طراحی مهندسی سیستم کنترل الکتریکی یک بازوی صنعتی با ۵ درجه آزادی

دکتر امیر ابوالفضل صورتگر
عضو هیات علمی دانشگاه اراک
a_a_suratgar@yahoo.com

دکتر محمد پویان
عضو هیات علمی دانشگاه شاهد
mmandolakani@yahoo.com

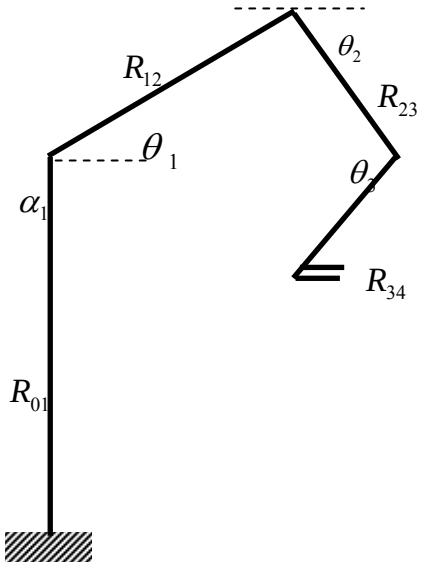
حامد عقیلی
کارشناسی ارشد الکترونیک و عضو باشگاه پژوهشگران جوان
H_aghili2004@yahoo.com

چکیده :

توسعه علم رباتیک موجب گشته تا روبات ها نیز در انواع مختلف و در راستای کاربرد های متنوع مورد استفاده قرار گیرند. رباتی هم که در این مقاله نحوه طراحی آن مطرح گشته دارای قابلیت های صنعتی و عمومی می باشد. یکی از خواص ویژه آن قابلیت نصب و کنترل با کامپیوتر های خانگی امروزی است که صرفاً با نصب پورت مربوط به کامپیوتر و نصب نرم افزار مورد استعمال سیستم ربات با کامپیوتر منزل شما کنترل می گردد از طرفی قابلیت کنترل آن با کامپیوترها وهر نوع میکرو کنترولر دیگر ممکن است آنچه که از طراحی این ربات انتظار می رود اینست که بازوی ربات بتواند جسمی را از نقطه‌ای معین برداشته و در ارتفاع دلخواه کاربر از مبداء مورد نظر به مقصد جابجا نموده و خود را به حالت اول بازگرداند. از طرفی باید این ربات برای استفاده کاربر نیز بسیار راحت بوده و با سؤالات پیچیده در مورد مشخصات جسم مورد نظر، کاربر را خسته ننماید. در جهت نیل به این اهداف ربات را از ابعاد مکانیکی، الکتریکی و کامپیوتری بررسی خواهیم کرد.

مقدمه:

امروزه با پیشرفت لحظه به لحظه در علوم مختلف، علم الکترونیک نیز با سرعتی دو چندان در جهت دستیابی به یافته های جدید پیش می رود، می رود که زندگی ماشینی ما انسانها را هر چه بیشتر در خود غوطه ور سازد. علم رباتیک نیز که به عنوان یکی از علوم روز و مباحث اصلی الکترونیک و مکانیک است، روز به روز خود را به انسانها نزدیکتر ساخته و وابستگی انسان را در صنایع به خود بیشتر نمایان می سازد. در این مقاله سعی شده است تا نمونه ای عملی و قابل ساخت از رباتها معرفی شود. از جمله خواص این ربات حفظ تعادل جسم سیال توسط یک بازوی صنعتی با۵ درجه آزادی می باشد که از لحاظ کنترلی بسیار مورد اهمیت است. از طرفی قابلیت کنترل آن با کامپیوترها وهر نوع میکرو کنترولر دیگرممکن است.



شکل ۱

متن مقاله :

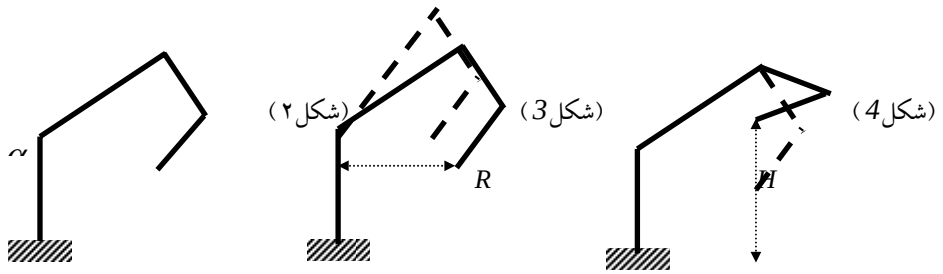
مطابق شکل ۱ بازوی ربات دارای ۵ درجه آزادی که شامل ۱ درجه آزادی در صفحه افقی و ۴ درجه آزادی در صفحه عمودی می باشد. به سبب اهمیت بحث کنترل پایدار در سیستم های رباتیک بازو باید توانایی برداشتن اجسام بصورت عمودی را بدون لب ریز شدن مایع داشته باشد. لذا حرکت های عمودی بازو باید بصورت هماهنگ و نظم یافته و بصورتی که انگشتی آن در همه حال حجم را موازی افق نگهدارد، باشند (مانند برداشتن لیوان آب توسط بازوی انسان) در این راستا سه فاز حرکت برای ربات پیش بینی شده است:

۱) فاز حرکت افقی(شکل۲)

۲) فاز یک حرکت عمودی که عامل تعیین شعاع (R) محل قرار گرفتن جسم می باشد (شکل۳).

