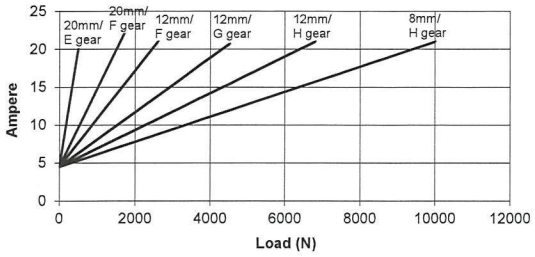
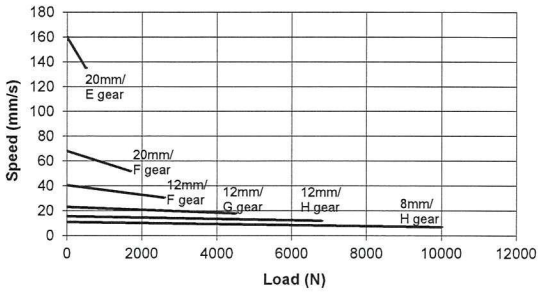
Speed and current curves:

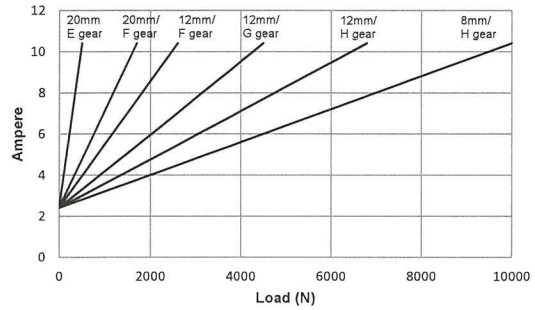
LA36 12V motor current vs. load



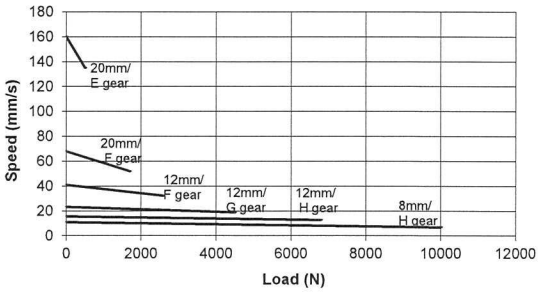
LA36 12V motor speed vs. load



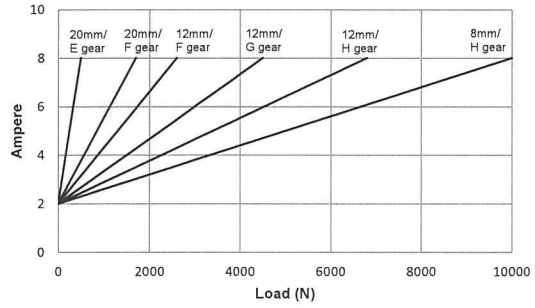
LA36 24V motor current vs. load



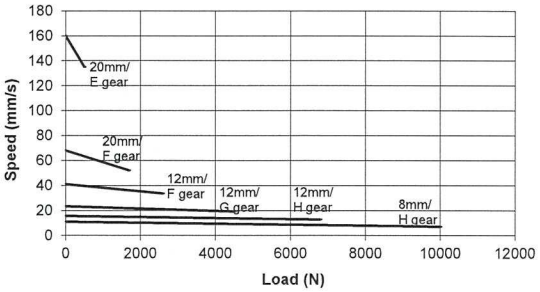
LA36 24V motor speed vs. load



LA36 36V motor current vs. load



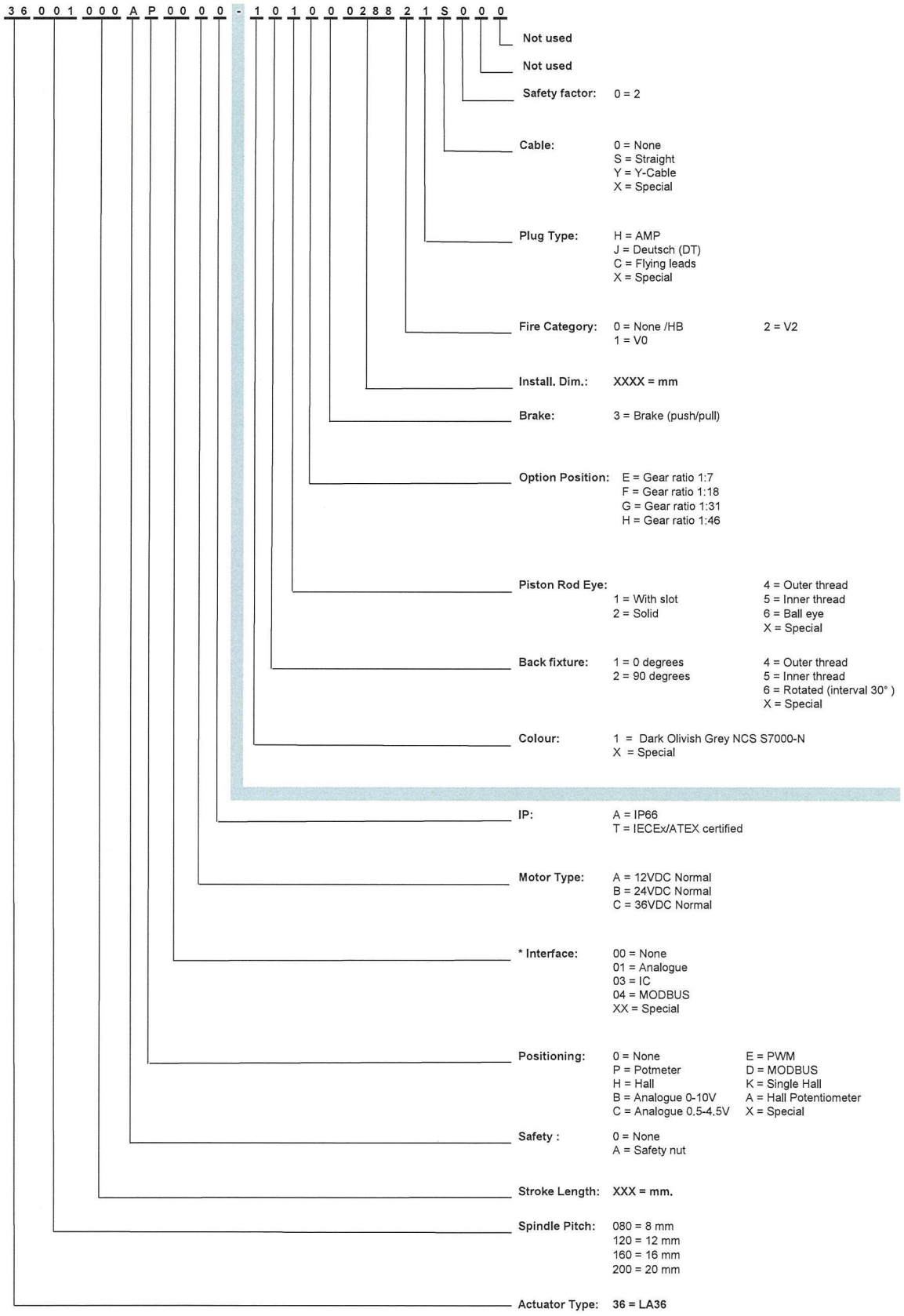
LA36 36V motor speed vs. load



All measurements above describe the spindle pitch (e.g. 20mm) and the gear type (e.g. E gear) of the actuator.

Speed and current are based on a nominal power supply of 12, 24, 36VDC.

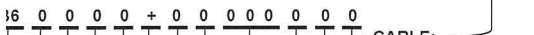
LA36
Ordering example Econ:



*

iFLEX options:	IC	LINbus	Modbus	Parallel
LA36 actuator:	✓	✓	✓	✓

LA36
Ordering example:



CABLE:
0 = No cable
1 = 1.5 m power cable (0367046-1500)
2 = 5 m power cable (0367046-5000)
3 = 0.2 m power cable with AMP connector (0367006)
4 = 1.5 m power and 1.5 signal (0367046-1500+0367049-1500)
5 = 5 m power and 5 m signal (0367046-5000+0367049-5000)
6 = 1.5 m Y-cable, power and signal in one (0367020)
7 = 5 m power Cable + Data cable M12x1 (Bus)

IP-DEGREE:
2 = Standard (IP66)
8 = IECEx/ATEX certified

MOTOR TYPE:
A = 12 V DC with clutch
B = 24 V DC with clutch
C = 36 V DC with clutch

STROKE LENGTH:
XXX = mm Acme spindle: 100, 150....999 mm

FEEDBACK:
0 = Standard (No feedback)

Standard feedback
B = Analogue feedback 0 - 10V
C = Analogue feedback 0.5 - 4.5V
H = Dual Hall
P = Potentiometer

IFLEX feedback
D = Bus
1 = Single Hall
2 = Analogue feedback 0 - 10V
3 = Analogue feedback 0.5 - 4.5V
4 = Feedback 4-20mA
5 = PWM 10-90%
6 = PWM 20-80%

Magnet

END STOP:
0 = Without limit switches
1 = With limit switches
2 = With limit switches and end-stop signals
3 = CS36
4 = CS36 with end-stop signals
5 = With potential free limit switches
7 = IC Basic A = Modbus : See Chapter 5.12.2
8 = IC Advanced B = Linbus
9 = IC Parallel

SAFETY NUT
+ = Standard
S = With safety nut - only in push

PISTON ROD EYE:

0 = M20 X 1 female adapter	0361016
1 = ø 12.9 mm hole, for 1/2" pin	0361018-B
2 = ø 12.2 mm hole, for 12 mm pin	0361109-B
3 = M12 X 1.75 male adapter	0361224
4 = M16 X 1.5 male adapter	0361135
5 = ø 12.2 hole with slot (Like LA34)	0361138
A = ø 12.2 hole with slot AISI 304	0361260
B = ø 12.9 hole with slot AISI 304	0361275
C = ø 12 H7 Ledøje AISI 304	0361350
D = ø 16 H7 Ledøje AISI 304	0361351

BACK FIXTURE:

0 = M20 X 1 female adapter	0361128
1 = ø 12.9 mm hole, for 1/2" pin	0361129
2 = ø 12.9 mm hole turned 90°, for 1/2" pin	0361129
3 = ø 12.2 mm hole, for 12 mm pin	0361119
4 = ø 12.2 mm hole turned 90°, for 12 mm pin	0361119
5 = M12 X 1.75 male adapter	0361126
6 = M16 X 1.5 male adapter	0361247
7 = ø 12.2 hole with slot (Like LA34)	0361140
8 = ø 12.2 hole with slot (LA34) turned 90°	0361140
A = ø 12.2 hole with slot AISI 304	0361261
B = ø 12.2 hole with slot AISI 304 turned 90°	0361261
C = ø 12.9 hole with slot AISI 304	0361276
D = ø 12.9 hole with slot AISI 304 turned 90°	0361276

GEARBOX:

	2.5 mm pitch	8 mm pitch	12 mm pitch	20 mm pitch
A = Gear ratio 1 : 18	N.A.	N.A.	2,600 N	1,700 N
B = Gear ratio 1 : 31	N.A.	N.A.	4,500 N	N.A.
C = Gear ratio 1 : 46	10,000 N	10,000 N	6,800 N	N.A.
F = Gear ratio 1 : 7	N.A.	N.A.	N.A.	500N

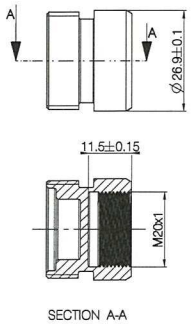
SPINDLE TYPE:
1 = 1-threaded acme spindle (2.5 mm pitch)
2 = 2-threaded acme spindle (8 mm pitch)
3 = 3-threaded acme spindle (12 mm pitch)
5 = 5-threaded acme spindle (20 mm pitch)
A = 2 + prepared for adjustable reed limit switches (on outer tube)
C = 3 + prepared for adjustable reed limit switches (on outer tube)
E = 5 + prepared for adjustable reed limit switches (on outer tube)

ACTUATOR TYPE:
36 = LA36

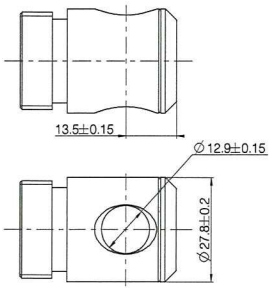
When ordering standard stroke length with endstop 1, 2, 3 or 4 the stroke length will be up to 4 mm shorter.

LA36 Piston Rod Eye

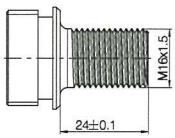
Option "0"
LINAK P/N: 0361016



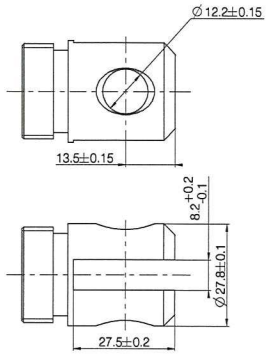
Option "1"
LINAK P/N: 0361018



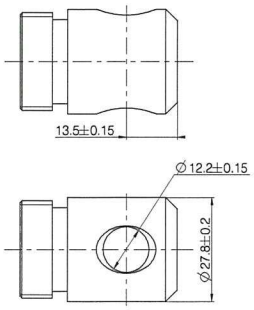
Option "4"
LINAK P/N: 0361135



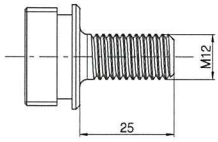
Option "5"
LINAK P/N: 0361138



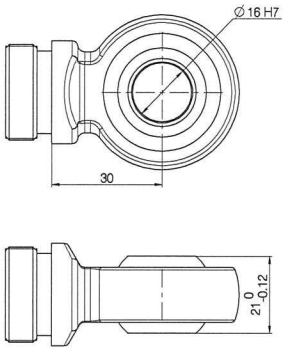
Option "2"
LINAK P/N: 0361109



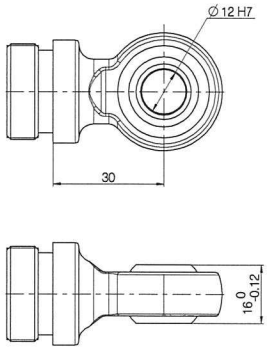
Option "3"
LINAK P/N: 0361224



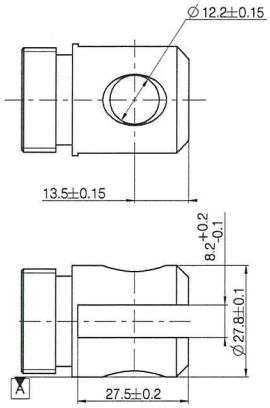
Option "D"
LINAK P/N: 0361351



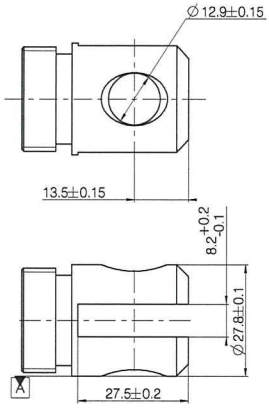
Option "C"
LINAK P/N: 0361285
10KN = Max. load 6800 N in pull



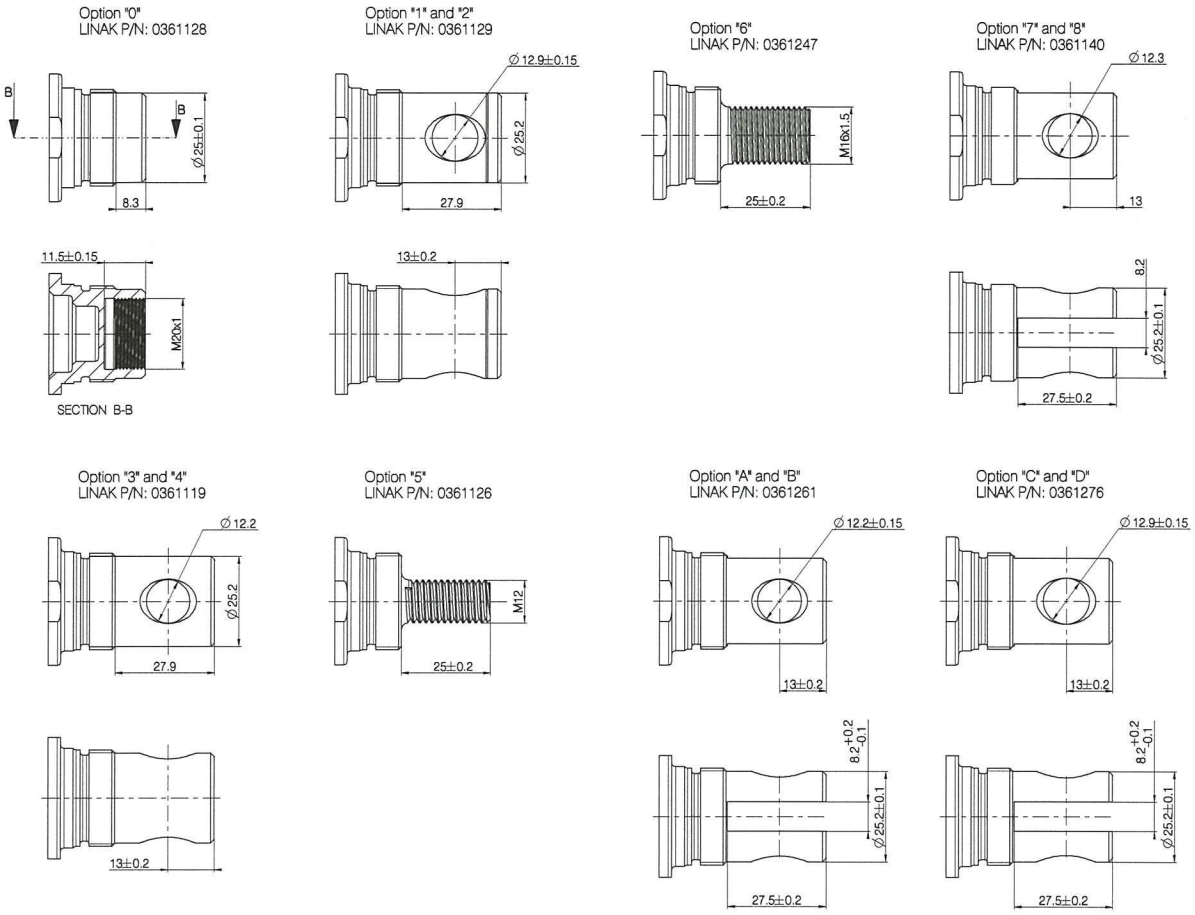
Option "A"
LINAK P/N: 0361260



Option "B"
LINAK P/N: 0361275

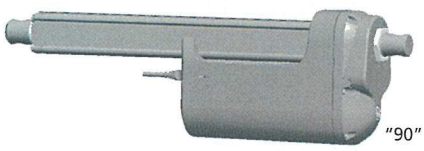


LA36 Back Fixture



LA36 Back Fixture Orientation

"0" Degrees



"90" Degrees

"30" Degrees

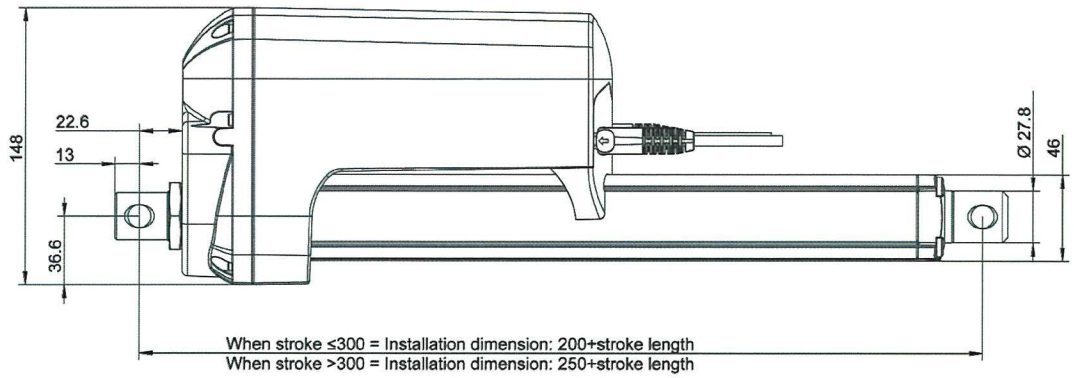
"60" Degrees

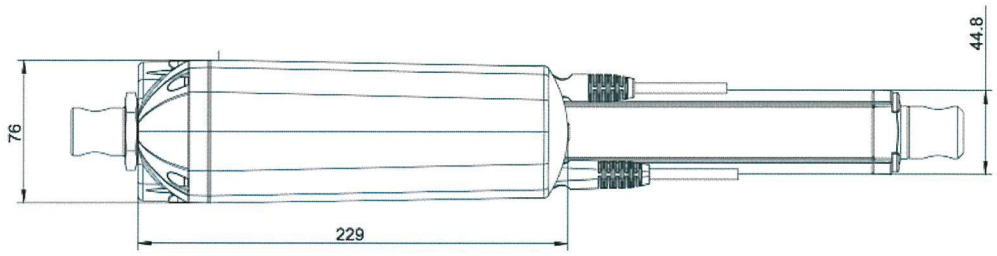
"120" Degrees

"150" Degrees

NB. All with tolerance of ±4°

LA36 built-in dimensions:





I/O specifications: Power supply - Motor.

