

Design of a multi-function high-speed digital baseband data acquisition system

Xin Pei^{1,2,3,4}, Jian Li^{1,3,4}, Na Wang^{1,3}, Toktonur Ergesh^{1,3,4}, Xue-Feng Duan^{1,3,4}, Jun Ma^{1,3,4} and Mao-Zheng Chen^{1,3,4}

¹ Xinjiang Astronomical Observatory, Chinese Academy of Sciences, Urumqi 830011, China; peixin@xao.ac.cn, na.wang@xao.ac.cn

² University of Chinese Academy of Sciences, Beijing 100049, China

³ Key Laboratory of Radio Astronomy, Chinese Academy of Sciences, Nanjing 210008, China

⁴ Xinjiang Key Laboratory of Microwave Technology, Urumqi 830011, China

Received 2021 April 22; accepted 2021 June 13

Abstract A multi-function digital baseband data acquisition system is designed for the sampling, distribution and recording of wide-band multi-channel astronomical signals. The system hires a SNAP2 board as a digital baseband converter to digitize, channelize and packetize the received signal. It can be configured dynamically from a single channel to eight channels with a maximum bandwidth of 4096 MHz. Eight parallel HASHPIPE instances run on four servers, each carrying two NVMe SSD cards, achieving a total continuous write rate of 8 GB s⁻¹. Data are recorded in the standard VDIF file format. The system is deployed on a 25-meter radio telescope to verify its functionality based on pulsar observations. Our results indicate that during the 30-minute observation period, the system achieved zero data loss at a data recording rate of 1 GB s⁻¹ on a single server. The system will serve as a verification platform for testing the functions of the QTT (QiTai radio Telescope) digital backend system. In addition, it can be used as a baseband/VLBI (Very Long Baseline Interferometry) recorder or D-F-engine of correlator/beamformer as well.

Key words: instrumentation: miscellaneous — techniques: miscellaneous — site testing

1 INTRODUCTION

In radio astronomy, the radiation of celestial objects detected by a radio telescope is measured through the electric field intensity. Baseband data are raw voltage data that carry the original information from the observing sources, which has not yet been manipulated by the sum of squares or integration. These data can be used for multi-scientific signal analysis in the time or frequency domain such as Very Long Baseline Interferometry (VLBI), pulsar coherent de-dispersion, Search for ExtraTerrestrial Intelligence (SETI), transient search, etc.

Since the first successful VLBI observation in 1967, the form of recording baseband data has been adopted by many VLBI data recorders. Mark I, Mark II, and Mark III all used magnetic tape media for data recording with maximum recording rates of 0.72, 4, and 224 Mb s⁻¹, respectively (Cornwell & Perley 1991). By 2006, all tape drives had been replaced by hard drives, and the data recording rate has improved significantly. Mark5A had a maximum data rate of 1024 Mbps for writing 16 disks simultaneously. Mark5C increased the data recording

speed to 4 Gbps in 2013. The Mark 6 has been planning to increase the data recording rate from the original 16 Gbps to 64 Gbps on four parallel systems (Whitney & Lapsley 2012).

So far, the highest data recording rate in astronomy is achieved by the SETI project team on the Green Bank Telescope (GBT). It uses 32 storage nodes, each with a rate of 6 Gb s⁻¹, reaching the total data recording rate of 24 GB s⁻¹ (MacMahon et al. 2018).

The SETI data acquisition system realizes a high data recording rate by simultaneously writing multiple storage nodes, which increases the complexity of the system and reduces the reliability. Compared with the ever-increasing bandwidth of ultra-wideband and multi-beam observation systems, the data recording rate of hard disk drives (HDDs) is very limited. With the development of electronic technology, solid-state drives (SSDs) with high read and write speeds and terabytes storage space have become available. Along with the cost of SSDs falling dramatically, we have the opportunity to explore

the possibility of using SSD cards for fast storage in astronomy.

To obtain higher observation sensitivity and field of view, more and more UWB (ultra-wideband) receivers and multi-beam receivers are planned and adopted by large radio telescopes (Ma et al. 2021), which poses a great challenge to signal acquisition and storage of raw voltage data. The QTT (QiTai radio Telescope) (Wang 2014) is planned to be built in Qitai county of Xinjiang, China. The telescope will be equipped with a fully steerable 110-meter aperture antenna capable of observation in the frequency band between 270 MHz and 115 GHz (Ma et al. 2019). The bandwidth of UWB receivers reaches 4 GHz, and the number of Phase Array Feed (PAF) receiver channels exceeds 100. Pulsar coherent de-dispersion, VLBI, time-domain signal analysis (transient and SETI), PAF beamforming and other signal processing require raw voltage data. These are the key factors that must be considered when designing the QTT data acquisition, distribution and storage system.

Aiming at the application requirements of the QTT digital backend system, this paper developed a multi-mode baseband data conversion and distribution system based on the SNAP2 platform, and implemented a high-throughput data recording system through NVMe SSD storage cards. The system uses the standard VDIF format for data output and file storage to adapt to different software and hardware.

2 SYSTEM DESIGN

2.1 Hardware Architecture

The system consists of two parts: a digital baseband data converter and data recorder. A block diagram of the system hardware architecture is shown in Figure 1. The input radio frequency signals are digitized by SNAP2¹ board and converted to baseband data, and then transmit to four servers for data recording through a 10/40/100 GbE high-speed network switch.

The digital baseband data converter is implemented on a SNAP2 board. SNAP2 is designed by the Institute of Automation, Chinese Academy of Sciences in 2017, which carries a Xilinx Ultrascale XCKU115-FLVF1924 Field Programmable Gate Array (FPGA), with four QSFP+ ports, four QDRII and one DDR3 memory for a high-speed data process, cache, and output. Two ADC cards each carry an EV10AQ190 chip connect to this board through two FMC interfaces for high-speed data sampling. It can be configured as 1-channel 5 GSPS, 2-channel 2.5 GSPS, or 4-channel 1.25 GSPS, with a sampling precision of 10-bit.

The baseband data recorder is realized on four servers. The server model is Supermicro 4029GP-TRT2.

It carries two Gold 6226R CPUs, eight 32 GB DDR4-2933 MHz memories, two 40/100 GbE network interface cards (NICs), and two 1 TB NVMe SSD storage cards. The NICs and NVMe SSDs are connected to the motherboard through PCIE3.0x8 slots.

