BroCam
A versatile PC-based CCD camera system

Jens Klougart

Niels Bohr Institute of Astronomy,
Physics and Geophysics.
Copenhagen University Observatory
Brorfelde, 4340 Tollose, Denmark

ABSTRACT

At the Copenhagen University, we have developed a compact CCD camera system for single and mosaic CCDs. The camera control and data acquisition is performed by a 486 type PC via a frame buffer located in one ISA-bus slot, communicating to the camera electronics on two optical fibers.
The PC can run as well special purpose DOS programmes, as in a more general mode under LINUX, a UNIX similar operating system. In the latter mode, standard software packages, such as SAOimage and Gnuplot, are utilized extensivelythereby reducing the amount of camera specific software. At the same time the observer feels at ease with the system in an IRAF-like environment. Finally, the LINUX version enables the camera to be remotely controlled.

CONTENTS

1. INTRODUCTION

This paper will introduce you to the Brorfelde (Copenhagen University, Denmark) CCD camera concept (BROCAM), presenting an overview of the hardware and especiallly the software developed for the operation of the camera.

For the last 5 years, CCD's have been investigated in Brorfelde. The necessary electronics and software has been devoloped, and a continuous ongoing activity for improving the overall performance of the total systems has led to consideral improvements, resulting in an easy to use and time efficient system for the observers at the telescopes, with virtually no tradeoff in basic performance like noise, charge transfer efficiency, linearity etc.

As CCD technology is fairly new, the major efforts have been focused on the performance of the CCD's, trying to make the basic systems behave well and to understand the physics of the detector.

User interfaces and application programs for the computers servicing the CCD cameras is often seen to be locally developed software, growing from each new need, doing the job in the laboratory, but seldom aimed at doing effective observations and very often bound to the specific type of computer used in the laboratory. And very often consideral effort has been put into doing advanced image analysis within this software.

The new approach taken in Brorfelde arises from the wish not to develop another image analysis program, but to provide the basic data acquisition for the CCD cameras, utilizing existing software packages. Then closely connecting the system to existing and well known image analysis software, making the observer feel comfortable within a well known environment. These packages could be IRAF, MIDAS or any other image analysis package running on a workstation with standard UNIX operating system and network interfaces. The system described here solves the basic problems of interfacing the specific hardware into the UNIX environment, at low cost, and with no performance penalty in form of added readout time or less functionality.

Brorfelde Image Acquisition System (BIAS), is developed for an IBM compatible PC running the UNIX-like clone LINUX. This freeware operating system provides most of the UNIX functionality, including network interface and X11. And it does allow us to use all kinds of standard interfaces using the PC's standard busses (ISA, PCI etc).

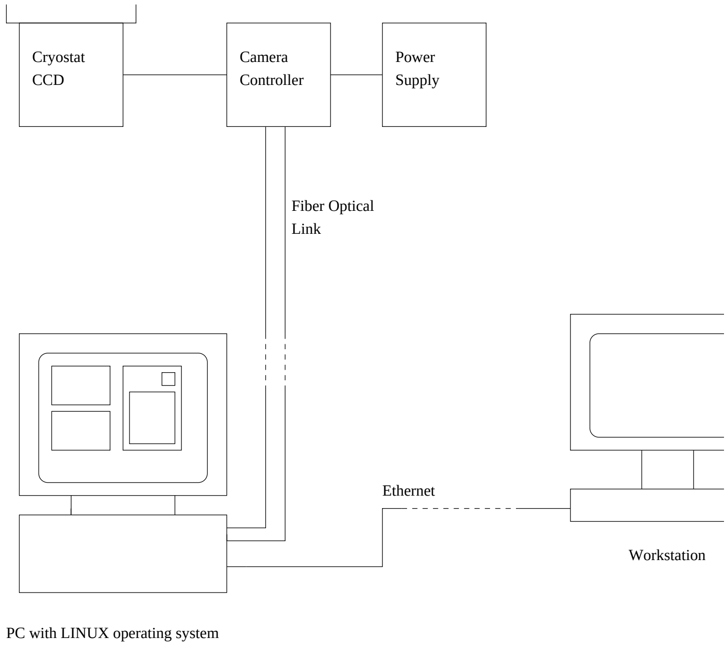
The Brorfelde cameras have been operated with the new BIAS acquisition software since october 1994 at the 2.5 m Nordical Optical Telescope at La Palma (Canary Islands) and at the Brorfelde Schmidt telescope (50/77 cm) in Denmark with considerable succes. We strongly believe that the flexibility and portability of the system ensures us many years of good operation with little effort in software development and maintanance.

