

Manual



FlexiVision

KAWASAKI PLUG-IN

ars

Feeding
Industrial
Robotics

INDICE

1. Installazione del Plug-In
2. Configurazione del Plug-In
2. Lista Comandi FlexiVision

Questo Plug-in è nato con l'idea di comunicare in maniera **rapida e sicura** con FlexiVision tramite i robot **Kawasaki**.

FlexiBowl[®] Plug-In

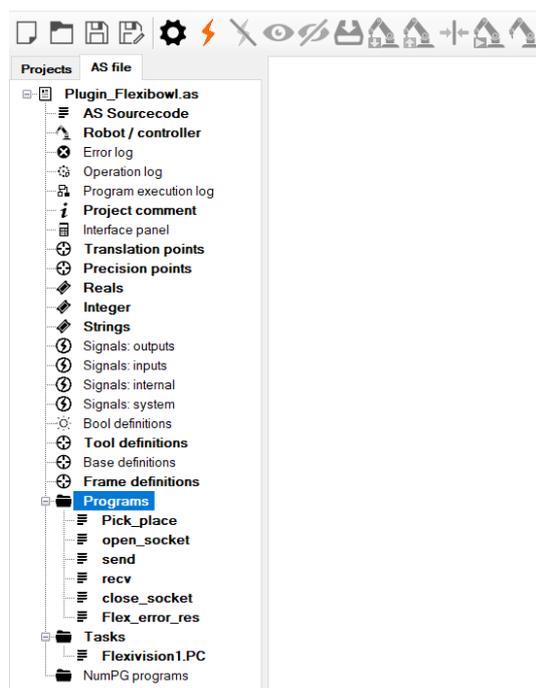


Installazione del Plug-In

Attraverso il programma **KIDE** aprire il file fornito da "ARS srl "**Flexivision_Plugin1.AS** tramite il percorso **File/Open**.



Al termine della procedura di caricamento dovranno essere visualizzate le seguenti voci in grassetto.



E' ora possibile procedere alla modifica dei programmi e dei dati contenuti all'interno del **file .AS**

Step 1.

Descrizione del programma di comunicazione Flexivision1.PC

Il programma **Flexivision1.PC** permette la comunicazione tra Flexivision e il Robot **Kawasaki**.

Di seguito viene riportata una descrizione delle principali variabili utilizzate nel codice. Tali variabili non devono essere modificate all'interno del programma **Flexivision1.PC** ma all'interno del programma **Pick_place**. (Vedi 3. Configurazione dei parametri dal Pick_place)

- **Port:** Indica la porta utilizzata per la comunicazione con Flexivision, di **default 8192**.
- **Flexi_resp:** Contiene la risposta da parte di Flexivision al comando inviato.
- **Pattern_id:** I valori assunti sono sotto riportati:
 - **-1:** Il comando inviato è **"test_Locator"** e non è stato rilevato nessun Pattern. (Risposta da Flexivision **"Null;Null;Null;Null"**)
 - **0:** Il comando inviato non restituire una locazione del Pattern. (ex: "get_Recipe")
 - **Val>0:** Id del Pattern rilevato. (**"Pattern_1;10;11.2;4;"** Val assume il valore 1)
- **Error_Flb:** I valori assunti sono sotto riportati:
 - **0:** Non è presente nessun errore.
 - **1:** E'avvenuta la disconnessione tra Flexivision e il Robot Kawasaki.
 - **2:** Flexivision ha inviato un errore (ex un errore sulla camera), tale errore può essere visualizzato in *Flexi_err_str*.
 - **3:** comando non definito per Flexivision.
- **Flexi_err_str:** Contiene i messaggi di errore generati.