Item	Specification	Comment
Power supply		
Input voltage	12 VDC, ± 20% 24 VDC, ± 10% 36 VDC, ± 10%	Cable dimension: 2 x 2.2 mm² (2 x AWG14) for all different voltages.
Duty cycle	20% at max. load	Ambient temperature 25° C
Current consumption	2 - 23 Amp. depending on load and voltage (see graphs)	
Connection	To extend actuator: Connect Brown to positive Connect Blue to negative To retract actuator: Connect Brown to negative Connect Blue to positive	Actuator direction can be controlled with a double-throw switch with the middle position *"off"*. Please note that for all iFLEX options the power supply must NOT be switched between plus and minus for extending or retracting the actuator.

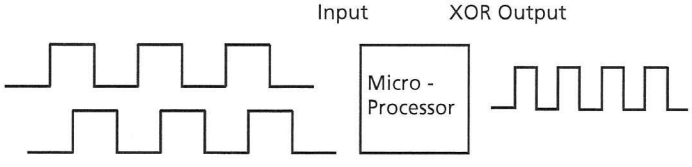
** For differentiated duty cycle see "Usage"*

Positioning feedback – Potentiometer.

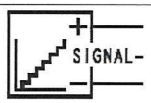
Item	Specification	Comment
Absolute positioning		
Potentiometer	Bourns 0-10 K ohm A 5%, 10-Turn	Type: 3540 Wirewound
Output range with 8 mm spindle pitch	0 K ohm = 0 mm stroke 10 K ohm = 333 mm stroke	The same for all LA36 8 mm models.e.g. 166.6 mm stroke = 5 Kohm.
Output range with 12 mm spindle pitch	0 K ohm = 0 mm stroke 10 K ohm = 500 mm stroke	The same for all LA36 12 mm models.e.g. 250 mm stroke = 5 Kohm.
Output range with 20 mm spindle pitch	0 K ohm = 0 mm stroke 10 K ohm = 833 mm stroke	The same for all LA36 20mm models.e.g. 416.5 mm stroke = 5 Kohm.
Linearity	± 0.25%	
Output protection	1 Kohm protection resistor	
Connection	Common - = Black +10V exitation = White 0 = 10V out = Violet	+10V or other value

NOTE: Please note that Potentiometer is not possible on varients with fast gear (Spindle pitch 20 mm, H Gear).

Positioning feedback – Hall sensors

Item	Specification	Comment
Relative positioning		
Signal description	Can be used for positioning.	
Input Voltage	12 – 36 V DC	Cable dimension: 6 x 0.5 mm² (6 x AWG20) for all different voltages.
Output voltage	Always the same as input voltage Note: max. output voltage 24V DC 12V : 11V ± 1V 24V : 23V ± 1V 36V : 35V ± 1V	
Resolution (Distance the piston rod moves per count)	LA362C: Actuator = 0.1 mm per count LA363C: Actuator = 0.2 mm per count LA363B: Actuator = 0.3 mm per count LA363A: Actuator = 0.4 mm per count LA365A: Actuator = 0.7 mm per count Movement per single Hall pulse: LA362C Actuator = 0.4 mm per pulse LA363C Actuator = 0.7 mm per pulse LA363B Actuator = 1.0 mm per pulse LA363A Actuator = 1.7 mm per pulse LA365A Actuator = 2.9 mm per pulse	The Hall sensor signals are generated by the turning of the actuator gearing. These signals can be fed into PLC. The PLC quadrature signals (fig. 1 below) can be used to register position of the piston rod. N.B. For more precise measurements, please contact LINAK A/S.
Frequency	Frequency is 14-26 Hz on XOR output depending on load. Every pulse is "ON" for 10 ms	Low frequency with a high load.Higher frequency with no load.
Current consumption (standby)	15 mA	When actuator is not running.
Switching capacity	Max. 12 mA	Max. 680n F
Connection	XOR Hall output = Violet Signal GND = Black	
Diagram of Single Hall:	 Fig. 1	

I/O Specifications: Analogue feedback.

Item	Specification	Comment
Description	The actuator can be equipped with electronic circuit that gives an analog feedback signal when the actuator moves	
Input voltage	12 - 36 V DC	Feedback circuit to be powered 1 second before motor runs, and until 1 second after the motor has stopped. Cable dimension 6 x 0,5 mm^2 (6 x AWG20)
Output voltage	0 -10 V (Option B) 0V = Fully retracted 10V = Fully extended 0.5 - 4.5V (Option C) 0.5V = Fully retracted 4.5V = Fully extended	+/- 0.2 V
Current consumption	Max. 40 mA	Also when actuator is not running
Connection	Supply: Brown Supply : Blue Signal power: White/Red Signal: Violet Signal GND: Black	Use cable 0367003-XXXX
Combinations	The Absolute positioning must be combined with limit switches. Can be combined with endstop signal.	

Note: It is recommendable to have the actuator to activate its limit switches on a regular basis.
Endstop signal: max 20 mA available.

I/O Specification: IC (Basic and Advanced)

Item	Specification	Comment
Description	Easy to use interface with integrated power electronics (H-bridge) for direct IC connection. Soft start of the actuator	
Power supply		
Input voltage	12VDC ± 20% 24VDC ± 10%	Cable dimension 2 x 2 mm^2 (2 x AWG14) for all voltages
Current consumption	12V, 4-26A depending on load 24V, 2-13A depending on load	
Duty cycle	20% at maximum load	
Power connection	Connect Brown to positive Connect Blue to negative	
Input: Signals to the actuator		
Outwards direction	Extends the actuator FW - Red (Pin 2)	
Inwards direction	Retracts the actuator BW - Black (Pin 1)	
On/off voltages	> 67% of V_{IN} = ON < 33% of V_{IN} = OFF	
Input current	> 10 mA	
Current consumption (standby)	70 mA	When actuator is not running.
Output: Signals from the actuator		
Signal GND	Minimising signal noise	To be used with all signal outputs
Actuator fully extended (OUT)	Signal when endstop switch in extended position is activated IN = Yellow (Pin 5)	Source current max. 100 mA Output voltage min. V_{IN} - 1V
Actuator fully retracted (IN)	Signal when endstop switch in retracted position is activated OUT = Green (Pin 6)	

Feedback: IC (Basic)

Item	Specification	Comment
Feedback, Hall	Single Hall signal	XOR: See fig. 1, page 9 Typical: Input voltage -1V Example on 24V version: Output voltage on IN = 23V (± 0.5V) Output voltage on OUT = 23V (± 0.5V) Source current max. 12mA
Feedback, Voltage	0 - 10V / 0.5 - 4.5V	Ripple max. 200mV Transaction delay max. 20ms Linear feedback 0.5% Source current max. 1mA
Connection		See User manual

Feedback: IC (Advanced).

Item	Specification	Comment
Feedback, PWM	Frequency: Up to 150 Hz ± 5Hz Duty cycle: Any low/high combination between 0 and 100 percent	Output voltage: (V_{IN} - 1V) ± 1V Open Drain source current max. 12 mA
Feedback, Hall	Single Hall signal	XOR: See fig. 1, page 9 Typical: Input voltage -1V Example on 24V version: Output voltage on IN = 23V (± 0.5V) Output voltage on OUT = 23V (± 0.5V) Source current max. 12mA
Feedback, Voltage	Any low/high voltage combination between 0 and 10 volts	Ripple max. 200 mV Transaction delay max. 20 ms Linear feedback 0.5% Source current max. 1mA
Feedback, Current	Any low/high current combination between 4 and 20 mA	Transaction delay max. 20 ms Linear feedback 0.5% Output: Source Serial resistance: 12V max. 300 Ohm 24V max. 900 Ohm
Connection		See user manual

I/O Specification: Parallel

Item	Specification	Comment
Description	The parallel drive option supports up to 8 actuators	
Power supply	12 V DC ± 20% 24 V DC ± 10%	Cable dimension 2 x 2 mm² (2 x AWG14) for all voltages
Current consumption	12 V, 4 - 26 A depending on load 24 V, 2 - 13 A depending on load	Consumption per actuator
Feedback	No feedback available during parallel drive	
Power connections	Black (Pin 1): Enable backward Red (Pin 2): Enable forward White (Pin 3): Signal GND Violet (Pin 4): Inter communication Yellow (Pin 5): Endstop signal out Green (Pin 6): Endstop signal in	Cable dimension 6 x 0.5 mm² (6 x AWG20) See user manual

Environmental test – Climatic

Test	Specification	Comment	TRD number
Cold test	EN60068-2-1 (Ab)	Storage at low temperature: Temperature: -40°C Duration: 72h Not connected Tested at room temperature.	TRD0509
	EN60068-2-1 (Ad)	Operating at low temperature: Temperature: -30°C Duration: 2h Actuator is not activated/connected Tested at low temperature.	TRD0509
Dry Heat	EN60068-2-2 (Bb)	Storage at high temperature: Temperature: +90°C Duration: 72h Actuator is not activated/connected. Tested at room temperature	TRD0510
		Storage at high temperature: Temperature: +70°C Duration: 1000h Actuator is not activated/connected Tested at high temperature.	TRD0507
	EN60068-2-2 (Bd)	Operating at high temperature: Temperature: +60°C Int. max. 17% Duration:700h Actuator is activated Tested at high temperature.	
Change of temperature	EN60068-2-14 (Na)	Rapid change of temperature: High temperature: +100°C in 60 minutes. Low temperature: -30°C in 60 minutes. Transition time:<10 seconds Duration: 100 cycles Actuator is not activated/connected. Tested at room temperature.	TRD0501
	EN60068-2-14 (Nb)	Controlled change of temperature: Temperature change 5°C pr. minute High temperature: +70°C in 60 minutes. Low temperature: -30°C in 30 minutes. 130 minutes pr. Cycle. Duration: 1.000 cycles (90days) Actuator is not activated/connected. Tested at 250, 500 and 1,000 cycles at low and high temperatures.	TRD0508
Damp heat	EN60068-2-30 (Db)	Damp heat, Cyclic: Relative humidity: 93-98% High temperature: +55°C in 12 hours Low temperature: +25°C in 12 hours Duration: 21cycles * 24hours Actuator is not activated/connected Tested within 1 hour after condensation, That means after upper temperature has been reached.	TRD0505
	EN60068-2-3 (Ca)	Damp heat, Steady state: Relative humidity: 93-95% Temperature: +40 ±2°C Duration: 56 days Actuator is not activated/connected. Tested within one hour after exposure.	TRD0518
Salt mist.	EN60068-2-52 (Kb)	Salt spray test: Salt solution: 5% sodium chloride (NaCl) 4 spraying periods, each of 2 hours. Humidity storage 7 days after each. Actuator not activated/connected. Exposure time: 500 hours	TRD0506

Degrees of protection	EN60529 – IP66	IP6X - Dust: Dust-tight, No ingress of dust. Actuator is not activated.	TRD0514
		IPX6 – Water: Ingress of water in quantities causing harmful effects is not allowed. Duration: 100 litres pr. minute in 3 minutes Actuator is not activated.	TRD0513
		IPX6 –Connected actuator: Actuator is driving out and in for 3 min. 100(l/min) jet of water is placed at the wiper ring for 3 (min).	TRD0513
	DIN40050 – IP69K	IPX6 –Connected actuator and push 6800 (N) Actuator is driving out and in for 3 min. and Push 6800(N) at the end-pos. 100 (l/min.) jet of water is placed at the wiper ring for 3 min.	TRD0513
		High pressure cleaner: Water temperature: +80°C Water pressure: 80 bar Spray angle: 45° Spray distance: 100mm Duration: From any direction 10 seconds of spraying followed by 10 seconds rest. Actuator is not activated. Ingress of water in quantities causing harmful effects is not allowed.	TRD0512
	DUNK test	The actuator has been warmed up to 115°C for 20 hours. After this it is cooled down in 20°C saltwater. Cooling time: 5 minutes Opened for checking salt deposit and water.	TRD0515
Chemicals	BS7691 / 96hours	Diesel 100% Hydraulic oil 100% Ethylene Glucol 50% Urea Nitrogen saturated solution Liquid lime 10% (Super- Cal) NPK Fertilizer (NPK 16-4-12) saturated Tested for corrosion.	TRD0525

Environmental test - Mechanical

Test	Specification	Comment	TRD number
Free fall		Free fall from all sides: Height of fall: 0.4 meter onto steel. Actuator not activated/connected.	TRD0511
Vibration	EN60068-2-36 (Fdb)	Random vibration: Short time test:6.29g RMS Actuator is not connected Long time test: 7.21g RMS Actuator is not connected Duration: 2 hours in each direction	TRD0502
	EN 60068-2-6 (Fc)	Sinus vibration: Frequency 5-25Hz: Amplitude = 3.3mm pp Frequency 25-200Hz: Acceleration 4g Number of directions: 3 (X-Z-Y) Duration: 2 hours in each direction. Actuator is not activated	TRD0517
Bump	EN60068-2-29 (Eb)	Bump test: Level: 40g Duration: 6 milliseconds Number of bumps: 500 shocks in each of 6 directions. Actuator is not connected.	TRD0503
Shock	EN60068-2-27 (Ea)	Shock test: Level: 100g Duration: 6 milliseconds Number of bumps: 3 shocks in each of 6 directions. Actuator is not connected.	TRD0504

Environmental test - Electrical

Test	Specification	Comment	TRD number
Power supply	ASAE EP455 (1990)	Operating voltages +10V - +16V Over voltage +26(V) / 5min. Reverse polarity –26(V) / 5min. Short circuit to ground 16 (V) / 5 min. Short circuit to supply 16(V) / 5 min.	TRD0522
HF-immunity	EN61000-6-2	Level: 30 V/m. at 26 MHz – 1000 mHz 80% 1 KHz	TRD0516
Emission	EN61000-6-4	Level is inside limits for 12 V motor	TRD0516
Insulation test		Level: 500 VAC/25-100hz for 1 minute	TRD0516
Automotive transients	ISO 7637	Load dump test only accepted on motor power connection.	TRD0521

Manual hand crank

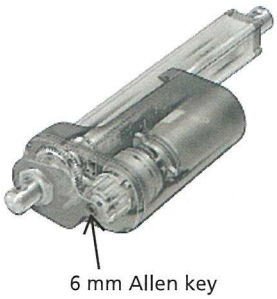
The manual hand crank can be used in the case of power failure.

The cover over the Allen Key socket must be unscrewed before the Allen Key can be inserted and the Hand Crank operated.

Hand Crank Torque: Max.16 Nm (at maximum load)

Piston Rod movement per turn

Gear A = 10.5 mm
Gear B = 6.0 mm
Gear C = 4.0 mm
Gear F = 27 mm



6 mm Allen key

Note:

- The power supply has to be disconnected during manual operation.
- If the actuator is operated as a Hand crank, it must be operated by hand or carefully by machine, otherwise there is a potential risk of overloading and hereby damaging the actuator. LA36 with CS or Modbus options only operate by hand.
- With stainless steel screws: 5 mm Allen Key

BILAG 07: Daugaard Elektronik – Anvendt visionkamera

Daugaard Elektronik ApS

Ringe den 07. august 2014

Vedr. EUDP roborudsugning og bidrag til projektet fra Daugaard Elektronik ApS

Efter at have modtaget information om projektet stod det klart, at DE´s primære indsats skulle fokusere på et visionssystem som kan medføre at udsugningen kan følge svejseflammen og dermed –røgen som skal udsuges. Lysbuen fra elektrosvejsning udsender et stærkt lys som selv markedets billigste webkamera kan detektere. Målsætningen for DE er derfor ved brug af markedets billigste komponenter samt specialdesignet software at frembringe et prisbilligt og enkelt design som kan løse opgaven. Proof-of-concept for designet blev leveret i 2013.

Følgende indgår I leverancen:

Network camera: Axis, type M10.

Digital I/O module: Advantech, type ADAM-6052.

Application Software: Daugaard Elektronik ApS., type DE_WeldingMonitor, V1.0.0.0

Næste skridt er at færdiggøre visionssystemet og optimere dets funktioner i samspil med de øvrige komponenter som indgår i det samlede system. Naturligvis under forudsætning af at det samlede system forekommer kommercielt attraktivt.

BILAG 08: Sick – Anvendt tilt sensor

SICK

Teknologisk Institut
Forskerparken 10
5220 Odense

28.08.2014

Sensorer til automatisk røgudsug

Der er benyttet flere typer sensorer til regulering af automatisk røgudsug.

I nedenstående forklares baggrund for udvælgelse af den enkelte sensor type.

Temperaturmåler TSP:
Denne sensor type er fremstillet i industrielt design og kan holde til intensiv brug i hårde miljøer.
Til automatisk røgudsug var ideen at den varme luft fra svejsestedet blev suget op til sugehovedet.
Her skulle 4 temperaturmålere registrere hvor den højeste temperatur forekomst var og regulere hovedets position herefter.
Praktiske forsøg viste at temperaturen i luftstrømmen var af beskeden størrelse og reguleringen derfor var minimal samtidigt med at opvarmning af sensor tog for lang tid.

Vinkelmåler NST6:
Denne sensor type er fremstillet i industrielt design og kan holde til intensiv brug i hårde miljøer.
For at optimere sug er det nødvendigt at kende position af sugehovedet, herunder tilt i forhold til et vandret plan.
Hertil benyttes vinkelssensor NST6. Denne fungerer som et waterpass og når libellen bevæger sig uden for 0 grader leveres afvigelse som et analog signal.

Vision sensor Inspector:
Denne sensor type er fremstillet i industrielt design og kan holde til intensiv brug i hårde miljøer.
Vision sensoren fungerer ved at en optisk kreds påvirkes af lys og danner et billede.
Indbygget avanceret algoritmer regner på det modtagne lys og omdanner dette til et simpelt signal som sendes til overordnet styring.
Inspector leveres med indbygget logik således forgår programmering og programafvikling i sensorens indbyggede computer.
Sensoren levere data, enten som et simpelt signal PNP som kan befinde sig i høj eller lav status, eller via et Ethernet interface hvor avanceret telegram opbygges til at passe til applikationen.

