Research and Development of Backend Readout Electronics for Hyper Suprime-Cam

FUJIMORI Hiroki
Department of Physics, University of Tokyo

January 2010

Abstract

Hyper Suprime-Cam (HSC) is the next generation wide-field camera proposed for the 8.2-meter diameter Subaru Telescope in Hawaii. One of the main purposes of the HSC is to give a new constraint on the nature of dark energy through weak lensing survey.

The HSC employs 116 pieces of 2k×4k Charged Coupled Devices (CCDs) to cover a 1.5 degree diameter field of view. The size of its image amounts to 2.3 GBytes, which is more than ten times larger than that of the existing camera. Its readout electronics is required to read the whole image data in under ten seconds.

In this research we have developed the back-end readout electronics (BEE) for HSC. The BEE is a digital component of the readout electronics and provides an interface between the analog front-end electronics (FEE) and the data acquisition (DAQ) system. The BEE sends a readout clock to the CCD and FEE, receives the image data from the FEE and sends them to the DAQ system.

We carried out the performance measurements of the BEE with the combination of four CCDs, one FEE and one BEE. The developed back-end readout electronics satisfied all requirements for BEE and no problems were found in the configuration.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Recent advancements in astronomical observation technology may help to change the humanity's understanding of the cosmos; one profound fact that has become clear is that the universe is expanding at an accelerating rate. This implies that there is some unknown energy which has a negative pressure. The source which causes the universe to expand acceleratingly is generally called dark energy. From the recent measurement, it was revealed that about 73 % of the total energy of the universe is made up from this unknown energy, 23 % is dark matter and the "usual" matter occupies only 4 % of the total universe. It is astonishing that most of the universe is filled with these peculiar components.

To identify what dark energy is after all, many observations have been proposed. One of these is weak lensing survey. A deflected image from a far galaxy includes the information of the mass structure along the light path. The time evolution of this mass structure is expected to tighten the cosmological parameters.

In this method however, a vast amount of galactic images are required to obtain an effective result. For this reason, a next generation camera mounted on the prime focus of the Subaru Telescope is under developing. This camera named Hyper Suprime-Cam (HSC) has the advantage of a wide field of view as well as superb resolution.

This thesis reports on the development of the back-end readout electronics for HSC and its performance. The size of the image from HSC amounts to more than 2 GBytes due to its wide field of view. This is more than 10 times the data produced from the existing camera. Fast readout electronics able to read this huge data within 10 seconds is required. The main readout module called GESiCA has already developed. Using this module, I designed the other back-end modules including implementations of FPGAs and micro-controllers to complete the whole back-end readout system.

This thesis will proceed in the following way. Physics and overview of HSC are described in chapter 2. Chapter 3 discusses the readout electronics for HSC and its requirements. The details of the developed back-end readout system are explained in chapter 4. Chapter 5 describes the performance tests and results of

the readout electronics. The system test with CCD cameras for HSC is written in chapter 6. Finally, our studies are summarized in chapter 7.

Chapter 2

Dark Energy and HSC

In this chapter, we explain the physical motivation for our studies. First the physics of dark energy is discussed in section 2-1. In section 2-2, we explain the overview of the Hyper Suprime-Cam.

2.1 Dark Energy and Weak Lensing Survey

Einstein equation is written as

$$G_{\mu\nu} + \Lambda g_{\mu\nu} = \frac{8\pi G}{c^4} T_{\mu\nu} \tag{2.1}$$

where $G_{\mu\nu}$ is the Einstein tensor, Λ is the cosmological constant, $g_{\mu\nu}$ is the metric tensor, G is gravitational constant and $T_{\mu\nu}$ is the energy-momentum tensor. Using the Robertson-Walker metric and assuming that the universe is homogeneous and isotropic, the Einstein equation leads to the next two equations:

$$\left(\frac{\dot{a}}{a}\right)^2 = \frac{8\pi G\rho}{3} - \frac{Kc^2}{a^2} + \frac{\Lambda c^2}{3} \tag{2.2}$$

$$\frac{\ddot{a}}{a} = -\frac{4\pi G}{3c^2}(\rho c^2 + 3P) + \frac{\Lambda c^2}{3}. \tag{2.3}$$

Here $a(t)$ is the scale factor of the universe, ρ is the energy density, K is the curvature of the space and P is the pressure. The former equation is specifically called the Friedmann equation and describes the time evolution of space.

The expansion rate of the universe is determined by the energy density of space. Let us consider the equation of state as written in the form

$$P = w\rho c^2 \quad (w : \text{constant}) \tag{2.4}$$

for the simple case. For example, if the universe is filled with the non-relativistic matter ($w = 0$) and the universe is flat ($K = 0$), the universe behaves as

$\ddot{a} \propto -t^{4/3} < 0$ which means the universe expands deacceleratingly. On the other hand, if the universe is filled with the cosmological constant ($w = -1$), the universe expands exponentially. The energy which causes the universe to expand acceleratingly is generally called *dark energy*.

Many observations have been made to determine the components of the universe. For example, from the examination of the relation between luminosity distance and redshift parameter z, the cosmological parameters can be calculated. Using the type Ia supernovae as the standard candles, two groups (Riess *et al.*, 1998, Perlmutter *et al.*, 1999) reported that the universe is expanding acceleratingly, which means that the main component of the universe is dark energy. Moreover the observations of the anisotropy pattern of the cosmic microwave background strongly supports the existence of dark energy. For a flat ΛCDM universe, the dark energy parameter is reported as [2, 3]

$$\Omega_\Lambda = \begin{cases} 0.713^{+0.027}_{-0.029}(\text{stat})^{+0.036}_{-0.039}(\text{sys}) & (\text{from type Ia supernovae observation}) \\ 0.742 \pm 0.030 & (\text{from WMAP observation}) \end{cases}. \tag{2.5}$$

Weak lensing survey is expected to tighten the constraint of the dark energy parameters. According to general relativity, the light path is distorted by the cosmological gravitational field. As a result, the observed shapes of distant galaxies are distorted by intervening foreground mass overdensities. This phenomenon is called weak lensing. By surveying the weak lensing effect, we can get information about the mass distribution of the universe. Since the mass distribution of space is affected by the cosmological parameters, we can give a new constraint on the nature of dark energy.

Figure 2.1 explains the gravitational lensing effect. The light path of the source galaxy at the distance D_s is distorted at the foreground object at the distance D_l. Deflected by the deflection angle $\vec{\hat{\alpha}}$, the true image at the position $\vec{\beta}$ is actually observed at $\vec{\theta}$. Here we can write the lens equation as

$$\vec{\beta} = \vec{\theta} - \frac{D_{ls}}{D_s}\vec{\hat{\alpha}}(\vec{\theta}) = \vec{\theta} - \vec{\alpha}(\vec{\theta}). \tag{2.6}$$

According to general relativity, the deflection angle $\vec{\alpha}$ is given as

$$\vec{\alpha}(\vec{\theta}) = \nabla\psi(\vec{\theta}) = \frac{4G}{c^2}\frac{D_l D_{ls}}{D_s}\int \mathrm{d}^2\vec{\theta}'\frac{\vec{\theta} - \vec{\theta}'}{|\vec{\theta} - \vec{\theta}'|^2}\Sigma(D_l\vec{\theta}') \tag{2.7}$$

where ψ is the two-dimensional gravitational lens potential of lens objects and Σ is the two-dimensional mass density on which mass density of lens objects is projected along the line of sight.

In equation (2.7), considering the gravitational mapping of the small deviation $(\delta\vec{\beta}, \delta\vec{\theta})$ around the center of the target galaxy $(\vec{\beta}, \vec{\theta})$, we obtain

$$\delta\vec{\beta} = \frac{\partial\vec{\beta}(\vec{\theta})}{\partial\vec{\theta}}\delta\vec{\theta} \equiv \boldsymbol{A}(\vec{\theta})\delta\vec{\theta} \tag{2.8}$$

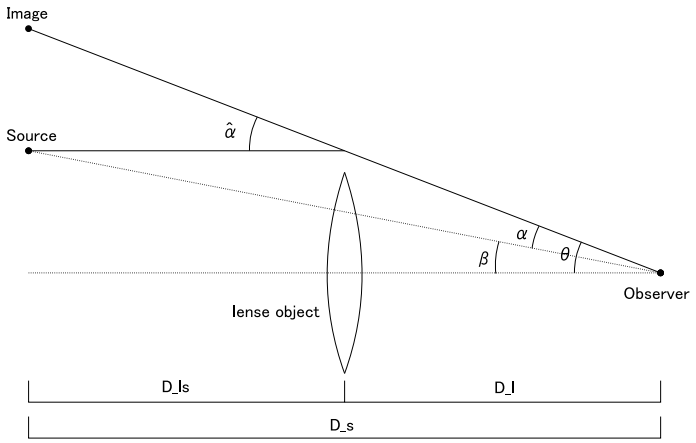

Figure 2.1: Gravitational lensing effect

where the Jacobian matrix $\boldsymbol{A}$ is written as the form of

$$\boldsymbol{A} = \begin{pmatrix} 1-\kappa-\gamma_1 & -\gamma_2 \\ -\gamma_2 & 1-\kappa+\gamma_1 \end{pmatrix} = (1-\kappa)\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} - \begin{pmatrix} \gamma_1 & \gamma_2 \\ \gamma_2 & -\gamma_1 \end{pmatrix}. \tag{2.9}$$

The transformation matrix $\boldsymbol{A}$ maps the original image $\vec{\beta}$ to the distorted image $\vec{\theta}$. Using the gravitational lens potential, κ and γ are given as

$$\kappa \equiv \frac{\psi_{11}+\psi_{22}}{2} \tag{2.10}$$

$$\gamma_1 \equiv \frac{\psi_{11}-\psi_{22}}{2} \tag{2.11}$$

$$\gamma_2 \equiv \psi_{12}. \tag{2.12}$$

From equation (2.9), κ can be interpreted as the source for changing the size of the galaxy image, while γ distorts the spherical shape to an ellipse as in figure 2.2. The former effect is called the convergence and the latter the shear.

Since an actual galaxy has an intrinsic ellipticity ε_i, the observed ellipticity $\varepsilon_i^{\rm obs}$ can be written as

$$\varepsilon_i^{\rm obs} \simeq \varepsilon_i + \gamma_i \tag{2.13}$$

postulating $\kappa, \gamma \ll 1$. Because we do not know the intrinsic form of the individual target galaxy, we can get no information about the mass distribution from a single ellipticity measurement. By observing numerous galaxies and taking average of them, however, systematic distortion can be obtained. That is because the contributions from the intrinsic ellipticity cancel by averaging them,

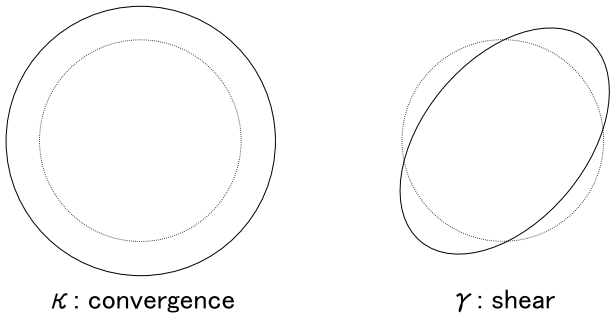

Figure 2.2: Convergence and shear. A dotted line represents the original galaxy. κ and γ shifts the original galaxy to the image shown with the solid line.

supposing the azimuths of the intrinsic ellipticity are independent to each other. Then, the mean of the ellipticity is expected to be

$$\langle \varepsilon_i^{\rm obs} \rangle = \langle \gamma_i \rangle \pm \frac{\sigma}{\sqrt{N}} \tag{2.14}$$

where N is the number of the observed galaxies and σ is the standard variation. This systematic distortion is called *cosmic shear*.

Therefore in order to calculate the mass distribution from the cosmic shear and to get an information about the cosmological parameters, the observation needs to

- take the precise image and
- survey over a wide field of the sky.

We need to observe a vast number of galaxies so that we can suppress the systematic error in equation (2.14). The Hyper Suprime-Cam described in the next section is expected to satisfy these requirements.

2.2 Subaru Telescope and Hyper Suprime-Cam

There are several projects for weak lensing surveys aiming to extract information on dark energy. Among these is a project using the Subaru Telescope. This project satisfies requirements for both precise image and wide field of view by mounting the next generation wide-field camera on the prime focus of the Subaru Telescope. In this section, we discuss the features of the Subaru Telescope and the next generation camera.

Subaru is an 8.2-meter optical-infrared telescope at the 4,200-meter summit of Mauna Kea, Hawaii, operated by the National Astronomical Observatory of Japan (NAOJ). The main features of Subaru are as follows:

- 8.2m primary mirror
  Subaru has an 8.2m large primary mirror which enables us to observe darker and further galaxies.

- high quality imaging
  The primary mirror is polished so smoothly that the average bump is only 12 nm. In addition, the mirror is supported by 261 robotic fingers named actuators to keep it in perfect shape no matter where the telescope is pointing in the sky. Advanced technologies like these make us able to obtain high quality images.

- prime focus
  Subaru is the only 8-meter class telescope in the world with the ability to mount an instrument at prime focus, which generally has the advantage of obtaining a wide field of view. A wide field can be surveyed in a single exposure, so that Subaru is suitable for observations like weak lensing survey.

Subaru currently mounts Suprime-Cam on its prime focus unit [8]. The field of view of Suprime-Cam is $34' \times 27'$, which covers an area comparable to the size of the full moon. The focal plane consists of ten high-resistivity Charge Coupled Devices (CCDs) and the total image size is 80 million pixels. The seeing of the acquired image is typically 0.6 arc seconds (FWHM), which is good enough for precise measurement of the shear effect [9].

Although the quality of the image by Suprime-Cam is suitable for dark energy survey, estimations show that the field of view is not sufficient. We are developing a next-generation wide-field prime focus camera, which is called Hyper Suprime-Cam (HSC) [10]. HSC has 116 pieces of CCDs as in figure 2.4. The field of view of HSC is 1.5 degrees in diameter in order to make the survey speed even faster, while maintaining the equivalent image quality as that of Suprime-Cam. The survey area is required to be 2,000 square degrees. Using HSC, a total exposure time is estimated to be 1,200 hours or 150 nights. The first light is scheduled on 2011, and the instruments of HSC are now being developed.

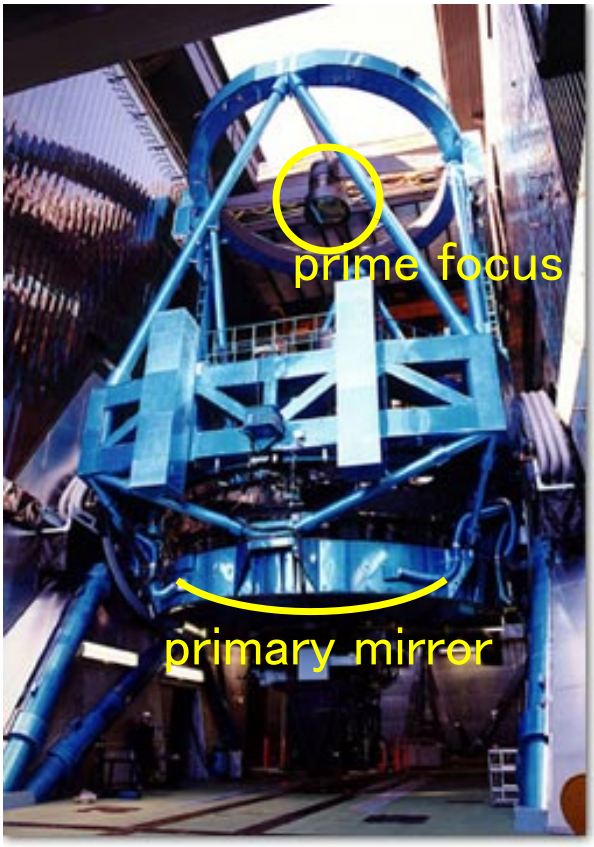

Figure 2.3: Subaru Telescope (from [7])

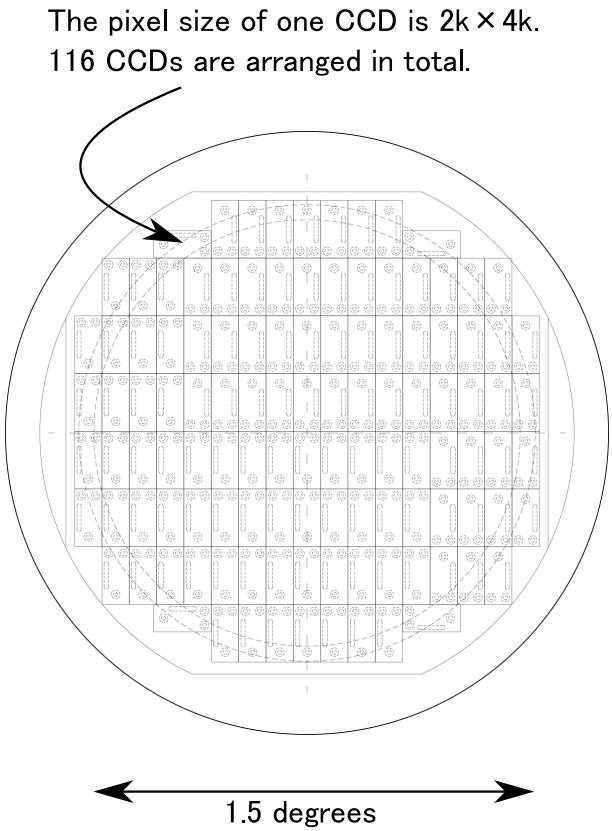

Figure 2.4: CCDs on HSC

Chapter 3

Overview of the Readout Electronics System for Hyper Suprime-Cam

In this chapter, an overview of the readout electronics system for Hyper Suprime-Cam is described. First we introduce the overview of the system structure in section 3-1. The readout system consists of front-end electronics (FEE) and back-end electronics (BEE). The interfaces of the systems are summarized. In the next section 3-2, we explain the FEE. FEE digitizes the CCD signals. The structure of FEE is explained. The discussion of BEE follows in section 3-3. BEE controls the FEE, reads the CCD image data from FEE and transmits them to a DAQ system. The function of BEE and its requirements are summarized in this section.

3.1 Design Overview

In order to acquire wide field of view of the camera, HSC is designed to use 116 pieces of 2k $\times$ 4k CCDs [10]. The analog data from the CCDs is digitized into 16 bits per pixel, hence the total data size is as large as about

$$2\text{k} \times 4\text{k} \times 2(\text{Byte/pixel}) \times 116 \simeq 2\text{GByte} \tag{3.1}$$

which is more than 10 times larger than that of the existing Suprime-Cam [8]. The readout electronics system is required to handle the CCD data and to send the whole data within 10 seconds in order to obtain the effective amount of data in the practical exposure time. The most challenging part of the HSC electronics system is to satisfy such fast readout speed.

Table 3.1 summarizes the requirements for the HSC electronics. Although the number of CCDs installed in HSC is 116, the readout electronics has to handle 128 CCDs for the maximum case. From the readout time, the electronics

Number of CCD	128
Total Signal Output	512 ports
Image Format	4272 × 2272 pixels (including 8 blank, 48 over-scan row & line)
Full well Capacity	$\geq$ 150,000 e^- (slow readout mode)
Data Resolution	16 bits
Data Size	2.31 GByte/frame (2427 MByte/frame)
Readout Time	min. 10 seconds/frame (standard mode)
Pixel Rate	max. 243 k/sec
Total Readout Noise $N_{\rm total}$ (*)	$<\sim$ 10 e^- (after installing to the prime focus) $<\sim$ 5 e^- (before installing to the prime focus)
Noise from CCD $N_{\rm CCD}$	4 e^-
Electronics Noise $N_{\rm electronics}$	$<\sim$ 3 e^-
Linearity	$\leq \pm 1\%$ (after installing to the prime focus)

Table 3.1: Requirement for the HSC readout electronics.
(*) Except for Poisson noise.

system is required to read the CCD image with the speed of at least 243k pixels per second. One shot can be read in about only 10 seconds at this speed.