2. AN OVERVIEW OF THE BRORFELDE CAMERA, BROCAM

Given the large number of miscellaneous CCD chips available with regard to sizes, readout schemes etc. the camera electronics has been developed to accomodate all known schemes. But for each new CCD chip adapted to the system, there is considerable work in optimizing parameters. I will here shortly describe the camera hardware in BROCAM.

2.1 Camera head

The camera head containing the CCD chip is either a cryostat cooled with liquid nitrogen and temperature control keeping the CCD at a selected temperature, typically -100 degree C, or for some purposes a Peltier cooled camera head operaing at -40 -> -50 degree C. For each different chip, printed circuit boards are designed for mounting the chip inside the unit, the board usually containing phase filtering RC circuits and the first amplifier stage for each readout amplifier chain. Much effort has been put into the overall thermal design for making the usage of nitrogen low, or to get the very best performance from the Peltier elements.



BROCAM SYSTEM

2.2 Drive electronics

The BROCAM drive electronics is fitted into 2 small chassis, one containing the controller, the other the raw power supply.
The raw power supplies are standard linear regulators making voltages appromimately 1 volt higher than required by the electronics in the controller. The controller chassis has low drop regulators doing the final regulation. In this way the final filtering of voltages is done very close to the drive electronics with low heat dissipation.

Besides the voltage regulators the controller box contains a number of euro-sized printed circuit boards, connected through a backplane. They are:

CPU board
Phase Driver board
Amplifier and A/D board
Miscellaneous interface

The CPU board is a Motorola 68020 uProcessor board with 128 kbyte memory, firmware EPROM, and a full duplex optical fiber interface for communicating with the main computer. The board is developed in Brorfelde on a 6-layer circuit board using power and ground planes to limit uProcessor noise. All basic timing is done by the uProcessor, this includes phase- and readout timing. As the processor runs at 20 MHz, the timing increment is 50 ns. No local memory for images is incorporated on the board, which means that the images must be sent through the fiber and received at the other end in real time. As for now the software for a CCD is specially adapted for that particular CCD and put into EPROM, but a scheme for downloading the software is being developed at the moment, thus making the controller software a configuration option. Commands to the camera are sent through the fiber, and status information and image data are sent back at a rate of 10 Mbit/sec.

Phase driving is accomplished with standard high output drive operating amplifiers. The software in the controller determines the timing.

The amplifier and A/D board has 2 signal chains and is fully processor controllable. The signal chain uses uses the 'double correlated sampler' principle of operation, and the A/D converters are 16 bit.

On the boards connecting to the CCD, hardware clamps has been incorporated, ensuring that no voltages exceeds the maximum ratings of the CCD. It is also worth while noting that the camera head electronics is electrically isolated from the main computer system, minimizing the risk of ground loop spikes damaging the CCD.

3. Interfacing the camera hardware

As mentioned above, the camera head contains no local storage for images. During readout the image data are transmitted in real time, which means one 16 bit pixel every approximately 5 -> 30 usec depending on the chip used and the number of output channels. To collect these data in real time, an interface board for an IBM compatible PC has been developed, the Frame Store. The board fits into the ISA bus of the PC and can be supplied with enough memory to accomodate even mosaics. As image data is transmitted from the controller, it is collected on the board's memory and can immediately be fetched by the PC's running program. All control to the board is done through the PC's I/O channels. Special hardware makes it possible to access image data at nearly the maximum speed of the ISA bus.

4. Using a DOS data acquisition program

For some years a program (QIMS) developed for DOS with TurboPascal has been used (and is still used), for the controlling of the Brorfelde camera. This program runs effectively the data acquisition in a stand alone environment, with the need for large local disk capacity and some local means of mass storage, tape or similar. The program has a large number of engeneering facilities, which makes it perfect for testing and debugging. Also incorporated is some image analysis software.

There is a number of problems running the data acquisition under DOS. The well known memory size limitation (< 1 MByte) gives constraints to the program development, both concerning program size and the allocation of memory data areas for images. Secondly the non-multitasking nature of DOS makes the system stand alone, meaning that all utilities should be incorporated in one program.