Link til yderligere information:

TSP temperaturmåler:
Datablad:
http://www.mysick.com/eCat.aspx?go=DataSheet&Cat=Row&At=Fa&Cult=English&Category=Dokumentationen&ProductID=44494

Brugermanual:
http://www.mysick.com/eCat.aspx?go=DataSheet&Cat=Row&At=Fa&Cult=English&Category=Dokumentationen&ProductID=44491

NST vinkelmåler:

Online datablad:

http://www.di-soric.com/produkt_en,,14941,detail.html

Vison sensor Inspector:

Brugermanual
http://www.mysick.com/eCat.aspx?go=DataSheet&Cat=Row&At=Fa&Cult=English&Category=Dokumentationen&ProductID=77340

Med venlig hilsen
SICK A/S

Henrik Bostrup
Produktchef

BILAG 09: Acceleratiometer

3-Axis, ±2 *g*/±4 *g*/±8 *g*/±16 *g* Digital Accelerometer

Data Sheet **ADXL345**

FEATURES

- **Ultralow power: as low as 23 µA in measurement mode and 0.1 µA in standby mode at V_S = 2.5 V (typical)**
- **Power consumption scales automatically with bandwidth**
- **User-selectable resolution**
  - **Fixed 10-bit resolution**
  - **Full resolution, where resolution increases with *g* range, up to 13-bit resolution at ±16 *g* (maintaining 4 mg/LSB scale factor in all *g* ranges)**
- **Patent pending, embedded memory management system with FIFO technology minimizes host processor load**
- **Single tap/double tap detection**
- **Activity/inactivity monitoring**
- **Free-fall detection**
- **Supply voltage range: 2.0 V to 3.6 V**
- **I/O voltage range: 1.7 V to V_S**
- **SPI (3- and 4-wire) and I²C digital interfaces**
- **Flexible interrupt modes mappable to either interrupt pin**
- **Measurement ranges selectable via serial command**
- **Bandwidth selectable via serial command**
- **Wide temperature range (−40°C to +85°C)**
- **10,000 *g* shock survival**
- **Pb free/RoHS compliant**
- **Small and thin: 3 mm × 5 mm × 1 mm LGA package**

APPLICATIONS

- **Handsets**
- **Medical instrumentation**
- **Gaming and pointing devices**
- **Industrial instrumentation**
- **Personal navigation devices**
- **Hard disk drive (HDD) protection**

GENERAL DESCRIPTION

The ADXL345 is a small, thin, ultralow power, 3-axis accelerometer with high resolution (13-bit) measurement at up to ±16 *g*. Digital output data is formatted as 16-bit twos complement and is accessible through either a SPI (3- or 4-wire) or I²C digital interface.

The ADXL345 is well suited for mobile device applications. It measures the static acceleration of gravity in tilt-sensing applications, as well as dynamic acceleration resulting from motion or shock. Its high resolution (3.9 mg/LSB) enables measurement of inclination changes less than 1.0°.

Several special sensing functions are provided. Activity and inactivity sensing detect the presence or lack of motion by comparing the acceleration on any axis with user-set thresholds. Tap sensing detects single and double taps in any direction. Free-fall sensing detects if the device is falling. These functions can be mapped individually to either of two interrupt output pins. An integrated, patent pending memory management system with a 32-level first in, first out (FIFO) buffer can be used to store data to minimize host processor activity and lower overall system power consumption.

Low power modes enable intelligent motion-based power management with threshold sensing and active acceleration measurement at extremely low power dissipation.

The ADXL345 is supplied in a small, thin, 3 mm × 5 mm × 1 mm, 14-lead, plastic package.

FUNCTIONAL BLOCK DIAGRAM

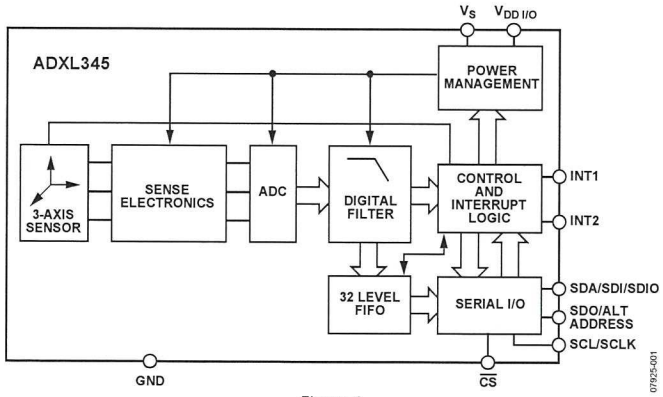
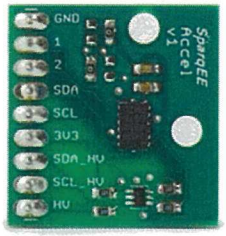


Figure 1.

Rev. D **Document Feedback**

One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, U.S.A.
Tel: 781.329.4700
Technical Support www.analog.com

SparqEE ACCELv1.0 Datasheet



The ACCELv1.0 is provided as a wide input range accelerometer based on the Analog Devices ADXL345. Please see the ADXL345 datasheet for further specifications and features.

Most development environments provide a 3.3V output but sometimes utilize higher voltage I/O lines (Arduino is 5V I/O, Raspberry Pi is 3.3V I/O). This board can support any voltage I/O lines up to 15V.

- SparqEE: http://sparqee.com/
- Accelerometer: http://www.sparqee.com/products/sparqee-accel
- Forum: http://www.forum.sparqee.com/
- Code/Drivers: http://www.sparqee.com/code-cell/

Features:

- 3.3-15V I2C line operation
- 2 Interrupt lines

Specification:

Power

- Supply: 2.0 to 3.6V (for 3.3V)
- I/O: 2.0 to 15V

Physical

- 24mm x 23mm (0.94" x 0.91")
- Weight: 0.10oz

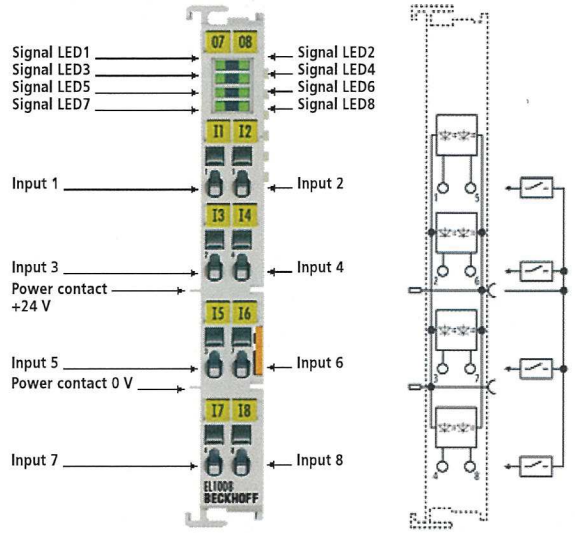
Operating Temperature

- -40°C to +85°C

Pin Assignment:

Pin	#	Function	Optional/Required	Voltage	I/O
GND	1	Ground connection	Required	0	I
1	2	Interrupt 1 from ADXL345	Optional	3.3V	O
2	3	Interrupt 2 from ADXL345	Optional	3.3V	O
SDA	4	I2C SDA line	Optional	3.3V	I/O

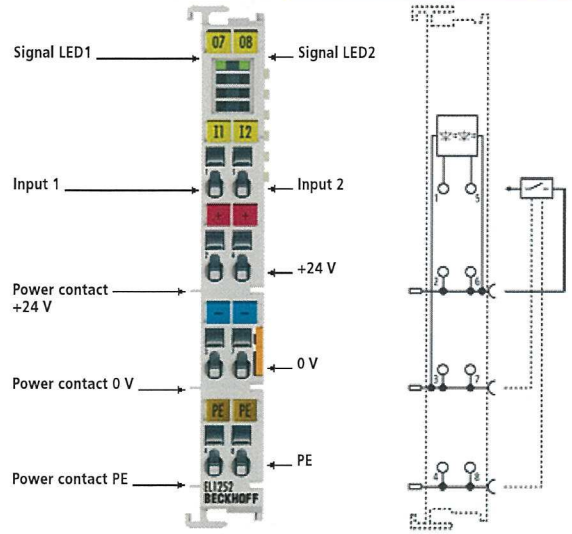
BILAG 10: Beckhoff



EL1008 | 8-channel digital input terminal 24 V DC, 3 ms

The EL1008 digital input terminal acquires the binary control signals from the process level and transmits them, in an electrically isolated form, to the higher-level automation unit. Digital input terminals from the EL100x series have a 3 ms input filter. The EtherCAT Terminals indicate their state via an LED.

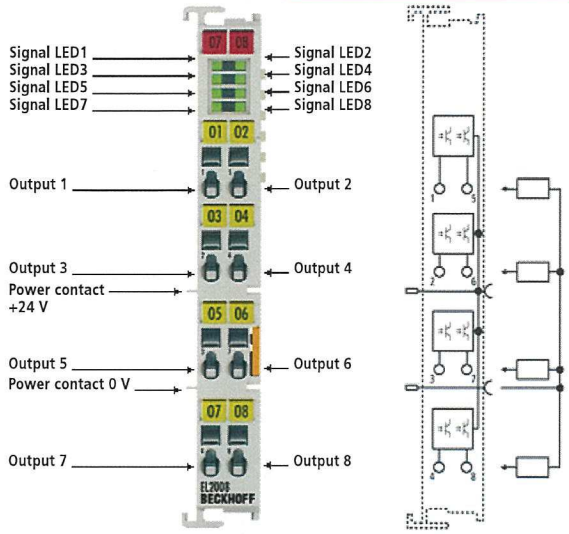
Technical data	EL1008 | ES1008
Connection technology	1-wire
Specification	EN 61131-2, type 1/3
Number of inputs	8
Nominal voltage	24 V DC (-15 %/+20 %)
"0" signal voltage	-3…+5 V (EN 61131-2, type 3)
"1" signal voltage	15…30 V (EN 61131-2, type 3)
Input current	typ. 3 mA (EN 61131-2, type 3)
Input filter	typ. 3.0 ms
Distributed clocks	–
Current consumption power contacts	typ. 2 mA + load
Current consumption E-bus	typ. 90 mA
Electrical isolation	500 V (E-bus/field potential)
Bit width in the process image	8 inputs
Configuration	no address or configuration setting
Special features	standard input terminal for bouncing signals (filter 3 ms)
Weight	approx. 55 g
Operating/storage temperature	0…+55 °C/-25…+85 °C
Relative humidity	95 %, no condensation
Vibration/shock resistance	conforms to EN 60068-2-6/EN 60068-2-27
EMC immunity/emission	conforms to EN 61000-6-2/EN 61000-6-4
Protect. class/installation pos.	IP 20/see documentation
Pluggable wiring	for all ESxxxx terminals
Approvals	CE, UL, Ex



EL1252 | 2-channel digital input terminal with time stamp

The EL1252 digital input terminal acquires the fast binary control signals from the process level and transmits them, in an electrically isolated form, to the controller. The signals are furnished with a time stamp that identifies the time of the last edge change with a resolution of 1 ns. This technology enables signals to be traced exactly over time and synchronised with the distributed clocks across the system. With this technology, machine-wide parallel hardware wiring of digital inputs or encoder signals for synchronisation purposes is often no longer required. In conjunction with the EL2252 EtherCAT Terminal (digital output terminal with time stamp), the EL1252 enables responses with equidistant time intervals, largely independent of the bus cycle time.

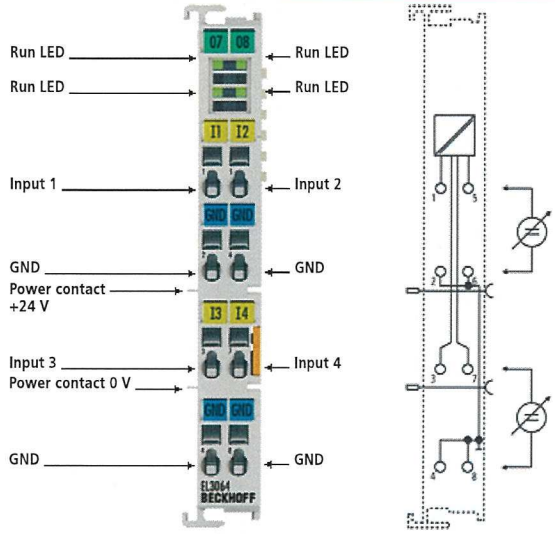
Technical data	EL1252 | ES1252
Connection technology	4-wire
Specification	similar to EN 61131-2, type 3, "0": -3…5 V DC, "1": 11…30 V DC, typ. 3 mA input current
Number of inputs	2
Nominal voltage	24 V DC (-15 %/+20 %)
"0" signal voltage	-3…+5 V (similar to EN 61131-2, type 3)
"1" signal voltage	11…30 V (similar to EN 61131-2, type 3)
Input current	typ. 3 mA (similar to EN 61131-2, type 3)
Input filter	typ. < 1 µs
Resolution time stamp	1 ns (channel 0/1)
Precision of time stamp in the terminal	10 ns (+ input delay)
Distributed clock precision	<< 1 µs
Input delay TON/TOFF	< 1 µs
Distributed clocks	yes
Time resolution signal	1 ns
Current consumption power contacts	typ. 6 mA + load
Current consumption E-bus	typ. 110 mA
Electrical isolation	500 V (E-bus/field potential)
Bit width in the process image	2 inputs + 36 byte time stamp
Configuration	no address or configuration setting
Special features	time stamp, latch last edge
Weight	approx. 55 g
Operating/storage temperature	0…+55 °C/-25…+85 °C
Relative humidity	95 %, no condensation
Vibration/shock resistance	conforms to EN 60068-2-6/EN 60068-2-27
EMC immunity/emission	conforms to EN 61000-6-2/EN 61000-6-4
Protect. class/installation pos.	IP 20/variable
Pluggable wiring	for all ESxxxx terminals
Approvals	CE, UL, Ex



EL2008 | 8-channel digital output terminal 24 V DC, 0.5 A

The EL2008 digital output terminal connects the binary control signals from the automation unit on to the actuators at the process level with electrical isolation. The EtherCAT Terminal indicates its signal state via an LED.

Technical data	EL2008
Connection technology	1-wire
Number of outputs	8
Rated load voltage	24 V DC (-15 %/+20 %)
Load type	ohmic, inductive, lamp load
Distributed clocks	–
Max. output current	0.5 A (short-circuit-proof) per channel
Short circuit current	typ. < 2 A
Reverse voltage protection	yes
Breaking energy	< 150 mJ/channel
Switching times	typ. Ton: 60 µs, typ. Toff: 300 µs
Current consumption E-bus	typ. 110 mA
Electrical isolation	500 V (E-bus/field potential)
Current consumption power contacts	typ. 15 mA + load
Bit width in the process image	8 outputs
Configuration	no address or configuration setting
Weight	approx. 55 g
Operating/storage temperature	0…+55 °C/-25…+85 °C
Relative humidity	95 %, no condensation
Vibration/shock resistance	conforms to EN 60068-2-6/EN 60068-2-27
EMC immunity/emission	conforms to EN 61000-6-2/EN 61000-6-4
Protect. class/installation pos.	IP 20/see documentation
Pluggable wiring	for all ESxxxx terminals
Approvals	CE, UL, Ex



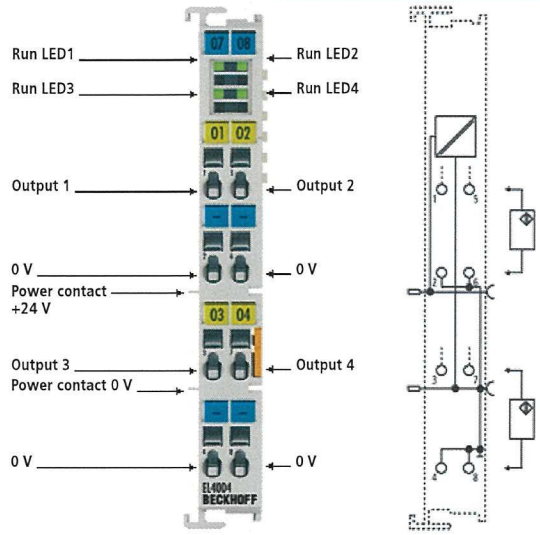
Top view

Contact assembly

EL3064 | 4-channel analog input terminal 0…10 V, single-ended, 12 bits

The EL3064 analog input terminal processes signals in the range between 0 and 10 V. The voltage is digitised with a resolution of 12 bits and is transmitted (electrically isolated) to the higher-level automation device. The EL3064 EtherCAT Terminal features 2-wire conductors for the four single-ended inputs with a common internal ground potential. The power contacts are connected through. The signal state of the EtherCAT Terminal is indicated by light emitting diodes.

Technical data	EL3064 | ES3064
Number of inputs	4 (single-ended)
Power supply	via the E-bus
Technology	single-ended
Signal voltage	0…10 V
Distributed clocks	–
Internal resistance	> 130 kΩ
Input filter limit frequency	1 kHz
Conversion time	0.625 ms default setting, configurable
Resolution	12 bits (16 bits presentation, incl. sign)
Measuring error	< ±0.3 % (relative to full scale value)
Electrical isolation	500 V (E-bus/signal voltage)
Current consumption power contacts	–
Current consumption E-bus	typ. 130 mA
Dielectric strength	max. 30 V
Bit width in the process image	inputs: 16 byte
Special features	activatable FIR/IIR filters, limit value monitoring
Weight	approx. 60 g
Operating/storage temperature	0…+55 °C/-25…+85 °C
Relative humidity	95 %, no condensation
Vibration/shock resistance	conforms to EN 60068-2-6/EN 60068-2-27
EMC immunity/emission	conforms to EN 61000-6-2/EN 61000-6-4
Protect. class/installation pos.	IP 20/variable
Pluggable wiring	for all ESxxxx terminals
Approvals	CE, UL, Ex



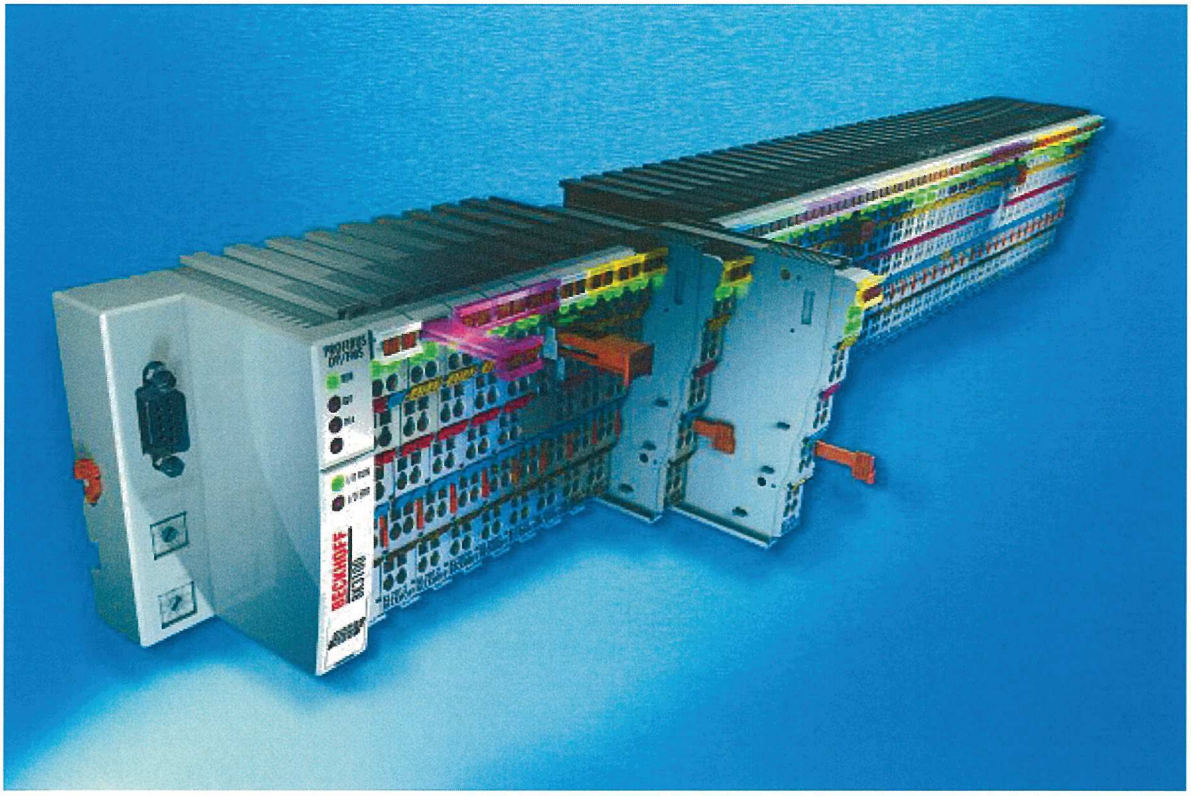
Top view

Contact assembly

EL4004 | 4-channel analog output terminal 0…10 V, 12 bits

The EL4004 analog output terminal generate signals in the range between 0 and 10 V. The voltage is supplied to the process level with a resolution of 12 bits and is electrically isolated. The output channels of the EtherCAT Terminal have a common ground potential. The EL4004 has four channels. The output stages are powered by the 24 V supply. The signal state of the EtherCAT Terminal is indicated by light emitting diodes.

