

Filename:	Foxtrot.pro
Directory:	C:\Users\Håkon eikrem\Dropbox\Prosjekt IS\PROGRAM\Program versjon Foxtrot
Change date:	28.11.13 16:23:31 / V2.3
Title:	
Author:	
Version:	
Description:	

0001	FUNCTION_BLOCK CALCULATED_ACTUALS
0002	(*Uses step reading from encoders to calculate hight and angle on each motor, pitch and roll*)
0003	VAR_INPUT
0004	END_VAR
0005	VAR_OUTPUT
0006	a_hight1: INT;
0007	a_hight2: INT;
0008	a_hight3: INT;
0009	a_angle1: REAL;
0010	a_angle2: REAL;
0011	a_angle3: REAL;
0012	a_roll: REAL;
0013	a_pitch: REAL;
0014	
0015	END_VAR
0016	VAR
0017	st1: STEP_TO_HIGHT;
0018	st2: STEP_TO_HIGHT;
0019	st3: STEP_TO_HIGHT;
0020	a: REAL;
0021	b: REAL;
0022	c: REAL;
0023	d: REAL;
0024	END_VAR
0001	st1(step:= (DATA_IN_M1));
0002	st2(step:= (DATA_IN_M2));
0003	st3(step:= (DATA_IN_M3));
0004	
0005	a_hight1:=st1.h;
0006	a_hight2:=st2.h;
0007	a_hight3:=st3.h;
0008	a_angle1:=st1.a;
0009	a_angle2:=st2.a;
0010	a_angle3:=st3.a;
0011	
0012	a:=(a_hight3-a_hight2);
0013	b:=(a/l3);
0014	a_roll:=am*ASIN(b);
0015	c:=(a_hight2+a_hight3)/2;
0016	d:=(a_hight1-c)/l4;
0017	a_pitch:=am*ASIN(d);
HIGHT_TO_STEP (FB-ST)	
0001	FUNCTION_BLOCK HIGHT_TO_STEP
0002	VAR_INPUT
0003	length:INT;
0004	END_VAR
0005	VAR_OUTPUT
0006	angle:REAL;
0007	step:INT;
0008	END_VAR
0009	VAR
0010	b: REAL;
0011	a: REAL;
0012	c: REAL;
0013	h: INT;
0014	END_VAR
0001	h:=length;
0002	IF (h > 480) THEN
0003	h:=480;
0004	END_IF
0005	IF (h < 260) THEN
0006	h:=260;
0007	END_IF
0008	
0009	a:=(l1*l1)+(h*h)-(l2*l2);
0010	b:=(2*l1*h);
0011	angle:=-60*ACOS(a/b)+125;
0012	step:= REAL_TO_INT((14000/125)*angle+2000);
ROLL_PITCH_TO_STEP (FB-ST)	
0001	FUNCTION_BLOCK ROLL_PITCH_TO_STEP (*ROLL PITCH CALCULATOR*)
0002	VAR_INPUT
0003	roll_angle:REAL;
0004	pitch_angle:REAL;
0005	END_VAR
0006	VAR_OUTPUT

0007	step1: INT;
0008	step2: INT;
0009	step3: INT;
0010	END_VAR
0011	VAR
0012	a: REAL;
0013	b: REAL;
0014	mid_length1: REAL;
0015	c: REAL;
0016	HTS1:HIGHT_TO_STEP;
0017	HTS2:HIGHT_TO_STEP;
0018	HTS3:HIGHT_TO_STEP;
0019	set_p1:INT;
0020	set_p2:INT;
0021	set_p3:INT;
0022	END_VAR
0001	IF(roll_angle >= max_roll) THEN
0002	roll_angle:=max_roll;
0003	ELSIF(roll_angle <= min_roll) THEN
0004	roll_angle:=min_roll;
0005	END_IF;
0006	
0007	IF(pitch_angle >=max_pitch) THEN
0008	pitch_angle:= max_pitch;
0009	ELSIF(pitch_angle <= min_pitch) THEN
0010	pitch_angle:=min_pitch;
0011	END_IF
0012	
0013	b:=(SIN(pitch_angle/am));
0014	
0015	set_p1:= REAL_TO_INT(mid_length+b*I6);
0016	mid_length1:=mid_length-b*I5;
0017	
0018	a:=(SIN(roll_angle/am)*I3)/2;
0019	set_p2:= REAL_TO_INT(mid_length1-a);
0020	set_p3:= REAL_TO_INT(mid_length1+a);
0021	
0022	HTS1(length:=set_p1);
0023	HTS2(length:=set_p2);
0024	HTS3(length:=set_p3);
0025	step1:=HTS1.step;
0026	step2:=HTS2.step;
0027	step3:=HTS3.step;
STEP_TO_HIGHT (FB-ST)	
0001	FUNCTION_BLOCK STEP_TO_HIGHT
0002	VAR_INPUT
0003	(* Antall steg *)
0004	stepp: INT;
0005	
0006	END_VAR
0007	VAR_OUTPUT
0008	(* Høyde fra aksling motor til senter kule på plattform *)
0009	h: INT;
0010	a: REAL; (* VINKEL *)
0011	END_VAR
0012	VAR
0013	b: REAL;
0014	a1: REAL;
0015	END_VAR
0001	b:=INT_TO_REAL(stepp);
0002	a:=(b/(14000))*120-120;
0003	a1 := a/am;
0004	h:=REAL_TO_INT(I1*COS(a1)+SQRT(I2*I2-I1*I1*SIN(a1)*SIN(a1)));
INPUT_MODE (PRG-ST)	
0001	PROGRAM INPUT_MODE
0002	VAR
0003	
0004	(* Verdier hentet fra UDP stream *)
0005	Input_Roll AT %MW1: WORD;
0006	Input_Pitch AT %MW0: WORD;
0007	sim_roll: REAL;
0008	sim_pitch: REAL;
0009	INPUT_roll_real: REAL;
0010	INPUT_pitch_real: REAL;
0011	rpts: ROLL_PITCH_TO_STEP;
0012	

0013	pitch_real: INT;
0014	roll_real: INT;
0015	roll_old: REAL:=0;
0016	pitch_old: REAL:=0;
0017	roll_new: REAL;
0018	pitch_new: REAL;
0019	roll_new_int: INT;
0020	pitch_new_int: INT;
0021	END_VAR
0001	roll_new_int :=WORD_TO_INT(Input_Roll);
0002	pitch_new_int:=WORD_TO_INT(Input_Pitch);
0003	
0004	roll_new:=2*INT_TO_REAL(-roll_new_int) / 1000;
0005	pitch_new:=2*INT_TO_REAL(pitch_new_int) / 1000;
0006	
0007	INPUT_roll_real:=(roll_new*0.1+roll_old*0.9);
0008	INPUT_pitch_real:=(pitch_new*0.1+pitch_old*0.9);
0009	IF SIMULATION_UDP THEN
0010	INPUT_roll_real := sim_roll;
0011	INPUT_pitch_real := sim_pitch;
0012	END_IF
0013	rpts(roll_angle:=INPUT_ROLL_REAL,pitch_angle:=INPUT_PITCH_REAL);
0014	
0015	p1_step:=rpts.step1;
0016	p2_step:=rpts.step2;
0017	p3_step:=rpts.step3;
0018	
0019	roll_old:=INPUT_roll_real;
0020	pitch_old:=INPUT_pitch_real;
MANUAL (PRG-ST)	
0001	PROGRAM MANUAL
0002	VAR
0003	Speed : INT := 200;
0004	END_VAR
0001	(* Begrenser verdien til pX_step *)
0002	IF p1_step > 14000 THEN p1_step := 14000; END_IF
0003	IF p2_step > 14000 THEN p2_step := 14000; END_IF
0004	IF p3_step > 14000 THEN p3_step := 14000; END_IF
0005	
0006	IF p1_step < 2000 THEN p1_step := 2000; END_IF
0007	IF p2_step < 2000 THEN p2_step := 2000; END_IF
0008	IF p3_step < 2000 THEN p3_step := 2000; END_IF
0009	
0010	(* Motor1 opp *)
0011	IF JOGP1 THEN
0012	p1_step := p1_step+Speed;
0013	END_IF
0014	
0015	(* Motor1 ned *)
0016	IF JOGM1 THEN
0017	p1_step := p1_step-Speed;
0018	END_IF
0019	
0020	(* Motor2 opp *)
0021	IF JOGP2 THEN
0022	p2_step := p2_step+Speed;
0023	END_IF
0024	
0025	(* Motor2 ned *)
0026	IF JOGM2 THEN
0027	p2_step := p2_step-Speed;
0028	END_IF
0029	
0030	(* Motor3 opp *)
0031	IF JOGP3 THEN
0032	p3_step := p3_step+Speed;
0033	END_IF
0034	
0035	(* Motor3 Ned *)
0036	IF JOGM3 THEN
0037	p3_step := p3_step-Speed;
0038	END_IF
0039	
0040	(*Kjører alle motorene opp *)
0041	IF GO_UP THEN
0042	p1_step := p1_step+Speed;
0043	p2_step := p2_step+Speed;

0044	p3_step := p3_step+Speed;
0045	END_IF
0046	
0047	(*Kjører alle motorene ned *)
0048	IF GO_DOWN THEN
0049	p1_step := p1_step-Speed;
0050	p2_step := p2_step-Speed;
0051	p3_step := p3_step-Speed;
0052	END_IF
WAVE_V3 (PRG-ST)	
0001	PROGRAM WAVE_V3
0002	VAR
0003	X_M1 :REAL;
0004	X_M2: REAL := -1.5;
0005	X_M3: REAL := 1.5;
0006	top_M1: BOOL;
0007	top_M2: BOOL;
0008	top_M3: BOOL;
0009	Speed1 : REAL := 0.1;
0010	Speed2 : REAL := 0.1;
0011	Speed3 : REAL := 0.1;
0012	HTS1: HIGHT_TO_STEP;
0013	HTS2: HIGHT_TO_STEP;
0014	HTS3: HIGHT_TO_STEP;
0015	END_VAR
0001	(* SINUS WAVE M1*)
0002	IF (X_M1 <= 1.57) AND NOT(top_M1) THEN
0003	X_M1 := (X_M1 + Speed1);
0004	END_IF
0005	
0006	IF (X_M1 >= 1.57) THEN
0007	top_M1 := TRUE;
0008	END_IF
0009	
0010	IF top_M1 THEN
0011	X_M1 := (X_M1 - Speed1);
0012	END_IF
0013	
0014	IF (X_M1 <= -1.57) THEN
0015	top_M1 := FALSE;
0016	END_IF
0017	(*
0018	HTS1(length:= 370 + REAL_TO_INT(110 * SIN (X_M1)));*)
0019	
0020	HTS1(length:= REAL_TO_INT(350 + 120 * SIN (X_M1)));
0021	
0022	(* SINUS WAVE M2*)
0023	IF (X_M2 <= 1.57) AND NOT(top_M2) THEN
0024	X_M2 := (X_M2 + Speed2);
0025	END_IF
0026	
0027	IF (X_M2 >= 1.57) THEN
0028	top_M2 := TRUE;
0029	END_IF
0030	
0031	IF top_M2 THEN
0032	X_M2 := (X_M2 - Speed2);
0033	END_IF
0034	
0035	IF (X_M2 <= -1.57) THEN
0036	top_M2 := FALSE;
0037	END_IF
0038	
0039	HTS2 (length:= 370 + REAL_TO_INT(100 * SIN (X_M2)));
0040	
0041	
0042	(* SINUS WAVE M3*)
0043	
0044	IF (X_M3 <= 1.57) AND NOT(top_M3) THEN
0045	X_M3 := (X_M3 + Speed3);
0046	END_IF
0047	
0048	IF (X_M3 >= 1.57) THEN
0049	top_M3 := TRUE;
0050	END_IF
0051	
0052	IF top_M3 THEN

0053	X_M3 := (X_M3 - Speed3);
0054	END_IF
0055	
0056	IF (X_M3 <= -1.57) THEN
0057	top_M3 := FALSE;
0058	END_IF
0059	
0060	HTS3 (length:= 370 + REAL_TO_INT(100 * SIN (X_M3)));
0061	
0062	
0063	
0064	
0065	p1_step:= HTS1.step;
0066	p2_step:= HTS2.step;
0067	p3_step:= HTS3.step;
0068	M1_ANGLE:=HTS1.angle;
0069	M2_ANGLE:=HTS2.angle;
0070	M3_ANGLE:=HTS3.angle;
Read_Param (PRG-ST)	
0001	PROGRAM Read_Param
0002	VAR
0003	
0004	Management: WORD;
0005	Word0 AT %QW2400: WORD;
0006	Word1 AT %QW2401: WORD;
0007	Read_value0 AT %IW2400: WORD;
0008	Read_value1 AT %IW2401: WORD;
0009	Out_put0: WORD;
0010	Out_put1: WORD;
0011	Out: INT;
0012	Data_Out: INT;
0013	Read_par: BOOL;
0014	Par_nr:INT;
0015	END_VAR
0001	IF(Read_par) THEN
0002	Management := INT_TO_WORD(1 + 16 + 32 + 64);
0003	Word0 := ROR(Management , 8);
0004	
0005	Par_nr := INT_TO_WORD(8398);
0006	Word1 := ROR(Par_nr, 8);
0007	
0008	Out_put0 := ROL(Read_value0 , 8);
0009	Out_put1 := ROL(Read_value1 , 8);
0010	
0011	Data_Out := WORD_TO_INT( Read_value0 + Read_value1);
0012	
0013	ELSE
0014	Management := INT_TO_WORD(1 + 16 + 32);
0015	Word0 := ROR(Management , 8);
0016	END_IF
Read_Pos (PRG-ST)	
0001	PROGRAM Read_Pos
0002	VAR_INPUT
0003	END_VAR
0004	VAR
0005	(*Motor 1*)
0006	In_data_1 AT %IW2405: WORD;
0007	In_data_2 AT %IW2406: WORD;
0008	Data_in_m1_temp: WORD;
0009	(*Motor 2*)
0010	In_data_3 AT %IW2412: WORD;
0011	In_data_4 AT %IW2413: WORD;
0012	Data_in_m2_temp: WORD;
0013	(*Motor 3*)
0014	In_data_5 AT %IW2419: WORD;
0015	In_data_6 AT %IW2420: WORD;
0016	data_in_m3_temp: WORD;
0017	CA: CALCULATED_ACTUALS;
0018	roll_actual AT %MW3: WORD;
0019	pitch_actual AT %MW2: WORD;
0020	PA: REAL;
0021	PR: REAL;
0022	END_VAR
0001	(*Leser av posisjonsdata til motor 1*)
0002	Data_in_m1_temp:= ROL(In_data_1 + In_data_2, 8);
0003	DATA_IN_M1 := WORD_TO_INT(Data_in_m1_temp);

