NexRobotEdu Introduction

Dec., 2015

NexRobotEdu Ver. 1.0.0.4

Agenda

- **NexRobotEdu Introduction**
  - Introduction
  - Robot Arm Hardware Spec.
  - Controller Setup
  - Software Setup
  - Architecture
  - Utility Operation
  - Origin Calibration
- **NexRobotEdu Example Codes**
  - 1.HMI and Host Service(Shared Memory)
  - 2.NexRobotEdu API introduction
  - 3.User RT Application
  - 4.Position、Velocity、Torque control example
- **NexRobotEdu – Trouble Shooting**

NexRobotEdu – Introduction

NexRobotEdu – Introduction - 1

- **Using NEXCOM Self-Development**
  - Robot Control Software Development Kit
  - NexECM
  - NET3600(IPC)
- **Run In Real-Time Extension(RTX)System**
  - RTX2012

NexRobotEdu – Introduction - 2

- **Robot Arm**
  - Mode Name
    - RA605
  - Drive/Motor
    - PANASONIC A5B EtherCAT Servo Drive
- **Digital IO(Slave)**
  - Using NEXCOM Self-Development
    - AXE9200 (Mounted On Controller)

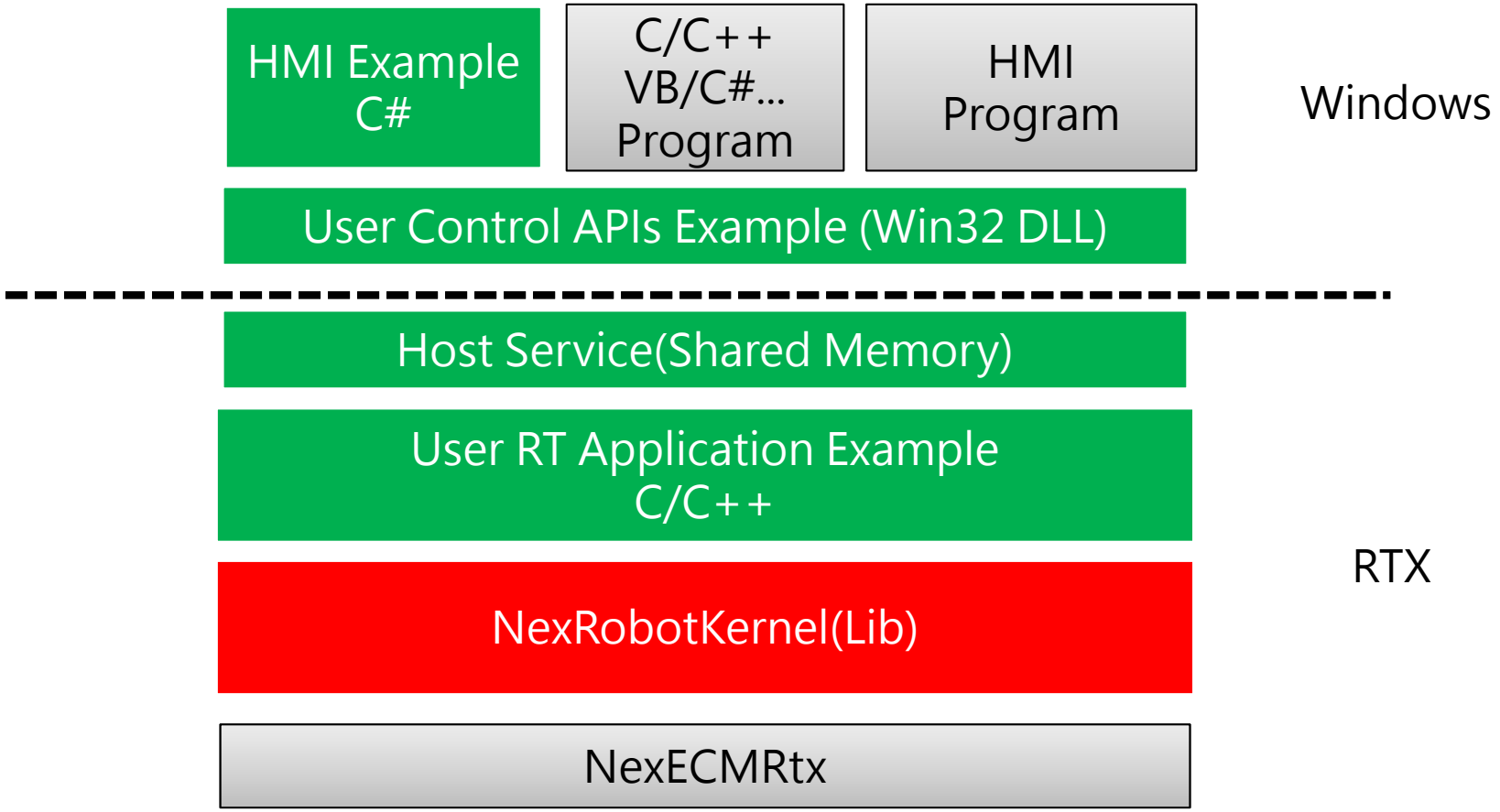
NexRobotEdu – Introduction - 3

Spec.

- Control Cycle Time
  - 1ms
- Control Mode
  - Synchronize Cyclic Position Mode (1ms)
  - Synchronize Cyclic Velocity Mode (1ms)
  - Synchronize Cyclic Torque Mode (1ms)
- Built-In RA605 Kinematics
- Providing Utility & Example(Source Codes)

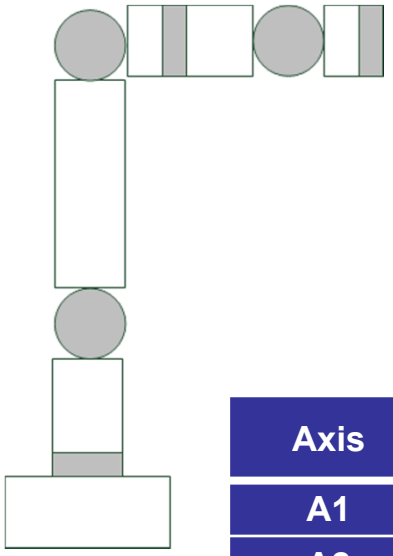
NexRobotEdu – Introduction - 4

System Block Diagram

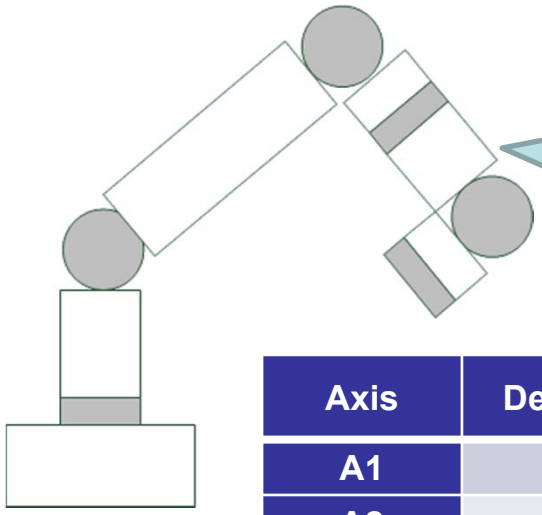


♦ Robot Arm Hardware Spec.

Robot Arm Posture Information



Axis	Degree
A1	0
A2	90
A3	0
A4	0
A5	0
A6	0

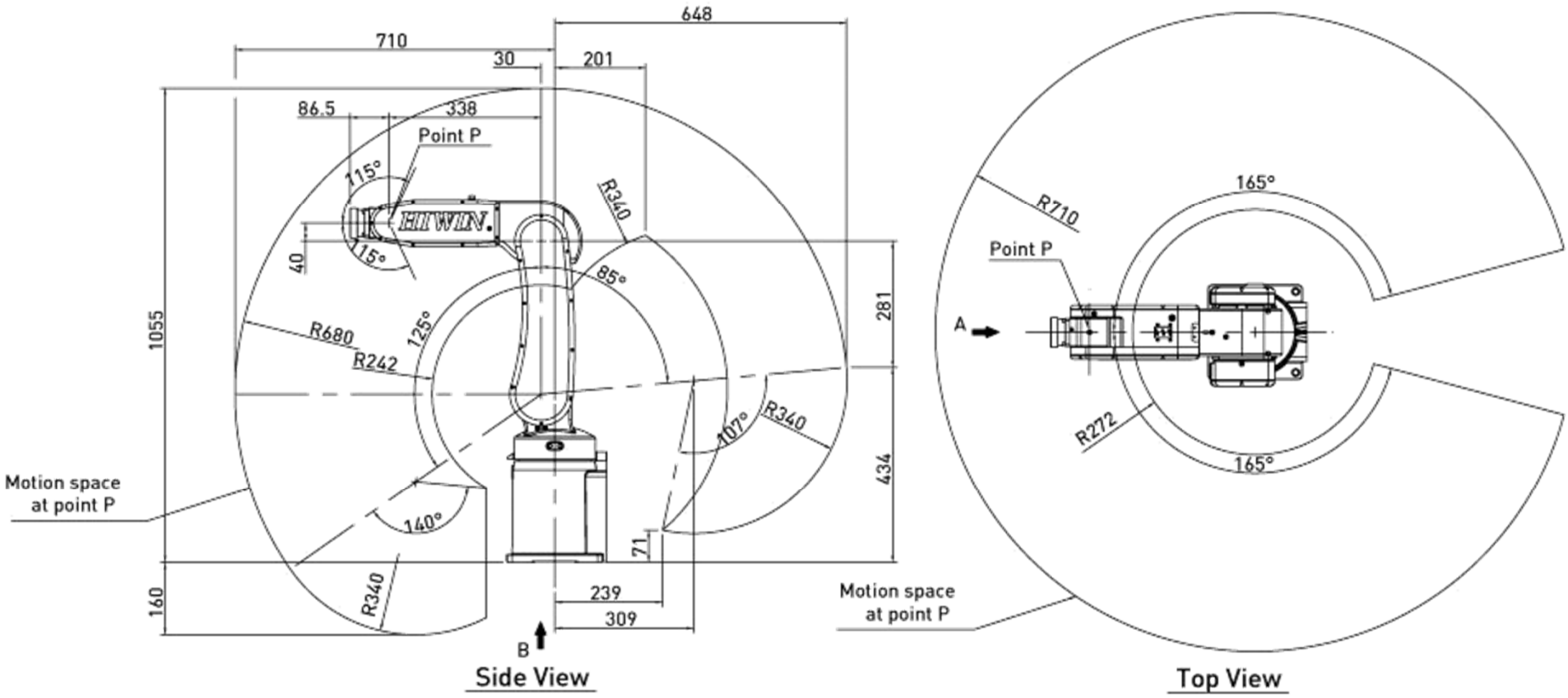


If A2=0
It possible hit the table!!

Axis	Degree
A1	0
A2	45
A3	0
A4	0
A5	-90
A6	0

Note	❖ The directions of six joints refer to label on robot arm.

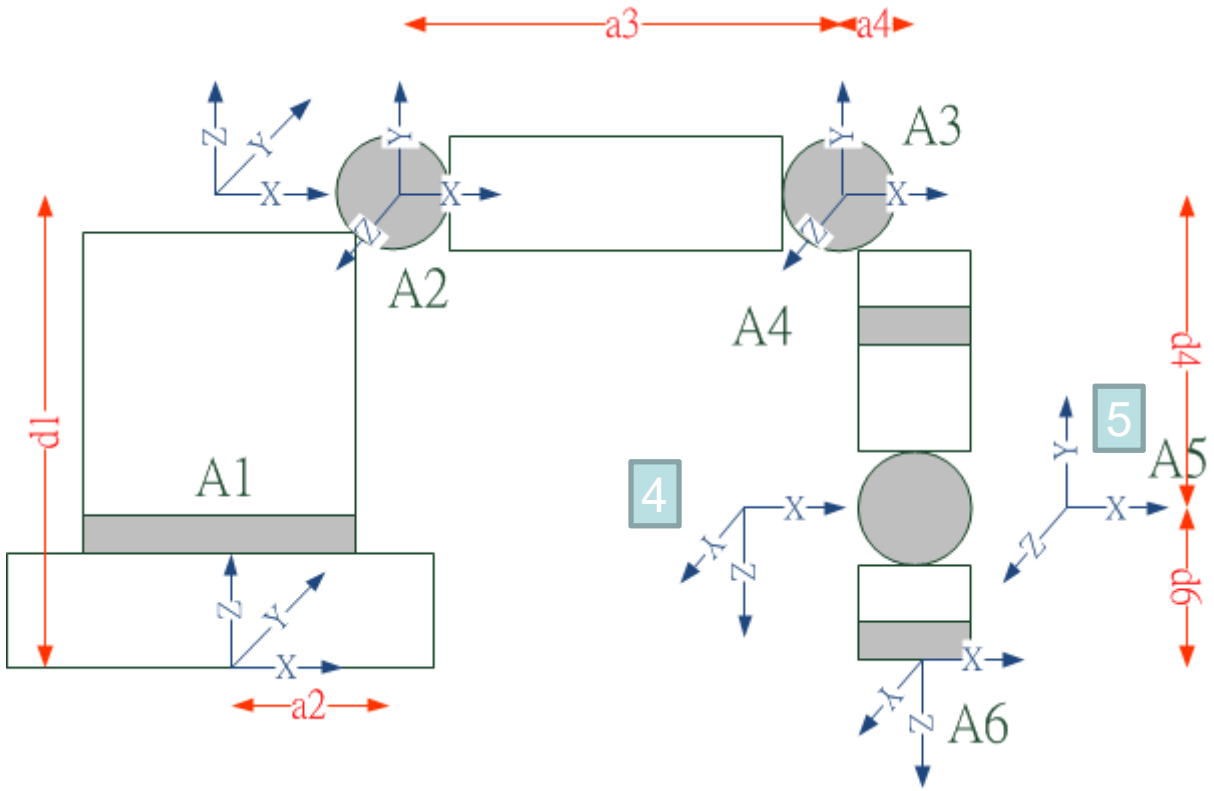
Robot Arm Operation Range



Robot Arm Spec.

Spec.		Value	Watts
Degree		6	
Rated Load		5 kg	
Operating Range	Maximum Reached Radius	710 mm (Point P)	
	J1	±165°	400W
	J2	+85° ~ -125°	400W
	J3	+185° ~ -55°	200W
	J4	±190°	100W
	J5	±115°	100W
	J6	±360°	100W
Weights		40kg	

Kinematics

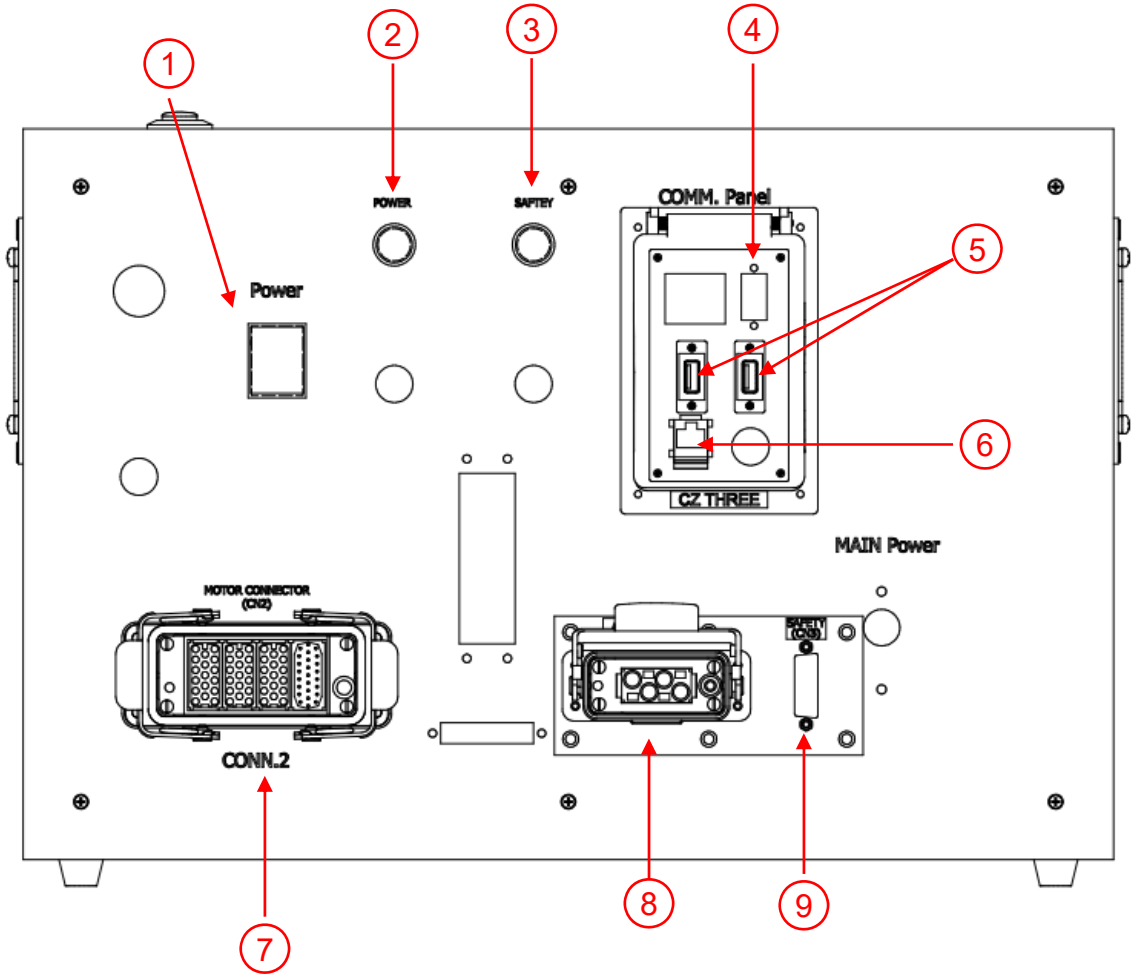


D – H Table				
Joint	a_i	α_i	d_i	θ_i
1	0	0	d1	θ1
2	a2	π/2	0	θ2
3	a3	0	0	θ3
4	a4	π/2	d4	θ4
5	0	-π/2	0	θ5
6	0	π/2	d6	θ6