Technical data	EL4004 | ES4004
Connection technology	2-wire, single-ended
Number of outputs	4
Power supply	24 V DC via power contacts
Signal voltage	0…10 V
Distributed clocks	yes
Distributed clock precision	<< 1 µs
Load	> 5 kΩ (short-circuit-proof)
Output error	< 0.1 % (relative to end value)
Resolution	12 bits
Electrical isolation	500 V (E-bus/signal voltage)
Conversion time	~ 250 µs
Current consumption power contacts	typ. 25 mA
Current consumption E-bus	typ. 140 mA
Bit width in the process image	4 x 16 bit AO output
Special features	Optional watchdog: user-specific output value with ramp; user synchronisation can be activated.
Weight	approx. 85 g
Operating/storage temperature	0…+55 °C/-25…+85 °C
Relative humidity	95 %, no condensation
Vibration/shock resistance	conforms to EN 60068-2-6/EN 60068-2-27
EMC immunity/emission	conforms to EN 61000-6-2/EN 61000-6-4
Protect. class/installation pos.	IP 20/variable
Pluggable wiring	for all ESxxxx terminals
Approvals	CE, UL, Ex



Operating instructions for

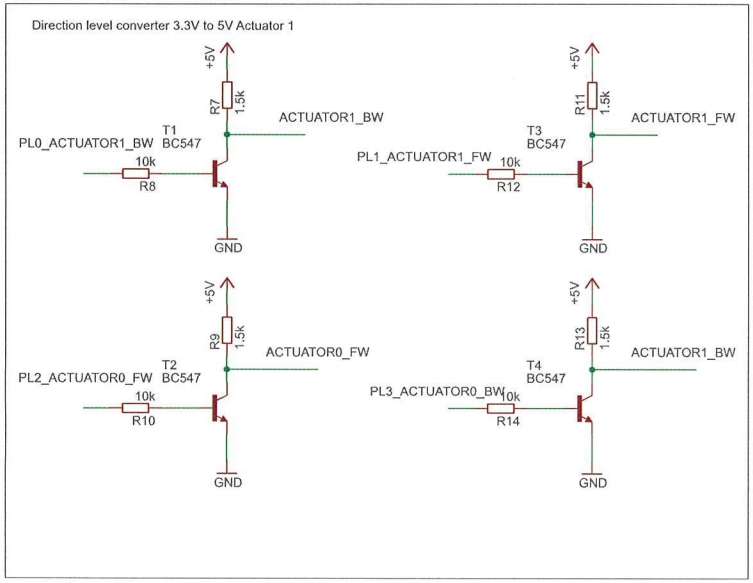
KL4001, KL4002 and KL4004

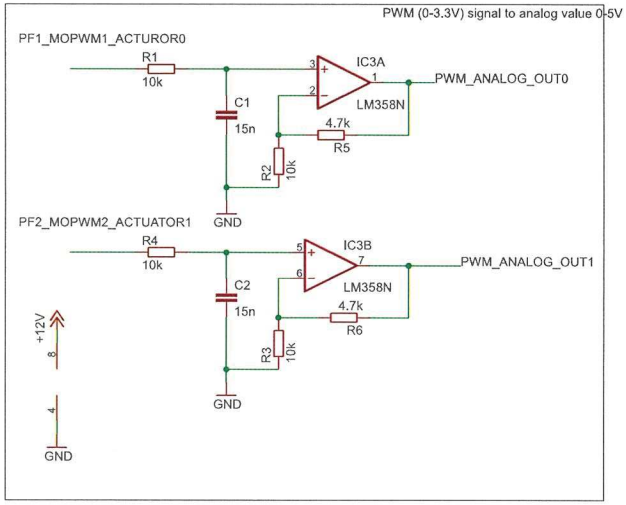
Single-, Two- and Four-Channel Analog Output Terminals
Signal range: 0 V to 10 V

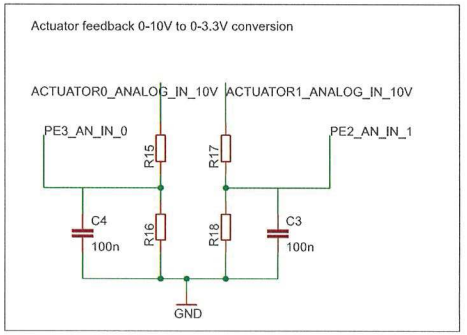
Version: 3.2
Date: 2013-04-02

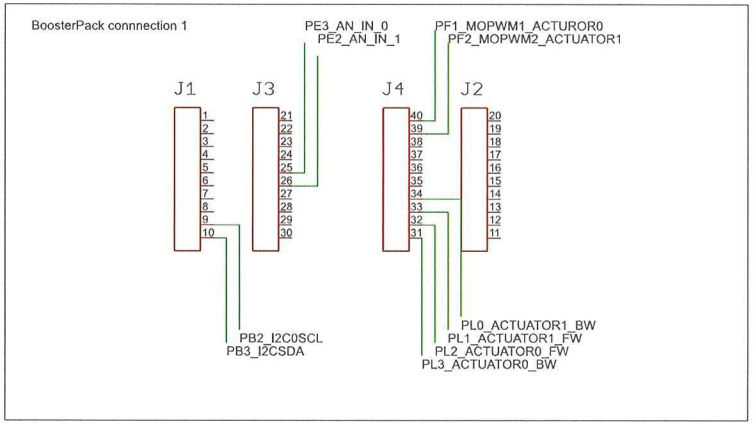
BECKHOFF

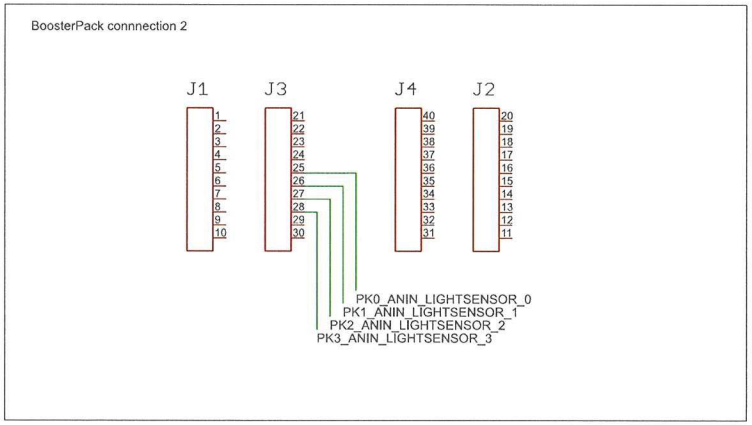
BILAG 11: Mikrocontroller styring

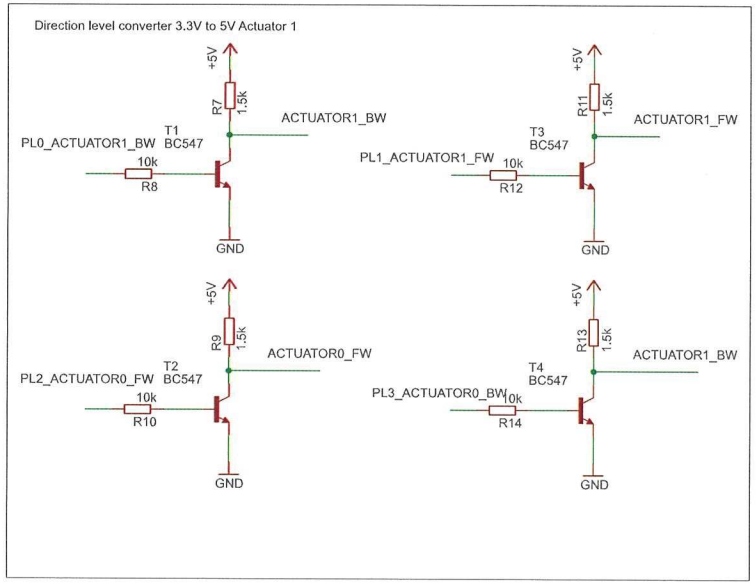

Direction level converter 3.3V to 5V Actuator 1
+5V
R7
1.5k
ACTUATOR1_BW
T1
PL0_ACTUATOR1_BW
BC547
10k
R8
GND
+5V
R11
1.5k
ACTUATOR1_FW
T3
PL1_ACTUATOR1_FW
BC547
10k
R12
GND
+5V
R9
1.5k
ACTUATOR0_FW
T2
PL2_ACTUATOR0_FW
BC547
10k
R10
GND
+5V
R13
1.5k
ACTUATOR1_BW
T4
PL3_ACTUATOR0_BW
BC547
10k
R14
GND

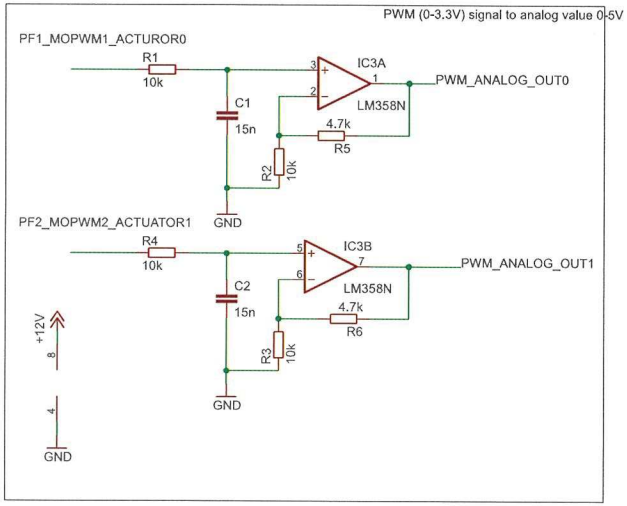

PWM (0-3.3V) signal to analog value 0-5V
PF1_MOPWM1_ACTUROR0
R1
10k
IC3A
LM358N
PWM_ANALOG_OUT0
C1
15n
4.7k
R5
R2
10k
GND
PF2_MOPWM2_ACTUATOR1
R4
10k
IC3B
LM358N
PWM_ANALOG_OUT1
C2
15n
4.7k
R6
R3
10k
GND
+12V
GND

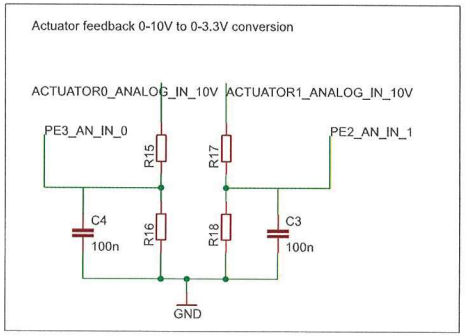

Actuator feedback 0-10V to 0-3.3V conversion
ACTUATOR0_ANALOG_IN_10V
ACTUATOR1_ANALOG_IN_10V
PE3_AN_IN_0
PE2_AN_IN_1
R15
R17
C4
100n
R16
R18
C3
100n
GND

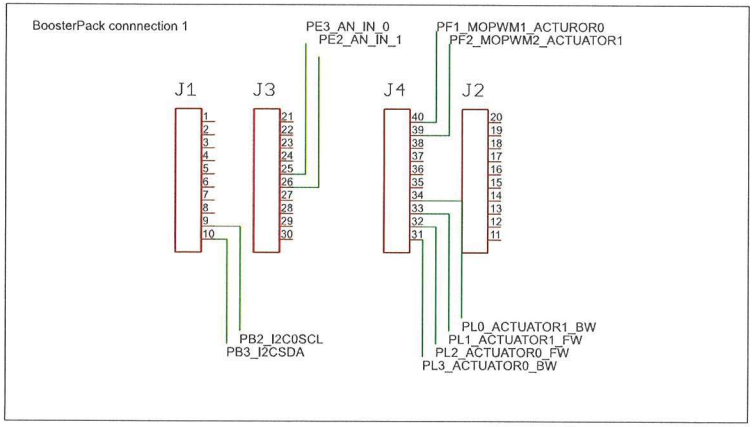

BoosterPack connnection 1
PE3_AN_IN_0
PE2_AN_IN_1
PF1_MOPWM1_ACTUROR0
PF2_MOPWM2_ACTUATOR1
J1
J3
J4
J2
PB2_I2C0SCL
PB3_I2CSDA
PL0_ACTUATOR1_BW
PL1_ACTUATOR1_FW
PL2_ACTUATOR0_FW
PL3_ACTUATOR0_BW

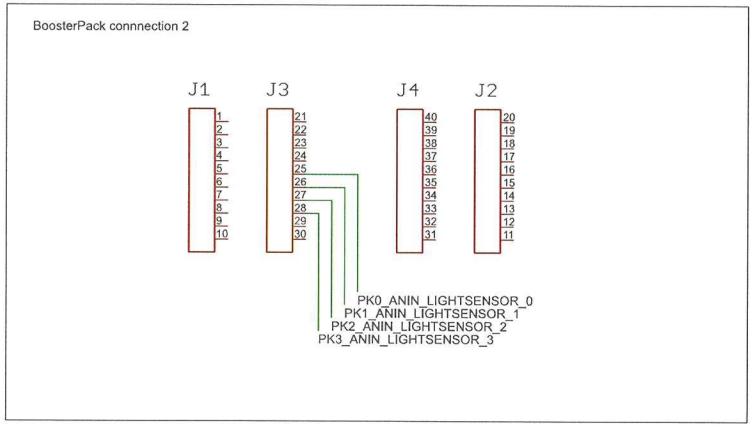

BoosterPack connnection 2
J1
J3
J4
J2
PK0_ANIN_LIGHTSENSOR_0
PK1_ANIN_LIGHTSENSOR_1
PK2_ANIN_LIGHTSENSOR_2
PK3_ANIN_LIGHTSENSOR_3

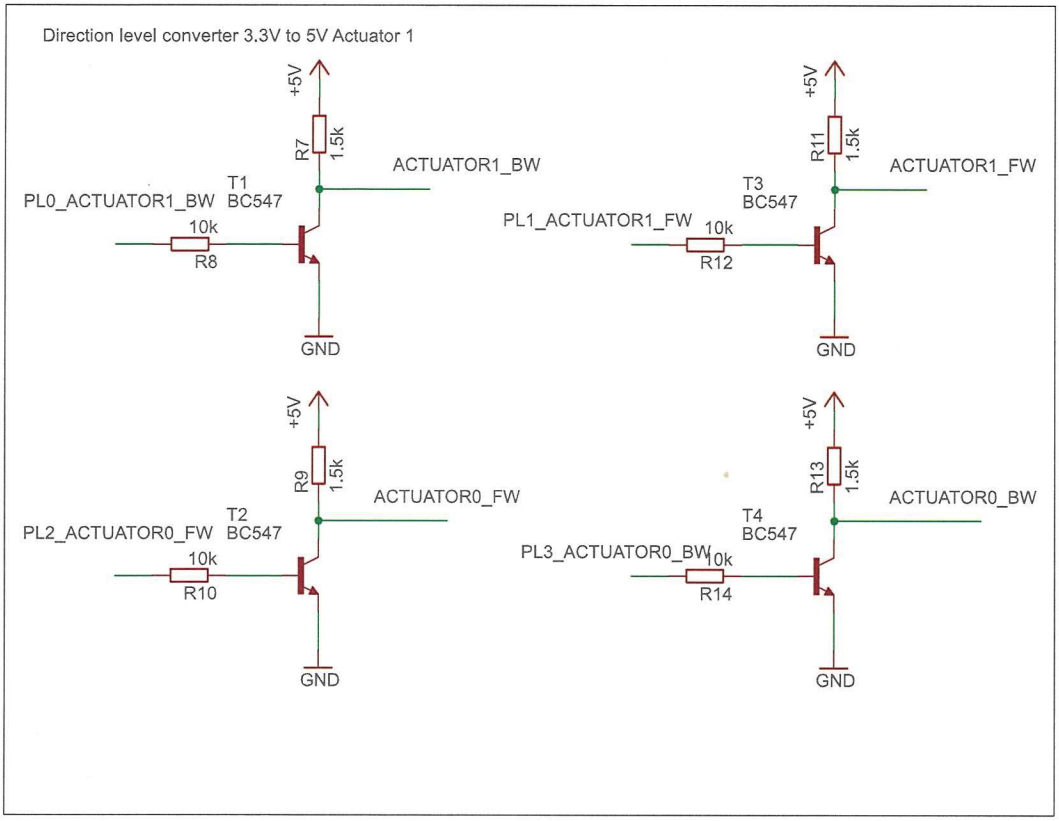

Direction level converter 3.3V to 5V Actuator 1
+5V
R7
1.5k
ACTUATOR1_BW
T1
PL0_ACTUATOR1_BW
BC547
10k
R8
GND
+5V
R11
1.5k
ACTUATOR1_FW
T3
PL1_ACTUATOR1_FW
BC547
10k
R12
GND
+5V
R9
1.5k
ACTUATOR0_FW
T2
PL2_ACTUATOR0_FW
BC547
10k
R10
GND
+5V
R13
1.5k
ACTUATOR1_BW
T4
PL3_ACTUATOR0_BW
BC547
10k
R14
GND

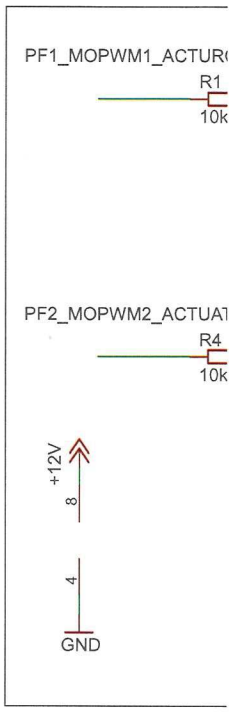

PWM (0-3.3V) signal to analog value 0-5V
PF1_MOPWM1_ACTUROR0
R1
10k
C1
15n
IC3A
LM358N
PWM_ANALOG_OUT0
4.7k
R5
R2
10k
GND
PF2_MOPWM2_ACTUATOR1
R4
10k
C2
15n
IC3B
LM358N
PWM_ANALOG_OUT1
4.7k
R6
R3
10k
GND
+12V
GND