2.2 Design of the SNAP2 Digital Baseband Data Converter

The FPGA firmware design and operation management of the SNAP2 board is based on the CASPER² toolkit. CASPER is the Collaboration for Astronomy Signal Processing and Electronics Research whose primary goal is to facilitate design reuse by developing platform-independent, open-source hardware and software to simplify the design process for radio astronomy instruments. Over the past decade, CASPER technology and related instruments have been used on hundreds of telescopes worldwide (Hickish et al. 2016). A Microblaze controller running the Linux operation system is used to manage and monitor the SNAP2 platform and communicate with the host server. The Casperfpga library installed on the host server establishes a connection with Microblaze, and FPGA firmware download, register configuration, data acquisition and status monitoring can all be easily completed through Python scripts.

2.2.1 Firmware design

Nine modes are designed to meet the requirements of QTT single-beam and multi-beam receivers for different sampling bandwidths and input channels. A detailed configuration of different modes is shown in Table 1. Modes 1–4 are designed for single beam dual-polarization receivers with bandwidths of 256, 512, 1024, or 2048 MHz, respectively. Using an external band-pass filter to select the frequency band of 2–4 GHz, mode 4 can reach the maximum instantaneous bandwidth of 4 GHz for a single input. This mode will be considered as one of the sampling schemes for the QTT wideband receivers. Capable of handling more inputs, modes 5–9 can be selected as the D-F-engine to digitize and channelize signals from multi-beam or PAF receivers with bandwidths of 256, 512, or 1024 MHz, respectively. The maximum output data rate of SNAP2 is 64 Gbps, which will be output through eight high-speed digital links with 8192 Mbps for each link. Considering the processing throughput of a single server, for the bandwidth greater than 512 MHz, the signal must be split into multiple sub-bands and processed by multiple servers.

¹ <https://casper.ssl.berkeley.edu/wiki/SNAP2>

² https://casper.ssl.berkeley.edu/wiki/Main_Page

Table 1 Configurations of Different Modes

Mode	N_inputs	BW(MHz)	N_bit	N_subband	Total DTR ¹ (Mbps)	N_lanes	DTR ¹ per lane(Mbps)
1	2	256	16	1	8192	2	4096
2	2	512	16	1	16384	2	8192
3	2	1024	16	2	32768	4	8192
4	2	2048	16	4	65536	8	8192
5	4	256	16	1	16384	4	4096
6	4	512	16	1	32768	4	8192
7	4	1024	16	2	65536	8	8192
8	8	256	16	1	32768	8	4096
9	8	512	16	1	65536	8	8192

DTR is Data Transfer Rate.

The block diagrams of firmware design are shown in Figure 2. Figure 2(a) shows the firmware design block diagram of mode 4, which is the most complicated scenario for two inputs. Figure 2(b) is the firmware design block diagram of mode 9, which is an example to illustrate eight input signal sampling, processing, and transmitting flow.

In mode 4 (see Fig. 2(a)), each ADC collects one signal at a sampling rate of 4096 MSPS and a sampling precision of 10-bit. A total of two signals are collected from the two ADCs. Since the sampling clock of ADC is higher than the highest clock frequency that FPGA can run, one channel is split into 16 parallel streams for processing inside FPGA. For example, in the ADC0 module, 16 channels (a0, a1, a2 ... a15) are output after the input signal 'a' is downsampled by 1/16. In mode 9 (see Fig. 2(b)), a total of eight signals are collected by two ADCs each with four inputs, at the sampling rate of 1024 MSPS with 10-bit precision. After the ADC module, one signal is divided into four parallel streams for further FPGA processing.

Polyphase filter bank (PFB) and FFT modules are used to channelize the input signals. The PFB is a set of digital filters that consist of a more computational efficient set of polyphase finite impulse response (FIR) filter frontends to provide better isolation between channels. The number of simultaneous inputs in PFB is set as 32 to match the outputs from ADC0 and ADC1. The input data type is UFix_10_0 (UFix represents Unsigned Fixed-point, whose bit-width is 10 and binary point is 0), and the output data type is Fix_18_17 (Signed Fixed-point with 18 bit-width and 17 binary point). The sizes of PFB in mode 4 and mode 9 are set to 32 768 and 8192, respectively, and they both use a 6-order Hamming window for filtering.

Real FFT modules are selected according to the real signal inputs of ADC. The number points of FFT must be the same as the size of PFB, which is set to 32 768 and 8192 for modes 4 and 9, respectively. The number inputs of FFT have to be identical with the number outputs from PFB, which is 32. In mode 4, the number of simultaneous streams is set to 2, and the number of simultaneous inputs in each stream is set to 16. In mode 9, these numbers are 8 and 4. The number of output channels is 16, which is

half of the FFT calculation. A total of 32 parallel real data streams enter and 16 complex data streams out after FFT calculation. The output data type of the FFT module is UFix_36_0.

From this point and onwards, modes 4 and 9 share the same modules of c_to_ri, Quant, Concat, Pack, and Eth.

The c_to_ri modules are used to separate the real and imaginary parts from the complex data in preparation for quantization. After the c_to_ri modules, re and im two signals are output. A total of 16 c_to_ri modules are used to process the 16 FFT output signals.

The Quant module receives 32 input signals, converts the data type of Fix_18_17 to UFix_8_0, and then outputs 32 signals. The number of input and output signals does not change, with each input signal corresponds to one output signal, for example, in00_i -> out00_I, in00_q -> out00_q. A 32-bit register is used to adjust the scale of quantization numbers.

The Concat modules combine four signals into one data stream, with the purpose of making each RF signal output through one network port. For example, out00_i, out00_q, out01_i, out01_q four streams come from RF_0 and are combined and sent out over Eth0. There are eight Concat modules that output eight streams, and the input data type is UFix_8_0 and the output data type is UFix_32_0.

The Pack modules packetize the incoming data in VDIF format, generate network control signals such as valid and eof, and configure IP addresses and port numbers for Eth modules. There are eight Pack modules, each receives one stream from the Concat module and outputs to an Eth module. Because the data width of 10 GbE is 64 bits, the Pack module aggregates two UFix_32_0 samples into a single UFix_64_0 data type. The Pack module comes with block random access memory (BRAM) to buffer the data before transmission.