NexRobotEdu – Controller Setup

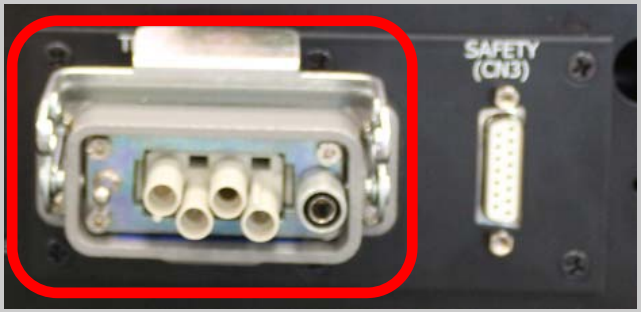
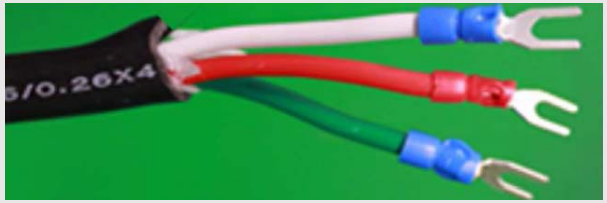
NexRobotEdu - Controller Setup - 1

NexRobotEdu - Controller Setup - 2



N	Name	Function Description
1	Power Switch	Switch Power ON/OFF
2	Power Light	Controller Is Power ON Or Not
3	Safety Light	Safety Enable Status
4	VGA Port	VGA Signal
5	USB Port	USB Signal
6	LAN Port	EtherCAT Signal
7	Motor Connector	Connect to Robot Arm
8	Power Connector	Power 1ØAC220 Input
9	Safety Port	Connect to External Emergency Stop

NexRobotEdu - Controller Setup - 3

Power Cable	
Power connector showed as right figure, it can be inserted by power cable.	
Power cable has three wires: red, white, green(1Ø220V). Please make sure the green wire connects to ground.	

NexRobotEdu - Controller Setup - 4

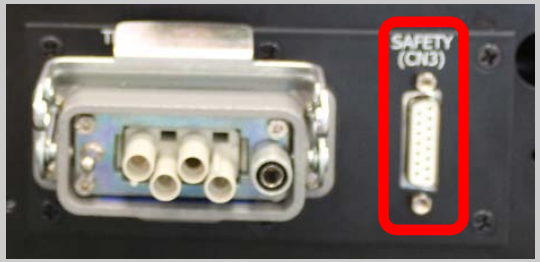
Power Cable

Danger	❖ Do not insert power cable when power is ON. ❖ Do not soak the Product to water, corrosive or flammable gases, and combustibles。 ❖ Do not use cables soaked in water or oil. ❖ Do not attempt to carry out wiring or manual operation with wet hand.
Caution	❖ Do not remove power cable when robot arm or controller is running.
Note	❖ Power cable input is 1ØAC220V. ❖ Power cable needs to grounded.

NexRobotEdu - Controller Setup - 5

Motor Power & Encoder Cable	
Motor power & encoder connector Showed as right figure, it can be inserted by motor power & encoder cable.	

NexRobotEdu - Controller Setup - 6

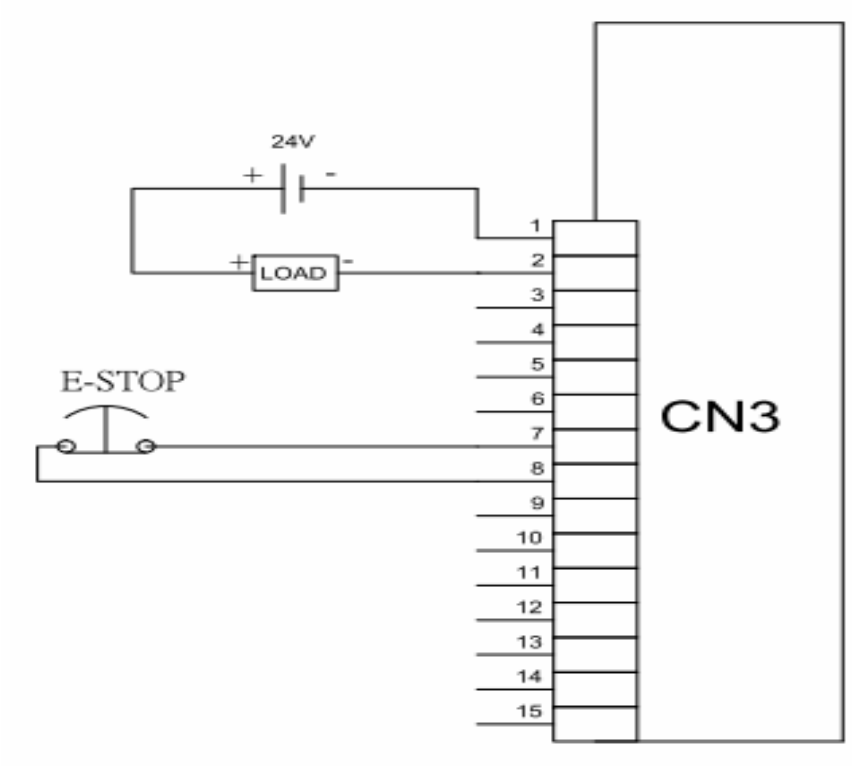
External Emergency Stop	
External emergency stop connector is showed as right figure, it can be inserted by External emergency stop.	
Before robot arm start running, be sure external emergency stop is under reset status.	

NexRobotEdu - Controller Setup - 7

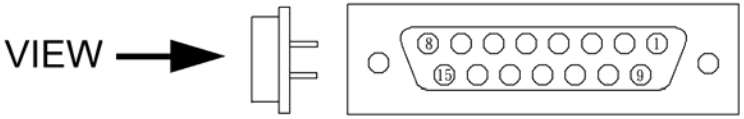
External Emergency Stop

Danger	❖ In order to ensure the safety of human and robot arm, external Emergency Stop must be reachable.
Note	❖ Before robot arm start running, be sure external emergency stop is under reset status.

NexRobotEdu - Controller Setup - 8



remark:
The pin 7 & 8 of external emergency stop is short-circuited at normal status; if status is open-circuited, it will trigger emergency stop signal to controller.

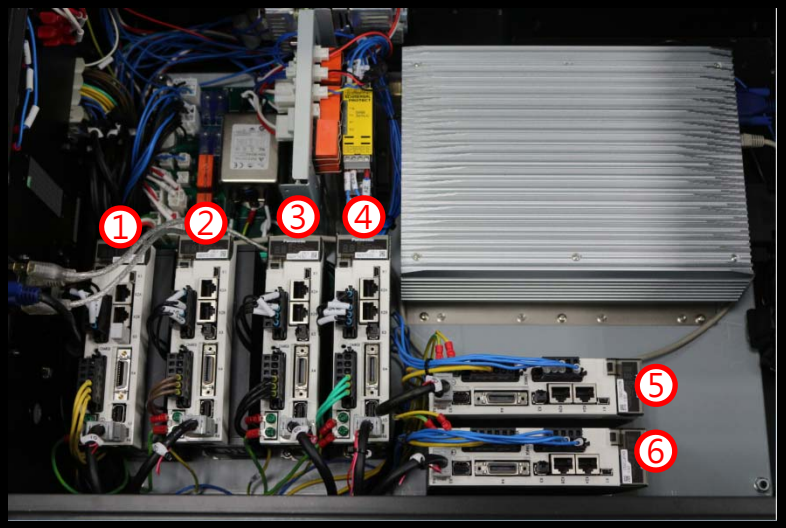


CN3 Safety

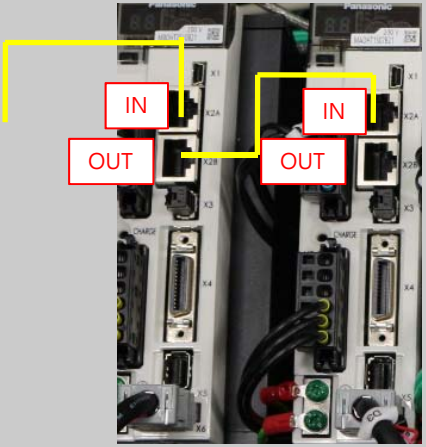
Note	❖ Before robot arm start running, please make sure the emergency stop is locked to connector.

NexRobotEdu - Controller Setup - 9

6 Servo Drives

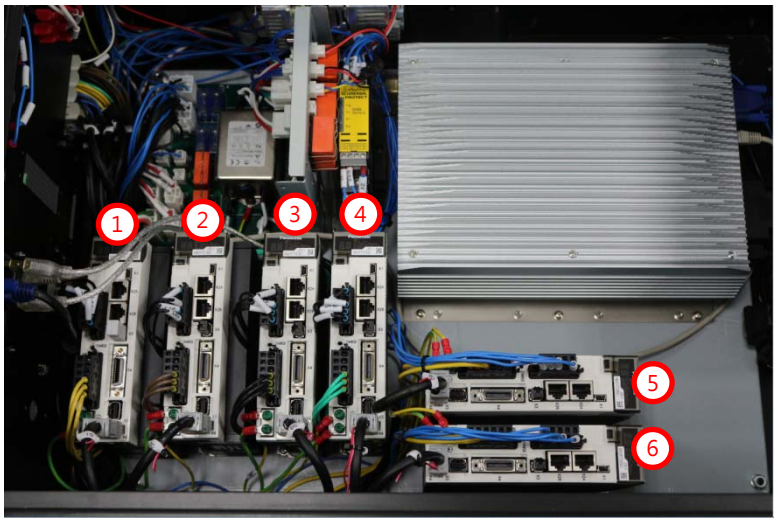


Input & Output of EtherCAT on Servo Drive is showed as right figure.



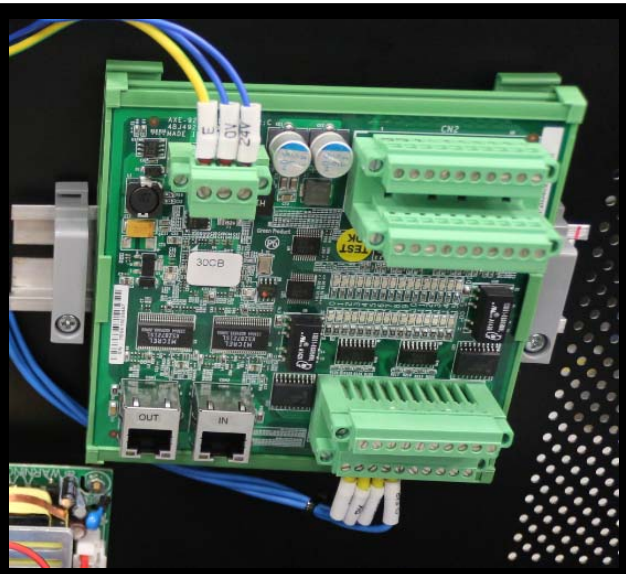
NexRobotEdu - Controller Setup - 10

6 Servo Drives



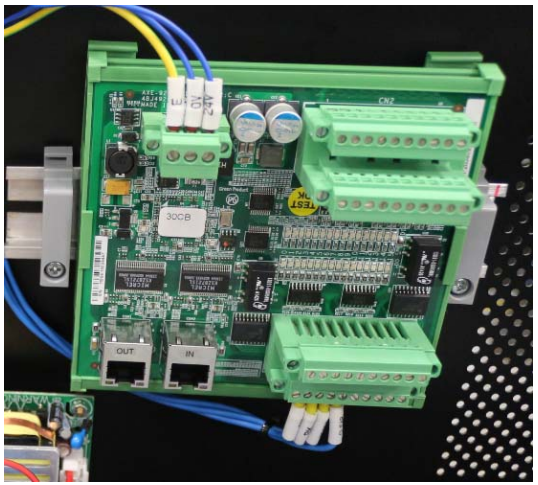
Note	❖ Use RJ45 cable link from port2 of IPC to the first servo drive. ❖ Be sure the RJ45 cable are connected from the first to the sixth servo drive sequentially.

NexRobotEdu - Controller Setup - 11

Digital I/O AXE9200	
System Default DO Pin	❖ DO[0] : Reset Enable ❖ DO[1] : Motor Brake ❖ DO[2] : Driver Clear

NexRobotEdu - Controller Setup - 12

Digital I/O
AXE9200

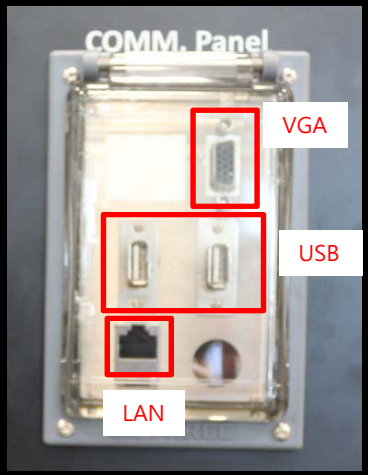


Note

- In order to make sure system can be excused normally, after controller first power on or external emergency stop, utility must be sent DO[0] signal to AXE9200.
- The 1st , 2nd and 3rd joint have a brake, the 4th , 5th and 6th joint have no brake. If robot arm needs to run, utility should be sent DO[1] signal to untie brakes.
- The RJ45 cable must be connected from sixth servo drive to AXE9200.

NexRobotEdu - Controller Setup - 13

Other Ports



There are one VGA port, one LAN port and two USB ports on controller side.