In addition, the electronics system has to suppress the readout noise in order to acquire a high resolution image of the galaxies. Although the total readout noise from the readout system including CCDs and electronics is required to be lower than 10 electrons after installation in the prime focus unit, it is desirable to suppress the noise to 5 electrons during the performance test at the ground. The CCDs for HSC intrinsically have 4 electron noise. Since the total readout noise $N_{\rm total}$ is described as $N_{\rm total} = \sqrt{N_{\rm CCD}^2 + N_{\rm electronics}^2}$, the electronics noise $N_{\rm electronics}$ is permitted up to 3 electrons.

The overview of the electronics system is shown in figure 3.1. The electronics is connected to the data acquisition (DAQ) system. The electronics is integrated into the prime focus camera unit (see figure 3.2), while the DAQ system is located in the observation room far from the camera unit.

The readout electronics for HSC can be divided into two parts; one is the front-end electronics (FEE) placed in the CCD dewar and the other is the back-end electronics (BEE) mounted outside of the dewar in the prime focus unit. The FEE is an analog circuit to readout CCD images. The FEE includes CCD clock drivers, pre-amplifiers, correlated double samplings (CDSs), 14-bit digital-to-analog converters (DACs) and 16-bit analog-to-digital converters (ADCs). The analog data from CCDs are first amplified by pre-amplifiers, integrated by CDSs, digitized by ADCs and sent to BEE systems. The detailed functions of the FEE are discussed in the next section.

The BEE is a digital circuit to intermediate between the FEE and the DAQ system. When the DAQ system commands the BEE to readout the CCD data,

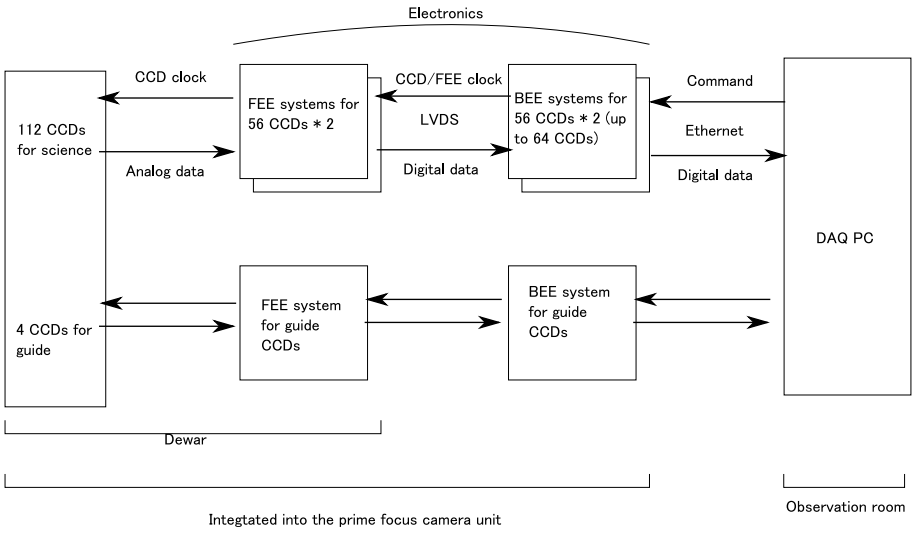

Figure 3.1: Overview of the readout electronics for HSC

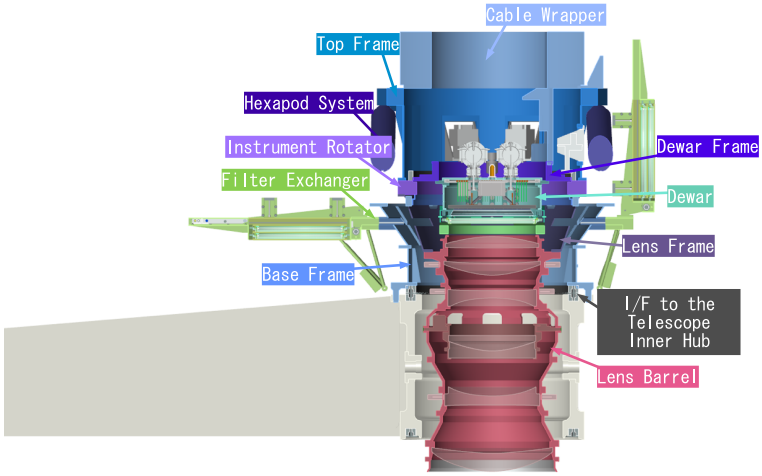

Figure 3.2: Schematic view of planned HSC structure ([10]). This unit is equipped to the prime focus shown in figure 2.3.

the BEE generates the CCD clock together with the FEE clock. After the BEE receives the digitized data from the FEE, it temporarily stores them in a frame memory, and then transfers to the DAQ system. The electronics for HSC is made in collaboration with National Astronomical Observatory of Japan (NAOJ); the FEE is developed at NAOJ, and we developed the BEE system at University of Tokyo. The research and development of the BEE system is the main topic of this thesis.

One set of FEE and BEE is designed to handle 56 CCDs (64 CCDs at maximum case). Two sets of this combination are used for reading 112 CCDs. The image data from these 112 CCDs are analyzed on the DAQ system. Other than the 112 CCDs, there are another 4 CCDs called guide CCDs. The image from the guide CCDs is used for controlling the angle of the telescope by chasing objects in the guide image. For the readout of the guide CCDs, it is planned to use one more set of the electronics system.

The FEE and BEE are connected via LVDS (low voltage differential signaling) only for noise immunity [11]. Analog signals do not cross the dewar at all.

The 1 Gbps Ethernet with Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) as parts of the Transmission Control Protocol / Internet Protocol (TCP/IP) is used for connecting the DAQ system to BEE. Ethernet and TCP have been widely used in a variety of fields and TCP/IP is the de facto standard network protocol implemented in many operating systems (OS). We use this method to transmit the huge image data to the DAQ PC with high data transfer throughput.

We also use this Ethernet with UDP protocol to control the electronics modules from the DAQ system. In the electronics system, all modules are connected via Universal Asynchronous Receiver Transmitter (UART). UART is a common method for serial communication and implemented in many devices such as micro-controllers. UART is often used with the standard EIA RS-232C [12], the well-known example of which is a communication between two computers via the serial ports. We use this scheme to communicate between the readout electronics system including FEE and BEE. When a command from the DAQ system is received in the electronics, the command is transformed from UDP to the UART protocol and sent to all FEE/BEE modules. Each module can also reply various kinds of information such as the internal status parameter to the DAQ system. The reply messages are converted from UART to the UDP protocol and sent to the DAQ system.

3.2 Front-end Electronics

A FEE system includes several FEE boards. Two types of FEE boards are made; one can handle four CCDs, and the other can handle six CCDs. Figure 3.3 is a FEE board for four CCDs. Ten FEE boards for four CCDs and three FEE board for six CCDs are used as one FEE module (see figure 3.4) so that one FEE module can handle 56 CCDs.

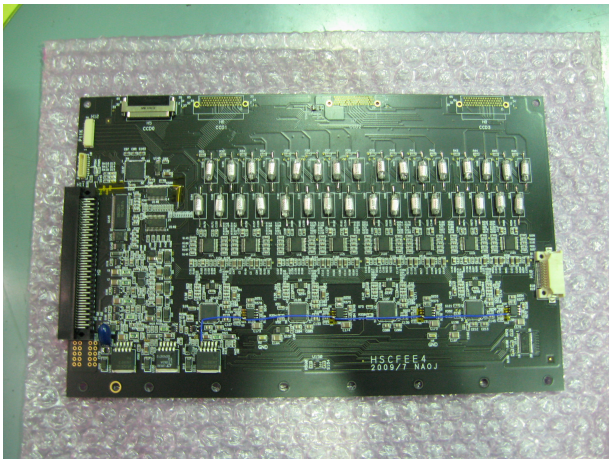

Figure 3.3: FEE board for four CCDs

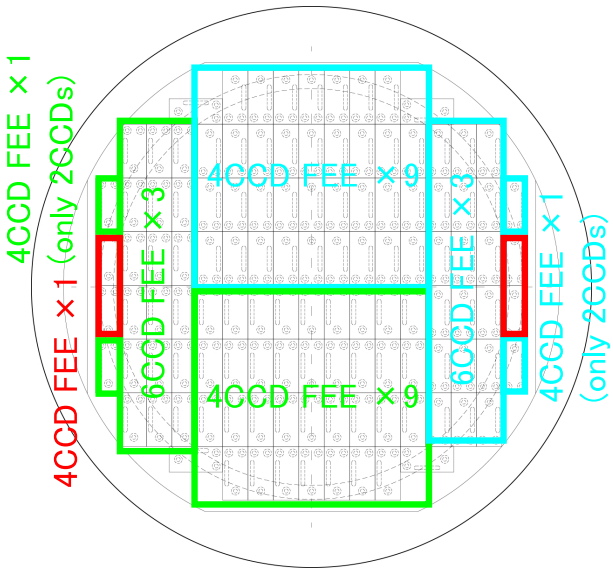

Figure 3.4: FEE board arrangement. CCDs colored with green and blue are used for analysis. Red-lined CCDs are guide CCDs. Three FEE systems are used in total.

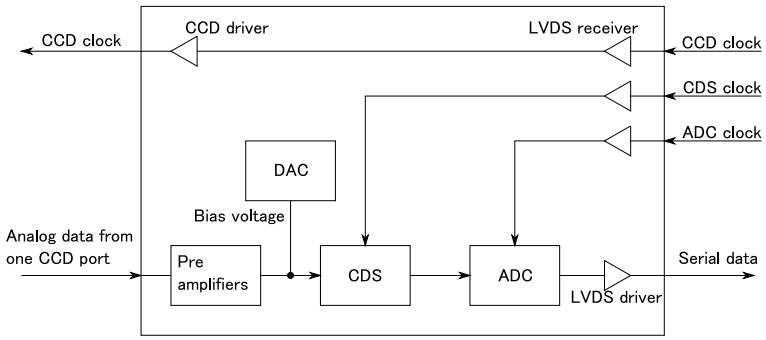

Figure 3.5: Overview of FEE

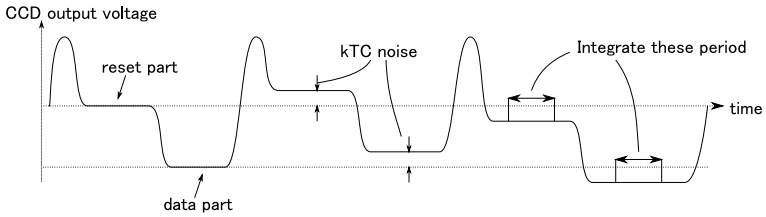

Figure 3.6: Typical signal from CCD

Each FEE board contains pre-amplifiers, CDSs, DACs and ADCs. Those components are connected as illustrated in figure 3.5. A CCD signal is first amplified by the pre-amplifiers. Typical output from the pre-amplifier is shown in figure 3.6. The reset part corresponding to 0 electrons comes first, followed by the data part to give the number of electrons in the pixel. When the signal is read from CCD, however, the noise is added to both the reset part and the data part. This noise is called kTC noise and it is an inevitable noise caused by the structure of CCD [14]. To cancel the kTC noise, we integrate both the data in the reset part and the data part for a common period. Their difference is considered to be output data. This method is called correlated double sampling (CDS). Note that bias voltage can be added to the CDS input using a 14-bit digital-to-analog converter (DAC) AD5555 (Analog Devices, Inc.). The main purpose of adding bias voltage is to reduce the possibility that the analog image data is cut off from the CDS range when the environment around CCDs changes. After the CDS circuit, the data are digitized with a 16-bit analog-to-digital converter (ADC) AD7686 (Analog Devices, Inc.).

In the previous paragraph, we explained the digitizing sequence for an analog signal from one CCD port. There are four output ports per CCD as shown in table 3.1. We use four parallel sequences of figure 3.5 per CCD. Here four AD7686s can be daisy-chained as shown in figure 3.7. This mode is called "chain mode with busy indicator" in the datasheet of AD7686 [13]. Using this mode, a sequence of one CCD channel can be written as in figure 3.8. The data

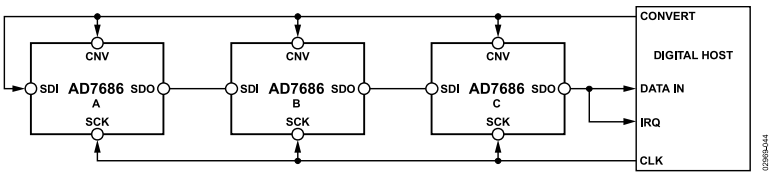

Figure 43. Chain Mode with Busy Indicator Connection Diagram

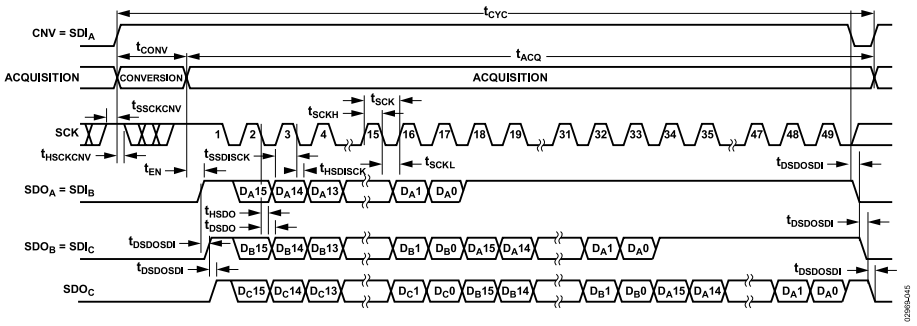

Figure 44. Chain Mode with Busy Indicator Serial Interface Timing

Figure 3.7: AD7686 used as "chain mode with busy indicator" (from [13]). Note that 3 ADCs are illustrated but we connect 4 ADCs daisy-chained.

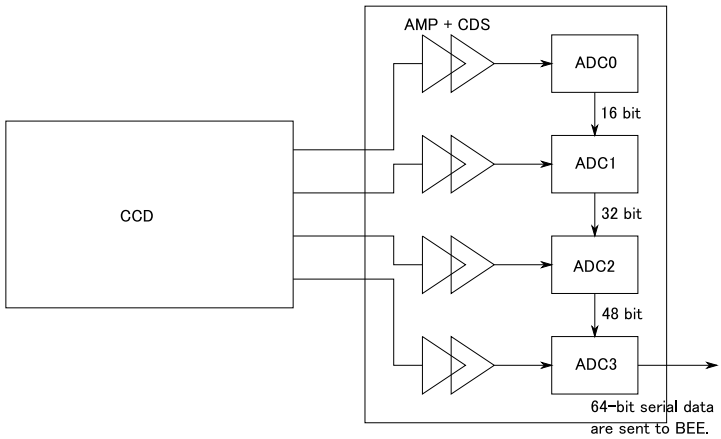

Figure 3.8: Sequence of one CCD channel

Number of Handling CCD	4 or 6
Operating Temperature	-100 C
DAC Resolution	14 bit
DAC Accuracy	1 LSB (INL max.)
CDS Gain	1.0 (when integral time = 2us)
ADC Resolution	16 bit
ADC Accuracy	2 LSB (INL max.)
ADC Operating Frequency	50 MHz
Number of ADC Daisy-chain	4
ADC Input Range	$0 \sim 3.0$ V
Output Format	64 bit serial with 50 MHz
Output Channel	4 or 6
Conversion Factor	$\sim$3.0 electrons / ADC count

Table 3.2: Specifications of One FEE Board

from four channels of one CCD is merged into one serial signal with data length of 64 bits (= 16 bits $\times$ 4). This serial 64-bit data are sent to the BEE system. The specifications of one FEE board are summarized in table 3.2.

Here we must pay attention to the fact that the CDS circuit requires a precise clock. The gain of the CDS circuit is determined by the period of the given clock. Hence if the integral period is fluctuated by the clock jitter, the gain of the CDS circuit also fluctuates, resulting in a source of readout noise. This is important when discussing the requirements for the BEE in the next section.

3.3 Back-end Electronics

BEE is designed to control the FEE and to send the digitized data from the FEE. First we summarize the required specifications of the BEE in section 3-3-1. To achieve the specifications, we designed the BEE system made up from seven boards. The idea of designed structure is explained in section 3-3-2. Finally, the photographs of the fabricated circuit boards are shown with their features in section 3-3-3.

3.3.1 Requirements for BEE

The BEE must generate the CCD and FEE clocks and read the digitized data from the FEE. The BEE also handles the UART communication to control the

FEE and BEE from the DAQ system as discussed in section 3-3-1. The power of CCD and FEE is planned to be supplied by the BEE. We need a power management system in the BEE.

We can classify the functions of the BEE into the following four items. The required specifications of individual functions are listed.

1. BEE/FEE Control by UART
    - The DAQ system must be able to stably communicate with the FEE and BEE by UART. UART is used mainly for the DAQ system to set the operating modes of the electronics system or to read the internal information in the system.
2. Power Management
    - The BEE system supplies power to all modules. Moreover the BEE is required to monitor the voltage and current of the generated power lines. The user is able to acquire the monitored power status via UART. If irregular voltage or current in any power lines is detected, the power must be shutdown to prevent fatal damage to the readout electronics and CCDs.
3. Generation of CCD and CDS Clocks
    - The BEE system is required to generate CCD and CDS clocks according to the user (DAQ system) definition.
    - Especially for the CDS clock, the noise caused by the integral period fluctuation has to be much lower than 3 electrons in a few electron region.
4. Image Data Readout from FEE and Data Transmission to the DAQ System
    - The BEE system is required to generate ADC clock when the user (DAQ system) requests the conversion of the CCD data.
    - The BEE system is required to readout the CCD data from the FEE, to store them temporary to a frame memory and to send them to the DAQ system by Ethernet.
    - Stable readout is required over the temperature range from -20 to +40 Celsius degrees.
    - The total readout noise including the FEE and BEE must be suppressed under 3 electrons in a few electron region.
    - The linearity is less than $\pm 1\%$.
    - As for the readout speed, the BEE is required to read out at least 243 kSPS (sampling per second) of conversion data. This value comes from the maximum pixel rate written in table 3.1.

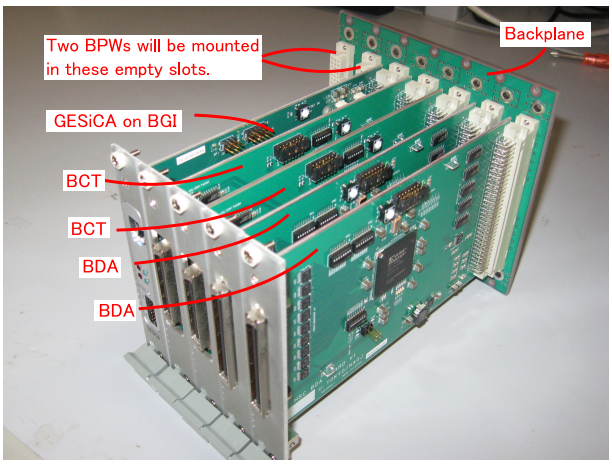

Figure 3.9: BEE system. From left to right, GESiCA on BGI, two BCTs and two BDAs are equipped. Two BPW boards are planned to be equipped in the empty slots.

As explained in section 3-1, the total electronics noise must be suppressed under 3 electrons. When the number of CCD electrons is large, however, the Poisson noise of electrons is so large that other noise is not important. Hence the total electronics noise is required to be smaller than 3 electrons in the "few electron region". The few electron region corresponds to the bias voltage from DACs on the FEE. Since a voltage corresponding to about 1,000 ADC counts is expected to be added normally by these DACs, we measure the electronics noise in such environment.

The BEE system system is designed to be used in the temperatures from -20 to +40 Celsius degrees. Since the Subaru telescope is located on the 4,200-meter summit of Mauna Kea, the temperature goes below freezing point. Hence, we must evaluate the specification of the BEE down to -20 degrees.

3.3.2 Structure of BEE

To satisfy the above requirements, we designed a back-end electronics system as shown in figure 3.9 and 3.10. The BEE is comprised of the following modules: Gigabit Ethernet SiTCP CMC board for Astronomy (GESiCA), Back-end GESiCA Interface (BGI) board, Back-end Power (BPW) board, Back-end Clock Transmission (BCT) board and Back-end Data Acquisition (BDA) board. GESiCA is the main board for the readout with a compact size of 149 mm × 79 mm (CMC card size). It is mounted on a BGI board. The BEE is designed as 3U Euro-card system (160 mm × 100 mm) ([15] - [17]). This small size was adopted because the BEE must be small and light in order to be mounted on the prime focus unit.