۳) فاز دو حرکت عمودی که عامل تعیین ارتفاع (H) محل قرار گرفتن جسم با توجه به شعاع (R) محل قرار گرفتن آن می باشد: (شکل۴).

مطابق شکل:



۱) در فاز حرکت افقی تنها موتور α_1 باید فعال باشد و سایر موتورها قفل باشند.

۲) در فاز یک حرکت عمودی موتورهای θ_1 و θ_2 همزمان باید فعال بوده و سایر موتورها باید قفل باشند.

۳) در فاز دو حرکت عمودی موتورهای $\theta_۲$ و $\theta_۳$ و $\theta_٤$ همزمان باید فعال شده باشند و سایر موتورها قفل گردند. در این فاز الزامی است تا نسبت سرعت دور موتور $\theta_۳$ به $\theta_٤$ دو برابر باشد .

با توجه به مطالب فوق الذکر سیستم کنترل حرکات ربات دارای ۳ تابع اصلی تحت عناوین تابع حرکت $(\alpha_۱)$، تابع حرکت ($\theta_۱$ و $\theta_۲$) و تابع حرکت ($\theta_۲$ و $\theta_۳$ و $\theta_٤$) می باشد، که ما برای راحتی کار این توابع را بترتیب با نام های $LoopI$ و $LoopII$ و $LoopIII$ خطاب می کنیم.

جهت کنترل تمام این موتورها باید سخت افزار و نرم افزار مطابق فلوچارت تعریف فعالیت های ربات با یکدیگر هماهنگ باشند.

برای تابع $LoopI$ لازم است تا پارامتر زاویه $\Delta\alpha_۱$ مشخص گردد، تا تغییرات زاویه ای حول محور عمودی بر روی صفحه افقی مشخص گردد.

برای تابع $LoopII$ لازم است تا تغییرات شعاع (ΔR) محلی که بازو از شعاع آن مختصات حرکت خواهد کرد مشخص گردد.

$$\Delta\theta_۱ = \Delta\theta_۲ = \cos^{-۱}\frac{R}{R_{۱۲}}$$

برای تابع $LoopIII$ لازمست تا ابتدا شعاع (R) مشخص گردد تا ارتفاع (H) نیز از روی همان شعاع محاسبه شده، تعیین گردد. این به معنی ترتیب اجرای عملیات ها و توابع نبوده بلکه برای ترتیب استفاده از نتایج محاسبات یک تابع در تابع بعدی می باشد:

$$\Delta\theta_۳ = ۲\Delta\theta_۲$$

$$\Delta\theta_3 = 2\sin^{-1}\left(\frac{\frac{< \theta_1 \text{مقابل} \quad \text{ضلع} \quad + R_{01} - H}{2}}{R_{23}}\right)$$

$$\Delta\theta_3 = 2\sin^{-1}(\frac{R\tan\Delta\theta + R_{01} - H}{2R_{23}})$$

اگر سیستم دارای یک خط مشترک ۴ بیتی برای *Address Bus* و یک خط مشترک ۴ بیتی برای *Data Bus* باشد (بدلیل اینکه پورت های *I/O* در *Micro controller* ها و *Parallel Port* از *P.C*، هشت بیتی هستند و ما حداقل ۴ بیت دیتا برای کنترل *Step Motor* ها نیاز داریم) لذا می باید توسط یک *Decoder* ۴ بیت آدرس دهی شده و از طریق *Data Bus* ۴ بیت دیتای مورد نظر جهت کنترل *Step Motor* ها به درایور موتور ها ارسال گردند. تقسیم بندی *Bit*های ورودی وخروجی بشرح ذیل است:

عملکرد سخت افزار	تابع در نرم افزار	*Address Bus*				*Date Bus*			
		B_7	B_6	B_5	B_4	B_3	B_2	B_1	B_0
Reset	*Reset*	0	0	0	0	*	*	*	*
کنترل موتور α_1	*Loop1*	0	0	0	1	*	*	*	*
کنترل موتور θ_1 و موتور θ_2 توأماً	*Loop 2*	0	0	1	0	*	*	*	*
کنترل موتور θ_2 و موتور θ_3 و موتور θ_4 توأماً	*Loop 3*	0	0	1	0	*	*	*	*
کنترل موتور گیره ای و فیدبک از موقعیت جسم	*Loop give*	0	1	0	0	*	*	*	*