0004	
0005	(*Leser av posisjonsdata til motor 2*)
0006	Data_in_m2_temp := ROL(In_data_3 + In_data_4, 8);
0007	DATA_IN_M2 := WORD_TO_INT(Data_in_m2_temp);
0008	
0009	(*Leser av posisjonsdata til motor 3*)
0010	data_in_m3_temp := ROL(In_data_5 + In_data_6, 8);
0011	DATA_IN_M3 := WORD_TO_INT(Data_in_m3_temp);
0012	ca();
0013	
0014	PA := (CA.a_pitch);
0015	
0016	PR := (CA.a_roll);
0017	
0018	roll_actual := REAL_TO_INT(PR*1000);
0019	pitch_actual := REAL_TO_INT(PA*1000);
M1 (FUN-ST)	
0001	FUNCTION M1 : BOOL
0002	VAR_INPUT
0003	SPEED_DIR:INT;
0004	END_VAR
0005	VAR
0006	P0_1_M1 AT %QW2404: WORD;
0007	P0_2_M1 AT %QW2405: WORD;
0008	END_VAR
0001	IF (ENABLE) THEN
0002	P0_1_M1 := ROR(INT_TO_WORD( 2 + 4),8);    (*Enable signal til M1 sent på ORD 1*)
0003	P0_2_M1 := ROR(INT_TO_WORD(SPEED_DIR), 8); (* Setter hastigheten til M1 på ord 2 *)
0004	ELSE
0005	M1_Cont_inhibit := TRUE;
0006	END_IF
M2 (FUN-ST)	
0001	FUNCTION M2 : BOOL
0002	VAR_INPUT
0003	SPEED_DIR:INT;
0004	END_VAR
0005	VAR
0006	P0_1_M2 AT %QW2411: WORD;
0007	P0_2_M2 AT %QW2412: WORD;
0008	END_VAR
0001	IF (ENABLE) THEN
0002	P0_1_M2 := ROR(INT_TO_WORD( 2 + 4), 8);    (*Enable signal til M2 sent på ORD 1*)
0003	P0_2_M2 := ROR(INT_TO_WORD(SPEED_DIR), 8); (* Setter hastigheten til M2 på ord 2 *)
0004	ELSE
0005	M2_Cont_inhibit := TRUE;
0006	END_IF
M3 (FUN-ST)	
0001	FUNCTION M3 : BOOL
0002	VAR_INPUT
0003	SPEED_DIR:INT;
0004	END_VAR
0005	VAR
0006	P0_1_M3 AT %QW2418: WORD;
0007	P0_2_M3 AT %QW2419: WORD;
0008	END_VAR
0001	IF (ENABLE) THEN
0002	P0_1_M3 := ROR(INT_TO_WORD( 2 + 4), 8);    (*Enable signal til M3 sent på ord 1*)
0003	P0_2_M3 := ROR(INT_TO_WORD(Speed_DIR), 8); (* Setter hastigheten til M3 på ord 2 *)
0004	ELSE
0005	M3_Cont_inhibit := TRUE;
0006	END_IF
PLC_PRG (PRG-ST)	
0001	PROGRAM PLC_PRG
0002	VAR
0003	RPTS: ROLL_PITCH_TO_STEP;
0004	Read_par: BOOL;
0005	
0006	END_VAR
0001	(* PROGRAM VELGER. *)
0002	
0003	IF(wave_run) AND NOT(input_run) AND NOT(manual_run) THEN
0004	WAVE_V3();
0005	END_IF;
0006	
0007	IF(input_run) AND NOT(wave_run) AND NOT(manual_run) THEN

0008	INPUT_MODE();
0009	END_IF;
0010	
0011	IF(manual_run) AND NOT(input_run) AND NOT(wave_run) THEN
0012	MANUAL();
0013	END_IF;
0014	
0015	(*Setter Plattformen i 0 pos.*)
0016	IF( NOT wave_run AND NOT input_run AND NOT manual_run) THEN
0017	RPTS(roll_angle:=0,pitch_angle:=0);
0018	p1_step:=RPTS.step1;
0019	p2_step:=RPTS.step2;
0020	p3_step:=RPTS.step3;
0021	END_IF;
0022	
SET_STEP (PRG-ST)	
0001	PROGRAM SET_STEP
0002	VAR
0003	SET_M1:DINT;
0004	STEP_M1: INT;
0005	SET_M2:DINT;
0006	STEP_M2: INT;
0007	SET_M3:DINT;
0008	STEP_M3: INT;
0009	END_VAR
0001	(* P1-3_STEP globale variabler for ønsket Steg *)
0002	(* DATA_IN_M1-M3 steg verdier avlest fra profibus *)
0003	(*M1*)
0004	IF(DATA_IN_M1 > P1_STEP) THEN
0005	(*ned*)
0006	STEP_M1:=REAL_TO_INT((P1_STEP-DATA_IN_M1)/3);
0007	IF(STEP_M1 < -2000) THEN
0008	STEP_M1:= -2000;
0009	END_IF;
0010	(*OPP*)
0011	ELSIF(DATA_IN_M1 < P1_STEP) THEN
0012	STEP_M1:=REAL_TO_INT((P1_STEP - DATA_IN_M1)/3);
0013	IF(STEP_M1 > 2000) THEN
0014	STEP_M1:=2000;
0015	END_IF;
0016	ELSE
0017	STEP_M1:=0;
0018	END_IF;
0019	M1(SPEED_DIR:=STEP_M1);
0020	
0021	
0022	(*M2*)
0023	IF(DATA_IN_M2 > P2_STEP) THEN
0024	(*NED*)
0025	STEP_M2:=REAL_TO_INT((P2_STEP-DATA_IN_M2)/3);
0026	IF(STEP_M2 < -2000) THEN
0027	STEP_M2:= -2000;
0028	END_IF;
0029	(*OPP*)
0030	ELSIF(DATA_IN_M2 < P2_STEP) THEN
0031	STEP_M2:=REAL_TO_INT((P2_STEP - DATA_IN_M2)/3);
0032	IF(STEP_M2 > 2000) THEN
0033	STEP_M2:=2000;
0034	END_IF;
0035	ELSE
0036	STEP_M2:=0;
0037	END_IF;
0038	M2(SPEED_DIR:=STEP_M2);
0039	
0040	(*M3*)
0041	IF(DATA_IN_M3 > P3_STEP) THEN
0042	STEP_M3:=REAL_TO_INT((P3_STEP-DATA_IN_M3)/3);
0043	IF(STEP_M3 < -2000) THEN
0044	STEP_M3:= -2000;
0045	END_IF;
0046	ELSIF(DATA_IN_M3 < P3_STEP) THEN
0047	STEP_M3:=REAL_TO_INT((P3_STEP - DATA_IN_M3)/3);
0048	IF(STEP_M3 < -2000) THEN
0049	STEP_M3:=-2000;
0050	END_IF;
0051	ELSE
0052	STEP_M3:=0;



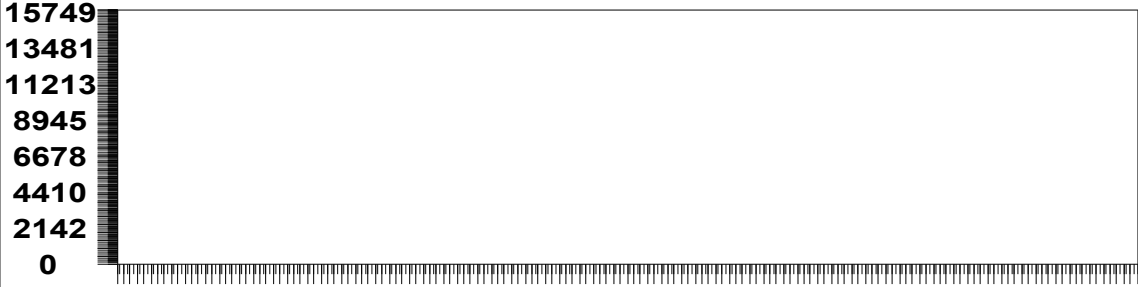
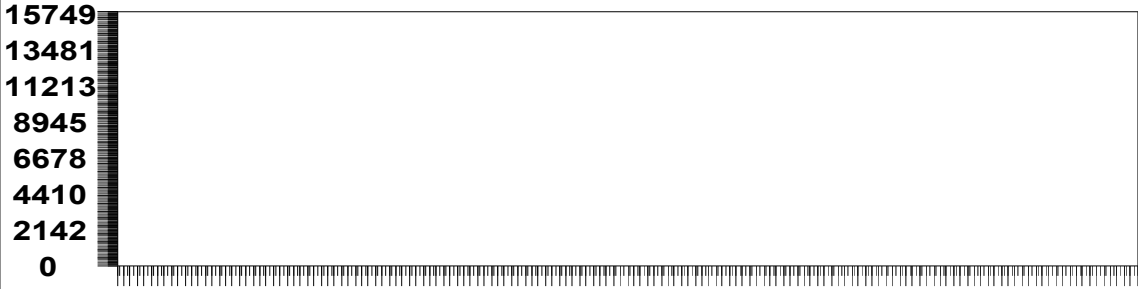
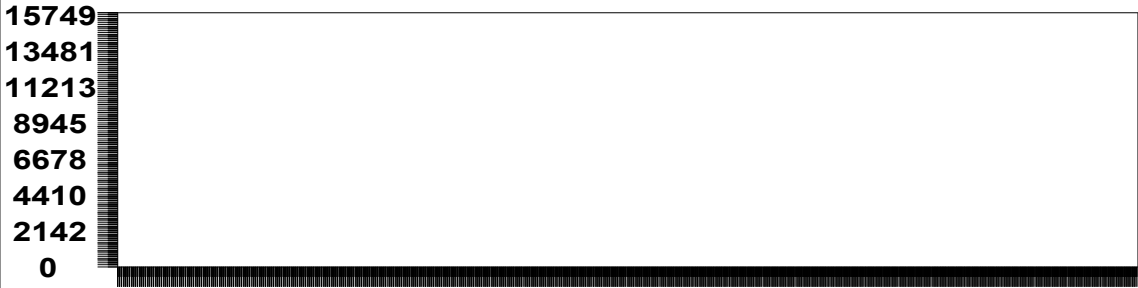
0053

END\_IF;

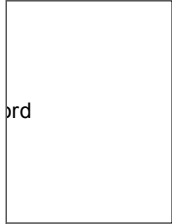
0054

M3(SPEED\_DIR:=STEP\_M3);

Graph



recc



HOME

MODE SELECT

Manual mode

Wave mode

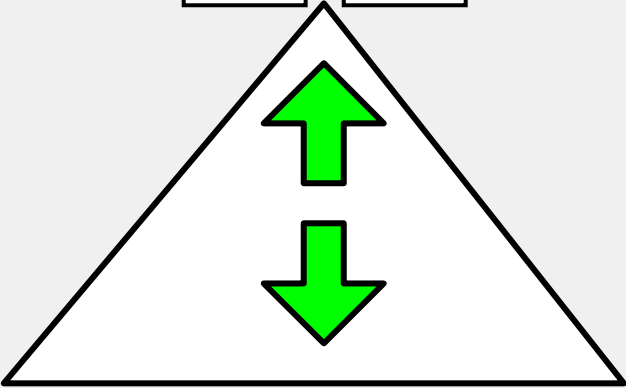
INPUT MODE

READ

Manual\_mode

POS: %i

M1 -M1 +



M3 -M3 +M2 -M2 +

POS: %iPOS: %i

BACKENABLESTART MANUAL

SET POSITION

Speed	M1	M2	M3
<div><div></div><div></div><div></div></div>	<div><div></div><div></div><div></div></div>	<div><div></div><div></div><div></div></div>	<div><div></div><div></div><div></div></div>
%i	<div></div>	<div></div>	<div></div>

UDP

START

ENABLE

SIMULATION

Pitch

Roll

%s

%s

BACK

Wave\_mode

BACK

WAVE MODE

START

ENABLE

wave m1:%s

%s

wave m2:%s

%s

wave m3:%s

%s

Global\_Variables

0001	VAR_GLOBAL
0002	(*Manual mode variabler*)
0003	STOP: BOOL;
0004	JOGM1: BOOL;
0005	JOGP1: BOOL;
0006	JOGM2: BOOL;
0007	JOGP2: BOOL;
0008	JOGM3: BOOL;
0009	JOGP3: BOOL;
0010	GO_UP: BOOL;
0011	GO_DOWN: BOOL;
0012	GO_HOME: BOOL;

0013	Manual_speed: INT:=300;
0014	
0015	(*Motor 1*)
0016	M1_Cont_inhibit AT%X2404.8 : BOOL;
0017	DATA_IN_M1: INT;
0018	Y_M1: INT;
0019	M1_ANGLE: REAL;
0020	
0021	(*Motor 2*)
0022	M2_Cont_inhibit AT%X2411.8 : BOOL;
0023	DATA_IN_M2: INT;
0024	Y_M2: INT;
0025	M2_ANGLE:REAL;
0026	
0027	(*Motor 3*)
0028	M3_Cont_inhibit AT%X2418.8 : BOOL;
0029	DATA_IN_M3: INT;
0030	Y_M3: INT;
0031	M3_ANGLE:REAL;
0032	
0033	
0034	
0035	
0036	(*MOTOR 1 Set Points*)
0037	p1_step: INT;
0038	(*MOTOR 2 Set Points*)
0039	p2_step: INT;
0040	(*MOTOR 3 Set points*)
0041	p3_step: INT;
0042	
0043	
0044	(*Platform angle Set Points*)
0045	pitch:REAL;
0046	roll:REAL;
0047	
0048	
0049	(*MODES*)
0050	ENABLE: BOOL; (*Enables the Drives*)
0051	manual_run: BOOL; (* Set specified position *)
0052	wave_run: BOOL; (* Wave PROGRAM *)
0053	input_run: BOOL; (* Roll / Pitch input from other sources *)
0054	
0055	SIMULATION_UDP: BOOL;
0056	
0057	LOGG :BOOL;
0058	
0059	END_VAR
0060	
0061	VAR_GLOBAL CONSTANT
0062	
0063	l1:INT:=150; (*lower arm in mm*)
0064	l2:INT:=330; (*upper arm in mm*)
0065	l3:INT:=1075; (*length between joints p1, p2, p3*)
0066	l4:INT:=931; (*length of platform center from p1*)
0067	l5:INT := 311;
0068	l6:INT := 620;
0069	mid_length:INT:=370; (*mid length of both arms combined*)
0070	min_angle:INT:= -125;
0071	max_angle:INT:= 0;
0072	max_roll:INT:=10;
0073	min_roll:INT:=-10;
0074	max_pitch:INT:=10;
0075	min_pitch: INT:=-10;
0076	am:REAL:=57.2958; (*arcminute constant*)
0077	
0078	END_VAR
Variable_Configuration	
0001	VAR_CONFIG
0002	END_VAR
Globale_Variablen	
0001	VAR_GLOBAL
0002	END_VAR
CAM Data	