One BEE system is operated with the sets of these seven boards. All boards

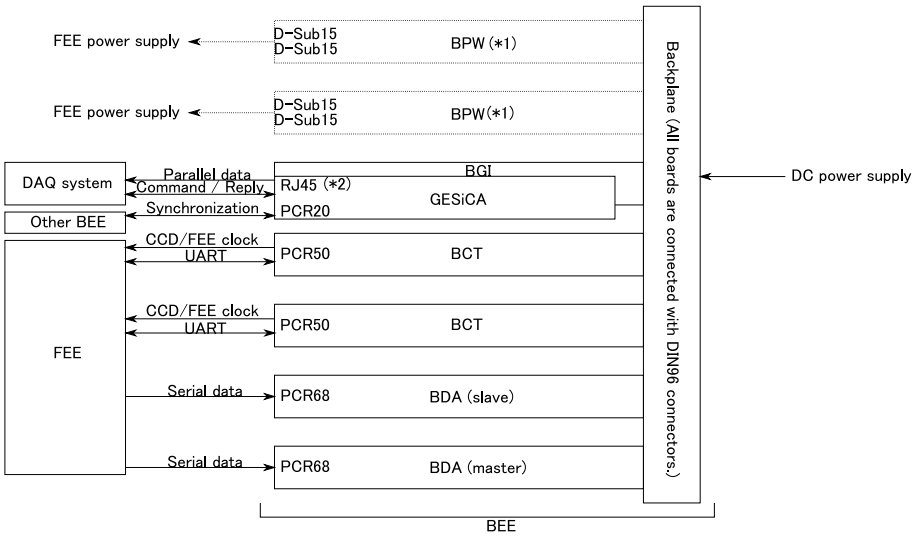

Figure 3.10: Seven boards of BEE.
(*1) BPW will be mounted in near future.
(*2) RJ45 connector will be replaced with a connector for an optical cable.

are connected to a backplane with DIN96 connectors, and all signals in the backplane are through LVDS to ensure the noise immunity. In the figure 3.10, Back-end Power (BPW) boards to supply power to the FEE are not designed yet. The FEE is being operated directly by the DC power supplies until the BPW completion in 2010.

Now let us discuss the idea of the designed structure for each function listed in section 3-3-1.

BEE/FEE control by UART

The structure overview of UART in the readout system is shown in figure 3.11 (Numbers in the figure correspond to the numbers in this paragraph). (1) The user on the DAQ system can send commands to all electronics modules via Ethernet using UDP protocol. (2) The commands on the UDP protocol are converted to the format of UART in GESiCA and (3) they are transmitted to all modules including the FEE. All boards have the functions to interpret UART signals and reply various messages to the DAQ system. Note that the FEE is given the UART signals via BCT boards.

Power management

Although the power of the FEE and CCDs is currently supplied directly from the DC power supply, the power supply structure in the readout system is

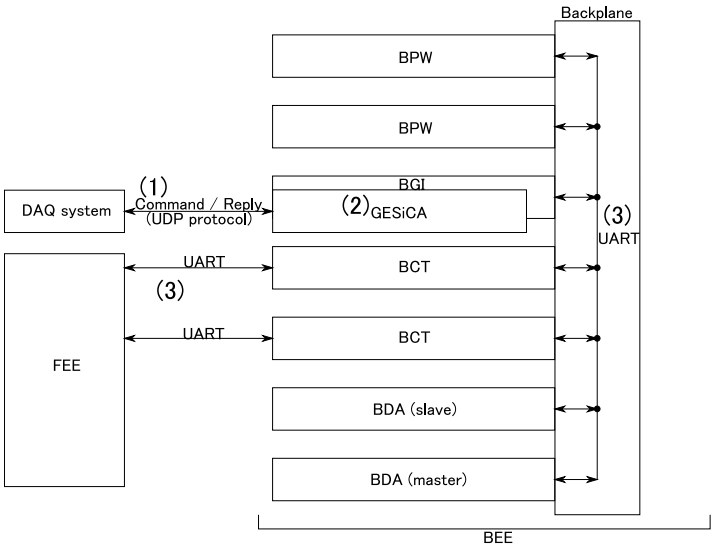

Figure 3.11: Structure overview for BEE/FEE control by UART. Numbers in this figure corresponds to the numbers in the text.

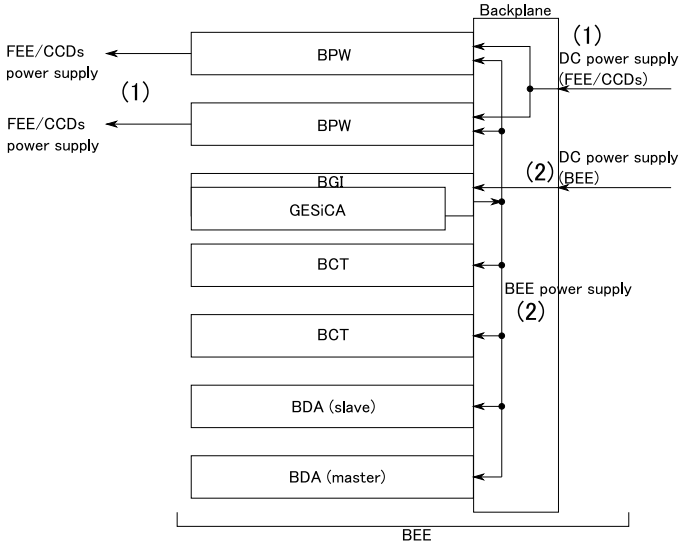

Figure 3.12: Structure overview for power management. Numbers in this figure corresponds to the numbers in the text.

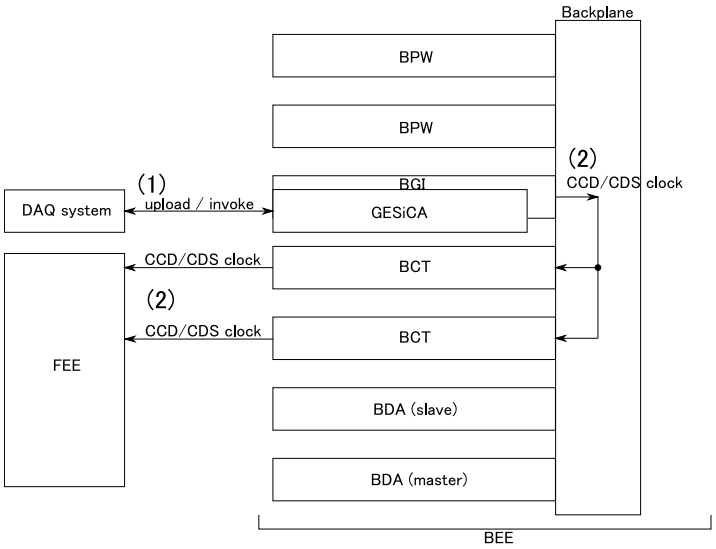

Figure 3.13: Structure overview for generation of the CCD/CDS clock. Numbers in this figure corresponds to the numbers in the text.

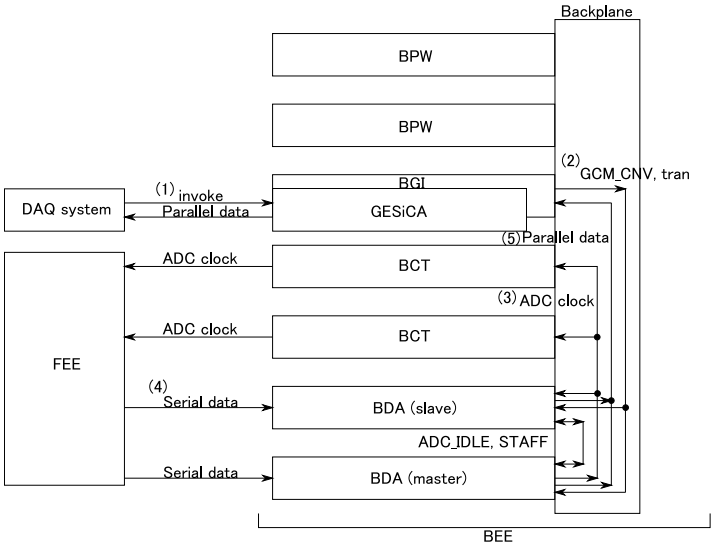

Figure 3.14: Structure overview for readout from the FEE and transmission to the DAQ system. Numbers in this figure corresponds to the numbers in the text. GCM_CNV signal is also explained in the text, while the other signal (GCM_tran, ADC_IDLE and STAFF) are explained in section 4-4.

designed as in figure 3.12. There are two external power supply lines; one for the FEE/CCDs and the other for the BEE. They are connected with solderless terminals on the backplane of the BEE. (1) The FEE and CCDs are given their power supplies via the two BPW boards. The BPWs monitor the condition of the power line. If there is any problem in the power line, the BPWs stop suppling DC power to the FEE. (2) Similarly, the BGI boards monitor the condition of the power line for the BEE and perform the same function if any problem are found in the line.

Generation of CCD and CDS clocks

The structure for generating CCD and CDS clocks is illustrated in figure 3.13. GESiCA has a clock generator with 32 output ports. (1) The user on the DAQ system can upload a user-defined clock pattern including CCD and CDS clocks via Ethernet using UDP protocol. Then when the "invoke" command is sent from the DAQ system to GESiCA, (2) the clock generator starts to output the specified pattern. The clock signals are given to the FEE via BCT boards. To ensure the driver capacity to all FEE modules, two identical BCT boards are used.

Image data readout from the FEE and data transmission to the DAQ system

The overview for the CCD data readout and the data transmission are shown in figure 3.14. The pattern of the clock generator in GESiCA includes GCM_CNV signal, which is a trigger for generating ADC clock. (1) By the "invoke" command, (2) GCM_CNV signal is output from the clock generator to the BDA boards. (3) Then the BDA master board generates the ADC clock, which is driven by the LVDS driver in BCT boards and sent to FEE. (4) The image data from FEE are sent in serial format to the BDA boards. (5) They are converted to parallel data in the BDA boards and sent to the DAQ system via GESiCA. The detailed sequence including an explanation of ADC_IDLE and STAFF signals is described in section 4-4.

3.3.3 Photographs of BEE Boards

The GESiCA board has already been developed (see [18] for the details). In this study we designed and fabricated the other BEE modules: BGI, BCT and BDA. All BEE boards are shown in figures 3.15 - 3.18. We discuss their features below.

- GESiCA (figure 3.15)
  - Image Data transmission to the DAQ system with Gigabit Ethernet GESiCA works as the interface between the DAQ system and BEE system and has a Gigabit Ethernet PHY device to communicate with

the DAQ system. A field programmable gate array (FPGA) Virtex-5 (Xilinx, Inc.) is employed to control the device to achieve the theoretical transfer limits of Gigabit Ethernet.

  - UART communication
    All boards in the electronics system are connected with the UART communication lines. To control the BEE and FEE from the DAQ, the commands are sent on UDP protocol with Ethernet from the DAQ system to GESiCA. GESiCA converts the messages from UDP to UART and sends them to all of the modules in the readout electronics. The reply UART messages from the electronics modules to the DAQ system are similarly handled in GESiCA.
  - CCD/CDS clock generator
    GESiCA can generate CCD/CDS clock patterns according to the user definitions. To generate the CCD/CDS clock, a low-jitter clock source XG-1000CA (Epson Toyocom Corp.) with jitter time of 3 ps (typical) is used. This is because the fluctuation of the CDS integral time given by the clock affects the ADC readout noise (see 5.3.1 for the details).
  - Frame memory
    GESiCA has a 2 GByte DDR2 SDRAM SO-DIMM as a frame memory. This capacity is enough to store the whole image data temporarily because one BEE handles half of the CCDs. The FPGA Virtex-5 is also used here to control the frame memory.

- BGI (Back-end GESiCA Interface) board (figure 3.16)
  - UART communication
    The user can get information such as the monitored power state of the BEE system via UART.
  - Power supply to BEE
    For the BEE power source, 5V DC Power is supplied via the backplane. This DC line is first taken into the BGI board and monitored. When the current and voltage have no problem, then BGI supplies power to all other BEE modules. If any power failure is detected, the power supply to the other module is stopped. To control the BGI board, an AVR micro-controller ATMega2561 (Atmel Corp.) is used. The reason of adopting the micro-controller is that there is a built-in 10-bit ADC to monitor the DC power line and also because it is easy for the micro-controller to handle UART.
  - LVDS interface
    Since GESiCA accepts only the CMOS level signals, the BGI board converts the signal level between CMOS and LVDS.

- BPW (Back-end Power) board (not yet fabricated)

  - UART communication
    The user on the DAQ system can know the monitored values of the DC power line for the FEE at any time using UART.
  - Power supply to FEE
    The FEE power sources are also supplied via the backplane. These power lines are first taken into the BPW board and monitored. When the current and voltage of the power line have no problem, then BPW supplies the power to FEE modules. If a power failure in the line is detected, power supply to the FEE is stopped. To monitor the power lines, AVR micro-controller ATMega2561 (Atmel Corp.) will be used as in BGI board.

- BCT (Back-end Clock Transmission) board (figure 3.17)
  - CCD/FEE clock driver
    The BCT works as an interface between the FEE and BEE. The CCD/CDS clock generated in GESiCA and ADC clock generated in BDA are sent from the LVDS driver on BCT to the FEE. To ensure the driver capacity to all FEE modules, two identical boards are used for sending the clocks.
  - UART communication
    A Spartan3AN FPGA (Xilinx, Inc.) is used for controlling the UART communication.

- BDA (Back-end Data Acquisition) board (figure 3.18)
  - UART
    The various parameter such as the readout frequency can be specified with UART.
  - ADC clock generator
    The BDA master board generates an ADC clock to read out the image data of CCDs when GCM_CNV signal from GESiCA is received. The slave board just receives the ADC clock generated by the master board to know when the conversion has taken place.
  - Readout the image data from FEE
    Each BDA board handles up to 32 CCDs, so that the whole BEE system can handle 64 CCDs. After the serial image data are received, they are deserialized to parallel 16 bits so that GESiCA can accept the data. The parallel data are then sent to GESiCA, first from the master board followed by the slave board. To implement these functions, a Spartan-3AN FPGA (Xilinx, Inc.) is used.

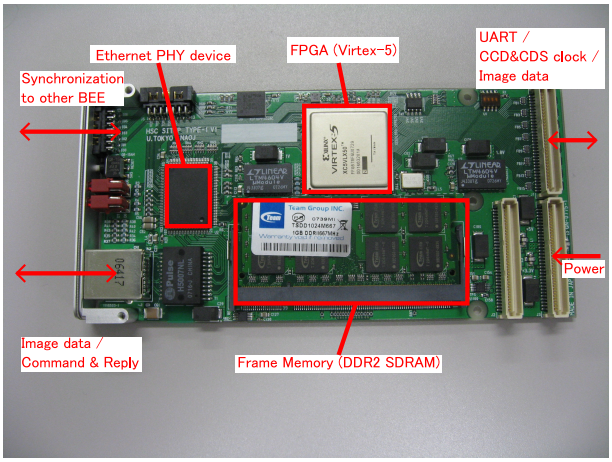

Figure 3.15: GESiCA

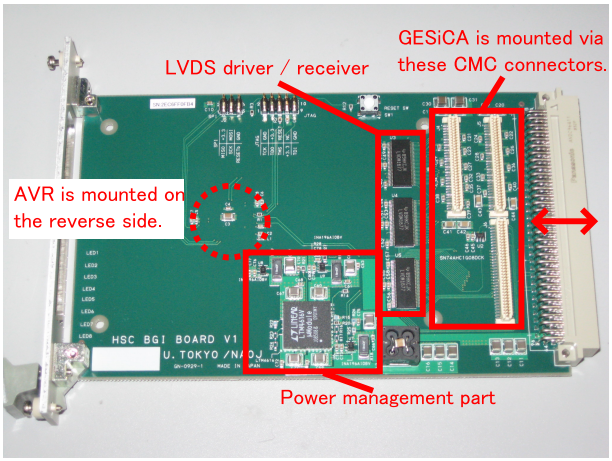

Figure 3.16: BGI

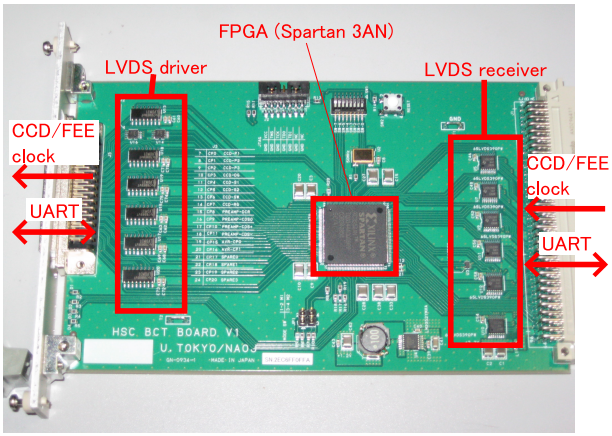

Figure 3.17: BCT

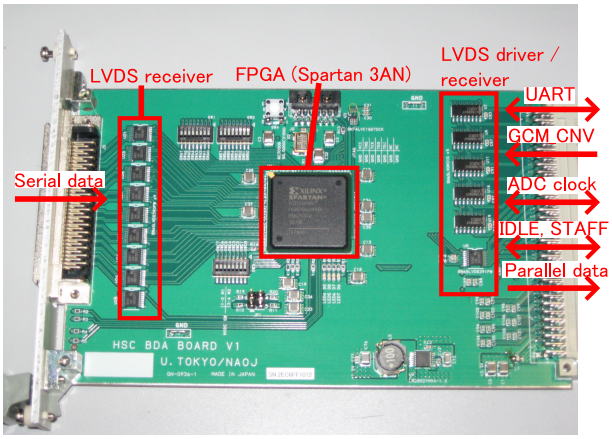

Figure 3.18: BDA

Chapter 4

Development of the BEE System

In this chapter, we describe the details of the developed back-end readout system. Discussion proceeds in the order of the items shown in section 3-3-1: UART, power management, CCD/CDS clocks and CCD image data readout and data transmission to the DAQ system.

4.1 BEE/FEE Control by UART

We employ UART for the communication between the modules of the FEE and BEE to control them from the DAQ system. UART is a well used method of serial asynchronous transmission. We use UART mainly for the DAQ system to set the operating modes of the electronics system or to read the internal information in the system. In this subsection, we discuss how we control the BEE and FEE by UART. First we explain how a data format is determined in UART communication. Next, the designed architecture to control the FEE and BEE is shown. The description of the communication protocol in the electronics system by UART follows. Then actual implementations of the control method for the BDA and BCT are given. Finally we discuss the reliability of UART for our configuration.

4.1.1 Data Format

UART is employed to pass serial communications in the electronics system. UART takes bytes of data and transmits the individual bits in a sequential fashion. The destination module reassembles the bits into complete bytes. In the readout system of the FEE and BEE, to allow two-way communication in the electronics modules, all electronics modules have an output port (TX) and an input port (RX). A TX of one module is connected to the RX(s) of other module(s).

Baud Rate	57.6kbps
Start Bit	1 bit
Data Bit	8 bit
Parity Bit	even parity
Stop Bit	1 bit

Table 4.1: Timing parameters of UART transmission in the readout system. Even parity is calculated by doing an exclusive-or of all the data bits.

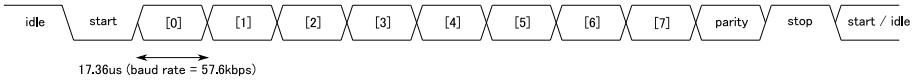

Figure 4.1: The data format of UART

UART allows data to be transmitted asynchronously without the sender having to send a clock signal to the receiver. Instead, a data format for the UART communication is pre-determined. When the sender is in idle state, the UART signal is kept to logic high. A data frame starts with a start bit (logic low), by which the receiver can know the sampling timing. The data bits then succeed, ending with the most significant bit. A parity bit is inserted by option for an error check, and the data frame ends with a stop bit (logic high). The data format is summarized in figure 4.1.

