# CTA CHEC Door Motors, Program Overview


## Robert Peck

**Primary email**: rp329@student.le.ac.uk

**Secondary email**: fusionforever235@gmail.com


This document contains notes on the behaviour of the programs used on the master and slave controllers for the doors on the CHEC cameras of the Cherenkov Telescope Array. Information on inputs outputs and approximate timings is given. Details of the workings of the programs are not described, for those refer to the comments within the NanoJEasy (Java based) code.

**Please note:** The programs are designed to be as fault tolerant as possible and should in theory only truly fail in the event of hardware damage to the electronics of the system. Prohibiting failure of a physical component the only way in which this program can cause damage in the event of an error is if the variable CloseDist is set to a high enough value to cause the doors to continue moving past the closed position. The value must be set by editing the “int CloseDist = [value];” line near the top of the program source code. It must be done for both the slave and master programs separately. Each must then be compiled in NanoJEasy and written to the relevant controller using the NanoPro software.

**Note also:** The programs are carefully designed so that they will not attempt to open any door that is already open or close any door that is already closed. Although in some cases the safety board may request an open door to open or a closed door to close this will not occur, infact it is expected that the safety board will continue to issue a command to close throughout the time the door should be closed for, or atleast maintain such a command for more than sufficient time to allow the doors to close. If both doors are open/closed then commands to open/close will be utterly ignored, if one is in the desired position and the other is not then only the one not in the desired position will start moving.

**Note also:** In this manual status and command codes such as 01, 11, 00 and 10 are referred to. There are however differences in the way that some of the inputs and outputs define their logic levels. Within the NanoJEasy code of the programs it is easy to see how the inputs work in a binary fashion to give a numerical value, with outputs however the decimal numbers used are those that if written in binary produce a zero on every line which is desired to be read as a 1 and vice versa. The camera safety board should send commands so that the voltage rises on a line when that line should send the master and slave what they consider to be a 1, the voltage should drop when the message which needs to be sent requires the master and slave to read a zero on the command line in question.

Command 01, Set the voltage on line A low and the voltage on line B high

Command 10, Set the voltage on line A high and the voltage on line B low

Command 11, Set the voltage on line A high and the voltage on line B high

Command 00, Set the voltage on line A low and the voltage on line B low

Status 01, Line A will read as a low voltage, line B will read as a high voltage

Status 10, Line A will read as a high voltage, line B will read as a low voltage

Status 00, Line A will read as a low voltage, line B will read as a low voltage

Status 11, Line A will read as a high voltage, line B will read as a high voltage

On the allow line, when the master sends what the slave reads as a 0 the voltage is low, the voltage on this line goes high when the master sends what the slave receives as a 1. Throughout this document and the comments in the code this higher voltage is referred to as being allow=1.

On the slave to master status lines the voltage is set high to send what the master reads as a 1 and low to send what the master reads as a 0. Just as a low voltage means 0 on the master to camera safety board status lines.

**Special notes about effects of closing if one door is already closed:**

1.If the slave is indeterminate or open and the master is already closed then a close command can be problematic. It may cause the slave's program to become trapped in a loop until a different command is given. This problem will not happen if both doors are initially indeterminate even if one has far less angle to turn through before closing than the other.

2.If the slave is already closed but the master is open or indeterminate then the master will close itself alone, this will cause rubbing on the seal during the final stages of closing. This problem will not happen if both doors are initially indeterminate even if one has far less angle to turn through before closing than the other.

# Effects of Inputs

The master and slave motor system recognises four possible inputs from the camera safety board:

**01**, this command will cause the doors to open if they are in indeterminate or closed positions, it will have no effect if both doors are open. If a single door is open it will not affect that one but will make the other move.

**10**, this command will cause the doors to close if they are in open or indeterminate positions, it will have no effect if both doors are already against their closed sensors. It will cause a single door

to move if one door is against the sensor but the other is not. After the doors are closed to the extent of being able to activate both closed sensors they will simultaneously close the last few steps together. They will be simultaneous as long as errors and physical damage do not occur.

