO and phase noise, and use USRP N310 as BPU with a fixed gain configuration.

The result in Fig. 16 shows that, for low-IF frequency range between 300 MHz and 3 GHz, the SNR is similar with and without the bridge board, implying that *the bridge board itself incurs negligible SNR degradation across a* $\sim$ *3 GHz wide bandwidth.* Since many existing SDRs [15, 19, 28] support this low-IF frequency range (with much narrower bandwidth than 3 GHz), the result implies M^3 can interface with such SDRs with high performance. The bridge board SNR shows a reasonable level of linearity (*std.* 1.3 dB) within this frequency range, which is mostly due to hardware non-linearity in frequency domain and can be considered part of the frequency-selective channel fading. Beyond 3.8 GHz low-IF, the SNR gap increases and varies more significantly (5 to 15 dB). Such gap can potentially be compensated by setting different baseband gains for different frequency bins.

Table 2: Phase Noise Measurements.

	Phase noise (dBc/Hz)				
	@1kHz	@10kHz	@100kHz	@1MHz	@10MHz
Ref clk	-51	-53	-75	-101	-119
LO	-44	-51	-72	-98	-116
low-IF+IF	-43	-51	-72	-97	-115
End-to-end	-30	-32	-45	-73	-87

Table 3: SNR for Different Baseband Units

		Heterodyne			Homodyne	
BPU device		N310	B210	WARP	FMC150	FMCDAQ2
Bandwidth (MHz)		62.5	30.72	20	20	500
Loopback SNR (dB)		42.2	34.7	29.3	34.4	33.4
OTA SNR (dB)	64 FFT	18.3	17.2	10.1	NaN	19.0
	16 FFT	12.5	10.5	13.8	9.2	16.6

To show the frequency linearity of the bridge board design in homodyne mode, we use FMCDAQ2 as the BPU and transmit a wideband (1 GHz) OFDM signal. The SNR through bridge board is 28.8 dB with high stability across subcarriers (a small *std.* 0.9 dB). For comparison, the direct DAC-to-ADC SNR is 33.4 dB. The SNR gap is higher than the heterodyne case, which is likely due to the I/Q imbalance incurred by the HMC8191 mixer.

Phase noise. Phase noise originates from the instability of reference clocks, which is negligible for low-frequency carriers, but amplified by $20\log(N)$ for every $\times N$ frequency multiplier [29]. So it is a critical metric for mmWave radios. Recall M^3 employs two stages of frequency multipliers: $\times 2$ on the bridge board and the $\times 6$ on the RF front end. The phase noise will be worsen by 6 dB and 15.6 dB, respectively. To validate the impacts on the actual hardware, we use a spectrum analyzer to measure the phase noise at different stages of the transceiver path. Table 2 summarizes the results at 4 measurement points. *(i) Ref clk:* 7.56 GHz reference clock which provides the baseline. *(ii) LO:* 15.12 GHz LO generated by the bridge board, measured at the bridge board output (with a DC input), which shows only around 3 dB degradation. *(iii) Low-IF+IF:* The bridge board's IF output (at 12.72 GHz), when feeding a low-IF input (at 2.4 GHz) from a signal generator. It shows similarly low phase noise with the 15.12 GHz LO, which means *the bridge board design does not add additional phase noise. (iv) End-to-end:* The phase noise at the IF port of the RX bridge board, after an over-the-air transmission. It is much higher because it accumulates the noise from low-IF, IF, and RF at both Tx and Rx side. Note that the end-to-end phase noise is -45 dBc/Hz at 100 kHz frequency offset, 27 dB higher than that at the IF. Such phase noise will substantially degrade OFDM performance when the subcarrier spacing falls below 100 kHz. This is the reason why practical mmWave systems adopt much wider subcarriers (*e.g.*, 5 MHz for 802.11ad).

End-to-end SNR. We further evaluate the end-to-end performance of M^3 using different BPUs with different bandwidth and heterodyne/homodyne interfaces. In this experiment, the Tx is a single phased array with a 90° beam (Fig. 14) pointing to the Rx.

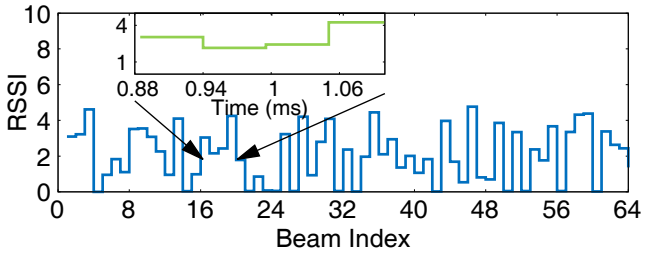

Figure 17: Real-time per-beam RSS Measurement.