For the receiver to reassemble the UART data correctly, some parameters for the asynchronous transmission must be determined in advance in the electronics system. The parameters in the developed readout electronics are listed in table 4.1. Note that a baud rate (= the transmission speed) in the table is an important parameter to synchronize between the sender and the receiver. The system frequency of the two modules must be accurate enough to agree with the baud rate. We will discuss in this point in 4.1.6.

4.1.2 Architecture

Figure 4.2 shows the architecture of the UART in the BEE system. The user can send commands to control all electronics modules in the FEE and BEE from the DAQ system via Ethernet using UDP protocol. The commands sent to GESiCA are converted to the serial data format of UART and they are transmitted to all modules in the FEE and BEE. As for the reply line from all modules in the FEE and BEE except GESiCA, all TX (transmission) ports are connected to the RX (receive) port in GESiCA. Messages from each of the modules are then converted to the UDP protocol and sent to the DAQ system.

Here we need to consider an interference problem in the BEE and FEE readout system. When more than one module in the system send their reply messages at the same time, their signals interfere with each other (This problem does not occur in the command line, because only GESiCA drives the line). We

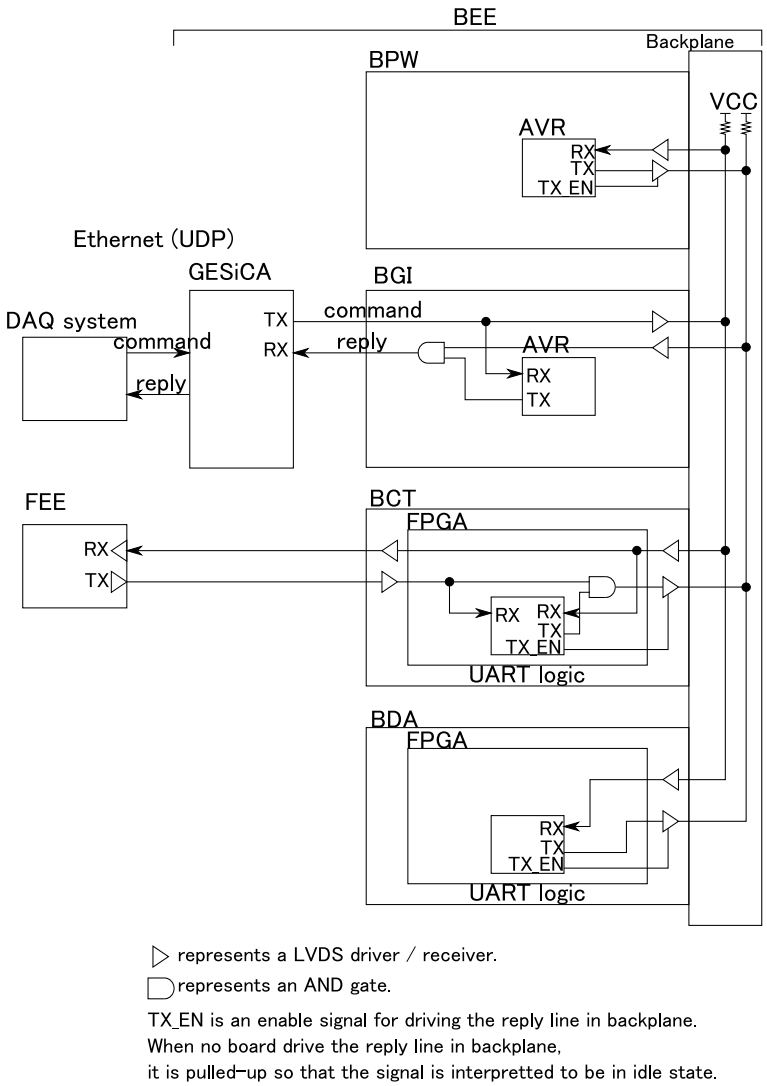

Figure 4.2: The architecture of UART communication

have solved the problem by introducing the transmission protocol described in the next subsection so that up to only one module can send the reply message in the line at a time. Provided that situation, all reply signals can be integrated with AND gates in the BGI and BCT as shown in figure 4.2 since the lines are kept at logic high in the idle state.

4.1.3 Protocol of the UART Communication in FEE and BEE

We use the UART transmission on the following protocol.

- Communications are always started from the DAQ system. All of the electronics modules never send any messages without receiving commands from the DAQ system.
- All modules have different 7-bit ID numbers.
- The most Significant Bit of the 8-bit data is used for specifying ID. When the MSB is activated, the rest of seven bits represents the ID number of the module that the DAQ system would like to communicate with. When the MSB is disabled, the normal character is sent in ASCII code.
- To enhance the reliability of the transmission in the electronics system, the modules are required to echo the received command from the DAQ system.
- The DAQ system starts the message with the ID specification, activating the MSB of the 8-bit data to be sent. Only the module with the requested ID number can drive the reply line to echo its ID number.
- After the ID specification, the normal message is sent. Both commands and replies are always ended with the special character set "CRLF". After each module sends the CRLF, it stops driving the reply line.
- Only after the DAQ system confirms the CRLF from the requested module can the next command be transmitted from the DAQ system.

With this protocol the interference problem is solved (discussed in section 5-1). Figure 4.3 shows a sample sequence of the communication on this protocol. In the sample, the DAQ system communicates to a board with ID = 5. In the case when a command "Sample" is issued and the ID 5 board replies "hello!", the communication goes as in the figure. Please refer to the appendix A for the actual UART ID list and the command lists.

4.1.4 Implementation (BDA)

For the AVR micro-controller used in BGI, the libraries to handle UART are prepared so that the implementation is simple. Here we explain the UART

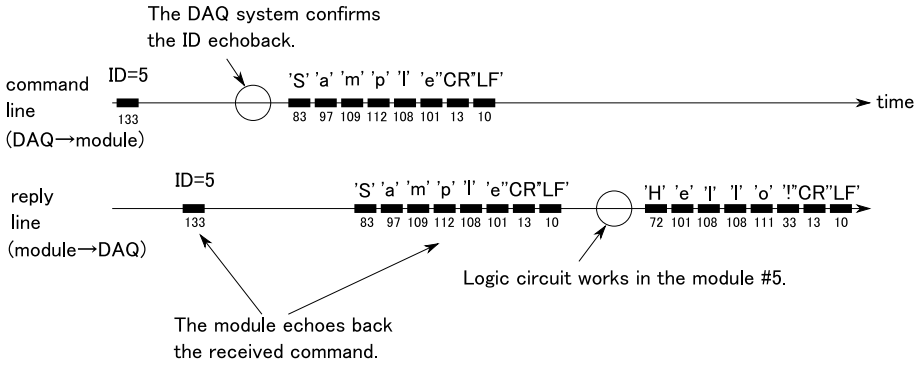

Figure 4.3: Sample diagram when a command "Sample" is sent to a module of ID 5 and it replies "Hello!". The numbers under the characters represent the actual send data. For example the ASCII code of 'S' is 83. Note that the ID specification number becomes (ID + 128), because the MSB is set at that time.

implementation to FPGAs to control the BDA and BCT boards. Let us see the implementation of UART in the BDA board first.

UART is employed mainly to read and write the various internal parameters in the BDA boards. The FPGA in the BDA board has a unit for reassembling the UART data. There is a table of the command list in the FPGA and the reassembled command is compared with the table. If the received command agrees with one of those in the internal list, the specified command is executed in the BDA FPGA.

The structure for handling the UART communication in the FPGA is illustrated in figure 4.4. Below we explain the detail sequence for operating one UART command. Numbers in the figure 4.4 correspond to those in the following explanations.

(1) RS232C_BEE_RX module is a clock recovery and a data recovery unit of the serial data RX_BK (BK stands for backplane side). RS232C_BEE_RX module always samples the RX_BK at the speed of 16 times the baud rate (57.6 kbps). When the clock recovery logic detects a high (idle) to low (start) transition on the RX_BK line, the start bit detection sequence is started and the clock recovery logic recovers the appropriate sample timings. The decision of the logic level of the received data bits is taken by doing a majority voting of the logic value to the three samples in the center of the received bit. See the figure 4.5. After checking parity, the received data BK_RD is handed to BK_rec_logic module.

(2) BK_rec_logic module includes the main logic for interpreting the received characters. If the MSB of the received character is high and the ID number matches with its own, TX_EN_BK_req signal is activated. With this signal, TX_EN_BK, which controls the LVDS driver enable, is driven to high level.

(3) At the same time, the received character is handed to TX_FIFO with

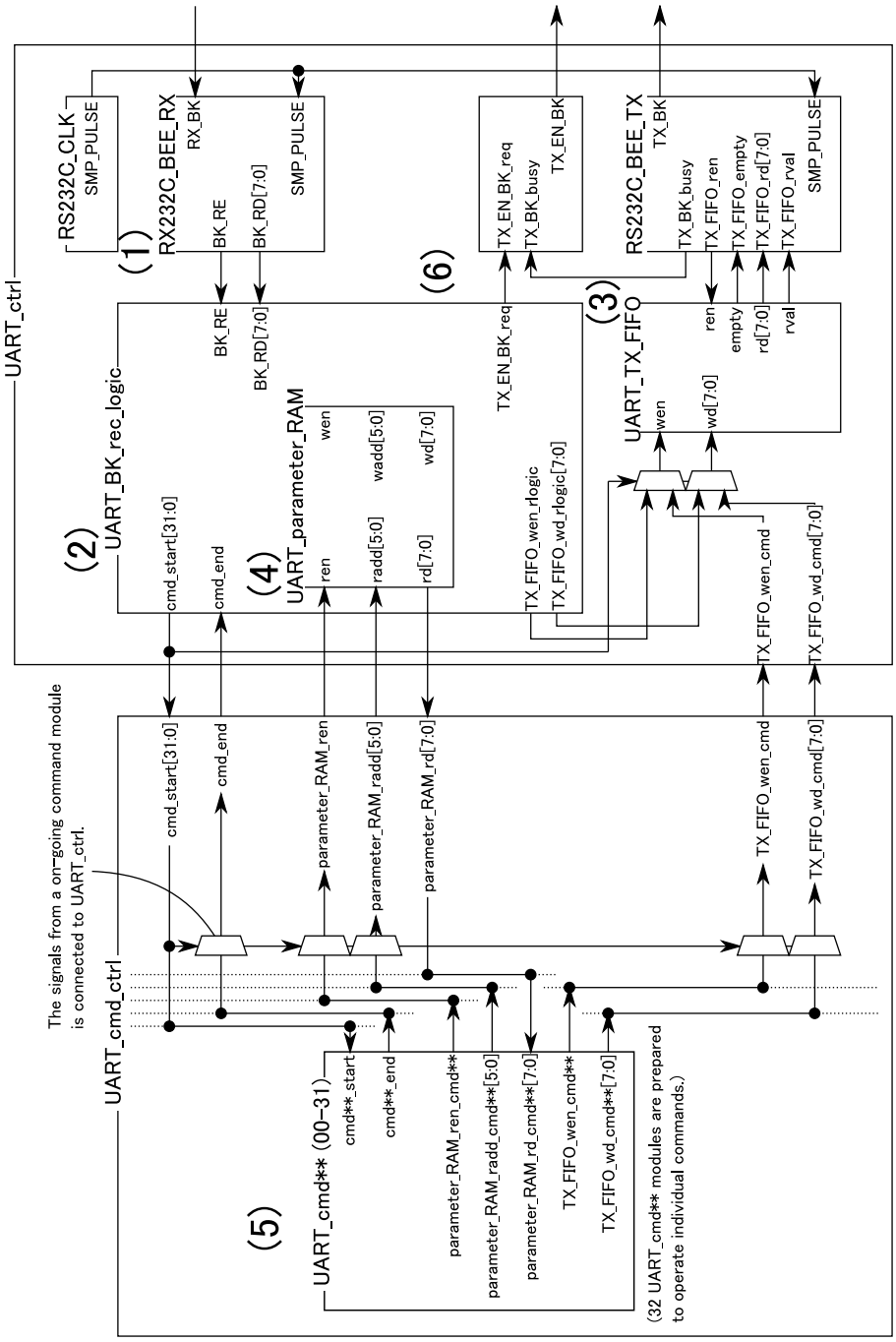

Figure 4.4: UART architecture in BDA

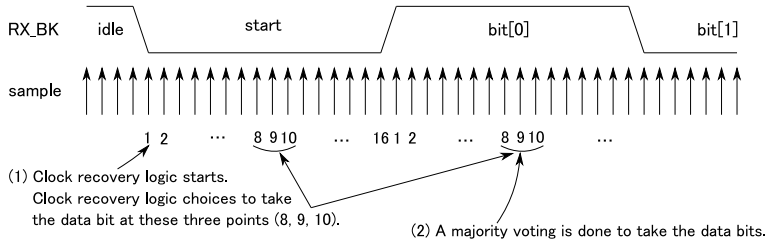

Figure 4.5: The method for recovering clock and data in UART module

TX_FIFO_wen_rlogic and TX_FIFO_wd_rlogic in order to echo the received characters to the DAQ system. RS232C_BEE_TX module converts the characters in TX_FIFO to UART format and sends them to the DAQ system.

(4) The string of the characters following the ID specification is compared with the predefined command strings in the BK_rec_logic module. The commands can be defined up to 31 types each of which is permitted up to 31 characters. If the received string agrees with one of the predefined string, the specified command operation is invoked by activating a corresponding bit in a cmd_start signal. If there are no commands to be matched with the internal command table, cmd00 is started, which replies "No such command". The commands can optionally have the arguments up to 63 words. If there are arguments in the received message, they are written to a memory called parameter_RAM. It is after the echo characters are handed down to the TX_FIFO and the writing to the parameter_RAM is completed that the specified command is started by the cmd_start signal. Note that more than two commands are not invoked at the same time. After one command is started, TX_FIFO_wen_cmd and TX_FIFO_wd_cmd are selected as the inputs to the TX_FIFO.

(5) The behavior of the individual command is described in the cmd** module (** ranges from 00 to 31). The command can access to the parameter_RAM via the signals parameter_RAM_ren / radd / rd. The interpretation of the arguments in the parameter_RAM is dependent on the cmd** module, so that the format of the arguments of the UART commands is flexible.

(6) When the process of the invoked command is over, cmd_end signal is returned to the BK_rec_logic module. Subsequently, TX_EN_BK_req signal is reset to the logic low. After all the characters in TX_FIFO are sent to the DAQ system, TX_EN_BK is disabled.

4.1.5 Implementation (BCT)

The UART architecture of the BDA board can be used in BCT for handling the commands from the DAQ system to the BCT boards. In addition to the function of handling the BCT commands, the BCT has to intermediate between the DAQ system and FEE to communicate between them. The commands from the DAQ system to FEE and its reply messages are basically passed through the BCT board. The BCT boards, however, need to control the TX_EN_BK LVDS

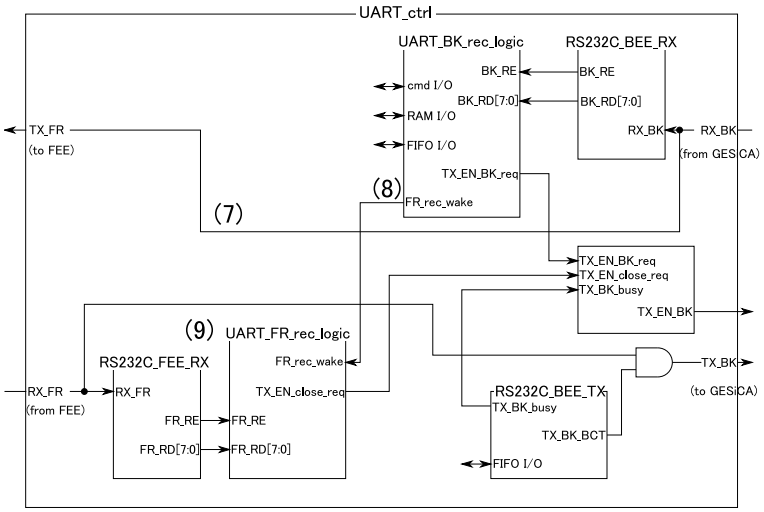

Figure 4.6: UART architecture in BCT. The common modules to figure 4.4 are omitted in this figure.

enable signal. For that purpose, BCT skims the reply messages from FEE to the DAQ system. Here the ID numbers of the FEE board which is connected to the BCT board are determined by the equipped position of the BCT to the backplane (see table A.1 in appendix A). Hence the BCT boards can know how they should control the TX_EN_BK LVDS enable signal. By this method, the UART communication between the DAQ system and FEE is carried out.

Figure 4.6 shows the simplified architecture in BCT. The modules common to the BDA are omitted in the figure. Below is the detailed sequence when the DAQ system communicates with the FEE. Numbers in the figure 4.4 corresponds to those in the following explanations.

(7) The command line RX_BK from the DAQ system is directly connected to TX_FR (FR stands for front-end side) so that the commands from the DAQ system can reach the FEE.

(8) When the ID specification number is the number of the BCT, the operation is the same to what is stated in the previous subsection. If the DAQ system specifies the ID number of the FEE module which the BCT board is connected to, FR_rec_wake signal is activated. At the same time TX_EN_BK (LVDS driver enable) signal is enabled.

(9) In BCT, RS232C_FEE_RX and FR_rec_logic modules are prepared. RS232C_FEE_RX is the same circuit to the RS232C_BEE_RX module. Although BCT just hands RX_FR down to TX_ BK line basically, BCT needs to know when TX_EN_BK should be disabled. In order to detect the end of the reply from the FEE, FR_rec_logic starts to read RX_FR when FR_rec_wake is activated. If the reply ends, TX_EN_close_req is set and TX_EN_BK is disabled.

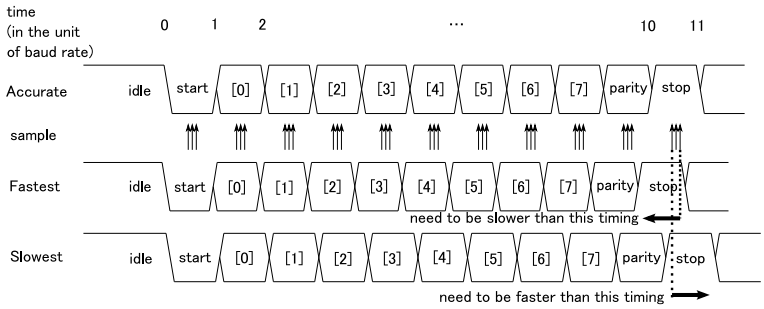

Figure 4.7: Explanation when the UART frequency is not accurate. The data transformation at the fastest / slowest frequency is illustrated.

4.1.6 Reliability of UART

In this section we consider the reliability of UART communication. Since the data transmission in UART is an asynchronous, we need to be careful about the accuracy of the system clock frequency used in the modules. If the system frequency is not accurate, the UART baud rates between the sender and the receiver in the electronics system will not agree, resulting in a communication failure. Refer to figure 4.7. Let us consider the case that the actual baud rate goes faster than 57.6 kbps for some reasons. In order to sample the last stop bit accurately, the transition from the stop bit to the next start / idle bit is before $(10 + \frac{9}{16})$ baud rate time. Similarly, in the case of the actual baud rate gets slower, the transition from the parity bit to the stop bit is after $(10 + \frac{7}{16})$ baud rate time. Therefore the error of the frequency is accepted within the following region,

$$f_{\text{accurate}} \times 0.947 = f_{\text{accurate}} \times \frac{10}{10 + \frac{9}{16}} \leq f_{\text{actual}} \leq f_{\text{accurate}} \times \frac{11}{10 + \frac{7}{16}} = f_{\text{accurate}} \times 1.054. \tag{4.1}$$

Now, the clock accuracy of the AVR is ±1% and the temperature variation in the range from -20 to 40 degrees is about ±1.2% [19]. Since the accuracy of the crystal oscillator used for the FPGA is much lower, the actual baud rate never deviates from the range of (4.1) and no errors are expected in the UART communication.

