AST/RO Technical Memorandum #15
August 9, 1993

OVERVIEW OF WORK DONE ON DOME CONTROL

Edgar Castro
Boston University; Dept. of Astronomy, 725 Commonwealth Ave; Boston, MA 02215

Abstract

This memo describes the work done on the dome drive and dome encoder operations, including more detailed technical, mechanical, and electrical information on the PCL-711 digital to analog converter, the implementation of the dome encoder, and the dome slit limit switch.

1. AST/RO DOME DRIVE AND ENCODER OPERATIONS

The construction and implementation of a dome drive and encoder system for the AST/RO test site in Boston is a necessary component for automating the AST/RO telescope operations. This system must move and track the dome accordingly with an operating telescope with accurate results. The stepper motor, motor driver, and 12-bit encoder allows this to be accomplished.

2. UNDERSTANDING HOWARD'S WORK

Using Jim Howard's initial programs left on the personal computer in the dome before primary operations moved from the lab to the dome, concepts of i/o from a computer to an external source were soldified using Turbo C. Also buses and addresses were more fully understood using the digital to analog card, which sends signals to the stepper motor control box, after transferring this D-to-A card to adelie from the personal computer.

3. PCL-711 DIGITAL TO ANALOG CONVERTER

The PCL-711 card address is 0x284 which can be found on a dip switch located on the face of the card. The 20-pin ports used are (CN1) for digital to analog conversion, and (CN3) for controlling the enable, disable, and directional highs and lows. This is digital output.

(CN1) The connections made to this port are (pin 17) for digital to analog conversion and (pin 19) for an analog ground. These connections follow a 20-pin ribbon cable to a mating board concieved by James Howard. The board was re-wired following Howard's initial scheme and placed within a small metal casing for protection. (Pin 17) is directly connected to the VCOin which is pin 9 on the National CD4046B Phase Lock Loop chip. And (pin 19) of the PCL-711 card is connected to a relative ground of the mating board.

(CN3) The connections made to this port are (pin 1) enable/disable control, (pin 2) for directional control, (pin 18) for digital ground, and pin 20) for +12volt power supply. (Pin 1) is connected directly to the stepper motor driver which is wired with an RS-232 port and cable. This connection is the red wire in the stepper motor cable. (Pin 2) is also directly connected to the stepper motor driver which is the yellow wire in the stepper motor cable. (Pin 18) is connected to a relative ground on the mating board. (Pin 20) is connected to the VDD which is pin 16 on the National CD4046B Phase Lock Loop chip.

Also available is a schematic of the wiring set-up, the National chip information sheet, the RS-232 wiring for the stepper driver, and Howard's notes on the mating board scheme within the grey dome folder. For additional reference of D/A card pinout see page 9 of the PCL-711 manual.

4. VARIABLE SPEED DOME DRIVE PROGRAM

Using Howard's initial programs as a base, The program stepramp.c was contructed piecing together enable.c, disable.c, rampup.c and rampdn.c so the user can specify a certain value for the step speed instead of using the top speed (rampup.c) all the time. This program admittedly crude was futher expounded upon with Professor Jackson's much more sophisicated and shorter step5.c program which is currently still available for use.

5. MOVING THE DOME MOTOR POSITION

It was decided last year that the previous position of the dome motor was not ideally located within the dome. It would have interfered with the telescope movement and tracking. With the help of Professor Jackson, the dome motor and platform were moved to its current position today. Shortly after its move, the dome motor fell off its base twice causing damage to its outer shell, which I take full responsibility for and since then have secured the base plate of the motor to the wooden platform which also holds the encoder mechanics. The motor damage was only cosmetic and is still fully operational.

6. ADJUSTMENTS TO THE DOME MOTOR MECHANICAL SETUP

After the move of the motor platform, it was observed that the dome gear occasionally jumped out of its track during use. It was determined that the gear shaft did not have a proper pressure applied down upon it to maintain its course on the track and that the rubber pressure piece slid off its shaft stabilizer collar when the dome was running at higher peds. So a small wooden holder was constructed which fits upon the shaft stabilizer collar and holds the rubber stopper without letting it slide off. Also some other collars were put on the shaft to minimize sliding of the two shaft stabilizing collars. These two actions have solved the problem of the keeping the gear within the track.

7. ENCODER PROGRAMMING AND LOGIC

Finding the inversion scheme was the primary problem of decoding the encoder properly. Maohai Huang developed a smooth program to read the bits from the encoder and different inversion schemes were tried and finally the correct one was found. This program domeencoder.c is currently being used in the subroutine dome_subs.c for decoding the encoder bits to decimal numbers for targeting the dome slit azimuth. Documentation on the encoder's algorithm and ribbon cable pinout can be found in the grey dome folder.

After getting raw decimal numbers from the encoder using Maohai's program, a way was devised to change those numbers to raw degree form and then to azimuth degrees. This scheme allows the dome to move 440 degrees clockwise or counter clockwise from a certain fixed north point. This program e_az.c can also be found within the dome_subs.c program.

Maohai Huang's domeencoder.c program contains the logic to decode the encoder to decimal form. The encoder read out in databits is from zero to 4095, 4096 databits total before the encoder "turns over" to zero again.

The basic scheme of decoding the encoder is to prevent "turning over" when moving the dome to a position. So the databit value of 2048 will be the centering databit for the encoder. This value will correspond with a dome slit position due north. So there will be roughly 2000 databits travel on either side of this position.

