



Version 5.10

Machine Interface

Product Description

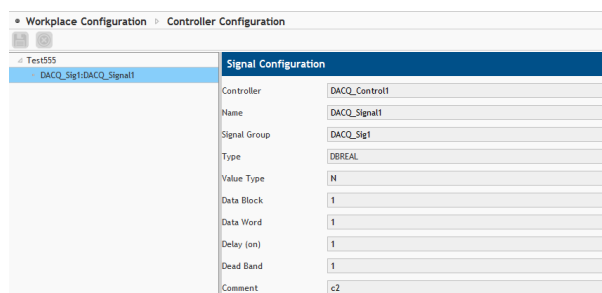
Document:	Product Description - Machine Interface
Created:	2016-07-18
Last changes:	2020-02-05
Author:	AEgilmez



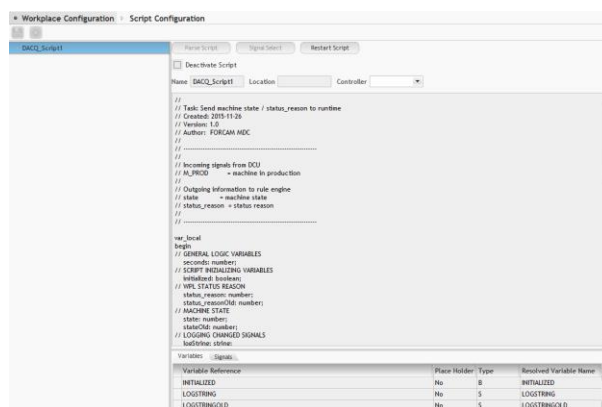
Product Description

FORCAM FORCE™ Machine Interface facilitates the automatic collection of stroke signals, quantity reports, machine status and error codes from a range of different systems and machines. The data collected can then be specified in greater detail on the FORCAM FORCE™ shop floor terminal, in other words system and machine stoppages or waste and rework quantities can be explained. The data collected is condensed into highly diagnostic information by other system components, analyzed and fed back to the overarching ERP system. FORCAM FORCE™ Workbench also makes it possible to retrospectively correct the quantities and operating states collected within the framework of data administration. Plug-ins for connecting different controls permit the flexible integration of heterogeneous machine parks.

Management with FORCAM FORCE™ Workbench



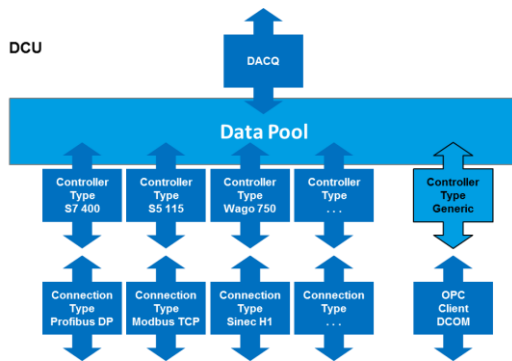
The configuration of control-specific plug-ins is carried out via the FORCAM FORCE™ Workbench web interface. Different parameters can be set depending on the controls. A comprehensive scripting language is available for normalizing the input signals. Templates for the configuration of workplaces (i.e. configuration of machines and systems) can also be set up in Workbench. To do this only one template needs to be set up and/or edited per machine type.



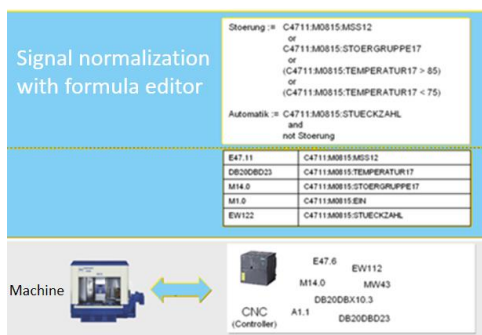
Normalization of communication and data



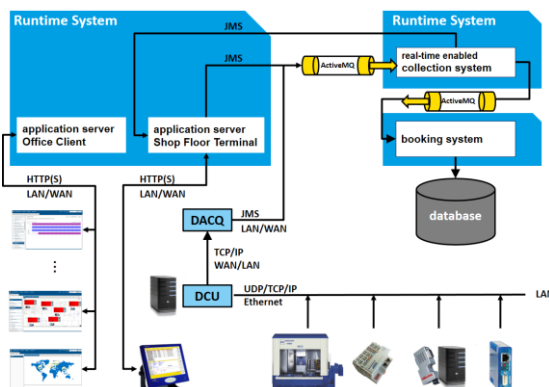
Normally, modern production facilities feature an Ethernet connection allowing them to be linked up to a Local Area Network (LAN) through which the machine data of interest can be selected using field bus logs. If an Ethernet connection isn't in place, it is usually possible to retrofit using so-called communication processors or adapters. Field bus controllers can also be used to record digital and analog signals directly.