4.2 Power Management

In this section, we describe the BEE power management by the BGI board. Figure 4.8 shows the power architecture in BGI. GESiCA requires 5 V and 3.3 V, while all the other modules in BEE requires only 3.3 V. There is a dual channel switching regulator LTM4616 (Linear Technology, Corp.) in BGI which is set to output 3.3 V; one channel for 3.3VG (G stands for GESiCA) and the other for 3.3VB (B stands for other BEE modules).

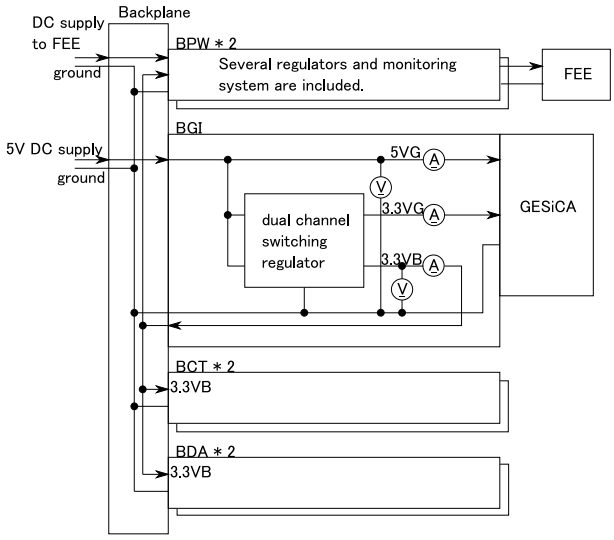

Figure 4.8: Architecture of the power monitoring in BGI

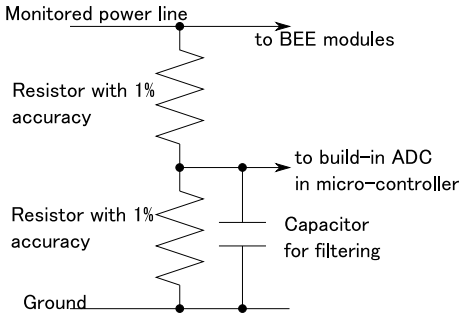

Figure 4.9: Voltage monitoring

The power lines 5VG, 3.3VG and 3.3VB are monitored by AVR ATMega2561 micro-controller on BGI board. The micro-controller has a 10-bit ADC whose absolute accuracy is typically 2.25 LSB [19]. Using this ADC, the voltages and currents of these power lines can be monitored. Note that the voltage of 3.3VB is not monitored because 3.3VB is used for the reference voltage for monitoring.

The input voltage range of the ADC is from 0 to the reference voltage (3.3V). Hence for monitoring the voltage of 5VG and 3.3VB, the target voltages are dropped to adequate levels by the resistors as in figure 4.9 before the ADC conversion. Since the accuracy of the used two resistors is 1% and the accuracy of the 10-bit ADC is 2.25 LSB, the voltage is expected to be measured with an

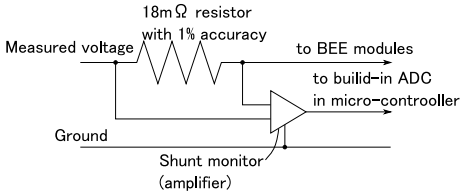

Figure 4.10: Current monitoring

error of

$$\text{accuracy (voltage)} = \sqrt{1^2 + 1^2 + \left(\frac{2.25}{1024} \times 100\right)^2} = 1.4(\%). \qquad (4.2)$$

For the measurement of the current, an 18 m$\Omega \pm 1\%$ shunt resistor is used. The voltage difference of the resistor is amplified with a gain of 20 by a shunt monitor INA196 (Texas Instruments, Inc.) (refer to the figure 4.10). Since the total output error of the amplifier is typically 1% [20], the current is expected to be measured with an error of

$$\text{accuracy (current)} = \sqrt{1^2 + 1^2 + \left(\frac{2.25}{1024} \times 100\right)^2} = 1.4(\%). \qquad (4.3)$$

Note that the purpose of power monitoring is to prevent fatal damage to the devices in the electronics system in an irregular situation. Therefore the accuracy of 1.4% is good enough.

4.3 Generation of CCD and CDS Clocks

GESiCA has a programmable clock generator with 32 output ports. Before the CCD/FEE operation, the user on the DAQ system defines the pattern of the clock in units of 10 ns and uploads it to the clock generator in GESiCA through Ethernet with UDP protocol. The user sends "invoke" command with UDP to GESiCA, then the specified clock pattern is output to the CCD/FEE via the BCT board. Please refer to [18] for the detailed architecture of the clock generator. The BEE system uses this function for sending CCD and CDS clocks. The generated CCD/CDS clocks are handed to two BCT boards and driven to the FEE by the LVDS drivers.

4.4 Image Data Readout from FEE and Data Transmission to the DAQ System

We describe the detailed BEE implementations of the CCD image readout from the FEE and the data transmission to the DAQ system. First we explain the

procedure of generating the ADC clock. The idea for capturing the serial data from the FEE follows. When capturing the data, we faced a skew problem. This challenging issue is solved by a calibration procedure. This solution is described in detail in this subsection. Finally we explain the architecture for transmitting the image data to the DAQ system. The expected maximum readout speed is calculated from this architecture.

4.4.1 Generation of ADC Clock

Difference of BDA master board and slave board

Please recall the sequence for the CCD image readout in figure 3.14. While the two BCTs behaves in the completely same way, the functions of the two BDAs have one difference; the master board generates ADC clock while the slave simply listens to the generated clock. This aim is to convert all connected ADCs at the same time.

Watching the ADC state using ADC_IDLE signal

The master board must know the state of ADCs connected to both the master and the slave board to generate the ADC clock. To communicate the ADC state between the two BDA boards, a two-way communication line called ADC_IDLE is prepared in the backplane of BEE (refer to figure 3.14).

The BDA can know the state of the connected ADCs from their serial outputs, because the serial data from ADCs contain busy signals as shown in figure 3.7. The rising transition represents the completion of the conversion and the falling transition after the ADC data readout means that the ADCs are in idle state.

ADC_IDLE signal is used to inform the state of all connected ADCs to the other BDA board. ADC_IDLE behaves as follows (refer to figure 4.11). ADC_IDLE is set to the logic high by default. Detecting the rising transitions from all 32 serial data channels, the master board drives the ADC_IDLE (master output) to the low level. Receiving the falling transition of ADC_IDLE (master output) from the BDA master board, the slave board waits for the end of the conversions of the ADCs connected to the slave board. After checking that, the slave board replies to the master board by resetting the ADC_IDLE (slave output) to low. The echo-back of the falling transition of ADC_IDLE from the slave to the master represents that all ADCs connected to the BEE system are ready for the data readout. The BDA master board can check the state of all ADCs in that way to generate the ADC clock.

Sequence for generating the ADC clock

Figure 4.11 shows the simplified sequence for generating the ADC clock (Numbers in the figure corresponds to this paragraph). (1) First the trigger pulse for starting the conversion named GCM_CNV is output from the clock generator in GESiCA. When the GCM_CNV trigger comes, (2) the conversion (CNV) signal

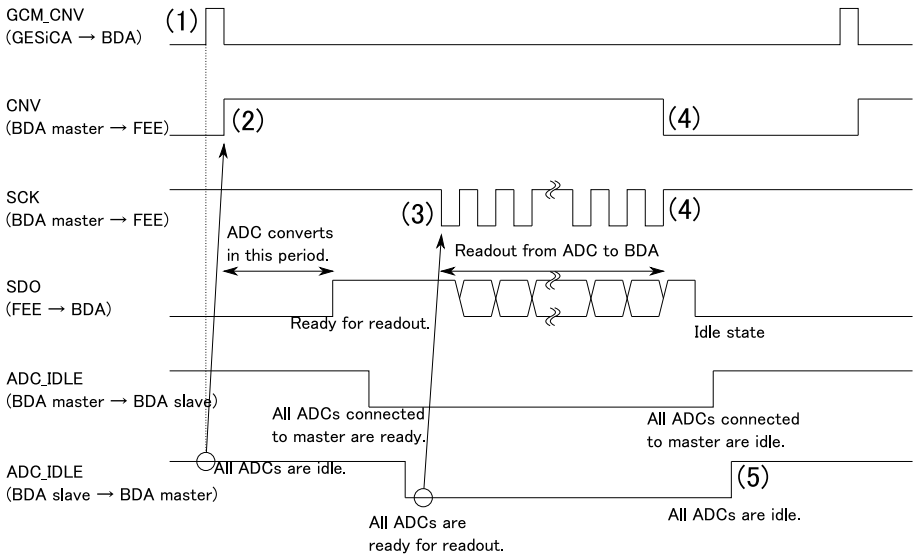

Figure 4.11: Sequence for generating ADC clock. Numbers in the figure corresponds to those in the text.

is generated after checking the ADC_IDLE from the slave board is high (idle state). (3) When the ADC_IDLE from the slave is driven to low, the readout clock (SCK) is sent from the master board to the ADCs. The ADC data are read at 50 MHz, which is the maximum speed for AD7686. (4) At the end of the readout, CNV and SCK are stopped. (5) After all ADCs are checked to be in idle state, ADC_IDLE is returned to the high level. The next conversion is available after this rising transition of ADC_IDLE (slave output).

Flexibility

Other than GCM_CNV there is another signal called GCM_tran which the clock pattern generator in GESiCA sends to the BDA boards. It is known that the readout from ADCs is affected with the FEE noise during the particular period. Although the GCM_tran is normally driven to the high level, it is driven to low in period where readout is forbidden. The BDA then temporarily stops generating the ADC clock until the GCM_tran returns to the high level.

In addition, the BDA are designed to be flexible about the change of ADC structure.

- The readout clock is output normally at 50 MHz (maximum speed), but it can be changed to the slower speed by option.
- Four ADCs are normally daisy-chained, but it can be optionally changed up to eight.

Note that in that situation, it is not guaranteed for the readout speed to be faster than 243kSPS. There are several internal parameters to control such as the ADC frequency, and these parameters can be specified using UART.

4.4.2 Capturing the Serial ADC Data

Challenging Situation - Skew Problem

Once the ADC readout clock is generated, the 64-bit serial data for each CCD are sent to the BDA at 50 MHz. As one BDA handles up to 32 CCDs, there are 32 input channels for each BDA board.

Here there is a challenging problem at the readout; the skew between the 32 serial data makes the readout difficult. The skew is considered to be caused mainly by the following:

- the difference of the length of the transmitting lines
- the dispersion of the propagation delay between the used ICs.

Let us make a rough estimation of the skew length. The difference of the transmission length on the printed circuit board is the order of a few tens of centimeters. Since the transmitting speed on the printed circuit is 15.4 cm/ns for the epoxy laminate board [21], the skew can be as large as 5ns. As for the difference of the propagation delay between the used devices, the output skew of the LVDS driver / receiver is 0.6ns at maximum ([22] - [26]). Since we use the LVDS driver / receiver at several times (on the BEE backplane interface and the FEE-BEE interface) and there is a contribution from the ADC output skew, we should estimate the skew to be several ns. Adding these two contributions together, the skew may be larger than 10 ns, which is not negligible against the readout frequency (50MHz = 20ns).

Additionally, the master and slave board are in different situations. When the slave receives the ADC clock generated by the master board, a propagation delay time arises. Therefore we need a mechanism to sample the ADC data at adequate timing for each channel.

Solution - Architecture of Capturing the Data

One of the solutions is that the FEE returns the readout clock ADC_SCK for every 116 CCD lines. But this is not practical, considering the number of signals drawn to the dewar are limited by the mechanical system.

We designed the architecture illustrated in figure 4.12. The basic idea is that the BDA samples the serial data at 200 MHz double data rate (effectively 400 MHz), which means the sampling is taken by 2.5 ns. On the other hand, ADC_SCK is sampled at 100 MHz double data rate. We use the data of the (delay parameter) $\times$ 2.5 ns after the falling edge of ADC_SCK. Each channel has an individual delay parameter so that the sampling timing can be specified for each channel. These delay parameters are determined by a calibration procedure (described soon).

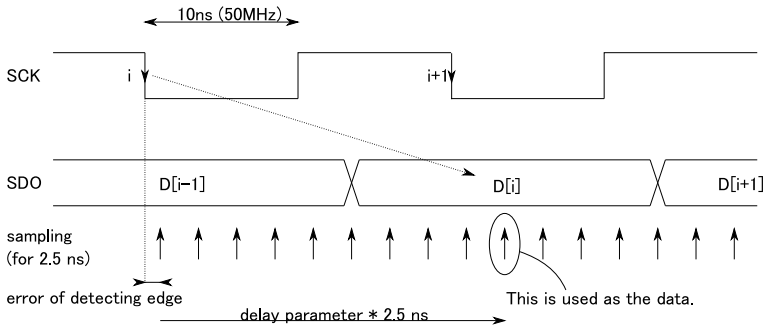

Figure 4.12: Basic idea for capturing the ADC data with skew

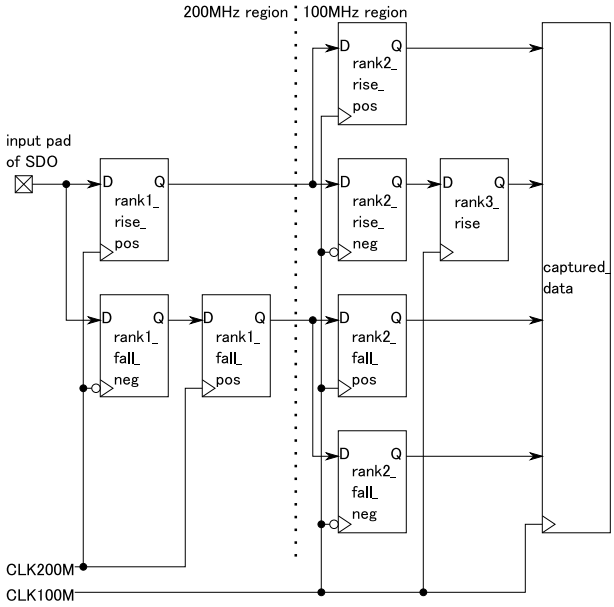

Figure 4.13: The unit for capturing the serial data from FEE

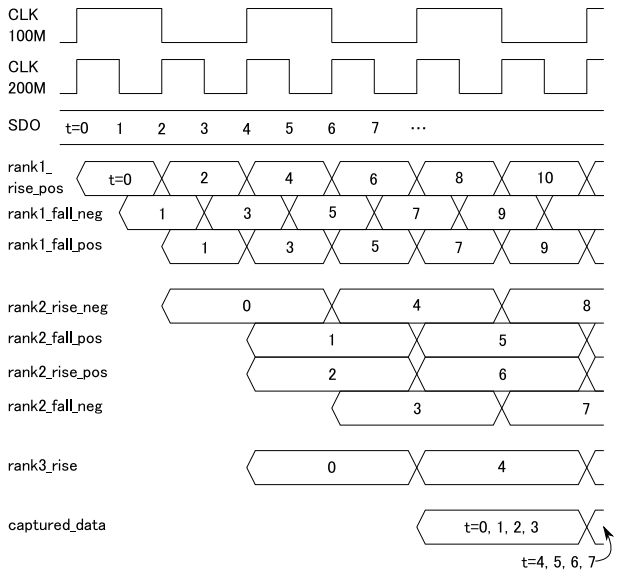

Figure 4.14: The data flow at the capturing unit. The continuous four data are taken into the final register at the same time.

Let us look at the architecture in more detail. The unit for capturing the serial data at 200 MHz double data rate is shown in figure 4.13. The data are first taken into the two registers clocked by 200 MHz. One register takes the data at the rising edge of the clock and the other at the falling edge. In this way the serial data is taken effectively at 400 MHz. After that, the data are handed down to the registers clocked at 100 MHz, which is a system clock. 200 MHz clock is synchronized with the 100 MHz system clock, so that the flow of the captured data can be written as figure 4.14.

The data captured in the above circuit at (delay parameter) $\times$ 2.5 ns after the capturing logic detects the falling edge of ADC_SCK are used as the true data.

The delay parameters can be calibrated for each channel through the following procedure. First, we add 25 - 50% of the full ADC input voltage for the bias voltage in FEE (refer to the figure 3.5). We can then fix the most significant bit of the ADC output as low, while the next significant bit as high. Since the accuracy of the 14-bit DAC for adding the bias voltage is 1 LSB [27], that is $1/2^{14} \times 100 = 0.006(\%)$, it is fully possible to control the output of 2-bit from the MSB. Setting the analog input to the ADC as above, the readout from ADC becomes as illustrated in figure 4.15. Starting the sampling from the detection of the falling edge of ADC_SCK, the captured data continues to high until MSB arrives. Low data are expected to be taken to the FPGA internal logic seven to nine times, followed by the high level of the next significant bit. To sample

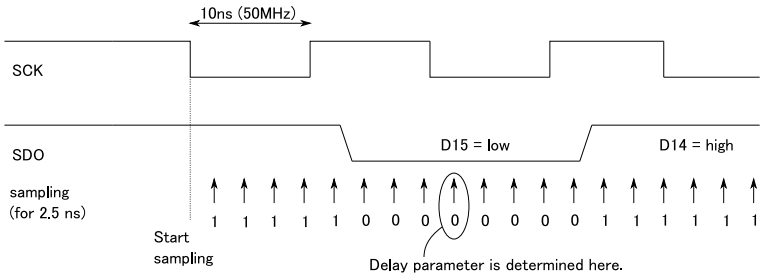

Figure 4.15: The method for calibrating the delay parameter. The delay parameter is set to 9 in the above case.

the data at the middle of the eye of the data, we set the delay parameter as (number of high before MSB) + (number of low = MSB) / 2. Repeating the calibration procedure for every CCD channel, all data from the FEE can be sampled at adequate time.

Now, let us consider the validity of the method. As we sample the data for 2.5 ns, the accuracy of finding the center of the data eye is 2.5 ns. In addition, since ADC_SCK is sampled at 100 MHz double data rate to detect its falling edge, the starting point for adding delays gets blurred for 5 ns. Therefore we may sample the data 7.5 ns away from the center of the data eye in the worst case. There is still room for 2.5 ns to sample the data which is longer than the setup / hold time of the FPGA [28]. Hence we can expect that the serial data are stably captured in the FPGA.

The results of the calibrations are shown and their validity are roughly discussed in the appendix B.

4.4.3 Transformation and Transmission of the ADC Data

Architecture

The data format from the FEE is a 64-bit serial for each CCD. GESiCA can accept however 16-bit parallel data, which is the size of the ADC resolution. The BDA converts the 64-bit image data to the 16-bit parallel data and sends them to GESiCA. The 16-bit image data is temporarily stored to a frame memory mounted in GESiCA, before GESiCA sends the whole image data to the DAQ system. The transmitting architecture in GESiCA is described in [18]. Here we explain the transformation and transmission of the ADC data in the BDA board.

The data transformation from 64-bit serial data to 16-bit parallel is described in figure 4.16. In the BDA FPGA, two types of memories are prepared. One is called preMEM, whose width is 128 bits and depth is 1 bit. This memory works as a buffer for the captured data of one CCD and 32 preMEMs are used in total. The other memory is called SDtoPD_MEM, whose width is 128 bits and depth is 32 bits. The serial data are converted to the parallel data through

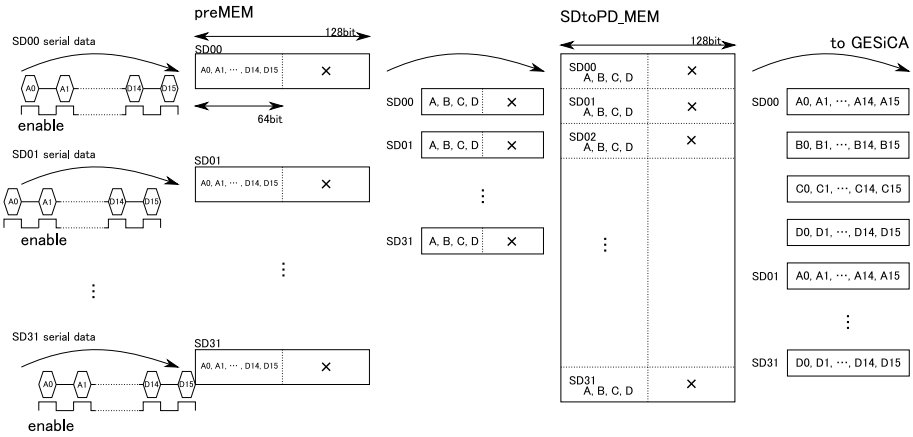

Figure 4.16: Transformation from serial to parallel

this memory. Here the width of these memories are 128 bits so that BDA can handle up to eight daisy-chained ADCs as stated in section 4-4-1. The 64 bits of the second half are set to zero for the normal readout.