The Eth modules receive data from Pack modules and output over the 10/40 GbE network. There are four 40 GbE interfaces (QSFP) on SNAP2, each can be split into four 10 GbE links, which can be achieved by a 1–4 breakout cable, and two QSFP ports are used to get eight 10 GbE links in

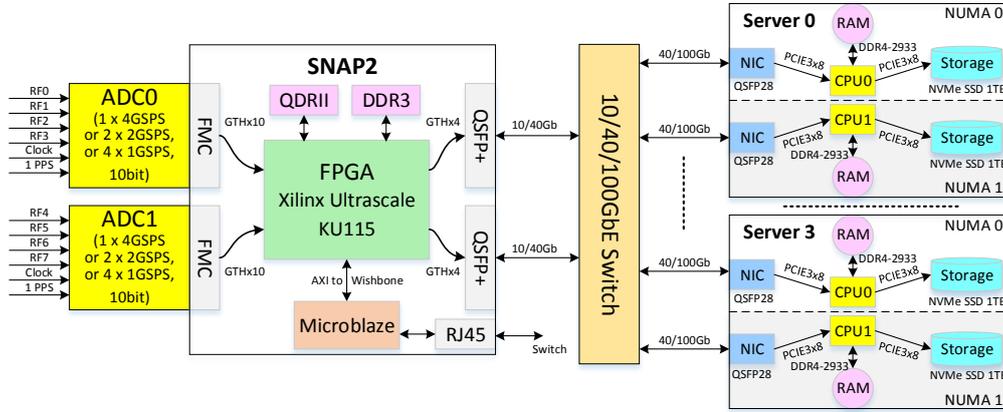


Fig. 1 A block diagram of the hardware architecture.

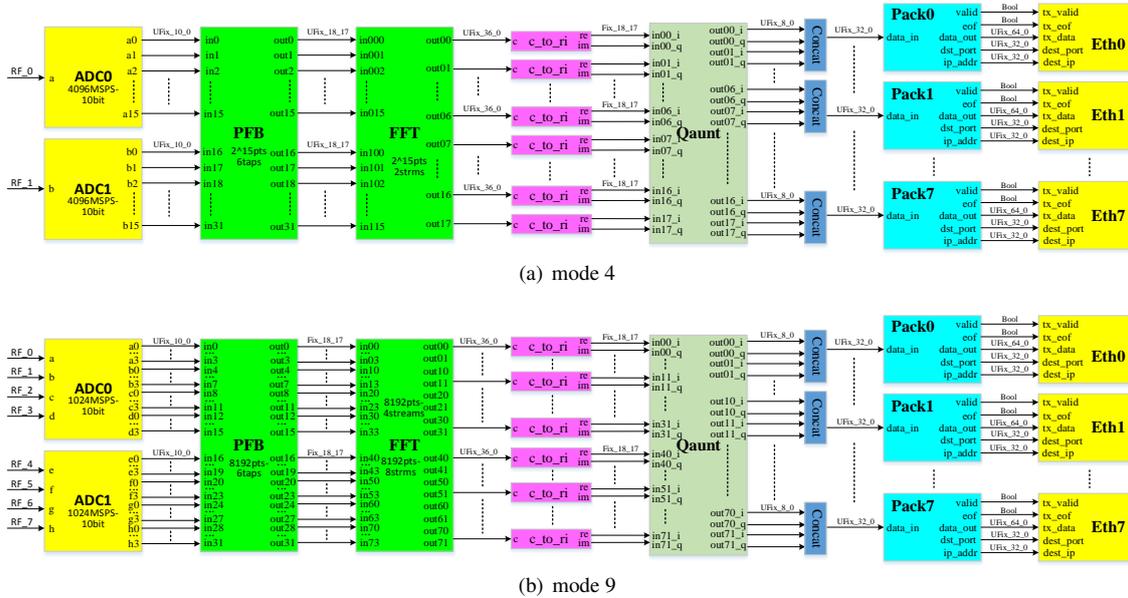


Fig. 2 A firmware design block diagram of different modes.

this design. The Eth modules are configured by enabling large TX Frames (8k+512), enable fabric MAC address, IP address, and UDP port on startup.

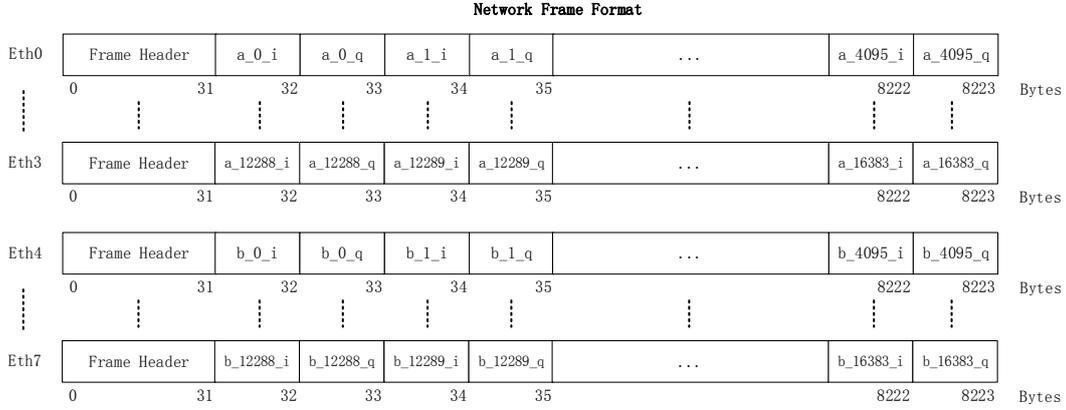
2.2.2 Output data format

The VDIF file format³, originally introduced and standardized by VLBI observations, is now being popularized in astronomy and supported by a variety of software and hardware systems. The most widely used software correlator - DiFX (Distributed FX) (Deller et al. 2011) fully supports the VDIF format. The Australian Parkes

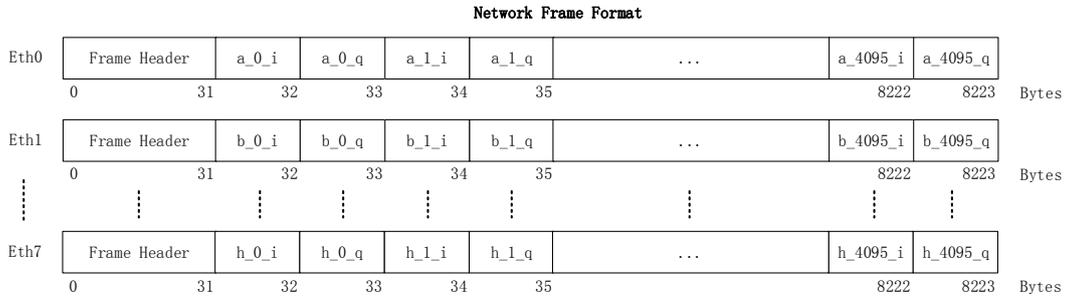
64-meter UWB system sends VDIF baseband data from FPGAs to Medusa HPCs for post-processing (Hobbs et al. 2020) such as pulsar timing, pulsar search, continuous spectrum, and VLBI, etc.