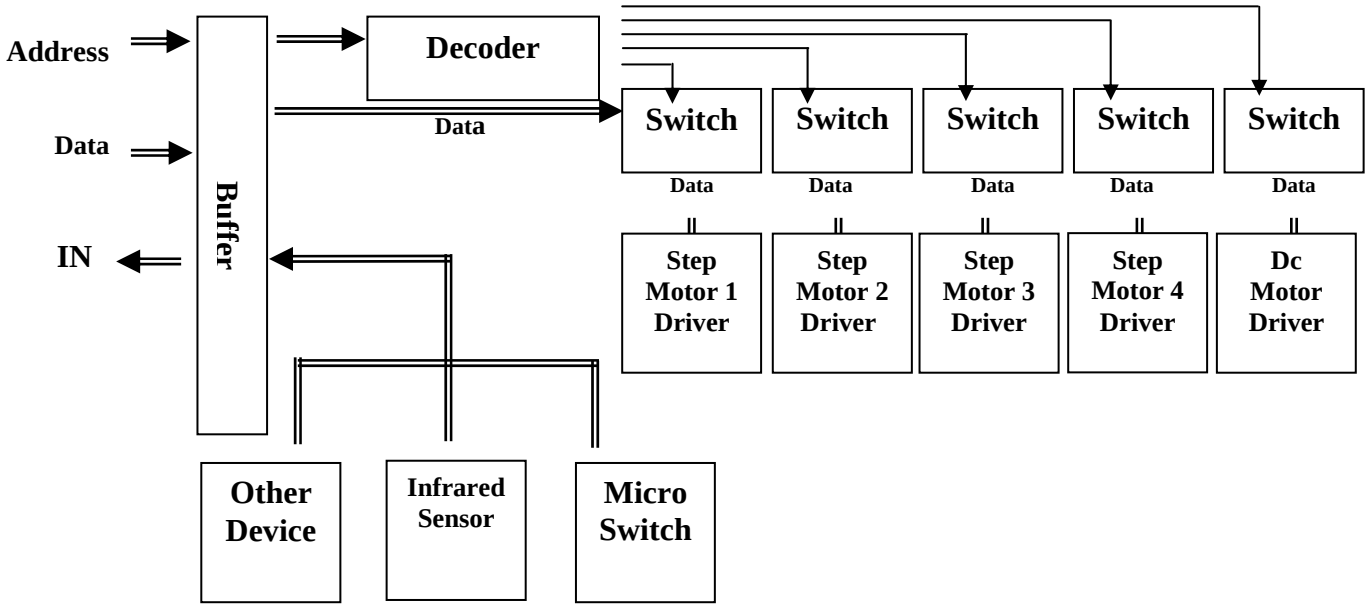
سخت افزار سیستم:

سخت افزار سیستم شامل یک بافر ورودی جهت انتقال اطلاعات کلیه *Bit* ها از *CPU* به سیستم سخت افزار یک *Max/Decoder* از نوع $۸ \rightarrow ۳$ و یا در صورت توسعه عملیات ها $۱۶ \rightarrow ٤$ و حداقل ۶ عدد بافر *Latch* ۴ بیتی قفل کردن ترانزیستورهای سوئیچینگ بیت های کنترل درایور *Sept Motor* ها می باشد.

در راستای کنترل عملیات ها از پایه *LE* در *IC Latch* ها برای چفت کردن *Data* درهر کدام از درگاهها و از پایه *OE* برای *Reset* کردن *Data* در خروجی سخت افزار استفاده می شود. *CPU*نیز که می تواند از هر نوعی اعم از *Micro controller* یا *Microprocessor* یا *P.C* باشد ، فقط لازم است تا ۸ بیت اطلاعات را بر روی بافر ورودی قرار دهد.

در بحث فیدبک، اینکه آیا جسم مورد نظر توسط ربات دیده شده است یا نه مطرح است . لذا چهار عدد دیود مادون قرمز فرستنده و همچنین گیرنده مادون قرمز نیز یک *IC* جهت کدینگ سیگنال ارسالی از فرستنده و یک عدد *IC* جهت کدینگ سیگنال دریافتی به ترتیب با شماره های *MC 14026* و *MC14027* وجود دارد. در این *IC* چهار بیت قابل انتقال جهت کدینگ است، لذا جهت دیدن جسم توسط سنسورها ،آن ها را ۲ به ۲ روبروی هم قرار می دهیم تا در یک صفحه بودن جسم را تشخیص داده و سپس از خروجی چهار بیتی گیرنده ها، یک *AND* ۴ بیتی با شماره *IC* ۷۴۰۸ را قرار می دهیم تا هر چهار *Bit* خروجی سنسورهای مادون قرمز توسط جسم قطع شده، سپس ربات تصمیم لازم را جهت گرفتن جسم بین انگشت هایش اتخاذ نماید.دیتای خروجی به صورت کد به *CPU* ارسال می گردد. این دیتای ارسالی باعث تصمیم گیری ربات مطابق *Loop give* می گردد.در صورت فعال شدن ۲ بیت از تعیین حالت موقعیت حضور یا عدم حضور جسم در محل مورد انتظار ربات برداشتن و عدم برداشتن آن مشخص می گردد. با این فیدبک *CPU* ربات تصمیم صحیح را اتخاذ می نماید.

نمودار سخت افزار:



به ازای هر *Sept Motor* یک بافر ۴ بیتی وجود دارد. برای همین باید برای حرکت ها متناسب با معادلات محاسبه شده در بخش های پیشین، موتورها آدرس دهی شوند. این معادلات توابع *LoopI* و *LoopII* و *LoopIII* و همچنین تابعی بنام *Loop give* برای که برای گرفتن یا رها کردن جسم به هنگام دیدن آن است، کنترل و تعریف می کنند.

تعریف عملیات های ربات:

الف) *CPU* مشخصات فنی ربات (طول ها و ...) را تحت عنوان، *Setup* دریافت می کند.