**00** and **11**, these commands will have no effect on a door which is not in motion, however if either is sent when doors are in motion the doors will immediately stop.

# Meanings of Outputs

There are 5 possible outputs, four of them are steady states on the two master to camera safety board status lines, one of them is a flashing sequence on this pair of lines. They are listed below:

**00**, this status indicates an error. Some errors may clear up after a few seconds and the status will return to 01, 10 or 11. it is best to ignore errors lasting less then three seconds, errors lasting longer than this are more likely to be true errors rather than random fluctuations. Other errors may persist for a longer period. Errors sent from slave to master can in some parts of the program have specialised meanings, in such cases the master will know what these mean and will make its reports to the camera safety board accordingly. Errors from master to camera safety board always indicate an error, this can be due to a loss of signal or if both the closed and open sensors of either door report as being simultaneously activated.

**11**, this status indicates that one or both doors are in an indeterminate position. It is also shown when one or both doors are in motion, and during the 17 second period from the detection and acceptance of a command to move and the motion beginning. It is possible when in an indeterminate position that one door could be closed and the other open, it is also possible that the doors may be at different angles, therefore when starting from an indeterminate position it is generally best to command the system to open, then wait until 01 is reported which will indicate both doors are open, before trying to close them again.

**01**, this status indicates that both doors are in the open position. If only one door is open then 01 will not be reported. There may also be periods of several seconds at the start and end of motions when the doors are open but a 11 is reported rather than a 01.

**10**, this status indicates that the doors are both closed however it does NOT distinguish between the doors being in the fully sealed closed position and the doors being closed enough to trigger the sensors but not properly sealed. If an error were to occur during the final synchronised steps of closing then after the flashing 00-10-00-… code had been sent 10 may be reported, this is because the closed sensors have no way to distinguish between a properly closed door and a door closed enough to trigger the sensors. A door in the closed position, whether truly closed or simply close enough to the sensors to appear closed, will refuse to obey any commands from the safety board asking it to close. To seal a door after a faulty closure it is therefore best to, once the flashing 10

and 00 period is over,  use 01 to command the door to open and give it around 40 seconds to make sure it has begun to move and got our of range of the closed sensor, then change the command to 10 and let the door close again, hopefully doing so successfully. Sometimes the master will continue to report 11 to the camera safety board for several seconds after completing the closure before it reports 10.

**Alternating pulses 0.5 seconds long of 00 and 10**, this is a special signal sent for around 20 seconds after a faulty closure of the doors. By noticing this special signal before the master goes back to reporting 10 (closed) or 00 (error) the camera safety board should be able to identify that a closing operation has encountered some kind of error in the final stages that has prevented a seal being achieved. After seeing this sequence of pulses the camera safety board should consider the door to be not properly closed until the door has been partially opened and then closed again. Because the closed sensors cannot distinguish a properly closed and sealed door from one that is simply closed enough to trigger the sensors it is likely that after a failed closing action the system may report 10 after the flashing period is over when the program returns to the start of it's overall loop.

# Timing of Signals and Movements

Signals mostly last more than 1 second, the inputs of both controllers only register signals that last for more than 1 second[1,5], except for at a few points in the final stages of closing. Inputs from the camera safety board should therefore be held constant until the desired position is achieved, or switched to an alternative command if stopping or reversing the direction is desired. Status outputs from the master motor to the camera safety board hold constant with a 00, 10, 01 or 11 value for as long as the system is in the error, closed, open or indeterminate states. With status outputs there is however one exception to this rule, a signal containing pulses of 0.5 seconds 00 followed by 0.5 seconds 10 repeating for around 20 seconds indicates that the doors have closed but failed to properly seal. Signals on the wires between the master and slave motors generally conform to this rule of holding constant for as long as they are the case however with some of them, such as the allow line, it is the moments that pulses end which are important.

Because the master can only send messages to the slave along a single wire the signal on this line can have different meanings at different points in the program, the programs on both controllers have been designed such that it should not be possible for the programs to run in such a way that the meaning of these signals becomes confused.