0001	(* Automatic generated CAM-Data *)
0002	VAR_GLOBAL
0003	END_VAR

CNC Data

0001	(* Automatic generated CNC-Data *)
0002	VAR_GLOBAL
0003	END_VAR

Drive Configuration Data

0001	(* Automatic generated Drive-Data *)
0002	VAR_GLOBAL
0003	END_VAR

Global Variables 0

0001	VAR_GLOBAL
0002	END_VAR

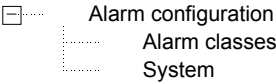
Globale\_Variablen

0001	VAR_GLOBAL
0002	END_VAR

Version

0001	VAR_GLOBAL CONSTANT
0002	rVersion:REAL:=1.001;
0003	END_VAR

Alarm configuration



PLC Configuration

\*PLC Configuration (Id.: 8765)

Node number: -1  
Input address: %IB0  
Output address: %QB0  
Diagnostic address: %MB0  
Download: 1  
AutoAdr: 1

\*K-Bus (\* WAGO-I/O-IPC Profibus DP Master, ext \*) [FIX] (Id.: 11994)

Node number: 0  
Input address: %IB0  
Output address: %QB0  
Diagnostic address: %MB0  
Download: 1  
AutoAdr: 1

\*0750-0631/0000-0004 Inkremental Encoder Interface (RS422 Input)[VAR] (Id.: 2000051001)

Node number: 0  
Input address: %IB0  
Output address: %QB0  
Diagnostic address: %MB0  
Download: 1  
AutoAdr: 1

Channels:

- AT %IX0.0: BOOL; (\* Latched Data Set \*) [CHANNEL (I)]
- AT %IX0.1: BOOL; (\* External Latch Ack. \*) [CHANNEL (I)]
- AT %IX0.2: BOOL; (\* Counter Set Acknowledge \*) [CHANNEL (I)]
- AT %IX0.3: BOOL; (\* Counter Underflow \*) [CHANNEL (I)]
- AT %IX0.4: BOOL; (\* Counter Overflow \*) [CHANNEL (I)]
- AT %IX0.5: BOOL; (\* External Error \*) [CHANNEL (I)]
- AT %IB2: BYTE; (\* Counter value Low Byte \*) [CHANNEL (I)]
- AT %IB3: BYTE; (\* Counter value High Byte \*) [CHANNEL (I)]
- AT %IX2.0: BOOL; (\* Signal Input Gate \*) [CHANNEL (I)]
- AT %IX2.1: BOOL; (\* Signal Input Latch \*) [CHANNEL (I)]
- AT %IX2.2: BOOL; (\* Signal Input external error \*) [CHANNEL (I)]
- AT %IX2.3: BOOL; (\* Signal Input C./C \*) [CHANNEL (I)]
- AT %IX2.4: BOOL; (\* Signal Input B./B \*) [CHANNEL (I)]
- AT %IX2.5: BOOL; (\* Signal Input A./A \*) [CHANNEL (I)]
- AT %IW3: WORD; (\* Latch value \*) [CHANNEL (I)]
- AT %QX0.0: BOOL; (\* Release Index Pulse \*) [CHANNEL (Q)]
- AT %QX0.1: BOOL; (\* Release Latch \*) [CHANNEL (Q)]

AT %QX0.2: BOOL; (\* Counter Set \*) [CHANNEL (Q)]  
AT %QB2: BYTE; (\* Set value Low Byte \*) [CHANNEL (Q)]  
AT %QB3: BYTE; (\* Set value High Byte \*) [CHANNEL (Q)]

\*Internal Digital I/O[FIX] (Id.: 9)

Node number: 1  
Input address: %IB512  
Output address: %QB512  
Diagnostic address: %MB0  
Download: 1  
AutoAdr: 1

\*Internal Digital Inputs[FIX] (Id.: 10)

Node number: 0  
Input address: %IB512  
Output address: %QB512  
Diagnostic address: %MB4  
Download: 1  
AutoAdr: 1  
Channels:  
DIN1 AT %IX2300.0: BOOL; (\* Internal Digital Input Bit 1 \*) [CHANNEL (I)]  
DIN2 AT %IX2300.1: BOOL; (\* Internal Digital Input Bit 2 \*) [CHANNEL (I)]

\*Internal Digital Outputs[FIX] (Id.: 11)

Node number: 1  
Input address: %IB513  
Output address: %QB512  
Diagnostic address: %MB8  
Download: 1  
AutoAdr: 1  
Channels:  
DOUT1 AT %QX2300.0: BOOL; (\* Internal Digital Output Bit 1 \*) [CHANNEL (Q)]  
DOUT2 AT %QX2300.1: BOOL; (\* Internal Digital Output Bit 2 \*) [CHANNEL (Q)]

\*Fieldbus variables[FIX] (Id.: 2010310001)

Node number: 3  
Input address: %IB0  
Output address: %QB0  
Diagnostic address: %MB0  
Download: 1  
AutoAdr: 1

\*Flag variables[FIX] (Id.: 2010320001)

Node number: 4  
Input address: %IB0  
Output address: %QB0  
Diagnostic address: %MB0  
Download: 1  
AutoAdr: 1

\*WAGO-FB-DPM[VAR] (Id.: 983083)

Node number: 2  
Input address: %IB4800  
Output address: %QB4800  
Diagnostic address: %MB8000  
Download: 1  
AutoAdr: 1  
Parameter:  
UpdateTask: All  
Bus Mode: Asynchron  
Device dependent data:  
GSD-File: 08A6\_V10.GSD  
Station number: 1  
Highest station number: 4  
Baud rate: 12000,00  
Auto Clear: 0  
Automatic Start: 1  
Slot Time (TSL): 1000 tBit  
Min.Station Delay (min TSDR): 11 tBit  
Max.Station Delay (max TSDR): 800 tBit  
Quiet Time (TQUI): 9 tBit  
Setup Time (TSET): 16 tBit  
Target Rotation Time (TTR): 6647 tBit  
Gap Update Factor: 10  
Max. Retry Limit: 4  
Min. Slave Interval: 1 100 µs  
Poll Timeout: 10 10 ms  
Data Control Time: 1200 ms

Watchdog Time (TWD): 1000 ms

Groups:

Gr 1: Freeze:1, Sync: 1  
Gr 2: Freeze:1, Sync: 1  
Gr 3: Freeze:1, Sync: 1  
Gr 4: Freeze:1, Sync: 1  
Gr 5: Freeze:1, Sync: 1  
Gr 6: Freeze:1, Sync: 1  
Gr 7: Freeze:1, Sync: 1  
Gr 8: Freeze:1, Sync: 1

\*MOVIDRIVE DFP21/MCH41[VAR] (Id.: 69230394)

Node number: 0

Input address: %IB4800

Output address: %QB4800

Diagnostic address: %MB8004

Download: 1

AutoAdr: 1

Device dependent data:

GSD-File: SEW\_6003.gsd

Station number: 2

Station active: 1

TSDR: 11

Watchdog: 1

Watchdog Time: 1000

Fail Save: 1

Freeze:0

Sync: 0

Lock/Unlock: 2

Group settings: 0

Ext\_User\_Prm\_Data:

"External Diagnosis (only DP)": Disabled (off)

Selected modules:

Param + 3 PD (4+3 words), 0xF3, 0xF2 (++)

Param + 3 PD (4+3 words)

AT %IW2400: WORD;

AT %IX2400.0: BOOL; (\* Bit 0 \*)  
AT %IX2400.1: BOOL; (\* Bit 1 \*)  
AT %IX2400.2: BOOL; (\* Bit 2 \*)  
AT %IX2400.3: BOOL; (\* Bit 3 \*)  
AT %IX2400.4: BOOL; (\* Bit 4 \*)  
AT %IX2400.5: BOOL; (\* Bit 5 \*)  
AT %IX2400.6: BOOL; (\* Bit 6 \*)  
AT %IX2400.7: BOOL; (\* Bit 7 \*)  
AT %IX2400.8: BOOL; (\* Bit 8 \*)  
AT %IX2400.9: BOOL; (\* Bit 9 \*)  
AT %IX2400.10: BOOL; (\* Bit 10 \*)  
AT %IX2400.11: BOOL; (\* Bit 11 \*)  
AT %IX2400.12: BOOL; (\* Bit 12 \*)  
AT %IX2400.13: BOOL; (\* Bit 13 \*)  
AT %IX2400.14: BOOL; (\* Bit 14 \*)  
AT %IX2400.15: BOOL; (\* Bit 15 \*)

AT %IW2401: WORD;

AT %IX2401.0: BOOL; (\* Bit 0 \*)  
AT %IX2401.1: BOOL; (\* Bit 1 \*)  
AT %IX2401.2: BOOL; (\* Bit 2 \*)  
AT %IX2401.3: BOOL; (\* Bit 3 \*)  
AT %IX2401.4: BOOL; (\* Bit 4 \*)  
AT %IX2401.5: BOOL; (\* Bit 5 \*)  
AT %IX2401.6: BOOL; (\* Bit 6 \*)  
AT %IX2401.7: BOOL; (\* Bit 7 \*)  
AT %IX2401.8: BOOL; (\* Bit 8 \*)  
AT %IX2401.9: BOOL; (\* Bit 9 \*)  
AT %IX2401.10: BOOL; (\* Bit 10 \*)  
AT %IX2401.11: BOOL; (\* Bit 11 \*)  
AT %IX2401.12: BOOL; (\* Bit 12 \*)  
AT %IX2401.13: BOOL; (\* Bit 13 \*)  
AT %IX2401.14: BOOL; (\* Bit 14 \*)  
AT %IX2401.15: BOOL; (\* Bit 15 \*)

AT %IW2402: WORD;

AT %IX2402.0: BOOL; (\* Bit 0 \*)  
AT %IX2402.1: BOOL; (\* Bit 1 \*)  
AT %IX2402.2: BOOL; (\* Bit 2 \*)  
AT %IX2402.3: BOOL; (\* Bit 3 \*)  
AT %IX2402.4: BOOL; (\* Bit 4 \*)  
AT %IX2402.5: BOOL; (\* Bit 5 \*)  
AT %IX2402.6: BOOL; (\* Bit 6 \*)



AT %IX2402.7: BOOL; (\* Bit 7 \*)  
AT %IX2402.8: BOOL; (\* Bit 8 \*)  
AT %IX2402.9: BOOL; (\* Bit 9 \*)  
AT %IX2402.10: BOOL; (\* Bit 10 \*)  
AT %IX2402.11: BOOL; (\* Bit 11 \*)  
AT %IX2402.12: BOOL; (\* Bit 12 \*)  
AT %IX2402.13: BOOL; (\* Bit 13 \*)  
AT %IX2402.14: BOOL; (\* Bit 14 \*)  
AT %IX2402.15: BOOL; (\* Bit 15 \*)  
AT %IW2403: WORD;  
AT %IX2403.0: BOOL; (\* Bit 0 \*)  
AT %IX2403.1: BOOL; (\* Bit 1 \*)  
AT %IX2403.2: BOOL; (\* Bit 2 \*)  
AT %IX2403.3: BOOL; (\* Bit 3 \*)  
AT %IX2403.4: BOOL; (\* Bit 4 \*)  
AT %IX2403.5: BOOL; (\* Bit 5 \*)  
AT %IX2403.6: BOOL; (\* Bit 6 \*)  
AT %IX2403.7: BOOL; (\* Bit 7 \*)  
AT %IX2403.8: BOOL; (\* Bit 8 \*)  
AT %IX2403.9: BOOL; (\* Bit 9 \*)  
AT %IX2403.10: BOOL; (\* Bit 10 \*)  
AT %IX2403.11: BOOL; (\* Bit 11 \*)  
AT %IX2403.12: BOOL; (\* Bit 12 \*)  
AT %IX2403.13: BOOL; (\* Bit 13 \*)  
AT %IX2403.14: BOOL; (\* Bit 14 \*)  
AT %IX2403.15: BOOL; (\* Bit 15 \*)  
AT %QW2400: WORD;  
AT %QX2400.0: BOOL; (\* Bit 0 \*)  
AT %QX2400.1: BOOL; (\* Bit 1 \*)  
AT %QX2400.2: BOOL; (\* Bit 2 \*)  
AT %QX2400.3: BOOL; (\* Bit 3 \*)  
AT %QX2400.4: BOOL; (\* Bit 4 \*)  
AT %QX2400.5: BOOL; (\* Bit 5 \*)  
AT %QX2400.6: BOOL; (\* Bit 6 \*)  
AT %QX2400.7: BOOL; (\* Bit 7 \*)  
AT %QX2400.8: BOOL; (\* Bit 8 \*)  
AT %QX2400.9: BOOL; (\* Bit 9 \*)  
AT %QX2400.10: BOOL; (\* Bit 10 \*)  
AT %QX2400.11: BOOL; (\* Bit 11 \*)  
AT %QX2400.12: BOOL; (\* Bit 12 \*)  
AT %QX2400.13: BOOL; (\* Bit 13 \*)  
AT %QX2400.14: BOOL; (\* Bit 14 \*)  
AT %QX2400.15: BOOL; (\* Bit 15 \*)  
AT %QW2401: WORD;  
AT %QX2401.0: BOOL; (\* Bit 0 \*)  
AT %QX2401.1: BOOL; (\* Bit 1 \*)  
AT %QX2401.2: BOOL; (\* Bit 2 \*)  
AT %QX2401.3: BOOL; (\* Bit 3 \*)  
AT %QX2401.4: BOOL; (\* Bit 4 \*)  
AT %QX2401.5: BOOL; (\* Bit 5 \*)  
AT %QX2401.6: BOOL; (\* Bit 6 \*)  
AT %QX2401.7: BOOL; (\* Bit 7 \*)  
AT %QX2401.8: BOOL; (\* Bit 8 \*)  
AT %QX2401.9: BOOL; (\* Bit 9 \*)  
AT %QX2401.10: BOOL; (\* Bit 10 \*)  
AT %QX2401.11: BOOL; (\* Bit 11 \*)  
AT %QX2401.12: BOOL; (\* Bit 12 \*)  
AT %QX2401.13: BOOL; (\* Bit 13 \*)  
AT %QX2401.14: BOOL; (\* Bit 14 \*)  
AT %QX2401.15: BOOL; (\* Bit 15 \*)  
AT %QW2402: WORD;  
AT %QX2402.0: BOOL; (\* Bit 0 \*)  
AT %QX2402.1: BOOL; (\* Bit 1 \*)  
AT %QX2402.2: BOOL; (\* Bit 2 \*)  
AT %QX2402.3: BOOL; (\* Bit 3 \*)  
AT %QX2402.4: BOOL; (\* Bit 4 \*)  
AT %QX2402.5: BOOL; (\* Bit 5 \*)  
AT %QX2402.6: BOOL; (\* Bit 6 \*)  
AT %QX2402.7: BOOL; (\* Bit 7 \*)  
AT %QX2402.8: BOOL; (\* Bit 8 \*)  
AT %QX2402.9: BOOL; (\* Bit 9 \*)  
AT %QX2402.10: BOOL; (\* Bit 10 \*)  
AT %QX2402.11: BOOL; (\* Bit 11 \*)  
AT %QX2402.12: BOOL; (\* Bit 12 \*)  
AT %QX2402.13: BOOL; (\* Bit 13 \*)  
AT %QX2402.14: BOOL; (\* Bit 14 \*)  
AT %QX2402.15: BOOL; (\* Bit 15 \*)  
AT %QW2403: WORD;