The serial data are sent from the capturing unit described in 4-4-1 with the enable signal. They are first written to the preMEM one by one. After all data are written into the preMEM, they are handed down to SDtoPD_MEM for one clock per CCD channel at a speed of 100 MHz. Since there are 32 channels in one BDA board, the transmission from the preMEM to SDtoPD_MEM is completed in 0.32 μs (= 10 ns $\times$ 32). As the output data format from the BDA to GESiCA must be 16-bit parallel, SDtoPD_MEM is read in units of 16 bits. Four ADC data from the first CCD are sent to GESiCA at the beginning, followed by the second CCD data and so on as shown in figure 4.16. The 64 bits of the second half are normally not read from the SDtoPD_MEM.

In the transmission from the BDA board to GESiCA, we need a mechanism for both BDAs not to send their data at the same time. For that purpose, a two-way communication line called STAFF is prepared in the backplane of the BEE (see figure 3.14). STAFF is a kind of token and a BDA with STAFF can transmit data to GESiCA. At the beginning, the master BDA board has STAFF and the 32 CCD data are sent from the master board to GESiCA. After the master completes its transmission, STAFF is handed from the master to the slave board and the slave board starts to transmit its data to GESiCA. STAFF is returned to the master board at the end of the slave transmission to GESiCA. The data are sent at 80 MHz clock, which means it takes $12.5(\text{ns}) \times \frac{64}{16} \times 32 = 1.6\mu$s for each BDA board.

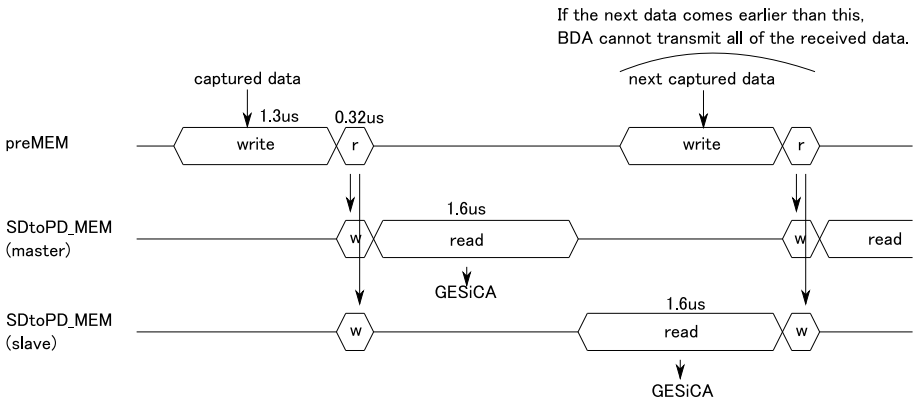

Figure 4.17: Timing chart of the transformation and transmission process at the maximum readout speed

Readout Speed

The readout speed, which is required to be faster than 243 kSPS, is determined by the transformation and transmission process. Figure 4.17 shows the timing chart of the process at the maximum readout speed. It takes 0.32 μs to write the data to SDtoPD_MEM, and 1.6 $\times$ 2 μs to read the data. In addition, about 0.3 μs is needed for system management, such as the state control of the BDA or the handover of STAFF. Adding these together, the maximum readout speed is expected to be about 3.82 μs = 262 kSPS. If the input speed exceeds the maximum readout speed, the next data are written to the preMEM before SDtoPD_MEM completely reads the previous data. As a result, data to the DAQ system were partially dropped. The measurement of the readout speed will be discussed in 5-4-2.

Chapter 5

Performance Measurement of the BEE System

In this chapter, we discuss the performance tests of the BEE. We evaluated the individual functions of the BEE using one FEE board without CCDs. The discussion proceeds in the order of the list in section 3-3-1: UART, power management, CCD/CDS clocks, and image readout and data transmission to the DAQ system.

5.1 BEE/FEE Control by UART

We explained in section 4-1-6 that the UART communication in the developed electronics system is expected to be stable enough. To confirm the stability of UART, we carried out the following two measurements:

- measure the signal pattern using oscilloscope
- repeat the UART communication 10,000 times to check if any error happens.

5.1.1 Signal Pattern

The former measurement is for confirming that UART signals are not affected from the external noise. The setup is illustrated in figure 5.1. One FEE board was connected to the BEE with a 3-meter cable which is planned to be used in the dewar. The DAQ system sent the character '5' to the electronics and its pattern at the input of the LVDS receiver on FEE was observed by the oscilloscope DPO4034 (Tektronix, Inc., the bandwidth is 350 MHz). Character '5', whose ASCII code is 0110101, was selected because in our system the character is sent as

0(start), 1(D0), 0(D1), 1(D2), 0(D3), 1(D4), 1(D5), 0(D6), 0(D7), 0(parity), 1(stop)

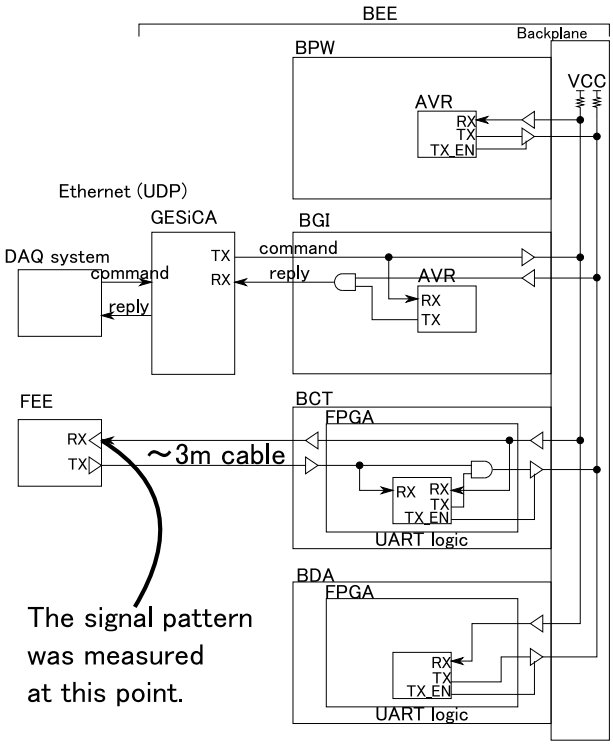

Figure 5.1: Setup for measuring the signal pattern of UART

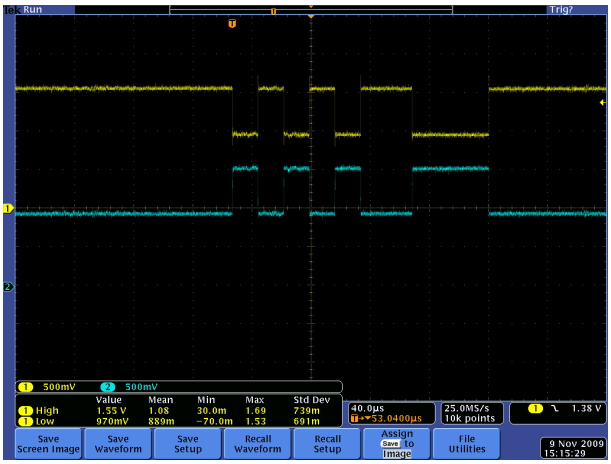

Figure 5.2: UART pattern observed with an oscilloscope. The yellow line is the positive side of LVDS signal, while the blue is the negative side.

in sequence which includes both short and long pulses and is suitable for the observation.

The result is shown in figure 5.2. Since we adopted LVDS, two lines are shown in the figure. The pattern was in perfect shape as expected. The skew between the two lines was not observed. We concluded that there is no problem in the signal pattern of UART.

5.1.2 Stability Test

The second measurement is the stability test. A "model" command to the FEE is for asking the model number. When the FEE receives the command, it replies "HSCFEE". Another command "serial" is for asking the serial number of the FEE board. When the FEE receives the command, the serial number of the FEE board is returned. Since fixed messages are replied by the FEE with these commands, we can check on the DAQ system if the right messages are received. We repeated the "model" and "serial" commands by turns and checked the replied messages on the DAQ system.

As a result, no error was detected in the 10,000 communications. We concluded that UART communication is surely stable.

5.2 Power Management

In this section, we mention the power monitoring for the BEE, carried out in BGI board. We first explain the voltage measurement by the micro-controller on BGI. Then we proceed to the current measurement.

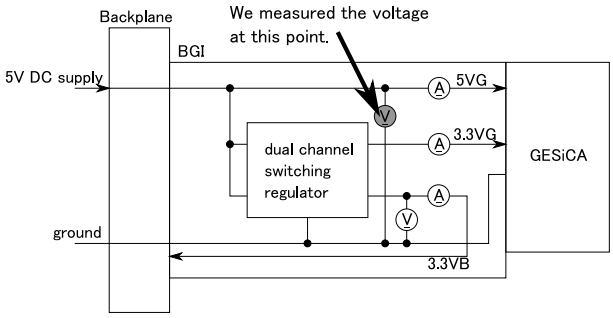

Figure 5.3: Setup for the voltage measurement

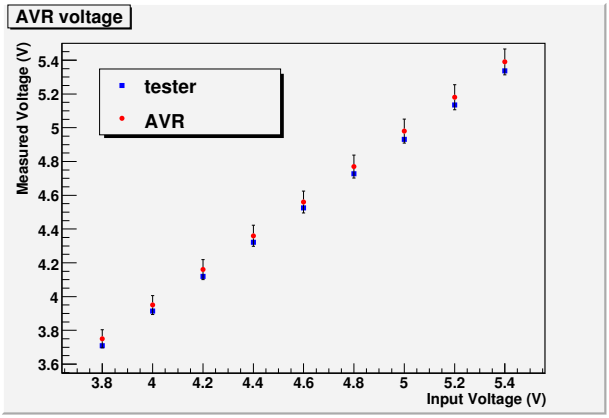

Figure 5.4: The result of monitoring voltage

5.2.1 Voltage Measurement

Setup

As shown in figure 5.3, the monitor point 5VG is directly connected to the DC power supply from the backplane. We compared the voltage measured by the AVR with that by the tester Fluke 187 (Fluke Corp.) on the 5VG monitoring point. Here we must recall that the reference voltage of ADC in AVR is 3.3V generated by a switching regulator LTM4616 (Linear Technology, Corp.). According to the datasheet [29], the voltage from 3.8V to 5.5V as a DC supply from the backplane is acceptable for the regulator to output 3.3 V. Hence we measured the voltage while varying the DC supply from the backplane in that range.

Result and Discussion

The result of monitoring the voltage is shown in figure 5.4. The error bars of the AVR readout are 1.4 % as in equation (4.2) and those of the tester are 0.025

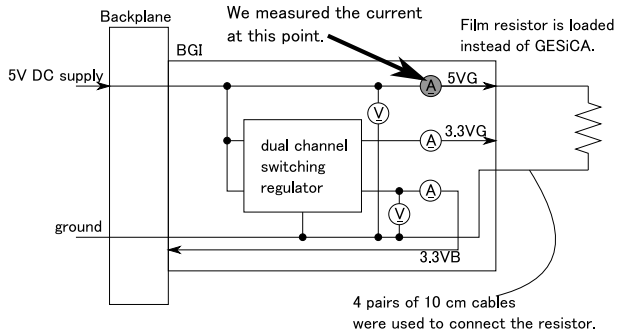

Figure 5.5: Setup for the current measurement

% from the datasheet [30]. The voltage measured by AVR completely agrees to that by the tester in the range of the error bar. Hence we concluded that the voltage monitoring for detecting an irregular voltage works well in this system.

5.2.2 Current Measurement

Setup

The setup for the current measurement is illustrated in figure 5.5. The 5VG monitoring point was used. To control the current, we loaded power film resistors MP2060 (Caddock Electronics, Inc.) instead of mounting GESiCA. 5 V DC was supplied from the backplane as in normal operation, which was read at 5.04 V by the AVR. 5.04 V / (loaded register) is expected as the measured current by the AVR. Film resistors of $1\Omega, 2\Omega, 5\Omega$ and 50Ω were loaded, corresponding to 5 A, 2.5 A, 1 A and 0.1 A, respectively. Their accuracies are 1 % [31], and they were connected to the BGI with four pairs of cables considering the loss of voltage.

Result and Discussion

The result of monitoring the current is shown in figure 5.6. The solid line is the theoretical line. The error bars of the AVR readout are 1.4 % as in equation (4.3). Although the current tends to be lower than expected, the measured value monotonically increases, which is a sufficient condition for detecting overcurrent. We do not have to read the current very accurately because the main purpose of monitoring is to prevent the fatal damage of the devices when excessive current flows for some reason. We are planned to limit the power supply around 3 to 4 A. The developed back-end electronics system can detect (at least) about 5 A, so that we concluded that we can use this monitoring system as a current monitor.

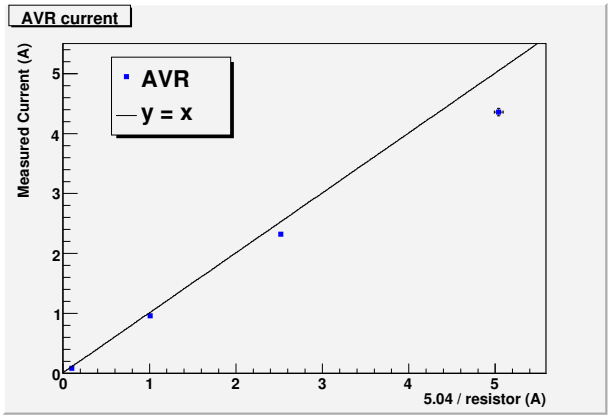

Figure 5.6: The result of monitoring current

5.3 Generation of CCD and CDS Clocks

The noise from the CDS clock jitter is required to be much lower than 3 electrons. The function of generating the CCD and CDS clock is included in GESiCA. The jitter of the CDS clock at the output of GESiCA is described in [18]. Here we measured the jitter at the input of the CDS integral circuit on FEE (refer to figure 5.7). We performed two types of the jitter measurement. First we measured the CDS clock jitter, while varying the integration period. Next, we measured the period of CDS clock, while varying the temperature, since the BEE system is designed to be operated in the range of -20 to +40 Celsius degrees.

Note that this section mentions the direct jitter measurement by an oscilloscope. We can also calculate the jitter from the ADC readout, which is described in section 5-4-1(2) and 5-4-1(3).

5.3.1 CDS Clock Jitter

Theoretical Expectation

As explained in section 3-3, an rms jitter of the CDS clock causes noise in the electronics. Here let us discuss how large the jitter affects the noise.

In an integration circuit in a CDS, a constant current $I_{\rm in}$ is integrated for an integration period T. An output from the CDS $V_{\rm out}$ is written as

$$V_{\rm out} = -\frac{1}{C}\int_0^T I_{\rm in}{\rm d}t = -\frac{T}{CR}V_{\rm in} \equiv -\alpha T V_{\rm in} = -gV_{\rm in} \tag{5.1}$$

where C, R is a capacitance and resistance of the circuit and $g \equiv \alpha T$ is a gain of the integration circuit. T has actually a jitter δT which is defined as the root mean square of T. From (5.1), the jitter affects the output $V_{\rm out}$ as

$$\delta V_{\rm out} = \alpha \delta T |V_{\rm in}|. \tag{5.2}$$

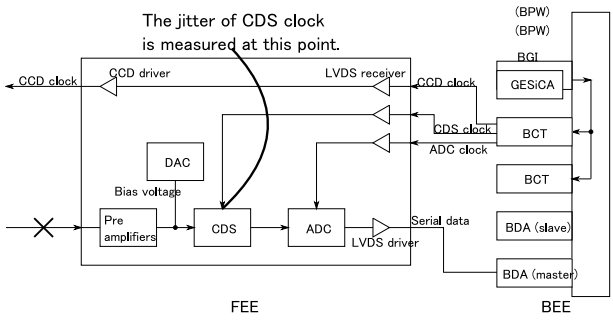

Figure 5.7: Setup for measuring the CDS clock jitter

$\alpha\delta T$ is interpreted as a fluctuation of the CDS gain. Using a mean of the integration time $\bar{T}$, a mean of the output is written as $\bar{V}_{\rm out} = -\alpha\bar{T}V_{\rm in}$. Hence (5.2) can be rewritten as

$$\delta V_{\rm out} = \frac{\delta T}{\bar{T}}|\bar{V}_{\rm out}|. \tag{5.3}$$

Now we must recall that the integration is carried out twice in order to find the difference between the reset part and the data part of the CCD signals (refer to 3-2). With $\bar{V}_{\rm out,\,1}, \bar{V}_{\rm out,\,2}$ as the mean output of the two parts, the noise of the CDS circuit is expressed as

$$N_{\rm CDS} = \sqrt{\bar{V}^2_{\rm out,\,1} + \bar{V}^2_{\rm out,\,2}}\frac{\delta T}{\bar{T}} \tag{5.4}$$

under the assumption that the jitter is Gaussian.

Setup

We measured the rms jitter of CDS clock at the input of the CDS amplifier on FEE (refer to figure 5.7). The integration is carried out normally for 2 μs, but we also measured the jitter while varying the integration time from 0.5 to 4 μs by uploading different clock patterns to the clock generator in GESiCA. To measure the jitter, a Tektronix DPO7104 oscilloscope whose bandwidth is 1 GHz and sampling rate is 20 GSampling/sec, and DPOJET, a software for analyzing jitter, are used. During the measurement, the temperature of the BEE is kept at 20 degrees. The data are sampled 10,000 times.

Result and Discussion

The figure 5.8 shows the result of measuring the rms jitter. The horizontal axis is the CDS integration time, while the vertical axis is the rms jitter in units of ps. A relative accuracy, which is a ratio of rms jitter to the integration time, is replotted in units of parts per million (ppm) in figure 5.9. Error bars come from the measurement accuracy announced in the datasheet of DPOJET [32].

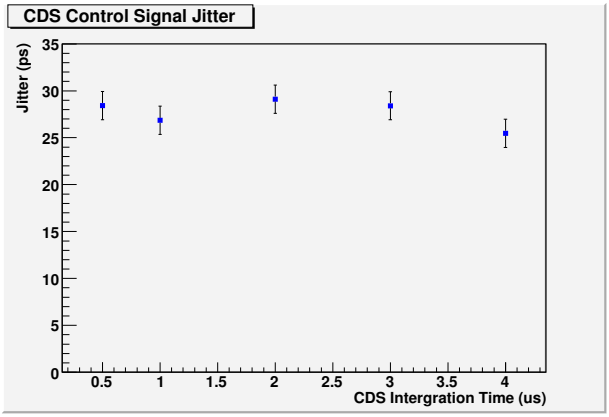

Figure 5.8: The result of measuring the rms jitter in units of ps

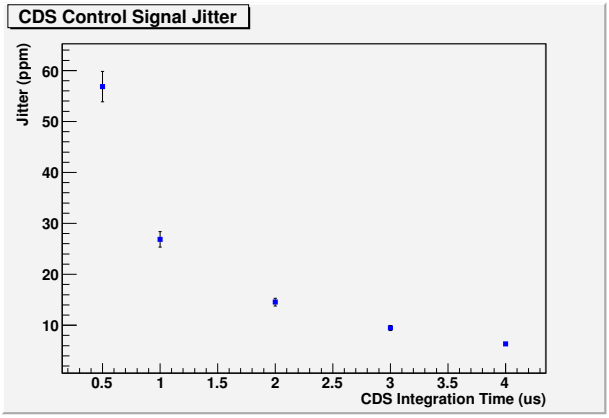

Figure 5.9: The result of measuring the rms jitter in units of ppm

The readout noise caused by the CDS clock jitter must be much lower than 3 electrons in a few electron region ($\sim$ 1,000 ADC counts; See section 3-3.). From the figure, the CDS jitter is $\sim$ 15 ppm when the integration time is 2 μs. Using (5.4), the noise in a few electron region is

$$N_{\text{jitter}} \sim \sqrt{1000^2 + 1000^2} \times 15 \times 10^{-6} = 0.021(\text{ADC count}). \tag{5.5}$$

Since the fullwell capacity of a CCD is $\sim$ 200,000 electrons and the ADC resolution is 16-bits, the conversion factor is $200,000/2^{16} \sim 3$ electrons / ADC count. From the above, the noise caused by the jitter of the CDS clock is

$$N_{\text{jitter}} \sim 0.021 \times 3 = 0.063\text{e}^- \ll 3\text{e}^-. \tag{5.6}$$

This is much lower than the requirement of 3 e^- and we can conclude that the requirement was satisfied.