Actual communication with the system controls takes place via FORCAM FORCE™ Data Collection Units (DCUs), that are adjusted to the different types of processors and field buses using plug-ins. DCUs function as middle-ware between machines and the FORCAM FORCE™ run-time environment. They ensure that only data relevant at a given time is transmitted with great efficiency (using event-driven differential telegrams).



DCUs can – depending on the type of controls and field bus – serve up to 100 controls of highly different designs simultaneously without presenting a special challenge to the PC hardware on which they are installed.



In order to increase the availability of a system such as this, two DCUs can be configured at any one time in such a way that they are monitoring each other. If a DCU paired up like this stops working, the DCU still functioning automatically assumes the tasks of the inoperative DCU (failover). Should the overarching system itself fail, then the Data Acquisition Unit (DACQ) downstream from the DCU buffers the data collected.

Error Code Mapping: Status Reason Mapping				
Variant: Variant 1				
Configuration			Status Reason Mapping	
Error Code	Error Text	Error Code Group	SWA Address	Status Reason
50001	Alarm1	Group1	IB1	STO - Disturbance
50002	Alarm2	Group2	IB2	999 - Undefined stoppage
3	Inverter Z-Axis 1 A.2	Electrical Error	ID0.1	999 - Undefined stoppage
4	Missing Parts Track 2	Logistic Error	ID0.2	STO - Disturbance
5	Inverter Z-Axis 1 A.2	Electrical Error	ID0.3	

Error code mapping (**Variation 1: Bit addressing**) enables external events to be assigned to error code groups, error codes and free text using importable tables.

Using an extremely rapid formula interpreter (DACQ scripting), any machine data can be translated (normalized) in the DACQ into standardized information, for example into quantities, production facility operating states like "production", "setup", "error", etc. Only once this information has been detached from machine type and specification is it transferred to the overarching system, where it is processed further before it is usually stored in the system's database. So-called error code mapping allows the simple allocation of system error signals to error texts and for them to be classified in a one-tier error hierarchy.

Standardized machine reports

Report	Function
Machine counter report	Incrementation of the machine counter by a specific number, specifying the counter number.
Machine stroke report	A number of machine cycles or strokes that is multiplied by the quantity factor per machine cycle or the stroke factor or the cavity to determine the quantity produced. This factor is typically saved with the work process.
Machine quantity report	A production result qualified by the machine or system itself as yield, waste quantity or rework quantity. This report is still not supported by the system configurations supplied as standard (templates of the rule engine) in version 5.4.5. It can be displayed to production personnel on the Shop Floor Terminal with a corresponding logic unit as recommended quantity. Direct input without confirmation by production personnel could therefore be possible in future
Machine status report	Report of a machine status (production or stoppage)

The reports standardized by the machine interface can be viewed in the “Reports (workplace view)” report.

Machine counter report and machine stroke report

Time stamp	Message	Workplace	Order	Operation	Details (Status)	Number	Yield Qty.	Scrap Qty.	Rework Qty.
Apr 8, 2016 11:00:39 AM	Machine Counter	MC760-2			Cnt. No.: 1	120	0	0	^
Apr 8, 2016 10:30:35 AM	Machine Counter	MC760-1			Cnt. No.: 1	30	0	0	
Apr 8, 2016 10:30:34 AM	Machine Hits	MH760-1				15	0	0	
Apr 8, 2016 10:00:37 AM	Machine Hits	MH760-2				60	0	0	

Machine quantity report

Time stamp	Message	Workplace	Order	Operation	Details (Status)	Number	Yield Qty.	Scrap Qty.	Rework Qty.
May 27, 2016 10:58:43 AM	Machine Quantity	Workplace2			Generic failure	0	0	1	0