The FPGA output of the system is designed in the standard VDIF format. The data processed by FPGA is output through multiple network modules, each of which forms a data link and is output through the UDP packet. The packet size is 8224 bytes, and each packet contains a VDIF data frame. The data frame formats of modes 4 and 9 are shown in Figure 3(a) and 3(b), respectively. Each data frame contains 32 bytes of the frame header, 8192 bytes of data, and each data point is represented by a 2-

³ <https://vlbi.org/vlbi-standards/vdif/>



(a) mode 4



(b) mode 9

Fig. 3 The output data frame format of different modes.

byte complex number, 1 byte real part and 1 byte imaginary part. The frame contains 4096 data samples.

2.2.3 FPGA resource utilization

Table 2 shows the utilization of FPGA resources. The utilization of FPGA in mode 9 is less than 26% in all items. The size of the PFB and FFT in mode 4 is 32 768, and this number in mode 9 is only 8192, so the consumption of LUT, LUTRAM, FF, BRAM, and DSP in Mode 4 is higher.

2.3 Design of the Baseband Data Recorder

The baseband data recorder is realized on four servers. Each server carries two NVMe SSD cards (WD_BLACK AN1500), with 2 TB of space. Each SSD card connects to one socket on the motherboard through a PCIe3.0x8 slot. The test results show that the continuous write rate of each SSD is around 1.8 GB s^{-1} , and the two SSDs can achieve a write rate of 3.6 GB s^{-1} .

HASHPIPE⁴, the High Availability SHared PIPEline Engine, provides a C application programming interface (API) for designing parallel pipelines, where processing blocks are run in separate threads and join with ring buffers. It is usually used to move data through an X-engine running on a CPU or GPU. HASHPIPE code is a derivative of GUPPI (Green Bank Ultimate Pulsar Processing Instrument) (DuPlain et al. 2008), and later modified by UC Berkeley for the Green Bank VEGAS multibeam spectrometer (Chennamangalam et al. 2014). Most recently it has been modified and adapted in the Serendip VI backend system and FAST multi-beam FRB/SETI backend system (Pei et al. 2019). The data recorder software for our system is designed based on HASHPIPE, and the software architecture is shown in Figure 4.

Each server running two HASHPIPE data recorder instances, with each instance consisting of three threads: a network thread (th_net), calculation thread (th_calc) and file write thread (th_wr). When the program starts, two ring

⁴ <https://github.com/david-macmahon/hashpipe>

Table 2 The Utilization of FPGA Resources

Resource	Available	Utilization (mode 4)	Utilization%(mode 4)	Utilization (mode 9)	Utilization%(mode 9)
LUT	663360	136176	20.53	118872	17.92
LUTRAM	293760	27206	9.26	24310	8.28
FF	1326720	235200	17.73	198144	14.93
BRAM	2160	607.5	28.13	447.5	20.72
DSP	5520	706	12.79	578	10.47
IO	728	189	25.96	189	25.96
GT	64	9	14.06	9	14.06
BUFG	1248	32	2.56	32	2.56
MMCM	24	1	4.17	1	4.17

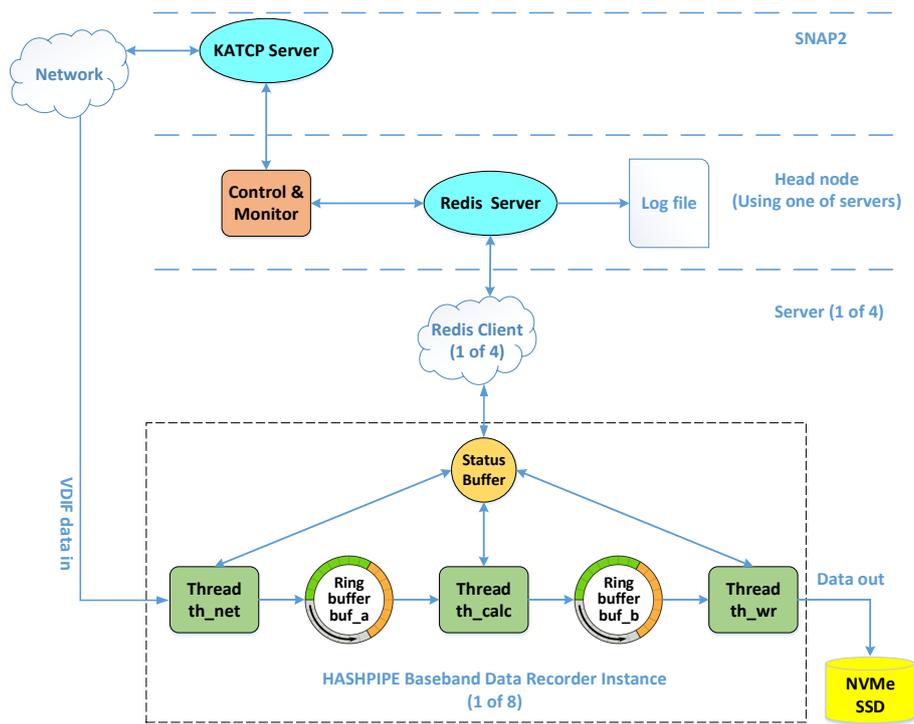


Fig. 4 The software architecture of baseband recorder.

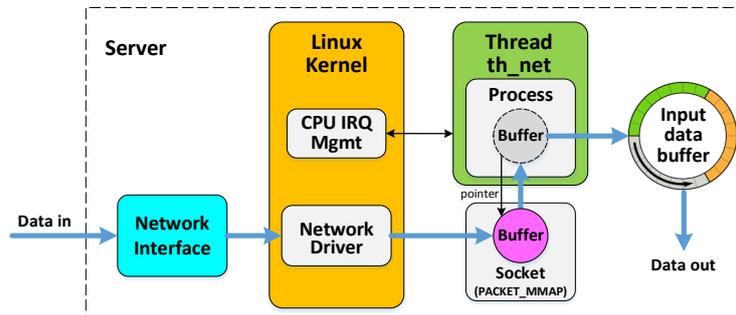


Fig. 5 A block diagram of receiving Packet_MMAP raw socket.

buffers (buf_a and buf_b) are allocated, among which buf_a and buf_b are input and output data buffers respectively. The data can be read or written between threads efficiently through ring buffers. Thread th_net and th_calc share ring buffer buf_a, thread th_calc and th_wr share ring buffer buf_b.