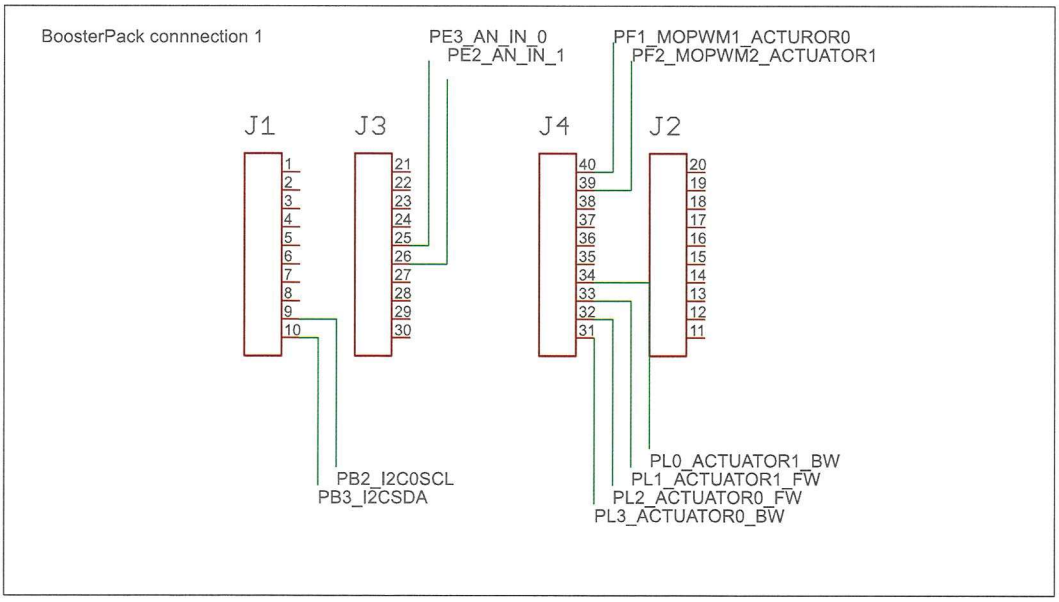

Actuator feedback 0-10V to 0-3.3V conversion
ACTUATOR0_ANALOG_IN_10V
ACTUATOR1_ANALOG_IN_10V
PE3_AN_IN_0
PE2_AN_IN_1
R15
R17
C4
100n
R16
R18
C3
100n
GND

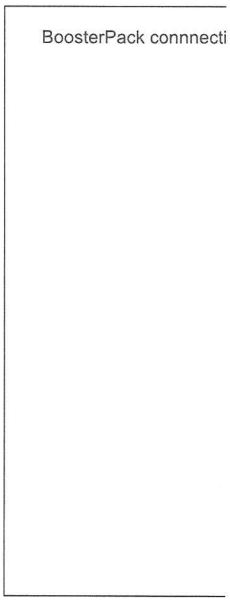

BoosterPack connnection 1
PE3_AN_IN_0
PE2_AN_IN_1
PF1_MOPWM1_ACTUROR0
PF2_MOPWM2_ACTUATOR1
J1
J3
J4
J2
PB2_I2C0SCL
PB3_I2CSDA
PL0_ACTUATOR1_BW
PL1_ACTUATOR1_FW
PL2_ACTUATOR0_FW
PL3_ACTUATOR0_BW


BoosterPack connnection 2
J1
J3
J4
J2
PK0_ANIN_LIGHTSENSOR_0
PK1_ANIN_LIGHTSENSOR_1
PK2_ANIN_LIGHTSENSOR_2
PK3_ANIN_LIGHTSENSOR_3

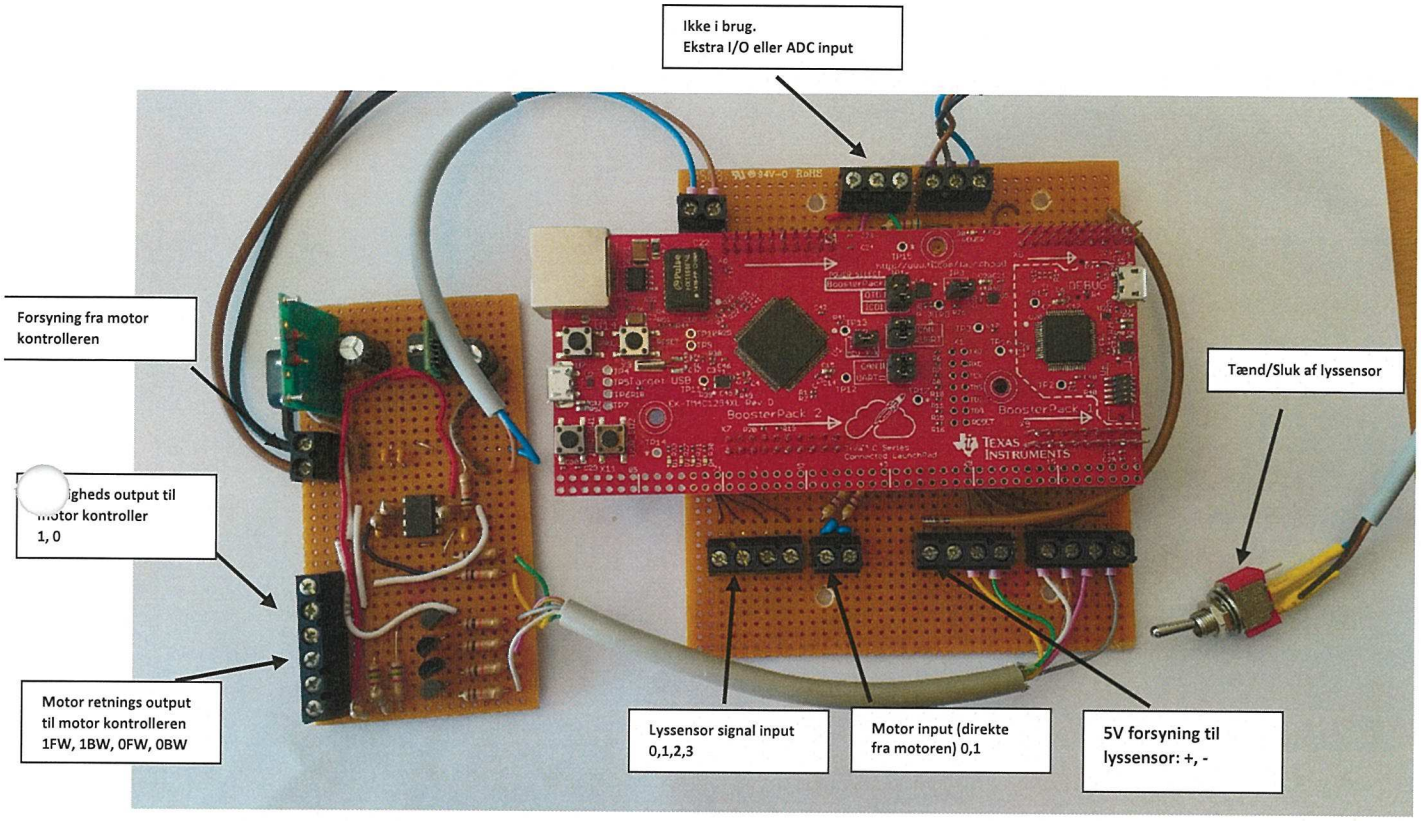

Direction level converter 3.3V to 5V Actuator 1
+5V
R7
1.5k
ACTUATOR1_BW
T1
PL0_ACTUATOR1_BW
BC547
10k
R8
GND
+5V
R11
1.5k
ACTUATOR1_FW
T3
BC547
PL1_ACTUATOR1_FW
10k
R12
GND
+5V
R9
1.5k
ACTUATOR0_FW
T2
PL2_ACTUATOR0_FW
BC547
10k
R10
GND
+5V
R13
1.5k
ACTUATOR0_BW
T4
BC547
PL3_ACTUATOR0_BW
10k
R14
GND


PF1_MOPWM1_ACTUR
R1
10k
PF2_MOPWM2_ACTUA
R4
10k
+12V
8
4
GND


BoosterPack connnection 1
PE3_AN_IN_0
PE2_AN_IN_1
PF1_MOPWM1_ACTUROR0
PF2_MOPWM2_ACTUATOR1
J1
J3
J4
J2
1
2
3
4
5
6
7
8
9
10
21
22
23
24
25
26
27
28
29
30
40
39
38
37
36
35
34
33
32
31
20
19
18
17
16
15
14
13
12
11
PB2_I2C0SCL
PB3_I2CSDA
PL0_ACTUATOR1_BW
PL1_ACTUATOR1_FW
PL2_ACTUATOR0_FW
PL3_ACTUATOR0_BW


BoosterPack connnecti

Tilslutning af Print



Tilslutningerne er markeret med pile og antallet og nummeringen af input/output.

Der er en fældes 5V forsyning til lyssensorerne, brun er den positive pol, blå er den negative og den sorte ledning er signal fra sensoren.

Til motor feedback er der placeret to inputs specielt til dette da det er en 0-10V spænding direkte fra motorerne. Tilsluttes disse i til et andet input eller output kan elektronikken risikerer at gå i stykker.

Til styring af motor benyttes motor kontrolleren, der er i alt seks kontrol input. Hvoraf de to er til at styrer retningen af motoren (om aktuatoren bliver kortere eller længere) og en til at styre hastigheden til hver af motorerne.

DC-forsyningen til systemet kommer fra den ene motor kontrollers egen DC-forsyning til motor kontrollerens egen styring.
Stelen på DC-forsyning fra begge motor kontrollere kobles sammen.

BILAG 12: Sick P30 kamera

OPERATING INSTRUCTIONS

Inspector P30

Vision Sensor

SICK
Sensor Intelligence.

QUICKSTART

Inspector P30

Vision sensor

EN

SICK
Sensor Intelligence.

Safety

- Read the operating instructions before using the Inspector.
- Connection, assembly, and settings must be performed by competent technicians.
- Do not connect external I/O signals to the Inspector while it is powered. This may damage the device.
- Make sure that any loose cable ends are properly separated or isolated before powering the Inspector. Otherwise the device may be damaged.
- Protect the Inspector from moisture and dirt during operation.
- Do not use the Inspector in areas with risk for explosion.
- To keep IP 67 classification, open and close the front window only with the supplied tool. Make sure that the seal fits properly.
- To avoid damages, only Inspector accessory lenses offered by SICK are allowed to be used.
- Minimize the risk of getting dust into the device by changing the lenses in a dust-free environment. Do not keep the device without the front window and wipe off the front window before you open it.

Product Features

- High-speed object positioning and guidance
- Robust pattern match algorithm to locate object of known shape, independent of position, rotation and scale
- Robust blob algorithm to locate multiple free-form objects
- Multi-reference object teaching
- Easy-to-use configuration in PC
- Emulator, log, statistics and record for production control
- Industrial Ethernet for monitoring over network
- Exchangeable lens
- Object position, rotation and additional information via Ethernet
- Two- and eight-directional guidance via binary outputs

Overview

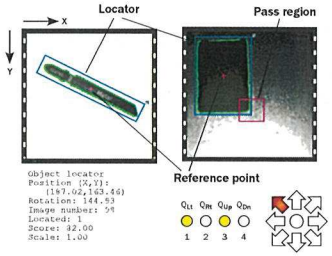
The Inspector P30 is used for high-speed positioning and guidance applications:

Positioning

Locate the object and report the position of a reference point on the object.

Guidance

Locate the object and give guidance for moving the Inspector to get the object inside a specified pass region.



Regardless of type of application, the Inspector P30 can use two different methods for locating the objects:

Object locator Locates object with a known shape. The shape is learned when configuring the Inspector.

Blob locator Locates free-form objects, based on their size and grayscale.

The position of the located object (or objects) can be retrieved via Ethernet.

Guidance is provided on 4 binary outputs, that indicate in which direction the Inspector should be moved in order to place the target inside the pass region. Two of the outputs provide the horizontal direction (Q_{Lt}, Q_{Rt}) and the other two the vertical direction (Q_{Up}, Q_{Dn}).

Mounting

1. Mount the Inspector at an appropriate distance from the objects to be inspected.

***An Inspector with ring light** may need to be mounted at a small angle to avoid too much reflexes.*

***An Inspector with Dome light** should be aligned with the objects, at a distance of approximately 50 mm to get optimal effect of the dome lighting.*

2. Connect the Ethernet connector on the Inspector to a network, or directly to an Ethernet connector on a PC.
3. If the input or output signals are used, or if an external lighting will be used, connect those devices to the Inspector's Power connector (B).

Note: Make sure that the loose ends of the I/O cable are separated before powering the Inspector.

4. Connect the Inspector to a 24 V DC power supply (B).
5. If a PC should be used for configuring the Inspector, install SOPAS Inspector by inserting the CD and following the instructions of the installation program.

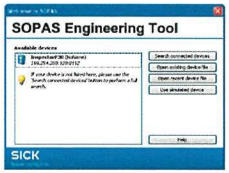
   If the installation does not start automatically, open the file `welcome.html` on the CD.

SOPAS Inspector

SOPAS Inspector is the PC application that is used for monitoring and configuring the Inspector. It can be installed on any PC, and then be used for accessing any Inspector that is connected to the same network as the PC.

To connect to an Inspector from SOPAS Inspector.

1. Make sure that the Inspector is powered, and connected to the PC or to the same network as the PC.
2. Start **SOPAS Inspector**.
3. In the Welcome dialog box, click on the Inspector in the list of available devices.



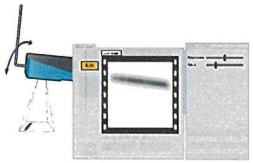
The SOPAS Inspector main window should now be displayed, showing the image from the Inspector.

- If SOPAS could not connect to the Inspector, the Connection Wizard is displayed, where you can change the IP address of your Inspector.
- If the device is not in the list, click **Search connected devices** to open the Connection Wizard.

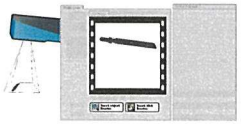
For more information about the Connection Wizard, see the online help.

Teaching with SOPAS Inspector

1. Switch the Inspector to Edit mode by clicking **Edit** in the main view.
2. Place an object in front of the Inspector, and adjust focus and exposure so that the live image is sharp and bright.

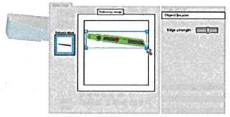


3. Click **Teach object locator** or **Teach blob locator**, depending on which method should be used for locating objects.



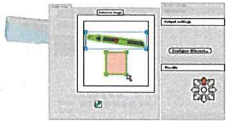
The Inspector captures a reference image of the object and displays it.

4. If necessary, adjust the size and location of the locator region, and adjust locator settings.

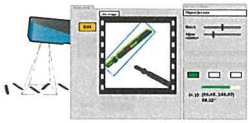


5 For positioning applications, set the format of the result output by clicking **Configure Ethernet result output** on the **Output settings** tab.

For guidance applications, place a pass region where the object's reference point should be located when on target.



6. Test the settings in the Live image and adjust them if necessary.



When the inspections are accurate enough, switch to Run to put the Inspector in operation. When asked, you should save the settings in the Inspector's flash memory to assure that they will be remembered if the power should be disconnected.

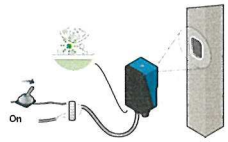
Teaching without PC

When teaching an Inspector which has not previously been configured, the result is a guidance application with the following default properties:

- An object locator covering the entire field of view.
- A circular pass region with a radius of 5 pixels, centered on the object's reference point.

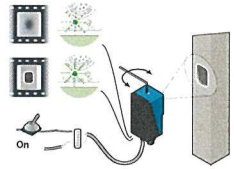
1. Place an object in front of the Inspector and connect in2 to +24 V.

   After about 3 seconds the Function LED starts to flash.



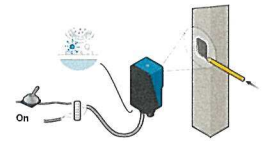
2. Adjust focus by turning the focus screw.

   The faster the Function LED flashes, the better the focus.



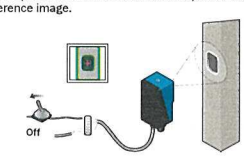
3. Find the field of view with the help of for example a pen.

   When the Inspector detects movements in the field of view, the Function LED changes color to blue.

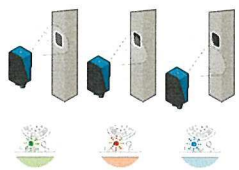


4. Disconnect in2 from the +24 V.

   The Inspector will now use the latest captured image as reference image.



5. The Inspector will then automatically switch to Run mode and start locating.



Object Locator

The object locator is used for locating objects with a known shape. A reference object can not have more than one object locator.

The Inspector locates the object by recognizing the contours of it, which is learned when configuring the Inspector. The contours that are used are highlighted in green in SOPAS Inspector.

- Adjust the amount of contours to learn with the **Edge strength** slider on the **Object locator** tab.
- Adjust the size, location, shape, and rotation of the object locator so that it covers the object properly. Use masks to mask out areas that normally varies on the object, for example if they have different labels.
- Adjust the reference point (pick point) on the object by dragging it.

Adjusting Match Settings

- Bright contours are more important than dim contours.
- Avoid highlighting contours outside the object.
- It is not necessary to get bright contours everywhere on the important contours.
- If objects are never rotated, deselect **Allow rotation** to improve both robustness and speed.
- If objects are always of the same size and at the same distance from the Inspector, deselect **Allow scaled objects** to improve both robustness and speed.

OK
Enough amount of learned contours

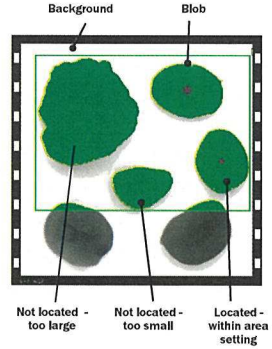
Bad
Too few contours

Bad
Too many contours, as well as contours on the background

Blob Locator

The blob locator is used for locating objects whose shape may vary from object to object. It can also be used for locating multiple objects at the same time.

- Adjust the grey scale color range of the pixels inside the blobs with the **Intensity** setting on the **Blob locator** tab.
- Adjust the size each blob must have to be considered an object
- Adjust the size, location, shape, and rotation of the blob locator region so that it covers the area where the objects are expected. Use masks to mask out areas inside the locator area.



The blob locator can locate up to 16 objects simultaneously, The result provided via Ethernet can be sorted by blob size or location.

For guidance application, the guidance is given for the first located object according to the sort order.

