

Author Roland Loy, Florian Walter
Date 10/04/2024
Revision 1.2

FLEXFEEDER UR PLUG-IN

Quickstart Guide Multifeeder Beispielprogramme

flexfeeder GmbH
Gerhard-Kindler-Straße 8
72770 Reutlingen

Phone +49 7121 8933-661

flexfeeder@flexfactory.com
flexfactory.com



Inhaltsverzeichnis

<i>Einleitung</i>	3
<i>1. Installation</i>	4
<i>2. Kalibrierung</i>	6
<i>3. Anpassung der Beispielprogramme</i>	7



EINLEITUNG

Dieses Dokument dient dem schnellen Einstieg in die Einrichtung und Konfiguration des flexfeeder UR Plug-in der Flexfactory AG für den Betrieb mehrerer Flexfactory flexfeeder X im Zusammenspiel mit einem UR-Roboter. Die im UR Plug-in mitgelieferten Beispielprogramme für Multifeeder-Anwendungen finden Sie im Softwarepaket im Unterverzeichnis „*examples*“.

Wir möchten Sie darauf hinweisen, dass wir die Teilnahme an einer unserer eintägigen Schulungen empfehlen. Diese Schulungen bieten Ihnen wertvolle Einblicke und vermitteln Ihnen das nötige Know-how, um das Beste aus unseren Produkten herauszuholen.

Für weitere Informationen und zur Anmeldung besuchen Sie bitte unsere Schulungsseite unter:

<https://www.flexfactory.com/kontakt-und-support/schulungen>



1. INSTALLATION

Bitte deinstallieren Sie ggf. ein bereits installiertes flexfeeder URCap bevor Sie das mitgelieferte URCap ``flexfeeder-1.1.0.urcap`` installieren.

Nach erfolgreicher Installation des URCaps können Sie unter ``Installation > URCaps > FFX installation`` bis zu max. vier flexfeeder Verbindungen einrichten.

Die grundlegende Einrichtung erfolgt analog zur Single-Feeder Variante. (siehe Dokumentation „*Quickstart & Reference Guide*“)

Für die mitgelieferten Multifeeder Beispielprogramme verwenden Sie bitte in der Übersicht die ersten beiden Plätze, um Ihre flexfeeder einzurichten.

Klicken Sie dafür in Zeile 1) und 2) jeweils auf das "+" Symbol. Bitte vergewissern Sie sich, dass die folgenden Einrichtungsschritte für beide flexfeeder durchgeführt wurden:

1. Tragen Sie die IP-Adresse des flexfeeder ein. Der Port sollte immer 8082 sein.
 - a. Klicken Sie auf *"Apply and test connection"*.
 - b. Wenn der flexfeeder erreichbar ist, sollte daneben ein grüner Haken und *"Flexfeeder is connected"* erscheinen.
2. Legen Sie die *"Above Shaker Position"* und die *"Out of View Position"* fest.
 - a. ***"Above Shaker Position"***:
Nach dem Greifen eines Teils aus dem Arbeitsraum wird diese Position angefahren. Z. B. über einer Ecke des Arbeitsraumes. Diese Position muss von jeder Greifposition auf der Abgreiffläche kollisionsfrei erreichbar sein.
 - b. ***"Out of View Position"***:
Pose des Roboterarms, die sich außerhalb des Sichtfelds der flexfeeder-Kameras befindet.
Hinweis: Der Greifer und alle Kabel sollten sich in dieser Pose ebenfalls außerhalb des Sichtfelds der Kameras befinden.



-
3. Durch einen Klick auf "< Overview" kommen Sie wieder zur Übersicht zurück.

Stellen Sie in der Übersicht sicher, dass die beiden flexfeeder aktiviert (Schalter auf On) sind.



2. KALIBRIERUNG

Legen Sie für jeden flexfeeder ein gesondertes Kalibrierprogramm auf dem UR an und führen Sie die Kalibrierung jeweils für beide flexfeeder nacheinander durch. Folgen Sie dabei den Vorgaben und Anweisungen, die in der Dokumentation „*Quickstart & Reference Guide*“ zum Thema Kalibrierung beschrieben sind.

Sollten Sie die Kalibrierung bereits zuvor für beide Geräte durchgeführt haben, können Sie diesen Schritt überspringen.

Hinweis: Die Durchführung der Kalibrierung ist im Idealfall nur einmal erforderlich, um die Abbildung zwischen den Koordinatensystemen der Kamera und des Roboters zu bestimmen.



3. ANPASSUNG DER BEISPIELPROGRAMME

Um die Beispielprogramme für den Produktivbetrieb mit Multifeeder-Anwendungen auszuführen, stehen zwei unterschiedliche Beispielprogramme zur Auswahl.

1. Abgreifen eines Bauteils von zwei flexfeeder (Laufzeit optimiert)

Von beiden flexfeeder wird das gleiche Bauteil sortenrein abgegriffen. Es sollen möglichst kurze Taktzeiten realisiert werden, wobei es keine Vorgabe gibt, von welchem der beiden flexfeeder das nächste Bauteil abgegriffen werden soll. Die Reihenfolge, von welchem flexfeeder als Nächstes abgegriffen wird, ist somit nur davon abhängig, auf welchem der beiden flexfeeder am schnellsten ein neues Bauteil zur Verfügung gestellt werden konnte. Dieses Beispielprogramm findet vor allem dann Anwendung, wenn es um Taktzeitoptimierung geht, da durch die beiden flexfeeder eine höhere Verfügbarkeit von greifbaren Bauteilen realisiert werden kann.

Beispielprogramm:

„/examples/ ffx_production_example_two_feeder.urp“

2. Alternierendes Abgreifen von zwei unterschiedlichen Bauteilen von zwei flexfeeder

Von beiden flexfeeder werden zwei unterschiedliche Bauteile sortenrein auf den beiden flexfeeder gefahren. Hierbei ist das abwechselnde Abgreifen der beiden unterschiedlichen Bauteile von jeweils einem der Feeder umgesetzt. Die Reihenfolge, von welchem flexfeeder als Nächstes abgegriffen wird, ist fix vorgegeben und wird strikt eingehalten. Ein gängiger Anwendungsfall für dieses Beispielprogramm ist insbesondere die Umsetzung einer Montageaufgabe.

Beispielprogramm:

„/examples/ffx_production_example_two_feeder_alternating.urp“



Bitte laden Sie das gewünschte Multifeeder UR-Programm:

``ffx_production_example_two_feeder.urp``

oder

``ffx_production_example_two_feeder_alternating.urp``

und vervollständigen Sie bitte die folgenden Schritte, um ein lauffähiges Beispielprogramm zu erzeugen:

1. Teachen Sie die ``start_pose``
2. Im Knoten ``FFX1 production`` müssen die folgenden Einstellungen noch vervollständigt werden.
 - a. Feeder-Job auswählen.
 - b. Pick Pose für den Feeder 1 teachen.
3. Im Knoten ``FFX2 production``
 - a. Feeder-Job auswählen.
 - b. Pick Pose für den Feeder 2 teachen.
4. In Folder ``Pick form feeder 1``
 - a. In Folder ``Grab tool action``
 - i. Greifaktion implementieren.
 - b. In Folder ``Place part``
 - i. Ablauf für die Ablage des Bauteils implementieren.
5. In Folder ``Pick form feeder 2``
 - a. In Folder ``Grab tool action``
 - i. Greifaktion implementieren.
 - b. In Folder ``Place part``
 - i. Ablauf für die Ablage des Bauteils implementieren.