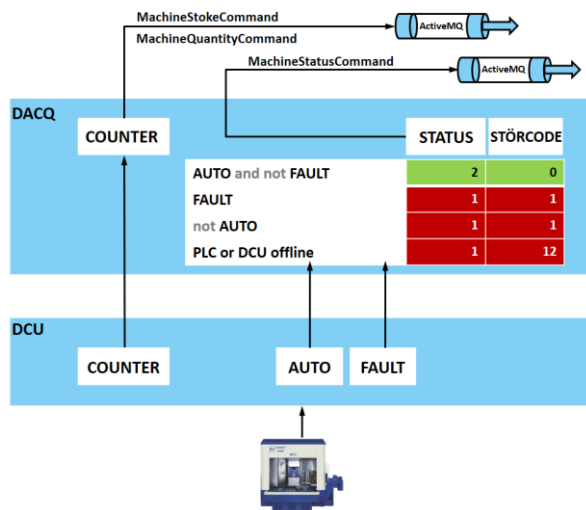
Machine status report

Time stamp	Message	Workplace	Order	Operation	Details (Status)
Jul 20, 2016 12:18:33 PM	Machine State	MC760-1			Undefined stoppage
Jul 20, 2016 12:08:30 PM	Machine State	MC760-1			Production
Jul 20, 2016 11:33:29 AM	Machine State	MC760-1			Undefined stoppage
Jul 20, 2016 11:33:28 AM	Machine State	MC760-2			Undefined stoppage
Jul 20, 2016 11:18:25 AM	Machine State	MC760-2			Production

It is important to ensure a consistent system configuration so that the OEE templates for data collection and booking logic are able to process the reports transmitted by the Shop Floor Terminal.

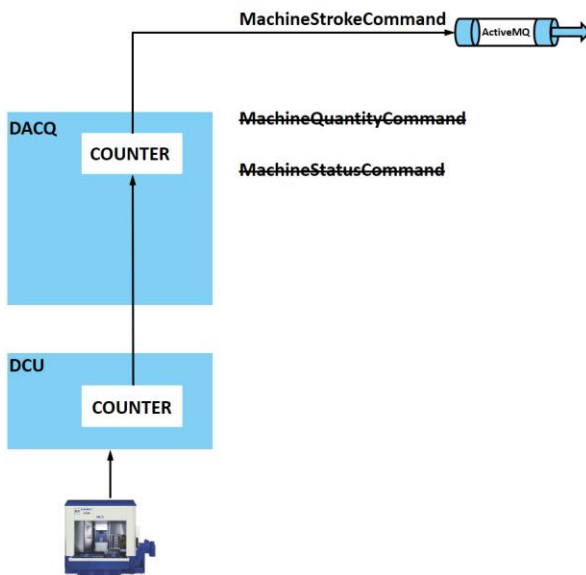
DACQ Templates

DACQ scripting permits the flexible combination of system control signals in order to determine the status of a machine. Four templates are supplied as standard, two templates for the FORCAM IO Box and two sample templates for a Siemens SPS. In each case there is one template for the OEE Standard (or OEE operation-free) in which the collection system anticipates status reports from machines, and one for OEE Auto Status in which the collection system determines the status of machines from the reported stroke signals.



OEE Standard

The OEE Standard templates supplied anticipate a COUNTER signal (strokes or quantities) from the machine to determine the quantity produced, and two other signals AUTO and FAULT for determining the operating state of the machine or system. When AUTO and not FAULT are set, this is interpreted as production, otherwise as stoppage. The determined status is sent to the collection system as a MachineStatusCommand. Depending on whether the counter signal refers to quantities or strokes, the MachineCountCommand or MachineStrokeCommand needs to be commented out in the script. If the DCU or system/machine controls are unavailable, stoppage status with error code 12 is generated.



OEE Auto Status

In the OEE Auto Status templates supplied, only a COUNTER signal is anticipated from the machine to determine the quantity produced. This does not refer to quantity reports from machines but to stroke reports that need to be multiplied by a stroke factor to determine the quantity actually produced. Accordingly, the DACQ script will only send machine stroke reports and no machine quantity reports. Since machine status is determined from the stroke reports only once in the collection system, the DACQ similarly does not send machine status reports.

Feature Specification

Special Features

- Automatic allocation of machine alerts to operating states
- High performance capability despite low processor requirements (up to 100 machines per data collection computer)
- High scalability due to virtually unlimited number of data collection computers (DCUs)
- Server-based with parameterization in web browser
- Plug-in concept for highly diverse control and report types
- High availability due to failover concept
- High data security due to consistent queuing
- Flexible combination of system control signals to determine machine status (via DACQ scripting)
- Direct selection of control signals via supported reports without a go-between (e.g. OPC server or MT Connect)

Functions

- Selection of data from supported controllers
- Translation and connection of data via a formula interpreter
- Event-based and highly efficient transfer of data to overarching system
- Configuration of evaluable "normalized" operating states
- Error code mapping