MODULAR SPHERICAL COORDINATES POSITIONING SYSTEM

prep. eng. **Marius Sebastian RUSU**
"Petru Maior" University Tg-Mureş, Romania
E-mail: mrusu@upm.ro
Keywords: positioning, spherical coordinates, step-by-step motor, microcontroller, PC

Abstract. The positioning systems usually focus on the Cartesian coordinates rather than the spherical ones. There are many applications that require spherical positioning. This paper presents such a positioning system and its possible applications. The mathematical conversions are also covered by a subchapter of this paper. The three main domains that encounter in this project are: mechanics, electronics and informatics. Due to its conception, this system is very flexible and can easily be adapted to applications that are not a direct target for it.

1. Introduction

Linear, bi-dimensional (2D) and three-dimensional (3D) positioning domain is served by a variety of technical solutions, starting from low precision – low power systems and ending with high precision – high power systems.

Usually, the 3D positioning systems are designed and work in Cartesian coordinates. This approach is suitable for transport positioning systems inside a limited volume.

In tracking (targeting) applications the Cartesian positioning systems have major handicap. In some of these applications Cartesian positioning systems are useless.

This paper intends to present a spherical coordinates (orientation) positioning system. As you can see in the following chapters, the applications and command sequences design can easily be done in both Cartesian and spherical coordinates.

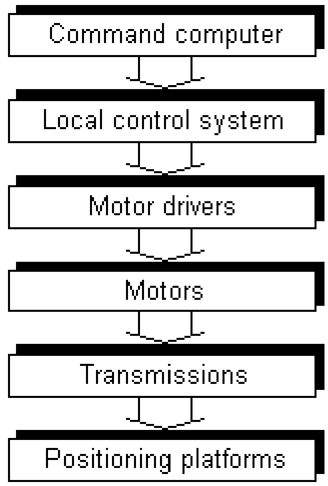
Another issue to be followed in the design of this system is its modularity. Each major function will be carried by a separate module, which can be replaced if needed. Therefore, the system is highly flexible.

2. Modular spherical coordinates positioning system architecture

The spherical coordinates positioning system is designed in a flexible mechatronic architecture. Being a mechatronic system, all the aspects of this domain will be encountered: mechanics, electronics and informatics.

This subchapter described the overall architecture of this project.

In the following picture, the block scheme is described, showing the base blocks of the modular spherical coordinates positioning system. Its function can be explained starting from the command computer which process the sequential model described by the user in a geometrical language, compiles it into a sequence of microinstructions that are necessary for the systems behavior and sends these microinstructions to the local control system. The local control system takes the microinstructions and decomposes them into the drivers command sequences. The drivers control the motors power flow as required at their command entries. Through mechanical transmissions the motors drive the positioning platforms.



Positioning system block scheme

Starting from this block scheme, the 3 types of elements will be described: mechanic, electronic and software and also their join across levels.

3. Mechanic blocks

The mechanical part of the positioning system is built of 4 base elements: system support, rotation axis, transmission elements and positioning platforms.

The system support must be rigid and must have the mechanical strength to sustain the mechanical elements and the motors. The rotation axis are also mechanical resistant elements as they ensure the loaded platforms rotation (motors, elements, etc.).

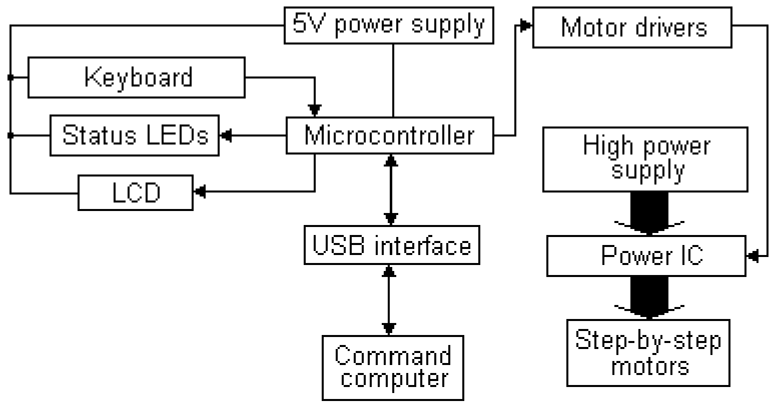
The transmission elements are designed with subunit transmission ratio gears for increased precision and handling. We have chosen this approach instead of belts because of the rigidity of the power transfer and the lack of elasticity and mechanical sliding.

The two platforms are designed such as one sustains the second engine, its transmission, the second platform and the positioned element, while the second platform sustains the positioned element.

In case the spherical positioning is a transport one (limited volume), the element for the second platform is the linear positioning system for the determination of the sphere radius. This way, there is another platform for the transported element at the end of the linear positioning system.

