2019 IEEE 13th International Symposium on Embedded Multicore/Many-core Systems-on-Chip (MCSoC)

A Preliminary Evaluation of Building Block Computing Systems

Sayaka Terashima, Takuya Kojima, Hayate Okuhara, Kazusa Musha, Hideharu Amano
Dept. of Information and Computer Science, Keio University, Japan
Email: wasmii@am.ics.keio.ac.jp
Ryuichi Sakamoto, Masaaki Kondo
Dept. of Information Science and Technology, The University of Tokyo, Japan
Mitaro Namiki
Graduate School of Technology, Tokyo University of Agriculture and Technology, Japan

***Abstract*—A building block computing system with inductive coupling Through Chip Interface (TCI) consists of 3-D chip stack, each of which is small dedicated chips. By changing the combination of stacked chips, various types of systems can be built. A MIPS R3000 compatible processor GeyserTT, a neural network accelerator SNACC and the shared memory for building the twin-tower of chips SMTT have been developed with a Renesas 65nm low leakage CMOS process. They provide the TCI IP (Intellectual Property), and an escalator network is built just by stacking them. This paper shows each chip evaluation results and performance estimation of stacking them with the RTL simulator. The performance of the single-tower and twin-tower configuration is estimated by RTL simulation when a part of Alexnet is implemented. The evaluation results showed that the single-tower configuration with GeyserTT+SNACC achieved about twice performance as the case with GeyserTT. Also, experimental results using each of the single real chip showed that all of them work at least 50MHz with extremely low power consumption. The twin-tower configuration achieved about 2x of the single-tower, that is about 6x of GeyserTT. The power consumption was about 276mW for the single-tower and 496mW for the twin-tower.**

I. Introduction

The increasing requirements for IT devices; various functions, high performance, and low energy, make it difficult to be satisfied with a single universal SoC (System-On-a-Chip). However, increasing NRE (Non-Recurrent Engineering) cost also makes it difficult to develop various types of SoCs for each application. An approach using SiP (System in Package) technologies is hopeful because various types of systems can be built from combinations of dedicated chips; processors, memory modules and accelerators. Wireless inductive coupling Through-chip Interface (TCI)[1] is a flexible SiP technology which enables three-dimensionally chip stacking with much smaller cost than TSV (Through Silicon Via).

Our research project aims to establish techniques for building a large system by combining multiple chips with the TCI like LEGO blocks.[2]. We call them a building block computing system. So far, we have developed an intellectual property (IP) of TCI and embedded into some test processor (or memory accelerator) chips: a MIPS R3000 compatible processor GeyserTT, CNN(Convolutional Neural Network) accelerator SNACC[3], a coarse-grained reconfigurable processor CC-SOTB2[4], a non-SQL database accelerator KVS chip[5], and a shared memory chip SMTT[6].

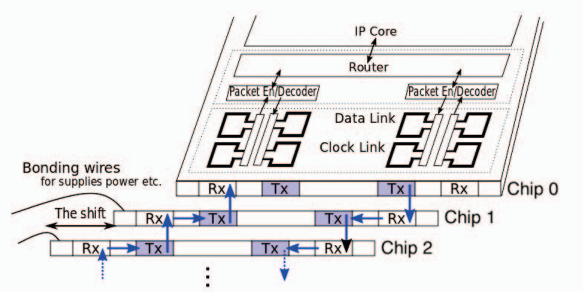

Fig. 1: 3D NoC using TCI

As a preliminary evaluation, in this paper, we focus on three chips: GeyserTT, SNACC, and SMTT. First, we will show each real chip evaluation, and based on the results, a simulation study is done for implementing CNN application on a chip stack with their combination in order to demonstrate scalability of the building block computing system.

The rest of the paper is organized as follows. First, a building computing system is introduced, and some of the chips used in our systems in Section II. Then, the design of SMTT and the architecture, a twin-tower system with SMTT are shown in Section III. Section IV presents execution time evaluation and results about one of CNN applications. Finally, we conclude in Section V.

II. Building Block Computing Systems

Building block computing systems enable to construct a scalable 3D stacked VLSIs by combining various types of chips: CPU, accelerators, and memory modules. Inductive coupling TCI is a key technology for inter-chip communication in this system.

A. Inductive Wireless Through Chip Interface

TCI is equipped with square coils implemented by general metal layers for building a data communication link. As shown

978-1-7281-4882-3/19/$31.00 ©2019 IEEE
DOI 10.1109/MCSoC.2019.00051

in Fig. 1, the transmitter coils are placed just above receivers' ones, and data are transferred between them through the magnetic field. Here, Tx is a transmission channel, and Rx is a receiver channel. A TCI link needs two inductors, one for a clock signal and the other for the data. The transferred data is synchronized with a high speed clock signal (1-8GHz) generated by an internal VCO.

TCI has the following advantages. First, the inductor consists of common wires in the CMOS process technology without the particular process technology, unlike the TSV. Also, ESD (Electro Static Discharge) protection device is unnecessary, since TCI is electrically contact-less. Moreover, a data transfer rate of more than 8Gbps is possible with power less than 10mW and low bit-error rate (BER $< 10^{-12}$)[7]. Thus, data correction code is not required.