NexRobotEdu - Controller Setup - 14

Power Switch
(Switch ON is power on IPC and Servo Drive)

NexRobotEdu - Software Setup

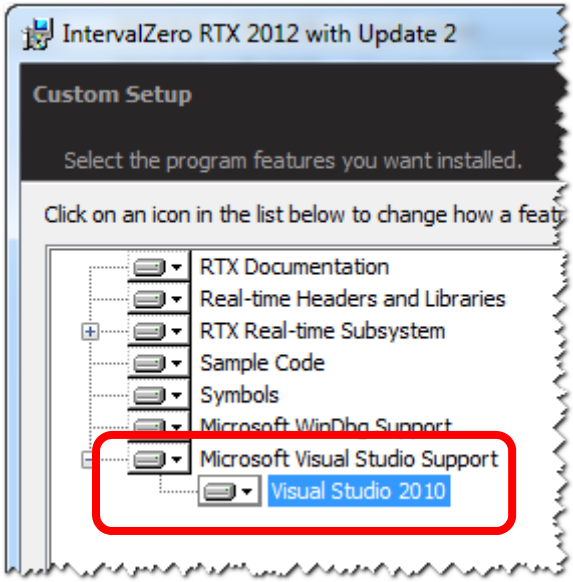
NexRobotEdu - Software Setup

Setup Process

1) Visual studio 2010 (VS)
2) RTX 2012 update3
3) WinPcap
4) NexRobotEdu

Remark

- VS→ RTX



WinPcap - Setup

♦ Setup Process

- Use default

NexRobotEdu - Setup

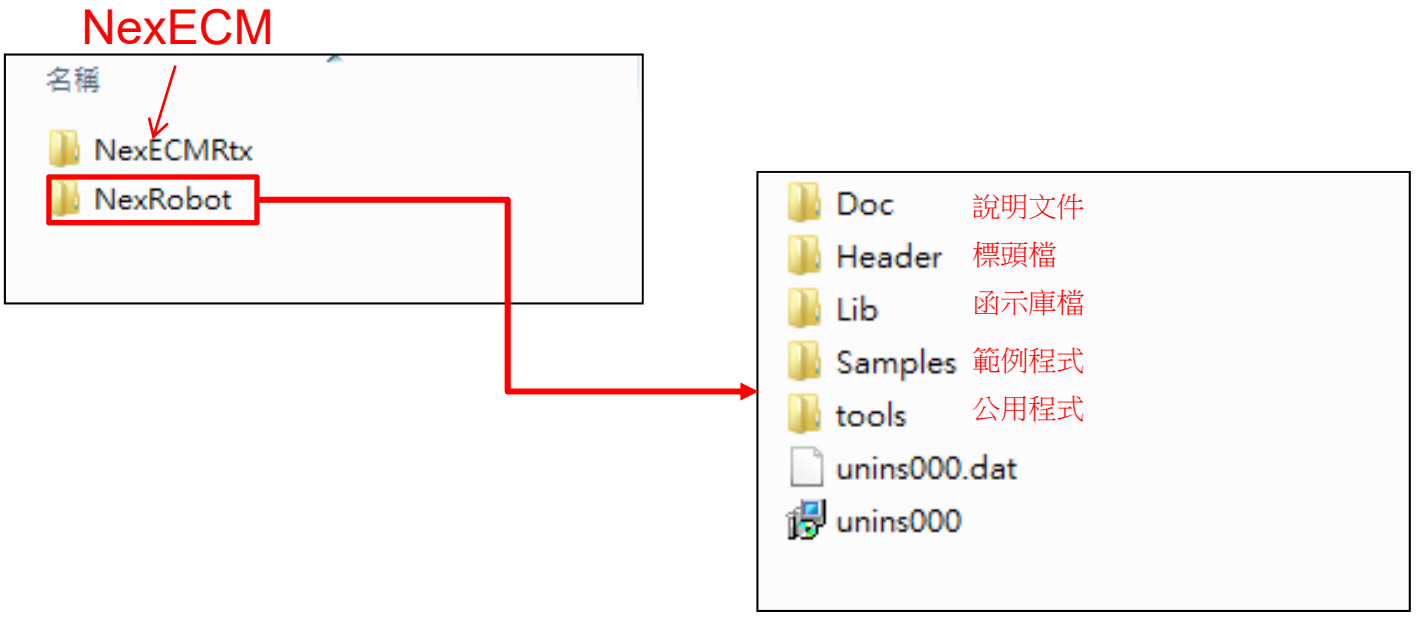
♦ Setup Process

- Use default

NexRobotEdu Architecture - Introduction

NexRobotEdu Architecture -1

- After install NexRobotEdu, the install folder is located at:
  C:\Program Files\NEXCOM\NexRobot\



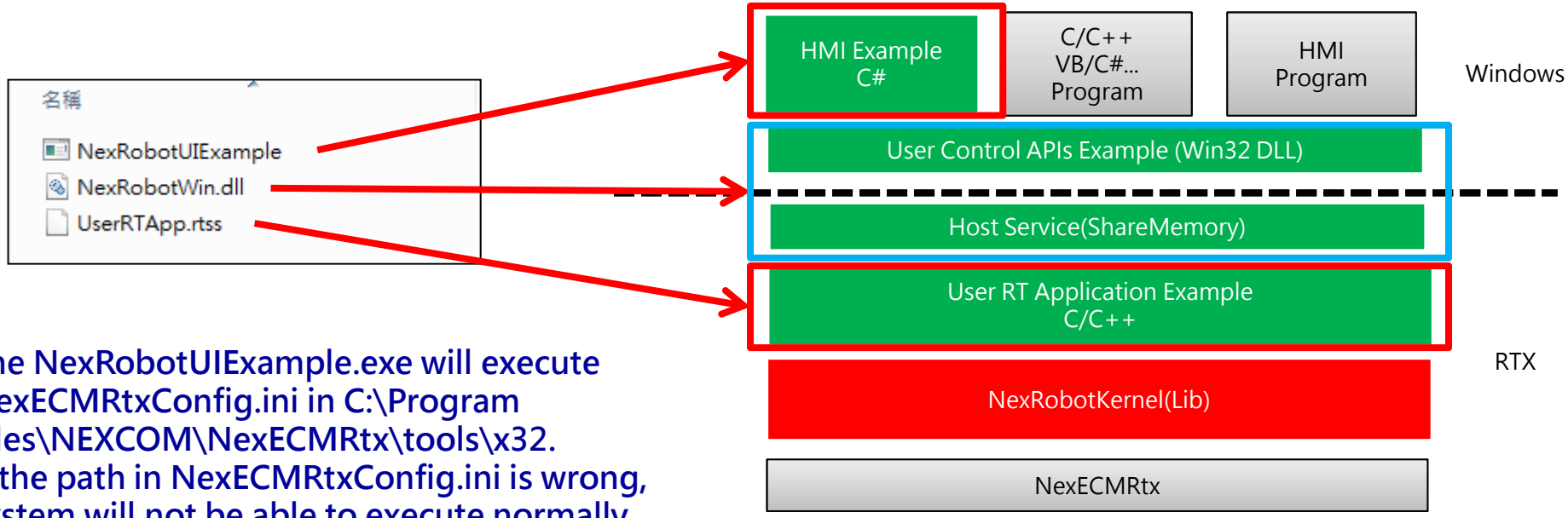
NexRobotEdu Architecture - 2

- After installation, these 3 new files in root directory of C disk as follows, and setting are completed automatically.

檔案名稱	格式	說明
NexRobotEdu	xml	NexRobot EtherCAT Network Information File
Robot_6R_Machine_Data	dat	Mastering Data of Robot Arm
RtxEcNic	ini	NexECM Nic Configuration File

NexRobotEdu Architecture - 3

- After installation, the utility is at C:\Program Files\NEXCOM\NexRobot\tools\x32.



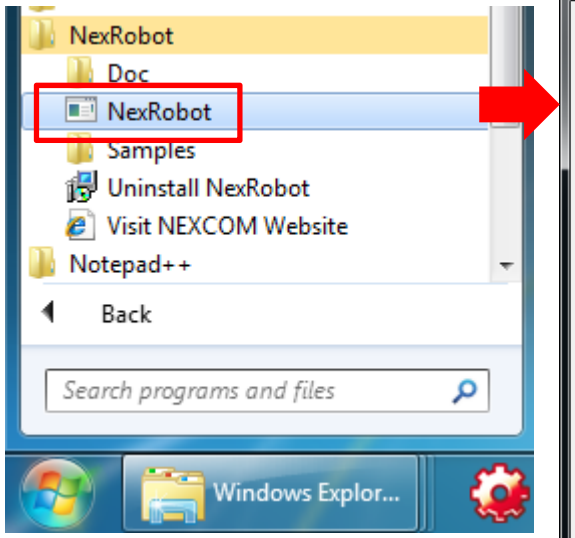
The NexRobotUIExample.exe will execute NexECMRtxConfig.ini in C:\Program Files\NEXCOM\NexECMRtx\tools\x32.
If the path in NexECMRtxConfig.ini is wrong, system will not be able to execute normally.

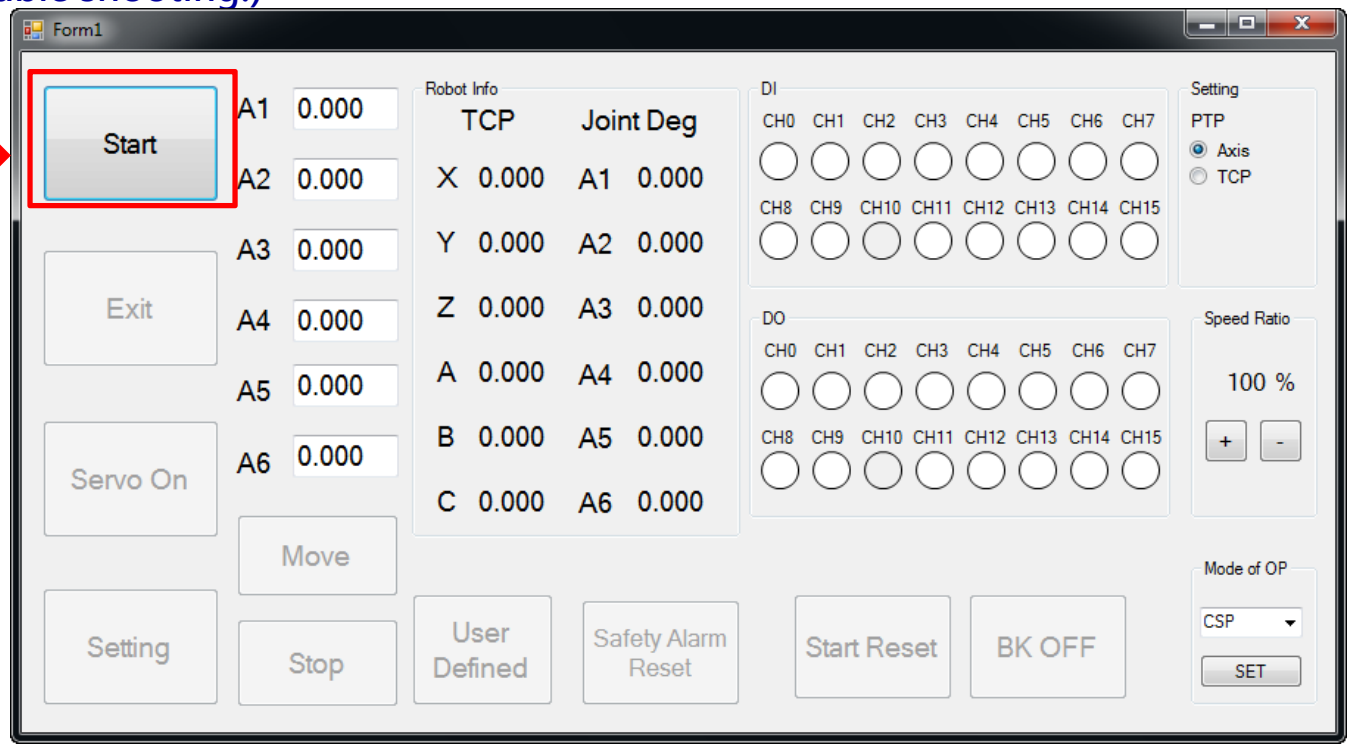
```
[PATH_ENI]
PATH = C:\NexRobotEdu.xml
OPTION = 0

[PATH_NEXECMRTX_DRIVER]
PATH = "C:\Program Files\NEXCOM\NexECMRtx\Lib\NexECMRtx\x32\NexECMRtx.rtss"

[PATH_USER_APP]
PATH = "C:\Program Files\NEXCOM\NexRobot\tools\x32\UserRTApp.rtss"
```

NexRobotEdu Utility - Operation

NexRobotEdu Utility – Operation 1

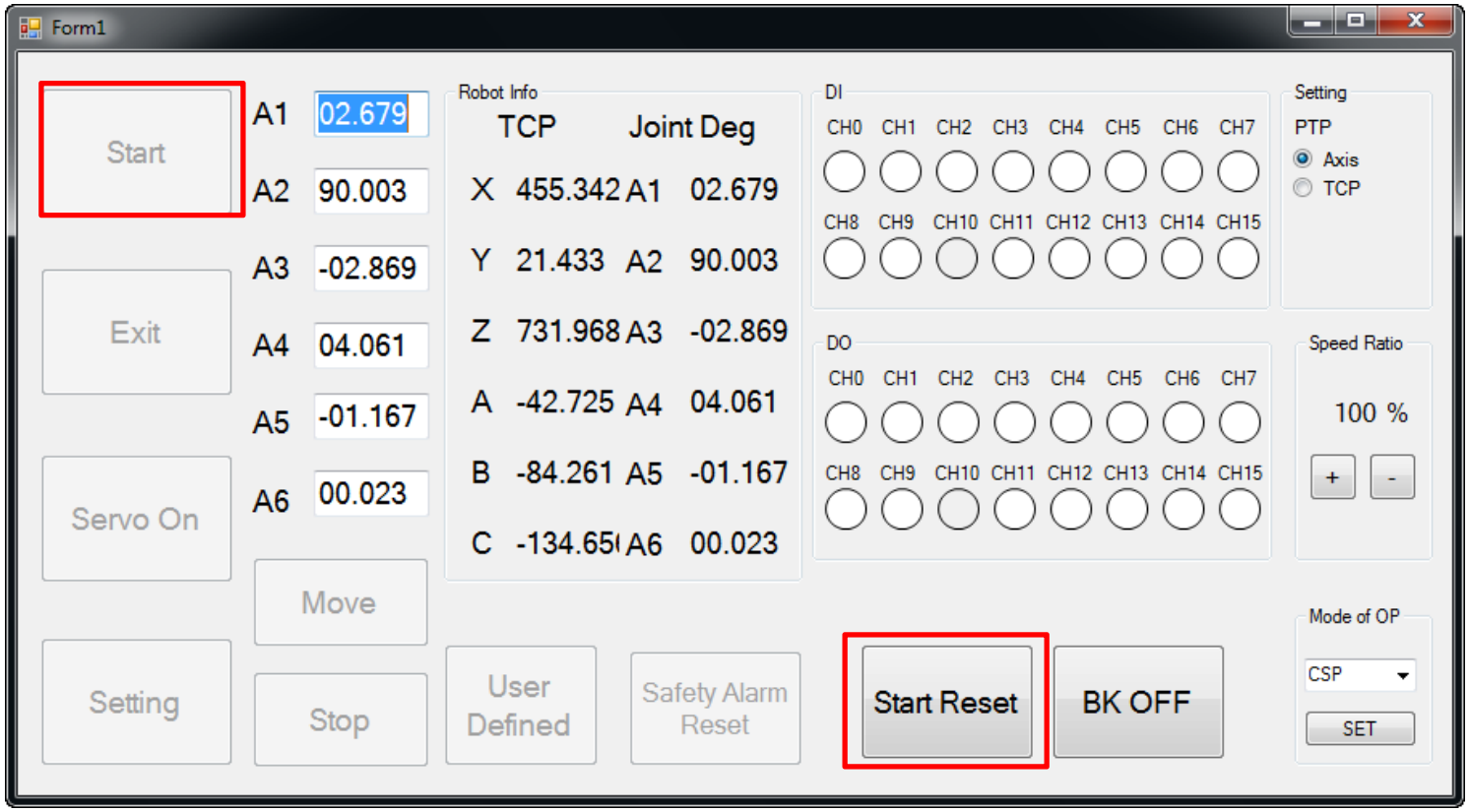
- After installation, from 『Start > NexRobot > NexRobot』 to execute utility.
- Click 『Start』 bottom to start NexRobotEdu system. (if you can not start successfully, please refer to trouble shooting.)





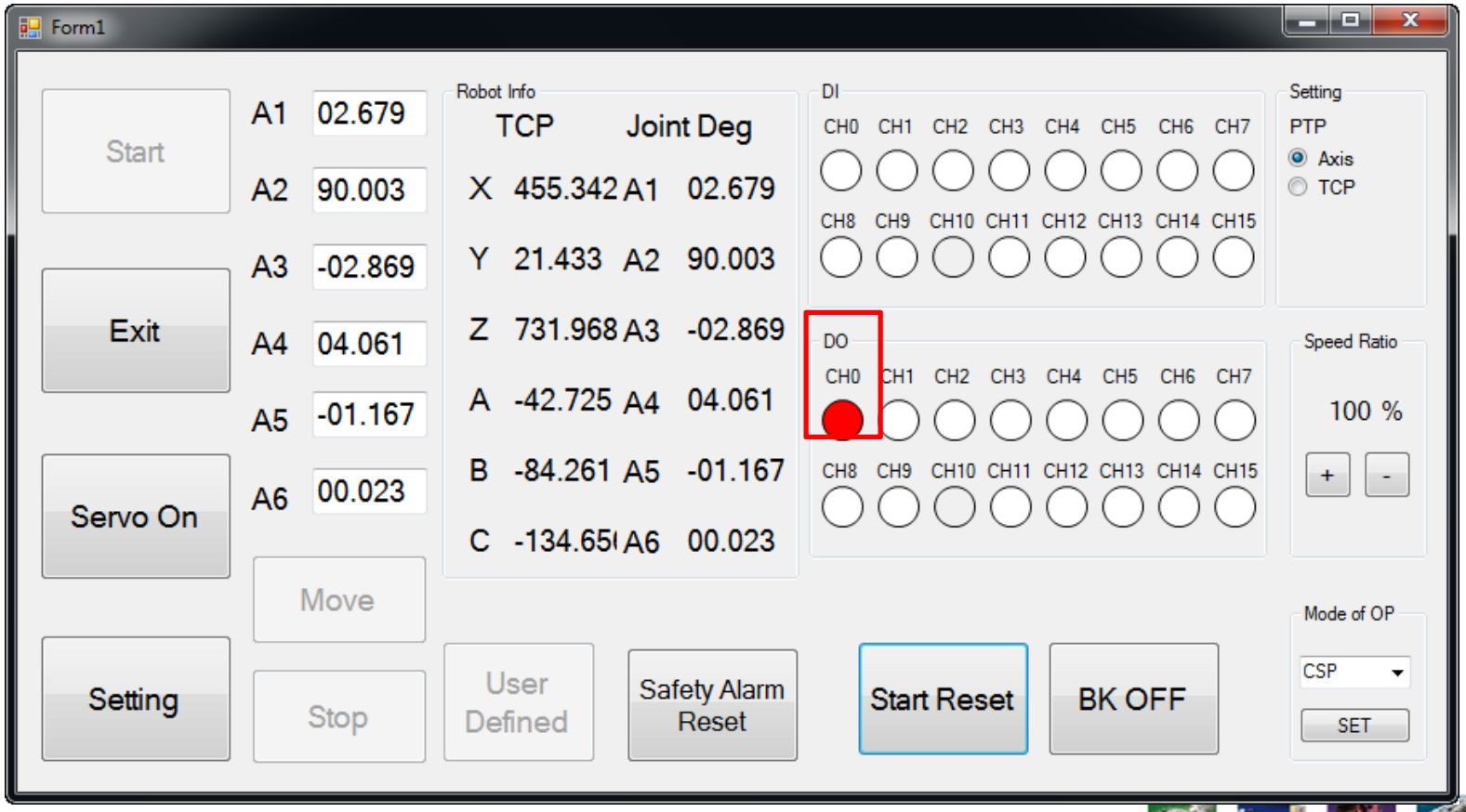
NexRobotEdu Utility - Operation 2

- After start successfully, 『Start』 bottom will be disabled, 『Start Reset』 bottom will be enabled.