Configurazione del Plug-In

- **Flexi_Flag:** Permette l'esecuzione del comando. Settando "**Flexi_Flag=0**" viene inviato il comando contenuto nella variabile string "**command**". Terminata la comunicazione "**Flexi_Flag**" viene automaticamente settata a 1.
- **Flexi_stop:** Settando "**Flexi_stop=1**" avviene l'invio del comando "**stop_Locator**". L'invio del comando "**stop_Locator**" può avvenire in qualsiasi posizione utilizzando le istruzioni:
Flag_Flexi=0
Flexi_stop=1
Avvenuto lo stop "**Flexi_stop**" viene riportata al valore automaticamente.
- **Flexi_stopped:** Viene settata automaticamente ad 1 quando è stato inviato il comando di "**stop_Locator**". Deve essere resettata al valore di default 0, tramite il programma **Pick_Place** per poter inviare altri comandi a Flexivision.
- **Command:** Contiene il comando inviato a Flexivision quando "**Flexi_Flag**" è posta a 0.
- **Flexi_point:** Contiene le coordinate restituite da Flexivision.

Il programma Flexivision1.PC utilizza i programmi:

- **Open_socket**
- **Send**
- **Recv**
- **Close_Socket**

Per una corretta gestione del Socket di comunicazione.

Step 2.

Configurazioni dei parametri dal programma **Pick_Place**

- **Coordinate "fisse": z_pos, rx_pos, ry_pos** non sono rilevate dal sistema di visione e dovranno essere configurate in base alla propria configurazione del robot. Tali coordinate sono riferite al sistema di riferimento di "**Flexi_frame**".
- **Flexi_frame:** Rappresenta il frame comune del robot e di Flexivision. Procedere alla calibrazione della camera e al "**teach**" del medesimo frame con il nome "**Flexi_frame**".
- **Configurazione del Tool**
- **Configurazione della comunicazione**

Configurazione del Plug-In

```
;CARICO LA RICETTA E VERIFICO CHE SIA STATA CARICATA
$command="set_Recipe=kawasaki_plugin"
Flexi_Flag = 0
WAIT Flexi_Flag==1
```

Per l'invio del comando a Flexivision è necessario assegnare il comando da inviare alla stringa **"command"**. Settando **"Flexi_Flag=0"** avviene l'invio del comando. **Flexi_Flag** viene settata a 1 automaticamente da **Flexivision1.PC** una volta ottenuta risposta da "Flexivision".

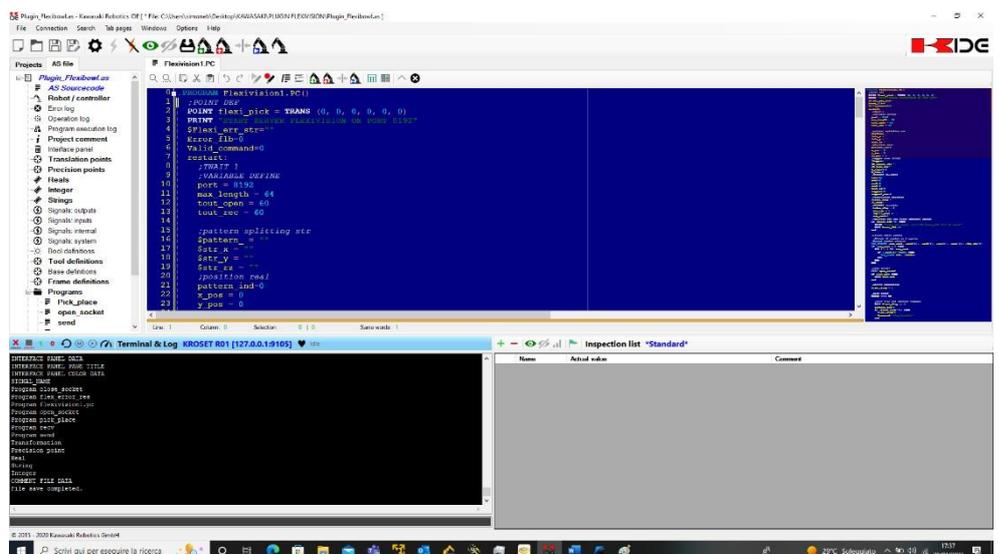
Il programma **Pick_Place** chiama la funzione **Flex_Error_Res** quando si verifica un errore. Tale funzione permette il reset della flag **"Error_Flb"**. Finchè non avviene il reset di tale Flag non sarà possibile inviare comandi a Flexivision.