Now it is found that the dome can move one full rotation plus some, in either direction from the center zero databit- 2048. The points where the dome becomes zero degrees again are approximately databits- 374 and 3720. These numbers are not exact due to the encoder reads only integers and after a few rotations the encoder zeroes are altered slightly. The degree per bit value is taken from data that ran the dome around two times and divided the delta in databits by 720 degrees, three different times. And the extreme values before the encoder "turns over" are in databits- zero(0) and 4095. Since one rotation is not exactly in integer form the result is a real number. The degrees per bit equals .214989549

Using this information the degree per bit value is multiplied by the integer number databits from

M.Huang's domeencoder.c program and the result is in a raw degree form, this is NOT an azimuth angle.

Here is the code which preforms the task of changing the raw degrees to an azimuth angle:

```
  double deg per bit=.214989549;

/* define the azimuthal zeroes */
  float zero=(float)(2048* deg per bit);
  float zero2=(float)(374* deg per bit);
  float zero3=(float)(3720* deg per bit);

/* convert databits from encoder to raw degrees*/
  degree=(float) (databits * deg per bit);

/* set azimuth zeros (or 360) */
/* These angles are fixed and will always be zero azimuth*/
  if (degree==zero | degree==zero2 | degree==zero3) az=0;

/* stop dome movement if reached encoder turnover limit if "degree"=0 */
/* this is equal to databits=0 to prevent turnover */
  else if (degree==0) adout(0);

/* stop dome movement if reached encoder turnover limit if "degree"=882 */
/* this is equal to databits=4095 to prevent turnover */
  else if (degree==882) adout(0);

/* convert raw degrees to azimuth degrees */
/* if the current encoder readout is less than the center zero databit
 2048*/
  else if (degree < zero)
    (
    az= fabs(degree-zero);
    if (degree < zero2) az=(float)(az-360);
    )

/* if the current encoder readout is more than the center zero databit
 2048*/
  else if (degree > zero)
    (
    az= fabs(degree-zero3);
    if (degree > zero3) az=(float)(360-az);
    )
```

For example if the encoder databit readout is 348 then the absolute value of 348∗ deg per bit −2048∗ deg per bit = 1748∗ deg per bit = 365.4822319 degrees but since this value is greater than 360 and less than the 374 zero databit this value must be converted again to confine the azimuth value from zero to 359.999999 degrees. So 360 is subtracted from 365.4822319 to give a final azimuth of 5.4822319 degrees.

8. MOVING DOME TO TELESCOPE AZIMUTH

Contained within the dome_subs.c program are the subroutines, var_ramp, Maohai's convert_domeencoder (domeencoder.c), enc_to_az, decide_cw_ccw, stepvalue, and shmemory which are used to move the dome to the telescope azimuth. Var_ramp is a modification on Jackson's step5.c program and shmemory was a collaborated effort with Maohai Huang.

The main program, dome.c, utilizes these subroutines within dome_subs.c to point the dome slit to the current telescope azimuth. It is known that there is an elevation offset to the azimuth and that problem is currently being worked on by Professor Jackson.

9. ENCODER MECHANICS

During the encoder programming it was realized that a new gear was needed to attach to the encoder to read positions of the dome. One was ordered from Ash-Dome and recieved after greatly delayed shipment. A scheme was devised and implemented to connect the encoder to a shaft that was connected directly to the new gear.

After it was realized that the encoder "turned over" too quickly if connected directly to the dome, a gear reduction scheme was devised by Jim Ingalls and myself and was constructed by Jim onto the wooden platform.

10. DOME SLIT SWITCH AND LIMIT SWITCH LEVER ARM

There was mechanical difficulty with the operation of opening and closing the dome slit with its motor. The drum switch was taken apart and cleaned out, a broken plastic piece was epoxyied and fixed and the six spring contacts were adjusted for better electrical contact. Finding that the problem was not solved, Ash-Dome was contacted for technical assitance and they advised checking the limit switch which enables and disables the ability of the motor to go in reverse. Apparently the arm to the limit switch was binding and so the mechanism was taken apart and oiled and re-assembled it so the bolt and nut joint was properly tightened and this solved the dome slit problem.

10. DOME LIMIT SWITCH CORRECTION

If the dome switch is not able to go in reverse then one of two errors is happening. 1) The dome slit limit switch is not working correctly. It is probably jammed up and needs to be worked free from its bind. 2) The switch is not making proper contact and must be cleaned and adjusted.

1) To fix the limit switch, a suitable ladder must be attained,(i.e. from the shop) and the metal piece which is connected to the limit switch arm must be removed. The connection is only a bolt and nut (9/16 hex head) and the limit arm should pop up when connected to nothing. If this does not occur then the switch is faulty and the Ash-Dome company must be contacted for a replacement or for futher information.

If the arm properly pops up then put back together the bolt and nut to the lever arm and make sure the bolt is tight enough to hold but loose enough to rotate in its hole, other wise the lever arm will not push back because it is binding. If this still does not correct the problem then it is possible to take apart the rest of the lever arm and make sure the bolts between them are not binding the moving action of the limit switch.

2) It is also possible for the contacts within the switch case to become dirty and make bad contact. If this happens wipe the six spring arms clean and bend them in to make sure they make good contact in all 3 switch positions.

Diagrams of the limit switch lever arm and the switch case are located in the grey dome folder.