Pulsar Astrophysics the Next Fifty Years
Proceedings IAU Symposium No. 337, 2017
P. Weltevrede, B.B.P. Perera, L.L. Preston
& S. Sanidas, eds.

doi:10.1017/S1743921317009036

An S-band Receiver and Backend System for MeerKAT

Ewan D. Barr on behalf of the MPIfR S-band team

Max-Planck-Institut für Radioastronomie,
Auf dem Hügel 69,
53121 Bonn,
Germany
email: ebarr@mpifr-bonn.mpg.de

Abstract. The MPIfR is working together with SKA-SA and the Universities of Manchester and Oxford to deploy three instruments on MeerKAT: An S-band receiver system, a dedicated beamforming cluster and a flexible pulsar search cluster. Together these instruments will provide MeerKAT with powerful tools for supporting a wide range of scientific applications and in particular will enable large-scale pulsar and fast transient surveys to be performed. In these proceedings we describe the design, implementation and deployment timeline for these instruments.

1. Introduction

In order to pave the way to construction of the Square Kilometre Array (SKA), several world-class precursor instruments have been constructed. The MeerKAT radio telescope serves as the precursor instrument for the mid-frequency component of the SKA. This 64 × 13.5-m dish interferometer located in the Karoo region of South Africa is rapidly approaching completion, at which point it will become the most sensitive radio telescope in the Southern hemisphere, with ~ 5 times the sensitivity of the Parkes radio telescope and ~ 20 times its field of view (FoV). In order to help maximise the scientific return, the MPIfR, with financial support from the Max Planck Society and in collaboration with the Universities at Manchester and Oxford, is working with SKA-SA to provide MeerKAT with additional capabilities in the form of three new large-scale instruments. Foremost amongst these instruments is a set of octave receivers covering 1.75 to 3.5 GHz (S-band). To support operations with these receivers, we will deploy a dedicated beamformer and a multi-purpose compute cluster intended to primarily support binary pulsar searches but capable of handling a wide range of processing tasks. The MPIfR will be allocated 3000 hours of dedicated observing time with this system when deployed and after that these instruments will be open to use by the community. In these proceedings we will cover the design and timeline of these instruments and discuss their principle use cases. For more details on the scientific drivers behind these instruments, we refer the reader to Kramer *et al.* (2016).

2. S-Band Receiver

The MPIfR S-band receiver system is designed to complement MeerKAT's existing and in-development receivers, providing coverage in the range 1.75 – 3.5 GHz filling much of the gap between the existing L-band (0.86 - 1.71 GHz) and planned X-band (8 – 14.5 GHz) systems. This frequency range is interesting for a broad set of scientific applications. From the perspective of pulsar astronomy the higher frequency offered by

https://doi.org/10.1017/S1743921317009036 Published online by Cambridge University Press

Table 1. Technical parameters of the MPIfR S-band receivers

Number of receivers	64 production + 2 prototype (spares)
Rx band	1.75 – 3.5 GHz
Digitizer bandwidth	1.75 GHz (12 bit)
Output bandwidth	875 MHz (see text)
Output bands	1.75 – 2.62 GHz, 1.96 – 2.84 GHz, 2.19 – 3.06 GHz 2.40 – 3.28 GHz, 2.62 – 3.50 GHz
Packetiser interface	SPEAD over 40 GbE
Stability	$>$ 1000 sec. spec Allan variance (over 1 MHz bandwidth)
Sensitivity	$T_{\rm sys}$ <25 K SEFD $<$ 500 Jy
Polarisation	H & V, IXR $<$ 20 dB

S-band reduces the effects of cold plasma dispersion and interstellar scattering. This makes it an important frequency range for both high-precision pulsar timing, where the aforementioned propagation effects act to deteriorate the sensitivity of pulsar timing arrays to gravitational waves (Cordes *et al.*, 2016), and for millisecond pulsar searches deep in the Galactic plane. Table 1 outlines the key parameters of the S-band receiver system.