5.3.2 CDS Period Variation over Temperature

Setup

We should also pay attention to the surrounding temperature since the BEE is required to be operated in the temperature from -20 to +40 Celsius degrees. The source of the CDS clock is a low-jitter crystal oscillator on GESiCA. If the frequency of the crystal oscillator varies with the temperature, the period of the CDS clock also varies, resulting the electronics noise.

We put the BEE system in a temperature chamber and varied the temperature in the range of -20 to +40 degrees. The integration time is fixed to 2 μs, which is the normal operation. The CDS clock is measured again at the input of the CDS amplifier by the oscilloscope DPO7104. The data are sampled 10,000 times.

Result and Discussion

Figure 5.10 shows the result of the period variation of the CDS clock over temperature. The error bars are determined by the accuracy of the oscilloscope.

The figure shows that the period differs 0.0002 μs over the operating temperature. The CDS gain varies in the order of $0.0002/2 = 10^{-4}$. In a few electron region, this gain variation corresponds to

$$N_{\text{temp}} = 10^3 \times 10^{-4} = 0.1(\text{ADC count}) = 0.3\text{e}^- \ll 3\text{e}^- \tag{5.7}$$

which satisfies the requirement. We concluded that the gain variation over temperature is satisfactory small.

5.4 Image Data Readout from FEE and Data Transmission to the DAQ System

In this section, we divide the BEE performance measurement of the readout function into two subsections; the former discusses the validity of the data read

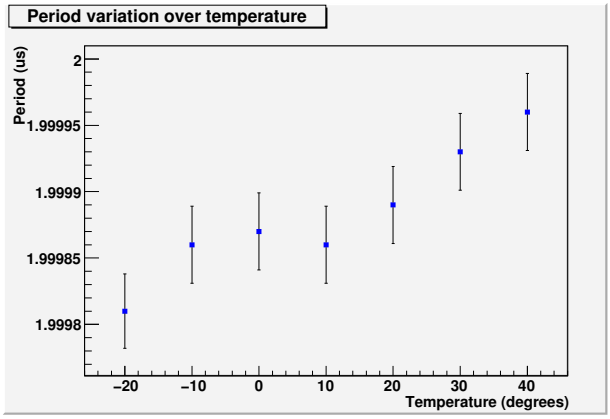

Figure 5.10: The result of measuring the rms jitter over temperature

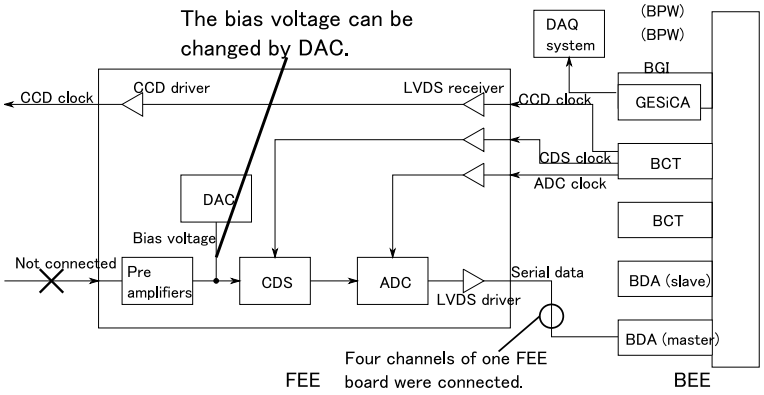

Figure 5.11: Setup for readout from FEE and transmission to the DAQ system. This setup is common to all measurements in section 5-4.

by the BEE and the latter discusses the readout speed. The former is comprised of three measurements. The first is a noise measurement, the second is a measurement of linearity and CDS clock jitter, and the last is a measurement with temperature variation.

5.4.1 Appropriateness of the Data

(1) The Readout Noise of the Electronics System

Setup

Firstly, we measured the total readout noise of the electronics system including the FEE and BEE. We evaluated the readout noise for a constant low level voltage input to the ADC. Without connecting CCDs, we controlled the analog input of ADCs, adding the bias voltage by DACs as illustrated in figure 5.11.

# of ADC	entries	mean	std.dev	minimum	maximum
0	2005712	766.783	0.865	763	771
1	2005712	739.935	0.853	736	744
2	2005712	816.403	0.863	812	821
3	2005712	759.751	0.857	755	764

Table 5.1: Readout noise of 0th channel in units of ADC counts

# of ADC	entries	mean	std.dev	minimum	maximum
0	2005712	884.934	0.859	881	889
1	2005712	881.304	0.861	877	885
2	2005712	869.275	0.904	865	874
3	2005712	894.466	0.855	891	899

Table 5.2: Readout noise of 3rd channel in units of ADC counts

In this measurement, we set the 14-bit DACs to output 0.047 V, which is 1/64 of the maximum 3.0 V and corresponds to the "few electron region". One FEE board which handles 4 CCDs was connected to the BEE system so that four channels of data were used for estimating the noise. The temperature was kept at 20 degrees and the data from the FEE were converted more than 2 million times, which corresponds to the size of one image shot, at 100 kSPS. (Recall that there are four output from CCDs and their CCDs are daisy-chained. Since one CCD has 2k $\times$ 4k = 8M pixels, ADCs are converted 8M/4 = 2M times in one shot.)

Result and Discussion

Table 5.1 is the result of the total electronics noise from the 0th channel. The statistics were calculated separately in each ADCs. The table shows that the standard deviation from the mean value is in the range of about 0.85 to 0.90 ADC counts. Moreover all two million data are within five ADC counts from the mean value and there are no outliers in every ADCs. This applied to all measured channels. For example, the data of the third channel are shown in table 5.2.

The fact that the standard deviations are in the range of 0.85 to 0.90 means that the total electronics noise including the FEE and BEE is as large as 0.85 to 0.90 ADC counts. This is equivalent to about 2.5 - 2.7 electrons noise, satisfying the requirement of 3 electrons. It is also important that there were no outliers in each channel, which is necessary for stable readouts. Note that we do not care that the mean values varies in different ADCs. The variation comes from properties of the FEE such as the small difference of the offset of the amplifiers and is able to be removed by an analysis process in the DAQ system.

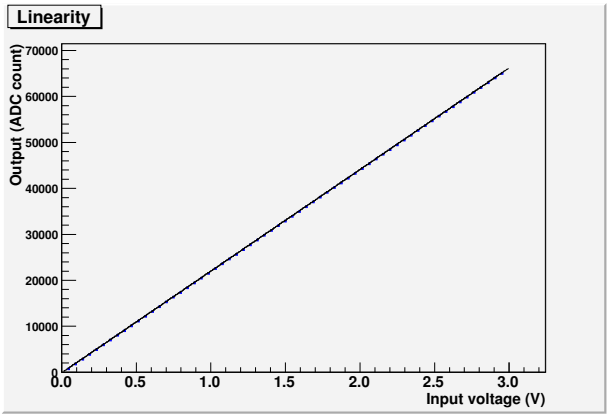

Figure 5.12: The result of measuring the linearity (0th channel)

(2) Linearity

Setup for Linearity Measurement

Next, we measured the linearity. The setup for this measurement is the same as the above except that we varied the analog input of ADCs by changing the bias voltage driven by DACs. We sampled 64 points in the bias voltage range from 0 V to 3 V (full range). This measurement clarifies not only the linearity of the electronics system but also the absence of any transmission error. If there was a bit defect during the transmission, the plotted result would not be linear.

Result and Discussion of Linearity Measurement

Figure 5.12 shows the result of the linearity measurement. The horizontal axis is an analog input to ADC in units of voltage, while the vertical axis shows the ADC counts read by the BEE. Each plot contains two million data from the 0th ADC in 0th CCD channel. The standard deviations are plotted as error bars but they are not able to be seen in the figure. The data are fitted to the equation of $y = ax + b$ using a chi-square method and the fitted line is drawn in the figure.

From the figure we can tell that the data are well-fitted to a straight line, which implies that there are no bit defects during the readout.

Figure 5.13 shows the residual distribution between the measurements and the fitted line. The data are again plotted in units of ADC counts and the standard errors are written as the error bars. This plot shows that the error from the fitted line is only 3 ADC counts at most. From the datasheet, the accuracy of the analog input driven by the DACs corresponds to 4 ADC counts in the worst case ([27]), and the accuracy of ADCs are 2 to 3 ADC counts at maximum ([13]). Therefore the measured three ADC counts error can be described by the FEE component.

The linearity was maintained in all ADCs in all channels. For example, the

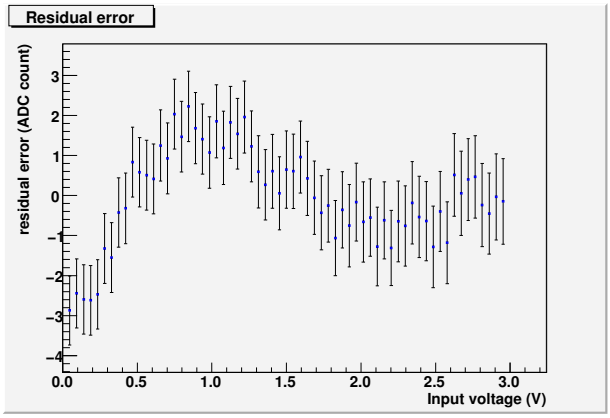

Figure 5.13: The residual error from the fitted line (0th channel)

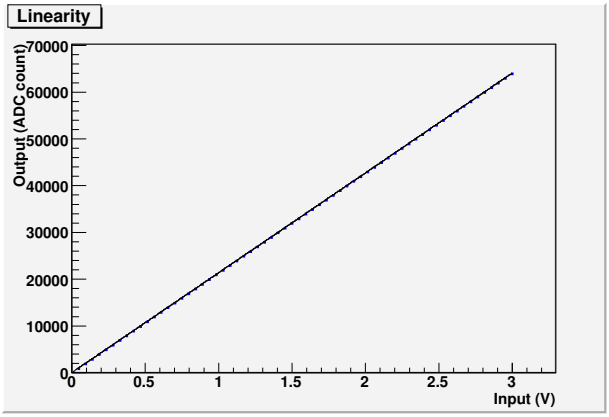

Figure 5.14: The result of measuring the linearity (3rd channel)

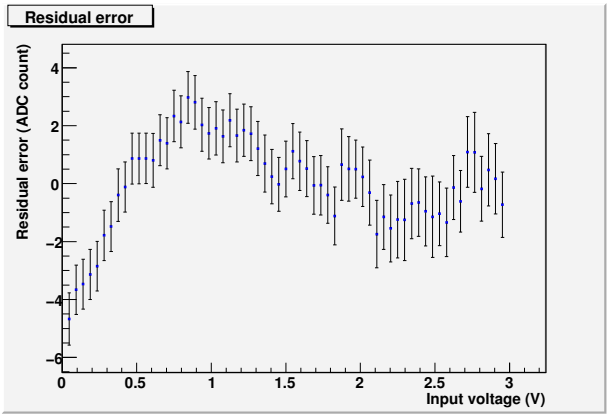

Figure 5.15: The residual error from the fitted line (3rd channel)

linearity plot for the 2nd ADC in 3rd CCD channel is shown in figure 5.14 and its residual distribution is plotted in figure 5.15.

From the above figures we concluded our readout system satisfies the requirement that the linearity must be less than the range of ±1%.

Theoretical Expectation for CDS Clock Jitter

It is not only the linearity that can be measured by varying the bias voltage. The CDS clock jitter can also be estimated from this observation. The total readout noise can be described as

$$N_{\text{total}} = \sqrt{N_{\text{int}}^2 + N_{\text{CDS}}^2} \tag{5.8}$$

where N_{int} is the intrinsic noise that the electronics system has (such as the accuracy of ADC) and N_{CDS} is the noise caused by the CDS clock jitter generated in the process of integration. Since N_{CDS} can be described as the equation (5.4), N_{total} can be rewritten as

$$N_{\text{total}} = \sqrt{N_{\text{int}}^2 + \bar{V}_{\text{out}}^2 \left(\frac{\delta T}{\bar{T}}\right)^2} \tag{5.9}$$

where $\bar{V}_{\text{out}}$ is the mean output voltage. We can suppose that N_{int} and $\delta T/\bar{T}$ are independent of $\bar{V}_{\text{out}}$. Then if we plot the readout noise against $\bar{V}_{\text{out}}$, the CDS clock jitter in the form of $\delta T/\bar{T}$ emerges as the gradient of the curve.

Result and Discussion of Clock Jitter

The noise curve is shown in figure 5.16. The horizontal axis is the mean ADC counts read by BEE. The vertical axis is the total readout noise in units of ADC counts, which is the very same as the magnitude of the error bars in figure 5.12

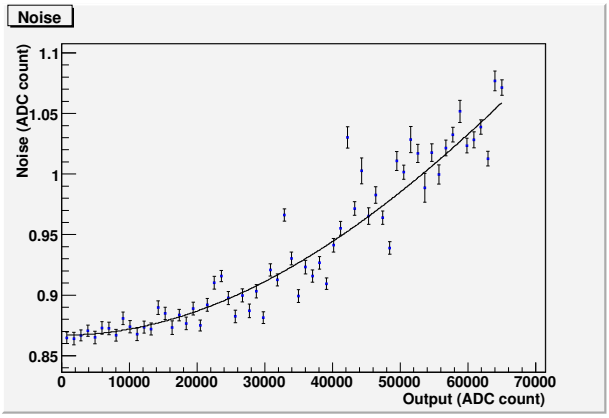

Figure 5.16: The readout noise when the analog input is varied

and figure 5.13. The error bars in figure 5.16 are the standard deviations which are re-calculated, treating the two million data as 129 sets of data including about 15,500 points. The data points are fitted to a curve $y = \sqrt{a^2 + b^2 x^2}$ using the chi-square method. As a result, the following parameters were obtained:

$$N_{\rm int} = 0.866 \pm 0.001(\text{ADC count}) \sim= 2.598 \pm 0.003\text{e}^- \tag{5.10}$$

$$\frac{\delta T}{\bar{T}} = 9.35 \pm 0.06(\text{ppm}). \tag{5.11}$$

$N_{\rm int}$ was already discussed in the previous subsection which satisfies the requirement. $\delta T/\bar{T}$ shows that the CDS clock jitter is 9.35 ppm. This corresponding to

$$N_{\rm jitter} \sim \sqrt{1000^2 + 1000^2} \times 9.35 \times 10^{-6} = 0.013(\text{ADC count}) = 0.04\text{e}^- \tag{5.12}$$

in the few electron region. Comparing the result of the direct measurement in section 5-3-1, the result by the method using the electronics readout system was consistent to first-order estimation. One possible reason that the direct measurement is a little larger is that noise other than the clock jitter affects the measurement by the oscilloscope. In any case, we can conclude that the noise from the CDS clock jitter is much smaller than 3 electrons, which satisfies the requirement.

(3) Readout over the Operating Temperature

Setup and Expectation

The third readout measurement was carried out while varying the temperature of the BEE. Measurements are carried out from -20 to +40 degrees in 10 degrees increments. Since the purpose of this measurement is the evaluation of the BEE behavior, only the BEE was put in the temperature chamber and

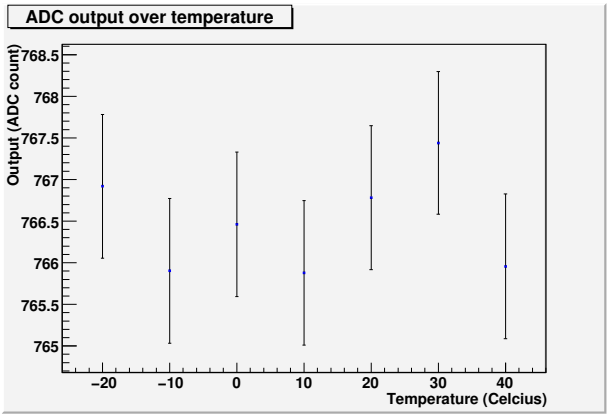

Figure 5.17: Output when the temperature is varied

the FEE was kept in room temperature ($\sim$ 20 degrees). The analog input was returned to 0.047 V, namely in the range of the few electron region. Other conditions such as the number of channels and the number of conversions were the same as the previous.

Since the analog input was kept at constant level, the variation of the CDS clock period is expected to be observed if any. If the period of the crystal oscillator is changed by temperature, the variation of the integral time will be reflected in the output.

Result and Discussion

The result of the readout when the temperature is varied is shown in figure 5.17. The vertical axis is the data read by BEE in units of ADC counts. The error bars are the standard deviations of the 2 million data. The obtained output distribution is flat within the error bars. There is no incremental or decremental structure in the figure. Hence we can say that the output variation caused by the integration period over the temperature is so small compared to the readout noise that they cannot be seen in the figure. This is consistent with the result of the direct measurement in section 5-3-2, which concluded that the noise from the period variation is less than 0.1 ADC counts. From the above discussions we can confirm that the noise from the temperature variation satisfies the requirement.

5.4.2 Readout Speed

This section discusses the measurement of the readout speed. The BEE is required to handle the FEE at more than 243 kSPS (samplings per second). We measured the maximum readout speed as follows.

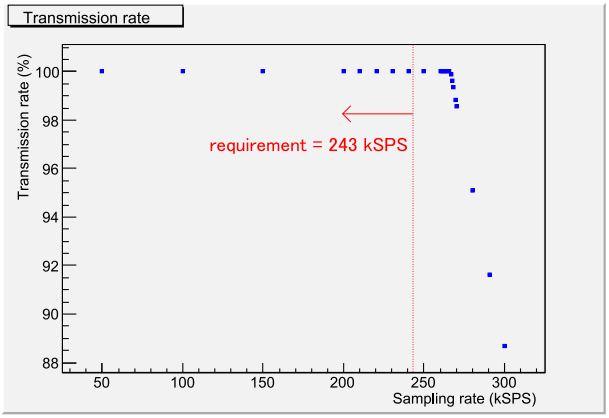

Figure 5.18: The result of measuring the transmission rate

Setup

The setup is the same as illustrated in figure 5.11. One FEE board which can handle four CCDs is connected to the BEE. There are only four inputs of serial data, though we must measure the readout speed when handling the 64 channels. We sent 0xFFFF for the other 60 channels to make up the whole data. Analog inputs of ADCs were set to 0.047V. The BEE was put in a 20 Celsius degree chamber, while the FEE was kept in room temperature. The conversion speed is determined by GCM_CNV generated by the clock generator in GESiCA as described in section 4-4-1. We varied the ADC conversion speed by changing the frequency of GCM_CNV. The conversion was carried out more than two million times, corresponding to a size of one CCD image shot. The whole data size then amounts to

$$2(\text{Byte/conversion}) \times 4(\text{daisy-chain}) \times 2\text{M} \times 64(\text{channels}) = 1\text{GByte}. \quad (5.13)$$

As described in section 4-4-3, when the conversion exceeds the maximum readout speed the BEE drops the data partially. We define the transmission rate as

$$\text{transmission rate} \equiv \frac{\text{total received bytes}}{\text{expected bytes} \simeq 1\text{GB}} \times 100(\%). \quad (5.14)$$

When the conversion speed is below the readout speed, the transmission rate is expected to be 100%. The fastest frequency where the transmission rate keeps 100% is considered to be the maximum readout speed. We measured the received bytes three times for each frequency.