NexRobotEdu Utility - Operation 3

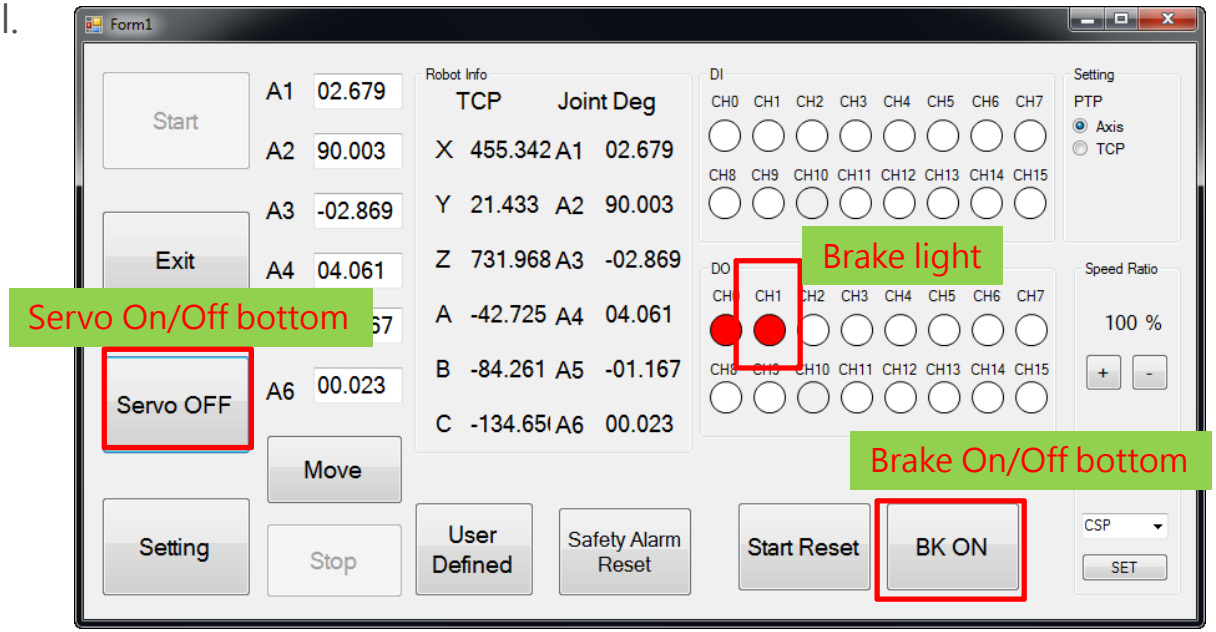
- After click 『Start Reset』 bottom, utility send DO[0] signal to AXE9200, system can operate normally. (Enable safety relay)
- At this time, the rest of bottoms will be enabled.



 Note	❖ Before using NEXCOM PTP function, Please execute Origin Calibration. (Refer to Robot Arm Origin Calibration」)

NexRobotEdu Utility - Operation 4

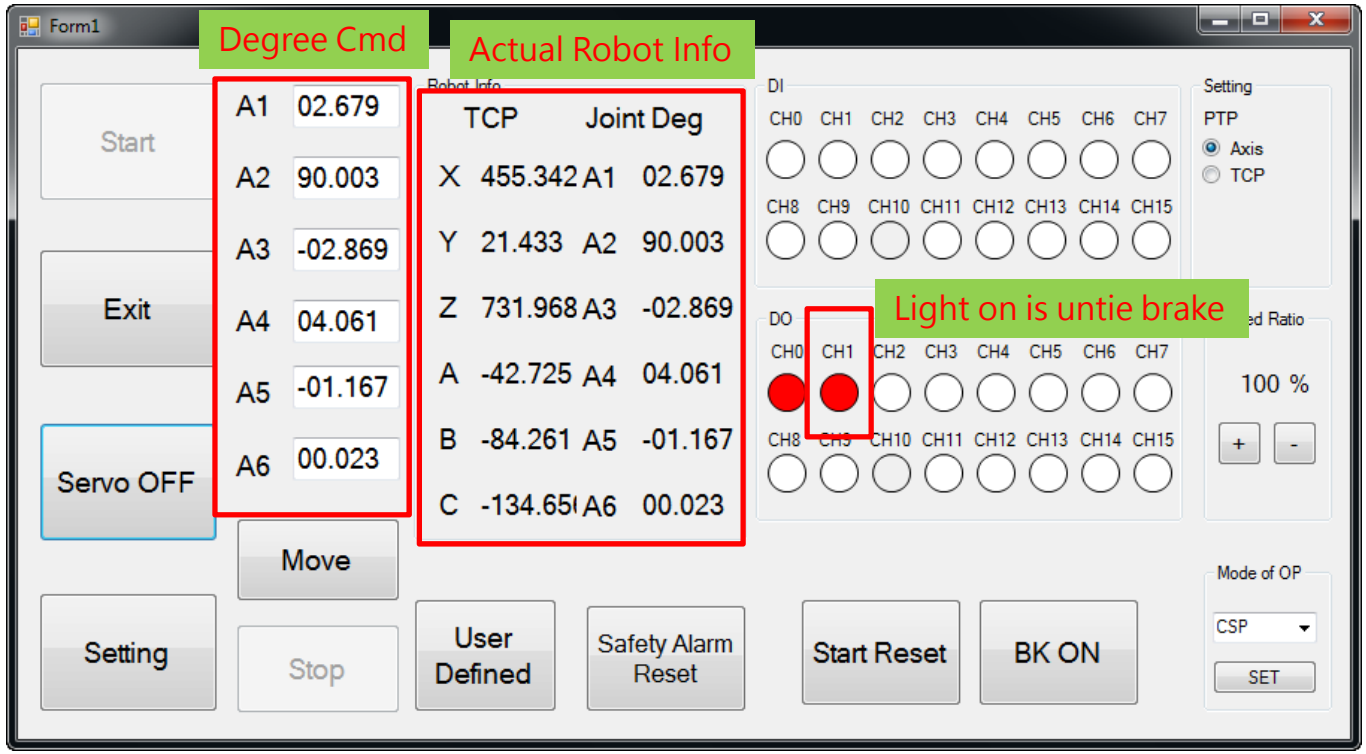
- **Servo On Process**
  - Click 『Servo On』 bottom, then click 『BK OFF』 bottom and robot can be moved now.
- **Servo Off Process**
  - Click 『BK ON』 bottom, then click 『Servo Off』 bottom and robot arm can be kept still.



Danger	❖ Before executing PTP, Please make sure nobody is nearby the working area of robot arm. ❖ In order to ensure the safety of human and robot arm, external Emergency Stop must be reachable.

NexRobotEdu Utility - Operation 5

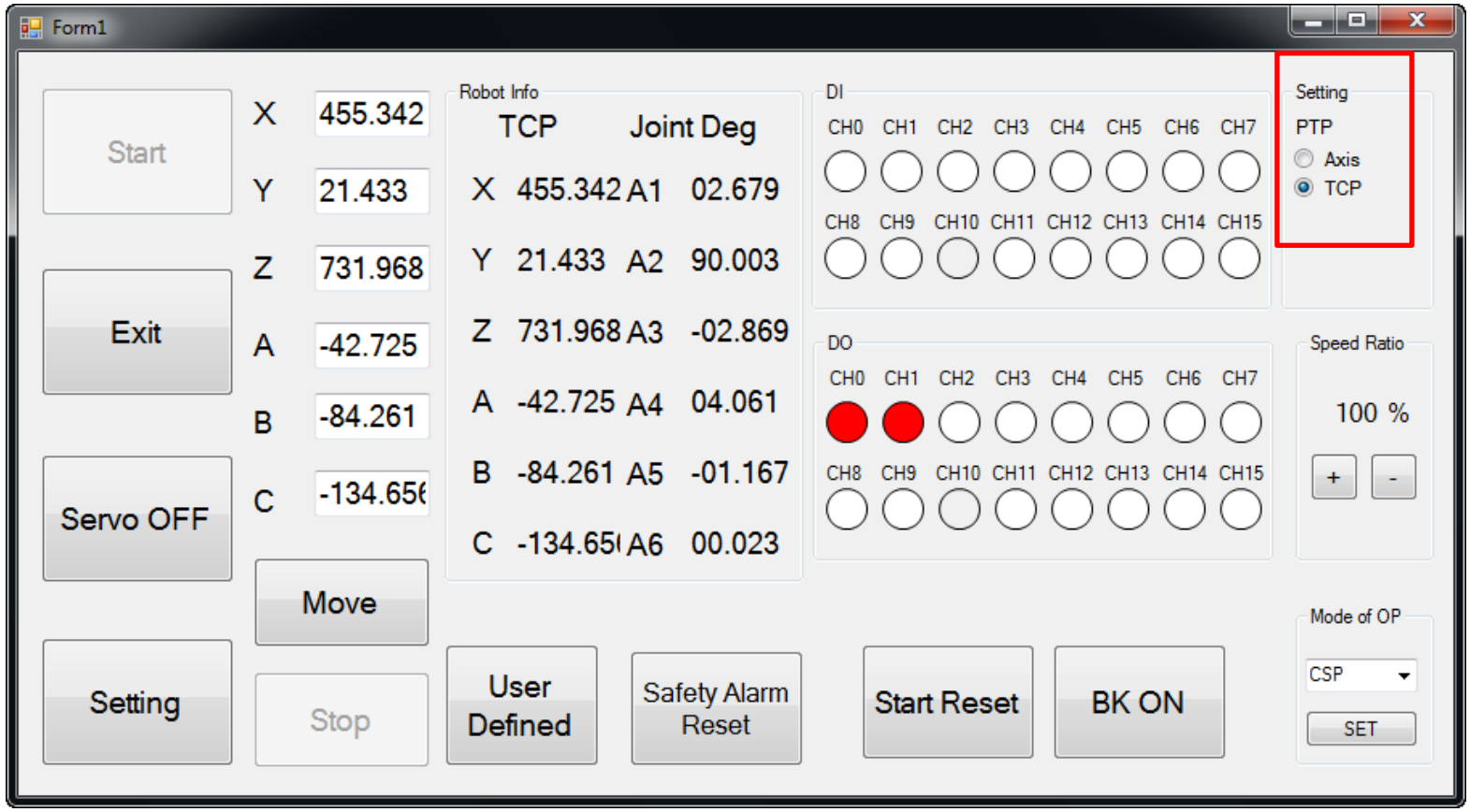
- After Servo On and untie brake, now can start to execute NEXCOM PTP function to move point to point.



Danger	❖ Before executing PTP, Please make sure nobody is nearby the working area of robot arm. ❖ In order to ensure the safety of human and robot arm, external Emergency Stop must be reachable.

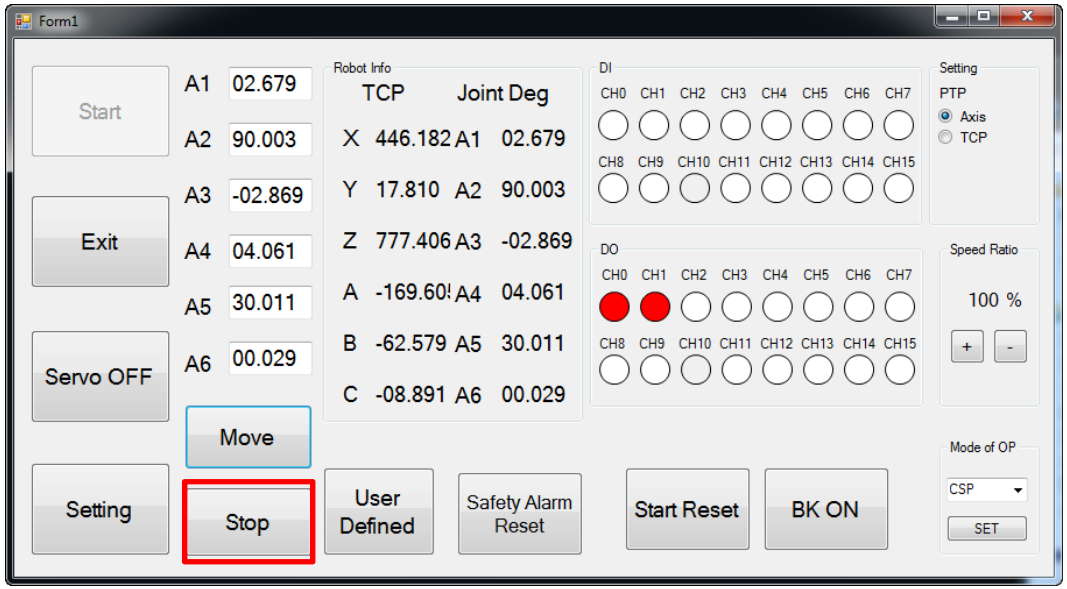
NexRobotEdu Utility - Operation 6

- **PTP has two method to execute :**
  - (1) AXIS
  - (2) TCP (Tool Center Point)



NexRobotEdu Utility - Operation 7

- If executed NEXCOM PTP function to move point to point, robot arm has no response, it could be caused by the following conditions:
  - Servo On failed (Please refer to Trouble Shooting)
  - Exceeds limit of robot arm (Please refer to Robot Arm Spec.)
- If executed NEXCOM PTP function to move point to point, press 『Stop』 bottom can stop this move.

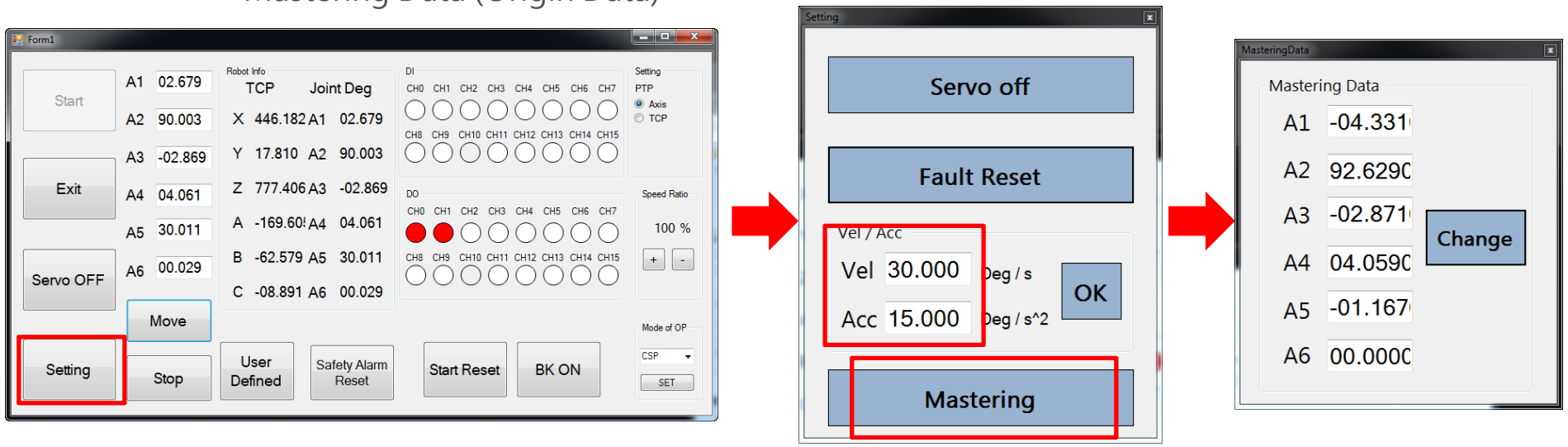


Note

- Before PTP, please make sure nobody is nearby the working area of robot arm.
- Before PTP, please make sure robot arm will not hit surrounding obstacles.
- Before PTP, please make sure robot arm has executed Origin Calibration.