Australia
Phone +61 3 9497 4100
1800 334 802 – tollfree
E-Mail sales@sick.com.au

Belgium/Luxembourg
Phone +32 (0)2 466 55 66
E-Mail info@sick.be

Brasil
Phone +55 11 3215-4900
E-Mail sac@sick.com.br

Canada
Phone +1(952) 941-6780
1 800-325-7425 – tollfree
E-Mail info@sickusa.com

Ceská Republika
Phone +420 2 57 91 18 50
E-Mail sick@sick.cz

China
Phone +852-2763 6966
E-Mail ghk@sick.com.hk

Danmark
Phone +45 45 82 64 00
E-Mail sick@sick.dk

Deutschland
Phone +49 211 5301-301
E-Mail kundenservice@sick.de

España
Phone +34 93 480 31 00
E-Mail info@sick.es

France
Phone +33 1 64 62 35 00
E-Mail info@sick.fr

Great Britain
Phone +44 (0)1727 831121
E-Mail info@sick.co.uk

India
Phone +91-22-4033 8333
E-Mail info@sick-india.com

Israel
Phone +972-4-999-0590
E-Mail info@sick-sensors.com

Italia
Phone +39 02 27 43 41
E-Mail info@sick.it

Japan
Phone +81 (0)3 3358 1341
E-Mail support@sick.jp

Magyarország
Phone +36 1 371 2680
E-Mail office@sick.hu

Nederlands
Phone +31 (0)30 229 25 44
E-Mail info@sick.nl

Norge
Phone +47 67 81 50 00
E-Mail austefjord@sick.no

Österreich
Phone +43 (0)22 36 62 28 8-0
E-Mail office@sick.at

Polska
Phone +48 22 837 40 50
E-Mail info@sick.pl

România
Phone +40 356 171 120
E-Mail office@sick.ro

Russia
Phone +7 495 775 05 30
E-Mail info@sick.ru

Schweiz
Phone +41 41 619 29 39
E-Mail contact@sick.ch

Singapore
Phone +65 6744 3732
E-Mail admin@sicksgp.com.sg

South Africa
Phone +27 11 472 3733
E-Mail info@sickautomation.co.za

South Korea
Phone +82-2 786 6321/4
E-Mail info@sickkorea.net

Slovenija
Phone +386 (0)1-47 69 990
E-Mail office@sick.si

Suomi
Phone +358-9-25 15 800
E-Mail sick@sick.fi

Sverige
Phone +46 10 110 10 00
E-Mail info@sick.se

Taiwan
Phone +886 2 2375-6288
E-Mail sales@sick.com.tw

Türkiye
Phone +90 216 528 50 00
E-Mail info@sick.com.tr

United Arab Emirates
Phone +971 4 8865 878
E-Mail info@sick.ae

USA/México
Phone +1(952) 941-6780
1 800-325-7425 – tollfree
E-Mail info@sickusa.com

More representatives and agencies at **www.sick.com**

8013718/WB68 - Printed in Germany (2012-06)

If this document is printed in a different size than the original size, the width of the squares must be measured manually before calibrating the device.

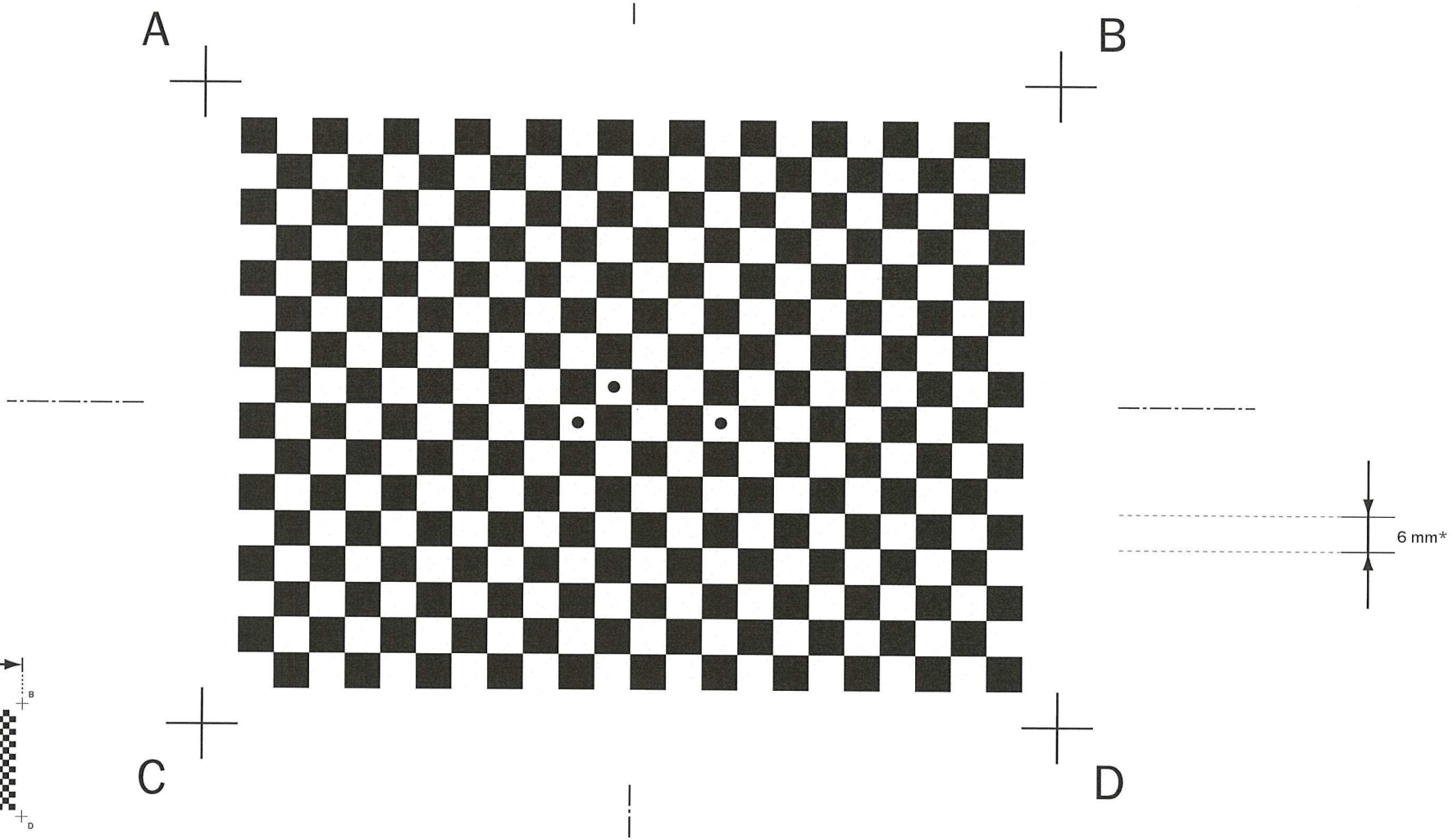
To print this document in its original size from Adobe Acrobat, make sure that the following settings are used in the Print dialog box:

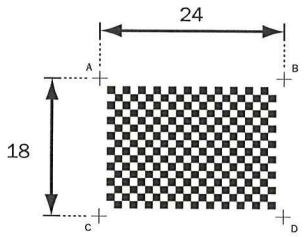
- "Page Scaling" is set to "None"
- "Choose paper source by PDF page size" is deselected

Falls dieses Dokument in einem anderen Format als das Original gedruckt ist, sind die quadratischen Flächen vor dem kalibrieren zu vermessen.

Um dieses Dokument aus Adobe Acrobat in Originalgröße zu drucken, müssen Sie folgende Einstellungen im Druckdialog wählen:

- „Seitenanpassung" auf „Keine"
- „Papierquelle anhand Seitengröße der PDF-Datei wählen" deaktiviert





* The measurement (6 mm) is only valid if this document is printed without being scaled.

Important: Mount the calibration target on a flat surface before calibrating the device.

* Die Messung (6 mm) ist nur für unskaliert gedruckte Dokumente gültig.

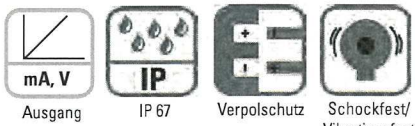
Wichtig: Legen Sie das Kalibrierziel auf eine ebene Oberfläche, bevor Sie die Kamera kalibrieren.

BILAG 13: Tiltsensor – datablad

Neigungssensor

NEU

Neigungssensor Typ IS40



Ausgang | IP 67 | Verpolschutz | Schockfest/ Vibrationsfest

Mit den Neigungssensoren der Typenreihe IS40 werden 2- dimensionale Neigungen gemessen. Je nach Ausführung sind Messbereiche von ±10°, ± 45° oder ± 60° lieferbar. Der kompakte und robuste Aufbau macht den Sensor zu einem geeigneten Winkelmessgerät in rauer Umgebung.

Innovativ:

- **Robuster Aufbau**
- **Hohe Auflösung und Genauigkeit**
- **Strom- oder Spannungsschnittstelle**
- **Hohe Schockfestigkeit**
- **Nullpunktjustierung**

RoHS

Kompakt:

- **kleine Bauform**
- **geringer Platzbedarf**

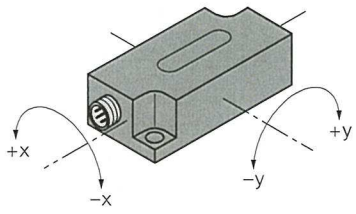
Viele Anwendungen:

- **Fahrzeugtechnik**
- **Solaranlagen**
- **Kran- und Hebetechnik**
- **Nutzkraftfahrzeuge**

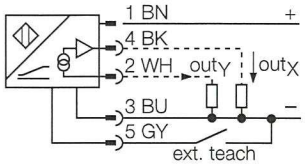
Technische Daten Neigungssensor

Spannungsversorgung:	5 VDC +/- 0,25 V oder 10 ... 30 V DC (je nach Ausführung)
Leerlaufstrom	≤ 20 mA
Verpolschutz:	Ja
Messbereich	±10°, ±45° ±60 ° (je nach Ausführung)
Auflösung:	≤ 0,05° <_ 0,1° < _ 0,15°
Wiederholgenauigkeit:	≤ 0,2 % vom Messbereich ≤ 0,1 % nach einer Aufwärmzeit von 30 min.
Absolute Genauigkeit:	0,3° bei Ausführung ±10° 0,5° bei Ausführung ±45° und ±60°
Temperaturdrift:	≤ 0,025%/K ±0,01 °/K (10°-Ausführung), ±0,03 °/K (45° und 60°-Ausführung)
Querempfindlichkeit:	3 %
Umgebungstemperatur:	-30 ... +70 °C
Ausgang:	Analogausgang
Spannungsausgang:	0,1 ... 4,9 V Kurzschlussfest gegen U_b
Stromausgang:	4 ... 20 mA
Ausgangsimpedanz:	99 ... 105 Ohm
Reaktionszeit:	0,1 ...0,5 s (Zeit, die das Ausgangssignal benötigt, um auf 90 % full scale zu gelangen, wenn der Winkel von -60° auf +60° geändert wird)
Nullpunktjustierung:	Bei Ausführung: ±10 ° => ±5 ° ±45 ° => ±15 ° ±60 ° => ±15 °
Anschluss:	M12-Steckverbinder
Vibrationsfestigkeit:	55 Hz (1 mm)
Schockfestigkeit:	30 g, 11 ms
Schutzart: IP67	
Gehäusewerkstoff:	Kunststoff PBT-GF20-V0
Gewicht: 50	g
CE-konform gemäß EN 61362-2-3 (EMV-Anforderungen für Messgrößenumformer)	

Neigungsrichtung:

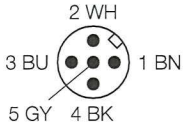


Anschlussbild:



ext. teach: wird dieser Eingang auf 0 V gelegt, wird der Ausgang des Inklinometers auf 0° zurückgesetzt.

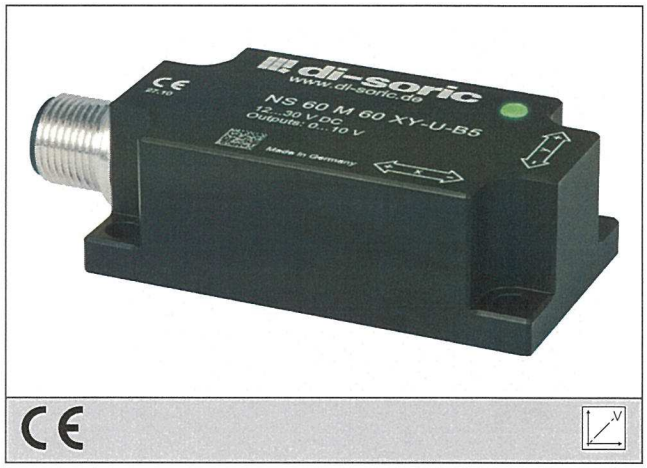
Steckerbelegung:

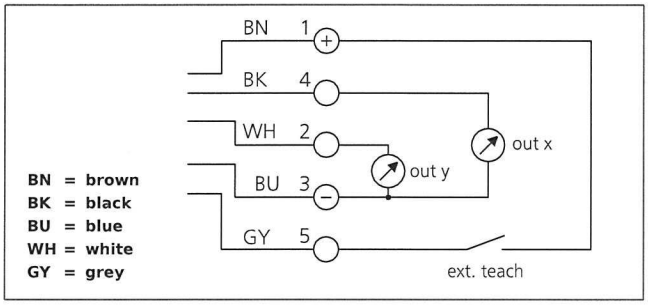


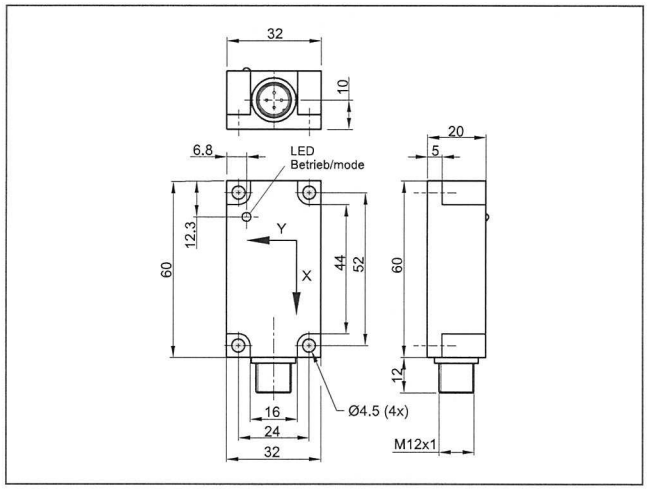
Neigungssensoren

NS 60 M 60XY-U-B5
Inclination Sensor

- Independent evaluation of two axes at the same time
- Measuring range ±60°
- Teach-in for zero position
- Two analog outputs 0 ... 10 V
- Robust metal casing







Safety instructions

The Instruments are not to be used for safety applications, in particular applications in which safety of persons depends on proper operation of the instruments.
These instruments shall exclusively be used by qualified personnel.
Repair only by di-soric.

TECHNICAL INFORMATION (typ.)	+20°C, 24V DC
Maximum rating	polarity-safe
Service voltage	12 ... 30 V DC
Internal power consumption	< 25 mA, plus 2x20 mA under load
External teach input	UB (> 6 V): Teach-in process / open: normal mode
Detection range	+/-60° per axis
Resolution	0,3°
Analog output	0 ... 10 V, min. 100kΩ, linearised
Output signal	5 V at 0°
Sensitivity	+/- 3%
Shock-/vibration load	30 g
Display	LED: green - Switching output / Teach-in status display
Response time	< 150 ms
Reproducibility	0,5°
Ambient temperature	-25 ... +70 °C
Temperature drift	0,03 °/K
Protection class	IP 67
Casing material	Aluminium anodized
Connection	Connector, M12

Inclinometers

Inclinometer | IS40 | Analogue

The inclinometer IS40 permits 2-dimensional inclinations to be measured. Versions are available for the measuring ranges ± 10°, ± 45° or ± 60°.

The compact robust construction makes this sensor the ideal device for measuring angles in harsh environments.

RoHS

Output | High IP value | Shock / vibration resistant | Reverse polarity protection

Innovative

- Rugged construction
- High resolution and accuracy
- Current or voltage interface
- High shock resistance
- Zero point adjustment

Compact / Many applications

- Small design – Minimal space requirement
- For use in vehicle technology, solar installations, commercial vehicles, cranes and hoists

Order code Inclinometer IS40

8.IS40 . 2 X X X 1
Type
a b c d e

a *Measuring direction*
2 = 2-dimensional X/Y

b *Measuring range*
1 = ± 10°
2 = ± 45°
3 = ± 60°

c *Interface*
1 = 4 ... 20 mA
3 = 0.1 ...4.9 V DC
4 = ratiometric 2% ... 98 % [1]

d *Supply voltage*
1 = 5 V DC [2]
2 = 10 ... 30 V DC

e *Type of connection*
1 = M12 connector

Connection Technology

Connectors, self-assembly (straight)	M12	**05.B-8151-0/9**
Cordset, pre-assembled with 2 m PVC cable	Coupling M12	**05.WAKS4.5-2/P00**

Additional connectors can be found in the Connection Technology section or in the Connection Technology area of our website at: www.kuebler.com/connection_technology.

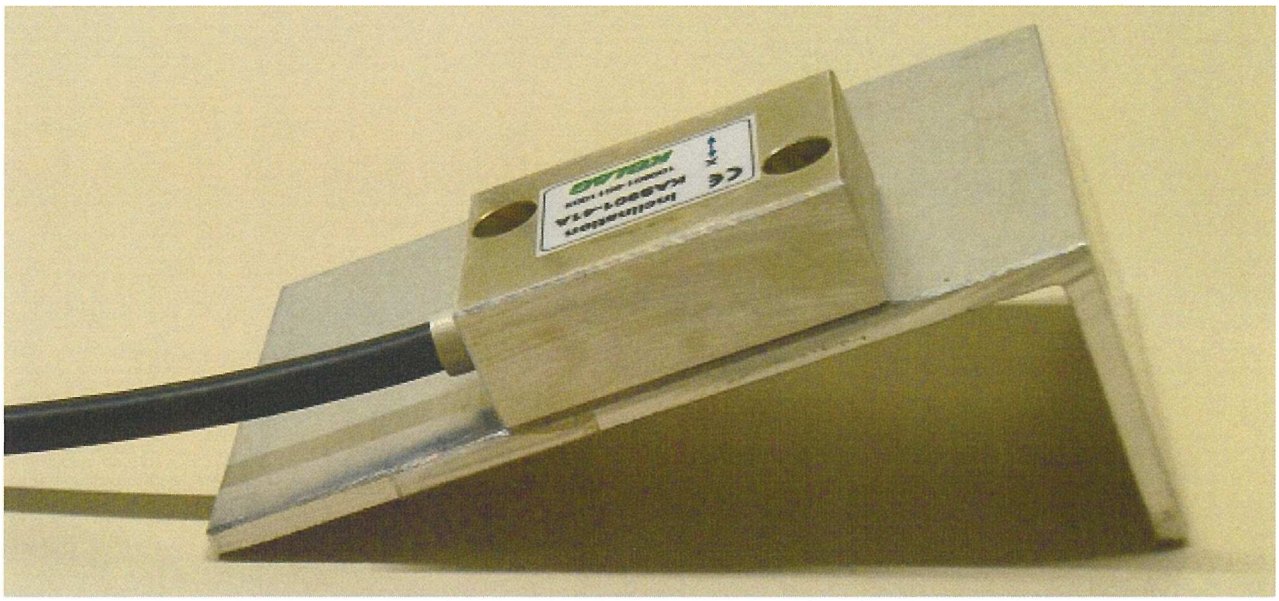
1) In relation to the supply voltage 5 V DC
2) Only in combination with interface 4

KELAG Künzli Elektronik AG
Ringstrasse 5
Ch-8630 Schwerzenbach
Schweiz / Switzerland

Telefon +41 (0)44 806 22 00
Fax +41 (0)44 806 22 08
E-Mail kelag@kelag.ch
Internet www.kelag.ch

Application manual

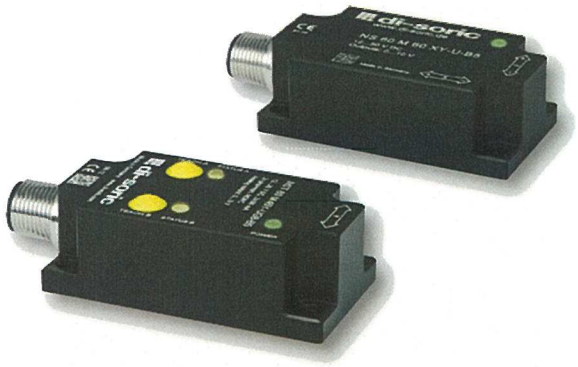
Acceleration, inclination and vibration sensor KAS901- (KAS903-) series



