Optimization of on-chip memory for coarse delay correction

Kaushal D. Buch, Sanjay Kudale and Sandeep Chaudhuri

Digital Backend Group, Giant Metrewave Radio Telescope (GMRT), NCRA-TIFR

1. Delay correction in radio telescope

Astronomical signals received by different antennas undergo different geometric and instrumental delays [1]. These need to be corrected before correlation, in order to get the actual spatial coherence of electric field distribution of the source [1]. This delay correction is usually carried out in the digital sub-system of the telescope. The delay is divided into two main components – coarse delay which is integral multiple of the sampling clock and fine or fractional delay is a sub-multiple of the sampling clock. For an FX correlator, currently planned for the GMRT upgrade, coarse delay correction is carried out in the time domain, whereas fine delay is correction in the frequency domain (i.e. post-FFT). Also note that for a super-heterodyne type of radio telescope receiver system, fringe stopping is required (GMRT upgrade will correct for it in the frequency domain). In frequency domain, this is a broad-band phase correction applied along with fine delay.

In this note, we focus only on the coarse delay correction and technique for optimization of hardware resources required for this. Thus, coarse delay comprises of the instrumental delay (corrected during calibration), transport delay (delay in the optical cable based on the distance from the reference antenna) and geometric delay (change in the path length of the signal depending on the array geometry and source co-ordinates, with respect to a reference antenna).

2. Coarse delay correction on ROACH-1 board

ROACH-1 board, designed by the CASPER collaboration, is widely used in the digital backend of modern-day radio telescopes. Main computing element on the board is the Xilinx Virtex-5 FPGA (XC5VSX95T -1 FF1136). This FPGA possesses on-chip memory (Block RAM or BRAM) for buffering data.

Coarse delay correction in hardware can be thought of as an offset between the write and read pointers of a memory [2]. Thus, a dual-port memory (Block RAM) is used to achieve this. The depth of the memory required depends on the delay and sampling clock. Conventionally, the correlators used the maximum delay (calculated for the worst case delay and source coordinates).

On Xilinx FPGA, each BRAM is a block of 2048 x 8 bits (actually it is 2048 x 9 bits). Hence, the BRAM utilization would be a multiple of this number. Also, current high speed hardware implementations used a poly-phase approach, whereby a signal sampled at a higher rate is processed at a lower rate using parallel hardware. Thus the FPGA receives N parallel channels of data at 1/N times the sampling clock. As this parallel data belongs to a single instance of time, the

delay correction has to spread across this N channels. Hence, each channel will be correcting a time delay of t_c/N, where t_c is the total coarse delay. Hence, the total minimum BRAM usage for this case would be 'N'.

3. Optimization procedure

F-engine of packetized correlator or a single-board correlator (Pocket Correlator) processes at least two antennas. As discussed in the earlier section, coarse delay correction directly impacts the BRAM utilization on an FPGA. As it turns out, if we consider the worst case delay (for the farthest antenna), BRAM availability may become a bottle-neck, not only restricting the increase in the sampling clock rate, but also effecting the use of other memory intensive blocks like FFT, PFB etc. One significant impact is the restriction it poses on the number of FFT channels. This is because the FFT block utilizes BRAM for buffering data.

For a radio telescope array, the delays are calculated with respect to a reference antenna. Usually, there are antennas which are relatively near (lesser delay) to the reference antenna. If a combination of such antennas is made with those antennas which are farther away (more delay) on one FPGA (ROACH board) then the BRAM requirement will reduce significantly. Thus, for a given array geometry and set of worst case delays corresponding to each of the array antennas, an optimal combination of large and small delay can be made to reduce the BRAM utilization.

The optimal combination can be computed manually or through some algorithm. A simple way of computing would be to list the delays on these antennas into two arrays. Sort one of the arrays in ascending order and the other in descending order. Selecting corresponding elements from these two arrays would provide an optimal combination. This sorting can be done either based on delay or an equivalent memory requirement.

4. Case Study for the GMRT array

GMRT array consists of 30 antennas, 12 antennas form the compact array and the other 18 are distributed in a Y-shaped geometry in east, west and south directions (each arm has six antennas). The reference antenna is C02. With reference to this antenna, the following table provides the worst case delays for the all the antennas of the array.

Note: The delay values may change if the reference antenna is changed.

Antenna	Positive maximum delay (sec) with respect to C02 antenna	Negative maximum delay (sec) with respect to C02 antenna	Absolute delay (sec) with respect to C02 antenna	Number of Block RAMs required (sampling freq = 800 MHz)
C00	7.29279E-07	-3.8599E-06	4.5892E-06	2
C01	1.75212E-06	-4.2742E-07	2.1795E-06	1
C02	2.03889E-06	2.03889E-06	4.0778E-06	2