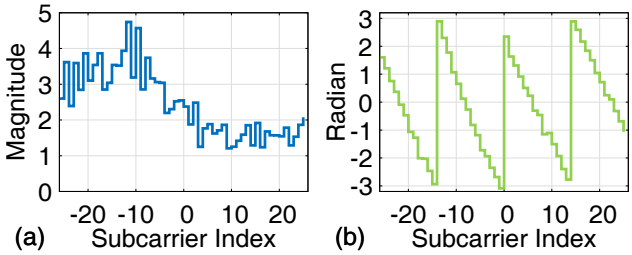

Figure 18: Example CSI measurement on one beam.

The Rx, 1 m away, is a single phased array with a quasi-omni beam (by only activating a single antenna element). We measure both over-the-air (OTA) and loopback SNR, the latter obtained by directly connecting the Tx and Rx port (or DAC and ADC) of the BPU. We oversample the OFDM symbols 2× to mitigate SNR degradation by the digital baseband. The OFDM FFT size is either 64 or 16, the latter corresponding to 4× wider subcarrier spacing.

From the results summarized in Table 3, we observe: *(i)* The OFDM FFT size has a significant performance impact. Typically, 64 FFT should lead to higher baseband SNR owing to finer grained channel estimation and equalization. However, it also translates into narrower subcarrier spacing, and thus lower RF SNR due to phase noise [34]. This tradeoff manifests itself differently for different BPU devices (*e.g.*, for the WARP and FMC150, 16 FFT leads to higher SNR instead). *(ii)* The performance of BPU itself does not determine the end-to-end performance, since the RF noise affects more than that of the BPU. For example, the B210 has comparable OTA SNR with N310, despite a lower loopback SNR. On the other hand, the FMCDAQ2 has the best OTA SNR even though the loopback SNR is relatively low.

We remark that the above measurement under-utilizes the beamforming gain at RX side, which is in quasi-omni mode (5 dB gain). The maximum beamforming gain of a 6×6 array for Rx is 14 dB according to the specifications of the PM, so the OTA SNR can be improved by 9 dB if Rx side beamforming is enabled. If all 8 arrays are used on both Tx and Rx, then the SNR of the link will further improve dramatically, by 18 dB for Tx and 9 dB for Rx, respectively. Following the Friis propagation model, the signal power degrades by around 36 dB when link distance increases from 1 m to around 63 m. This means the same level of SNR in Table 3 can be achieved even at 63 m link distance. Furthermore, baseband processing gain can further improve the SNR (*e.g.*, through DSSS modulation).

Integrating control and data path for real-time beam scanning. We now showcase the joint control and data operations in a real-time mmWave communication system. Here the BPU continuously transmits OFDM frames, while the control FPGA changes the Tx beam each frame. Fig. 17 plots the measured RSS for 64 Tx beams,

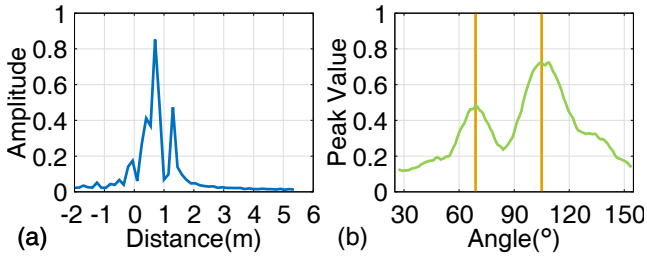

Figure 19: Performance of software defined FMCW radar.

with a fixed Rx beam. The zoomed-in subfigure shows consecutive switching between 4 beams, each with a frame duration of 60μs. Fig. 18 further shows a sample of CSI measurement (per-subcarrier magnitude and phase) from one beam. The frame duration is limited by the sampling rate of N310. With a multi-Gsps BPU, M^3 can achieve 802.11ad-compatible frame duration or beam switching frequency (Sec. 5). With such fine-grained per-beam channel measurement capabilities, M^3 will enable research into real-time mmWave systems.

Software defined radar showcase. To verify the software-defined radar design, we set up two example scenes. First, we use a radar with an FMCW waveform to determine the distance of the objects by finding the peaks in frequency domain. Two objects are placed at 90° relative to the antenna plane with 0.5 m distance difference, which results in the two spectrum peaks shown in Fig. 19(a). The two peaks are separated by around 0.5 m, which shows the feasibility of range detection. Second, two objects are placed at 70° and 110° angles relative to the antenna plane at the same distance from the phased array. To estimate the relative angles of the objects, the Rx phased array scans 128 beams evenly partitioning the polar angle on X-Z plane, while the Tx illuminates the scene using a quasi-omni beam and an FMCW waveform. By taking the peak value of the frequency spectrum of each angle and apply a moving average with a 20 points window, we obtain the reflection response of different angle. The peaks are at 69° and 105°, approximately the direction of the two targets. This beam scanning approach represents a very rudimentary method of angle estimation. The angular resolution can be improved by leveraging the vast literature of phased array radars, *e.g.*, eigen-space methods such as MUSIC [41], joint AoD/AoA estimation [57], *etc.*.

10 CASE STUDIES

In this section, we conduct two case studies to demonstrate the use of M^3 in mmWave MIMO networking and sensing.