The S-band receivers adhere to MeerKAT's interfaces and, when roll-out is complete, will be installed on all 64 antennas. In order to reduce problems associated with transmission loss and dynamic range compression, digitisation happens at the receiver. Digitised data flows from the receiver to a packetiser that is located in the antenna pedestal where it is converted into a SPEAD-compliant form before being sent via 40 GbE to the main MeerKAT data network. As can be seen from Table 1 the output bandwidth of the packetiser is currently capped at 875 MHz. This constraint is imposed by the processing capabilities of the MeerKAT F-engine. It is expected that this constraint will be removed as MeerKAT development continues, allowing the full bandwidth of the receiver to be exploited. To handle this constraint, the current packetiser implementation includes a digital band selection filter allowing the usable bandwidth to be centred at 2.2, 2.4, 2.6, 2.8 or 3.1 GHz.

3. Total Power Beamformer

The primary science driver of the S-band project is to enable MeerKAT to perform large-scale pulsar and fast transient surveys. For such surveys to be efficient we must utilise as much of the primary-beam FoV as possible, necessitating the development of a high-performance beamformer. In beamforming, the voltage signals from each antenna in the array are coherently added to produce a tied-array beam on the sky. By appropriately delaying the signals from each antenna prior to addition, the tied-array beam can be steered to arbitrary positions on the sky. As the diameter of the tied-array beam is inversely proportional to the distance between antennas being summed, it is necessary to produce many beams to cover the FoV of the primary beam. Assuming a central frequency of 2.2 GHz and use of only the MeerKAT core (the inner 1km of the array) it would require $\sim$ 5500 tied-array beams to fully cover the primary beam.

To support multi-beam beamforming on MeerKAT we will deploy a 32-node dedicated GPU cluster. This cluster connects to the primary MeerKAT data network via 64 × 40 GbE interfaces allowing it to ingest the full data rate output by the MeerKAT wide-band channelisers ($\sim$ 1.9 Tb/s). Due to the flexible configuration of the MeerKAT network, the beamformer will be able to process data from any of the receivers that are serviced by

https://doi.org/10.1017/S1743921317009036 Published online by Cambridge University Press

MeerKAT's wide-band channelisers. At the time of writing the cluster nodes are expected be dual-socket with 2 × 10-core CPUs, 2 TB SSD storage, 256 – 512 GB RAM, 2 GPUs and 2 × 40 GbE single-port NICs. The exact node configuration will only be known after completion of the tendering process in the coming months.

In terms of operation, each cluster node will ingest $\sim$ 29 Gb/s per NIC. This corresponds to a portion of the full 875-MHz bandwidth for all antennas and both polarisations. The amount of bandwidth processed by each node is dependent on the number of antennas in the current subarray. In line with the majority of MeerKAT software applications that communicate with the data network, we use the SPEAD2[1] library to achieve high performance. Data captured with the SPEAD2 library are stored in PSR-DADA[2] ring buffers from where they can be copied to the GPUs for beamforming, total power detection and integration. Here the ring buffers have a dual-purpose, providing both a robust means of inter-process communication and a large transient buffer capable of storing up to $\sim$ 30 seconds of raw channeliser data. The cluster will respond to external transient triggers, rapidly dumping portions of raw data for subsequent imaging, coherent dedispersion and polarisation analysis.

In order to maximise the number of beams that the cluster will be capable of producing, development efforts were undertaken to produce an optimal beamforming algorithm for GPUs. This resulted in the production of Beanfarmer[3], which leverages the DP4A hardware instruction on Nvidia GP102, 104 and 106 GPUs to produce an algorithm capable of 16 Top/s of beamforming performance on a single GPU. The DP4A instruction works by fusing four int8 multiply adds into a single hardware instruction, allowing for roughly a factor of four performance improvement over a floating point beamformer. Assuming that the beamformer cluster has to run at 85% of real time (i.e. leaving 15% for management and preprocessing overheads) then we may use Beanfarmer to determine the number of beams that the cluster will be able to produce. Using an Nvidia Titan X Pascal GPU, and assuming we are using the full bandwidth and that we integrate 16 samples before outputting a beam, we find that Beanfarmer is capable of producing up to 1760 beams when using a 32-antenna subarray and up to 960 beams when using the complete 64-antenna array. Beams produced by the cluster will be routed back into the MeerKAT data network where the primary consumers will be a pulsar search cluster (see below) and also a transient search cluster that is part of the MeerTRAP project (ERC Advanced Grant; PI B.W.Stappers).

