Automated Alignment of Fuel Assembly Handling Tool

Pushpak Samgir,Rohan Pradhan,Ayush Sharma, Abhishek Patil

Student StudentStudentStudent

Department of Electronics and Telecommunication

SIES Graduate School of Technology,Nerul, Navi Mumbai

Abstract :The Advanced Heavy-Water Reactor (AHWR) is the latest Indian design for a next-generation nuclear reactor. AHWR contains large number of fuel cluster which are placed inside the reactor. Transportation of fuel cluster from Reactor unit to storage bay with utmost accuracy is done through an algorithm designed to lift the cluster.

Keywords—AHWR, PHWR

1. INTRODUCTION

The Advanced Heavy-Water Reactor (AHWR) is the latest Indian design for a next-generation nuclear reactor that burns thorium in its fuel core since India has a large abundance of thorium reserve compared to uranium which is used in other reactor.The proposed design of the AHWR is that of a heavy-water-moderated nuclear power reactor that will be the next generation of the PHWR type. It is being developed at Bhabha Atomic Research Centre (BARC), in Mumbai, India and aims to meet the objectives of using thorium fuel cycles for commercial power generation. The AHWR is a vertical pressure tube type reactor cooled by boiling light water under natural circulation. A unique feature of this design is a large tank of water on top of the primary containment vessel, called the gravity-driven water pool (GDWP).

Since AHWR is a very large project, in which fuel alignment system is a small part of it. AHWR contains large number of fuel cluster which are placed inside the reactor. Each fuel cluster has its own lifespan, once the lifespan of a particular cluster is finished that cluster is then discarded. AHWR consist of 2 sections, the 1st one where the fuel cluster is kept when it is operating i.e. the reactor and the 2nd one where once the lifespan is over the fuel cluster is shifted i.e. the storage bay. Storage bay exist because of following reason i.e. even after the lifespan of fuel cluster is over it still consists some radioactivity. Thus to neutralize the fuel cluster we keep the fuel cluster in the storage bay for several years until its radioactivity is reduced to level where it doesn't harm anyone. Inside the storage bay, the fuel cluster is organized in its pre-decided way; this is where our project comes into picture. Once the fuel cluster is brought in the storage bay it is placed in the decided slot with the help of 'Fuel Alignment system'.

.

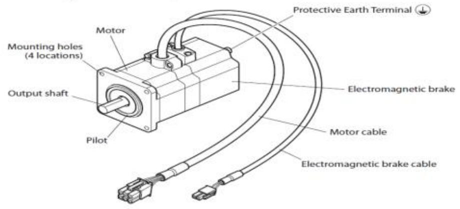
2.LITERATURE SURVEY

This project will upgrade the manual handling of fuel cluster with automated control of fuel cluster handling tool. Which will eventually improve the accuracy of fuel handling. The fuel being radioactive in nature needs to be disposed with utmost accuracy, as it has a huge risk of radioactive leakage. Automated alignment of fuel cluster handling tool is the newtechnology being developed for proper disposal of the burned off fuel cluster.

3.MECHANISM

A.STEPPER MOTOR

The main purpose of our project is move the fuel cluster from one place to another desired place. This is achieved by connecting the fuel cluster with the locking system. The movement of this locking system is controlled by the stepper motor. The RKII Series stepper motor is compact and produces high torque. A full-time micro stepping driver controlled by a digital system improves the vibration characteristics of the 0.72° stepper motor.


Protective Earth Terminal ⏚
Motor
Mounting holes (4 locations)
Output shaft
Pilot
Electromagnetic brake
Motor cable
Electromagnetic brake cable

B,STEPPER MOTOR DRIVER

Stepper motor is used for the movement of arm. Our Stepper motor driver as well as the stepper motor belongs to Oriental motor. Stepper motor driver provide us various feature that are required for our project. Out of the various feature, our main reason for choosing the oriental motor stepper motor driver is that it supports Modbus communication, which is the main element of our project. Along with the availability of MEXE02 i.e. software designed by oriental motor for testing purpose, we can control the stepper motor by Modbus communication as well as USB protocol. Driver provides us with CN5 ‘Sensor Signal’ which is used to control the limit switch which is one of the important element of our project.