10.1 mmWave MIMO Hybrid Beamforming

Prior research in mmWave MIMO either used statistical channel models [4, 5, 16] or synthesized MIMO by moving a single phased array to different locations [17, 18]. Here we conduct the first mmWave MIMO link measurement using M^3 to understand the various design choices that may affect its performance.

10.1.1 SU-MIMO Hybrid Beamforming. MmWave MIMO transmitters can send multiple streams of data either to a single user (SU-MIMO) or to multiple users (MU-MIMO). MmWave MIMO generally operates in two stages to realize *hybrid beamforming*

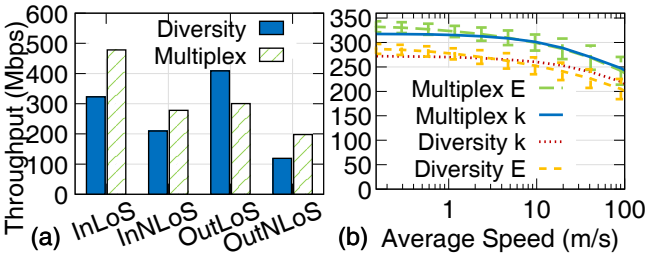

Figure 20: SU-MIMO throughput: (a) Scenario comparison using k-best, $k = 50$, (b) Throughput after considering beam training overhead (k: k-best; E: Evolutionary algorithm).

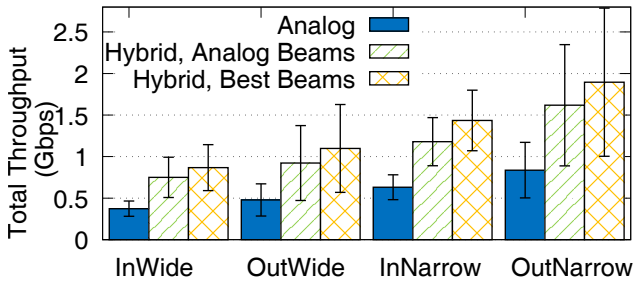

Figure 21: MU-MIMO capacity in different scenarios: In/Out: Indoor/Outdoor; Wide/Narrow: different beamwidths.

[47, 61]: *(i)* Select the best beam indices for each phased array (analog beamforming); *(ii)* Measure the channel with the selected beams to achieve either diversity or multiplexing gain (digital beamforming).

Whereas the second stage is similar to legacy MIMO in 802.11ac, the IEEE 802.11ay task group has proposed a number of possible algorithms for the first stage [4, 16]: *(i) Exhaustive beamforming* measures the available throughput for all available combinations of phased array configurations; *(ii) k-best algorithm* ranks the configurations based on a coarse sector-level beam sweep and tries the top k configurations. *(iii) Evolutionary algorithm* randomly selects configurations to try using a probability derived from the sector sweep, and stops early when the link metric exceeds a certain threshold.

We evaluated 2×2 mmWave MIMO performance in a multipath-rich conference room and a multipath-poor outdoor environment, with a metal box 2 m away from the Tx in each, to create NLoS blockage effect. The Rx moved past the obstacle and samples the channel at 9 locations, with 4 experiencing blockage. We first collected sector sweeps for each phased array using N310 as BPU. Next, we tried all possible combinations of phased array sector selections, transmitting OFDM symbols and training preambles simultaneously from the Tx phased arrays. Using the received data, we estimated the SNR achievable via diversity using the beamforming MMSE SNR estimator [10], as well as the SNR of each multiplexed channel, and found the corresponding channel capacities for the available bandwidth. We then ran the exhaustive algorithm and two heuristic algorithms on the data set.

Fig. 20(a) compares the capacity for diversity vs. multiplexing in these scenarios, using the same beam training method (k-best). We observe that the multiplexing capacity is ∼ 50% higher than diversity capacity in the indoor environment, but ∼ 30% lower in outdoor LoS scenarios. However, in outdoor NLoS scenarios, multiplexing achieves higher throughput. Overall, multiplexing gain is most prominent in multipath-rich and NLoS environment. This suggests that *it would be beneficial to adaptively select the digital beamforming mode based on the channel context.* Such context information may be provided through alternative channels such as an optical sensor [20]. The capacity is lower than the theoretical maximum in 802.11ad because the BPU has only 100 MHz of bandwidth, and the corresponding maximum throughput is 325 Mbps for a single link with this bandwidth.

Fig. 20(b) shows the average throughput among all scenarios after accounting for beam training overhead. We emulate the variation in re-training time due to different mobility levels by varying the temporal separation between measurements, since these algorithms perform complete re-training whenever the channel changes. The evolutionary algorithm performs better in lower mobility. The algorithms have similar performance under high mobility, but k-best may be preferred since the evolutionary algorithm is stochastic and less predictable.