Although the TCI requires a large footprint of the coil, digital circuits can be located inside it. Only the metal layers used for the inductor is actually unavailable. We need to make each chip thin for increasing transfer efficiency so that the strength of the magnetic field depends on the distance between the transceiver and the receiver. Now, we make the chip thickness 40μm to 80μm in order to reduce the size of the coil. In this work, low power LSIs are stacked with the TCI so that heat dissipation does not matter much.

B. IP (Intellectual Property) of TCI

We developed IPs (Intellectual Properties) on Renesas 65nm SOTB process supported by VDEC. TCI IP consists of coils, transmitter, receiver, and SERDES (Serializer/De-serializer). As shown in Fig. 2, the coils for the data and clock signals are realized by duplex winding for its transceiver and receiver, which allows to switch the communication direction of the link within a few clock cycles. The internal VCO's frequency is designed to be 2.5GHz. Hence, 35-bit data can be transferred at 50MHz of the operational frequency. In other words, this IP can be treated as a simple 35-bit uni-directional registered channel. The diameter of each coil is 240μm × 240μm to build a link between the chip with 80μm thickness. The size of the entire IP is 510μm × 410.8μm. We also developed the link layer and the router layer on the physical layer, and an escalator network can be formed just by stacking chips like Fig. 1 [8]. It is important to mention that the link direction of each the TCI IP have to be fixed to one direction to form the escalator network.

We developed three chips implemented with the TCI IP. All of them uses Renesas 65nm SOTB process, and designed with the same design environment described in Table I.

C. GeyserTT: a host chip

GeyserTT (Geyser for Twin Tower) is an embedded CPU for host processor of the stacked chip. As shown in Fig. 3, it is composed of three TCI IPs, Geyser CPU core, DMAC, and External Bus Controller. Geyser core is a MIPS R3000 compatible CPU with 8KB two-way set associative cache for instruction and data. It also has an integrated TLB with 16-entry. An embedded operating system TOPPERS[9] is working on it. It transfers data between data cache and local memory of other stacked chips logically mapped onto the same address space through the TCI IPs. The embedded DMAC manages block data transfer between a data cache block and the memory of stacked chips. Since GeyserTT is located at the top of the chip stack, it has the TCI IPs only for the down direction. As shown in Fig. 4, GeyserTT has three down direction TCI IPs so that three types of chip stacking can be constructed. However, this paper employs only the simple 3D stacking. GeyserTT also manages I/O of the chip stack.

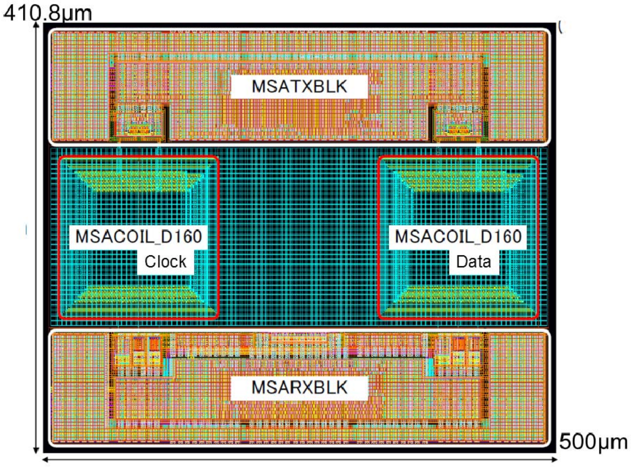

Fig. 2: The layout of TCI IP

TABLE I: Spec. of family chips

GeyserTT	MIPS R3000 host processor
SNACC	Neural Network Accelerator
SMTT	Shared Memory for Twin Tower
Process	Renesas 65nm DLSOTB_V3 CMOS 7 Metal
Area	3mm × 6mm (GeyserTT, SNACC) 6mm × 6mm (SMTT)
Chip Thickness	80μm
Target Freq.	50MHz (GyserTT, SNACC) 100MHz (SMTT)
TCI IP	35bit/50MHz
CAD	Synopsys Design Compiler 2016.03-SP4 Synopsys IC Compiler 2016.03-SP4

D. SNACC: a neural network accelerator

SNACC (Scalable Neural Acceleration Cores with Cubic integration)[3] is an accelerator chip for CNN accelerations. It consists of four SIMD cores, implemented its original instruction set and local memory designed for CNN. The instruction width is 16-bits, and 16 general purpose registers are provided. Fig. 5 shows the schematic diagram of the local memory configuration of the SNACC. Each core has five memory modules, INST, DATA, RBUF, LUT, and WBUF, for instruction codes, input data, weight data, a lookup table,