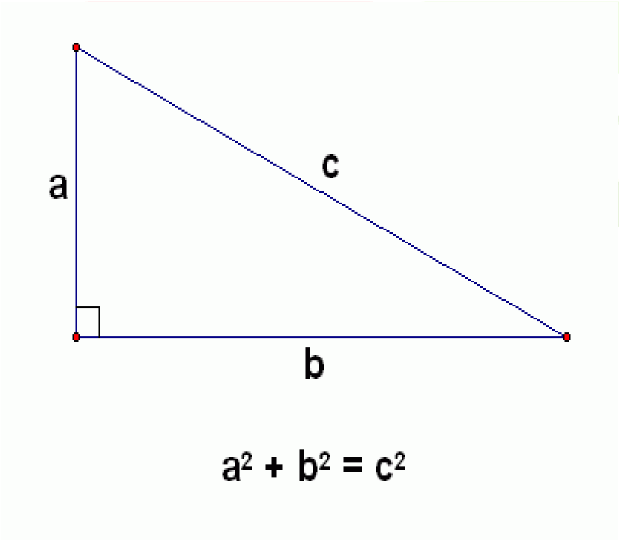
C. LASER SENSOR

In the process of moving the fuel cluster, the most important parameter that come into picture is detection of centre of the fuel cluster.

Once the center is detected, the locking system will be able to lock the fuel cluster for the purpose of movement. Detection of center is done with the help of pythagoras theorem i.e. laser sensor detects the centre of the laser line and calculates the base distance of the triangle. Since the radius of the cluster (which acts as the hypotenuse) is known, laser sensor travels perpendicular to the base line of the laser sensor upto the centre of the cluster which is calculated using, (hypotenuse= √ base^2 － height^2). Once the centre is detected the fuel cluster is locked into the arm and moved.

D.Software used

1) MEXE02



MEXE02 is software provided by the oriental motor for the testing purpose of their stepper motor and stepper motor driver. It is embedded with various functions to stepper motor in every possible limit and extension. During our initial stage of our project we tested the motor using MEXE02 through USB connection. The testing of motor included the forward and reverse movement, limits of the motor in the area of speed, sequential rotation i.e. continuous rotation through 2 different commands, linked motion command i.e. rotation of motor in 2 different directions using 2 different commands but it is continuous movement without any interference or any separate command. These were the basic function but main function includes status monitoring of RS 485 communication, Alarm monitoring for testing the limit switch of our project and also the JOG operation which is further required in our project.

.

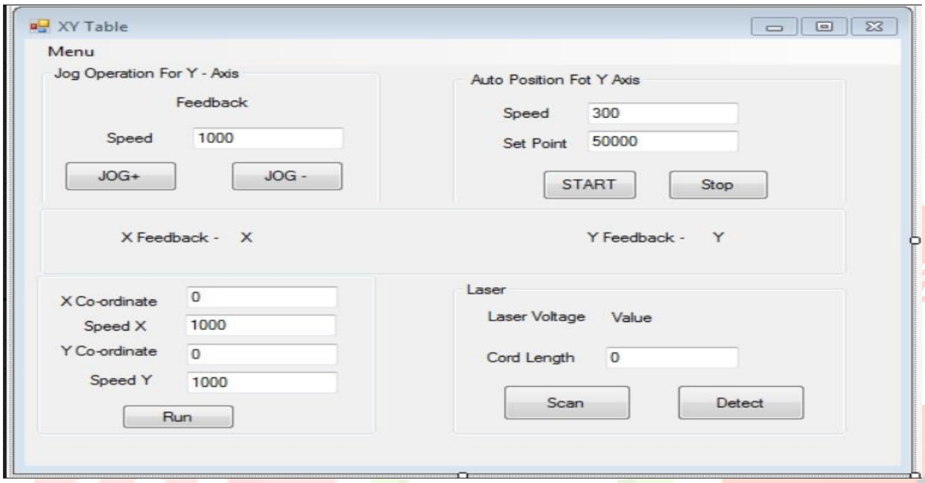
2) RADZIO MODBUS SIMULATOR

Radzio is Modbus stimulator that we used to test our established Modbus communication link. It is the most important parameter of our project. In our project Radzio played an important role as all the available or the known register in our project were yet to test and using Radzio we were able to accomplish the connection between stepper motor and our system and we were able send data to the register, upon which stepper motor used to enact their rotation direction, speed, number of step etc. As we have used 2 stepper motor drivers, it was important to operate the stepper motor driver sequential with a single command. It was important to test both the stepper motor driver which was difficult to test in MEXE02 software.

3) Establishment of the Modbus Communication:

Establishment of Modbus communication of the important part of the project, as our project is completely based on Modbus communication. There are 2 type Modbus based communication, they are Modbus RTU and Modbus TCP/IP. We have used Modbus RTU as it is most suitable for our project. Modbus RTU is an open, serial (RS-232 or RS-485) protocol derived from the Master/Slave architecture. It is a widely accepted protocol due to its ease of use and reliability. Modbus RTU is widely used within Building Management Systems (BMS) and Industrial Automation Systems (IAS). This wide acceptance is due in large part to Modbus RTU's ease of use. Modbus RTU messages are a simple 16-bit CRC (Cyclic-Redundant Checksum). The simplicity of these messages is to ensure reliability. Due to this simplicity, the basic 16-bit Modbus RTU register structure can be used to pack in floating point, tables, ASCII text, queues, and other unrelated data. MODBUS is considered an application layer messaging protocol, providing Master/Slave communication between devices connected together through buses or networks.