10.1.2 MU-MIMO Hybrid Beamforming. For MU-MIMO testing, we collected channel data using two codebooks, corresponding to narrow and wide beams (Sec. 6). Due to limited M^3 radios available, we created a 2×2 MU-MIMO setup by using the same 2 RF chain Tx but moving the 2 RF chain Rx to the same locations as in the SU-MIMO experiment. With the channel data, we determined the total capacity of the channel for three beamforming strategies: *(i)* analog beamforming alone, using the best capacity among all beam selections, *(ii)* mmWave MIMO using zero-forcing with the globally optimal beam selections, *(iii)* mmWave MIMO as in (ii), but using the beams selected in (i). We evaluated 50 random sets of Rx locations for each test scenario.

From the results (Fig. 21), we observe that, unlike in the SU-MIMO experiments, MU-MIMO achieves higher capacity in the outdoor environment because multipath introduces additional interference for widely-spaced Rx. Zero-forcing increases the achievable indoor capacity by 2.2× and outdoor capacity by 2.3×, which suggests that *there are large performance gains available from hybrid beamforming in MU-MIMO scenarios, rather than analog beamforming alone.* These capacity gains are approximately equal for both codebooks (narrow and wide), which suggests that *interference significantly impairs capacity even with a more directional codebook.* Moreover, we note that using MU-MIMO with the optimal beams from analog beamforming achieves 1.9× improvement over analog beamforming alone, which means that *we can harvest MU-MIMO capacity gains by simply using zero-forcing along with a beam selection algorithm such as [17, 24], without requiring multiple measurement rounds as proposed in existing approaches [18].*

10.2 Multi-Array mmWave Radar

To showcase the spatial resolution advantages of the M^3 radar with multiple phased arrays, we place two metal boxes (5 × 10 cm cross section) 8 cm apart, and place them 1 m in front of the radar. An OFDM signal is transmitted by a quasi-omni Tx to illuminate the scene. We vary the number of Rx phased arrays and generate corresponding AoA pseudo-spectrum using the MUSIC algorithm [12]. As shown in Fig. 22, the two objects' angles cannot be separated at

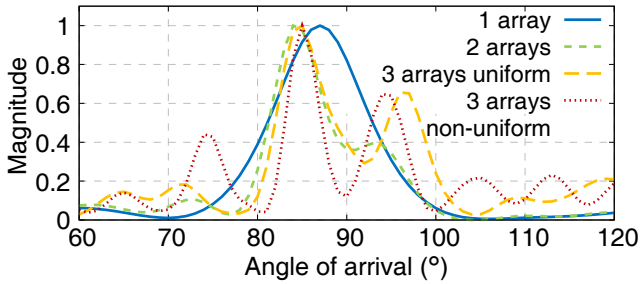

Figure 22: Power spectrum of different angles of arrivals.

all (*i.e.*, a single AoA peak) with only one phased array and hardly with 2 arrays. With 3 phased arrays uniformly placed adjacently (*i.e.*, half wavelength separation between elements on the edge), the angular resolution is sufficiently high to clearly differentiate the two targets. In addition, we investigate a non-uniform array layout, where the 1st and 2nd array are adjacent and the 3rd array is L (the width of one phased array) away from the 2nd one. Since this layout expands the effective antenna aperture size, the angular resolution is improved accordingly, but at the cost of prominent side lobes, which can be precluded based on prior knowledge (*e.g.*, knowing only two objects exist). In a nutshell, M^3 *can improve the angular resolution of the Radar practically by coherently combining multiple phased arrays, and through non-uniform array layout.* In addition, the angular resolution can be further enhanced by exploiting all the 4 Tx and 4 Rx arrays. In effect, by separating adjacent Rx arrays by L and Tx arrays by $4L$, we can effectively improve the antenna aperture to $16L$, and the Radar resolution would be equivalent to a phased array with $96 \times 11 = 1056$ elements [26, Ch 2.3]!

11 RELATED WORK

Early experimental work on mmWave focused on channel measurements using specialized sounding hardware [7, 63], with a rotating horn antenna to capture the spatial channel profile (AoA, AoD, *etc.*). A more flexible approach, using a mmWave frequency converter [51] attached to a programmable BPU, was adopted in early 60 GHz networking and sensing experiments [40, 43, 46, 55, 67, 68]. The channel response is emulated through an angle-wise multiplication between an ideal phased array's gain pattern (computed based on its codebook), and the channel's AoA/AoD profile measured between the Tx and Rx's horn antennas. E-Mi [57] adopted time-lapse emulation, creating a virtual phased array by placing an omni-directional mmWave antenna at different locations. Both approaches assume the channel is stationary during the antenna rotation/movement, which is not applicable when the devices or nearby objects are moving.

