MODULE Module1

VAR robtarget Objetivo:= [[0,0,0],[0,0,0,0],[0,0,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];

TASK PERS wobjdata obj\_trab:=[FALSE,TRUE,"",[[0,0,0],[1,0,0,0]],[[0,0,0],[1,0,0,0]]];

PERS string recv\_data;

VAR string msg:="";

VAR string old\_msg;

VAR string ident:="";

VAR num inicio;

VAR num final;

VAR num total;

VAR num string\_len:=0;

VAR bool okey;

VAR num tool := 0;

VAR num tool\_old := 0;

VAR num tipo\_mov:=0;

PERS bool confirmacion :=FALSE;

PERS bool confirmacion2;

PERS num control :=0;

PROC main()

!Inicializacion de variables

!IMPORTANTE: Iniciar ambas tareas con el puntero en main cada vez que se use el programa

confirmacion := FALSE;

old\_msg := "";

WaitTime 2;

WHILE TRUE DO

msg := recv\_data;

!TPwrite "Main";

!TPWrite old\_msg;

TPErase; !Muy importante añadir un TPWrite o un TPErase, en caso de no incluirlo no se ejecutará.

WaitTime(0.1);

!Si llega un mensaje nuevo de mas de tres caracteres:

IF (msg <> old\_msg) and (StrLen(msg)>3) THEN

TPWrite msg;

WaitTime(0.01);

!Extraemos el identificador, el cual define el tipo de instruccion

ident := StrPart(msg,1,3);

IF ident = "MOV" THEN !Instruccion de movimiento

string\_len := StrLen(msg);

total := string\_len-7-1;

!Se almacena el tipo de movimiento (MoveJ o MoveL) en la variable tipo\_mov

okey := StrToVal(StrPart(msg,5,1),tipo\_mov);

!Se confirma al programa que ha llegado el mensaje

Proc\_confirmacion;

!Se inicia el proceso "Movimiento"

Movimiento;

ELSEIF ident = "SIG" THEN !Instruccion de señal

!De momento solo sirve para activar o desactivar la herramienta, pero es

!facilmente ampliable para su uso con otras salidas

!Se extrae del mensaje el valor al que se debe poner la salida de la herramienta

!y, en caso de ser distinto al anterior, se cambia su valor

okey := StrToVal(StrPart(msg,4,1),tool);

IF (tool <> tool\_old) and (okey = TRUE) THEN

TPWrite("Herramienta = "+NumToStr(tool,0));

WaitTime(0.01);

tool\_old := tool;

SetDO DTool,tool;

ENDIF

!Se confirma al programa que se ha realizado la accion pertinente

Proc\_confirmacion;

ELSEIF ident = "CNF" THEN !Programa a la espera de un mensaje de confirmacion

!Se envia un mensaje de confirmacion

Proc\_confirmacion;

ENDIF

!Se guarda el ultimo mensaje recibido como tal

old\_msg:=msg;

ENDIF

ENDWHILE

ENDPROC

PROC Movimiento()

VAR num control :=0; !Variable que controla el bucle while

WHILE control = 0 DO

msg := recv\_data;

WaitTime(0.1);

!En el momento en el que se reciba una nueva instruccion, se comprueba su tipo y se

!actua de manera acorde.

!Se asume que los mensajes recibidos llegan en forma de secuencia "MOV - WOB - POS"

!Para mejorar el funcionamiento del programa deberia ampliarse para cuando se produzcan

!anomalias y se incumpla esta condicion

IF msg <> old\_msg THEN

ident := StrPart(msg,1,3);

TPWrite ident;

WaitTime(0.01);

IF ident = "RTY" THEN

!Mensaje de retry, indica que Visual Components va a reintentar el envio de la

!instrucción desde el principio

RETURN;

ELSEIF ident = "WOB" THEN

!Cuando se recibe un mensaje con identificado WOB, se guardan sus valores

!en el objeto de trabajo "obj\_trab"

inicio := StrMatch(msg,1,"X")+1;

final := StrMatch(msg,1,"Y");

total := final-inicio;

okey := StrToVal(StrPart(msg,inicio,total),obj\_trab.uframe.trans.x);

inicio := StrMatch(msg,1,"Y")+1;

final := StrMatch(msg,1,"Z");

total := final-inicio;

okey := StrToVal(StrPart(msg,inicio,total),obj\_trab.uframe.trans.y);

inicio := StrMatch(msg,1,"Z")+1;

final := StrMatch(msg,1,"QS");

total := final-inicio;

okey := StrToVal(StrPart(msg,inicio,total),obj\_trab.uframe.trans.z);

inicio := StrMatch(msg,1,"QS")+2;