4. Electronic blocks

The electronic part of the positioning system is made of two big circuits: low power command circuit, high power motor circuit.



Electronic block scheme

The schematics are built around a Microchip 18F family microcontroller. This microcontroller grabs microinstructions from the command computer through the USB interface, processes them and send the sequences to the step-by-step motor drivers. The status of the local control system (microcontroller) can be traced using the status LEDs and the LCD display. To interfere with the system behavior a keyboard is available. The power supply for the area of the schematics is a 5V low power one.

The high power part of the schematics contains the power ICs that actually connects the motor to its power supply. The commands for these ICs are given by the drivers. The power flow is represented by the thick line.

This system can work in two regimes: *interface regime* and *autonomous control regime*.

The *interface regime* converts and instantaneously executes the commands from the command computer. For this regime, a command computer is required (PC, microcontroller, process computer, etc.)

The *autonomous control regime* sets the microcontroller into program interpretation mode, meaning it will run a minimal program of microinstructions stored in its memory in single shot mode (only once) or periodically. This mode replace the command computer, the microinstructions are stored internally. This way, no command computer is needed and the system can run independently. This also ensures its portability.

The schematics are physically distributed on a few PCBs: the microcontroller, the driver, and the USB interface on one board, the LEDs on one board, the LCD independently, the keyboard independently, the power supplies independently, the power ICs on another board. This way, the electronics are modular and the blocks can easily be replaced or upgraded if the connection protocol is followed.

5. Informatics blocks

Probably the most complex aspect of the positioning system is the software one. The entire behavior is directed by the states implemented in the software.

The software blocks are located in the microcontroller and in the command computer.

The microcontroller application describes the two regimes: interface regime and autonomous control regime. The LCD display control and status LEDs control, as well as the keyboard scan run independently from the regime. That is, they run in both regimes. For displaying the dedicated libraries are used. The keyboard scan is handled on a timer interrupt.

In the interface regime, when receiving a microinstruction on the USB port, the microcontroller decomposes it into step-by-step driver commands and send them to the driver. In the autonomous control regime, the source for the microinstructions is the microcontroller’s memory. Because of the physical limitations, the program stored by the microcontroller’s memory is not very big.

The command computer applications can be easily extended. The base and required elements are the USB interface program and the command program. The USB interface program receives the bytes sequences, packs them and sends them to the port as well as receives the sequences from the port, unpacks them and sends them to the higher level application. The command program receives a file containing a microinstructions sequence and sends these microinstructions through the USB interface program to the microcontroller. This command program is responsible with serializing the decision and cycling microinstructions. This program’s role is taken by the correspondent program in the microcontroller in the autonomous control regime. This program acts like a command interpreter.

The application suite also contains the positioning language compiler that converts the two specific languages into microinstructions sequences. The positioning language supports spherical coordinates as well as Cartesian coordinates. For generating the source file for this compiler a text editor can be used, or a dedicated graphical editor. Also, some extensions for design environments such as Autocad, 3D Studio, Orcad, etc. can be built to allow exporting from these environments into compilers source files.

Another application can be the TCP server that interfaces the TCP clients and the positioning system.

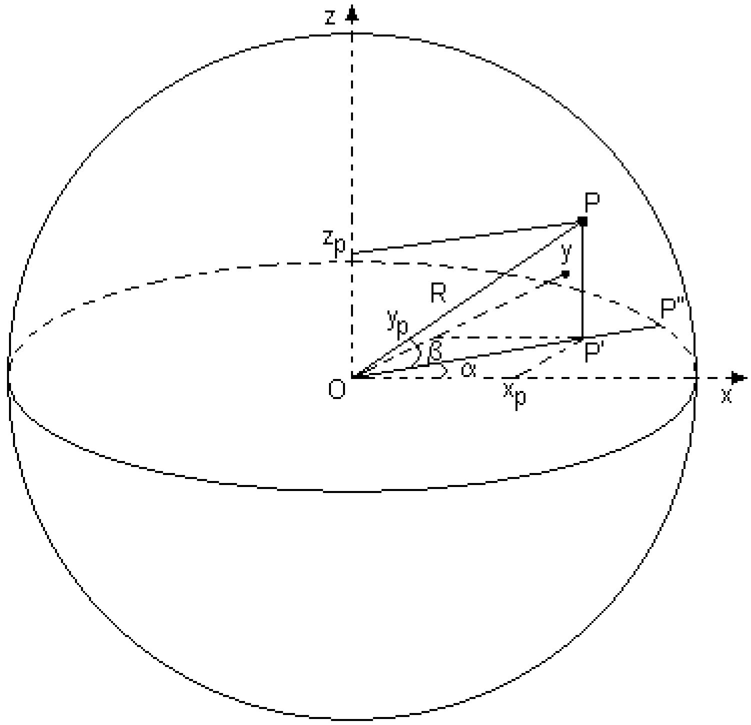
6. Geometrical modeling. Coordinates conversion.