OpenMili [65] represents the first 60 GHz software radio with a programmable phased array, fabricated using discrete RF components. Despite its flexibility, OpenMili has a few limitations: relatively high cost (>$15K per node); small (4-element) phased array with low-resolution (only 5 beam patterns); no support for mmWave MIMO—a key feature in next-generation mmWave standards [22, 37, 69]. MiRa [2] is a 24 GHz RF front-end built from discrete components (power amplifier, mixer, LO), acting as a daughterboard for USRPs. It has an 8-element phased array, fabricated using a PCB and 18-24 GHz phase-shifter ICs. As in OpenMili, the on-board phased array tends to incur high fabrication cost and cannot easily scale. Moreover, it highly depends on the availability of phase-shifter ICs—which are only available below 24 GHz (the reason why the 60 GHz OpenMili has to use delay lines as phase shifters instead). X60 [39] is the first 60 GHz software radio with an integrated 12-element phased array. However, it only offers 25 prescribed beamforming vectors and does not allow reconfiguring the codebook. Its high cost also hinders its wide adoption.

The latest versions of commodity 802.11ad radios are exposing certain PHY layer parameters to the host drivers, allowing reconfiguration of codebook, beam index, beacon interval, *etc.*, at a coarse time scale [45, 52, 56]. Palacios *et al.* [35] proposed a method to reconstruct the CSI by scanning the RSS of different beams on a Talon AD7200 802.11ad radio. Wang *et al.* [52] adopted the Airfide 802.11ad device [3] as a partially programmable mmWave radio to investigate multi-array multi-beam management. These systems do not support real-time beam control, customized baseband waveforms, and MIMO operations.

Recent demands in automotive sensing and wireless health have revived the design of consumer-grade mobile mmWave radar. Examples include TI's 76-81 GHz MIMO radar with 2Tx, 4Rx [48], and Vayyar's 62-69 GHz radar with 20Tx and 20Rx antennas[50]. To reduce cost, such radar devices adopt predefined waveforms (*e.g.*, FMCW) to avoid digital baseband processing. In contrast, M^3 can act as a more flexible software-defined radar with arbitrary I/Q waveform generation and processing. Its massive phased array and hybrid beamforming architecture can potentially enable orders of magnitude higher spatial resolution.

Despite the well established potential of mmWave massive MIMO in theory [11, 47, 49], experimental validation is limited due to lack of a flexible and affordable platform. Recent mmWave MIMO systems [17, 18] studied the problems of efficient beam training and MU-MIMO user selection, and created a virtual MIMO by moving a single X60 radio as in [57]. Such approaches cannot strictly ensure channel coherency across different measurement locations/timestamps, and are not applicable to mobile and dynamic environment.

12 CONCLUSION

We have demonstrated the feasibility of reengineering a commodity 802.11ad mmWave radio into a low-cost massive MIMO software radio, *i.e.*, M^3. Our design choices in M^3 focus on optimizing the radio performance, while keeping its architecture simple and scalable. Our experiments have verified the effectiveness of the M^3 design. Considering its flexibility, performance, and affordability, we expect M^3 to change the landscape of research in mmWave networking and sensing. Since commodity mmWave radios tend to share similar architectures, our design can potentially be applied to create mmWave software radios on other frequency bands (*e.g.*, based on 5G NR radios).

ACKNOWLEDGMENTS

We appreciate the insightful comments and feedback from the anonymous reviewers and shepherd. The work reported in this paper is supported in part by the NSF under Grant CNS-1506657, CNS-1617321, CNS-1854472, CNS-1952942, CNS-1925767.

REFERENCES

[1] 802.11ay IEEE 802 LAN/MAN Standards Committee. 2018. Status of Project IEEE 802.11ay. (2018). COSTActionCA1510

[2] Omid Abari, Haitham Hassanieh, Michael Rodreguiz, and Dina Katabi. 2016. Poster: A Millimeter Wave Software Defined Radio Platform with Phased Arrays. In *Proc. of ACM MobiCom (Poster/Demo Sessions)*.

[3] Airfide Networks. 2018. Airfide–A 5G Company. (2018). http://airfidenet.com

[4] Alireza Tarighat, Payam Torab, Brima Ibrahim, Vipin Aggarwal, and Vinko Erceg. 2015. A Framework for MIMO Operation over mmWave Links. (2015). IEEE802.11-15/0334r0

[5] A. Alkhateeb, O. El Ayach, G. Leus, and R. W. Heath. 2014. Channel Estimation and Hybrid Precoding for Millimeter Wave Cellular Systems. *IEEE Journal of Selected Topics in Signal Processing* 8, 5 (2014).

[6] Analog Devices. 2019. HMC8191 6 GHz to 26.5 GHz Wideband I/Q Mixer. (2019). https://www.analog.com/media/en/technical-documentation/data-sheets/hmc8191.pdf

[7] C.R. Anderson and T.S. Rappaport. 2004. In-Building Wideband Partition Loss Measurements at 2.5 and 60 GHz. *IEEE Transactions on Wireless Communications* 3, 3 (2004).

[8] Constantine A Balanis. 2016. *Antenna Theory: Analysis and Design*. John Wiley & Sons.