AT %QX2403.0: BOOL; (\* Bit 0 \*)  
AT %QX2403.1: BOOL; (\* Bit 1 \*)  
AT %QX2403.2: BOOL; (\* Bit 2 \*)  
AT %QX2403.3: BOOL; (\* Bit 3 \*)  
AT %QX2403.4: BOOL; (\* Bit 4 \*)  
AT %QX2403.5: BOOL; (\* Bit 5 \*)  
AT %QX2403.6: BOOL; (\* Bit 6 \*)  
AT %QX2403.7: BOOL; (\* Bit 7 \*)  
AT %QX2403.8: BOOL; (\* Bit 8 \*)  
AT %QX2403.9: BOOL; (\* Bit 9 \*)  
AT %QX2403.10: BOOL; (\* Bit 10 \*)  
AT %QX2403.11: BOOL; (\* Bit 11 \*)  
AT %QX2403.12: BOOL; (\* Bit 12 \*)  
AT %QX2403.13: BOOL; (\* Bit 13 \*)  
AT %QX2403.14: BOOL; (\* Bit 14 \*)  
AT %QX2403.15: BOOL; (\* Bit 15 \*)  
AT %IW2404: WORD;  
AT %IX2404.0: BOOL; (\* Bit 0 \*)  
AT %IX2404.1: BOOL; (\* Bit 1 \*)  
AT %IX2404.2: BOOL; (\* Bit 2 \*)  
AT %IX2404.3: BOOL; (\* Bit 3 \*)  
AT %IX2404.4: BOOL; (\* Bit 4 \*)  
AT %IX2404.5: BOOL; (\* Bit 5 \*)  
AT %IX2404.6: BOOL; (\* Bit 6 \*)  
AT %IX2404.7: BOOL; (\* Bit 7 \*)  
AT %IX2404.8: BOOL; (\* Bit 8 \*)  
AT %IX2404.9: BOOL; (\* Bit 9 \*)  
AT %IX2404.10: BOOL; (\* Bit 10 \*)  
AT %IX2404.11: BOOL; (\* Bit 11 \*)  
AT %IX2404.12: BOOL; (\* Bit 12 \*)  
AT %IX2404.13: BOOL; (\* Bit 13 \*)  
AT %IX2404.14: BOOL; (\* Bit 14 \*)  
AT %IX2404.15: BOOL; (\* Bit 15 \*)  
AT %IW2405: WORD;  
AT %IX2405.0: BOOL; (\* Bit 0 \*)  
AT %IX2405.1: BOOL; (\* Bit 1 \*)  
AT %IX2405.2: BOOL; (\* Bit 2 \*)  
AT %IX2405.3: BOOL; (\* Bit 3 \*)  
AT %IX2405.4: BOOL; (\* Bit 4 \*)  
AT %IX2405.5: BOOL; (\* Bit 5 \*)  
AT %IX2405.6: BOOL; (\* Bit 6 \*)  
AT %IX2405.7: BOOL; (\* Bit 7 \*)  
AT %IX2405.8: BOOL; (\* Bit 8 \*)  
AT %IX2405.9: BOOL; (\* Bit 9 \*)  
AT %IX2405.10: BOOL; (\* Bit 10 \*)  
AT %IX2405.11: BOOL; (\* Bit 11 \*)  
AT %IX2405.12: BOOL; (\* Bit 12 \*)  
AT %IX2405.13: BOOL; (\* Bit 13 \*)  
AT %IX2405.14: BOOL; (\* Bit 14 \*)  
AT %IX2405.15: BOOL; (\* Bit 15 \*)  
AT %IW2406: WORD;  
AT %IX2406.0: BOOL; (\* Bit 0 \*)  
AT %IX2406.1: BOOL; (\* Bit 1 \*)  
AT %IX2406.2: BOOL; (\* Bit 2 \*)  
AT %IX2406.3: BOOL; (\* Bit 3 \*)  
AT %IX2406.4: BOOL; (\* Bit 4 \*)  
AT %IX2406.5: BOOL; (\* Bit 5 \*)  
AT %IX2406.6: BOOL; (\* Bit 6 \*)  
AT %IX2406.7: BOOL; (\* Bit 7 \*)  
AT %IX2406.8: BOOL; (\* Bit 8 \*)  
AT %IX2406.9: BOOL; (\* Bit 9 \*)  
AT %IX2406.10: BOOL; (\* Bit 10 \*)  
AT %IX2406.11: BOOL; (\* Bit 11 \*)  
AT %IX2406.12: BOOL; (\* Bit 12 \*)  
AT %IX2406.13: BOOL; (\* Bit 13 \*)  
AT %IX2406.14: BOOL; (\* Bit 14 \*)  
AT %IX2406.15: BOOL; (\* Bit 15 \*)  
AT %QW2404: WORD;  
AT %QX2404.0: BOOL; (\* Bit 0 \*)  
AT %QX2404.1: BOOL; (\* Bit 1 \*)  
AT %QX2404.2: BOOL; (\* Bit 2 \*)  
AT %QX2404.3: BOOL; (\* Bit 3 \*)  
AT %QX2404.4: BOOL; (\* Bit 4 \*)  
AT %QX2404.5: BOOL; (\* Bit 5 \*)  
AT %QX2404.6: BOOL; (\* Bit 6 \*)  
AT %QX2404.7: BOOL; (\* Bit 7 \*)  
AT %QX2404.8: BOOL; (\* Bit 8 \*)  
AT %QX2404.9: BOOL; (\* Bit 9 \*)

AT %QX2404.10: BOOL; (\* Bit 10 \*)  
AT %QX2404.11: BOOL; (\* Bit 11 \*)  
AT %QX2404.12: BOOL; (\* Bit 12 \*)  
AT %QX2404.13: BOOL; (\* Bit 13 \*)  
AT %QX2404.14: BOOL; (\* Bit 14 \*)  
AT %QX2404.15: BOOL; (\* Bit 15 \*)  
AT %QW2405: WORD;  
AT %QX2405.0: BOOL; (\* Bit 0 \*)  
AT %QX2405.1: BOOL; (\* Bit 1 \*)  
AT %QX2405.2: BOOL; (\* Bit 2 \*)  
AT %QX2405.3: BOOL; (\* Bit 3 \*)  
AT %QX2405.4: BOOL; (\* Bit 4 \*)  
AT %QX2405.5: BOOL; (\* Bit 5 \*)  
AT %QX2405.6: BOOL; (\* Bit 6 \*)  
AT %QX2405.7: BOOL; (\* Bit 7 \*)  
AT %QX2405.8: BOOL; (\* Bit 8 \*)  
AT %QX2405.9: BOOL; (\* Bit 9 \*)  
AT %QX2405.10: BOOL; (\* Bit 10 \*)  
AT %QX2405.11: BOOL; (\* Bit 11 \*)  
AT %QX2405.12: BOOL; (\* Bit 12 \*)  
AT %QX2405.13: BOOL; (\* Bit 13 \*)  
AT %QX2405.14: BOOL; (\* Bit 14 \*)  
AT %QX2405.15: BOOL; (\* Bit 15 \*)  
AT %QW2406: WORD;  
AT %QX2406.0: BOOL; (\* Bit 0 \*)  
AT %QX2406.1: BOOL; (\* Bit 1 \*)  
AT %QX2406.2: BOOL; (\* Bit 2 \*)  
AT %QX2406.3: BOOL; (\* Bit 3 \*)  
AT %QX2406.4: BOOL; (\* Bit 4 \*)  
AT %QX2406.5: BOOL; (\* Bit 5 \*)  
AT %QX2406.6: BOOL; (\* Bit 6 \*)  
AT %QX2406.7: BOOL; (\* Bit 7 \*)  
AT %QX2406.8: BOOL; (\* Bit 8 \*)  
AT %QX2406.9: BOOL; (\* Bit 9 \*)  
AT %QX2406.10: BOOL; (\* Bit 10 \*)  
AT %QX2406.11: BOOL; (\* Bit 11 \*)  
AT %QX2406.12: BOOL; (\* Bit 12 \*)  
AT %QX2406.13: BOOL; (\* Bit 13 \*)  
AT %QX2406.14: BOOL; (\* Bit 14 \*)  
AT %QX2406.15: BOOL; (\* Bit 15 \*)

\*MOVIDRIVE DFP21/MCH41[VAR] (Id.: 69230394)

Node number: 1

Input address: %IB4814

Output address: %QB4814

Diagnostic address: %MB8028

Download: 1

AutoAdr: 1

Device dependent data:

GSD-File: SEW\_6003.gsd

Station number: 3

Station active: 1

TSDR: 11

Watchdog: 1

Watchdog Time: 1000

Fail Save: 1

Freeze:0

Sync: 0

Lock/Unlock: 2

Group settings: 0

Ext\_User\_Prm\_Data:

"External Diagnosis (only DP)": Disabled (off)

Selected modules:

Param + 3 PD (4+3 words), 0xF3, 0xF2 (++)

Param + 3 PD (4+3 words)

AT %IW2407: WORD;  
AT %IX2407.0: BOOL; (\* Bit 0 \*)  
AT %IX2407.1: BOOL; (\* Bit 1 \*)  
AT %IX2407.2: BOOL; (\* Bit 2 \*)  
AT %IX2407.3: BOOL; (\* Bit 3 \*)  
AT %IX2407.4: BOOL; (\* Bit 4 \*)  
AT %IX2407.5: BOOL; (\* Bit 5 \*)  
AT %IX2407.6: BOOL; (\* Bit 6 \*)  
AT %IX2407.7: BOOL; (\* Bit 7 \*)  
AT %IX2407.8: BOOL; (\* Bit 8 \*)  
AT %IX2407.9: BOOL; (\* Bit 9 \*)  
AT %IX2407.10: BOOL; (\* Bit 10 \*)

AT %IX2407.11: BOOL; (\* Bit 11 \*)  
AT %IX2407.12: BOOL; (\* Bit 12 \*)  
AT %IX2407.13: BOOL; (\* Bit 13 \*)  
AT %IX2407.14: BOOL; (\* Bit 14 \*)  
AT %IX2407.15: BOOL; (\* Bit 15 \*)  
AT %IW2408: WORD;  
AT %IX2408.0: BOOL; (\* Bit 0 \*)  
AT %IX2408.1: BOOL; (\* Bit 1 \*)  
AT %IX2408.2: BOOL; (\* Bit 2 \*)  
AT %IX2408.3: BOOL; (\* Bit 3 \*)  
AT %IX2408.4: BOOL; (\* Bit 4 \*)  
AT %IX2408.5: BOOL; (\* Bit 5 \*)  
AT %IX2408.6: BOOL; (\* Bit 6 \*)  
AT %IX2408.7: BOOL; (\* Bit 7 \*)  
AT %IX2408.8: BOOL; (\* Bit 8 \*)  
AT %IX2408.9: BOOL; (\* Bit 9 \*)  
AT %IX2408.10: BOOL; (\* Bit 10 \*)  
AT %IX2408.11: BOOL; (\* Bit 11 \*)  
AT %IX2408.12: BOOL; (\* Bit 12 \*)  
AT %IX2408.13: BOOL; (\* Bit 13 \*)  
AT %IX2408.14: BOOL; (\* Bit 14 \*)  
AT %IX2408.15: BOOL; (\* Bit 15 \*)  
AT %IW2409: WORD;  
AT %IX2409.0: BOOL; (\* Bit 0 \*)  
AT %IX2409.1: BOOL; (\* Bit 1 \*)  
AT %IX2409.2: BOOL; (\* Bit 2 \*)  
AT %IX2409.3: BOOL; (\* Bit 3 \*)  
AT %IX2409.4: BOOL; (\* Bit 4 \*)  
AT %IX2409.5: BOOL; (\* Bit 5 \*)  
AT %IX2409.6: BOOL; (\* Bit 6 \*)  
AT %IX2409.7: BOOL; (\* Bit 7 \*)  
AT %IX2409.8: BOOL; (\* Bit 8 \*)  
AT %IX2409.9: BOOL; (\* Bit 9 \*)  
AT %IX2409.10: BOOL; (\* Bit 10 \*)  
AT %IX2409.11: BOOL; (\* Bit 11 \*)  
AT %IX2409.12: BOOL; (\* Bit 12 \*)  
AT %IX2409.13: BOOL; (\* Bit 13 \*)  
AT %IX2409.14: BOOL; (\* Bit 14 \*)  
AT %IX2409.15: BOOL; (\* Bit 15 \*)  
AT %IW2410: WORD;  
AT %IX2410.0: BOOL; (\* Bit 0 \*)  
AT %IX2410.1: BOOL; (\* Bit 1 \*)  
AT %IX2410.2: BOOL; (\* Bit 2 \*)  
AT %IX2410.3: BOOL; (\* Bit 3 \*)  
AT %IX2410.4: BOOL; (\* Bit 4 \*)  
AT %IX2410.5: BOOL; (\* Bit 5 \*)  
AT %IX2410.6: BOOL; (\* Bit 6 \*)  
AT %IX2410.7: BOOL; (\* Bit 7 \*)  
AT %IX2410.8: BOOL; (\* Bit 8 \*)  
AT %IX2410.9: BOOL; (\* Bit 9 \*)  
AT %IX2410.10: BOOL; (\* Bit 10 \*)  
AT %IX2410.11: BOOL; (\* Bit 11 \*)  
AT %IX2410.12: BOOL; (\* Bit 12 \*)  
AT %IX2410.13: BOOL; (\* Bit 13 \*)  
AT %IX2410.14: BOOL; (\* Bit 14 \*)  
AT %IX2410.15: BOOL; (\* Bit 15 \*)  
AT %QW2407: WORD;  
AT %QX2407.0: BOOL; (\* Bit 0 \*)  
AT %QX2407.1: BOOL; (\* Bit 1 \*)  
AT %QX2407.2: BOOL; (\* Bit 2 \*)  
AT %QX2407.3: BOOL; (\* Bit 3 \*)  
AT %QX2407.4: BOOL; (\* Bit 4 \*)  
AT %QX2407.5: BOOL; (\* Bit 5 \*)  
AT %QX2407.6: BOOL; (\* Bit 6 \*)  
AT %QX2407.7: BOOL; (\* Bit 7 \*)  
AT %QX2407.8: BOOL; (\* Bit 8 \*)  
AT %QX2407.9: BOOL; (\* Bit 9 \*)  
AT %QX2407.10: BOOL; (\* Bit 10 \*)  
AT %QX2407.11: BOOL; (\* Bit 11 \*)  
AT %QX2407.12: BOOL; (\* Bit 12 \*)  
AT %QX2407.13: BOOL; (\* Bit 13 \*)  
AT %QX2407.14: BOOL; (\* Bit 14 \*)  
AT %QX2407.15: BOOL; (\* Bit 15 \*)  
AT %QW2408: WORD;  
AT %QX2408.0: BOOL; (\* Bit 0 \*)  
AT %QX2408.1: BOOL; (\* Bit 1 \*)  
AT %QX2408.2: BOOL; (\* Bit 2 \*)  
AT %QX2408.3: BOOL; (\* Bit 3 \*)