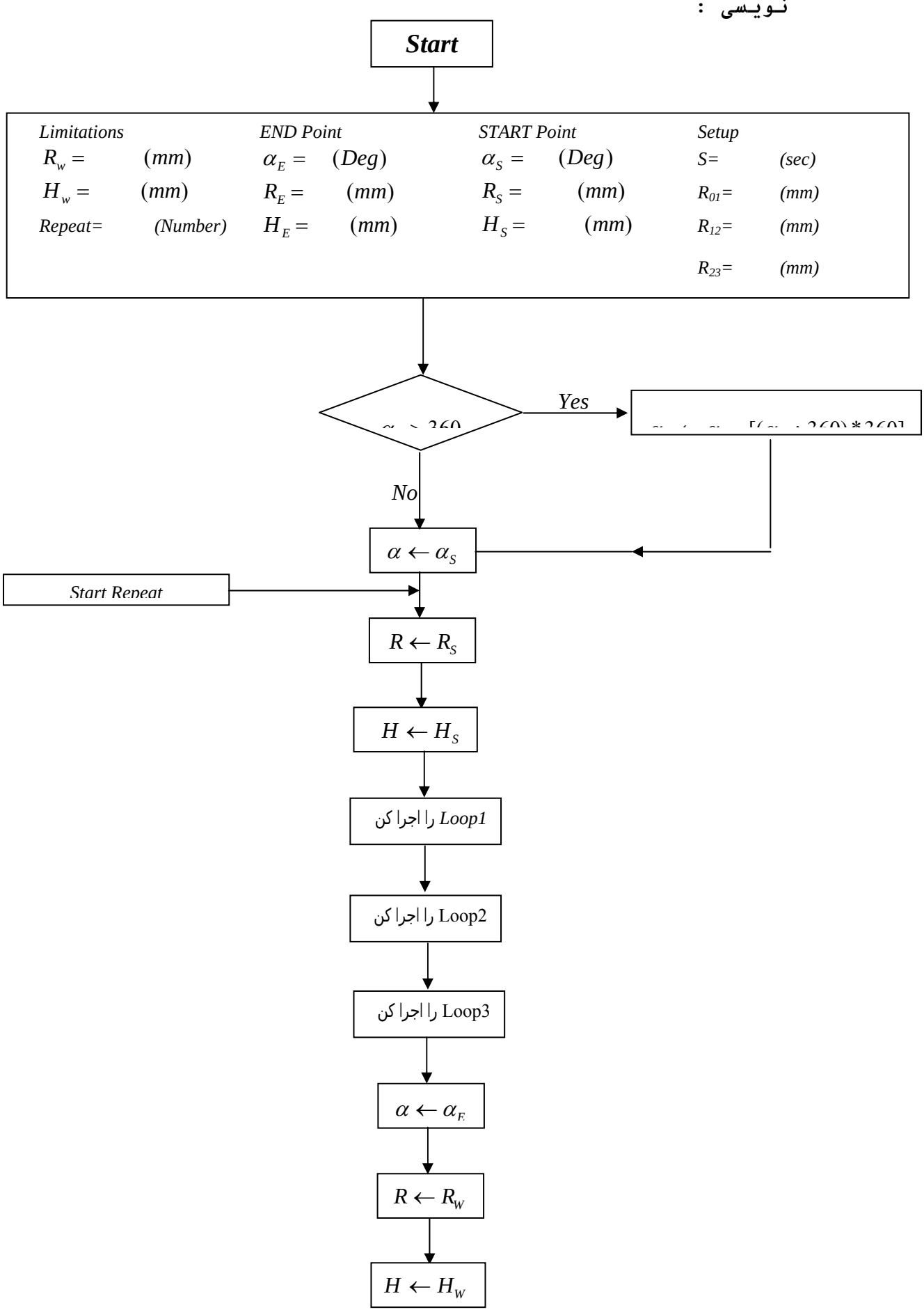
ب) *CPU* مختصات محل مبدأ و مقصد و نیز مسیر حرکت و چرخش بازو را در همین برداشتن و جابجا کردن جسم از طریق صفحه کلید دریافت می کند.

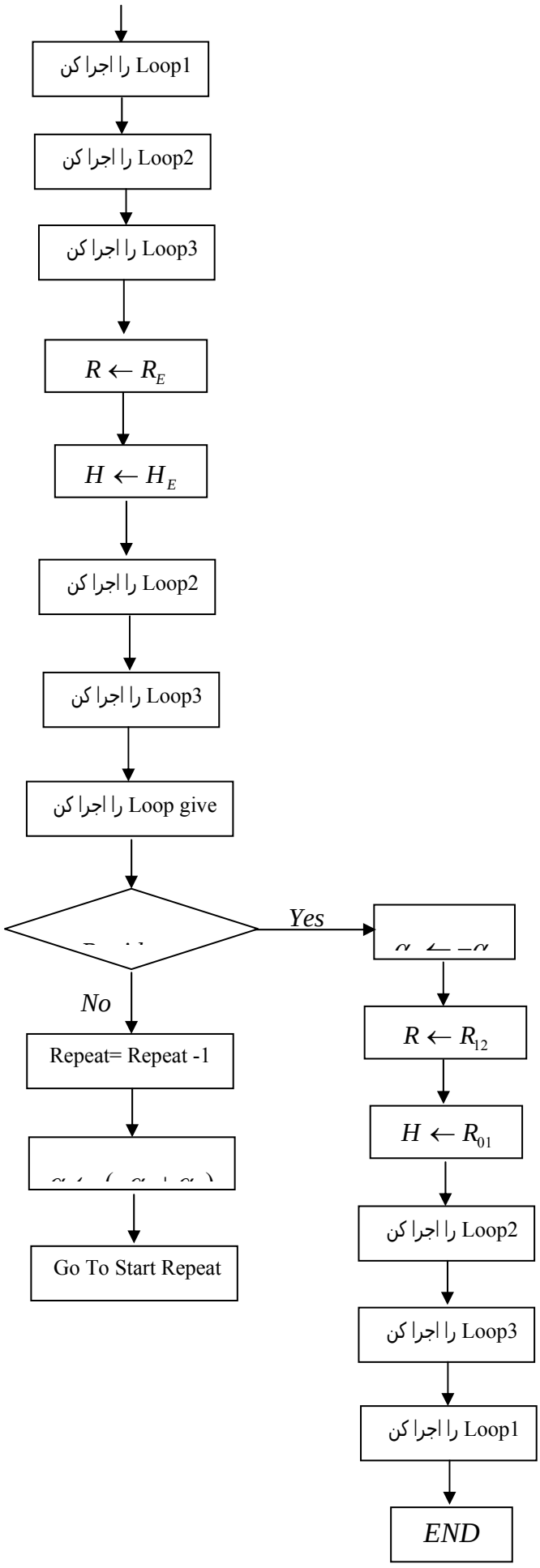
پ) جسم را از محل مبدأ پس از مراجعه برداشته و به سمت مقصد حرکت کند.