Inserire all'interno della funzione **"Flex_Error_Res"** eventuali procedure da eseguire al verificarsi di un errore.

Step 3.

Caricamento del Plug-In sul controller reale

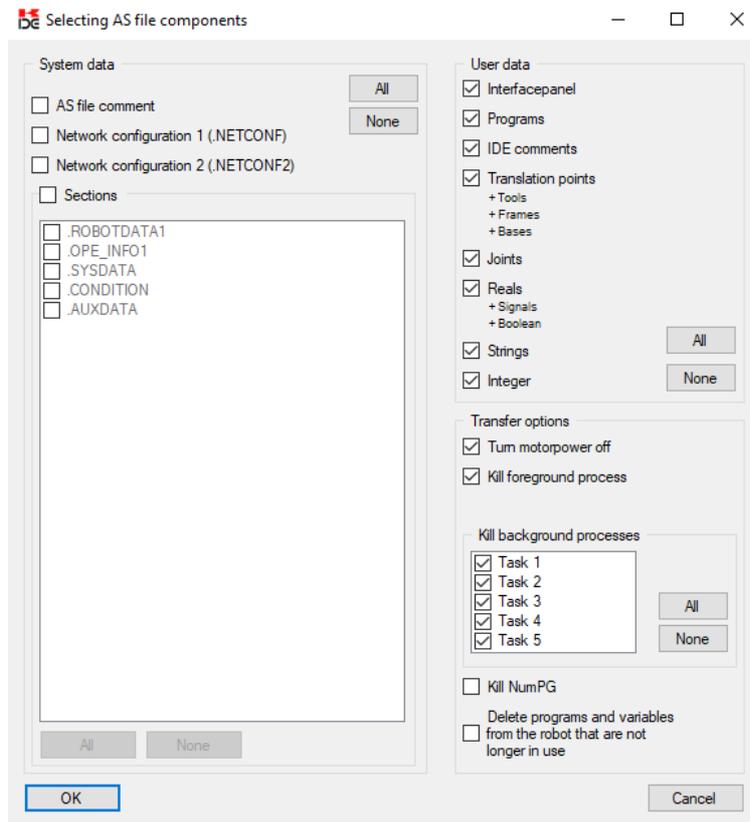
Apportate le modifiche necessarie procedere al caricamento del Plug-in sul controller reale. La connessione con il controller fisico viene stabilita tramite l'icona ⚡



Configurazione del Plug-In

Step 4.

Inviare il file **.AS** al controller attraverso l'icona  selezionando le spunte su tutti gli **"User data"** come visibile in seguito.



Lista Comandi FlexiVision

Per inviare il comando a FlexiVision è necessario modificare il valore della stringa "command".

N_Mission	Command	Action
1	"start_Locator"	Starts the parts localization process by recalling the FlexiBowl handling routine in case there are no parts that can be picked up. Return: "Pattern1;x;y;r".
2	"stop_Locator"	Stops the process of locating the object with the aid of the FlexiBowl.
3	"turn_Locator"	If no parts are picked up, by this command the operator can make the Flexibowl rotate and the "start_Locator" routine start. Return: "Pattern1;x;y;r".
4	"test_Locator"	Starts the process of locating the object without the aid of the FlexiBowl. Return: "Pattern1;x;y;r".
5	"start_Control"	Starts the inspection cycle. Return: "Control1;x;y;r".
6	"state_Locator"	Locator status diagnostics is shown: Return: "Locator is Running" "Locator is in Error" "Locator is not Running".
7	"start_Empty"	Start the FlexiBowl® Quick-Emptying sequence. Return: "start_Empty ended"
8	"get_Recipe"	The name of the recipe currently loaded on FlexiVision is shown. Return: "recipe name".
9	"set_Recipe=recipe name"	The recipe corresponding to the sent "recipe name" is loaded.