Result and Discussion

The result of measuring the transmission rate is shown in figure 5.18. The horizontal axis is the ADC conversion speed displayed in units of SPS (samplings

per second). The vertical axis is the transmission rate in units of percents. The transmission rates were perfectly 100% under 265.3 kSPS. For data over 266.7 kSPS, the standard deviations are used as the error bars but being 0.0004% at most they cannot be seen in the figure. In the data taken at 265.3 kSPS, the standard deviation is about the same as in section 5-4-1(1) and there were no outliers in any channel.

From the above, we can say that the maximum readout speed is 265 kSPS, which satisfies the requirement of 243 kSPS. This result is consistent with the fact that the maximum readout speed is expected to be around 262 kSPS from the way of the FPGA implementation in section 4-4-3.

Chapter 6

Readout from CCD

The performance of individual functions described in section 3-3-1 was measured in chapter 5, but during the measurements the CCDs were not connected to the FEE system. As a final measurement, we tested the readout from the CCDs to confirm the read operation as a whole readout system.

Setup

Four CCDs were connected to one FEE board. The CCDs and FEE were put in a dewar and they were cooled down to -100 Celsius degrees. Together with these, a LED (light-emitting diode) was equipped in the dewar as in the figure 6.1. LED was directed to the ceiling of the dewar and we took the image of the reflected light. A gradient pattern is expected to be taken if the reflected light is correctly read.

Result

The images from four CCDs are shown in figure 6.2. Each CCD image has black strips because there are several blank or over-scan regions in CCD as shown in table 3.1. Color was added later to represent the magnitude of detected light. The four CCDs were not arranged in the same way as in figure 6.2 so the direction of four CCD images in the figure is not correct.

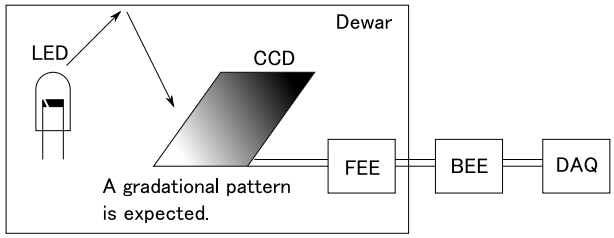

Figure 6.1: Setup for the readout from the CCDs

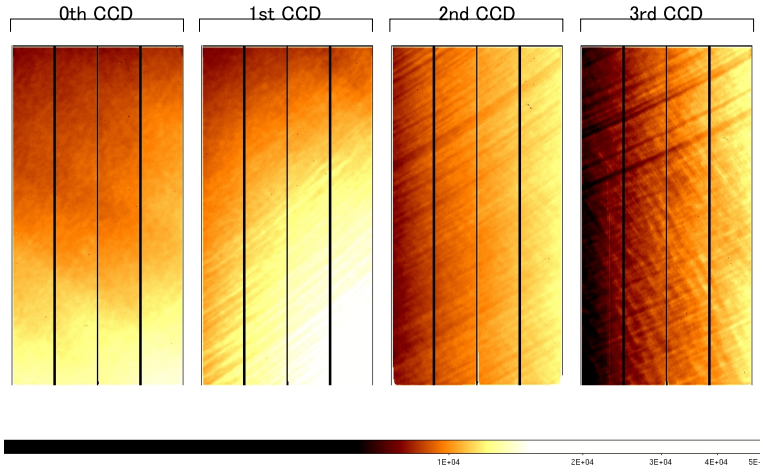

Figure 6.2: The readout image from CCDs

We can see the expected gradient pattern in all CCD images. We concluded that we succeeded in reading the images from the CCDs with the developed electronics system.

Chapter 7

Conclusion

7.1 Summary

We have developed the back-end readout electronics (BEE) for Hyper Suprime-Cam. The main function of the BEE can be divided into four items: UART, power management, clock generation and readout from the FEE. Using the FEE system developed by NAOJ, we measured the BEE performances for individual functions.

- The signal pattern of UART was in perfect shape and passed a 10,000-times stability test with no problems. We concluded that the communication via UART is very stable.
- The monitored voltage of the power line by the micro-controller agreed with measurements by a tester. The current monitoring was also checked and we concluded that the system can detect an abnormal voltage and an excessive current in order to prevent the fatal damage to the devices.
- The noise from the CDS clock jitter is required to be much lower than 3 electrons. The measured result was 0.063 electrons. In addition, the CDS period variation over the operating temperature was 0.3 electrons. We concluded that the noise from the CDS clock is sufficiently small.
- The total electronics noise including the FEE and BEE was found to be 2.5 - 2.7 electrons, which satisfies the requirement of 3 electron limit. The linearity was extremely good and well exceeds the requirement of 1% . The maximum readout speed was measured to be 265 kSPS. This satisfies the requirement of 243 kSPS.

We also performed a system test with a combination of four CCDs, one FEE and one BEE for HSC and we could acquire the expected images. From the above results, we concluded that the developed back-end readout electronics satisfies all requirements for BEE and did not find any problems with the configuration.

7.2 Future Prospects

Although the basic system was established in this research, the following issues still remain.

- BPW board
  The BPW boards which supply power to the FEE modules are not currently mounted on the BEE system. The power monitoring is carried out in the same way as for the BGI boards so that no problems are expected to arise when the BPW boards are mounted to the electronics system.

- Replacing 1000BASE-T with 1000BASE-SX
  Since the distance between the BEE and the DAQ system in the control room is expected to be of a few 100 m, it is necessary to employ optical communication for Gigabit Ethernet. Although the current GESiCA employs 1000BASE-T (unshielded twisted pair cable), this is planned to be changed to 1000BASE-SX (optical fiber).

- Synchronization of readout between three BEEs
  Although we evaluated one readout electronics system, three BEEs are planned to be used. There remains the question of how to synchronize the readout between three BEEs.

After solving these items, we will then have to confirm the readout from all 116 CCDs.

Appendix A

UART ID List and Command Lists

The UART ID numbers are listed in table A.1. The current UART commands are listed in table A.2, A.3 and A.4.

Board	ID number
GESiCA	1
BPW (1)	2
BPW (2)	3
BGI	4
BCT (1)	5
BCT (2)	6
BDA slave	7
BDA master	8
FEE connected to BCT (1)	64 - 79
FEE connected to BCT (2)	80 - 95

Table A.1: UART ID numbers

Command	Explanation
vol0	Read current of the 5VG power line.
vol1	Read current of the 3.3VG power line.
vol2	Read current of the 3.3VB power line.
vol3	Read voltage of the 5VG power line.
vol4	Read voltage of the 3.3VB power line.

Table A.2: UART command list for BGI board

Number	Command	Explanation
0	(reserved)	When the received command does not agree with the pre-defined arguments listed below, this command is invoked and the message "No Such Command." is sent.
1	led_en-w	Write LED display mode. When this parameter is cleared to zero, all LEDs on BCT are turned off.
2	led_en-r	Read LED display mode.
3	latch_mode-w	Write latch mode. When this parameter is set to one, all signals driven from the BCT board are latched until this parameter is returned to zero.
4	latch_mode-r	Read latch mode.
5	FEE_id-w	Write ID number of FEE to connect to BCT.
6	FEE_id-r	Read ID number of FEE to connect to BCT.

Table A.3: UART command list for BCT board

Number	Command	Explanation
0	(reserved)	When the received command does not agree with the pre-defined arguments listed below, this command is invoked and the message "No Such Command." is sent.
1	led_en-w	Write LED display mode. When this parameter is cleared to zero, all LEDs on BDA are turned off.
2	led_en-r	Read LED display mode.
3	adc_freq-w	Write ADC clock frequency parameter.
4	adc_freq-r	Read ADC clock frequency parameter.
5	adc_chain_num-w	Write the number of ADC daisy-chained.
6	adc_chain_num-r	Read the number of ADC daisy-chained.
7	adc_cnv_gen_mode-w	Write ADC convergence mode. When this parameter is set to one, the convergence is carried out by GCM_CNV signal.
8	adc_cnv_gen_mode-r	Read ADC convergence mode.
9	adc_sd_cap_mode-w	Write ADC serial capture mode. When this parameter is set to one, the delay parameters for the data capturing can be changed by the command adc_cap_dly-w.
10	adc_sd_cap_mode-r	Write ADC serial capture mode.
11	adc_sd_valid-w	Write ADC serial data valid bit. The channels connected to FEE can be specified by this command.
12	adc_sd_valid-r	Read ADC serial data valid bit.
13	adc_pd_tran_mode-w	Write ADC parallel data transmission mode. This parameter is cleared to zero by default and only the valid data specified in adc_sd_valid-w are sent to GESiCA. When this bit is set to one, 32 channels of data are sent to GESiCA. This mode is used when the measurement of readout speed.
14	adc_pd_tran_mode-r	Read ADC parallel data transmission mode.
15	calib_start-e	Execute calibration.
16	calib_result-r	Read calibration result. The 64 data sampled from the falling edge of ADC clock with the 2.5 ns interval are sent.
17	adc_cap_dly-w	Write delay parameters for the data capturing. The user specifies the delay parameters, interpreting the output of calib_result_r command.
18	adc_cap_dly-r	Read delay parameters for the data capturing.

Table A.4: UART command list for BDA board

Appendix B

Rough Estimation about the Delay Parameter

As explained in section 4-4-2, the delay time from the falling edge to the MSB of the serial data is measured by the calibration process in units of 2.5 ns. In this appendix, we make a rough estimation whether the delay time measured by calibration agrees with the expected transmission delay announced in the datasheets of the used devices.

B.1 Result of the Calibration

The delay times from the falling edge of the ADC clock to the MSB determined by the calibration process are shown in table B.1. We connected one FEE board for four CCDs so that the delay time is measured in four data channels. In measurement A, a 2-meter cable was used and the delay time was measured by the master board. In measurement B, a 5.6-meter cable was used. In measurement C, the same cable for the measurement A was used, but the delay time was measured by the slave board. Below we discuss how these delay times are explained.

Measurement	ADC 0/1/2/3 Delay (ns)	Mean Delay (ns)	Cable Length (m)	Board
A	60.0 / 60.0 / 57.5 / 57.5	58.8	2.0	master
B	82.5 / 82.5 / 80.0 / 80.0	81.3	5.6	master
C	50.0 / 52.5 / 47.5 / 47.5	49.4	2.0	slave

Table B.1: Result of calibration

Device	Explanation	Location	Input/Output Delay (ns)
Spartan3AN	FPGA output	BDA	max. 4.37
SN65LVDM176	LVDS driver	BDA	min. 0.5, typ. 1.7, max. 2.7
SN65LVDM176	LVDS receiver	BCT	min. 0.5, typ. 1.7, max. 2.7
Spartan3AN	FPGA input	BCT	max. 2.68
Spartan3AN	FPGA output	BCT	max. 4.37
SN65LVDS31	LVDS driver	BCT	min. 1.8, typ. 2.3, max. 2.9
SN65LVDS386	LVDS receiver	FEE	min. 1.0, typ. 2.6, max. 4.0
TC7WZ34FK	buffer	FEE	min. 0.8, typ. 2.3, max. 3.6
AD7686	ADC	FEE	min. 15.0, max. 25.0
SN65LVDS389	LVDS driver	FEE	min. 0.9, typ. 1.7, max. 2.9
SN65LVDS390	LVDS receiver	BDA	min. 1.0, typ. 2.6, max. 4.0
Total (*)			$\simeq$46.3

Table B.2: Input/Output delay of devices concerned with the measured delay time ([13], [22] - [28], [33]).
(*) "Total" is the summation of the typical values except that we take the maximum for FPGA and 20 ns for AD7686.

B.2 Discussion 1 (Measurement A & B)

Time measured by the calibration is comprised of

$$t_{\text{total}} = t_{\text{transmission in cables}} + t_{\text{transmission in boards}} + t_{\text{input/output delay of devices}}. \tag{B.1}$$

The first two items are generally called propagation delay. This propagation delay is determined by the physical structure such as the cable length. Here we measured the delay times with the cable lengths to be 2.0 m and 5.6 m. Hence we can remove the propagation delay of the cables by extrapolating the results of the measurement A and B. The transmission delay for the imaginary cable length of 0 m is calculated to be 46.3 ns. We then have to consider whether this value is consistent with $t_{\text{transmission in boards}} + t_{\text{input/output delay of devices}}$.

First, because of the spacial sizes of FEE and BEE, the line length of the signals are estimated as a few tens of centimeters. The propagation velocity of signals in the practical epoxy laminates is about 14.4 cm/ns [21], the propagation delay time in the boards is estimated as a several ns, say 4 ns.

As for the input/output delay of devices, the devices in table B.2 are concerned with the measured delay time. The total input/output delay of devices is roughly estimated to be about 46 ns(, though its uncertainty is larger than 10 ns).

From the above, the expected delay time is roughly

$$t_{\text{transmission in boards}} + t_{\text{input/output delay of devices}} \simeq 4 + 46.3 = 50.3(\text{ns}), \tag{B.2}$$

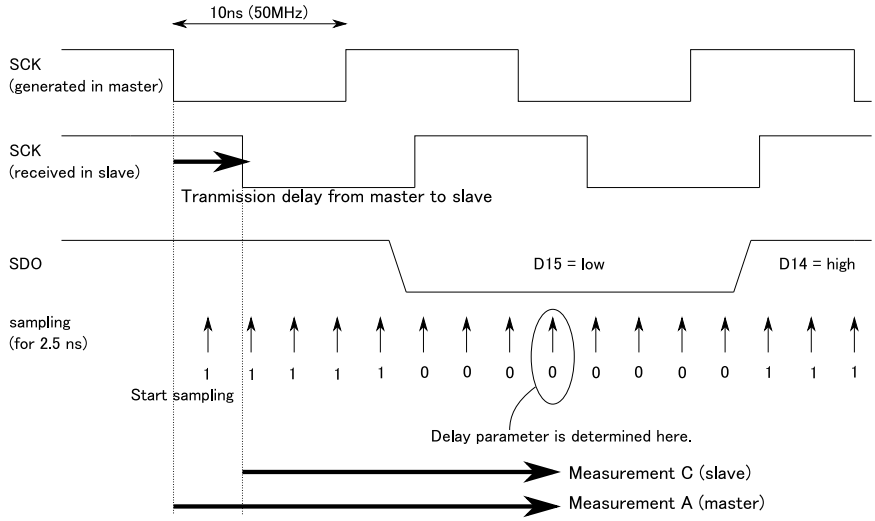

Figure B.1: Explanation for the discussion 2. Time from the falling edge of ADC clock to the MSB of the serial data is measured by the calibration process. In slave board, the ADC clock is delayed for the transmission from the master to the slave. Subtracting the result of measurement B from the measurement A, we can estimate how long it takes for the ADC clock to transmit to the slave board.

which agrees with the measured value 46.3 ns in order.

B.3 Discussion 2 (Measurement A & C)

As explained in section 4-4-1, the BDA slave board just receives the ADC clock generated in the BDA master board. Hence the transmission delay of the ADC clock from the master to the slave board affects the delay parameter as illustrated in figure B.1. From the measurement A and C in table B.1, the measured delay time from the master board to the slave board can be calculated as 9.4 ns. Let us see how this delay time is explained.

Since we mention the transmission within the BEE system, $t_{\text{transmission in cables}}$ is considered to be zero. As for $t_{\text{transmission in boards}}$, the signal line length is about 10 to 20 cm from the master to the slave, so that the propagation delay time in boards are roughly estimated as about 1 ns. Finally the delays from the devices are shown in table B.3. Hence the expected delay time from the

Device	Explanation	Location	Input/Output Delay (ns)
Spartan3AN	FPGA output	BDA master	max. 4.37
SN65LVDM176	LVDS driver	BDA master	min. 0.5, typ. 1.7, max. 2.7
SN65LVDM176	LVDS receiver	BDA slave	min. 0.5, typ. 1.7, max. 2.7
Spartan3AN	FPGA input	BDA slave	max. 2.68
Total (*)			$\simeq 10.5$

Table B.3: Input/Output delay of devices concerned with the receive of ADC clock on the DAQ slave board ([22], [28]).
(*) "Total" is the summation of the typical values for LVDS and the maximum values for FPGA.

datasheet is roughly

$$t_{\text{transmission in boards}} + t_{\text{input/output delay of devices}} \simeq 1 + 10.5 = 11.5(\text{ns}). \quad \text{(B.3)}$$

This value agrees with the measured value in order estimation.

We confirmed from the above two discussions that we are taking the data at the right timing and the calibration method works well.

Acknowledgment

I would like to express my great appreciation to my supervisor H. Aihara for his overall support. I greatly acknowledge T. Abe for his sincere advice. I am grateful to M. Iwasaki and H. Kakuno for their daily discussion.

I am indebted to T. Uchida in KEK electronics group for his useful advice on my research and for many discussion about the development of the BEE. I am also grateful to H. Nakaya who is a member of National Astronomy Observatory of Japan for his valuable discussion and his help for the performance measurement of BEE.

I am also thankful to M. Miyazawa and the other members of Gnomes Design Co., Ltd. for their wonderful fabrication of the printed circuit boards in the BEE.

My appreciation also goes to the members of the Aihara laboratory. I would like to acknowledge H. Miyatake for useful advice and S. Mineo for many discussions about the development of the BEE. I am grateful to Y. Nakahama, H. Nakayama, C. Ng and S. Sugihara for a nice life at the University of Tokyo. I am also thankful to K. Kuniko who is a staff member in Hongo for supporting my research activity.

Finally, I would like to express my gratitude to my parents.

Bibliography

[1] S. Dodelson, "Modern Cosmology", Academic Press (2003)

[2] M. Kowalski *et al.*, "Improved Cosmological Constraints from New, Old, and Combined Supernova Data Sets", ApJ **686**, 749-778 (2008)

[3] J. Dunkley *et al.*, "Five-year Wilkinson Microwave Anisotropy Probe Observations: Likelihoods and Parameters from the WMAP Data", ApJS **180**, 306-329 (2009)

[4] K. Sato and T. Futamase, "Cosmology I - the Beginning of the Universe (Modern Astronomy series 2)", Nihon-hyoronsha (2008)

[5] T. Futamase *et al.*, "Cosmology II - the Evolution of the Universe (Modern Astronomy series 3)", Nihon-hyoronsha (2007)

[6] Y. Taniguchi *et al.*, "Galaxy I - the Structure of the galaxies and the Universe (Modern Astronomy series 4)", Nihon-hyoronsha (2007)

[7] http://subarutelescope.org/index.html

[8] S. Miyazaki *et al.*, "Subaru Prime Focus Camera – Suprime-Cam", PASJ **54**, 833-853 (2002)

[9] R. Massey *et al.*, "The Shear TEsting Programme 2: Factors Affecting High Precision Weal Lensing Analysis", Mon. Not. Roy. Astron. Soc. **376**, 13-38 (2007)

[10] HSC project members and Subaru telescope members, "Hyper Suprime-Cam Design Review" (2009)

[11] TIA/EIA-644 standard

[12] EIA-232-D/E standard

[13] Analog Devices, "AD7686 Data Sheet"

[14] K. Yonemoto, "The Basics and Applications of CCD/CMOS Image Sensors", CQ Publications (2003)

[15] IEEE 1101.1 standard

[16] IEEE 1101.10 standard

[17] IEEE 1101.11 standard

[18] H. Miyatake, "Research and Development of Readout System for New Wide-Field CCD Camera of Subaru Telescope", University of Tokyo, Master thesis (2009)

[19] Atmel, "ATMega2561 Data Sheet"

[20] Texas Instruments, "INA196 Data Sheet"

[21] H. Johnson and M. Graham, "High-speed Signal Propagation *Advanced Black Magic*", Prentice Hall PTR (2003)

[22] Texas Instruments, "LVDM176 Data Sheet"

[23] Texas Instruments, "LVDS31 Data Sheet"

[24] Texas Instruments, "LVDS386 Data Sheet"

[25] Texas Instruments, "LVDS389 Data Sheet"

[26] Texas Instruments, "LVDS390 Data Sheet"

[27] Analog Devices, "AD5555 Data Sheet"

[28] Xilinx, "Spartan-3AN FPGA Family Data Sheet"

[29] Linear Technology, "LTM4616 Data Sheet"

[30] Fluke, "Fluke187 Data Sheet"

[31] Caddock, "MP2060 Data Sheet"

[32] Tektronix, "DPOJET Data Sheet"

[33] Toshiba, "TC7WZ34FK Data Sheet"