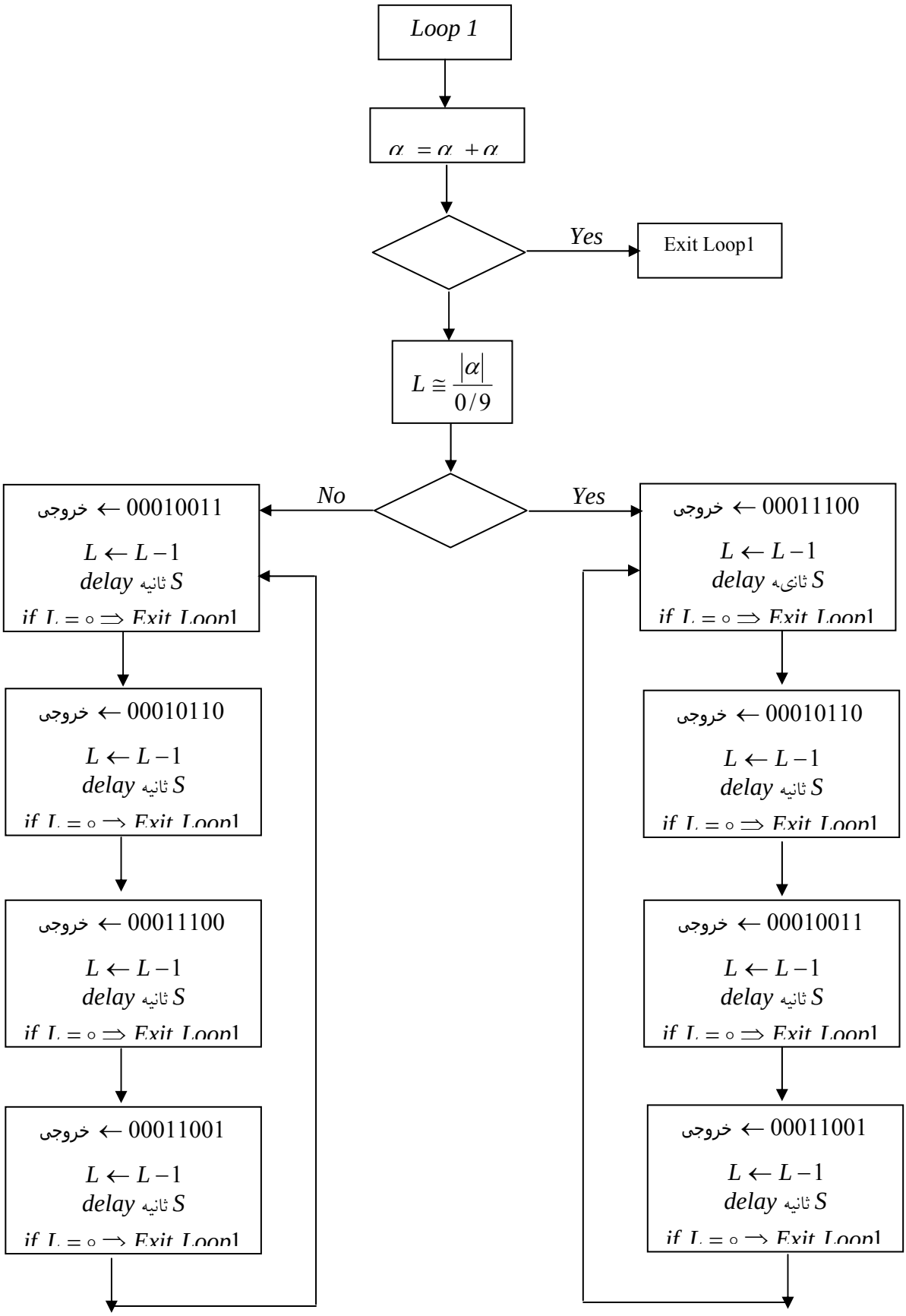
ت) پس از گذاشتن جسم در مقصد بر روی زمین در صورت درخواست تکرار عملیات آن را انجام دهد.