AT %QX2408.4: BOOL; (\* Bit 4 \*)  
AT %QX2408.5: BOOL; (\* Bit 5 \*)  
AT %QX2408.6: BOOL; (\* Bit 6 \*)  
AT %QX2408.7: BOOL; (\* Bit 7 \*)  
AT %QX2408.8: BOOL; (\* Bit 8 \*)  
AT %QX2408.9: BOOL; (\* Bit 9 \*)  
AT %QX2408.10: BOOL; (\* Bit 10 \*)  
AT %QX2408.11: BOOL; (\* Bit 11 \*)  
AT %QX2408.12: BOOL; (\* Bit 12 \*)  
AT %QX2408.13: BOOL; (\* Bit 13 \*)  
AT %QX2408.14: BOOL; (\* Bit 14 \*)  
AT %QX2408.15: BOOL; (\* Bit 15 \*)  
AT %QW2409: WORD;  
AT %QX2409.0: BOOL; (\* Bit 0 \*)  
AT %QX2409.1: BOOL; (\* Bit 1 \*)  
AT %QX2409.2: BOOL; (\* Bit 2 \*)  
AT %QX2409.3: BOOL; (\* Bit 3 \*)  
AT %QX2409.4: BOOL; (\* Bit 4 \*)  
AT %QX2409.5: BOOL; (\* Bit 5 \*)  
AT %QX2409.6: BOOL; (\* Bit 6 \*)  
AT %QX2409.7: BOOL; (\* Bit 7 \*)  
AT %QX2409.8: BOOL; (\* Bit 8 \*)  
AT %QX2409.9: BOOL; (\* Bit 9 \*)  
AT %QX2409.10: BOOL; (\* Bit 10 \*)  
AT %QX2409.11: BOOL; (\* Bit 11 \*)  
AT %QX2409.12: BOOL; (\* Bit 12 \*)  
AT %QX2409.13: BOOL; (\* Bit 13 \*)  
AT %QX2409.14: BOOL; (\* Bit 14 \*)  
AT %QX2409.15: BOOL; (\* Bit 15 \*)  
AT %QW2410: WORD;  
AT %QX2410.0: BOOL; (\* Bit 0 \*)  
AT %QX2410.1: BOOL; (\* Bit 1 \*)  
AT %QX2410.2: BOOL; (\* Bit 2 \*)  
AT %QX2410.3: BOOL; (\* Bit 3 \*)  
AT %QX2410.4: BOOL; (\* Bit 4 \*)  
AT %QX2410.5: BOOL; (\* Bit 5 \*)  
AT %QX2410.6: BOOL; (\* Bit 6 \*)  
AT %QX2410.7: BOOL; (\* Bit 7 \*)  
AT %QX2410.8: BOOL; (\* Bit 8 \*)  
AT %QX2410.9: BOOL; (\* Bit 9 \*)  
AT %QX2410.10: BOOL; (\* Bit 10 \*)  
AT %QX2410.11: BOOL; (\* Bit 11 \*)  
AT %QX2410.12: BOOL; (\* Bit 12 \*)  
AT %QX2410.13: BOOL; (\* Bit 13 \*)  
AT %QX2410.14: BOOL; (\* Bit 14 \*)  
AT %QX2410.15: BOOL; (\* Bit 15 \*)  
AT %IW2411: WORD;  
AT %IX2411.0: BOOL; (\* Bit 0 \*)  
AT %IX2411.1: BOOL; (\* Bit 1 \*)  
AT %IX2411.2: BOOL; (\* Bit 2 \*)  
AT %IX2411.3: BOOL; (\* Bit 3 \*)  
AT %IX2411.4: BOOL; (\* Bit 4 \*)  
AT %IX2411.5: BOOL; (\* Bit 5 \*)  
AT %IX2411.6: BOOL; (\* Bit 6 \*)  
AT %IX2411.7: BOOL; (\* Bit 7 \*)  
AT %IX2411.8: BOOL; (\* Bit 8 \*)  
AT %IX2411.9: BOOL; (\* Bit 9 \*)  
AT %IX2411.10: BOOL; (\* Bit 10 \*)  
AT %IX2411.11: BOOL; (\* Bit 11 \*)  
AT %IX2411.12: BOOL; (\* Bit 12 \*)  
AT %IX2411.13: BOOL; (\* Bit 13 \*)  
AT %IX2411.14: BOOL; (\* Bit 14 \*)  
AT %IX2411.15: BOOL; (\* Bit 15 \*)  
AT %IW2412: WORD;  
AT %IX2412.0: BOOL; (\* Bit 0 \*)  
AT %IX2412.1: BOOL; (\* Bit 1 \*)  
AT %IX2412.2: BOOL; (\* Bit 2 \*)  
AT %IX2412.3: BOOL; (\* Bit 3 \*)  
AT %IX2412.4: BOOL; (\* Bit 4 \*)  
AT %IX2412.5: BOOL; (\* Bit 5 \*)  
AT %IX2412.6: BOOL; (\* Bit 6 \*)  
AT %IX2412.7: BOOL; (\* Bit 7 \*)  
AT %IX2412.8: BOOL; (\* Bit 8 \*)  
AT %IX2412.9: BOOL; (\* Bit 9 \*)  
AT %IX2412.10: BOOL; (\* Bit 10 \*)  
AT %IX2412.11: BOOL; (\* Bit 11 \*)  
AT %IX2412.12: BOOL; (\* Bit 12 \*)  
AT %IX2412.13: BOOL; (\* Bit 13 \*)

```

    AT %IX2412.14: BOOL; (* Bit 14 *)
    AT %IX2412.15: BOOL; (* Bit 15 *)
AT %IW2413: WORD;
    AT %IX2413.0: BOOL; (* Bit 0 *)
    AT %IX2413.1: BOOL; (* Bit 1 *)
    AT %IX2413.2: BOOL; (* Bit 2 *)
    AT %IX2413.3: BOOL; (* Bit 3 *)
    AT %IX2413.4: BOOL; (* Bit 4 *)
    AT %IX2413.5: BOOL; (* Bit 5 *)
    AT %IX2413.6: BOOL; (* Bit 6 *)
    AT %IX2413.7: BOOL; (* Bit 7 *)
    AT %IX2413.8: BOOL; (* Bit 8 *)
    AT %IX2413.9: BOOL; (* Bit 9 *)
    AT %IX2413.10: BOOL; (* Bit 10 *)
    AT %IX2413.11: BOOL; (* Bit 11 *)
    AT %IX2413.12: BOOL; (* Bit 12 *)
    AT %IX2413.13: BOOL; (* Bit 13 *)
    AT %IX2413.14: BOOL; (* Bit 14 *)
    AT %IX2413.15: BOOL; (* Bit 15 *)
AT %QW2411: WORD;
    AT %QX2411.0: BOOL; (* Bit 0 *)
    AT %QX2411.1: BOOL; (* Bit 1 *)
    AT %QX2411.2: BOOL; (* Bit 2 *)
    AT %QX2411.3: BOOL; (* Bit 3 *)
    AT %QX2411.4: BOOL; (* Bit 4 *)
    AT %QX2411.5: BOOL; (* Bit 5 *)
    AT %QX2411.6: BOOL; (* Bit 6 *)
    AT %QX2411.7: BOOL; (* Bit 7 *)
    AT %QX2411.8: BOOL; (* Bit 8 *)
    AT %QX2411.9: BOOL; (* Bit 9 *)
    AT %QX2411.10: BOOL; (* Bit 10 *)
    AT %QX2411.11: BOOL; (* Bit 11 *)
    AT %QX2411.12: BOOL; (* Bit 12 *)
    AT %QX2411.13: BOOL; (* Bit 13 *)
    AT %QX2411.14: BOOL; (* Bit 14 *)
    AT %QX2411.15: BOOL; (* Bit 15 *)
AT %QW2412: WORD;
    AT %QX2412.0: BOOL; (* Bit 0 *)
    AT %QX2412.1: BOOL; (* Bit 1 *)
    AT %QX2412.2: BOOL; (* Bit 2 *)
    AT %QX2412.3: BOOL; (* Bit 3 *)
    AT %QX2412.4: BOOL; (* Bit 4 *)
    AT %QX2412.5: BOOL; (* Bit 5 *)
    AT %QX2412.6: BOOL; (* Bit 6 *)
    AT %QX2412.7: BOOL; (* Bit 7 *)
    AT %QX2412.8: BOOL; (* Bit 8 *)
    AT %QX2412.9: BOOL; (* Bit 9 *)
    AT %QX2412.10: BOOL; (* Bit 10 *)
    AT %QX2412.11: BOOL; (* Bit 11 *)
    AT %QX2412.12: BOOL; (* Bit 12 *)
    AT %QX2412.13: BOOL; (* Bit 13 *)
    AT %QX2412.14: BOOL; (* Bit 14 *)
    AT %QX2412.15: BOOL; (* Bit 15 *)
AT %QW2413: WORD;
    AT %QX2413.0: BOOL; (* Bit 0 *)
    AT %QX2413.1: BOOL; (* Bit 1 *)
    AT %QX2413.2: BOOL; (* Bit 2 *)
    AT %QX2413.3: BOOL; (* Bit 3 *)
    AT %QX2413.4: BOOL; (* Bit 4 *)
    AT %QX2413.5: BOOL; (* Bit 5 *)
    AT %QX2413.6: BOOL; (* Bit 6 *)
    AT %QX2413.7: BOOL; (* Bit 7 *)
    AT %QX2413.8: BOOL; (* Bit 8 *)
    AT %QX2413.9: BOOL; (* Bit 9 *)
    AT %QX2413.10: BOOL; (* Bit 10 *)
    AT %QX2413.11: BOOL; (* Bit 11 *)
    AT %QX2413.12: BOOL; (* Bit 12 *)
    AT %QX2413.13: BOOL; (* Bit 13 *)
    AT %QX2413.14: BOOL; (* Bit 14 *)
    AT %QX2413.15: BOOL; (* Bit 15 *)

```

\*MOVIDRIVE DFP21/MCH41[VAR] (Id.: 69230394)

Node number: 2

Input address: %IB4828

Output address: %QB4828

Diagnostic address: %MB8052

Download: 1

AutoAdr: 1

Device dependent data:

GSD-File: SEW\_6003.gsd

Station number: 4

Station active: 1

TSDR: 11

Watchdog: 1

Watchdog Time: 1000

Fail Save: 1

Freeze:0

Sync: 0

Lock/Unlock: 2

Group settings: 0

Ext\_User\_Prm\_Data:

"External Diagnosis (only DP)": Disabled (off)

Selected modules:

Param + 3 PD (4+3 words), 0xF3, 0xF2 (++)

Param + 3 PD (4+3 words)

AT %IW2414: WORD;

AT %IX2414.0: BOOL; (\* Bit 0 \*)

AT %IX2414.1: BOOL; (\* Bit 1 \*)

AT %IX2414.2: BOOL; (\* Bit 2 \*)

AT %IX2414.3: BOOL; (\* Bit 3 \*)

AT %IX2414.4: BOOL; (\* Bit 4 \*)

AT %IX2414.5: BOOL; (\* Bit 5 \*)

AT %IX2414.6: BOOL; (\* Bit 6 \*)

AT %IX2414.7: BOOL; (\* Bit 7 \*)

AT %IX2414.8: BOOL; (\* Bit 8 \*)

AT %IX2414.9: BOOL; (\* Bit 9 \*)

AT %IX2414.10: BOOL; (\* Bit 10 \*)

AT %IX2414.11: BOOL; (\* Bit 11 \*)

AT %IX2414.12: BOOL; (\* Bit 12 \*)

AT %IX2414.13: BOOL; (\* Bit 13 \*)

AT %IX2414.14: BOOL; (\* Bit 14 \*)

AT %IX2414.15: BOOL; (\* Bit 15 \*)

AT %IW2415: WORD;

AT %IX2415.0: BOOL; (\* Bit 0 \*)

AT %IX2415.1: BOOL; (\* Bit 1 \*)

AT %IX2415.2: BOOL; (\* Bit 2 \*)

AT %IX2415.3: BOOL; (\* Bit 3 \*)

AT %IX2415.4: BOOL; (\* Bit 4 \*)

AT %IX2415.5: BOOL; (\* Bit 5 \*)

AT %IX2415.6: BOOL; (\* Bit 6 \*)

AT %IX2415.7: BOOL; (\* Bit 7 \*)

AT %IX2415.8: BOOL; (\* Bit 8 \*)

AT %IX2415.9: BOOL; (\* Bit 9 \*)

AT %IX2415.10: BOOL; (\* Bit 10 \*)

AT %IX2415.11: BOOL; (\* Bit 11 \*)

AT %IX2415.12: BOOL; (\* Bit 12 \*)

AT %IX2415.13: BOOL; (\* Bit 13 \*)

AT %IX2415.14: BOOL; (\* Bit 14 \*)

AT %IX2415.15: BOOL; (\* Bit 15 \*)

AT %IW2416: WORD;

AT %IX2416.0: BOOL; (\* Bit 0 \*)

AT %IX2416.1: BOOL; (\* Bit 1 \*)

AT %IX2416.2: BOOL; (\* Bit 2 \*)

AT %IX2416.3: BOOL; (\* Bit 3 \*)

AT %IX2416.4: BOOL; (\* Bit 4 \*)

AT %IX2416.5: BOOL; (\* Bit 5 \*)

AT %IX2416.6: BOOL; (\* Bit 6 \*)

AT %IX2416.7: BOOL; (\* Bit 7 \*)

AT %IX2416.8: BOOL; (\* Bit 8 \*)

AT %IX2416.9: BOOL; (\* Bit 9 \*)

AT %IX2416.10: BOOL; (\* Bit 10 \*)

AT %IX2416.11: BOOL; (\* Bit 11 \*)

AT %IX2416.12: BOOL; (\* Bit 12 \*)

AT %IX2416.13: BOOL; (\* Bit 13 \*)

AT %IX2416.14: BOOL; (\* Bit 14 \*)

AT %IX2416.15: BOOL; (\* Bit 15 \*)

AT %IW2417: WORD;

AT %IX2417.0: BOOL; (\* Bit 0 \*)

AT %IX2417.1: BOOL; (\* Bit 1 \*)

AT %IX2417.2: BOOL; (\* Bit 2 \*)

AT %IX2417.3: BOOL; (\* Bit 3 \*)

AT %IX2417.4: BOOL; (\* Bit 4 \*)

AT %IX2417.5: BOOL; (\* Bit 5 \*)

AT %IX2417.6: BOOL; (\* Bit 6 \*)

AT %IX2417.7: BOOL; (\* Bit 7 \*)