The thread th_net receives high-speed VDIF data packets from SNAP2 and put them in buffer buf_a. The thread th_calc reads data from buffer buf_a and reassembles the data in specified type and format, and then the results are copied to buffer buf_b. The thread th_calc only performed simple data processing in this design. To some

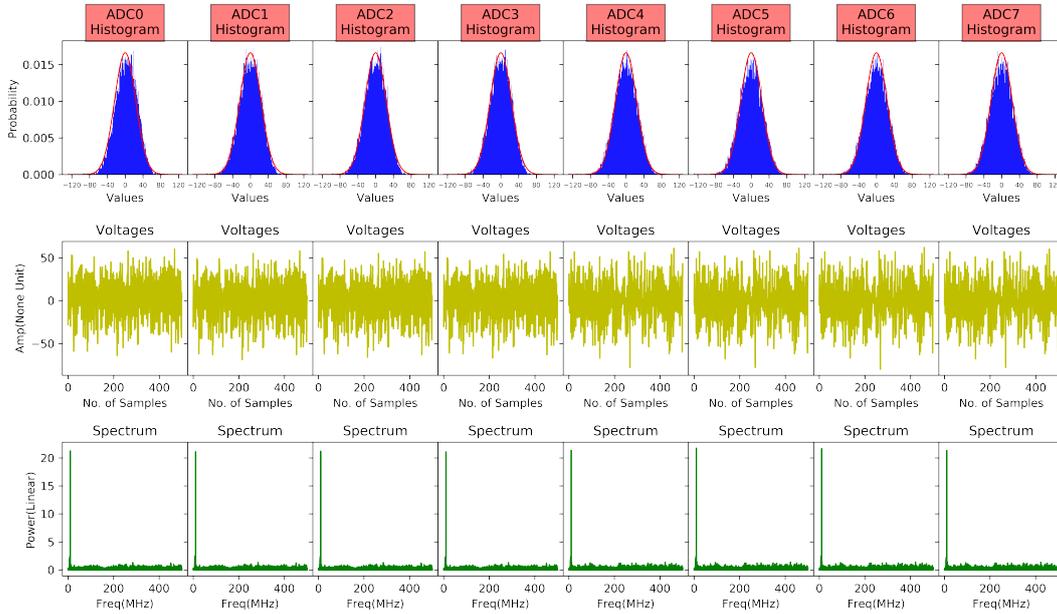


Fig. 6 A plot of ADC raw data captured by SNAPSHOT module.

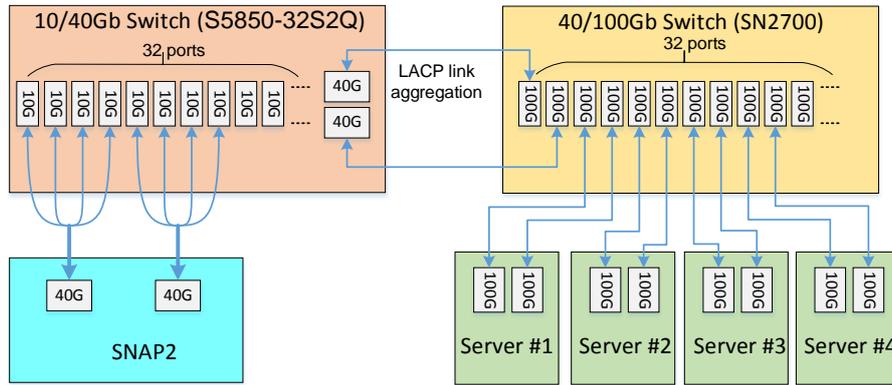


Fig. 7 The high speed network architecture.

extent, it offers potential solutions for real-time computing that might involve GPUs. In this case, the data must first be copied from the CPU memory to the GPU memory, and then the calculation results are copied back from the GPU memory to the CPU memory after the GPU computation is completed. Thread `th_wr` reads the data from buffer `buf_b` and makes the file header according to the observation settings, and then writes the file header and data to disk.

To avoid data loss when three threads execute asynchronously, multiple buffer blocks can be created for data buffer `buf.a` and `buf.b`. A HASH table is established to identify the processing state of each thread and buffer, and each thread queries the status of the HASH table for operation selections, such as wait, execute, write, read, and so on. The status buffer also provides real-time statistics

and updates on the number and loss rate of data frames, the number of file writes, and the size of the current write, etc.

To synchronize eight instances from four servers, a Redis database server is set up on one of the servers (see the Head node in Fig. 4), and a status key `is_start` is added in the database. When HASHPIPE instances detect that `is_start` equal 1, it will start receiving network packets and store data in the next second. The Redis gateway periodically transfers status buffer to the Redis server.

Besides, the Control & Monitor runs on the Head node. It accesses the registers of SNAP2 through the KATCP service running on SNAP2, which configures the parameters such as software synchronization, software reset, FFT gain, quantization scaling, 10 GbE network

```

Current Status: Instance 0
-----
BINDHOST : ens11f0          BINDPORT : 60000
DATADIR  : /buff0/snap2_vlbi/  DATSAVMB : 643
ISSAVE   : 1                NFILESAV : 1
NETBKOUT : 4                NETDROPS : 0
NETMCNT  : 4                NETRCV   : 81920
NETSTAT  : receiving        PKTLOSS  : 0
PKTSIZE  : 8232            LOSSRATE : 0
OUTSTAT  : processing

Last update: Thu Dec 3 19:02:18 2020 -- Press 'q' to quit, 0-9 to select

```

Fig. 8 A print of HASHPIPE status monitor on instance 0.

port enables and reset, self-test waveform enable, VDIF settings, etc. In addition, it can read ADC SNAPSHOT, calculate and print statistical histograms, spectrum diagram, and RMS, and monitor the running state of the network port, the power consumption, temperature, and fan state of FPGA, etc.

The parameters and status from Control & Monitor are periodically transferred to the Redis Server running in the server's memory. To avoid data loss caused by power outages, the database is written into a log file for regular backup.

3 PERFORMANCE OPTIMIZATION

To improve the performance of packets receiving and data writing, network optimization, non-uniform memory access (NUMA) setting, NIC card tuning, and I/O strategy are considered, as described below.

3.1 Acquiring Raw Socket

In Linux 2.4/2.6, network packets are transmitted in a single-mode such that one system call can only transmit one packet, and multiple packets need to be called for several times, so the capture process is inefficient.

To increase the transmission efficiency of network packets in the Linux system, the Packet.MMAP mechanism is adopted in the system. This mechanism creates a ring buffer between the user and the kernel, which means that network packets can be directly transferred to the buffer. In this way, multiple packets can be transmitted by single read or send operation, which minimizes the number of system calls and improves transmission efficiency.

A block diagram of receiving Packet.MMAP raw socket is shown in Figure 5. The Packet.MMAP is open in the network thread when initializing the socket, and a size scalable ring buffer is allocated in the kernel and mapped to the userspace. The network driver moves the packets from the network interface to the ring buffer ceaselessly.