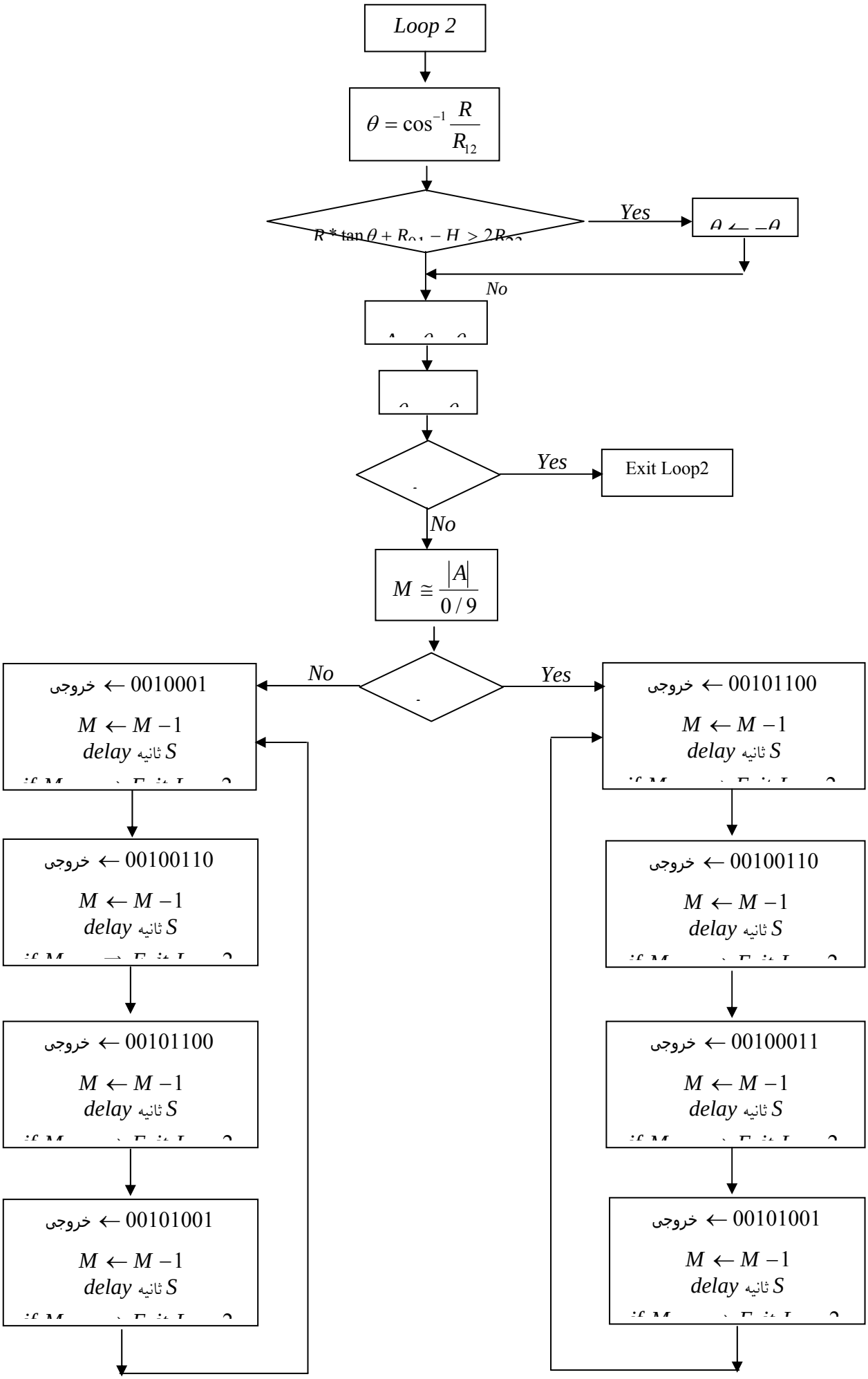
ث) در صورت عدم درخواست تکرار عملیات مجدداً به محل اول حرکت (نه مبدأ) برگردد و پارک شود.