NexRobotEdu Utility - Operation 8

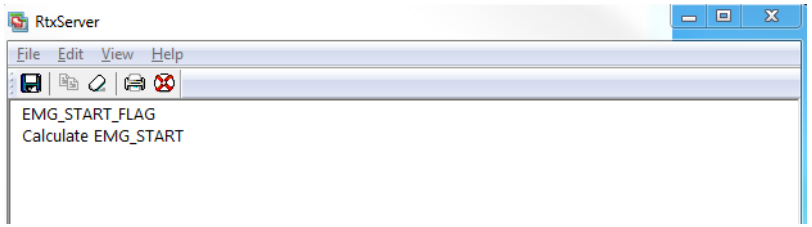
- **Utility can set following parameters:**
  - PTP velocity (deg/sec) and acceleration (deg/sec^2)
  - Mastering Data (Origin Data)



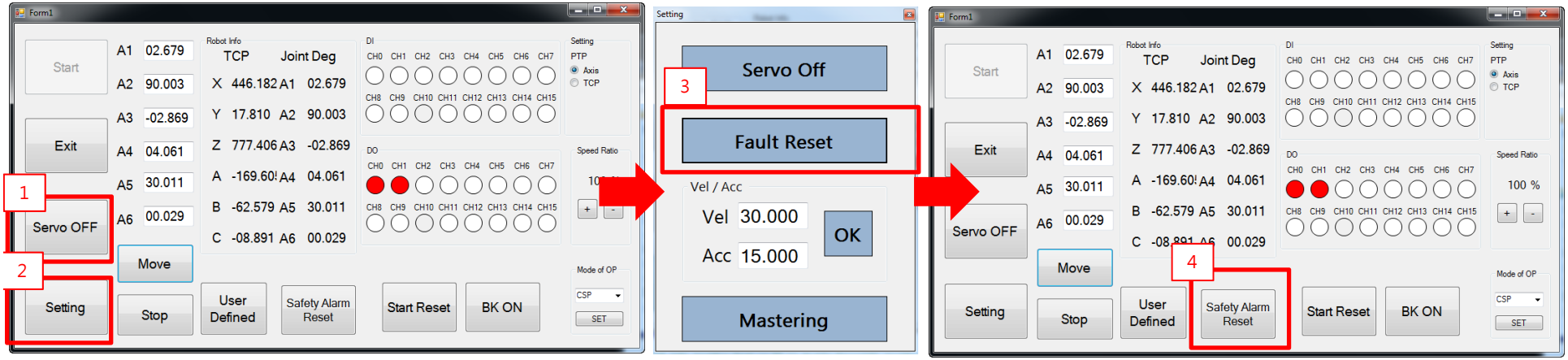
注意	❖ If the velocity and acceleration value are setted to high, it will be stopped by safety of servo drives.

NexRobotEdu Utility - Operation 9

- **How to reset alarm of servo drive**
  - If there is any alarm in servo drive, PTP will stop automatically. At this time, the RtxServer will print message as following figure.



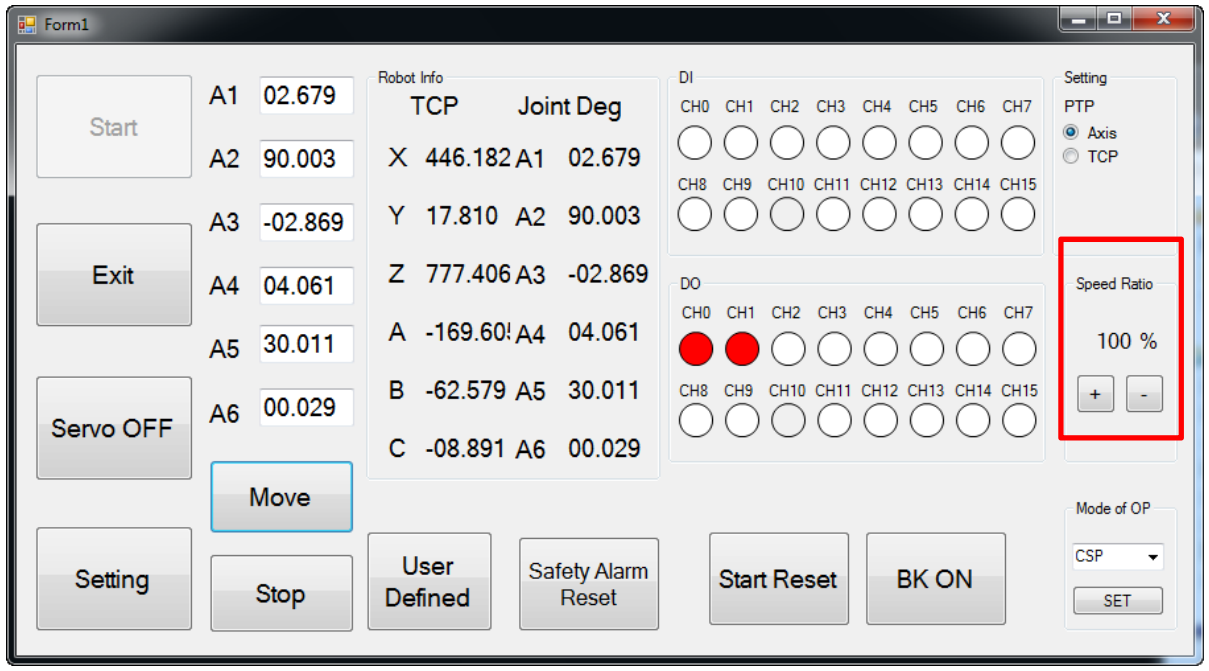
  - Reset Process : (1)Servo Off (2)Setting>Fault Reset (3)Safety Alarm Reset



NexRobotEdu Utility - Operation 10

Control speed ratio of each joint

- Default is 100%
- Range is 0 ~ 100%
- Every + and - is 10%
- If necessary, this value can be adjusted.



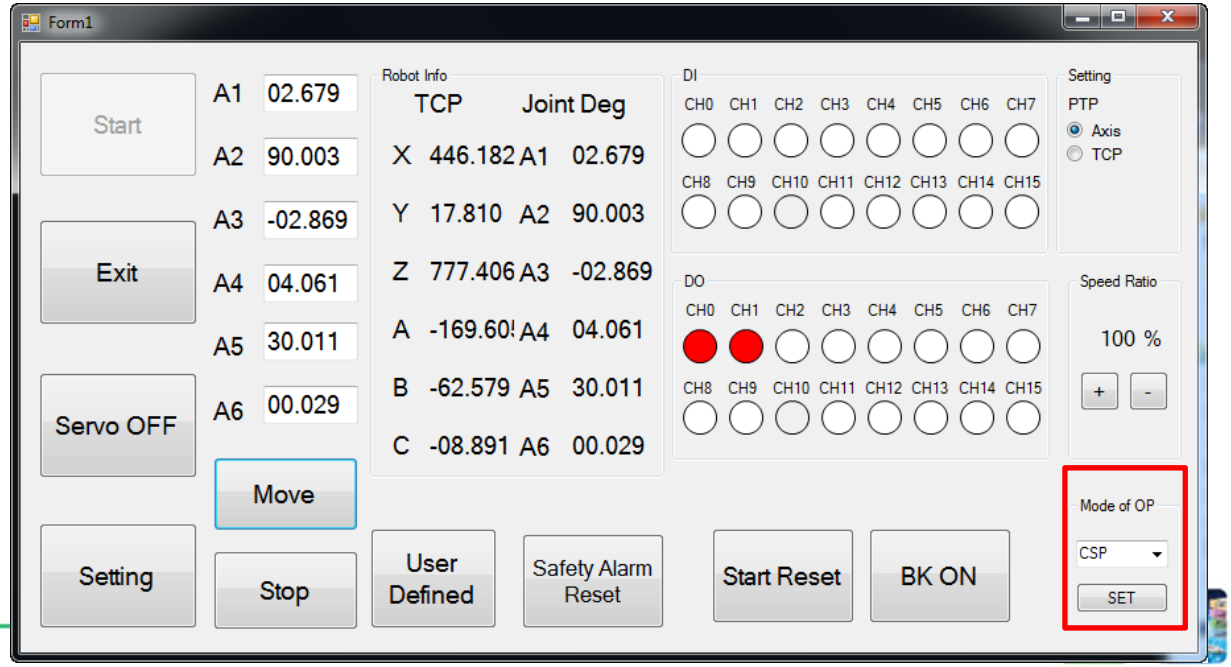
NexRobotEdu Utility - Operation 11

Control Mode of Servo Drive (Mode of OP)

- Three modes can be chose: CSP、CSV、CST
- Before changing mode, utility must have Servo Off and brake first, otherwise drives will not be able to change mode。

Each Control Mode Has A Corresponding User Defined(For Example)

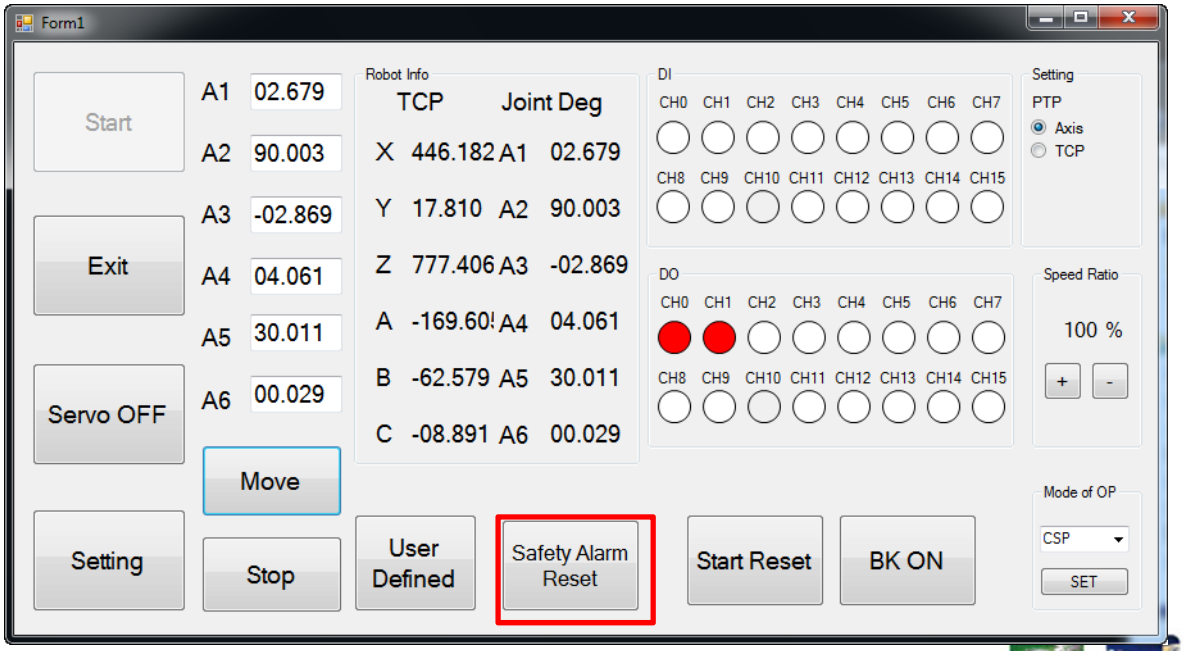
- First joint move 1 degree.
- First joint move by trapezoidal acceleration and deceleration planning.
- First joint move by specified torque command.



NexRobotEdu Utility - Operation 12

Safety Relay

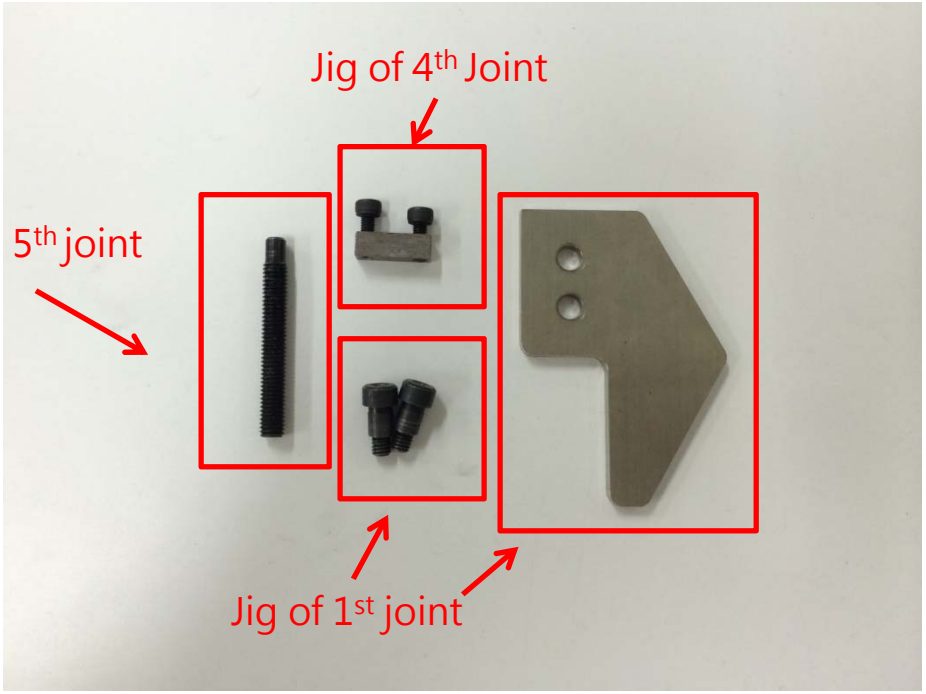
- For safety, controller uses a safety relay to increase safety.
- It is a two-stage mechanism, the 1st one is hardware(external emergency stop)；the 2nd one is software(『Safety Alarm Reset』 bottom)。
- When encountered danger and pressed emergency stop bottom, driver will servo off and brake at each joint.
- Dismissed danger condition and rest emergency stop. At this time, can not have servo on directly, have to press Safety Alarm Reset to have servo on.



- Robot Arm Origin Calibration

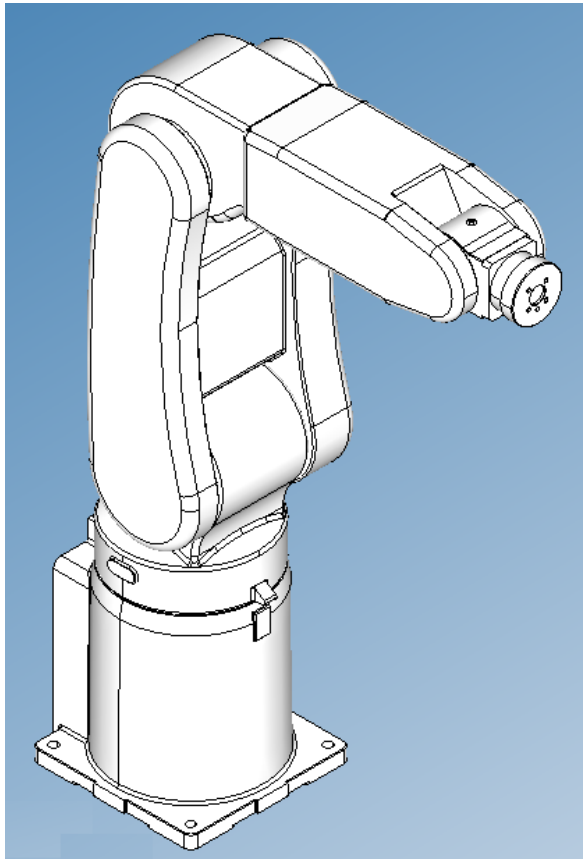
Origin Calibration - Jig

- Robot arm origin calibration used jigs are showed as following figure.