Once the ring buffer is full, the user thread obtains multiple network packets by directly reading the local ring buffer. Reading packets in this way simply wait for data, and do not require frequent interrupt requests to the system. One can also minimize package duplicates by using shared buffers between the kernel and the user.

Another approach to improve the packets receiving efficiency in this design is called a 'raw socket' – which bypasses the TCP/UDP layer and gives access and manipulate the header and trailer information un-extracted packets from lower layers.

3.2 NUMA Binding

The components of the server are connected via the system bus. For multiple CPU computers, each CPU acts as the center of the NUMA node that connects to NIC, GPU, RAM, and other components through its unique system bus. Components have to bind to the corresponding CPU according to the physical locations, otherwise access across NUMA nodes is much slower than access within nodes.

The servers in this system each are equipped with two CPUs, two NICs, two GPUs, and two SSD cards that need to be inserted into the PCI-E slots corresponding to each CPU socket.

In Linux, the 'numactl' command-line is generally used to bind a process to a specified core. Before binding, the topology of NUMA sockets has to be cleared. On each server of this design, there are 32 CPU cores in each CPU, core 0–15, 32–47 from CPU 0 on NUMA 0, core 16–31, 47–63 from CPU 1 on NUMA 1. Six CPU cores are bound for two instances, each with three cores. The detailed NUMA setting is shown in Table 3. NUMA Node 0 is assigned for instance 0, which binds CPU core 8, 9, 10 for thread th_net, thread th_calc, and thread th_wr, respectively, the ring buffer is allocated on RAM 0. NUMA Node 1 is assigned for instance 1, which binds CPU core 16, 17, 18 for thread th_net, thread th_calc and thread th_wr, respectively, the ring buffer is allocated on RAM 1.

3.3 40/100 GbE Network Tuning

The settings for continuous transmission of a high-speed data network in the Linux system kernel are not optimal. To maximize the throughput of 40/100 GbE network, some settings need to be made on the system, which includes increasing TCP buffer size, using Jumbo frames, binding interrupts to the corresponding CPU cores, changing the CPU governor to 'Performance', etc.

Table 3 NUMA Assignment

Instance	inst 0			inst 1		
Threads	Th_net	Th_calc	Th_wr	Th_net	Th_calc	Th_wr
CPU cores	8	9	10	16	17	18
Devices	NIC 0	GPU 0	NVMe SSD 0	NIC 1	GPU 1	NVMe SSD 1
NUMA	Node 0			Node 1		

	Byte 3		Byte 2	Byte 1	Byte 0
Word 0	I	L	Second from reference epoch		
Word 1	Unassigned		Ref Epoch	Data Frame # within second	
Word 2	V	$\log_2(\#chns)$		Data Frame Length (units of 8 bytes)	
Word 3	C	bits/sample-1	Thread ID		Station ID
Word 4	EDV		Extended User Data		
Word 5	Extended User Data				
Word 6	Extended User Data				
Word 7	Extended User Data				

Fig. 9 VDIF frame header definition.

3.4 Direct I/O

Linux allows applications to bypass the buffer cache when executing disk I/O and pass data directly from userspace to a file or disk device through Direct I/O. To achieve faster data write rates, the O_DIRECT flag is configured to the C file operation to force the in-memory data to be written directly to SSD.

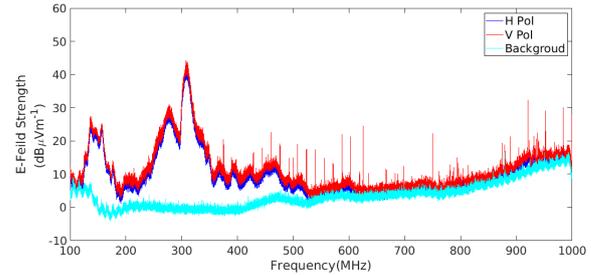
4 TESTING

4.1 Lab Test

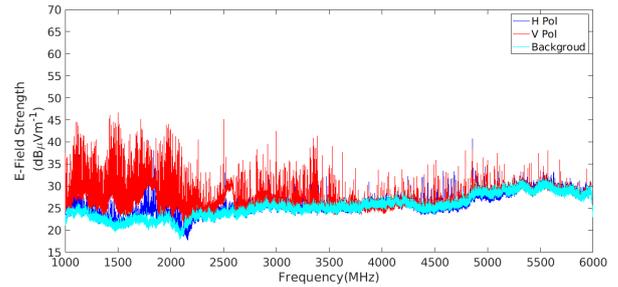
4.1.1 ADC sampling testing

A signal generator (the model is R&S SMA100B) is used to generate a sinusoidal signal, which is combined with a noise source through a reverse-connected power splitter. The SNAP2 runs in mode 9, where the combined signal is split into eight channels and injected into two ADCs. The ADC samples are captured by the SNAPSHOT module and buffered in SNAP2’s BRAM. A plot of captured raw data is shown in Figure 6. The eight columns represent each ADC input, and three rows from the top are histogram of samples, raw voltage samples, and power spectrum after FFT transformation, respectively. The signal generator outputs a sine wave signal with a frequency of 10.5 MHz and a power of -20 dBm.

The first line shows beautiful Gaussian distribution maps, while the second line only shows noises. That is because the weak signals are submerged by noise and cannot be identified in the time domain. After channelizing the sinusoidal signal and transforming it into



(a) From 100 MHz to 1000 MHz



(b) From 1000 MHz to 6000 MHz

Fig. 10 EMC test results of SNAP2 board.

the frequency domain, the straight-line signals appear at exactly 10.5 MHz as shown in the third row (The y-axis represents the amplitude (Linear) of the signal, and the x-axis is the frequency (MHz)).

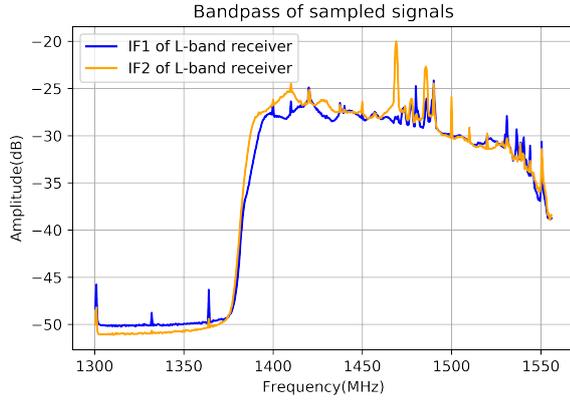


Fig. 11 The spectra of the two polarizations.