AT %IX2417.8: BOOL; (\* Bit 8 \*)  
AT %IX2417.9: BOOL; (\* Bit 9 \*)  
AT %IX2417.10: BOOL; (\* Bit 10 \*)  
AT %IX2417.11: BOOL; (\* Bit 11 \*)  
AT %IX2417.12: BOOL; (\* Bit 12 \*)  
AT %IX2417.13: BOOL; (\* Bit 13 \*)  
AT %IX2417.14: BOOL; (\* Bit 14 \*)  
AT %IX2417.15: BOOL; (\* Bit 15 \*)  
AT %QW2414: WORD;  
AT %QX2414.0: BOOL; (\* Bit 0 \*)  
AT %QX2414.1: BOOL; (\* Bit 1 \*)  
AT %QX2414.2: BOOL; (\* Bit 2 \*)  
AT %QX2414.3: BOOL; (\* Bit 3 \*)  
AT %QX2414.4: BOOL; (\* Bit 4 \*)  
AT %QX2414.5: BOOL; (\* Bit 5 \*)  
AT %QX2414.6: BOOL; (\* Bit 6 \*)  
AT %QX2414.7: BOOL; (\* Bit 7 \*)  
AT %QX2414.8: BOOL; (\* Bit 8 \*)  
AT %QX2414.9: BOOL; (\* Bit 9 \*)  
AT %QX2414.10: BOOL; (\* Bit 10 \*)  
AT %QX2414.11: BOOL; (\* Bit 11 \*)  
AT %QX2414.12: BOOL; (\* Bit 12 \*)  
AT %QX2414.13: BOOL; (\* Bit 13 \*)  
AT %QX2414.14: BOOL; (\* Bit 14 \*)  
AT %QX2414.15: BOOL; (\* Bit 15 \*)  
AT %QW2415: WORD;  
AT %QX2415.0: BOOL; (\* Bit 0 \*)  
AT %QX2415.1: BOOL; (\* Bit 1 \*)  
AT %QX2415.2: BOOL; (\* Bit 2 \*)  
AT %QX2415.3: BOOL; (\* Bit 3 \*)  
AT %QX2415.4: BOOL; (\* Bit 4 \*)  
AT %QX2415.5: BOOL; (\* Bit 5 \*)  
AT %QX2415.6: BOOL; (\* Bit 6 \*)  
AT %QX2415.7: BOOL; (\* Bit 7 \*)  
AT %QX2415.8: BOOL; (\* Bit 8 \*)  
AT %QX2415.9: BOOL; (\* Bit 9 \*)  
AT %QX2415.10: BOOL; (\* Bit 10 \*)  
AT %QX2415.11: BOOL; (\* Bit 11 \*)  
AT %QX2415.12: BOOL; (\* Bit 12 \*)  
AT %QX2415.13: BOOL; (\* Bit 13 \*)  
AT %QX2415.14: BOOL; (\* Bit 14 \*)  
AT %QX2415.15: BOOL; (\* Bit 15 \*)  
AT %QW2416: WORD;  
AT %QX2416.0: BOOL; (\* Bit 0 \*)  
AT %QX2416.1: BOOL; (\* Bit 1 \*)  
AT %QX2416.2: BOOL; (\* Bit 2 \*)  
AT %QX2416.3: BOOL; (\* Bit 3 \*)  
AT %QX2416.4: BOOL; (\* Bit 4 \*)  
AT %QX2416.5: BOOL; (\* Bit 5 \*)  
AT %QX2416.6: BOOL; (\* Bit 6 \*)  
AT %QX2416.7: BOOL; (\* Bit 7 \*)  
AT %QX2416.8: BOOL; (\* Bit 8 \*)  
AT %QX2416.9: BOOL; (\* Bit 9 \*)  
AT %QX2416.10: BOOL; (\* Bit 10 \*)  
AT %QX2416.11: BOOL; (\* Bit 11 \*)  
AT %QX2416.12: BOOL; (\* Bit 12 \*)  
AT %QX2416.13: BOOL; (\* Bit 13 \*)  
AT %QX2416.14: BOOL; (\* Bit 14 \*)  
AT %QX2416.15: BOOL; (\* Bit 15 \*)  
AT %QW2417: WORD;  
AT %QX2417.0: BOOL; (\* Bit 0 \*)  
AT %QX2417.1: BOOL; (\* Bit 1 \*)  
AT %QX2417.2: BOOL; (\* Bit 2 \*)  
AT %QX2417.3: BOOL; (\* Bit 3 \*)  
AT %QX2417.4: BOOL; (\* Bit 4 \*)  
AT %QX2417.5: BOOL; (\* Bit 5 \*)  
AT %QX2417.6: BOOL; (\* Bit 6 \*)  
AT %QX2417.7: BOOL; (\* Bit 7 \*)  
AT %QX2417.8: BOOL; (\* Bit 8 \*)  
AT %QX2417.9: BOOL; (\* Bit 9 \*)  
AT %QX2417.10: BOOL; (\* Bit 10 \*)  
AT %QX2417.11: BOOL; (\* Bit 11 \*)  
AT %QX2417.12: BOOL; (\* Bit 12 \*)  
AT %QX2417.13: BOOL; (\* Bit 13 \*)  
AT %QX2417.14: BOOL; (\* Bit 14 \*)  
AT %QX2417.15: BOOL; (\* Bit 15 \*)  
AT %IW2418: WORD;  
AT %IX2418.0: BOOL; (\* Bit 0 \*)



AT %IX2418.1: BOOL; (\* Bit 1 \*)  
AT %IX2418.2: BOOL; (\* Bit 2 \*)  
AT %IX2418.3: BOOL; (\* Bit 3 \*)  
AT %IX2418.4: BOOL; (\* Bit 4 \*)  
AT %IX2418.5: BOOL; (\* Bit 5 \*)  
AT %IX2418.6: BOOL; (\* Bit 6 \*)  
AT %IX2418.7: BOOL; (\* Bit 7 \*)  
AT %IX2418.8: BOOL; (\* Bit 8 \*)  
AT %IX2418.9: BOOL; (\* Bit 9 \*)  
AT %IX2418.10: BOOL; (\* Bit 10 \*)  
AT %IX2418.11: BOOL; (\* Bit 11 \*)  
AT %IX2418.12: BOOL; (\* Bit 12 \*)  
AT %IX2418.13: BOOL; (\* Bit 13 \*)  
AT %IX2418.14: BOOL; (\* Bit 14 \*)  
AT %IX2418.15: BOOL; (\* Bit 15 \*)  
AT %IW2419: WORD;  
AT %IX2419.0: BOOL; (\* Bit 0 \*)  
AT %IX2419.1: BOOL; (\* Bit 1 \*)  
AT %IX2419.2: BOOL; (\* Bit 2 \*)  
AT %IX2419.3: BOOL; (\* Bit 3 \*)  
AT %IX2419.4: BOOL; (\* Bit 4 \*)  
AT %IX2419.5: BOOL; (\* Bit 5 \*)  
AT %IX2419.6: BOOL; (\* Bit 6 \*)  
AT %IX2419.7: BOOL; (\* Bit 7 \*)  
AT %IX2419.8: BOOL; (\* Bit 8 \*)  
AT %IX2419.9: BOOL; (\* Bit 9 \*)  
AT %IX2419.10: BOOL; (\* Bit 10 \*)  
AT %IX2419.11: BOOL; (\* Bit 11 \*)  
AT %IX2419.12: BOOL; (\* Bit 12 \*)  
AT %IX2419.13: BOOL; (\* Bit 13 \*)  
AT %IX2419.14: BOOL; (\* Bit 14 \*)  
AT %IX2419.15: BOOL; (\* Bit 15 \*)  
AT %IW2420: WORD;  
AT %IX2420.0: BOOL; (\* Bit 0 \*)  
AT %IX2420.1: BOOL; (\* Bit 1 \*)  
AT %IX2420.2: BOOL; (\* Bit 2 \*)  
AT %IX2420.3: BOOL; (\* Bit 3 \*)  
AT %IX2420.4: BOOL; (\* Bit 4 \*)  
AT %IX2420.5: BOOL; (\* Bit 5 \*)  
AT %IX2420.6: BOOL; (\* Bit 6 \*)  
AT %IX2420.7: BOOL; (\* Bit 7 \*)  
AT %IX2420.8: BOOL; (\* Bit 8 \*)  
AT %IX2420.9: BOOL; (\* Bit 9 \*)  
AT %IX2420.10: BOOL; (\* Bit 10 \*)  
AT %IX2420.11: BOOL; (\* Bit 11 \*)  
AT %IX2420.12: BOOL; (\* Bit 12 \*)  
AT %IX2420.13: BOOL; (\* Bit 13 \*)  
AT %IX2420.14: BOOL; (\* Bit 14 \*)  
AT %IX2420.15: BOOL; (\* Bit 15 \*)  
AT %QW2418: WORD;  
AT %QX2418.0: BOOL; (\* Bit 0 \*)  
AT %QX2418.1: BOOL; (\* Bit 1 \*)  
AT %QX2418.2: BOOL; (\* Bit 2 \*)  
AT %QX2418.3: BOOL; (\* Bit 3 \*)  
AT %QX2418.4: BOOL; (\* Bit 4 \*)  
AT %QX2418.5: BOOL; (\* Bit 5 \*)  
AT %QX2418.6: BOOL; (\* Bit 6 \*)  
AT %QX2418.7: BOOL; (\* Bit 7 \*)  
AT %QX2418.8: BOOL; (\* Bit 8 \*)  
AT %QX2418.9: BOOL; (\* Bit 9 \*)  
AT %QX2418.10: BOOL; (\* Bit 10 \*)  
AT %QX2418.11: BOOL; (\* Bit 11 \*)  
AT %QX2418.12: BOOL; (\* Bit 12 \*)  
AT %QX2418.13: BOOL; (\* Bit 13 \*)  
AT %QX2418.14: BOOL; (\* Bit 14 \*)  
AT %QX2418.15: BOOL; (\* Bit 15 \*)  
AT %QW2419: WORD;  
AT %QX2419.0: BOOL; (\* Bit 0 \*)  
AT %QX2419.1: BOOL; (\* Bit 1 \*)  
AT %QX2419.2: BOOL; (\* Bit 2 \*)  
AT %QX2419.3: BOOL; (\* Bit 3 \*)  
AT %QX2419.4: BOOL; (\* Bit 4 \*)  
AT %QX2419.5: BOOL; (\* Bit 5 \*)  
AT %QX2419.6: BOOL; (\* Bit 6 \*)  
AT %QX2419.7: BOOL; (\* Bit 7 \*)  
AT %QX2419.8: BOOL; (\* Bit 8 \*)  
AT %QX2419.9: BOOL; (\* Bit 9 \*)  
AT %QX2419.10: BOOL; (\* Bit 10 \*)

```
AT %QX2419.11: BOOL; (* Bit 11 *)
AT %QX2419.12: BOOL; (* Bit 12 *)
AT %QX2419.13: BOOL; (* Bit 13 *)
AT %QX2419.14: BOOL; (* Bit 14 *)
AT %QX2419.15: BOOL; (* Bit 15 *)
AT %QW2420: WORD;
AT %QX2420.0: BOOL; (* Bit 0 *)
AT %QX2420.1: BOOL; (* Bit 1 *)
AT %QX2420.2: BOOL; (* Bit 2 *)
AT %QX2420.3: BOOL; (* Bit 3 *)
AT %QX2420.4: BOOL; (* Bit 4 *)
AT %QX2420.5: BOOL; (* Bit 5 *)
AT %QX2420.6: BOOL; (* Bit 6 *)
AT %QX2420.7: BOOL; (* Bit 7 *)
AT %QX2420.8: BOOL; (* Bit 8 *)
AT %QX2420.9: BOOL; (* Bit 9 *)
AT %QX2420.10: BOOL; (* Bit 10 *)
AT %QX2420.11: BOOL; (* Bit 11 *)
AT %QX2420.12: BOOL; (* Bit 12 *)
AT %QX2420.13: BOOL; (* Bit 13 *)
AT %QX2420.14: BOOL; (* Bit 14 *)
AT %QX2420.15: BOOL; (* Bit 15 *)
```

Sampling Trace

No trace loaded

Task configuration

Task configuration

System events

NewTask (PRIORITY := 15, INTERVAL := T#1ms)

PLC\_PRG();

Read\_Pos();

Read\_Param();

SET\_STEP();

Watch- and Recipe Manager

Workspace

Parameter Manager

0001	Parameter-Manager
0002	=====

Cross Reference List

Y_M3		
	_global_init (22)	Global Write
Y_M2		
	_global_init (18)	Global Write
Y_M1		
	_global_init (14)	Global Write
X_M3		
	_global_init (3)	Local Write
	WAVE_V3 (44)	Local Read
	WAVE_V3 (45)	Local Read
	WAVE_V3 (45)	Local Write
	WAVE_V3 (48)	Local Read
	WAVE_V3 (53)	Local Read
	WAVE_V3 (53)	Local Write
	WAVE_V3 (56)	Local Read
	WAVE_V3 (60)	Local Read
X_M2		
	_global_init (2)	Local Write
	WAVE_V3 (23)	Local Read
	WAVE_V3 (24)	Local Read
	WAVE_V3 (24)	Local Write
	WAVE_V3 (27)	Local Read
	WAVE_V3 (32)	Local Read
	WAVE_V3 (32)	Local Write
	WAVE_V3 (35)	Local Read
	WAVE_V3 (39)	Local Read
X_M1		
	_global_init (1)	Local Write
	WAVE_V3 (2)	Local Read
	WAVE_V3 (3)	Local Read
	WAVE_V3 (3)	Local Write
	WAVE_V3 (6)	Local Read
	WAVE_V3 (11)	Local Read
	WAVE_V3 (11)	Local Write
	WAVE_V3 (14)	Local Read