One example of the difficulties, is the reorganisation of image data when more than 1 output amplifier is used in the camera. When no camera head data storage is present, data are sent with a pixel from each amplifier at a time, making the data interleaved on the Frame Store. The total reorganization is time consuming and must use a second image buffer on the Frame Store.

Usually the PC running DOS is connected to UNIX workstations through ethernet with a commercial network software package, and with in the QIMS software, it is possible to transmit data to a workstation, even online. But some configuration difficulties arise implementing this in different environments at the telescope sites.

5. Setting up the demands

As the DOS environment became limiting, what were our demands for a new system for controlling the CCD data acquisition.

1: The Frame Store, temporarely holding image data, must be interfaced to the system, with fast access to it from the host computer.
2: The system should run a multitasking operating system including network utilities, allowing several programs to run simultaneously and to have remote access to data.
3: The operating system must allow you to acces the hardware by simple means of writing drivers. This is seldom possible in a simple way on commercial workstations.
4: There should be a graphical interface for displaying images.
5: The system should be able to run standard image processing software.
6: Development tools must be present (compilers, linkers).
7: The final software should be portable.

The 1. demand does gives major limitations, as most standard workstations does not have provisions for customized input/output, or the possibilities are limited by proprietary interfaces and/or the presence (or lack) of drivers. Some workstations can be purchased with the VME bus, some called industrial workstations, but pricing is prohibitive. Another possibility was to build up a system of VME boards, but this solution was also abandoned due to the total price.

Using a PC seemed to be a good solution, but we needed another operating system than DOS. Windows was considered, but as we became aware of the LINUX free software, a UNIX-like clone for the PC's, we made a number of tests to see whether this would fullfill our needs. Our conclusion was that the only restriction was, that some image processing tools, especially our own preference IRAF, was not yet ported to LINUX.

After extensive testing of the operating system, the hardware interface and the possibility to recompile miscellaneous software packages under LINUX, it was decided to write the new Brorfelde CCD acquisition system, using LINUX on a powerfull PC. The software package was named BIAS, for Brorfelde Image Acquisition System.

6. LINUX

LINUX is a freeware UNIX-like clone developed by a number of people around the world, some writing the kernel, some contributing with utilities and others developing the XFree X11 interface. This development process is continuing as new hardware sees the market, and new software packages are released.

As we made our first installation, it became clear that this was a very ’clean’ and easy to use system, a fact that some of the major workstation suppliers could learn from. A full blown installation takes some 160 MByte disk, and when using X11 on the system, I will recommend at least 8 MByte memory, more if doing some large image analysis. The network software works with most standard ethernet boards, and X11 works with a large number of video boards. A 17 inch SuperVGA monitor is also recommended.

Included in LINUX are most standard utilities, GNU C-compiler, X11, TeX and full source code for the kernel. New kernels with user selected drivers for a vast amount of hardware interface, can easily be built with only a fair amount of UNIX knowledge.

On the systems allready installed, an time server enquiry program is run to ensure precise UT (Universal TIME) on the machine.

7. Interfacing the camera to LINUX / PC.

Interfacing the existing Frame Store board to the LINUX system was a crucial point, but a few simple tests showed that this is perfectly possible, even without writing appropriate drivers into the kernel of LINUX. Instead a low level daemon was written, hiding the ’dirty’ low level input/output.

8. The BIAS programs

The BIAS software system consists of a number of programs running under LINUX/X11, interconnecting though named pipes. Pipes are a special operating system construction, working like FIFO’s (first in, first out memory), and behaving to a program like a file which can be read or written. For 2 way communication between processes, 2 pipes are used. The system also uses the multitasking features of the operating system, starting background processes as needed.

For some interfacing between programs, intermediate files are created, especially for the plotting utility.

The process intercommunication allows for the use of several existing software packages, which reduces the basic code size of the BIAS programs and thus saving development time. Actually no X11-windows code has been written for these programs.

The individual programs and processes of BIAS are:

1: A script, which starts all processes, and kills them when exiting the system.
2: Low level daemon (background program).
3: Status display daemon.
4: SAOimage, image display program.
5: Main BIAS command program.
6: GNU-plot program.

Optional terminals are opened at request for on-line help and viewing image headers.