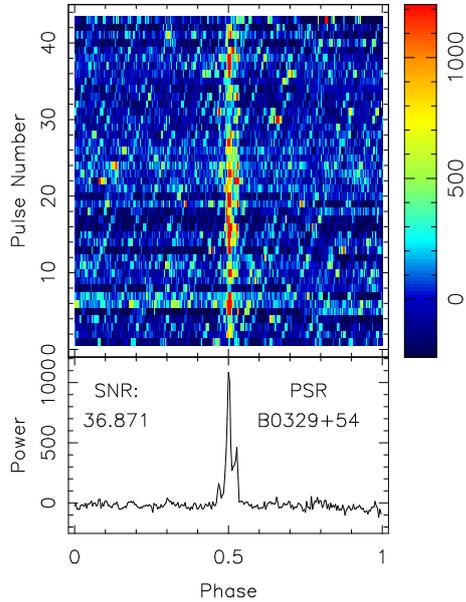


Fig. 12 A pulse stack plot of the observed pulsar B0329+54.

4.1.2 Network configuration and data integrity testing

The high-speed network architecture is shown in Figure 7. Two switches are used for 10–40–100 GbE network negotiation, which are FS S5850-32S2Q and Mellanox SN2700. The S5850-32S2Q is used in 10/40 GbE networks, with thirty-two 10 GbE (SFP+) ports and two 40 GbE (QSFP+) ports, and eight 10 GbE ports are connected to SNAP2 board by two 1–4 (40–10 GbE) breakout cables. The SN2700 is used in 40/100 GbE networks, with thirty-two 100 GbE ports (QSFP28), and each server connects to this switch with two ports occupying a total of eight ports. The two switches are connected by two links using Link Aggregation Control Protocol (LACP) link aggregation with a bandwidth of 80 Gb s^{-1} between them.

Twenty data blocks are allocated on each ring buffer when the HASHPIPE instance is initialized. Each block

has the data size of $20480 \times 8232 \text{ B} = \sim 160.8 \text{ MB}$, giving totally: $2 \text{ instances} \times 2 \text{ ring buffers} \times 20 \text{ blocks} \times 160.8 \text{ MB/block} = \sim 6.28 \text{ GB}$ on each server. After setting the value of key `is_start` to 1 in the Redis database, all eight HASHPIPE instances begin to capture network packets. HASHPIPE provides the Ruby scripts ‘`hashpipe_status_monitor.rb`’ to monitor the running HASH status of all instances. A print of status instance 0 is shown in Figure 8, which can display the running state of three threads in real-time, such as network packet receiving counting, packet loss rate, and file storage counting.

The data loss rate is tested under multiple modes with a different number of inputs, bandwidths, servers, and data rates. The test results are shown in Table 4. The maximum total data recording rate of 8 GB s^{-1} is achieved in modes 4, 7, and 9 by four servers each with two instances. There are two different data recording rates on a single server, and the data loss rate increases with the data recording rate. For the data recording rates of 1 and 2 GB s^{-1} on a single server, the data loss rates are 0 and 0.01%, respectively.

4.1.3 VDIF header checking

The number of VDIF data frames in a second should be an integer according to the VDIF data specification. Each frame has 8192 samples, the number of data frames per second is $1/(8192 \times 1/1024) \times 10^6 = 125000$ when the sampling rate is 1024 MSPS, and the number of data frames per second is $1/(8192 \times 1/512) \times 10^6 = 62500$ when the sampling rate is 512 MSPS. Each frame has an 8-byte header, which is separated by four words, each of four bytes. The detailed definition of the VDIF header⁵ is shown in Figure 9.

The packet headers are extracted from the captured VDIF frames and verified by epoch time conversion and data information checking. In the VDIF header, bits 24–29 in word 1 represent epoch number, which starts from 2000 Jan 1, and increases by one every six months meaning that a total of six digits can be expressed as of 2031 Dec 31. The bits 0–29 in word 0 represent epoch seconds starting from the date of epoch number. The bits 0–23 in word 1 represent the number of frames in a second, and the maximum number is 124 999 in this testing. The UTC is 2020-11-26 01:46:28 when the VDIF packets are captured. The reference epoch number should be 41, and the seconds count of reference epoch should be $148 (N_{\text{date}}) \times 86400 (\text{sec per date}) + 1 (N_{\text{hour}}) \times 3600 (\text{sec per hour}) + 46 (N_{\text{min}}) \times 60 (\text{sec per min}) + 28 = 12793588$. Word 0 from the captured header in hexadecimal is `0xc336f4`, which is identical to our calculation.

⁵ <https://vlbi.org/wp-content/uploads/2019/03/VDIF-specification-Release-1.0-ratified.pdf>

Table 4 The Test Results of Data Loss Rate under Different Data Recording Rates

Total DRR ¹ (GB s ⁻¹)	Mode	N_servers	N_instances	DRR ¹ per server(GB s ⁻¹)	DLR ² (%)
1/2/4	1/5/8	1/2/4	2/4/8	1	0
2/4/8	2/3/4/6/7/9	1/2/4	2/4/8	2	0.01

1. DRR is Data Recording Rate; 2. DLR is Data Loss Rate.

Word 1 from the captured header is 0x29015661, which represents the reference epoch number of 41 and the number of frames in a second is 87 649.

Word 2 from the captured header is 0xc000404, which represents the number of channels at $2^{12} = 4096$, and the data frame length is 1028 under the unit of 8 bytes.

Word 3 from the captured header is 0xa0015572 signifying that the complex type is selected, with the Thread ID being 1, and the station ID being ‘Ur’ (in Ascii code).

4.1.4 EMC test

The electromagnetic radiation generated by the digital processing device will affect the observation of the radio telescope. The SNAP2 board (referred to as the EUT, Equipment Under Test) is tested in a microwave anechoic chamber at a distance of 1m from the receive antenna over the range 100 MHz to 1 GHz using a large horn antenna, and from 1 GHz to 6 GHz using another small horn antenna. The measurement method and procedure conform to the GJB151B-2013 (GJB151B-2013, 2013) standard, and the maximum detector is used for power detection in the electric field. The test results are shown in Figure 10, the blue and red lines represent the horizontal and vertical polarization components, respectively, and the green line represents the background noise. It can be seen from Figure 10(a) that between 100 MHz and ~ 520 MHz, the EUT emits strong radiation and raises the background noise, the maximum emission is around $43 \text{ dB}\mu\text{V m}^{-1}$ in both polarization. Occasional narrowband emissions are seen from ~ 350 MHz to 1000 MHz. In Figure 10(b), narrow-band radiations appear in the entire test band between 1000 MHz and 6000 MHz, especially from 1000 MHz to ~ 3500 MHz, the radiation is dense and the values are relatively high. These tests are carried out without any electromagnetic shielding and protection of the high-speed processing device. In practical applications, SNAP2 needs to be placed in an electromagnetic shielding cabinet to reduce electromagnetic radiation.