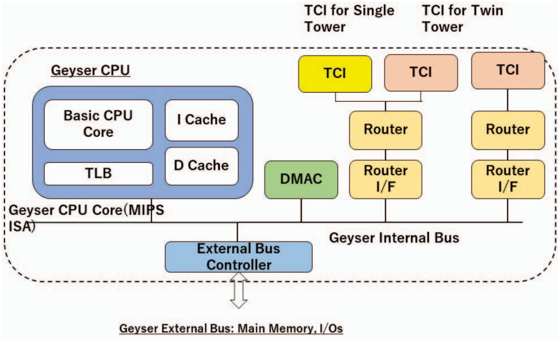

Fig. 3: The overview of GeyserTT

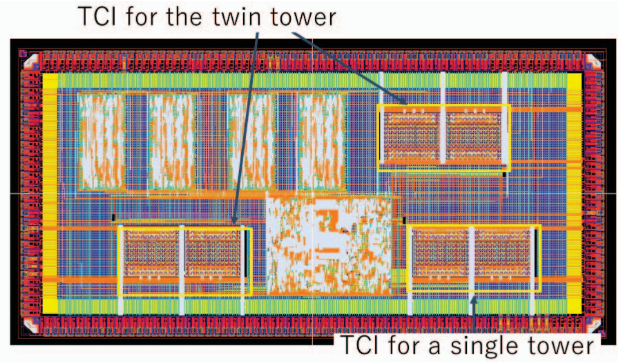

Fig. 4: The layout of GeyserTT

and output data respectively. DATA and WBUF are double-buffered so that the data transfer and processing can be overlapped. Each core including four local memories has an independent address space except WBUF. The address space of WBUF is shared with four cores. In this way, the computation result for each core can be shared.

The instruction set of the SNACC core consists of R-type (register-register) and I-type (register immediate) instructions. However, unlike the standard 32-bit RISC instruction set architecture, only two operands are specified. One of the biggest features of SNACC is SIMD instructions which perform mad (multiply add) instruction and madlp (multiply add with loop) instruction. Fig. 6 shows the SIMD unit for the mad instruction. The SIMD unit can handle fixed-point arithmetic four 16-bits data or eight 8-bits data. Each multiplier unit receives two input data from the DATA and RBUF memory, and then, these products are summed by an adder unit. The Max unit calculates the maximum value from all the inputs. Output data from the adder unit and the max unit is chosen by a multiplexer and is stored in the register *r13*. In addition, an activation function defined by the lookup table is applied to output data from the adder, and the result is stored in the register *r11*. The madlp instruction iterates this mad instruction a specified number of times. For control the SIMD instructions, each core provides eight 8-bits control registers, and four 32-bits SIMD registers.

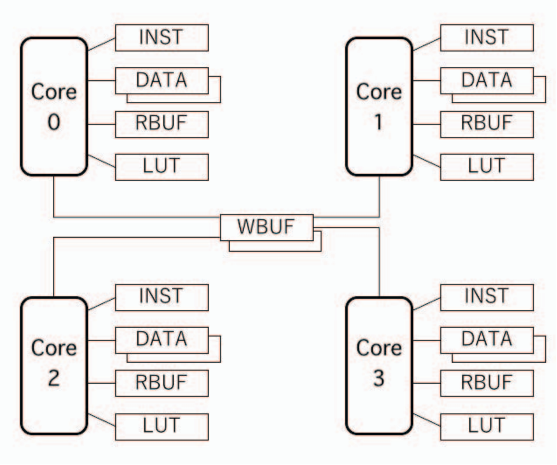

Fig. 5: The overview of SNACC with local memory modules

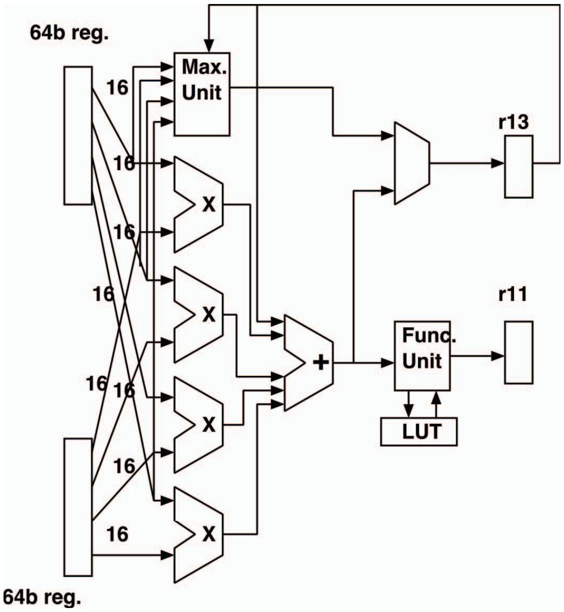

Fig. 6: SIMD unit for product-sum operation

Fig. 7 shows the layout of SNACC. It uses 36 small memory IPs for various types of memory modules. The four cores are implemented in four rectangle squares in the chip. Note that, SNACC has 4 of the TCI IPs for the up/down link of the escalator network.

E. SMTT (Shared Memory for Twin Tower)

SMTT is a shared memory chip for building block computing systems with TCI. SMTT has 32KB × 4 SRAM modules, and they are divided into eight banks, each of which can be accessed independently. The most important property of the SMTT is allowing the twin tower chip stacking structure.