In the LINUX X11 environment, the window manager (fvwm), gives the possibility of having a number of virtual workspaces, thus enabling you to switch the whole X11 screen by the click of the mouse. In such another workspace, IRAF, or another image analysis program, can be started. As IRAF is not yet ported to LINUX, the program is run on another workstation, using the LINUX-PC as an X-terminal. Image data can be acquired via NFS, by mounting the PC's hard disk on the file system of the workstation. Or the BIAS system can store data directly on the workstation's harddisk.

8.1 Startup script

This script is a normal shell script setting up environment variables, and starting the processes. After quitting BIAS, all processes are killed, and intermediate files are erased. The script places the windows appropriately on the screen and selects appropriate fonts.

8.2 Low level daemon

This small program runs in the background, translating simple commands received from a pipe to low level input/output commands accessing the hardware. Image data and status information is sent back through another pipe upon request. Even with large amounts of image data, this only gives a minor delay of accessing data and gives a clean interface to the hardware.

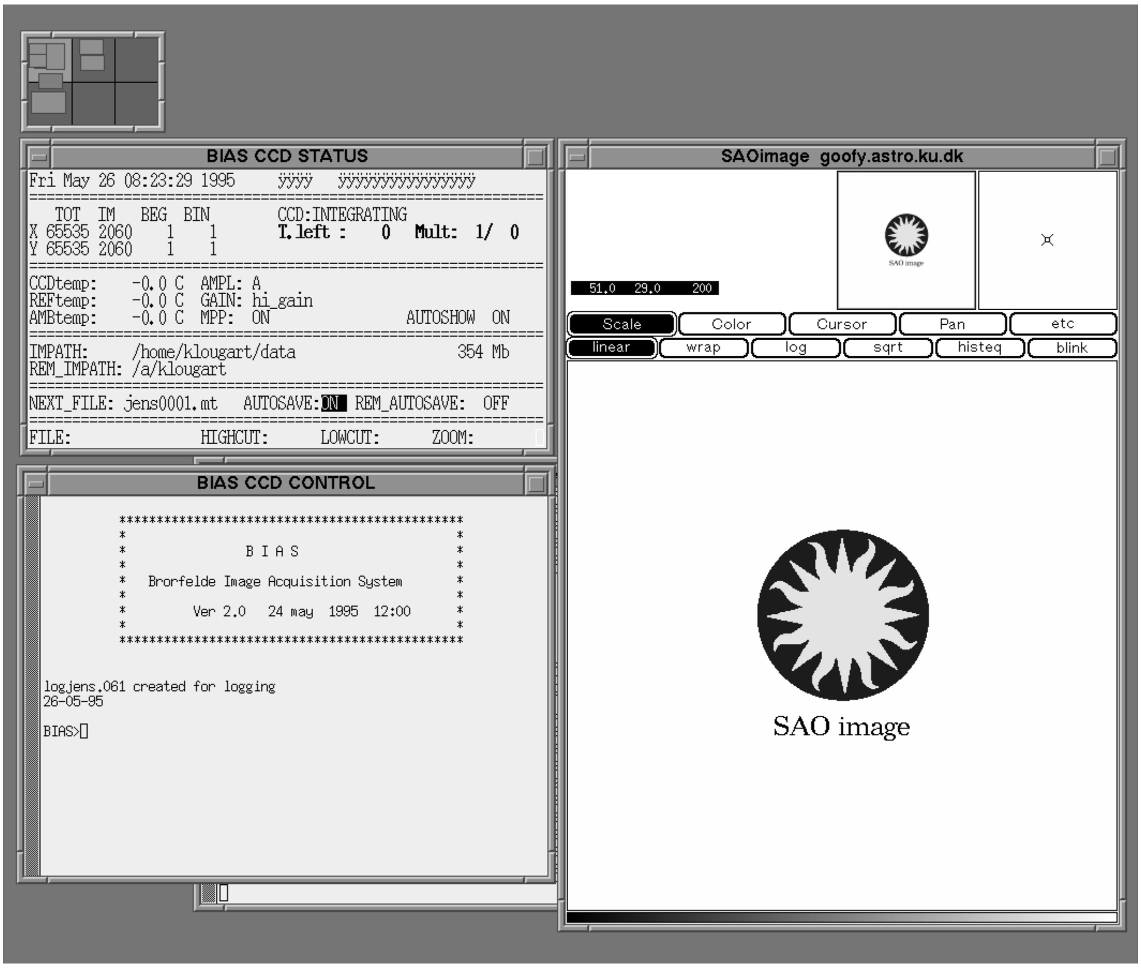
The daemon must be run with special acces rights, but is started by the user automatically when stating the system.