[9] M. Boers, B. Afshar, I. Vassiliou, S. Sarkar, S. T. Nicolson, E. Adabi, B. G. Perumana, T. Chalvatzis, S. Kavvadias, P. Sen, W. L. Chan, A. H. Yu, A. Parsa, M. Nariman, S. Yoon, A. G. Besoli, C. A. Kyriazidou, G. Zochios, J. A. Castaneda, T. Sowlati, M. Rofougaran, and A. Rofougaran. 2014. A 16TX/16RX 60 GHz 802.11ad Chipset With Single Coaxial Interface and Polarization Diversity. *IEEE Journal of Solid-State Circuits* 49, 12 (2014).

[10] S. Boumard. 2003. Novel noise variance and SNR estimation algorithm for wireless MIMO OFDM systems. In *GLOBECOM '03. IEEE Global Telecommunications Conference (IEEE Cat. No.03CH37489)*, Vol. 3. 1330–1334 vol.3. https://doi.org/10.1109/GLOCOM.2003.1258454

[11] S. A. Busari, K. M. S. Huq, S. Mumtaz, L. Dai, and J. Rodriguez. 2018. Millimeter-Wave Massive MIMO Communication for Future Wireless Systems: A Survey. *IEEE Communications Surveys Tutorials* 20, 2 (2018).

[12] Jack Capon. 1969. High-resolution frequency-wavenumber spectrum analysis. *Proc. IEEE* 57, 8 (1969), 1408–1418.

[13] E. Cohen, M. Ruberto, M. Cohen, O. Degani, S. Ravid, and D. Ritter. 2013. A CMOS Bidirectional 32-Element Phased-Array Transceiver at 60 GHz With LTCC Antenna. *IEEE Transactions on Microwave Theory and Techniques* 61, 3 (2013).

[14] S. Emami, R. F. Wiser, E. Ali, M. G. Forbes, M. Q. Gordon, X. Guan, S. Lo, P. T. McElwee, J. Parker, J. R. Tani, J. M. Gilbert, and C. H. Doan. 2011. A 60GHz CMOS Phased-Array Transceiver Pair for Multi-Gb/s Wireless Communications. In *IEEE International Solid-State Circuits Conference*.

[15] Ettus Research LLC. [n. d.]. Universal Software Radio Peripheral (USRP). ([n. d.]). http://www.ettus.com/

[16] Felix Felhauer, Dana Ciochina, Thomas Handte, Nabil Loghin, and Fares Zenaidi. 2016. Low Complexity Beamtraining for Hybrid MIMO. (2016).

[17] Yasaman Ghasempour, Muhammad K. Haider, Carlos Cordeiro, Dimitrios Koutsonikolas, and Edward Knightly. 2018. Multi-Stream Beam-Training for mmWave MIMO Networks. In *Proceedings of ACM MobiCom*.

[18] Y. Ghasempour, M. K. Haider, and E. W. Knightly. 2018. Decoupling Beam Steering and User Selection for MU-MIMO 60-GHz WLANs. *IEEE/ACM Transactions on Networking* 26, 5 (2018).

[19] Great Scott Gadgets. [n. d.]. HackRF One. ([n. d.]). https://greatscottgadgets.com/hackrf/one/

[20] Muhammad Kumail Haider, Yasaman Ghasempour, Dimitrios Koutsonikolas, and Edward W. Knightly. 2018. LiSteer: MmWave Beam Acquisition and Steering by Tracking Indicator LEDs on Wireless APs. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking (MobiCom)*.

[21] Robert C Hansen. 2009. *Phased array antennas*. John Wiley & Sons, Chapter 2.1, 7–17.

[22] Y. Huang, Y. Li, H. Ren, J. Lu, and W. Zhang. 2018. Multi-Panel MIMO in 5G. *IEEE Communications Magazine* 56, 3 (2018).

[23] IEEE Standard. 2012. 802.11™: Amendment 3: Enhancements for Very High Throughput in the 60 GHz Band. (2012).

[24] Suraj Jog, Jiaming Wang, Junfeng Guan, Thomas Moon, Haitham Hassanieh, and Romit Roy Choudhury. 2019. Many-to-Many Beam Alignment in Millimeter Wave Networks. In *USENIX Symposium on Networked Systems Design and Implementation (NSDI)*.

[25] Vladimir Kondratiev. 2018. wil6210 Linux Driver for Qualcomm 60 GHz NIC. (2018). https://wireless.wiki.kernel.org/en/users/drivers/wil6210

[26] Jian Li and Petre Stoica. 2009. *MIMO Radar Signal Processing*. Wiley Online Library, Chapter 2.3, 73–77.

[27] Jaime Lien, Nicholas Gillian, M. Emre Karagozler, Patrick Amihood, Carsten Schwesig, Erik Olson, Hakim Raja, and Ivan Poupyrev. 2016. Soli: Ubiquitous Gesture Sensing with Millimeter Wave Radar. *ACM Trans. on Graphics* 35, 4 (2016).