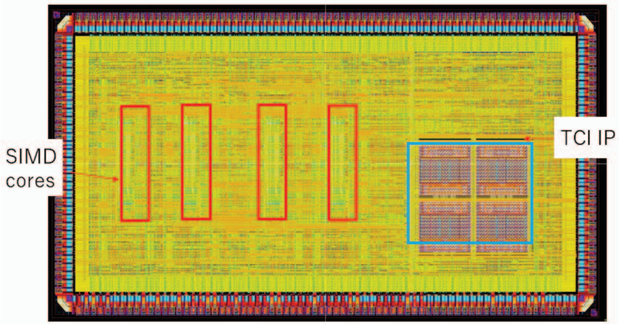

Fig. 7: The layout of SNACC

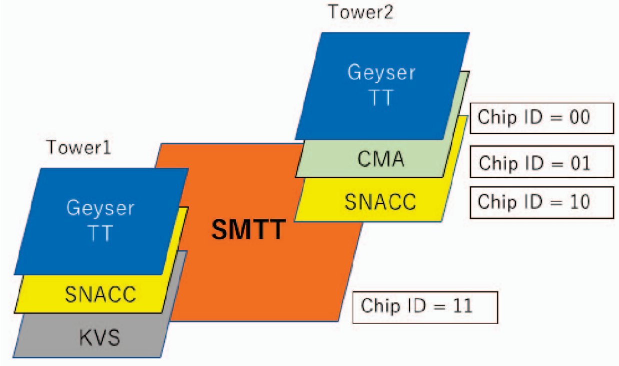

Fig. 8: Overview of twin tower system using SMTT

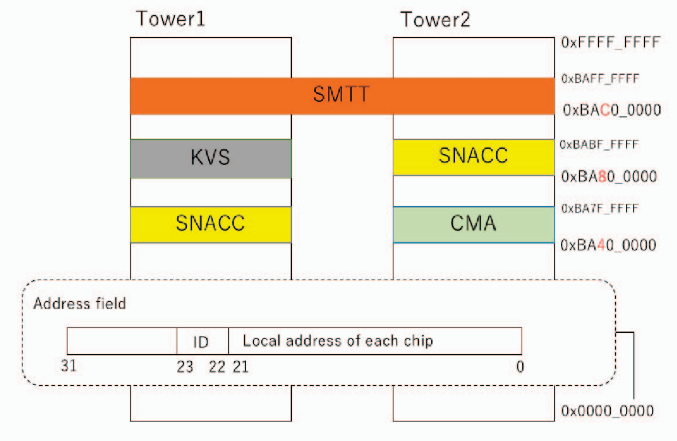

Fig. 9: Address map of the twin tower system

Two different chip stacks can be built on the SMTT chip as shown in Fig. 8. The memory on the SMTT is shared with chips in both towers. Thus, the SMTT behaves as a hub chip of two towers. The top of each tower is GeyserTT, and the bottom of them is SMTT in any case. We can insert any accelerators including SNACC between the SMTT and GeyserTT.

The address of each tower chip is mapped into the address space of GeyserTT as illustrated in Fig. 9. All chips in the same tower can access memory modules on the SMTT assigned into the same address space of the memory map, although the address spaces of each tower are independent. Since the memory on the SMTT is 8-way interleaved, chips in both towers can access them simultaneously if access conflicts do not occur. An arbiter is provided for the exclusive access control from both chip stacks.

Chip IDs are assigned to the stacked chips according to the stacking order, and GeyserTT identifies the chips by using the IDs. The lower 22-bit of the address are local addresses of each stacking chip, and upper 8-bit indicates GeyserTT virtual address space identifier. A host CPU GeyserTT can access the chips of own tower by just loading and storing the assigned address. A DMA transfer between accelerators of the same tower is available. Accelerators in the different towers share the data through the SMTT.

An atomic operation Fetch&Dec is usually employed for synchronization among multiple processors with a shared variable. The SMTT has 32 synchronous registers and, they are mapped to the same address in both towers. When a chip reads the synchronous register, then its value is decremented atomically whereas writing to it is executed normally. If the result is zero, it is never decremented anymore. In this way, it is easy to implement a counting semaphore and barrier synchronization with the synchronous registers.

Fig. 10 shows the layout of SMTT. Unlike GeyserTT and SNACC, it is implemented on 6mm × 6mm chip die. We used 32KB single port SRAM IP as each memory bank of SMTT, for forming in 256KB memory in total. On the top right and lower left on this chip, there are two TCI IPs, and we can form two chip towers on it. Controllers and synchronous registers other than IP of SRAM and TCI are widely spreading over the whole chip. Also, Fig. 11 shows the photograph of chip stacking with the SMTT and the GeyserTT.

III. Real Chip Evaluation

First of all, we examined the operation of each chip, and confirmed that all of them work well. The chip photo of SNACC-SMTT is shown in Fig. 12. Before building three-chip stack GeyserTT+SNACC+SMTT, two-chip stacks were developed and tested. GeyserTT+SNACC stack is available, while SNACC+SMTT is now under testing.

Here, we show the real chip evaluation results of each chip, and estimate the power consumption of three-chip stack from the result. The performance is estimated with RTL simulation study.