C03	2.95433E-06	2.92341E-07	3.2467E-06	2
C04	2.10866E-06	-1.7656E-06	3.8742E-06	2
C05	9.78764E-07	-8.0678E-07	1.7855E-06	1
C06	9.44403E-08	-1.461E-06	1.5554E-06	1
C08	1.7744E-07	-3.1977E-06	3.3751E-06	2
C09	1.44088E-06	3.42536E-07	1.7834E-06	1
C10	6.26651E-07	-3.639E-06	4.2657E-06	2
C11	1.09159E-06	-3.5006E-06	4.5922E-06	2
C12	-1.1671E-06	-5.7803E-06	6.9474E-06	4
C13	-9.3276E-08	-8.9836E-06	9.0768E-06	4
C14	1.31147E-06	-4.1187E-06	5.4302E-06	4
W01	8.6441E-07	-1.0533E-05	1.1398E-05	8
W02	-4.4385E-06	-2.7255E-05	3.1693E-05	16
W03	-1.2721E-05	-5.2875E-05	6.5596E-05	32
W04	-2.227E-05	-8.1242E-05	0.00010351	64
W05	-3.4422E-05	-0.00011129	0.00014571	64
W06	-5.0765E-05	-0.00014842	0.00019919	128
E02	-3.1703E-06	-2.3129E-05	2.6299E-05	16
E03	-9.7484E-06	-4.3099E-05	5.2847E-05	32
E04	-1.9023E-05	-7.48E-05	9.3822E-05	64
E05	-3.318E-05	-0.00010522	0.0001384	64
E06	-4.1625E-05	-0.00012822	0.00016985	128
S01	-8.4366E-06	-2.8619E-05	3.7056E-05	16
S02	-1.6137E-05	-4.6279E-05	6.2416E-05	32
S03	-2.4353E-05	-6.9378E-05	9.3732E-05	64
S04	-3.1273E-05	-9.4717E-05	0.00012599	64
S06	-4.717E-05	-0.00014122	0.00018839	128

The above mentioned table shows the delay values and maximum number of BRAM (each BRAM is 2048 * 8 bits) required for each antenna. The table below shows a concise summary of BRAM utilization for different approaches and also indicates the saving in the number of BRAM.

Antenna	**Number of Block RAMs required (sampling freq = 800 MHz)**	**Number of Block RAMs required for poly-phase (by 4) architecture**	**Number of Block RAMs required (worst case delay)**
C00	2	4	128
C01	1	4	128
C02	2	4	128
C03	2	4	128
C04	2	4	128
C05	1	4	128
C06	1	4	128

C08	2	4	128
C09	1	4	128
C10	2	4	128
C11	2	4	128
C12	4	4	128
C13	4	4	128
C14	4	4	128
W01	8	8	128
W02	16	16	128
W03	32	32	128
W04	64	64	128
W05	64	64	128
W06	128	128	128
E02	16	16	128
E03	32	32	128
E04	64	64	128
E05	64	64	128
E06	128	128	128
S01	16	16	128
S02	32	32	128
S03	64	64	128
S04	64	64	128
S06	128	128	128

In order to optimize on the requirement of BRAM, sort the array into ascending and descending order and then combine the elements of the array. The following is the optimal delay antenna combination in terms of BRAM usage. Hence, based on table below, if the antennas are combined on the FPGA based on their delay requirements, a significant saving in BRAM utilization can be achieved.

Antenna Pair	**Number of Block RAMs* required for poly-phase (by 4) architecture**	**Number of Block RAMs* required (worst case delay)**
C00 – W06	4 +128 = 136	128 + 128 = 256
C01 – E06	4 +128 = 136	128 + 128 = 256
C02 – S06	4 +128 = 136	128 + 128 = 256
C03 – W04	4 + 64 = 68	128 + 128 = 256
C04 – W05	4 + 64 = 68	128 + 128 = 256
C05 – E04	4 + 64 = 68	128 + 128 = 256
C06 – E05	4 + 64 = 68	128 + 128 = 256
C08 – S03	4 + 64 = 68	128 + 128 = 256
C09 – S04	4 + 64 = 68	128 + 128 = 256
C10 – W03	4 + 32 = 36	128 + 128 = 256
C11 – E03	4 + 32 = 36	128 + 128 = 256
C12 – S02	4 + 32 = 36	128 + 128 = 256
C13 – W02	4 + 16 = 20	128 + 128 = 256

C14 – E02	4 + 16 = 20	128 + 128 = 256
W01 – S01	8 + 16 = 20	128 + 128 = 256

*considering 2048 *9 bit or 18kbit BRAM

The above mentioned table shows the BRAM utilization on FPGA with the optimal delay combination. However, in order to keep design generic, we choose to use the maximum delay i.e. using 136 BRAMs. Hence, the absolute saving in BRAMs is 256 – 136 = 120 BRAMs.

This saving in BRAMs is useful for increasing the spectral resolution (i.e. no. of FFT points) of the correlator. We tried a packetized correlator F-engine having worst case delay in both the channels and compared it with F-engine having an optimal delay combination. The implementation results shown below show that with this optimization, we can fit double the number of spectral channels (i.e. 4096).

No. of Spectral Channels	**Number of FFT points**	**BRAM usage**	**Coarse delay**
2048	4096	179/244 (73%)	Chan-1 = 256K Chan-2 = 8K
4096	8192	208/244 (85%)	Chan-1 = 256K Chan-2 = 8K
4096	8192	323/244 (132%) **Over-mapped**	Chan-1 = 256K Chan-2 = 256K

5. Implications on the existing packetized design

The current F-engine would need a design modification and corresponding changes in the delay script. X-engine will not need any changes.

The proposed scheme would pose some restrictions on the antennas that could be combined on a ROACH board. Also, as we now combine two antennas (single polarization) per ROACH board, hence, a board failure would lead to loss of one polarization from both the antennas.

6. Conclusion

A method for optimization of on-chip memory for coarse delay correction was described. The approach followed by a case study for the GMRT array showed that there is a significant saving in the BRAM utilization. Also, it was shown that this saving helps in doubling the number of spectral channels for a correlator.

7. Acknowledgements

The authors would like to thank Ms. Mekhala Muley for reviewing this note.

8. References

Jayaram N. Chengalur, “Delay Tracking and Fringe Stopping for the GMRT correlator”, NCRA Memo no. R0177, 18th October, 1998.

Jayaram N. Chengalur, Yashwant Gupta and K.S. Dwarkanath (eds.), “Low Frequency Radio Astronomy”, NCRA-TIFR.