8.3 Status display daemon

For displaying status information of the BIAS system and the CCD, an xterm window is started with this small program running, taking care of updating display data. The xterm is the standard X11 xterminal program allowing you to place text in any position of the screen.

8.4 SAOimage

SAOimage is an image display program developed at the Smithsonian Astrophysical Observatory, commonly used with IRAF. The program interfaces other modules mainly through named pipes. It gives a number of excellent facilities to the user such as using other color pallettes, panning and zooming etc. I will not go into further details about this program. With proper setup, images can be printed on a Postscript printer.



8.5 Main BIAS program

The main BIAS program runs in an xterminal window, giving you the possibility to execute a number of simple commands for doing your observations effectively. With virtually no effort in reading large manuals, it is possible to start doing observations.

At start-up, the status window shows information concerning the CCD setup (geometry, binning, temperature etc.), and setup for the BIAS program, such as paths, filenames, disk space available and information on the current image shown in SAOimage.

At the ’BIAS>’ prompt, commands are given to the system, with optional data entered either directly on the command line, or by just entering the command and ’enter’. You will be prompted for data, with the latest value as the default.

Some of the most common commands are:

help	This start an xterminal window, which runs the data viewing utility ’less’, with available commands and a short description.
quit	Exit BIAS.
exp [T [F]]	Start a normal exposure with optional integration time T (in seconds) and filter number F.
mexp [T [M [F]]]	Start M integrations of each T seconds using filter F.
dark [T [F]]	This is for dark exposures.
mdark [T [M [F]]]	This is for multiple dark exposures.
focus [T [F]]	This command starts a focus batch. Each exposure is T seconds with filter F. At the completion of each exposure you are prompted to start a new (after moving the telescope and focus). The readout will then start. Unfortunately no connection to the Telescope Control System is implemented at this time.
head [name]	View a file. Starts an xterm window with ’less’ on the filename. Works like ’help’. Lets you view the FITS file header.
autosave	Store the CCD images in the specified ’impath’. When files are stored, there is a check for the remaining diskspace. A warning is given at 10 Mbyte remaining, with 2 Mbyte left, the file is not saved.
autosave-	Do not store images.
impath [path]	Set the directory (or path) where images and logfiles are stored.

filename [name]	Specify the next filename to use. Default is an automatic numbering mechanism.
imex	Start a pseudo IRAF imexamine in the SaoImage window. The plot utility ’gnuplot’ is used for graphical output. Only a subset of commands are implemented.
disp [name]	Display a FITS image in SaoImage from the harddisk.

Furthermore there are commands for entering information into the FITS header, changing the image display cuts and setting all kinds of parameters for the CCD such a partial readout and binning. Also readout amplifier selection, gain and miscellaneous CCD modes can be specified. A logfile of all the activity is created with time on each command. Images are stored in standard FITS image output format.

9. WHAT’S NEXT

The system described here is at the moment still in a very early phase, allthough it is run at a few sites. We try collecting wishes, and the detection of possible errors in order to get the system running smoothly and safe, with emphasis on doing the acquisition with this system, and keeping advanced image processing in the standard packages.

Still I would like to emphasize some very important items, that still need to be implemented in BIAS. First is the need to interface to the Telescope Control System, to get telescope coordinates, siderial time and all other relevant information into the FITS header. It is also a strong wish to be able to implement an automatic focus sequence.

With a close interface to the Telescope Control, automated sequences of observations will be possible, as well as automatic on-line data reduction. We hope to be able to implement this in the near future on several telescopes. Also a tip/tilt camera for image motion restoration is being built in Brorfelde, which will interface to the science-CCD through the LINUX-PC.

My hope is that the people developing LINUX for the PC’s will continue their good work, allowing us to utilize low cost hardware with all the benefits of the standard interfaces and software. It gives us the privilege of being independent of specific commercial companies, thus being able to spend most of our effort (and money) on the science and the astronomy.