A. Operational Frequency

Renesas SOTB 65nm LSTP process focuses on leakage reduction, and the operational frequency is not so high with 0.75V standard power supply voltage (Vdd). Fig. 13 shows the power consumption versus operational frequency. A simple memory check program is executed on GeyserTT, a simple computational kernel is executed with a core of SNACC, and in the case of SMTT, just writing and reading specific data is iteratively executed. Although Vdd is set to be 0.75V, the forward body bias (VBN) is used to enhance the performance as shown in later. The maximum operational frequency with 0.75V Vdd is 52MHz for GeyserTT, 60MHz for SNACC and 130MHz for SMTT with the forward body bias. Each chip except the SMTT, a simple memory chip was designed to work with 50MHz, the 35-bit transfer rate of TCI. Thus, the achieved operational frequency is reasonable considering the complexity of each chip.

B. Power Consumption of each chip

One of the benefits of the SOTB (Silicon on Thin BOX) process is high controllability of body biasing. There are a lot of studies to make the use of body biasing for the optimization of power consumption[10]. However, recent LSTP process shifts to reducing leakage power by increasing the threshold of each transistor. Even with the zero bias, which gives the same voltage as the GND (VBN=0) to NMOS and Vdd (VBP=Vdd) to PMOS, the leakage current is quite small. Fig. 14 shows the leakage current versus the body bias to NMOS (VBN). Here, we used the balanced bias, that is (VBP = VDD-VBN), so only values of VBN are shown. It is apparent that the leakage current is well suppressed even with the strong forward biasing. Note that the leakage of SMTT largely increases, since it has relatively large SRAM modules. To generate the body bias voltages, we can employ an on-chip body bias generator, which requires only a few micro-watt of power overhead [11], [12], [13]. However, in this evaluation, they are supplied by off-chip stable power sources.

According to Fig. 13, using the forward biasing is efficient to boost the performance without increasing the power consumption rather than increasing Vdd. GeyserTT consumes around 35mW at 50MHz (target frequency of this design) while SNACC needs only around 3.5mW at the same frequency. Therefore, the power consumption of GeyserTT is about 10 times as that of SNACC. Here, SNACC only uses a core, and when four cores are used, it will become about 4 times. Nevertheless, the power consumption is still much lower than that if GeyserTT. SMTT, a shared memory consumes smaller power than the other two chips. For instance, in case of 50MHz, SMTT can work with only 2mW power consumption. It is shown that when the three-chip stack is used, the total power consumption becomes only about 40mW.