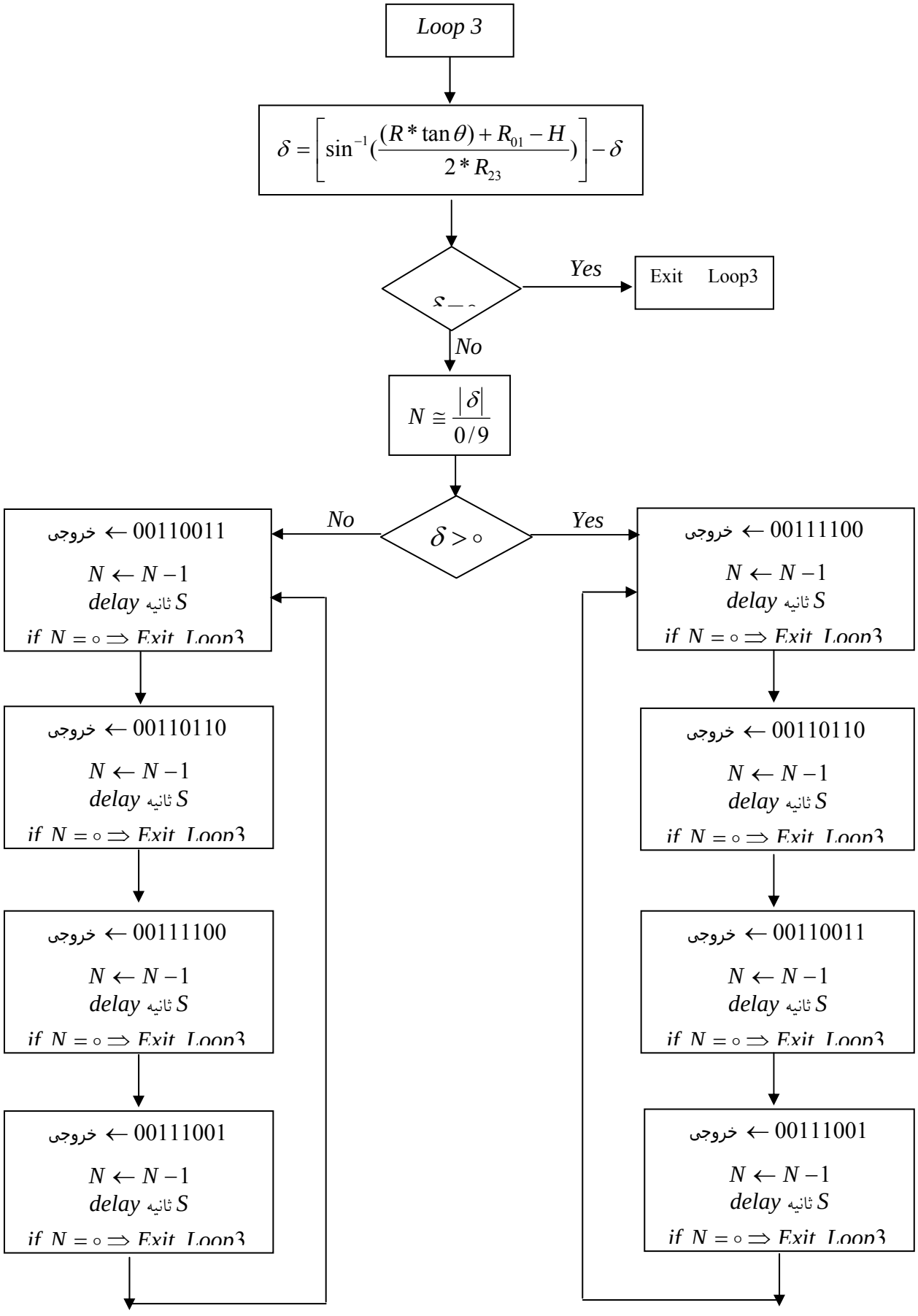
فلوچارت برنامه نویسی :