[28] Mango Communications. 2016. Wireless Open-Access Research Platform (WARP). (2016). http://warpproject.org/trac/

[29] Microsemi. [n. d.]. A Tutorial on Phase Noise. ([n. d.]). https://www.vectron.com/products/appnotes/phase.htm

[30] Microtik. 2017. 60 GHz Access Point with 3 Titled Phased-Array. (2017). https://mikrotik.com/product/wap_60gx3_ap#fndtn-specifications

[31] Mini-Circuits. 2018. BFCN-7331+ Typical Performance Data. (2018). https://www.minicircuits.com/pages/s-params/BFCN-7331+_VIEW.pdf

[32] Jeffrey A. Nanzer. 2013. *Microwave and Millimeter-Wave Remote Sensing for Security Applications*. Artech House.

[33] National Instruments. 2017. mmWave Transceiver System. (2017). http://www.ni.com/sdr/mmwave/

[34] Richard van Nee and Ramjee Prasad. 2000. *OFDM for wireless multimedia communications*. Artech House, Boston.

[35] Joan Palacios, Daniel Steinmetzer, Adrian Loch, Matthias Hollick, and Joerg Widmer. 2018. Adaptive Codebook Optimization for Beam Training on Off-the-Shelf IEEE 802.11ad Devices. In *Proceedings of ACM MobiCom*.

[36] Avishek Patra, Philipp Geuer, Andrea Munari, and Petri Mähönen. 2018. mmWave Radar Based Gesture Recognition: Development and Evaluation of a Low-Power, Low-Complexity System. In *Proceedings of the ACM Workshop on Millimeter Wave Networks and Sensing Systems (mmNets)*.

[37] Mattia Rebato, Michele Polese, and Michele Zorzi. 2018. Multi-Sector and Multi-Panel Performance in 5G mmWave Cellular Networks. *CoRR* abs/1808.04905 (2018). http://arxiv.org/abs/1808.04905

[38] Ali Sadri. 2013. mmWave Technology Evolution: From WiGig to 5G Small Cells. In *Invited Talk, IEEE International Conference on Communications (ICC)*.

[39] Swetank Kumar Saha, Yasaman Ghasempour, Muhammad Kumail Haider, Tariq Siddiqui, Paulo De Melo, Neerad Somanchi, Luke Zakrajsek, Arjun Singh, Owen Torres, Daniel Uvaydov, Josep Miquel Jornet, Edward Knightly, Dimitrios Koutsonikolas, Dimitris Pados, and Zhi Sun. 2017. X60: A Programmable Testbed for Wideband 60 GHz WLANs with Phased Arrays. In *Proceedings of the 11th Workshop on Wireless Network Testbeds, Experimental Evaluation and Characterization (WiNTECH)*.

[40] S. Sanjib, V. Venkateswaran, X. Zhang, and P. Ramanathan. 2015. 60 GHz Indoor Networking through Flexible Beams: A Link-Level Profiling. In *ACM SIGMETRICS*.

[41] Ralph Schmidt. 1986. Multiple Emitter Location and Signal Parameter Estimation. *IEEE transactions on antennas and propagation* 34, 3 (1986), 276–280.

[42] T. Sowlati, S. Sarkar, B. G. Perumana, W. L. Chan, A. Papio Toda, B. Afshar, M. Boers, D. Shin, T. R. Mercer, W. Chen, A. Grau Besoli, S. Yoon, S. Kyriazidou, P. Yang, V. Aggarwal, N. Vakilian, D. Rozenblit, M. Kahrizi, J. Zhang, A. Wang, P. Sen, D. Murphy, A. Sajjadi, A. Mehrabani, E. Kornaros, K. Low, K. Kimura, V. Roussel, H. Xie, and V. Kodavati. 2018. A 60-GHz 144-Element Phased-Array Transceiver for Backhaul Application. *IEEE Journal of Solid-State Circuits* 53, 12 (2018).

[43] D. Steinmetzer, J. Chen, J. Classen, E. Knightly, and M. Hollick. 2015. Eavesdropping with periscopes: Experimental security analysis of highly directional millimeter waves. In *IEEE Conference on Communications and Network Security (CNS)*. 335–343.

[44] Daniel Steinmetzer, Daniel Wegemer, and Matthias Hollick. 2017. Talon Tools: The Framework for Practical IEEE 802.11ad Research. (2017). https://seemoo.de/talon-tools

[45] Sanjib Sur, Ioannis Pefkianakis, Xinyu Zhang, and Kyu-Han Kim. 2017. WiFi-Assisted 60 GHz Networks. In *Proc. of ACM MobiCom*.

[46] Sanjib Sur, Xinyu Zhang, Parmesh Ramanathan, and Ranveer Chandra. 2016. BeamSpy: Enabling Robust 60 GHz Links Under Blockage. In *USENIX NSDI*.

[47] A. L. Swindlehurst, E. Ayanoglu, P. Heydari, and F. Capolino. 2014. Millimeter-Wave Massive MIMO: the Next Wireless Revolution? *IEEE Communications Magazine* 52, 9 (2014).