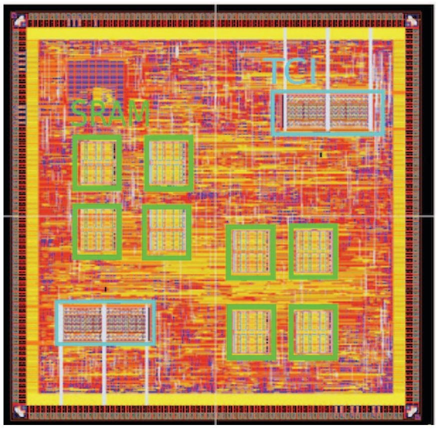

Fig. 10: SMTT chip layout

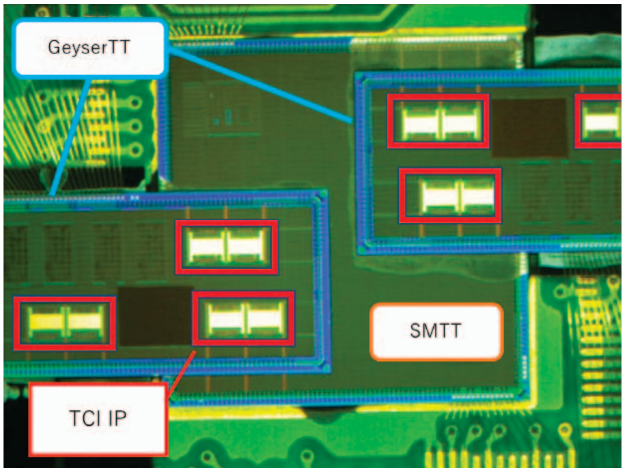

Fig. 11: Chips photograph of stacking chips with TCI IP

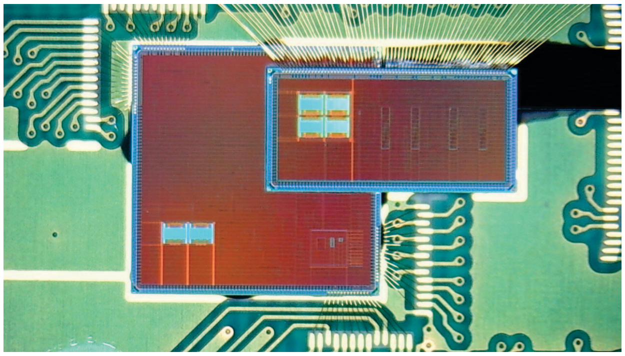

Fig. 12: The chip stack of SNACC+SMTT

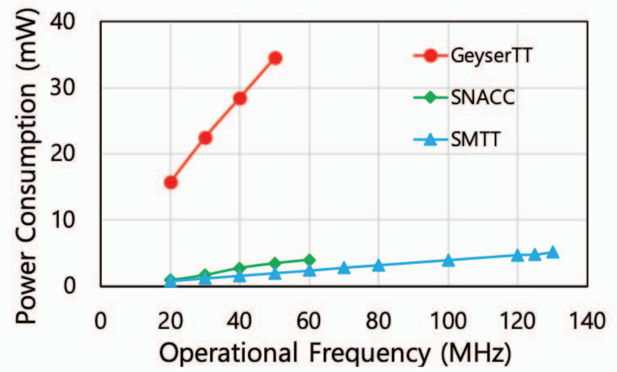

Fig. 13: Power Consumption versus Operational Frequency

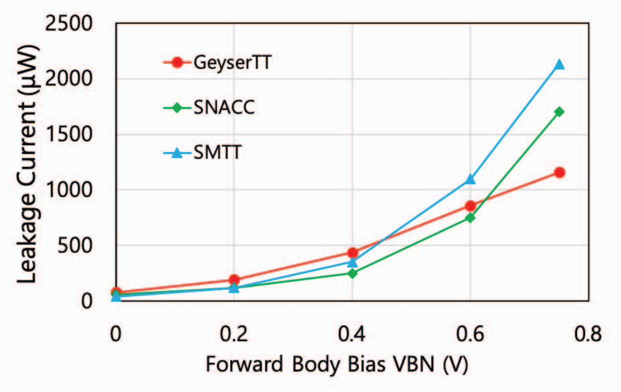

Fig. 14: Leakage Current versus Forward Body Bias VBN

TABLE II: Power for the TCI (per IP)

	Design	Measured
Trans.Volt.(V)	1.2	1.7-2.9
Trans.Power (mW)	19.32	38.9(Max)
Receiv.Volt.(V)	1.2	1.2-1.7
Receiver (mW)	17.0	20.1(Max)

C. Power Consumption of TCI

Table II shows the power consumption of a TCI IP. Note that it includes everything around an IP; transceivers, receivers, and SERDESs for the clock and data links. Through the evaluation, it did not work with the designed power supply voltage (1.2V), and worked with much higher voltage; 1.7V-2.9V depending on the location of the IP. It is hypothesized that resistance on the power grid for TCI IP might degrade the power voltage. This problem can be fixed in the next chip implementation by increasing I/O pad for the power supply, refining the power grid and improving the layout of IP core itself. Fig. 15 shows a breakdown of the power consumption. As GeyserTT+SNACC+SMTT chip stack uses 8 IPs, the dominant factor of the power consumption is apparently TCI. It occupies about 85% of the total power consumption 276mW. The twin-tower configuration with (GeyserTT+SNACC)×2+SMTT consumes about 490mW in total.

IV. Simulation Study

Simulation tools used in this evaluation are summarized in Table III. All parameters in the simulation are based on the real chip implementation. Here, we evaluated the execution time of a CNN application in several building block systems which consist of GeyserTT, SNACC, and SMTT.

TABLE III: Simulation tools

Logic simulation	Cadence NC-Verilog 10.20-s131
Operation clock frequency	50MHz
Power simulation	Synopsys Prime Time 2012.12-SP3

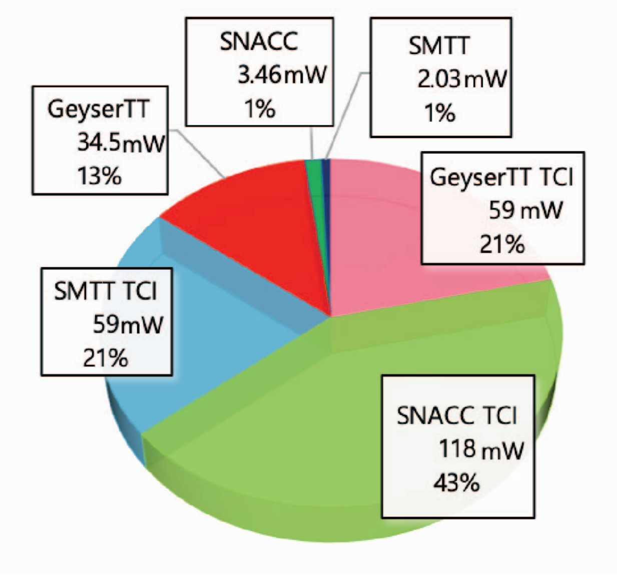

Fig. 15: Power breakdown of the GeyserTT+SNACC+SMTT chipstack

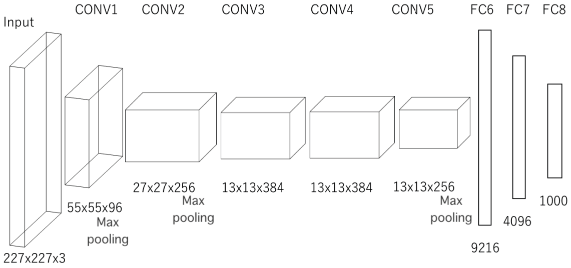

Fig. 16: The architectures of AlexNet

A. Target Application

The CNN adopted here is a type of feedforward neural network mainly for image recognition. A feedforward neural network connects between all to all nodes in general, on the other hand, a CNN connects only locally node depending on a filter called kernel to process more efficiently.