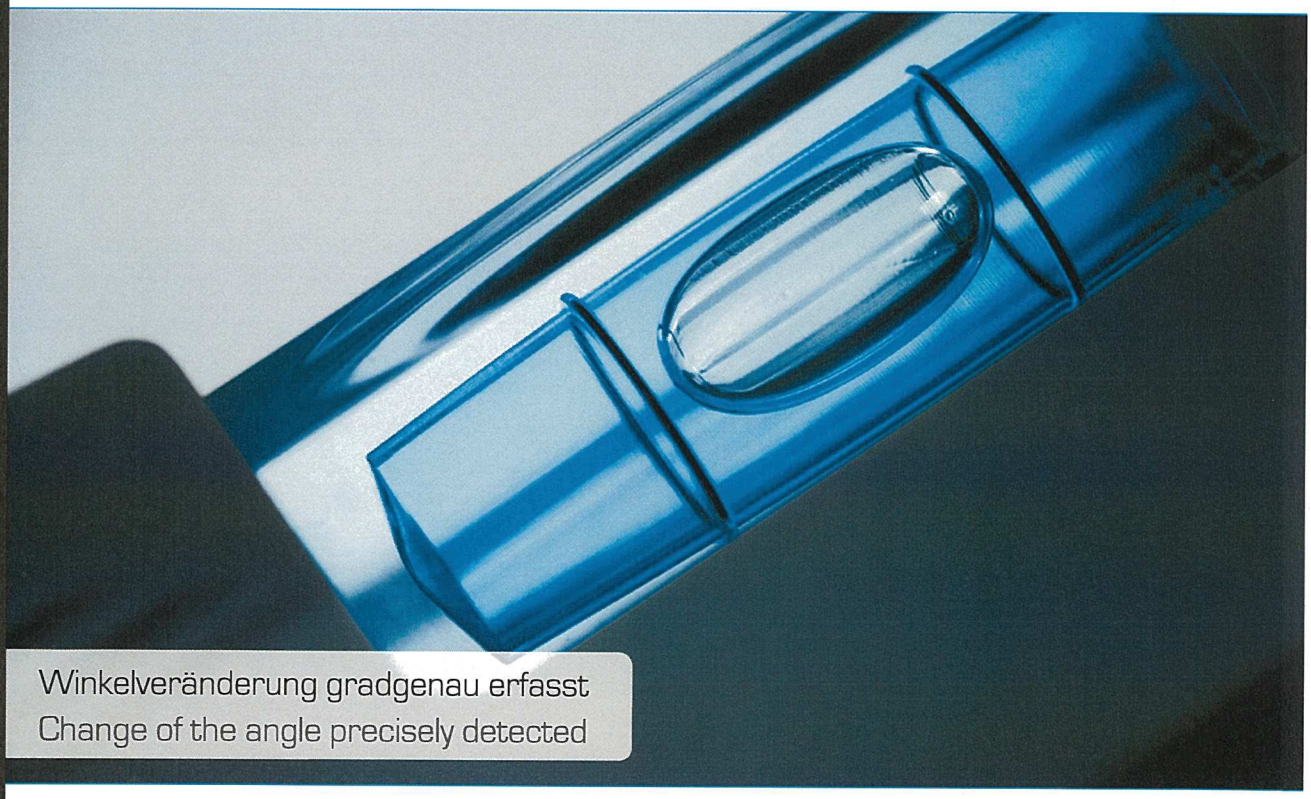
Contents

di-soric

Neigungssensoren
Inclination Sensors



Made in Germany



Winkelveränderung gradgenau erfasst
Change of the angle precisely detected

PL_072.0412de

BILAG 14: ADAM modul

ADAM-6000 Series

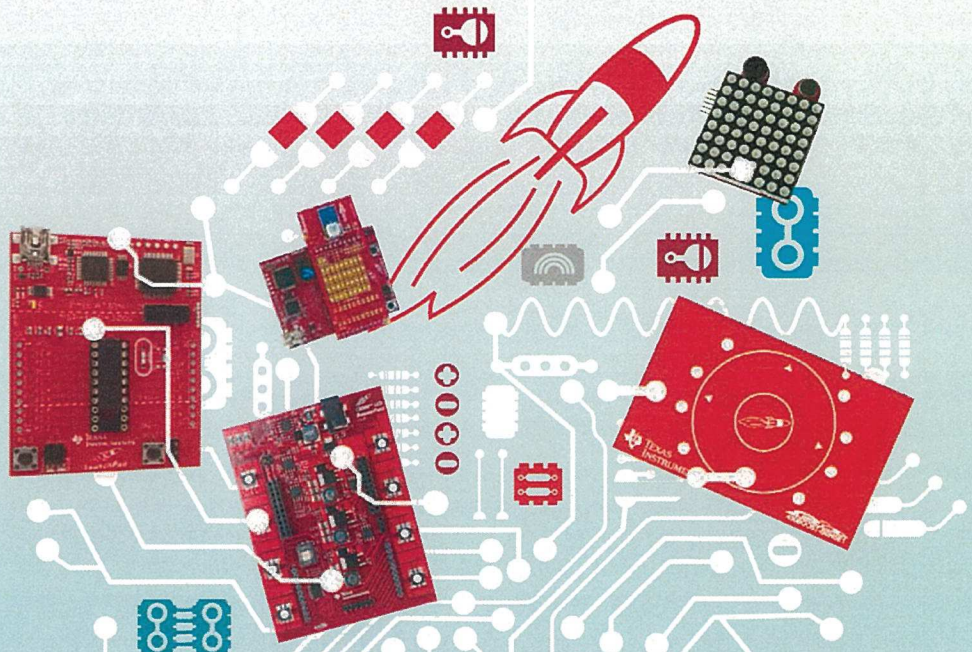
Ethernet-based
Data Acquisition and
Control Modules

User Manual

BILAG 15: Micro Tiva

Lift-off with the LaunchPad Ecosystem

Texas Instruments



ti.com/launchpad

2014

Tiva™ C Series TM4C1294 Connected LaunchPad Evaluation Kit

EK-TM4C1294XL

User's Guide

Literature Number: SPMU365A
March 2014–Revised March 2014

Meet the Tiva™ C Series TM4C1294

Connected LaunchPad Evaluation Kit

Part Number: EK-TM4C1294XL

Texas Instruments

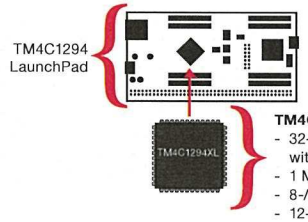
A closer look at your new LaunchPad

Featured microcontroller: Tiva C Series TM4C1294

This LaunchPad is ideal for...

- Industrial applications, including remote monitoring, networked automation, embedded gateways, test & measurement... and more
- Beginners & experienced developers with multiple points of entry into software development (Energia for beginners & industrial-grade tools like CCS, Keil, and IAR for more advanced designers)

What comes in the box?



TM4C1294NCPDTI Microcontroller
- 32-bit ARM® Cortex™-M4 120-MHz CPU with floating point
- 1 MB Flash / 256 kB RAM / 6 kB EEPROM
- 8-/16-/32-bit EPI
- 12-bit SAR ADC (2MSPS), Comparators, Timers and DMA
- Advanced connectivity integration:
  - 2 CAN Modules
  - QSSI/UART/I2C
  - Integrated Full- & Low-speed USB 2.0
  - 10/100 Ethernet MAC + PHY

This Quick Start Guide

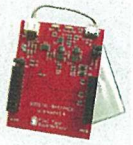
Micro-USB Cable

Ethernet Cable

Software can be downloaded online @ www.ti.com/tivaware

BoosterPack Ecosystem

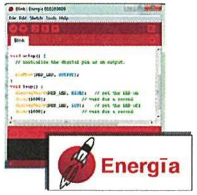
Sensor Hub BoosterPack
- InvenSense MPU-9150 9-axis MEMS motion sensor
  - 3-axis gyroscope
  - 3-axis accelerometer
  - 3-axis compass
- Bosch Sensortec BMP180 pressure sensor
- Sensirion SHT32 humidty & ambient temperature sensor
- Intersil ISL29023 light & IR sensor
- TI TMP006 contactless temp sensor



Fuel Tank BoosterPack
- Untether your LaunchPad projects!
- Rechargeable 4.44Wh battery
- I²C fuel gauge
- LED charge-level indicator
- Provides 5V & 3.3V sources

>> See them all @ ti.com/boosterpacks

Software Tools



Energia
A simple open-source & community-driven code editor based on the Wiring framework.

Robust collection of easy-to-use function calls. APIs, and examples to get you started quickly.

>> www.energia.nu

Professional Software tools
LaunchPad is also supported by professional IDEs that provide industrial-grade features and full debug capability. Set breakpoints, watch variables & more with Connected LaunchPad.

Code Composer Studio™ IDE

>> www.ti.com/ccs

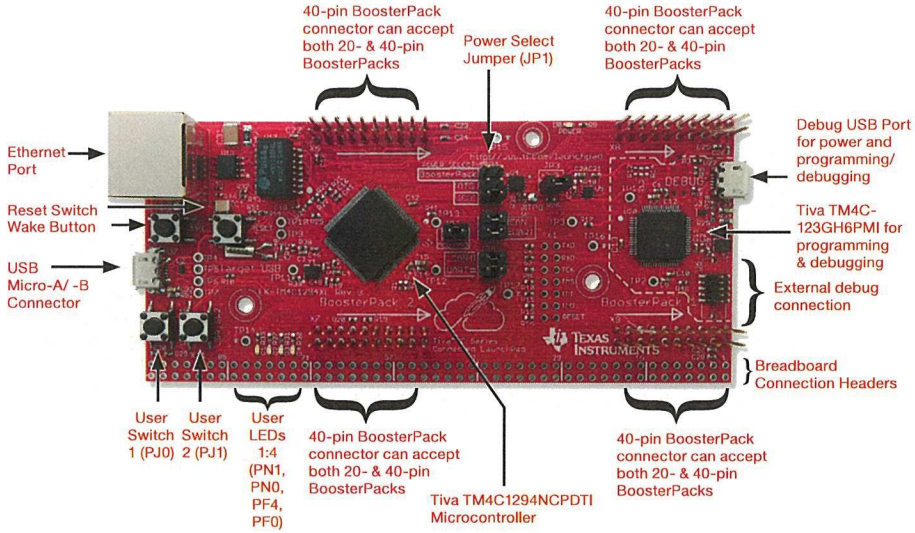
Third party IDE options

KEIL™ Tools by ARM

mentor embedded

IAR SYSTEMS

EK-TM4C1294XL Overview



SPMZ858

Tiva™ C Series TM4C1294 Connected LaunchPad Evaluation Kit

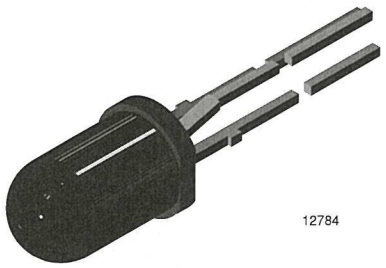
EK-TM4C1294XL

User's Guide

Literature Number: SPMU365A
March 2014–Revised March 2014

BILAG 16: Lyssensor – datablad

Silicon NPN Phototransistor



FEATURES

- Package type: leaded
- Package form: T-1¾
- Dimensions (in mm): Ø 5
- High radiant sensitivity
- Daylight blocking filter matched with 940 nm emitters
- Fast response times
- Angle of half sensitivity: $\varphi = \pm 15°$
- Base terminal connected
- Material categorization: For definitions of compliance please see www.vishay.com/doc?99912

Pb-free

e4

RoHS COMPLIANT

DESCRIPTION

BPV11F is a silicon NPN phototransistor with high radiant sensitivity in black, T-1¾ plastic package with base terminal and daylight blocking filter. Filter bandwidth is matched with 900 nm to 950 nm IR emitters.

APPLICATIONS

- Detector for industrial electronic circuitry, measurement and control

PRODUCT SUMMARY			
COMPONENT	I_{ca} (mA)	φ (deg)	$\lambda_{0.5}$ (nm)
BPV11F	9	± 15	900 to 980

Note

- Test condition see table "Basic Characteristics"

ORDERING INFORMATION			
ORDERING CODE	PACKAGING	REMARKS	PACKAGE FORM
BPV11F	Bulk	MOQ: 4000 pcs, 4000 pcs/bulk	T-1¾

Note

- MOQ: minimum order quantity

ABSOLUTE MAXIMUM RATINGS (T_{amb} = 25 °C, unless otherwise specified)				
PARAMETER	TEST CONDITION	SYMBOL	VALUE	UNIT
Collector base voltage		V_{CBO}	80	V
Collector emitter voltage		V_{CEO}	70	V
Emitter base voltage		V_{EBO}	5	V
Collector current		I_C	50	mA
Collector peak current	$t_p/T = 0.5$, $t_p \leq 10$ ms	I_{CM}	100	mA
Power dissipation	$T_{amb} \leq 47$ °C	P_V	150	mW
Junction temperature		T_j	100	°C
Operating temperature range		T_{amb}	- 40 to + 100	°C
Storage temperature range		T_{stg}	- 40 to + 100	°C
Soldering temperature	t ≤ 5 s, 2 mm from body	T_{sd}	260	°C
Thermal resistance junction/ambient	Connected with Cu wire, 0.14 mm²	R_{thJA}	350	K/W

TSL260, TSL261, TSL262
IR LIGHT-TO-VOLTAGE OPTICAL SENSORS

SOES008A – DECEMBER 1992 – REVISED FEBRUARY 1993

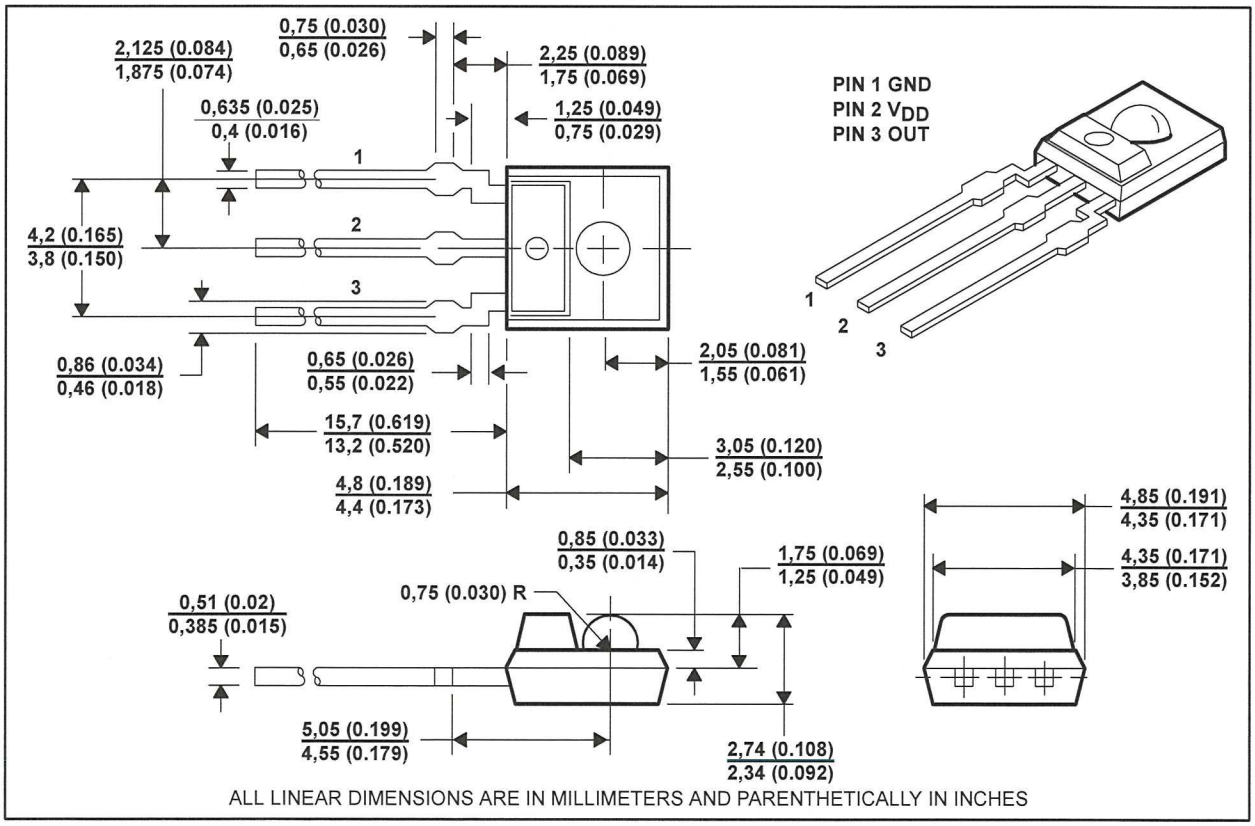
- **Integral Visible Light Cutoff Filter**
- **Monolithic Silicon IC Containing Photodiode, Operational Amplifier, and Feedback Components**
- **Converts Light Intensity to Output Voltage**
- **High Irradiance Responsivity Typically 42 mV/(μW/cm²) at λ_p = 940 nm (TSL260)**
- **Low Dark (Offset) Voltage . . . 10 mV Max at 25°C, V_{DD} = 5 V**
- **Single-Supply Operation**
- **Wide Supply Voltage Range . . . 3 V to 9 V**
- **Low Supply Current . . . 800 μA Typical at V_{DD} = 5 V**
- **Advanced LinCMOS™ Technology**

description

The TSL260, TSL261, and TSL262 are light-to-voltage optical sensors each combining a photodiode and a transimpedance amplifier (feedback resistor = 16 MΩ, 8 MΩ, and 2 MΩ, respectively) on a single monolithic integrated circuit. The output voltage is directly proportional to the infrared light intensity (irradiance) on the photodiode. The TSL260, TSL261, and TSL262 utilize Texas Instruments silicon-gate LinCMOS™ technology, which provides good amplifier offset-voltage stability and low power consumption.

mechanical data

The photodiode/amplifier chip is packaged in a black, infrared-transmissive plastic package. The integrated photodiode active area is typically 1,0 mm² (0.0016 in²), 0.5 mm² (0.00078 in²), and 0.26 mm² (0.0004 in²) for the TSL260, TSL261, and TSL262, respectively.



ALL LINEAR DIMENSIONS ARE IN MILLIMETERS AND PARENTHETICALLY IN INCHES

LinCMOS is a trademark of Texas Instruments Incorporated.