This subchapter tries to present the geometrical conversions for the coordinates from the spherical domain to Cartesian domain and vice-versa.

As shown in the precedent chapters this system can be used in both finite distance positioning and infinite distance positioning (infinite angular positioning). This paper presents only the finite constant distance positioning.

For the coordinates conversions a unique starting geometrical model will be used. This model includes both coordinates systems. Also, only one reference system is considered in the center of the sphere.

The following picture shows the 3D base model.

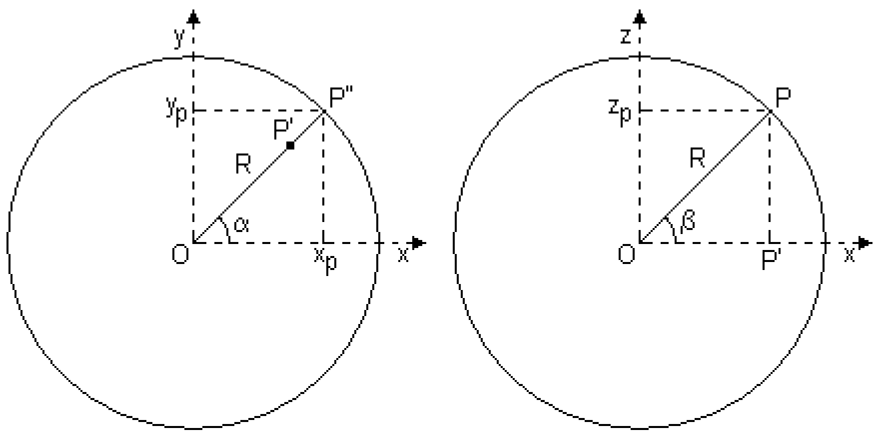


Coordinates systems (Cartesian and spherical)

The notations on the picture are:

a) P – current point (where the positioning is required)
b) x_p, y_p, z_p – Cartesian coordinates for point P (in xOyz system)
c) α, β, R – spherical coordinates for point P (in spherical system)
d) P' – the projection of point P on plane xOy
e) P" – the intersection between the sphere and the planes OPP' and xOy (first quadrant)
f) O – the origin (positioning system location)

The xOy and OPP' sections are shown bellow



Coordinates section of the positioning

As previously shown, there are two conversion directions: spherical-Cartesian and Cartesian-spherical

Starting from simple trigonometric equations, one can obtain the **spherical-Cartesian conversion** as described by the following equations system:

$$\begin{cases} x_p = R \cdot \cos\alpha \\ y_p = R \cdot \sin\alpha \\ z_p = R \cdot \sin\beta \end{cases} \quad (1)$$

In a similar way, one can obtain the **Cartesian-spherical conversion** in the equations system bellow:

$$\begin{cases} \alpha = arctg\frac{y_p}{x_p} \\ R = \sqrt{x_p^2 + y_p^2} \\ \beta = \arcsin\frac{z_p}{\sqrt{x_p^2 + y_p^2}} \end{cases} \quad (2)$$

These conversions allow building the design application in any of the coordinates system, the final compiling being handled by the command computer.

7. Conclusions

This positioning system is useful in limited spherical volume transport applications, as well as in tracking and targeting applications in unlimited volume. Unlimited volume targeting behavior can be suitable for tracking devices, radars, projectors, vector lighting, vector plasma jet devices etc.

The modularity of the design brings increased flexibility, the electronic and informatics modules being easily replaced, as far as the interface protocol is respected.

8. Bibliography

[1] Duka, A.V, Rusu, M.S., Haller, P. – Programarea interfețelor în LabWindows, Îndrumător de laborator, Ed. UPM, Tg-Mureş 2005

[2] Dumitriu, L., Iordache, M. – Teoria modernă a circuitelor electrice, Ed. All, Bucureşti, 1998

[3] Microchip WebSite – www.microchip.com

[4] Morar, Al. – Sisteme de comandă a motoarelor pas cu pas implementate pe calculatoare personale, Ed. UPM, Tg-Mureş, 2002

[5] Popa, I. – Inginerie software pentru conducerea proceselor industriale, Ed. All, Bucureşti, 1998

[6] Suciu, M., Popescu, D., Traian, I. – Microprocesoare, microcalculatoare şi roboți în automatizări industriale, Ed. Tehnică, Bucureşti 1986

[7] Ştefan, Gh. – Circuite şi sisteme digitale, Ed. Tehnică, Bucureşti, 2000