[48] Texas Instruments. 2018. Single-chip 76-GHz to 81-GHz automotive radar sensor integrating DSP and MCU. (2018). http://www.ti.com/product/AWR1642

[49] Eric Torkildson, Bharath Ananthasubramaniam, Upamanyu Madhow, and Mark Rodwell. 2006. Millimeter-wave MIMO: Wireless Links at Optical Speeds. In *IEEE Allerton Conference*.

[50] Vayyar Inc. 2019. VTRIG-74: 3D Millimeter Wave Imaging Kit. (2019). https://www.minicircuits.com/WebStore/vtrig_74.html

[51] VubIQ Inc. 2013. V60WGD03 60 GHz Waveguide Development System. (2013). http://vubiq.com/v60wgd03/

[52] Song Wang, Jingqi Huang, Xinyu Zhang, Hyoil Kim, and Sujit Dey. 2020. X-Array: Approximating Omnidirectional Millimeter-Wave Coverage Using an Array of Phased Arrays. In *Proc. of ACM MobiCom*.

[53] WARP Project. 2019. OFDM Reference Design. (2019). https://warpproject.org/trac/wiki/WARPLab/Examples/OFDM

[54] WARP: Wireless Open Access Research Platform. [n. d.]. ([n. d.]). http://warpproject.org/trac

[55] Teng Wei and Xinyu Zhang. 2015. mTrack: High-Precision Passive Tracking Using Millimeter Wave Radios. In *ACM MobiCom*.

[56] Teng Wei and Xinyu Zhang. 2017. Pose Information Assisted 60 GHz Networks: Towards Seamless Coverage and Mobility Support. In *ACM MobiCom*.

[57] Teng Wei, Anfu Zhou, and Xinyu Zhang. 2017. Facilitating Robust 60 GHz Network Deployment By Sensing Ambient Reflectors. In *USENIX NSDI*.

[58] Wilocity Ltd. 2014. Single transmission line for connecting radio frequency modules in an electronic device. (2014). https://patents.google.com/patent/US8670322

[59] Klaus Witrisal and Carles Antón-Haro. 2018. Whitepaper on New Localization Methods for 5G Wireless Systems and the Internet-of-Things. (2018). COSTActionCA1510

[60] Z. Xiao, T. He, P. Xia, and X. Xia. 2016. Hierarchical Codebook Design for Beamforming Training in Millimeter-Wave Communication. *IEEE Transactions on Wireless Communications* 15, 5 (May 2016), 3380–3392. https://doi.org/10.1109/TWC.2016.2520930

[61] Xiufeng Xie, Eugene Chai, Xinyu Zhang, Karthikeyan Sundaresan, Amir Khojastepour, and Sampath Rangarajan. 2015. Hekaton: Efficient and Practical Large-Scale MIMO. In *Proceedings of the ACM Annual International Conference on Mobile Computing and Networking (MobiCom)*.

[62] Xilinx. 2018. UltraScale+ RFSoC. (2018). https://www.xilinx.com/products/silicon-devices/soc/rfsoc.html

[63] Hao Xu, V. Kukshya, and T.S. Rappaport. 2002. Spatial and Temporal Characteristics of 60-GHz Indoor Channels. *IEEE Journal on Selected Areas in Communications* 20, 3 (2002).

[64] Zhicheng Yang, Parth H. Pathak, Yunze Zeng, Xixi Liran, and Prasant Mohapatra. 2017. Vital Sign and Sleep Monitoring Using Millimeter Wave. *ACM Trans. on Senor Networks* 13, 2 (2017).

[65] Jialiang Zhang, Xinyu Zhang, Pushkar Kulkarni, and Parameswaran Ramanathan. 2016. OpenMili: A 60 GHz Software Radio Platform With a Reconfigurable Phased-Array Antenna. In *ACM MobiCom*.

[66] Xinyu Zhang. 2017. WiMi: A Software Radio for Millimeter-Wave Networking and Sensing. (2017). http://xyzhang.ucsd.edu/wimi

[67] A. Zhou, T. Wei, X. Zhang, and H. Ma. 2018. FastND: Accelerating Directional Neighbor Discovery for 60-GHz Millimeter-Wave Wireless Networks. *IEEE/ACM Transactions on Networking* 26, 5 (2018).

[68] A. Zhou, X. Zhang, and H. Ma. 2017. Beam-forecast: Facilitating Mobile 60 GHz Networks via Model-Driven Beam Steering. In *Proc. of IEEE INFOCOm*.

[69] P. Zhou, K. Cheng, X. Han, X. Fang, Y. Fang, R. He, Y. Long, and Y. Liu. 2018. IEEE 802.11ay-Based mmWave WLANs: Design Challenges and Solutions. *IEEE Communications Surveys Tutorials* 20, 3 (2018).

[70] Yanzi Zhu, Yuanshun Yao, Ben Y Zhao, and Haitao Zheng. 2017. Object recognition and navigation using a single networking device. In *Proceedings of ACM MobiSys*.