4.2 Observing Experiment

The system is deployed on the Nanshan 25-meter radio telescope (NSRT) for the observation experiments. The system is configured as mode 1 with 2 inputs and 256 MHz bandwidth. Two signals from the L-band

cryogenic receiver are fed to ADC, and the receiver’s RF frequency range is 1400 – 1720 MHz. One server is used to record data at the storage rate of 1 GB s^{-1} . The file size is set as 16 GB to record 32 seconds of data. About 5 minutes of data was recorded when NSRT was tracking pulsar B0329+54 giving a total of ~ 300 GB data recorded in 20 files. The two polarization signals are stored in separate files. Figure 11 shows the spectra of the two polarization signals. We can see that the band-pass range of these signals is from 100 MHz to 240 MHz, which is consistent with the actual signal link. The IF signal starts at 100 MHz and inserts a 240 MHz low-pass filter before entering the ADC. One of the 32-second long data files is selected and processed in DiFX. The signal is folded every ~ 0.714 seconds after de-dispersion, and a distinct pulse appears as shown in Figure 12.

5 DISCUSSION AND CONCLUSION

The maximum continuous write rate of the HDD is around 200 MB s^{-1} (7200 RPM HDD). For applications with a higher writing rate, Redundant Array of Independent Disks (RAID) cards are generally used and configured as RAID 0 to achieve parallel writing of multiple HDDs. This method requires high-performance RAID cards and multiple hard drives, which increases the cost of the server. With the development of SSD technology, a continuous write rate of $\sim 2 \text{ GB s}^{-1}$ can be achieved on a single NVMe SSD, which is comparable to the rate of at least 10 HDDs in RAID 0 parallel mode. The design of the QTT backend system will adopt SSD as fast storage, triggering the storage program to quickly save the raw voltage data when an astronomical signal is detected. This simply requires the HPCs to add SSDs, which is straightforward. As the price of SSDs drops, we expect more SSDs to be used for fast data dumping in astronomy in the future.

The system uses the PACKET_MMAP method to improve the efficiency of network packets fetching. This method maps the NIC buffer to the ring buffer of user thread, and grab multiple packets at one time, which reduces the number of CPU calls and interrupts, thus improving efficiency. However, this method still consumes CPU resources. For applications that require higher speed (above 40 Gb s^{-1} per server), RDMA (remote direct memory copy) is recommended as this method bypasses Kernel and copy data from memory to the device without CPU calls. Data Plane Development Kit (DPDK) and

ibverbs are RDMA methods, which are widely used in the development of astronomical instruments. However, the RDMA method only supports certain models of network cards. For example, the Mellanox ConnectX-4/5 EN series need to install specific network card drivers and modify some configurations in the operating system. This method limits the versatility and increases the complexity of use. The PACKET_MMAP method is less efficient than the RDMA method, but it supports most network cards and operating systems, which is easy to use and more versatile.

With the improvement of GPU computing performance, more and more designs transfer high-intensity computing work from FPGAs to GPUs, while FPGAs only perform some simple channelizing and packetizing, and then outputs the raw voltage data. Our design of the digital baseband converter and recorder is aligned with the philosophy of this trend. As part of the QTT signal sampling and processing platform, the design will be integrated into the QTT digital backend system for VLBI data recording, transients detection, coherent pulsar de-dispersion, adaptive radio-frequency interference (RFI) mitigation, etc. By slightly changing the output data format of SNAP2, it can be used as a D-F engine and work with an X-Engine in the Correlator or B-Engine in the Beamformer system.

Acknowledgements We thank Jeff Cobb and David MacMahon for their help in designing the HASHPIPE software, and Jie Hao, Lin Shu, Qiu-xiang Fan and Liang-tian Zhao for providing SNAP2 ADC calibration and development libraries, Dan Werthimer and Jack Hickish for their advice on the architecture design of the QTT signal processing system, Danny Price and Jason Manley for their advice on tuning high-speed networks, Zhigang Wen for his help on the pulsar data processing, and Qi Liu for his help on the EMC test. We also thank Rai Yuen for his help with writing and grammar. Finally, we thank the referee for helpful comments. This work was funded by the National Natural Science Foundation of China (NSFC, Nos. 12073066, 61931002 and 12073067), and the Youth Innovation Promotion Association of Chinese Academy of Sciences (CAS, No. 2020063). The research is partly

supported by the Operation, Maintenance and Upgrading Fund for Astronomical Telescopes and Facility Instruments, budgeted from the Ministry of Finance of China (MOF) and administrated by the CAS.

References

- Cornwell, T. J. & Perley, R. A. 1991, *Astronomical Society of the Pacific Conference Series*, Vol. 19, *Radio Interferometry: Theory, Techniques, and Applications*
- Chennamangalam, J., Scott, S., Jones, G., et al. 2014, *PASA*, 31, e048
- Deller, A. T., Brisken, W. F., Phillips, C. J., et al. 2011, *PASP*, 123, 275
- DuPlain, R., Ransom, S., Demorest, P., et al. 2008, in *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, 7019, *Advanced Software and Control for Astronomy II*, eds. A. Bridger & N. M. Radziwill, 70191D
- Hickish, J., Abdurashidova, Z., Ali, Z., et al. 2016, *Journal of Astronomical Instrumentation*, 5, 1641001
- Hobbs, G., Manchester, R. N., Dunning, A., et al. 2020, *PASA*, 37, e012
- Ma, J., Pei, X., Wang, N., et al. 2019, *Scientia Sinica Physica, Mechanica & Astronomica*, 49, 099502
- Ma, J., Wu, Y., Xiao, S., Niu, S.-P., & Wang, K. 2021, *RAA (Research in Astronomy and Astrophysics)*, 21, 088
- MacMahon, D. H. E., Price, D. C., Lebofsky, M., et al. 2018, *PASP*, 130, 044502
- Pei, X., Li, J., Li, S., & Niu, C. 2019, *Scientia Sinica Physica, Mechanica & Astronomica*, 49, 099508
- Peoples Republic of China General Equipment Department, GJB151B-2013 *Electromagnetic Emission and Susceptibility Requirements and Measurements for Military Equipment and Subsystems*, Peoples Republic of China National Military Standards, 2013
- Wang, N. 2014, *Scientia Sinica Physica, Mechanica & Astronomica*, 44, 783
- Whitney, A., & Lapsley, D. 2012, in *Seventh General Meeting (GM2012) of the international VLBI Service for Geodesy and Astrometry (IVS)*, 86