Also because of the single wire from master to slave, the program contains some deliberate delays especially during the times at which this allow line has a value of 1. Some of these delays are several seconds long, others are as high as 15 seconds. They are to ensure the slave has time to run around the program loop it is currently within and therefore guarantee it the chance of

detecting the allow=1 input. Due to these delays, and other delays put in place to give the motors time to synchronise, the system may take between around 20 seconds from a new command being sent[2] to the motors beginning motion. The delays could be even longer if random errors prevent stable readings being achieved as the program will wait and do nothing until stable readings are received. The delay from the stopping of a door due to a command change to the start of the newly specified motion will be around 25 seconds.

The speed of the doors in motion at motorSpeed=500 is such as to give a 180 degree rotation in approximately 30 seconds, the total time for opening may therefore be as high as 65 seconds, if an opening command has not completed and reported 01 within 80 seconds it is likely that something is wrong. The total time for closing is longer due to the synchronised steps at the end, closing operations should be given atleast 100 seconds before assuming an error has occurred. If a door encounters brief random errors during motion it will be paused for a second, the other door will generally not pause with it but will in some cases stop if one door[3] encounters an error. Each door can therefore be delayed during opening or the main part of the closing sequence by as many seconds as the number of brief errors it experiences. The final part of closing should always be synchronised however if an error were to occur on one of the doors it might fail to perform these final steps at all. Should actual operation require faster opening and closing times it is advisable to increase the speed of the motors rather than alter the delays and risk the programs on the two controllers being unable to synchronise so reliably.

Below are outlined the sequences of signals and motions for opening and closing:

**Opening**

1.The positions of the doors are read from the sensors, the camera safety board is informed of the state of the motor system.

2.The master and slave motors read the 01 command from the safety board and prepare to act on it, if either is already open then at this point the programs on the motors will run such that if either door is not open it can open, but a door that is already open will not move.

3.The master then sends an allow=1 pulse to the slave, if the slave is not already open then it will be listening for this pulse. The slave will detect this pulse within the 15 seconds for which the pulse is active wherever the slave was in its program (excluding unusual places visited during the final steps of closing) when the pulse began. At this same moment as the allow=1 pulse begins the signal from master to safety board will change to 11. If both doors are already open the allow=1 pulse will not be sent, the status will not change to 11 and the individual motors will continue waiting for a 10 close command.

4.The master terminates the allow=1 pulse setting the allow line back to 0. This is a signal to the master and slave to start a 2 second countdown then begin simultaneously opening.

5.The doors now move, they should be synchronised however if either encounters an error, or they began from indeterminate positions that were not the same, this will not be the case. Brief errors

lasting less than 1 second will cause the motor that experiences them to pause for 1 second, errors or changes of command that persist for longer will result in the stopping of the motor experiencing them. If the slave is stopped due to an error then the master will generally stop too. If the master is stopped due to an error then the slave may continue moving unless it encounters an error or change of command.

6.This continues for each door until the open sensor on each door is tripped, if one door opens first the other continues to operate until it is open too, or until it encounters an error or command change. For the door to be stopped in motion the open sensor need only be tripped for a millisecond, but it must then remain tripped for atleast a second, otherwise the motor will begin moving again.

7.Once both doors are open the status 01 is sent to the camera safety board by the master. And the system is ready for another command.

**Closing**

1.The positions of the doors are read from the sensors, the camera safety board is informed of the state of the motor system.

2.The master and slave motors read the 10 command from the safety board and prepare to act on it, if either is already closed[4] then at this point the programs on the motors will run such that if either door is not closed it can close, but a door that is already closed will not move.

3.The master then sends an allow=1 pulse to the slave, if the slave is not already closed then it will be listening for this pulse. The slave will detect this pulse within the 15 seconds for which the pulse is active wherever it was in its program (excluding unusual places visited during the final steps of closing)when the pulse began. At this same moment as the allow=1 pulse begins the signal from master to safety board will change to 11. If both doors are already closed the allow=1 pulse will not be sent, the status will not change to 11 and the individual motors will continue waiting for a 01 open command.

4.The master terminates the allow=1 pulse setting the allow line back to 0. This is a signal to the master and slave to start a 2 second countdown then begin simultaneously closing.