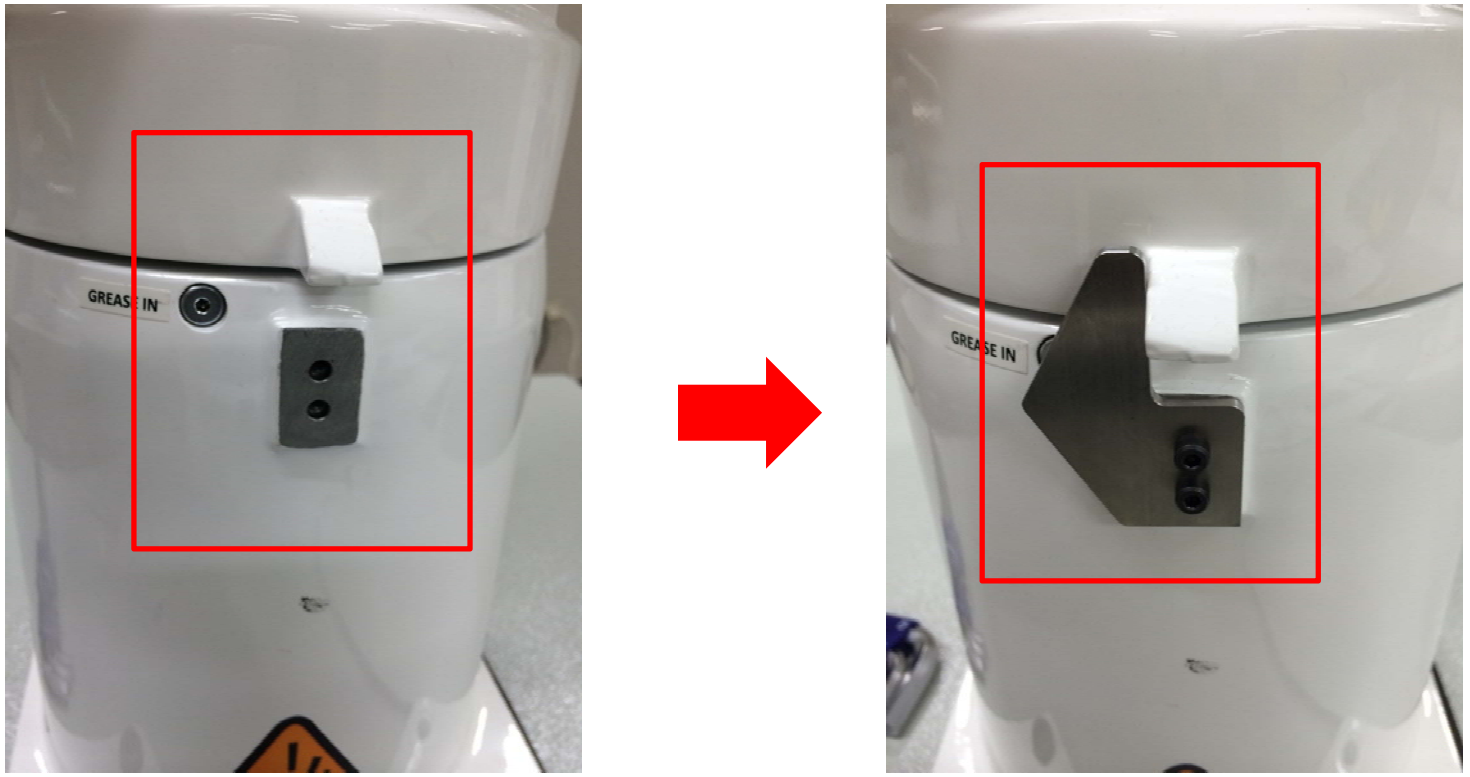
Origin Calibration - Mechanical zero

- Mechanical zero position is showed as following figure, each joint degree is:
  - A1 : 0°
  - A2 : 90°
  - A3 : 0°
  - A4 : 0°
  - A5 : 0°
  - A6 : 0°



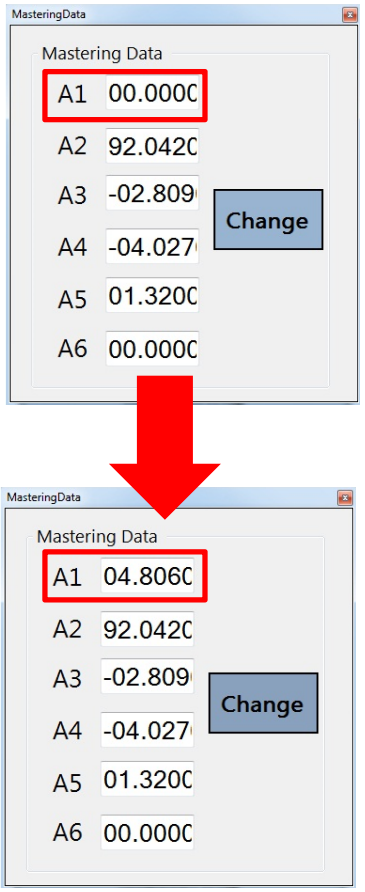
Origin Calibration - First Joint

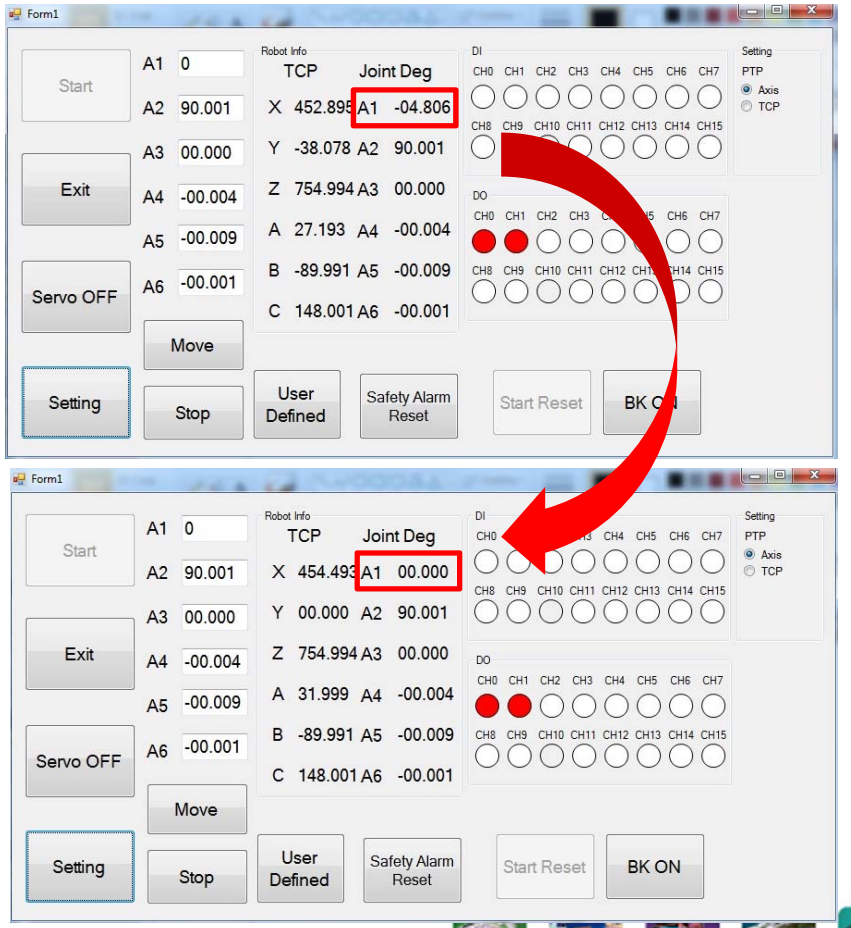
- After lock the first joint jig on first joint as following figure, then move robot arm to align jig by hand.



Origin Calibration - First Joint - Setup

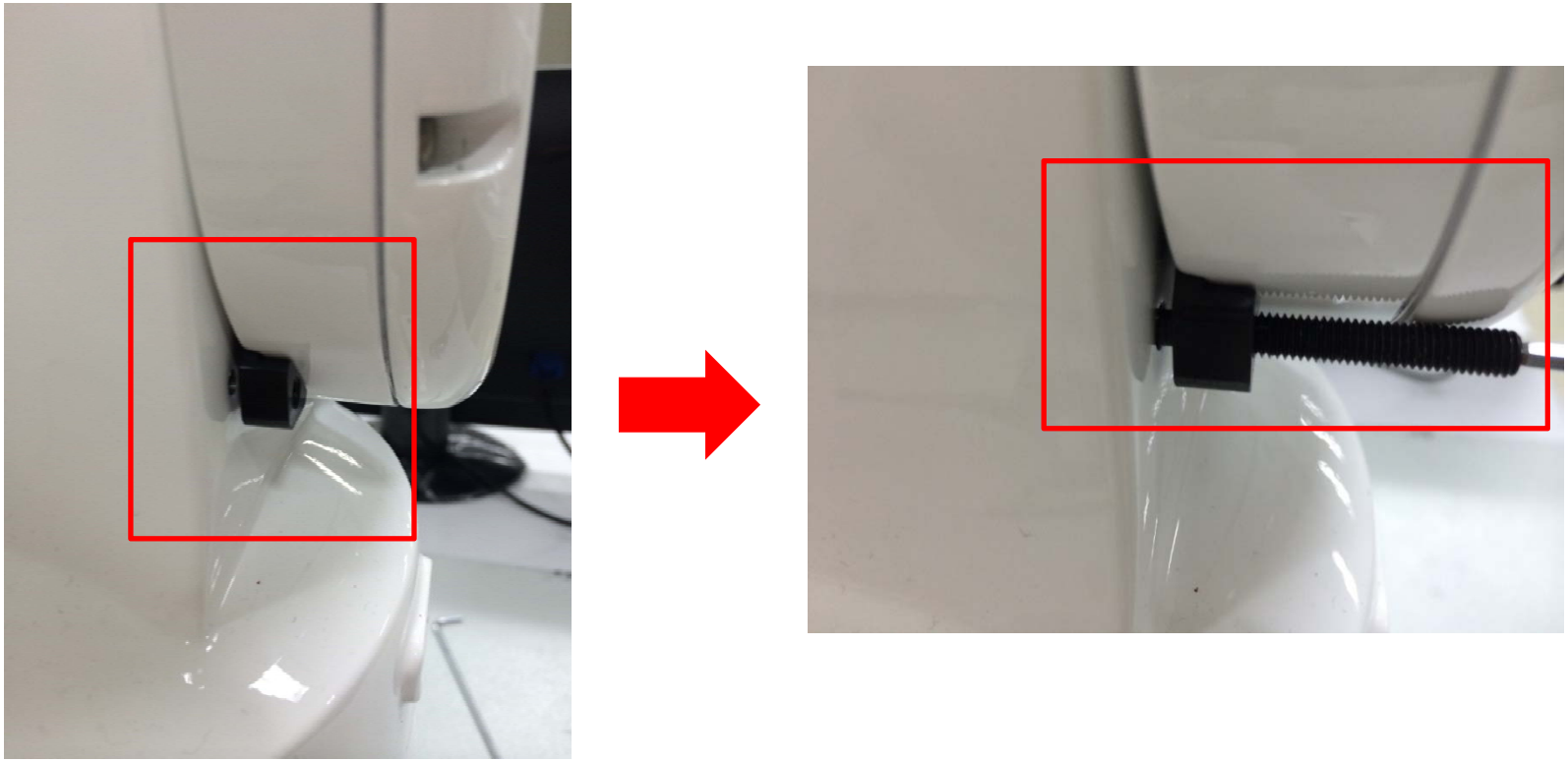
- Click Setting→Mastering in utility, then set all value to zero first.
- At this time, copy the A1 value on utility, paste this value to Mastering Data and reverse.





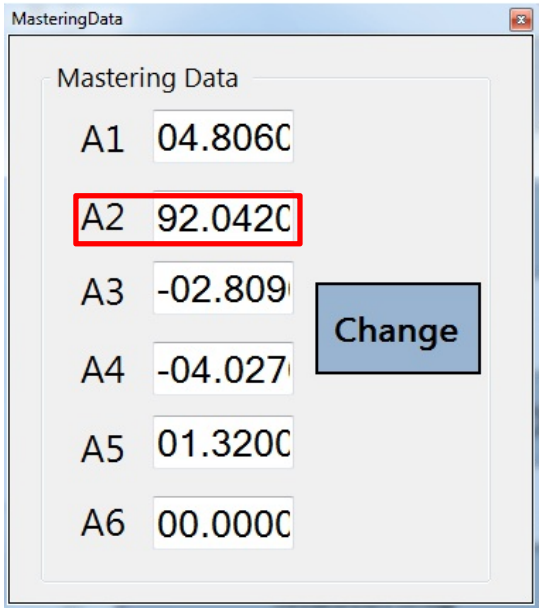
Origin Calibration - Second Joint

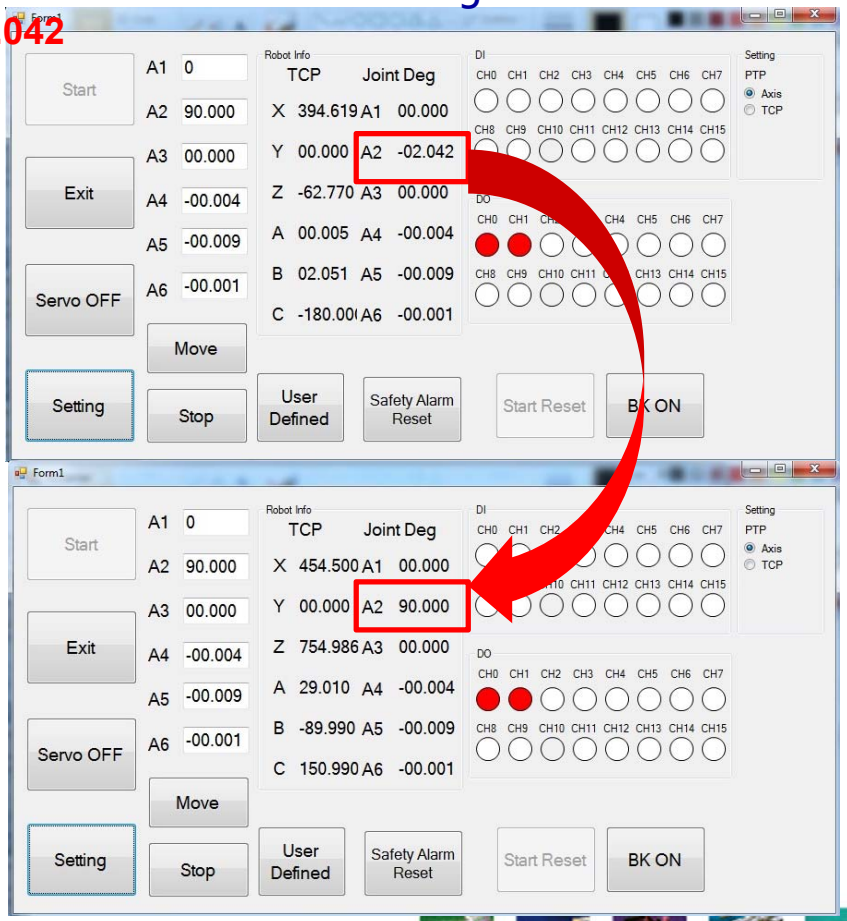
- Move second joint to align jig by hand and screw jig as following figure.