Here, the famous AlexNet[14], a winner of the Image Net 2012[15] was implemented as a target application. Although it is now old-fashioned, it is sufficiently general and used in many studies. Fig. 16 illustrates the architectures of AlexNet. AlexNet consists of five convolutional layers (CONV), three pooling layers (POOL), and three fully-connected layers (FC). It can classify images of 227 × 227 pixels into a thousand of groups.

In this evaluation, we implement the last two fully-connected layers (FC7 and FC8) as a benchmark. FC layers take charge of the final classification of the values extracted from the convolution layer and the pooling layer. In general, comparing with computation capacity, memory bandwidth is the bottleneck for FC layers. On the other hand, convolution layer suffers less from that bottleneck. The proposed system with TCI brings about flexibility of chip stacking whereas it limits the memory bandwidth due to the same design of TCI IP for all chips. Thus, we choose the FC layers to demonstrate scalability of the building block computing system. Applications are written in C program, and compiled with MIPS cross-compiler except for program code for SNACC core. The FC layers is executed with fixed-point operation because GeyserTT and SNACC does not support floating-point instructions. Therefore, the pre-trained parameters are truncated to fixed-point numbers.

The calculation in the FC layer is performed according to the following equation.

$$a[i] = \sum_{j} w[i,j]x[j] + b[i] \qquad (1)$$

Table IV lists the parameter of the FC7.

TABLE IV: Benchmark parameter

layer	input	output	kernel	bias
FC7	4096	4096	(4096, 4096)	4096
FC8	4096	1000	(1000, 4096)	1000

Each node has weight ($w[i,j]$) and bias ($b[i]$) values between all input and output values, and this weight and bias values are different depending on the learning data.

B. Target Hardware

We tried four system configurations: a single GeyserTT, a single-tower: GeyserTT+SNACC, a twin-tower (GeyserTT)×2+SMTT, and a twin-tower with SNACC: (GeyserTT+SNACC)×2+SMTT.

First, we simulated a simple GeyserTT (one core) and GeyserTT (two cores) with SMTT. GeyserTT fetches the input data, weight data, and bias data for the application from an external memory and executes the FC layers. In case of the twin-tower configuration, processing for each layer is divided into two and executed in parallel. After the FC7 layer is calculated, both of GeyserTT share the results for the next FC8 by storing them in the different memory bank of SMTT. Next, we added SNACC chips for CNN acceleration between GeyserTT and SMTT. All four cores in the SNACC are utilized. In this system, GeyserTT transfers the required data and controls the SNACC chip. Program in SNACC cores handles same 16-bit fixed-point number as that of GeyserTT. The FC layers are accelerated with "madlp" instruction. The execution time of the systems using SNACC includes data transfer time between GeyserTT and SNACC. Data transfer and processing are overlapped on a double-buffering basis, For the twin-tower, SNACC (four cores)×2 and GeyserTT×2 are connected by SMTT as a bridge, and the results in the WBUF are transferred to SMTT memory. Thereby, SNACC chips in both tower can share the results like the GeyserTT×2+SMTT configuration.

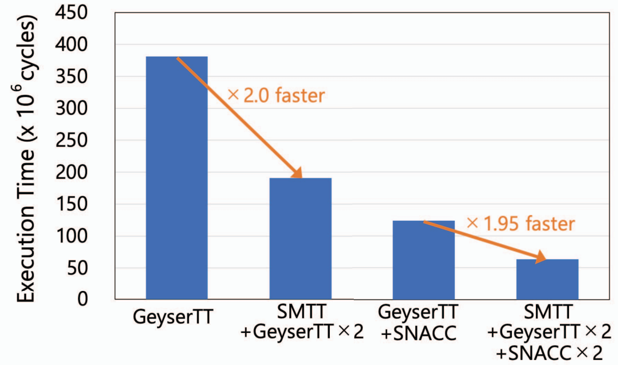

Fig. 17: Simulated execution times of four systems

C. Evaluation Results

The simulation results for each system are shown in Fig. 17. Whereas execution cycles of GeyserTT (one core) is about 3.81×10^8 cycles, GeyserTT (two cores) with SMTT takes only 1.91×10^8 cycles. Although the twin-tower system needs some cycles to synchronize both cores, it causes slight time overhead compared to computation and data transfer time. As a result, 50.0 % time reduction is achieved. The SNACC single-tower without SMTT takes 1.24×10^8 cycles. The result demonstrates that the SIMD instruction improves execution times compared to GeyserTT (one core) system. Besides, the SNACC chips (eight cores) in the twin-tower system consume only 6.34×10^7 cycles. The size of weight data in FC layers is too large for the local memory in SNACC so that GeyserTT have to replace the weight data over and over. The transfer time of input data (1KB) is about 20000 cycles while calculation using the 1KB data takes approximately 1400cycles. Hence, even though the processing and data transfer are overlapped, the transfer time occupies a large proportion of the execution cycles, and it was impossible to shorten the execution time anymore. Consequently, the twin-tower system with SNACC executes the FC layers 1.95× faster than the single-tower system including SNACC.