	WAVE_V3 (20)	Local Read
top_M1		
	_global_init (4)	Local Write
	WAVE_V3 (2)	Local Read
	WAVE_V3 (7)	Local Write
	WAVE_V3 (10)	Local Read
	WAVE_V3 (15)	Local Write
Word1%QW2401		
	READ_PARAM (6)	Local Write
	_global_init (3)	Local Write
Word1%QW2401		
	READ_PARAM (6)	Local Write
	_global_init (3)	Local Write
Word0%QW2400		
	READ_PARAM (3)	Local Write
	READ_PARAM (15)	Local Write
	_global_init (2)	Local Write
STEP_M1		
	SET_STEP (6)	Local Write
	SET_STEP (7)	Local Read
	SET_STEP (8)	Local Write
	SET_STEP (12)	Local Write
	SET_STEP (13)	Local Read
	SET_STEP (14)	Local Write
	SET_STEP (17)	Local Write
	SET_STEP (19)	Local Read
	_global_init (2)	Local Write
Word0%QW2400		
	READ_PARAM (3)	Local Write
	READ_PARAM (15)	Local Write
	_global_init (2)	Local Write
wave_run		
	PLC_PRG (3)	Global Read
	PLC_PRG (7)	Global Read
	PLC_PRG (11)	Global Read
	PLC_PRG (16)	Global Read
	_global_init (31)	Global Write
top_M3		
	_global_init (6)	Local Write
	WAVE_V3 (44)	Local Read
	WAVE_V3 (49)	Local Write
	WAVE_V3 (52)	Local Read
	WAVE_V3 (57)	Local Write
stepp		
	STEP_TO_HIGHT (1)	Local Read
	STEP_TO_HIGHTinit (3)	Local Write
	CALCULATED_ACTUALS (1)	Local Write
	CALCULATED_ACTUALS (2)	Local Write
	CALCULATED_ACTUALS (3)	Local Write
top_M2		
	_global_init (5)	Local Write
	WAVE_V3 (23)	Local Read
	WAVE_V3 (28)	Local Write
	WAVE_V3 (31)	Local Read
	WAVE_V3 (36)	Local Write
STOP		
	_global_init (1)	Global Write
STEP_M3		
	SET_STEP (42)	Local Write
	SET_STEP (43)	Local Read
	SET_STEP (44)	Local Write
	SET_STEP (47)	Local Write
	SET_STEP (48)	Local Read
	SET_STEP (49)	Local Write
	SET_STEP (52)	Local Write
	SET_STEP (54)	Local Read
	_global_init (6)	Local Write
STEP_M2		
	SET_STEP (25)	Local Write
	SET_STEP (26)	Local Read
	SET_STEP (27)	Local Write
	SET_STEP (31)	Local Write
	SET_STEP (32)	Local Read
	SET_STEP (33)	Local Write
	SET_STEP (36)	Local Write
	SET_STEP (38)	Local Read
	_global_init (4)	Local Write
step3		

step2	ROLL_PITCH_TO_STEP (27)	Local Write
	ROLL_PITCH_TO_STEP... (15)	Local Write
	PLC_PRG (20)	Local Read
	INPUT_MODE (17)	Local Read
st1	ROLL_PITCH_TO_STEP (26)	Local Write
	ROLL_PITCH_TO_STEP... (14)	Local Write
	PLC_PRG (19)	Local Read
	INPUT_MODE (16)	Local Read
step1	CALCULATED_ACTUALS... (1)	Local Read
	CALCULATED_ACTUALS (1)	Local Write
	CALCULATED_ACTUALS (5)	Local Read
	CALCULATED_ACTUALS (8)	Local Read
st3	ROLL_PITCH_TO_STEP (25)	Local Write
	ROLL_PITCH_TO_STEP... (13)	Local Write
	PLC_PRG (18)	Local Read
	INPUT_MODE (15)	Local Read
st2	CALCULATED_ACTUALS... (3)	Local Read
	CALCULATED_ACTUALS (3)	Local Write
	CALCULATED_ACTUALS (7)	Local Read
	CALCULATED_ACTUALS (10)	Local Read
SPEED_DIR	CALCULATED_ACTUALS... (2)	Local Read
	CALCULATED_ACTUALS (2)	Local Write
	CALCULATED_ACTUALS (6)	Local Read
	CALCULATED_ACTUALS (9)	Local Read
Speed2	M1 (3)	Local Read
	SET_STEP (19)	Local Write
SIMULATION_UDP	_global_init (8)	Local Write
	WAVE_V3 (24)	Local Read
	WAVE_V3 (32)	Local Read
Speed1	INPUT_MODE (9)	Global Read
	_global_init (33)	Global Write
RPTS	_global_init (7)	Local Write
	WAVE_V3 (3)	Local Read
	WAVE_V3 (11)	Local Read
roll_new	INPUT_MODE (13)	Local Write
	INPUT_MODE (15)	Local Read
	INPUT_MODE (16)	Local Read
	INPUT_MODE (17)	Local Read
SPEED_DIR	_global_init (7)	Local Read
	INPUT_MODE (4)	Local Write
	INPUT_MODE (7)	Local Read
roll_actual%MW3	_global_init (12)	Local Write
	M2 (3)	Local Read
Speed	SET_STEP (38)	Local Write
	READ_POS (18)	Local Write
rVersion	MANUAL (12)	Local Read
	MANUAL (17)	Local Read
	MANUAL (22)	Local Read
	MANUAL (27)	Local Read
	MANUAL (32)	Local Read
	MANUAL (37)	Local Read
	MANUAL (42)	Local Read
	MANUAL (43)	Local Read
	MANUAL (44)	Local Read
	MANUAL (49)	Local Read
	MANUAL (50)	Local Read
	MANUAL (51)	Local Read
	_global_init (1)	Local Write
Read_value1%IW2401	_global_init (71)	Global Write
	READ_PARAM (9)	Local Read
	READ_PARAM (11)	Local Read

Read_value0%IW2400		
READ_PARAM (8)	Local Read	
READ_PARAM (11)	Local Read	
Read_par		
READ_PARAM (1)	Local Read	
_global_init (8)	Local Write	
min_angle		
_global_init (8)	Global Write	
SPEED_DIR		
M3 (3)	Local Read	
SET_STEP (54)	Local Write	
mid_length1		
ROLL_PITCH_TO_STEP (16)	Local Write	
ROLL_PITCH_TO_STEP (19)	Local Read	
ROLL_PITCH_TO_STEP (20)	Local Read	
ROLL_PITCH_TO_STEP... (3)	Local Write	
step		
HIGHT_TO_STEP (12)	Local Write	
HIGHT_TO_STEPinit (7)	Local Write	
ROLL_PITCH_TO_STEP (25)	Local Read	
ROLL_PITCH_TO_STEP (26)	Local Read	
ROLL_PITCH_TO_STEP (27)	Local Read	
WAVE_V3 (65)	Local Read	
WAVE_V3 (66)	Local Read	
WAVE_V3 (67)	Local Read	
sim_roll		
INPUT_MODE (10)	Local Read	
_global_init (3)	Local Write	
sim_pitch		
INPUT_MODE (11)	Local Read	
_global_init (4)	Local Write	
Speed3		
_global_init (9)	Local Write	
WAVE_V3 (45)	Local Read	
WAVE_V3 (53)	Local Read	
set_p3		
ROLL_PITCH_TO_STEP (20)	Local Write	
ROLL_PITCH_TO_STEP (24)	Local Read	
ROLL_PITCH_TO_STEP... (10)	Local Write	
set_p2		
ROLL_PITCH_TO_STEP (19)	Local Write	
ROLL_PITCH_TO_STEP (23)	Local Read	
ROLL_PITCH_TO_STEP... (9)	Local Write	
SET_M2		
_global_init (3)	Local Write	
roll_old		
INPUT_MODE (7)	Local Read	
INPUT_MODE (19)	Local Write	
_global_init (10)	Local Write	
roll_new_int		
INPUT_MODE (1)	Local Write	
INPUT_MODE (4)	Local Read	
_global_init (14)	Local Write	
rpts		
PLC_PRG (17)	Local Write	
PLC_PRG (18)	Local Read	
PLC_PRG (19)	Local Read	
PLC_PRG (20)	Local Read	
_global_init (1)	Local Read	
pitch_real		
_global_init (8)	Local Write	
SET_M1		
_global_init (1)	Local Write	
pitch		
_global_init (27)	Global Write	
roll_real		
_global_init (9)	Local Write	
roll		
_global_init (28)	Global Write	
Par_nr		
READ_PARAM (5)	Local Write	
READ_PARAM (6)	Local Read	
_global_init (9)	Local Write	
roll_angle		
ROLL_PITCH_TO_STEP (1)	Local Read	
ROLL_PITCH_TO_STEP (2)	Local Write	
ROLL_PITCH_TO_STEP (3)	Local Read	
ROLL_PITCH_TO_STEP (4)	Local Write	

	ROLL_PITCH_TO_STEP (18)	Local Read
	ROLL_PITCH_TO_STEP... (11)	Local Write
	PLC_PRG (17)	Local Write
	INPUT_MODE (13)	Local Write
roll_actual%MW3		
	_global_init (5)	Local Write
	READ_POS (18)	Local Write
PR		
	_global_init (8)	Local Write
	READ_POS (16)	Local Write
	READ_POS (18)	Local Read
p2_step		
	PLC_PRG (19)	Global Write
	INPUT_MODE (16)	Global Write
	MANUAL (3)	Global Read
	MANUAL (3)	Global Write
	MANUAL (7)	Global Read
	MANUAL (7)	Global Write
	MANUAL (22)	Global Read
	MANUAL (22)	Global Write
	MANUAL (27)	Global Read
	MANUAL (27)	Global Write
	MANUAL (43)	Global Read
	MANUAL (43)	Global Write
	MANUAL (50)	Global Read
	MANUAL (50)	Global Write
	SET_STEP (23)	Global Read
	SET_STEP (25)	Global Read
	SET_STEP (30)	Global Read
	SET_STEP (31)	Global Read
	_global_init (25)	Global Write
	WAVE_V3 (66)	Global Write
Read_par		
	_global_init (2)	Local Write
p1_step		
	PLC_PRG (18)	Global Write
	INPUT_MODE (15)	Global Write
	MANUAL (2)	Global Read
	MANUAL (2)	Global Write
	MANUAL (6)	Global Read
	MANUAL (6)	Global Write
	MANUAL (12)	Global Read
	MANUAL (12)	Global Write
	MANUAL (17)	Global Read
	MANUAL (17)	Global Write
	MANUAL (42)	Global Read
	MANUAL (42)	Global Write
	MANUAL (49)	Global Read
	MANUAL (49)	Global Write
	SET_STEP (4)	Global Read
	SET_STEP (6)	Global Read
	SET_STEP (11)	Global Read
	SET_STEP (12)	Global Read
	_global_init (24)	Global Write
	WAVE_V3 (65)	Global Write
P0_2_M3%QW2419		
	M3 (2)	Local Write
	M3 (3)	Local Write
pitch_angle		
	ROLL_PITCH_TO_STEP (7)	Local Read
	ROLL_PITCH_TO_STEP (8)	Local Write
	ROLL_PITCH_TO_STEP (9)	Local Read
	ROLL_PITCH_TO_STEP (10)	Local Write
	ROLL_PITCH_TO_STEP (13)	Local Read
	ROLL_PITCH_TO_STEP... (12)	Local Write
	PLC_PRG (17)	Local Write
	INPUT_MODE (13)	Local Write
P0_2_M2%QW2412		
	M2 (2)	Local Write
	M2 (3)	Local Write
set_p1		
	ROLL_PITCH_TO_STEP (15)	Local Write
	ROLL_PITCH_TO_STEP (22)	Local Read
	ROLL_PITCH_TO_STEP... (8)	Local Write
Read_value1%IW2401		
	READ_PARAM (9)	Local Read
	READ_PARAM (11)	Local Read
Read_value0%IW2400		

	READ_PARAM (8)	Local Read
	READ_PARAM (11)	Local Read
pitch_old		
	INPUT_MODE (8)	Local Read
	INPUT_MODE (20)	Local Write
	_global_init (11)	Local Write
pitch_new_int		
	INPUT_MODE (2)	Local Write
	INPUT_MODE (5)	Local Read
	_global_init (15)	Local Write
SET_M3		
	_global_init (5)	Local Write
pitch_new		
	INPUT_MODE (5)	Local Write
	INPUT_MODE (8)	Local Read
	_global_init (13)	Local Write
pitch_actual%MW2		
	_global_init (6)	Local Write
	READ_POS (19)	Local Write
PA		
	_global_init (7)	Local Write
	READ_POS (14)	Local Write
	READ_POS (19)	Local Read
pitch_actual%MW2		
	_global_init (6)	Local Write
	READ_POS (19)	Local Write
P0_2_M3%QW2419		
	M3 (2)	Local Write
	M3 (3)	Local Write
P0_2_M2%QW2412		
	M2 (2)	Local Write
	M2 (3)	Local Write
p3_step		
	PLC_PRG (20)	Global Write
	INPUT_MODE (17)	Global Write
	MANUAL (4)	Global Read
	MANUAL (4)	Global Write
	MANUAL (8)	Global Read
	MANUAL (8)	Global Write
	MANUAL (32)	Global Read
	MANUAL (32)	Global Write
	MANUAL (37)	Global Read
	MANUAL (37)	Global Write
	MANUAL (44)	Global Read
	MANUAL (44)	Global Write
	MANUAL (51)	Global Read
	MANUAL (51)	Global Write
	SET_STEP (41)	Global Read
	SET_STEP (42)	Global Read
	SET_STEP (46)	Global Read
	SET_STEP (47)	Global Read
	_global_init (26)	Global Write
	WAVE_V3 (67)	Global Write
P0_2_M1%QW2405		
	M1 (2)	Local Write
	M1 (3)	Local Write
P0_1_M3%QW2418		
	M3 (1)	Local Write
	M3 (2)	Local Write
P0_1_M2%QW2411		
	M2 (1)	Local Write
	M2 (2)	Local Write
P0_1_M1%QW2404		
	M1 (1)	Local Write
	M1 (2)	Local Write
Out_put1		
	READ_PARAM (9)	Local Write
	_global_init (5)	Local Write
mid_length		
	ROLL_PITCH_TO_STEP (15)	Global Read
	ROLL_PITCH_TO_STEP (16)	Global Read
	_global_init (7)	Global Write
max_roll		
	ROLL_PITCH_TO_STEP (1)	Global Read
	ROLL_PITCH_TO_STEP (2)	Global Read
	_global_init (10)	Global Write
max_pitch		
	ROLL_PITCH_TO_STEP (7)	Global Read

	ROLL_PITCH_TO_STEP (8)	Global Read
	_global_init (12)	Global Write
manual_run		
	PLC_PRG (3)	Global Read
	PLC_PRG (7)	Global Read
	PLC_PRG (11)	Global Read
	PLC_PRG (16)	Global Read
	_global_init (30)	Global Write
P0_2_M1%QW2405		
	M1 (2)	Local Write
	M1 (3)	Local Write
Management		
	READ_PARAM (2)	Local Write
	READ_PARAM (3)	Local Read
	READ_PARAM (14)	Local Write
	READ_PARAM (15)	Local Read
	_global_init (1)	Local Write
P0_1_M3%QW2418		
	M3 (1)	Local Write
	M3 (2)	Local Write
min_roll		
	ROLL_PITCH_TO_STEP (3)	Global Read
	ROLL_PITCH_TO_STEP (4)	Global Read
	_global_init (11)	Global Write
Out_put0		
	READ_PARAM (8)	Local Write
	_global_init (4)	Local Write
max_angle		
	_global_init (9)	Global Write
M3_Cont_inhibit%QX2418.8		
	M3 (5)	Global Write
	_global_init (20)	Global Write
M3_ANGLE		
	_global_init (23)	Global Write
	WAVE_V3 (70)	Global Write
M2_ANGLE		
	_global_init (19)	Global Write
	WAVE_V3 (69)	Global Write
Manual_speed		
	_global_init (11)	Global Write
M1_ANGLE		
	_global_init (15)	Global Write
	WAVE_V3 (68)	Global Write
M3		
	M3 (3)	Local Write
	M3 (0)	Local Read
	SET_STEP (54)	Local Read
I6		
	ROLL_PITCH_TO_STEP (15)	Global Read
	_global_init (6)	Global Write
P0_1_M2%QW2411		
	M2 (1)	Local Write
	M2 (2)	Local Write
I4		
	CALCULATED_ACTUALS (16)	Global Read
	_global_init (4)	Global Write
Out		
	_global_init (6)	Local Write
min_pitch		
	ROLL_PITCH_TO_STEP (9)	Global Read
	ROLL_PITCH_TO_STEP (10)	Global Read
	_global_init (13)	Global Write
M3_Cont_inhibit%QX2418.8		
	M3 (5)	Global Write
	_global_init (20)	Global Write
M2_Cont_inhibit%QX2411.8		
	M2 (5)	Global Write
	_global_init (16)	Global Write
M2_Cont_inhibit%QX2411.8		
	M2 (5)	Global Write
	_global_init (16)	Global Write
LOGG		
	_global_init (34)	Global Write
I3		
	ROLL_PITCH_TO_STEP (18)	Global Read
	CALCULATED_ACTUALS (13)	Global Read
	_global_init (3)	Global Write
M2		