final := StrMatch(msg,1,"Qi");

total := final-inicio;

okey := StrToVal(StrPart(msg,inicio,total),obj\_trab.uframe.rot.q1);

inicio := StrMatch(msg,1,"Qi")+2;

final := StrMatch(msg,1,"Qj");

total := final-inicio;

okey := StrToVal(StrPart(msg,inicio,total),obj\_trab.uframe.rot.q2);

inicio := StrMatch(msg,1,"Qj")+2;

final := StrMatch(msg,1,"Qk");

total := final-inicio;

okey := StrToVal(StrPart(msg,inicio,total),obj\_trab.uframe.rot.q3);

inicio := StrMatch(msg,1,"Qk")+2;

string\_len := StrLen(msg);

total := string\_len-inicio-1;

okey := StrToVal(StrPart(msg,inicio,total),obj\_trab.uframe.rot.q4);

!Se confirma al programa que se ha realizado la accion pertinente

Proc\_confirmacion;

ELSEIF ident = "POS" THEN

!Cuando se recibe un mensaje de tipo POS, se guardan sus valores en la

!variable de posicion "Objetivo"

inicio := StrMatch(msg,1,"X")+1;

final := StrMatch(msg,1,"Y");

total := final-inicio;

okey := StrToVal(StrPart(msg,inicio,total),Objetivo.trans.x);

!TPWrite StrPart(msg,inicio,total);

inicio := StrMatch(msg,1,"Y")+1;

final := StrMatch(msg,1,"Z");

total := final-inicio;

okey := StrToVal(StrPart(msg,inicio,total),Objetivo.trans.y);

!TPWrite StrPart(msg,inicio,total);

inicio := StrMatch(msg,1,"Z")+1;

final := StrMatch(msg,1,"QS");

total := final-inicio;

okey := StrToVal(StrPart(msg,inicio,total),Objetivo.trans.z);

!TPWrite StrPart(msg,inicio,total);

inicio := StrMatch(msg,1,"QS")+2;

final := StrMatch(msg,1,"Qi");

total := final-inicio;

okey := StrToVal(StrPart(msg,inicio,total),Objetivo.rot.q1);

!TPWrite StrPart(msg,inicio,total);

inicio := StrMatch(msg,1,"Qi")+2;

final := StrMatch(msg,1,"Qj");

total := final-inicio;

okey := StrToVal(StrPart(msg,inicio,total),Objetivo.rot.q2);

!TPWrite StrPart(msg,inicio,total);

inicio := StrMatch(msg,1,"Qj")+2;

final := StrMatch(msg,1,"Qk");

total := final-inicio;

okey := StrToVal(StrPart(msg,inicio,total),Objetivo.rot.q3);

!TPWrite StrPart(msg,inicio,total);

inicio := StrMatch(msg,1,"Qk")+2;

string\_len := StrLen(msg);

total := string\_len-inicio-1;

okey := StrToVal(StrPart(msg,inicio,total),Objetivo.rot.q4);

!TPWrite StrPart(msg,inicio,total);

!Se actualiza la variable "control" para salir del bucle

control:=1;

!Se confirma al programa que se ha realizado la accion pertinente

Proc\_confirmacion;

ENDIF

!Se guarda el ultimo mensaje recibido como tal

old\_msg := msg;

ENDIF

ENDWHILE

!Se manda la instruccion al robot de moverse en función del tipo de movimiento(MOV),

!objeto de trabajo (WOB) y posicion (POS)

!Una forma de ampliar el programa seria añadir funcionalidades como el cambio de herramienta,

!la configuracion (cf1,cfx,...) o la velocidad de movimiento

IF tipo\_mov=0 THEN

MoveL Objetivo,v50,fine,TVentosa \WObj:=obj\_trab;

ELSE

MoveJ Objetivo,v50,fine,TVentosa \WObj:=obj\_trab;

ENDIF

!Se confirma al programa que se ha realizado la accion pertinente

Proc\_confirmacion;

ENDPROC

PROC Proc\_confirmacion()

!Este proceso indica a la tarea de segundo plano que debe enviar una confirmacion al programa

!a traves de la variable "confirmacion".

!Una vez sea enviada, la tarea de segundo plano indica a esta que se ha realizado a través de la

!variable "confirmacion2".

!Este comportamiento es posible gracias al empleo de variables tipo PERS.

confirmacion := TRUE;

WaitTime(0.1);

WHILE confirmacion = TRUE DO

IF confirmacion2 = TRUE THEN

confirmacion:=FALSE;

TPWrite "Confirmacion completada";

WaitTime(0.5);

ENDIF

ENDWHILE

ENDPROC

ENDMODULE