IV.GUI DESIGNED FOR THE CONTROL



Speed 1000

Speed 300

Set Point 50000

X Co-ordinate 0

Speed X 1000

Y Co-ordinate 0

Speed Y 1000

Cord Length 0

The main objective is to relocate the fuel cluster from one place to another inside the storage bay.GUI is divided into 2 parts i.e. "JOG Operation" and "Auto Position". The first one i.e. the "JOG Operation" is used to provide movement of stepper motor in a particular way. The stepper motor rotates until and unless the jog command is going and it stops once it stops receiving the jog command. The Jog operation section consist of Feedback, Speed control, JOG+ and JOG-. As the name suggest, feedback is used to provide the real time feedback of the stepper motor rotation. Feedback is necessary for many reasons such as the operator must have confirmation that the motor have moved from one place to another as per the command he/she has given. Another reason is that, if there is failure of software or hardware the operator will have the feedback, as the operator would be keen to know the last operation performed and the feedback will help to solve the problem. Also it is useful during the operation of command given through the "X co-ordinate" and "Y co-ordinate".

Another feature of jog operation is "JOG+"; we have designed the operation in such a way that, when the operator will press the jog+ button the motor will start motion in the forward direction until the operator stops pressing the button. It is useful when the operator does not know how much count needs to be provided to reach a point. In such condition operator will use jog command which will help the operator to move the motor in the desired way to reach the destination. Similarly we have provided "JOG – " command which will provide us with the same function but the only difference is that the motor operate in opposite direction (reverse direction). Next function is the speed control, as we move the motor in forward direction and reverse direction we have provided an option to control the speed of the motor. This will help the operator to move the motor faster if the destination is far away and slow down the motor as the destination arrives closer.

The second mode of the GUI is "Auto Position". This section consist feedback monitor, speed control option, set point option and two buttons which allows the motor to move forward, reverse and stop the motor. As discussed earlier feedback monitor is provided to show how much step the stepper motor took before stop command. Also the speed control option is provided to vary the speed of the stepper motor according to the operator requirement. "Set Point" is an option which is available in auto position and not in the jog operation because as discussed above jog operation is used when the operator is having no idea about the number of counts or the distance to reach the destination. But in auto position mode set point acts as end point command for the stepper motor driver. The driver stop the motor from rotating i.e. the driver set the stop bit to 1 and the motor stops rotating. For example if the operator set the set point to 60000 then the motor will stop rotating as the count reaches to 60000. In this scenario there are 2 situations that can occur i.e. ‘Absolute count’ and the ‘Incremental count’. In ‘Absolute count’, if the initial position of the stepper motor is 20000 then the driver will command the motor to rotate for another 40000 count until the total count reaches 60000. While in ‘Increment count’, if the initial position is assumed to be 20000 then the driver will command the motor to rotate another 60000 steps until the final count reaches 80000 count. These are the 2 situation that can occur while setting the set point depending upon what is selected absolute or incremental.

Auto position section is provided with 2 buttons that is start and stop. As the name suggest when the start button is pressed there are 3 command that goes to the driver i.e. first one is the speed that is set on the speed control option which goes to the register 1153(L) of stepper motor driver, the second is the set point which is inputted which goes to the register 1025(L) of the driver and the last one is the start command i.e. the value 8 which goes to the register 125. When all these command are inputted in the following register the motor will move as expected. The same start button is used to move the motor in forward as well as in reverse direction. For example, let us assume that the initial position of the motor is 10000 when the start command is given with speed of 1000 and set point of 20000, then the motor will move in forward direction. Now in second scenario when the operator will input the set point as 15000 then the motor will move in reverse direction as the current position of the motor is 20000. This is how the start is used to move the motor in both the direction forward as well as reverse. Next is the stop button which is used to stop the motor at any given instance of time. It is achieved by sending value 32 to the register 125.

There is also a separate section in the GUI which is labelled as ‘X - Coordinate’ and ‘Y - Coordinate’. This section is created for the convenience of operator i.e. if the operator wants to move both the motors simultaneously and reach the destination then he will just input the distance (mm) and then the driver will take care of the rest. For example if the operator provide 20mm in x direction and 50mm in y direction then the driver will initiate the process and then the motor will reach at the destination mentioned in the textbox. We have developed this section as it is the base of our project. Our final aim is that we just click the ‘Start’ button and then the system start scanning for the object in both x direction and y direction.

REFERENCES

[1] "India all set to tap thorium resources". Dec 2012

[2] http://www.orientalmotor.co.in/products/st/

[3] MEXE02. Data setting software mexe02 operating manual. mexe02, 54(6):2677–2686, Dec 1999.

[4] http:www.baumer.com/ch/en/product-overview/distance-measuement/c/287