Origin Calibration - Second Joint - Setup

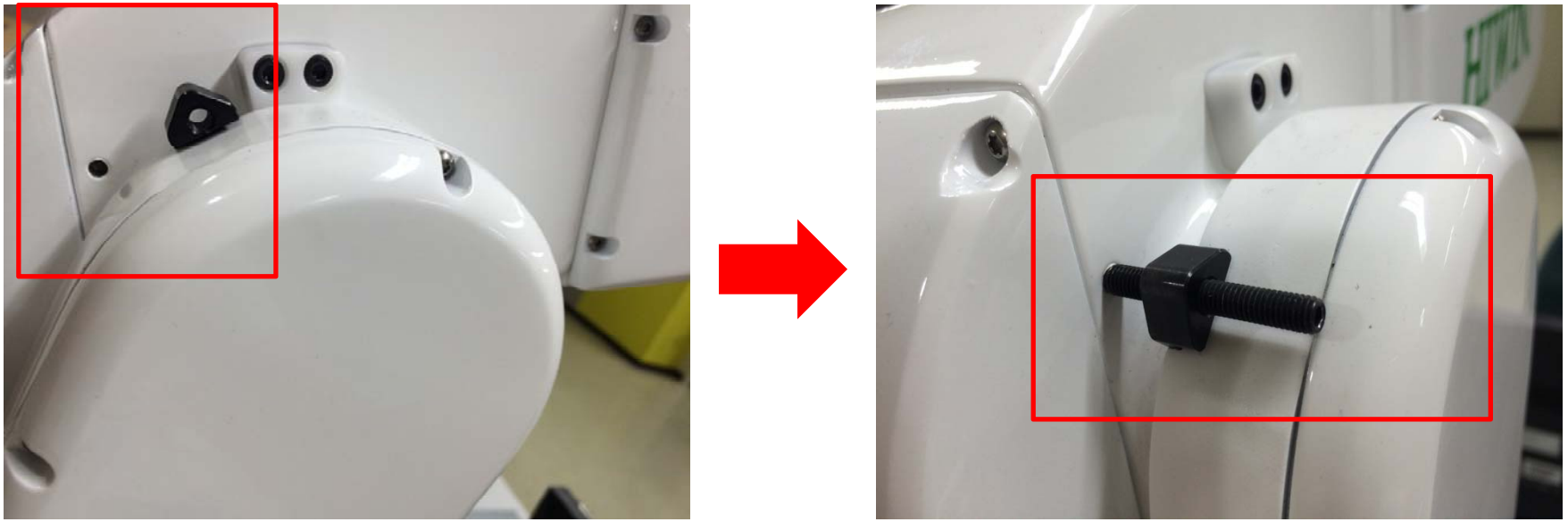
- **The origin position of second joint is 90˚**
- **First set zero to Mastering Data of second joint.**
- **At this time,** copy the A2 value on utility, paste this value to Mastering Data and reverse.
  Ex: A2 value = -02.042, Mastering Date = 92.042





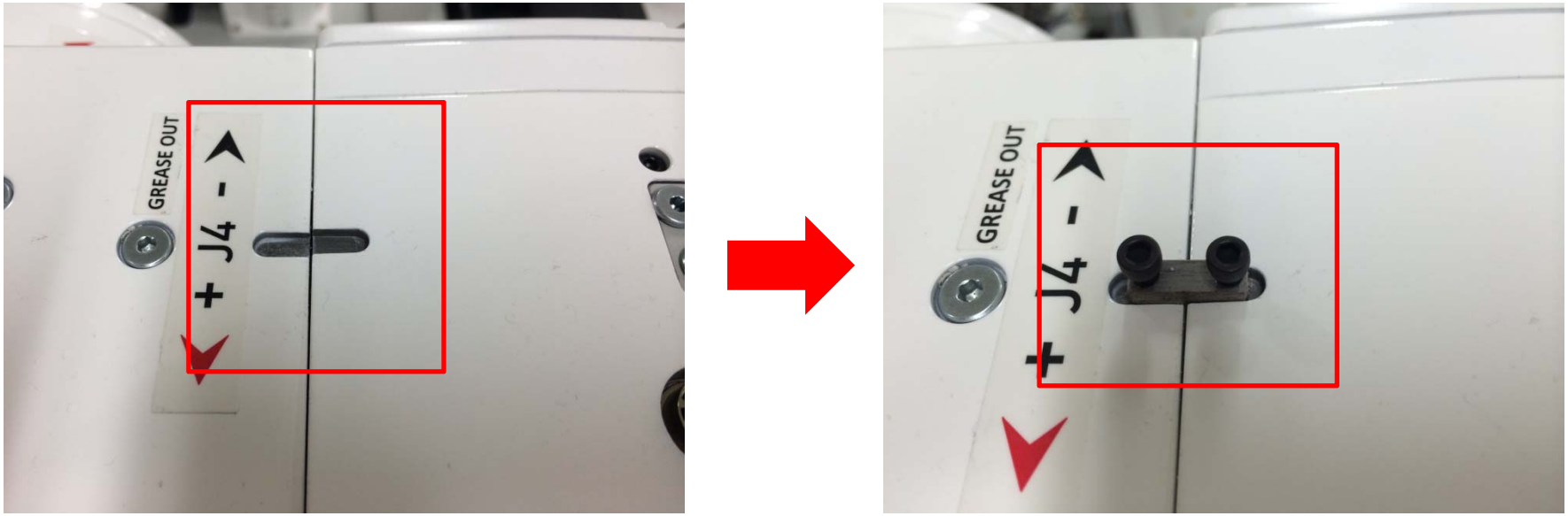
Origin Calibration - Third Joint

- Move third joint to align jig by hand and screw jig as following figure.
- Mastering Data is set in the same way as the first joint.



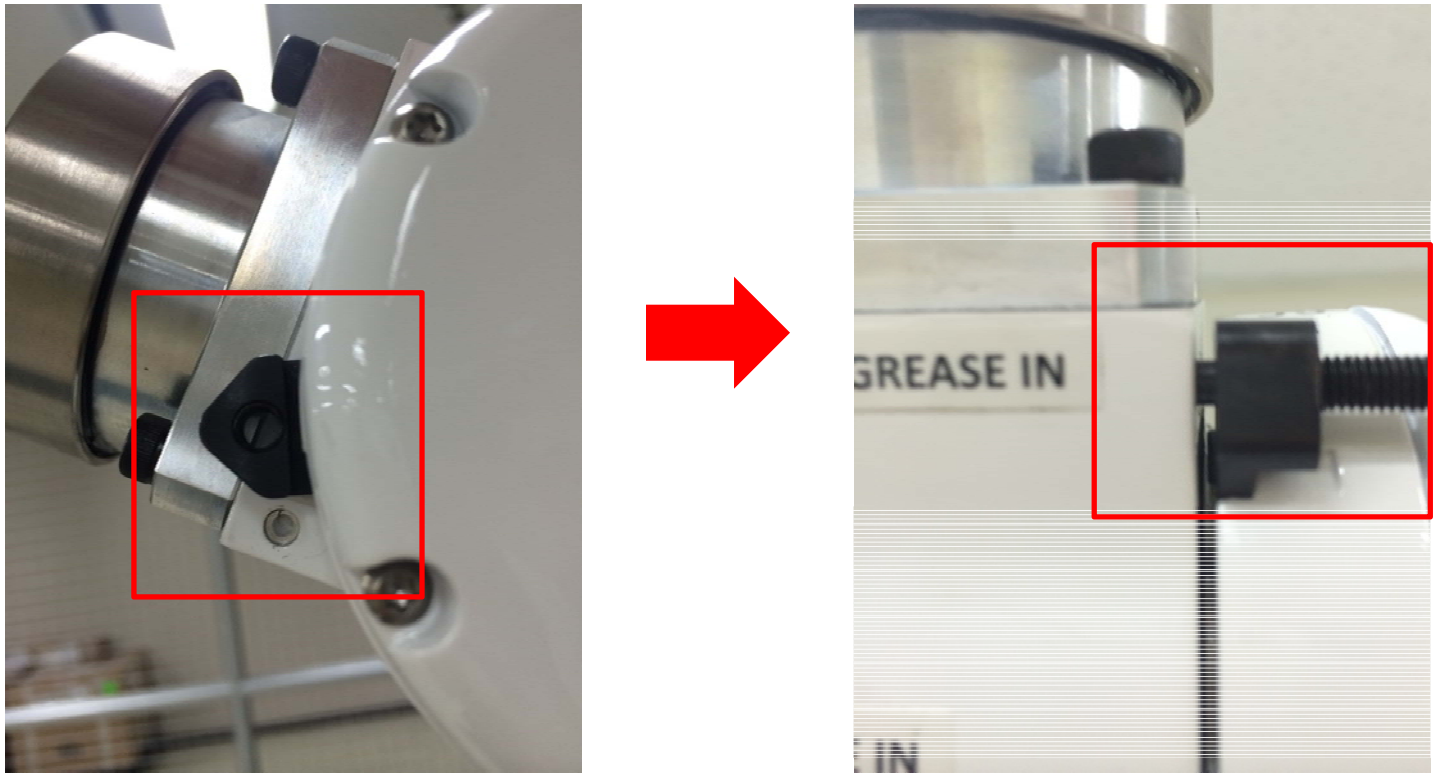
Origin Calibration - Forth Joint

- Move forth joint to align jig by hand and screw jig as following figure.
- Mastering Data is set in the same way as the first joint.



Origin Calibration - Fifth Joint

- Move fifth joint to align jig by hand and screw jig as following figure.
- Mastering Data is set in the same way as the first joint.



Origin Calibration - sixth Joint

- If there is no tool mounted on flange, it does not need calibration.
- If there is a tool mounted on flange, the origin is set by user.

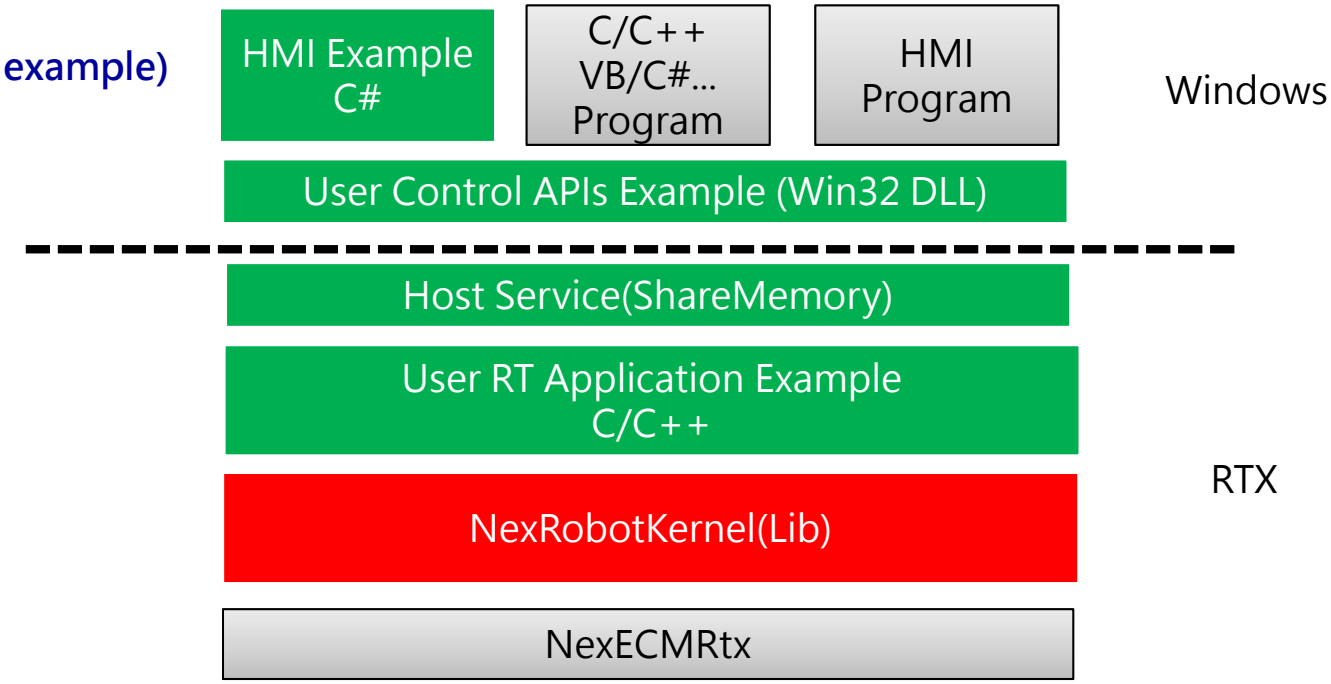
NexRobotEdu - Example Codes

Example Codes

- The example codes folder is from "Start > NexRobot > Samples."
- There are 3 projects:

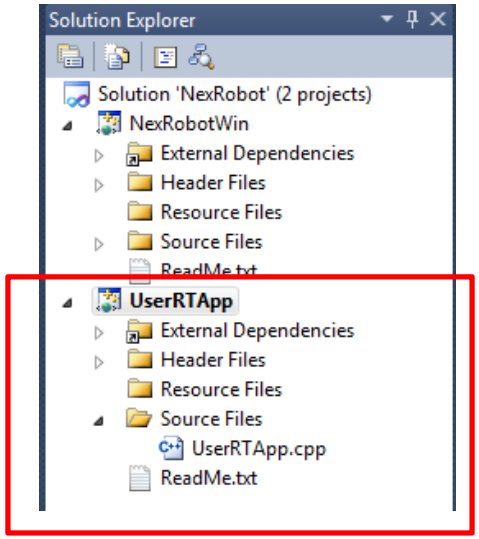
  Win32
  - NexRobotUIExample (C# example)
  - NexRobotWin(dll example)

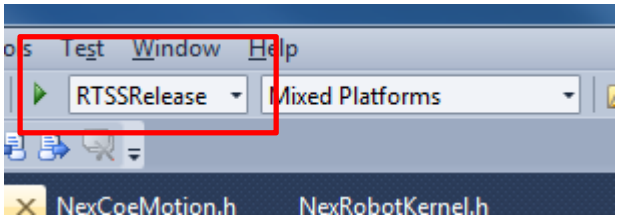
  RTX
  - UserRTApp(RTX example)



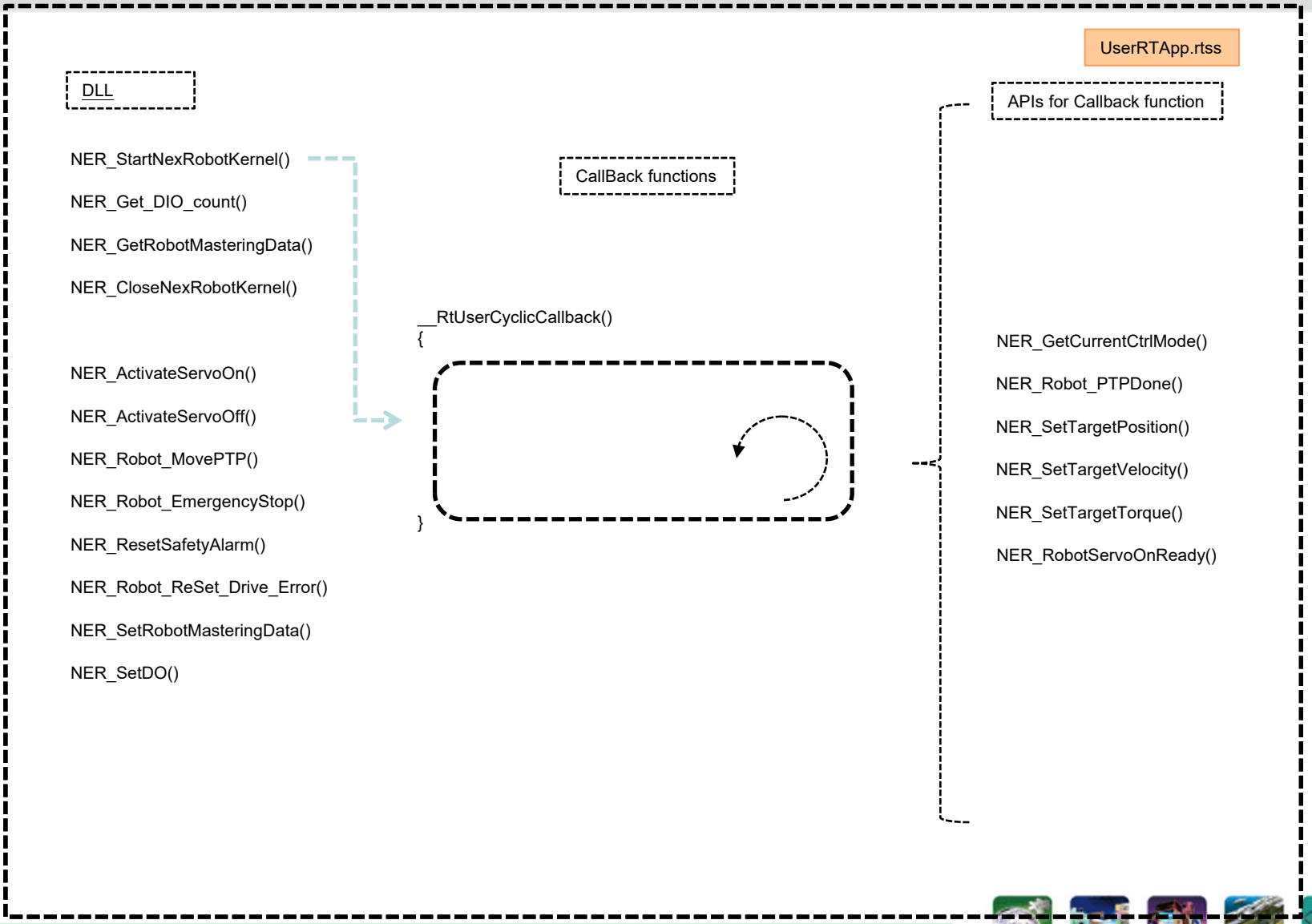
UserRTApp - 1

- UserRTApp is a RTX project, Before compiling, be sure the debug mode is:
  - RTSSRelease or RTSSDebug
- Used Library
  - NexRobotKernel.lib (C:\Program Files\NEXCOM\NexRobot\Lib)
- Other setting is default.