V. Conclusions

Here, a MIPS R3000 compatible processor GeyserTT, a neural network accelerator SNACC and a shared memory for building the twin-tower of chips SMTT have been developed with a Renesas 65nm CMOS SOTB process. The real chip evaluation result showed that all of them work at least 50MHz with extremely low power consumption by using forward body biasing. The performance of the single-tower and the twin-tower configuration is estimated by RTL simulation when a part of Alexnet is implemented. The evaluation results showed that the single-tower configuration with GeyserTT and

SNACC achieved about twice performance as the case with a single GeyserTT. The twin-tower configuration also achieved around 2× of the single-tower, that is about 6× of the single GeyserTT. The power consumption was about 276mW for the single-tower and 496mW for the twin-tower.

The current problems are mostly around the TCI IP. Some combination of chips are now under testing. Also, the voltage level must be much higher (2.9V) than the designed one (1.2V). We will try other combinations of family chips: KVS and CCSOTB2, and establish the way to use IP stably.

Acknowledgment

This work is partially supported by JSPS KAKENHI S Grant Number 25220002 and JSPS KAKENHI B Grant Number 18H03215. This work is supported by VLSI Design and Education Center(VDEC), the University of Tokyo in collaboration with Synopsys, Inc and Cadence Design Systems, Inc.

References

[1] Y. Take, H. Matsutani, D. Sasaki, M. Koibuch, T. Kuroda, and H. Amano, "3D NoC with Inductive-Coupling Links for Building-Block SiPs," *IEEE Transactions on Computers*, vol. 63, no. 3, pp. 748–763, 2014.

[2] H.Amano, K.Usami, M.Kondo, H.Nakamura, M.Namiki, and H.Matsutani, "Kakenhi-s: A study on building-block computing systems using inductive coupling interconnect," http://www.am.ics.keio.ac.jp/kaken_s/.

[3] R.Sakamoto, R.Takata, J.Ishii, M.Kondo, H.Nakamura, T.Ohkubo, T.Kojima, and H.Amano, "The design and implementation of scalable deep neural network accelerator cores," in *Proc. of IEEE 11th International Symposium on Embedded Multicore/Many-core Systems-on-Chip (MCSoC-17)*, Sep. 2017.

[4] T.Kojima, N.Ando, H.Okuhara, N.A.V.Doan, and H.Amano, "Body Bias Optimization for Variable PipelinedCGRA," in *Proceedings of the Field-Programmable Logic and Applications (FPL)*, Sep. 2017.

[5] Y.Tokuyoshi, H.Matsutani, and H.Amano, "Key-value Store Chip Design for Low Power Consumption," *CoolChips 22*, April 2019.

[6] S.Terashima, T.Kojima, H.Okuhara, Y.Matsushita, N.Ando, M.Namiki, and H.Amano, "A shared memory chip for twin-tower of chips,," *SASIMI2018*, March 2018.

[7] N.Miura, H.Ishikuro, T.Sakurai, and T.Kuroda, "A 0.14pJ/b Inductive-Coupling Inter-Chip Data Transceiver with Digitally-Controlled Precise Pulse Shaping," *IEEE International Solid-State Circuits Conference*, pp. 358–608, February 2007.

[8] A.Nomura, Y.Matsushita, J.Kadomoto, H.Matsutani, T.Kuroda, and H.Amano, "3d chip stack with inductive coupling thruchip interface," *International Journal of Network and Computing*, vol. 8, no. 1, 2018.

[9] "TOPPERS Project," (Date last accessed 22-May-2019). [Online]. Available: https://www.toppers.jp/en/project.html

[10] J. T. Kao, M. Miyazaki, and A. Chandrakasan, "A 175-mv multiply-accumulate unit using an adaptive supply voltage and body bias architecture," *IEEE journal of solid-state circuits*, vol. 37, no. 11, pp. 1545–1554, 2002.

[11] H. Okuhara, A. Ben Ahmed, and H. Amano, "Digitally assisted on-chip body bias tuning scheme for ultra low-power vlsi systems," *IEEE Transactions on Circuits and Systems I: Regular Papers.*, 2018.

[12] H. Nagatomi, N. Sugii, S. Kamohara, and K. Ishibashi, "A 361nA Thermal Run-away Immune VBB Generator using Dynamic Substrate Controlled Charge Pump for Ultra Low Sleep Current Logic on 65nm SOTB," in *Proceedings of the SOI-3D-Subthreshold Microelectronics Technology Unified Conference*, Oct. 2014, pp. 1–2.

[13] M. Blagojević, M. Cochet, B. Keller, P. Flatresse, A. Vladimirescu, and B. Nikolić, "A fast, flexible, positive and negative adaptive body-bias generator in 28nm FDSOI," in *2016 IEEE Symposium on VLSI Circuits (VLSI-Circuits)*. IEEE, 2016, pp. 1–2.

[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*, ser. NIPS'12. USA: Curran Associates Inc., 2012, pp. 1097–1105. [Online]. Available: http://dl.acm.org/citation.cfm?id=2999134.2999257

[15] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.