I2	M2 (3)	Local Write
	M2 (0)	Local Read
	SET_STEP (38)	Local Read
P0_1_M1%QW2404	HIGHT_TO_STEP (9)	Global Read
	STEP_TO_HIGHT (4)	Global Read
	_global_init (2)	Global Write
M1_Cont_inhibit%QX2404.8	M1 (1)	Local Write
	M1 (2)	Local Write
	M1 (5)	Global Write
M1_Cont_inhibit%QX2404.8	_global_init (12)	Global Write
	M1 (5)	Global Write
	_global_init (12)	Global Write
JOGP3	MANUAL (31)	Global Read
	_global_init (7)	Global Write
I1	HIGHT_TO_STEP (9)	Global Read
	HIGHT_TO_STEP (10)	Global Read
	STEP_TO_HIGHT (4)	Global Read
JOGP2	_global_init (1)	Global Write
	MANUAL (21)	Global Read
	_global_init (5)	Global Write
JOGM2	MANUAL (26)	Global Read
	_global_init (4)	Global Write
M1	M1 (3)	Local Write
	M1 (0)	Local Read
	SET_STEP (19)	Local Read
INPUT_roll_real	INPUT_MODE (7)	Local Write
	INPUT_MODE (10)	Local Write
	INPUT_MODE (13)	Local Read
length	INPUT_MODE (19)	Local Read
	_global_init (5)	Local Write
	HIGHT_TO_STEP (1)	Local Read
JOGP1	HIGHT_TO_STEPinit (5)	Local Write
	ROLL_PITCH_TO_STEP (22)	Local Write
	ROLL_PITCH_TO_STEP (23)	Local Write
I5	ROLL_PITCH_TO_STEP (24)	Local Write
	WAVE_V3 (20)	Local Write
	WAVE_V3 (39)	Local Write
JOGM3	WAVE_V3 (60)	Local Write
	MANUAL (11)	Global Read
	_global_init (3)	Global Write
JOGM1	ROLL_PITCH_TO_STEP (16)	Global Read
	_global_init (5)	Global Write
	MANUAL (36)	Global Read
input_run	_global_init (6)	Global Write
	MANUAL (16)	Global Read
	_global_init (2)	Global Write
Input_Roll%MW1	PLC_PRG (3)	Global Read
	PLC_PRG (7)	Global Read
	PLC_PRG (11)	Global Read
Input_Roll%MW1	PLC_PRG (16)	Global Read
	_global_init (32)	Global Write
	INPUT_MODE (1)	Local Read
Input_Pitch%MW0	_global_init (1)	Local Write
	INPUT_MODE (1)	Local Read
	_global_init (1)	Local Write
In_data_6%IW2420	INPUT_MODE (2)	Local Read
	_global_init (2)	Local Write
	READ_POS (10)	Local Read

In_data_6%IW2420		
READ_POS (10)	Local Read	
In_data_5%IW2419		
READ_POS (10)	Local Read	
In_data_5%IW2419		
READ_POS (10)	Local Read	
INPUT_pitch_real		
INPUT_MODE (8)	Local Write	
INPUT_MODE (11)	Local Write	
INPUT_MODE (13)	Local Read	
INPUT_MODE (20)	Local Read	
_global_init (6)	Local Write	
In_data_4%IW2413		
READ_POS (6)	Local Read	
In_data_4%IW2413		
READ_POS (6)	Local Read	
In_data_3%IW2412		
READ_POS (6)	Local Read	
In_data_3%IW2412		
READ_POS (6)	Local Read	
In_data_2%IW2406		
READ_POS (2)	Local Read	
In_data_2%IW2406		
READ_POS (2)	Local Read	
Input_Pitch%MW0		
INPUT_MODE (2)	Local Read	
_global_init (2)	Local Write	
In_data_1%IW2405		
READ_POS (2)	Local Read	
In_data_1%IW2405		
READ_POS (2)	Local Read	
HTS3		
ROLL_PITCH_TO_STEP (24)	Local Write	
ROLL_PITCH_TO_STEP (27)	Local Read	
ROLL_PITCH_TO_STEP... (7)	Local Read	
HTS3		
_global_init (12)	Local Read	
WAVE_V3 (60)	Local Write	
WAVE_V3 (67)	Local Read	
WAVE_V3 (70)	Local Read	
HTS2		
ROLL_PITCH_TO_STEP (23)	Local Write	
ROLL_PITCH_TO_STEP (26)	Local Read	
ROLL_PITCH_TO_STEP... (6)	Local Read	
HTS2		
_global_init (11)	Local Read	
WAVE_V3 (39)	Local Write	
WAVE_V3 (66)	Local Read	
WAVE_V3 (69)	Local Read	
HTS1		
ROLL_PITCH_TO_STEP (22)	Local Write	
ROLL_PITCH_TO_STEP (25)	Local Read	
ROLL_PITCH_TO_STEP... (5)	Local Read	
HTS1		
_global_init (10)	Local Read	
WAVE_V3 (20)	Local Write	
WAVE_V3 (65)	Local Read	
WAVE_V3 (68)	Local Read	
h		
HIGHT_TO_STEP (1)	Local Write	
HIGHT_TO_STEP (2)	Local Read	
HIGHT_TO_STEP (3)	Local Write	
HIGHT_TO_STEP (5)	Local Read	
HIGHT_TO_STEP (6)	Local Write	
HIGHT_TO_STEP (9)	Local Read	
HIGHT_TO_STEP (10)	Local Read	
HIGHT_TO_STEPinit (4)	Local Write	
GO_UP		
MANUAL (41)	Global Read	
_global_init (8)	Global Write	
GO_DOWN		
MANUAL (48)	Global Read	
_global_init (9)	Global Write	
h		
STEP_TO_HIGHT (4)	Local Write	
STEP_TO_HIGHTinit (4)	Local Write	
CALCULATED_ACTUALS (5)	Local Read	
CALCULATED_ACTUALS (6)	Local Read	

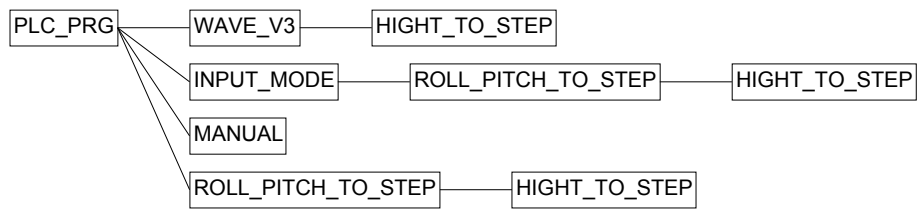
CALCULATED_ACTUALS (7)	Local Read
DOUT2%QX2300.1	
_global_init (35)	Global Write
_global_init (36)	Global Write
DOUT1%QX2300.0	
_global_init (34)	Global Write
_global_init (35)	Global Write
DOUT1%QX2300.0	
_global_init (34)	Global Write
_global_init (35)	Global Write
GO_HOME	
_global_init (10)	Global Write
ENABLE	
M1 (1)	Global Read
M2 (1)	Global Read
M3 (1)	Global Read
_global_init (29)	Global Write
DOUT2%QX2300.1	
_global_init (35)	Global Write
_global_init (36)	Global Write
Data_in_m2_temp	
_global_init (2)	Local Write
READ_POS (6)	Local Write
READ_POS (7)	Local Read
Data_Out	
READ_PARAM (11)	Local Write
_global_init (7)	Local Write
data_in_m3_temp	
_global_init (3)	Local Write
READ_POS (10)	Local Write
READ_POS (11)	Local Read
DATA_IN_M3	
SET_STEP (41)	Global Read
SET_STEP (42)	Global Read
SET_STEP (46)	Global Read
SET_STEP (47)	Global Read
CALCULATED_ACTUALS (3)	Global Read
_global_init (21)	Global Write
READ_POS (11)	Global Write
DATA_IN_M2	
SET_STEP (23)	Global Read
SET_STEP (25)	Global Read
SET_STEP (30)	Global Read
SET_STEP (31)	Global Read
CALCULATED_ACTUALS (2)	Global Read
_global_init (17)	Global Write
READ_POS (7)	Global Write
DATA_IN_M1	
SET_STEP (4)	Global Read
SET_STEP (6)	Global Read
SET_STEP (11)	Global Read
SET_STEP (12)	Global Read
CALCULATED_ACTUALS (1)	Global Read
_global_init (13)	Global Write
READ_POS (3)	Global Write
Data_in_m1_temp	
_global_init (1)	Local Write
READ_POS (2)	Local Write
READ_POS (3)	Local Read
d	
CALCULATED_ACTUALS... (7)	Local Write
CALCULATED_ACTUALS (16)	Local Write
CALCULATED_ACTUALS (17)	Local Read
CA	
_global_init (4)	Local Read
READ_POS (12)	Local Read
READ_POS (14)	Local Read
READ_POS (16)	Local Read
c	
HIGHT_TO_STEPinit (3)	Local Write
b	
HIGHT_TO_STEP (10)	Local Write
HIGHT_TO_STEP (11)	Local Read
HIGHT_TO_STEPinit (1)	Local Write
b	
ROLL_PITCH_TO_STEP (13)	Local Write
ROLL_PITCH_TO_STEP (15)	Local Read
ROLL_PITCH_TO_STEP (16)	Local Read

	ROLL_PITCH_TO_STEP... (2)	Local Write
c	ROLL_PITCH_TO_STEP... (4)	Local Write
c	CALCULATED_ACTUALS... (6)	Local Write
	CALCULATED_ACTUALS (15)	Local Write
	CALCULATED_ACTUALS (16)	Local Read
b	STEP_TO_HIGHT (1)	Local Write
	STEP_TO_HIGHT (2)	Local Read
	STEP_TO_HIGHTinit (1)	Local Write
am	ROLL_PITCH_TO_STEP (13)	Global Read
	ROLL_PITCH_TO_STEP (18)	Global Read
	STEP_TO_HIGHT (3)	Global Read
	CALCULATED_ACTUALS (14)	Global Read
	CALCULATED_ACTUALS (17)	Global Read
	_global_init (14)	Global Write
a_roll	CALCULATED_ACTUALS... (14)	Local Write
	CALCULATED_ACTUALS (14)	Local Write
	READ_POS (16)	Local Read
a_pitch	CALCULATED_ACTUALS... (15)	Local Write
	CALCULATED_ACTUALS (17)	Local Write
	READ_POS (14)	Local Read
angle	HIGHT_TO_STEP (11)	Local Write
	HIGHT_TO_STEP (12)	Local Read
	HIGHT_TO_STEPinit (6)	Local Write
	WAVE_V3 (68)	Local Read
	WAVE_V3 (69)	Local Read
	WAVE_V3 (70)	Local Read
a_hight2	CALCULATED_ACTUALS... (9)	Local Write
	CALCULATED_ACTUALS (6)	Local Write
	CALCULATED_ACTUALS (12)	Local Read
	CALCULATED_ACTUALS (15)	Local Read
a1	STEP_TO_HIGHT (3)	Local Write
	STEP_TO_HIGHT (4)	Local Read
	STEP_TO_HIGHTinit (2)	Local Write
b	CALCULATED_ACTUALS... (5)	Local Write
	CALCULATED_ACTUALS (13)	Local Write
	CALCULATED_ACTUALS (14)	Local Read
a_hight3	CALCULATED_ACTUALS... (10)	Local Write
	CALCULATED_ACTUALS (7)	Local Write
	CALCULATED_ACTUALS (12)	Local Read
	CALCULATED_ACTUALS (15)	Local Read
a_angle3	CALCULATED_ACTUALS... (13)	Local Write
	CALCULATED_ACTUALS (10)	Local Write
a_hight1	CALCULATED_ACTUALS... (8)	Local Write
	CALCULATED_ACTUALS (5)	Local Write
	CALCULATED_ACTUALS (16)	Local Read
a_angle2	CALCULATED_ACTUALS... (12)	Local Write
	CALCULATED_ACTUALS (9)	Local Write
a_angle1	CALCULATED_ACTUALS... (11)	Local Write
	CALCULATED_ACTUALS (8)	Local Write
a	HIGHT_TO_STEP (9)	Local Write
	HIGHT_TO_STEP (11)	Local Read
	HIGHT_TO_STEPinit (2)	Local Write
a	ROLL_PITCH_TO_STEP (18)	Local Write
	ROLL_PITCH_TO_STEP (19)	Local Read
	ROLL_PITCH_TO_STEP (20)	Local Read
	ROLL_PITCH_TO_STEP... (1)	Local Write
a	STEP_TO_HIGHT (2)	Local Write
	STEP_TO_HIGHT (3)	Local Read
	STEP_TO_HIGHTinit (5)	Local Write
	CALCULATED_ACTUALS (8)	Local Read

a

CALCULATED_ACTUALS (9)	Local Read
CALCULATED_ACTUALS (10)	Local Read
CALCULATED_ACTUALS... (4)	Local Write
CALCULATED_ACTUALS (12)	Local Write
CALCULATED_ACTUALS (13)	Local Read

Call Tree of PLC\_PRG (PRG-ST)



	Page
Project information	A
CALCULATED_ACTUALS (FB-ST)	1
HIGHT_TO_STEP (FB-ST)	2
ROLL_PITCH_TO_STEP (FB-ST)	2
STEP_TO_HIGHT (FB-ST)	3
INPUT_MODE (PRG-ST)	3
MANUAL (PRG-ST)	4
WAVE_V3 (PRG-ST)	5
Read_Param (PRG-ST)	6
Read_Pos (PRG-ST)	6
M1 (FUN-ST)	7
M2 (FUN-ST)	7
M3 (FUN-ST)	7
PLC_PRG (PRG-ST)	7
SET_STEP (PRG-ST)	8
Graph	9
HOME	10
Manual_mode	10
UDP	11
Wave_mode	11
Global_Variables	11
Variable_Configuration	12
Globale_Variablen	12
CAM Data	12
CAM Data	13
CNC Data	13
Drive Configuration Data	13
Global Variables 0	13
Globale_Variablen	13
Version	13
Alarm configuration	13
PLC Configuration	13
Sampling Trace	25
Task configuration	25
Watch- and Recipe Manager	25
Workspace	25
Parameter Manager	25
Cross Reference List	25
Call Tree of PLC_PRG (PRG-ST)	36