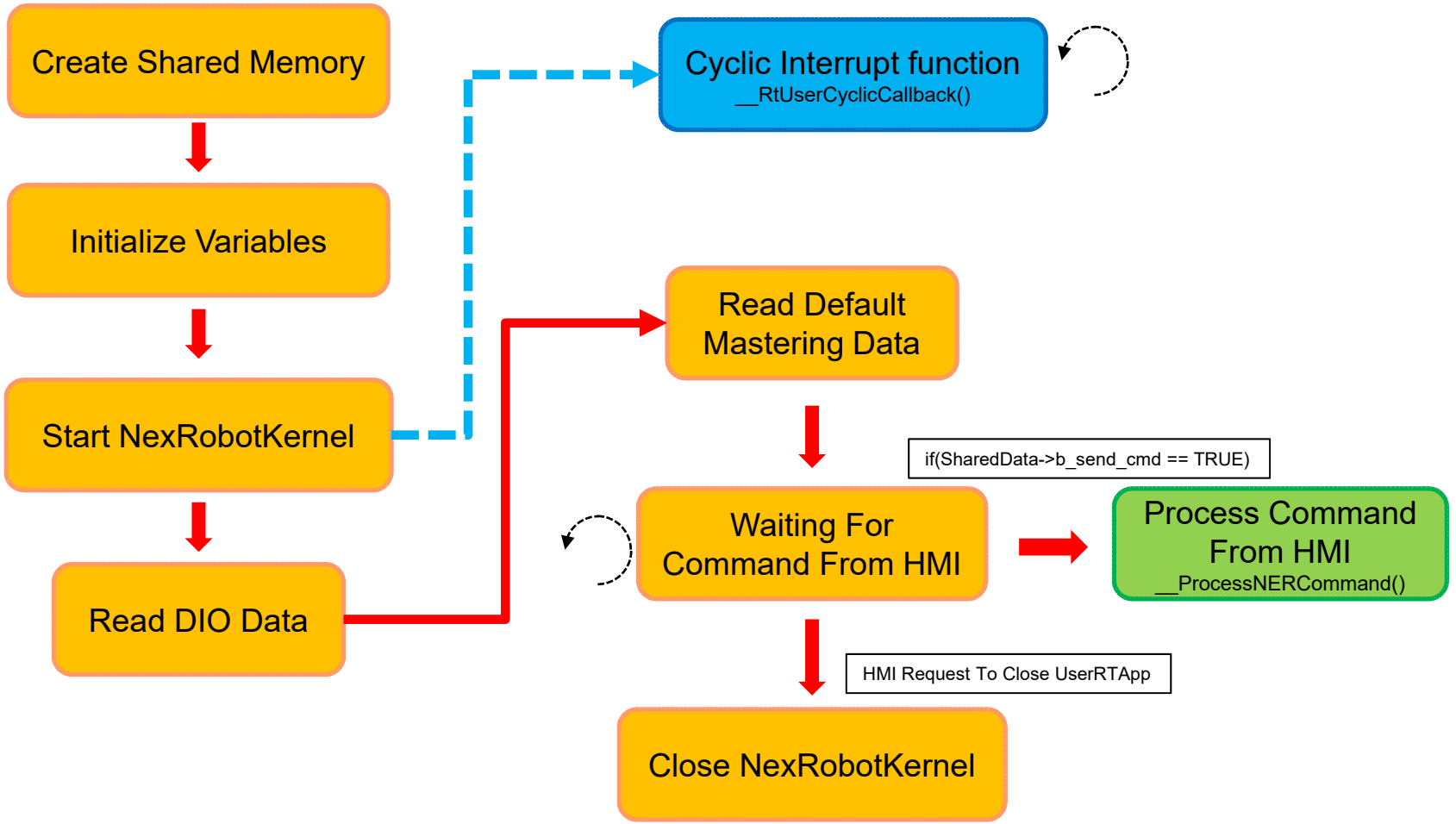


UserRTApp - 2


UserRTApp.rtss
DLL
APIs for Callback function
NER_StartNexRobotKernel()
NER_Get_DIO_count()
NER_GetRobotMasteringData()
NER_CloseNexRobotKernel()
CallBack functions
__RtUserCyclicCallback()
{
}
NER_ActivateServoOn()
NER_ActivateServoOff()
NER_Robot_MovePTP()
NER_Robot_EmergencyStop()
NER_ResetSafetyAlarm()
NER_Robot_ReSet_Drive_Error()
NER_SetRobotMasteringData()
NER_SetDO()
NER_GetCurrentCtrlMode()
NER_Robot_PTPDone()
NER_SetTargetPosition()
NER_SetTargetVelocity()
NER_SetTargetTorque()
NER_RobotServoOnReady()

UserRTApp - 3

- Main() is the entry point of this program, the architecture is as bellow:



UserRTApp - 4

- When start NexRobotKernel, it must be assigned control mode (Please refer to API manual)
- If using NEXCOM PTP function, the control mode must be 「 NER_CYCLIC_POSITION 」
- Support cycle time is 1ms now.
- After start NexRobotKernel, the cyclic interrupt function(__RtUserCyclicCallback()) will start to interrupt at every 1ms.
- At every interruption, system can get new robot arm data as following table.

項目	內容
Motor	Current encoder value(inc)
	Current velocity feedback(inc/s)
	Current torque feedback(%)
	Current joint degree(deg)
	Current joint velocity(deg/s)
	Mastering Data(deg)
Robot	Tool center position
	Number of Digital I/O
	Status of Digital I/O

UserRTApp - 5

- User can self-develop algorithm, and send command in cyclic interrupt function.
- Example of cyclic interrupt function: (Please refer to API manual)
  - (1) NER_SetTargetPosition
  - (2) NER_SetTargetVelocity
  - (3) NER_SetTargetTorque
- If user send command value exceed joint limit of robot arm, system will stop all move, status will be changed to emergency stop.
- If status is emergency stop, user can not send any command.
- Process for reset emergency stop status is as below:

Waiting for robot arm stop → Servo Off → Reset Safety Alarm

 注意	❖ All command are active at Servo On only. ❖ All command are cyclic, command send to drive, drive will reach as far as possible on a periodic command before the next period of time. ❖ Excessive position command will cause the drive to generate an alarm.
 危險	❖ The system can only be prevented joint limit, but robot arm still has a chance with any object collision(include human). ❖ Before move robot arm, please be sure nothing or human is nearby working area of robot arm. ❖ In order to ensure the safety of human and robot arm, external Emergency Stop must be reachable.

UserRTApp - 6

- HMI can send command to UerRTApp through DLL.
- Between DLL and UserRTApp use shared memory to transfer data.
- In UserRTApp, to execute command from shared memory is: __ProcessNERCommand().
- UserRTApp offers 3 types of motion example for users.

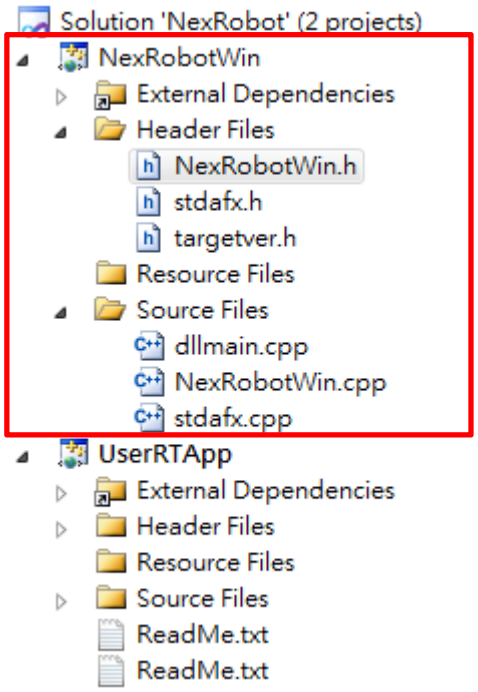
Types	Definition
First joint move 1 degree	From 0 degree, every cycle time send 0.001 degree command(0、0.001、0.002 … 0.999、1.000) , total send 1000 cycle times.(Last 1 second)
First joint move by trapezoidal acceleration and deceleration planning.	Set maximum velocity and acceleration, every cycle time send velocity calculated by trapezoidal acceleration and deceleration planning.
First joint move by specific torque command.	Set target torque, every cycle time send target torque, last 10 second.

Danger

- The system can only be prevented joint limit, but robot arm still has a chance with any object collision(include human).
- Before move robot arm, please be sure nothing or human is nearby working area of robot arm.
- In order to ensure the safety of human and robot arm, external Emergency Stop must be reachable.

NexRobotWin - 1

- NexRobotWin is a win32 DLL project.
- Using Header:
  - NexRobotKernel.h (C:\Program Files\NEXCOM\NexRobot\Header)
- NexRobotWin offers examples how to pack RTX shared memory to win32 DLL.




```
// type 1 : MoveJoint, type 2 : MoveTCP
int FNTYPE NexR_MovePtp(U16_T  type,
                        F64_T *target_data,
                        F64_T *max_vel,
                        F64_T *acc);

int FNTYPE NexR_GetCmdDone(void);

int FNTYPE NexR_ReSetDriveError(void);

int FNTYPE NexR_GetMasteringData(F64_T *masteringdata);

int FNTYPE NexR_SetMasteringData(F64_T *masteringdata);

int FNTYPE NexR_SetDIO(unsigned int DO_Index,unsigned int *output_data);

int FNTYPE NexR_GetDIOCount(unsigned int &DIO_Cnt);

int FNTYPE NexR_GetDIOSize(unsigned int DIO_Index,unsigned &inputinbyte,unsigned &outputinbyte);

int FNTYPE NexR_EmgStop(void);

int FNTYPE NexR_ResetSafetyAlarm(void);

int FNTYPE NexR_UserDefined(void);
```

NexRobotWin - 2

win32

```
// type 1 : MoveJoint, type 2 : MoveTCP
int FNTYPE NexR_MovePtp(U16_T  type,
                        F64_T *target_data,
                        F64_T *max_vel,
                        F64_T *acc);

int FNTYPE NexR_GetCmdDone(void);

int FNTYPE NexR_ReSetDriveError(void);

int FNTYPE NexR_GetMasteringData(F64_T *masteringdata);

int FNTYPE NexR_SetMasteringData(F64_T *masteringdata);

int FNTYPE NexR_SetDIO(unsigned int DO_Index,unsigned int *output_data);

int FNTYPE NexR_GetDIOCount(unsigned int &DIO_Cnt);

int FNTYPE NexR_GetDIOSize(unsigned int DIO_Index,unsigned &inputinbyte,unsigned &outputinbyte);

int FNTYPE NexR_EmgStop(void);

int FNTYPE NexR_ResetSafetyAlarm(void);

int FNTYPE NexR_UserDefined(void);
```

RTX

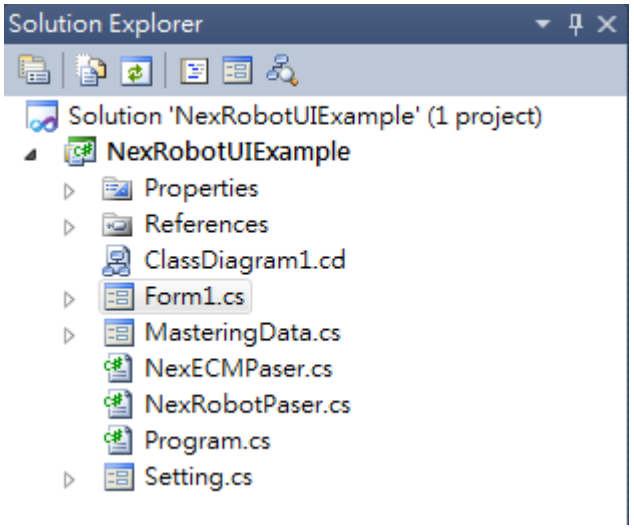
main()

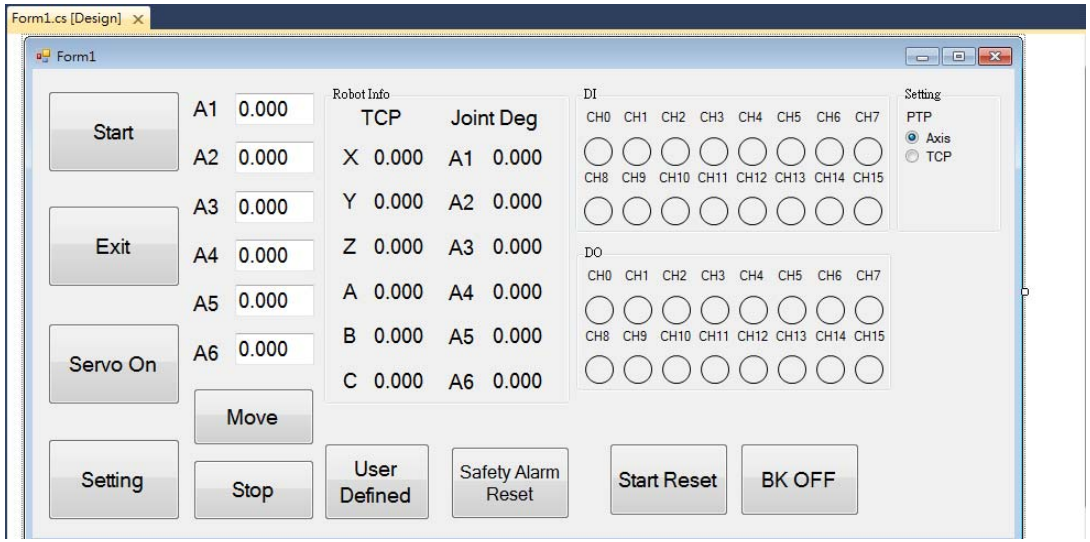
```
while(!g_terminate)
{
    if(SharedData->b_send_cmd == TRUE) //receive cmd from win32 system
    {
        // Reset
        SharedData->b_send_cmd = FALSE;

        // Process Receive Command Here
        __ProcessNERCommand();

        // Set receive command flag to true
        SharedData->b_rec_cmd = TRUE;
    }

    Sleep(4); // cmd polling cycle set as 4ms

}
```

Shared Memory

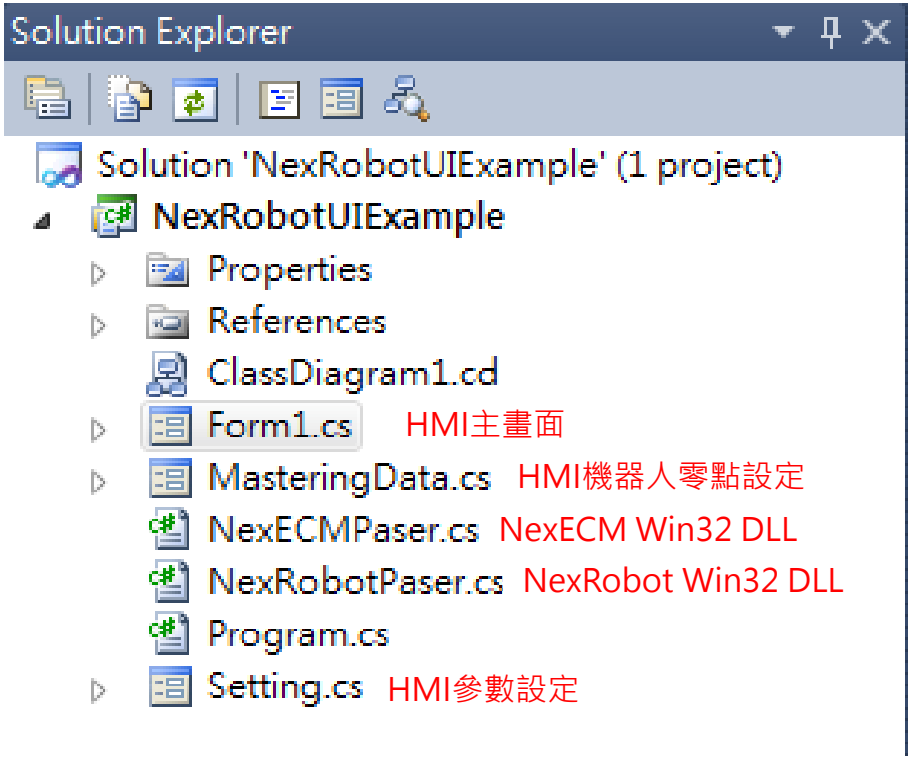
NexRobotUIExample - 1

- NexRobotUIExample is a win32 C# project.
- NexRobotUIExample offers examples how to use C# to control RTX file through a win32 DLL from NexRobotWin.





NexRobotUIExample - 2



NexRobotEdu – Trouble Shooting

Trouble Shooting

- **Click 『Start』 bottom to start system has an error.**
  - Please check hardware setup is correct or not.(Could be RJ45 cable not well-connected)
  - Please check software installation process is correct or not.(Could be environment setup incorrect)
  - Please check DO[0] is sent or not.
- **Servo On failed.**
  - Please check emergency stop is connected to CN3 connector.
  - Please check emergency stop is pressed or not.
  - Once pressed emergency stop, after reset you have to click 『Start Reset』 bottom.
- **Alarm and noise sound after run robot arm.**
  - Please check robot arm has braked or not.
  - Please check DO[1] is sent signal or not.(untie brake=ON；brake=OFF)
- **Command are A1=0、A2=90、 A3=0 、 A4=0 、 A5=0 、 A6=0, robot arm is not origin position.**
  - Please re-execute Origin Calibration.
- **Close UIExample window by top right X, window has an error message.**
  - Please use 『EXIT』 bottom in UIExample window to close, because only 『EXIT』 bottom can close NexECM communication, this is a correct shutdown process.


IntelligentWorks
EmbeddedPro & AutomationPro
Versatile Solutions for the
Multimedia Age
HOLIDAY SALE
NEXCOM SMARTPHONE
Empowering the
Mobile Workforce
A Passion for Quality,
Security and Connectivity
Cyber Smart, Hyper Vision

Thank You!