4. Pulsar Search Cluster

With a beamformer capable of producing such a large complement of beams, we require a correspondingly powerful instrument to analyse those beams. To fill this role we have designed a 59-node, multi-purpose GPU cluster. Here eight nodes, each with one 40-GbE NIC and one 56-Gb/s InfiniBand (IB) adapter, will be dedicated to ingest of tied-array beams from the MeerKAT data network (again using the SPEAD2 and PSR-DADA libraries). Data arriving at these nodes will be written to a 2.2-PB distributed file system via an IB cluster interconnect. The remaining 51 nodes are intended to operate as non-real-time workers and will be outfitted with 2 × 10-core CPUs, 128 GB RAM, an IB adapter and 2 GPUs. At the peak ingest rate of the cluster ($\sim$ 280 Gb/s) we will completely fill the 2.2-PB file system in less than 18 hours of observing. As such, it

[1] `https://github.com/ska-sa/spead2`
[2] `http://psrdada.sourceforge.net/`
[3] `https://github.com/ewanbarr/beanfarmer`

https://doi.org/10.1017/S1743921317009036 Published online by Cambridge University Press

necessary to rapidly process and delete beams in order to provide space for new observations. This in turn means that the cluster must be highly-available so that observing time is not wasted. To support this paradigm we will rely on existing, well-supported, well-tested open source software including Docker, Kubernetes and BeeGFS[4]. Through the combination of these software packages the cluster will behave like a cloud service, with high-level abstraction of the cluster resources and strict process isolation.

To support pulsar searching on the cluster we will deploy the prototype SKA pulsar searching stack: Panda, Cheetah and AstroAccelerate [5]. These C++/CUDA applications support the composition of complex asynchronous processing pipelines for pulsar and transient searching. Furthermore, due to the strict budget and power constraints on the SKA, these codes have been extensively engineered for performance. Of particular importance here is the implementation of a Fourier-domain acceleration search in the AstroAccelerate library. Dimoudi *et al.* (2017) have shown that a 2^{23}-point time series with 64-μs sampling can be searched for pulsed signals smeared over up to 200 Fourier bins in $\sim$ 240 ms on an Nvidia GTX 1080 Ti GPU. Using this number we can make a rough estimate of the parameter space we may be able to probe. Assuming that the beamformer is producing 1024 beams and we have $\sim$ 10-minute observations, an 18-hour observing session will result in $\sim 110,000$ individual beams to process. If we further assume that we have five days between observing sessions and 102 GPUs available for processing, then we find that it is theoretically possible to process $\sim$ 1300 dispersion trials per beam with the full acceleration range in 80% of real time (the other 20% is assumed to be used for other processing steps).

5. Timeline

At the time of writing we have just been granted permission to go to tender and expect to begin purchasing equipment in January 2018. Deployment of the system will begin in March 2018 with commissioning activities to begin in April. Science operations are expected to start post-July 2018.

References

Cordes, J. M. & Shannon, R. M., Stinebring D. R. 2016, *ApJ*, 817, 1

Dimoudi, S., Adamek, K., Thiagaraj, P., Karastergiou, A., & Armour, W. 2017, *ApJS*, manuscript submitted for publication.

Kramer, M., Menten, K., Barr, E. D., Karuppusamy, R., Kasemann, C., Klein, B., Ros, E., & Wieching, G. 2016, *PoS*, in proceedings of "MeerKAT Science: On the Pathway to the SKA", manuscript submitted for publication.

[4] https://www.docker.com/, https://kubernetes.io/, https://www.beegfs.io/

[5] https://gitlab.com/SKA-TDT/, https://github.com/AstroAccelerateOrg/astro-accelerate

https://doi.org/10.1017/S1743921317009036 Published online by Cambridge University Press