5.The doors now move, they should be synchronised however if either encounters an error, or they began from indeterminate positions that were not the same, this will not be the case. Brief errors lasting less than 1 second will cause the motor that experiences them to pause for 1 second, errors or changes of command that persist for longer will result in the stopping of the motor experiencing them. If the slave is stopped due to an error then the master will always stop too. If the master is stopped due to an error then the slave may continue moving unless it encounters an error or change of command.

6.This continues for each door until the closed sensor on each door is tripped, if one door closes first the other continues to operate until it is closed too, or until it encounters an error or command change. For the door to be stopped in motion the closed sensor need only be tripped for a millisecond, but it must then remain tripped for atleast a second, for greater reliability assume 2 seconds, otherwise the motor will begin moving again.

7.Once the master door is close enough to the closed sensor to trigger it the master begins waiting for the slave to report that it has triggered its own closed sensor by sending a 10 status to the master. If the slave has reached the position necessary to activate the sensor before the master then the master detects this as soon as it also reaches that position. The output to the camera safety board is still 11.

8.The master now sends a pulse of allow=1 to the slave.

9.The slave has been waiting for this pulse since the moment it reported 10 to the master, once the allow=1 pulse has been detected the slave waits until the pulse ends.

10.When the master terminates the allow=1 pulse the master and slave both begin a 2 second countdown after which they simultaneously both move a fixed number of steps in the closing direction. If this fixed number is set too high it becomes possible for the door to be driven through the sensor so the value of the variable CloseDist must be set with extreme caution.

11.Once the fixed number of steps is complete the slave begins reporting that it is closed, 10 on the status lines. The master is programmed to wait until 100 milliseconds after the time at which the slave would begin reporting 10 before the master checks for the 10 signal.

12.If this 10 signal from the slave is detected by the master the master can then stop reporting 11 to the safety board and report 10 to the safety board to say that the doors have successfully closed. If 10 coming from the slave is not detected by the master then the slave has not closed properly and the master sends the flashing 00-10-00… error message to the camera safety board. This flashing error message would also be sent if either the master or slave had encountered an error at any point in the closing sequence after what is described as “7” here. After the error message is finished then depending on exact circumstances a 00 error may continue to be reported, a closed position may be reported or and indeterminate position may be reported. But the flashing 00-10-00-… code should have alerted the camera safety board to the fact that even if 10 is now reported the doors have not properly sealed.

# Further Notes

1. During the closing/opening operation the doors will stop immediately when the closing/opening sensor is pressed, when an error occurs or when the command from the camera safety board changes. They will then wait 1 second, if the sensor is no longer active after that second, or the error has ceased or the command returned to the command that set the doors in motion, then

motion will resume. If the sensor remains active after 1 second, or an error or change of command persists for this long then motion will not resume. The system will report it's new status to the safety board and wait for further instructions from it.

2. When the motor is moving stopping is effectively instantaneous (< 1 millisecond), the delays then come before motion in another direction begins.

3.An error on the slave will always cause the master to stop however an error on the master will not always stop the slave. It may be worth considering setting the camera safety board up in such a way that upon receiving an error status from the master for more than a few seconds it sends out a 11 or 00 command for 2 seconds atleast to stop the slave. This type of programming in the camera safety board may however introduce the risk of causing un-necessary stops. When the master does stop because of an error on the slave it will stop as soon as the slave sends the error message but the sending of the error message may come some seconds after the slave has encountered the error.

4.If either is already activating its closed sensor, regardless of whether it is closed enough to be properly sealed.

5.This checking that the sensor signal or command has lasted for more than a second is done by taking two readings of an input 1 second apart and making sure they are the same. For absolute reliability an input should therefore last longer than the time taken for the part of the program it is found in to loop twice. During the main parts of opening and closing sequences this means that for absolutely reliable detection of an input it should last for 2 seconds. Two random fluctuation errors occurring precisely one second apart could also appear to be a signal that was not truly being sent, but it is considered unlikely that random errors so precisely separated in time should ever occur.