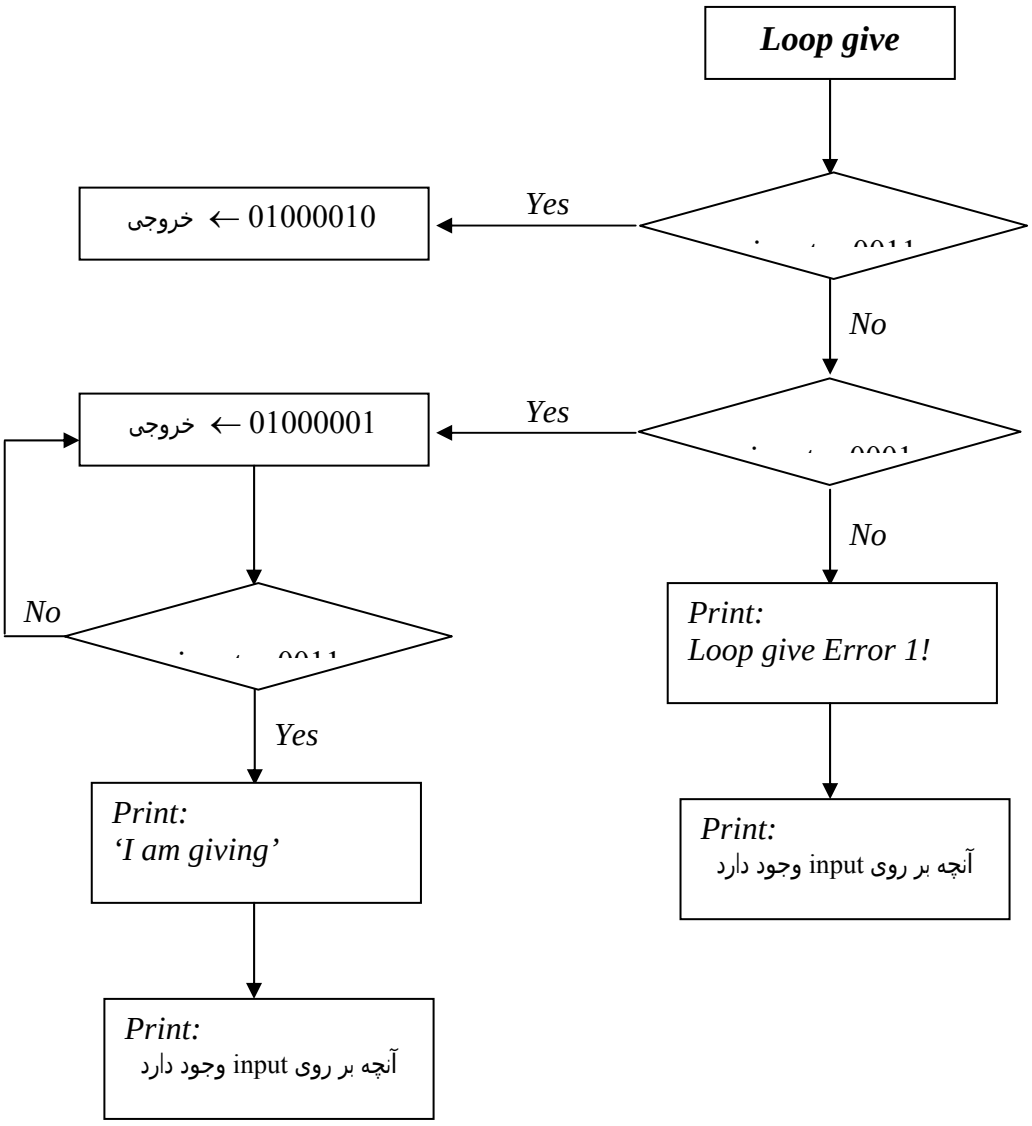



Loop1 را اجرا کن
Loop2 را اجرا کن
Loop3 را اجرا کن
$R \leftarrow R_E$
$H \leftarrow H_E$
Loop2 را اجرا کن
Loop3 را اجرا کن
Loop give را اجرا کن
Yes
No
Repeat= Repeat -1
Go To Start Repeat
$R \leftarrow R_{12}$
$H \leftarrow R_{01}$
Loop2 را اجرا کن
Loop3 را اجرا کن
Loop1 را اجرا کن
END


Loop 1
Yes
Exit Loop1
$L \cong \frac{|\alpha|}{0/9}$
No
Yes
خروجی ← 00010011
$L \leftarrow L-1$
delay ثانیه S
if L. = ۰ ⇒ Exit Loop1
خروجی ← 00010110
$L \leftarrow L-1$
delay ثانیه S
if L. = ۰ ⇒ Exit Loop1
خروجی ← 00011100
$L \leftarrow L-1$
delay ثانیه S
if L. = ۰ ⇒ Exit Loop1
خروجی ← 00011001
$L \leftarrow L-1$
delay ثانیه S
if L. = ۰ ⇒ Exit Loop1
خروجی ← 00011100
$L \leftarrow L-1$
delay ثانیه S
if L. = ۰ ⇒ Exit Loop1
خروجی ← 00010110
$L \leftarrow L-1$
delay ثانیه S
if L. = ۰ ⇒ Exit Loop1
خروجی ← 00010011
$L \leftarrow L-1$
delay ثانیه S
if L. = ۰ ⇒ Exit Loop1
خروجی ← 00011001
$L \leftarrow L-1$
delay ثانیه S
if L. = ۰ ⇒ Exit Loop1


Loop 2
$\theta = \cos^{-1}\frac{R}{R_{12}}$
Yes
No
Yes
Exit Loop2
No
$M \cong \frac{|A|}{0/9}$
Yes
No
خروجی ← 00101100
$M \leftarrow M-1$
delay ثانیه S
خروجی ← 00100110
$M \leftarrow M-1$
delay ثانیه S
خروجی ← 00100011
$M \leftarrow M-1$
delay ثانیه S
خروجی ← 00101001
$M \leftarrow M-1$
delay ثانیه S
خروجی ← 0010001
$M \leftarrow M-1$
delay ثانیه S
خروجی ← 00100110
$M \leftarrow M-1$
delay ثانیه S
خروجی ← 00101100
$M \leftarrow M-1$
delay ثانیه S
خروجی ← 00101001
$M \leftarrow M-1$
delay ثانیه S


Loop 3
$\delta = \left[\sin^{-1}\left(\frac{(R*\tan\theta)+R_{01}-H}{2*R_{23}}\right)\right]-\delta$
Yes
Exit Loop3
No
$N \cong \frac{|\delta|}{0/9}$
No
$\delta > 0$
Yes
خروجی ← 00110011
$N \leftarrow N-1$
delay ثانیه S
if $N = 0 \Rightarrow$ Exit Loop3
خروجی ← 00110110
$N \leftarrow N-1$
delay ثانیه S
if $N = 0 \Rightarrow$ Exit Loop3
خروجی ← 00111100
$N \leftarrow N-1$
delay ثانیه S
if $N = 0 \Rightarrow$ Exit Loop3
خروجی ← 00111001
$N \leftarrow N-1$
delay ثانیه S
if $N = 0 \Rightarrow$ Exit Loop3
خروجی ← 00111100
$N \leftarrow N-1$
delay ثانیه S
if $N = 0 \Rightarrow$ Exit Loop3
خروجی ← 00110110
$N \leftarrow N-1$
delay ثانیه S
if $N = 0 \Rightarrow$ Exit Loop3
خروجی ← 00110011
$N \leftarrow N-1$
delay ثانیه S
if $N = 0 \Rightarrow$ Exit Loop3
خروجی ← 00111001
$N \leftarrow N-1$
delay ثانیه S
if $N = 0 \Rightarrow$ Exit Loop3



فلوچارت برنامه نویسی این سیستم مطابق تعریف عملیاتهای ربات تدوین و سعی شده تا سازگار با $P.C$ یا میکروکنترلر یا میکروپروسسورها باشد.

مراجع:

۱) سخنرانی دکتر کارلوس – اردیبهشت ۸۰ – دانشگاه تبریز

۲)اینترنت – مجله IEEE ۱۹۹۸، ۱۹۹۷