TEXAS INSTRUMENTS
POST OFFICE BOX 655303 • DALLAS, TEXAS 75265
POST OFFICE BOX 1443 • HOUSTON, TEXAS 77251-1443

BILAG 17: PLC styring – Programkode

```
(* @NESTEDCOMMENTS := 'Yes' *)
(* @PATH := '\/Actuator' *)
(* @OBJECTFLAGS := '0, 8' *)
(* @SYMFILEFLAGS := '2048' *)
PROGRAM Act_Out
VAR
        Act1: CTUD;
        puls: BOOL;
        Count: BOOL;
        ACT2: CTUD;
        ton_ContOffDelay: TON;
        ton_ContOnDelay: TON;
        ActCtrl1: AktuatorCtrl;
        ActCtrl2: AktuatorCtrl;
        fbNulstil: Nulstil;
        bNul_Fwd_1:BOOL;
        Speed1 AT %Q*: INT;
        Speed2 AT %Q* : INT;
        Status AT %I*:INT;
END_VAR
(* @END_DECLARATION := '0' *)
_LD_BODY
_NETWORKS : 16
_NETWORK

_COMMENT
''
_END_COMMENT
_LD_ASSIGN
_LD_AND
_LD_OPERATOR : 2
_LD_CONTACT
Nulstil
_EXPRESSION
_NEGATIV
_FUNCTIONBLOCK
ActCtrl1
_BOX_EXPR : 3
_EMPTY
_OPERAND
_EXPRESSION
_POSITIV
Actuators[0]
_OPERAND
_EXPRESSION
_POSITIV
1
_EXPRESSION
_POSITIV
AktuatorCtrl
_OUTPUTS : 2
_OUTPUT
_POSITIV
_NO_SET
Actuators[0].BWD
_OUTPUT
_POSITIV
_NO_SET
Actuators[0].InLimit
_EXPRESSION
_POSITIV
_EXPRESSION
_POSITIV


ENABLELIST : 0
```

```
ENABLELIST_END
_OUTPUTS : 1
_OUTPUT
_POSITIV
_NO_SET
Actuators[0].FWD
_NETWORK

_COMMENT
''
_END_COMMENT
_LD_ASSIGN
_LD_AND
_LD_OPERATOR : 2
_LD_CONTACT
Nulstil
_EXPRESSION
_NEGATIV
_FUNCTIONBLOCK
ActCtrl2
_BOX_EXPR : 3
_EMPTY
_OPERAND
_EXPRESSION
_POSITIV
Actuators[1]
_OPERAND
_EXPRESSION
_POSITIV
1
_EXPRESSION
_POSITIV
AktuatorCtrl
_OUTPUTS : 2
_OUTPUT
_POSITIV
_NO_SET
Actuators[1].BWD
_OUTPUT
_POSITIV
_NO_SET
Actuators[1].InLimit
_EXPRESSION
_POSITIV
_EXPRESSION
_POSITIV


ENABLELIST : 0
ENABLELIST_END
_OUTPUTS : 1
_OUTPUT
_POSITIV
_NO_SET
Actuators[1].FWD
_NETWORK

_COMMENT
''
_END_COMMENT
_LD_ASSIGN
_FUNCTIONBLOCK
fbNulstil
_BOX_EXPR : 0
_EXPRESSION
_POSITIV
Nulstil
_OUTPUTS : 3
```

```
_OUTPUT
_POSITIV
_NO_SET
_EMPTY
_OUTPUT
_POSITIV
_NO_SET
_EMPTY
_OUTPUT
_POSITIV
_NO_SET
_EMPTY
_EXPRESSION
_POSITIV


ENABLELIST : 0
ENABLELIST_END
_OUTPUTS : 1
_OUTPUT
_POSITIV
_NO_SET
bNul_Fwd_1
_NETWORK

_COMMENT
''
_END_COMMENT
_LD_ASSIGN
_LD_AND
_LD_OPERATOR : 2
_LD_OR
_LD_OPERATOR : 2
_LD_AND
_LD_OPERATOR : 2
_LD_CONTACT
Actuators[0].FWD
_EXPRESSION
_POSITIV
_LD_CONTACT
Actuators[0].InLimit
_EXPRESSION
_NEGATIV
_EXPRESSION
_POSITIV
_LD_CONTACT
fbNulstil.Fwd_1
_EXPRESSION
_POSITIV
_EXPRESSION
_POSITIV
_LD_CONTACT
Started
_EXPRESSION
_POSITIV
_EXPRESSION
_POSITIV
_EXPRESSION
_POSITIV


ENABLELIST : 0
ENABLELIST_END
_OUTPUTS : 1
_OUTPUT
_POSITIV
_NO_SET
bFwd_1
```

```
_NETWORK

_COMMENT
''
_END_COMMENT
_LD_ASSIGN
_LD_AND
_LD_OPERATOR : 2
_LD_OR
_LD_OPERATOR : 2
_LD_AND
_LD_OPERATOR : 2
_LD_CONTACT
Actuators[1].FWD
_EXPRESSION
_POSITIV
_LD_CONTACT
Actuators[1].InLimit
_EXPRESSION
_NEGATIV
_EXPRESSION
_POSITIV
_LD_CONTACT
fbNulstil.Fwd_2
_EXPRESSION
_POSITIV
_EXPRESSION
_POSITIV
_LD_CONTACT
Started
_EXPRESSION
_POSITIV
_EXPRESSION
_POSITIV
_EXPRESSION
_POSITIV


ENABLELIST : 0
ENABLELIST_END
_OUTPUTS : 1
_OUTPUT
_POSITIV
_NO_SET
bFwd_2
_NETWORK

_COMMENT
''
_END_COMMENT
_LD_ASSIGN
_LD_AND
_LD_OPERATOR : 2
_LD_OR
_LD_OPERATOR : 2
_LD_AND
_LD_OPERATOR : 2
_LD_CONTACT
Actuators[0].BWD
_EXPRESSION
_POSITIV
_LD_CONTACT
Actuators[0].InLimit
_EXPRESSION
_NEGATIV
_EXPRESSION
_POSITIV
_LD_CONTACT
```

```
fbNulstil.Bwd_1
_EXPRESSION
_POSITIV
_EXPRESSION
_POSITIV
_LD_CONTACT
Started
_EXPRESSION
_POSITIV
_EXPRESSION
_POSITIV
_EXPRESSION
_POSITIV


ENABLELIST : 0
ENABLELIST_END
_OUTPUTS : 1
_OUTPUT
_POSITIV
_NO_SET
bBwd_1
_NETWORK

_COMMENT
''
_END_COMMENT
_LD_ASSIGN
_LD_AND
_LD_OPERATOR : 2
_LD_OR
_LD_OPERATOR : 2
_LD_AND
_LD_OPERATOR : 2
_LD_CONTACT
Actuators[1].BWD
_EXPRESSION
_POSITIV
_LD_CONTACT
Actuators[1].InLimit
_EXPRESSION
_NEGATIV
_EXPRESSION
_POSITIV
_LD_CONTACT
fbNulstil.Bwd_2
_EXPRESSION
_POSITIV
_EXPRESSION
_POSITIV
_LD_CONTACT
Started
_EXPRESSION
_POSITIV
_EXPRESSION
_POSITIV
_EXPRESSION
_POSITIV


ENABLELIST : 0
ENABLELIST_END
_OUTPUTS : 1
_OUTPUT
_POSITIV
_NO_SET
bBwd_2
_NETWORK
```

```
_COMMENT
''
_END_COMMENT
_LD_ASSIGN
_LD_CONTACT
Actuators[0].Reset
_EXPRESSION
_POSITIV
_EXPRESSION
_POSITIV


ENABLELIST : 1
_ASSIGN
_OPERATOR
_BOX_EXPR : 1
_ENABLED
_OPERAND
_EXPRESSION
_POSITIV
500
_EXPRESSION
_POSITIV
MOVE
_EXPRESSION
_POSITIV
_OUTPUTS : 1
_OUTPUT
_POSITIV
_NO_SET
Actuators[0].CmdPos
ENABLELIST_END
_OUTPUTS : 0
_NETWORK

_COMMENT
''
_END_COMMENT
_LD_ASSIGN
_LD_CONTACT
Actuators[1].Reset
_EXPRESSION
_POSITIV
_EXPRESSION
_POSITIV


ENABLELIST : 1
_ASSIGN
_OPERATOR
_BOX_EXPR : 1
_ENABLED
_OPERAND
_EXPRESSION
_POSITIV
500
_EXPRESSION
_POSITIV
MOVE
_EXPRESSION
_POSITIV
_OUTPUTS : 1
_OUTPUT
_POSITIV
_NO_SET
Actuators[1].CmdPos
ENABLELIST_END
```

```
_OUTPUTS : 0
_NETWORK

_COMMENT
''
_END_COMMENT
_LD_ASSIGN
_EMPTY
_EXPRESSION
_POSITIV


ENABLELIST : 0
ENABLELIST_END
_OUTPUTS : 2
_OUTPUT
_NEGATIV
_SET
Actuators[0].Reset
_OUTPUT
_NEGATIV
_SET
Actuators[1].Reset
_NETWORK

_COMMENT
''
_END_COMMENT
_LD_ASSIGN
_LD_CONTACT
Nulstil
_EXPRESSION
_POSITIV
_EXPRESSION
_POSITIV


ENABLELIST : 1
_ASSIGN
_OPERATOR
_BOX_EXPR : 1
_ENABLED
_OPERAND
_EXPRESSION
_POSITIV
50
_EXPRESSION
_POSITIV
MOVE
_EXPRESSION
_POSITIV
_OUTPUTS : 1
_OUTPUT
_POSITIV
_NO_SET
Actuators[0].Speed
ENABLELIST_END
_OUTPUTS : 0
_NETWORK

_COMMENT
''
_END_COMMENT
_LD_ASSIGN
_LD_CONTACT
Nulstil
_EXPRESSION
_POSITIV
```

```
_EXPRESSION
_POSITIV


ENABLELIST : 1
_ASSIGN
_OPERATOR
_BOX_EXPR : 1
_ENABLED
_OPERAND
_EXPRESSION
_POSITIV
50
_EXPRESSION
_POSITIV
MOVE
_EXPRESSION
_POSITIV
_OUTPUTS : 1
_OUTPUT
_POSITIV
_NO_SET
Actuators[1].Speed
ENABLELIST_END
_OUTPUTS : 0
_NETWORK

_COMMENT
''
_END_COMMENT
_LD_ASSIGN
_EMPTY
_EXPRESSION
_POSITIV


ENABLELIST : 1
_ASSIGN
_OPERATOR
_BOX_EXPR : 2
_ENABLED
_OPERAND
_EXPRESSION
_POSITIV
Actuators[1].Speed
_OPERAND
_EXPRESSION
_POSITIV
16#00FF
_EXPRESSION
_POSITIV
MUL
_EXPRESSION
_POSITIV
_OUTPUTS : 1
_OUTPUT
_POSITIV
_NO_SET
Speed2
ENABLELIST_END
_OUTPUTS : 0
_NETWORK

_COMMENT
''
_END_COMMENT
_LD_ASSIGN
_EMPTY
```

```
_EXPRESSION
_POSITIV


ENABLELIST : 1
_ASSIGN
_OPERATOR
_BOX_EXPR : 2
_ENABLED
_OPERAND
_EXPRESSION
_POSITIV
Actuators[0].Speed
_OPERAND
_EXPRESSION
_POSITIV
16#00FF
_EXPRESSION
_POSITIV
MUL
_EXPRESSION
_POSITIV
_OUTPUTS : 1
_OUTPUT
_POSITIV
_NO_SET
Speed1
ENABLELIST_END
_OUTPUTS : 0
_NETWORK

_COMMENT
''
_END_COMMENT
_LD_ASSIGN
_LD_CONTACT
Nulstil
_EXPRESSION
_POSITIV
_EXPRESSION
_POSITIV


ENABLELIST : 1
_ASSIGN
_OPERATOR
_BOX_EXPR : 1
_ENABLED
_OPERAND
_EXPRESSION
_POSITIV
50
_EXPRESSION
_POSITIV
MOVE
_EXPRESSION
_POSITIV
_OUTPUTS : 1
_OUTPUT
_POSITIV
_NO_SET
Actuators[0].Speed
ENABLELIST_END
_OUTPUTS : 0
_NETWORK

_COMMENT
''
```

```
_END_COMMENT
_LD_ASSIGN
_LD_CONTACT
Nulstil
_EXPRESSION
_POSITIV
_EXPRESSION
_POSITIV


ENABLELIST : 1
_ASSIGN
_OPERATOR
_BOX_EXPR : 1
_ENABLED
_OPERAND
_EXPRESSION
_POSITIV
50
_EXPRESSION
_POSITIV
MOVE
_EXPRESSION
_POSITIV
_OUTPUTS : 1
_OUTPUT
_POSITIV
_NO_SET
Actuators[1].Speed
ENABLELIST_END
_OUTPUTS : 0

END_PROGRAM


(* @NESTEDCOMMENTS := 'Yes' *)
(* @PATH := '\/Actuator' *)
(* @OBJECTFLAGS := '0, 8' *)
(* @SYMFILEFLAGS := '2048' *)
PROGRAM ActuatorMain
VAR
        i: INT;
        RunOnce: BOOL := TRUE;
        _X: REAL;
        _Y: REAL;
        ForwardKin_FB: ForwardKin;
        InverseKin_FB: InverseKin;
        _V1: REAL;
        _V2: REAL;
        V1_beregnet: REAL;
        V2_beregnet: REAL;
        Pos2: REAL;
        Pos1: REAL;
        dX: REAL;
        dY: REAL;
        Test_Tilt: BOOL := TRUE;
        dXYtool2XYworld_FB: dXYtool2XYworld;
        Test1: REAL := 30;
        Test2: REAL := 45;
        iN: INT;
        iS: INT;
        iE: INT;
        iW: INT;
        _StepSize: INT := 50;
        _s1: REAL;
        _s2: REAL;
        _S_Max : REAL;
        _d1: REAL;
```

```
        _d2: REAL;
        _i1: REAL;
        _dT : REAL := 0.001;
        _Kp : REAL :=0.02;
        _Ki : REAL := 0.02;
        TiltYValue: REAL;
        _deadband: REAL := 4.0;
        _gain: REAL := 0.3;
        AngOffset: REAL:= 0.0;
        TiltXValue: REAL;
        inStart AT %I* : BOOL;
        inStop AT %I* : BOOL;
        inSmokeActive AT %I*: BOOL;
        inVisionActive AT %I* : BOOL;
        outStarted AT %Q* : BOOL;
        timeNulstil : TON;
        MinX : REAL := -5000;
        MaxX : REAL := 5000;
        MinY : REAL := 0000;
        MaxY : REAL := 5000;
        useLys: BOOL := TRUE;
        TiltXZero: REAL := 0.0;
        TiltYZero: REAL := 0.0;
        TiltDeadband: REAL;
END_VAR

VAR_INPUT
        _i2: REAL;
END_VAR
(* @END_DECLARATION := '0' *)


FOR  i := 0 TO 1 DO
        Actuators[i].LimMin := 0;
        Actuators[i].LimMax := 1000;
        Chk_Limit(index:=i);
END_FOR;

Actuators[0].Position := TRUNC(( iPos_1/32768.0) * 1000);
Actuators[1].Position := TRUNC(( iPos_2/32768.0) * 1000);


IF NOT Started THEN
        Actuators[0].CmdPos :=Actuators[0].Position;
        Actuators[1].CmdPos :=Actuators[1].Position;
END_IF

RunOnce := FALSE;

IF inStart THEN
        Started := TRUE;
END_IF

IF NOT inStop THEN
        Started := FALSE;
        Nulstil := FALSE;
END_IF

timeNulstil(IN:= (Started AND inStart),PT:=T#10s);
IF timeNulstil.Q THEN
        Nulstil := TRUE;
END_IF


outStarted := Started;


Act_Out;
```

```
V1_beregnet := Calc_V1();
V2_beregnet := Calc_V2();

Pos1 := Calc_Pos1(V:=V1_beregnet);
Pos2 := Calc_Pos2(V:=V2_beregnet);

ForwardKin_FB(V1:=V1_beregnet, V2:=V2_beregnet, X=>_X, Y=>_Y);


TiltXValue :=    ((Tilt_Y/32768.0)-0.5)  ;
TiltYValue :=   (0.5-(Tilt_X/32768.0))  ;
IF NOT Started THEN
        TiltXZero := TiltXValue;
        TiltYZero := TiltYValue;
END_IF
TiltXValue := TiltXValue - TiltXZero;
TiltYValue := TiltYValue - TiltYZero;


IF Nulstil AND (ABS(TiltXValue) + ABS(TiltYValue)> 0.1)  THEN
        Nulstil := FALSE;
END_IF

IF (NOT useLys) AND inVisionActive  AND  (bWM_N OR bWM_S OR bWM_E OR bWM_W)
THEN
        iN := BOOL_TO_INT(bWM_N);
        iS := BOOL_TO_INT(bWM_S);
        iE := BOOL_TO_INT(bWM_E);
        iW := BOOL_TO_INT(bWM_W);

        _StepSize :=50;
        _gain := 1.0;
        _Ki := 0.1;
        _Kp := 0.05;

        dXYtool2XYworld_FB(X_Tool := (iS -iN) *_StepSize , Y_Tool := (iW -
iE)*_StepSize, V_1 := V1_beregnet, V_2 := V2_beregnet, X_world => dX, Y_World =>
dY);

        InverseKin_FB(X:=MIN(MAX(_X+dX,MinX),MaxX),
Y:=MIN(MAX(_Y+dY,MinY),MaxY), V1=>_V1, V2=>_V2);

        Actuators[0].CmdPos := TRUNC(Calc_Pos1(V:= _V1));
        Actuators[1].CmdPos := TRUNC(Calc_Pos2(V:= _V2));
END_IF

(* 4 stk. lyssensor*)

IF ( useLys) AND inVisionActive  AND  (bWM_N OR bWM_S OR bWM_E OR bWM_W)  THEN
        iN := BOOL_TO_INT(bWM_N);
        iS := BOOL_TO_INT(bWM_S);
        iE := BOOL_TO_INT(bWM_E);
        iW := BOOL_TO_INT(bWM_W);

        _StepSize :=50;
        _gain := 0.15;
        _Ki := 0.2;
        _Kp := 0.1;

        dXYtool2XYworld_FB(X_Tool := (iN -iS) *_StepSize , Y_Tool := (iW -
iE)*_StepSize, V_1 := V1_beregnet, V_2 := V2_beregnet, X_world => dX, Y_World =>
dY);

        InverseKin_FB(X:=MIN(MAX(_X+dX,MinX),MaxX),
Y:=MIN(MAX(_Y+dY,MinY),MaxY), V1=>_V1, V2=>_V2);

        Actuators[0].CmdPos := TRUNC(Calc_Pos1(V:= _V1));
```

```
        Actuators[1].CmdPos := TRUNC(Calc_Pos2(V:= _V2));
END_IF

IF inSmokeActive  AND  (SmokeDetectLen > 10)  THEN
        _StepSize := 200;

        _gain := 0.1;
        _Ki := 0.1;
        _Kp := 0.05;

        dXYtool2XYworld_FB(X_Tool :=
_StepSize*COS((SmokeDetectAng-20.0)*PI/180.0)  , Y_Tool :=
_StepSize*SIN((SmokeDetectAng-20.0)*PI/180.0), V_1 := V1_beregnet, V_2 :=
V2_beregnet, X_world => dX, Y_World => dY);

        InverseKin_FB(X:=MIN(MAX(_X+dX,MinX),MaxX),
Y:=MIN(MAX(_Y+dY,MinY),MaxY), V1=>_V1, V2=>_V2);

        Actuators[0].CmdPos := TRUNC(Calc_Pos1(V:= _V1));
        Actuators[1].CmdPos := TRUNC(Calc_Pos2(V:= _V2));
END_IF

Test_Tilt := TRUE;
IF Test_Tilt THEN
        _StepSize := 1000;
        TiltDeadband := 0.10;
        IF (ABS(TiltXValue) >TiltDeadband) OR (ABS(TiltYValue) >TiltDeadband)
THEN  (*  0.1 = 9 degres *)
                dXYtool2XYworld_FB(X_Tool :=
_StepSize*TiltXValue*COS(AngOffset*PI/180.0) +
_StepSize*TiltYValue*SIN(AngOffset*PI/180.0)  , Y_Tool :=
_StepSize*TiltXValue*SIN(AngOffset*PI/180.0) +
_StepSize*TiltYValue*COS(AngOffset*PI/180.0), V_1 := V1_beregnet, V_2 :=
V2_beregnet, X_world => dX, Y_World => dY);

                InverseKin_FB(X:=MIN(MAX(_X+dX,MinX),MaxX),
Y:=MIN(MAX(_Y+dY,MinY),MaxY), V1=>_V1, V2=>_V2);

                Actuators[0].CmdPos := TRUNC(Calc_Pos1(V:= _V1));
                Actuators[1].CmdPos := TRUNC(Calc_Pos2(V:= _V2));
                _gain := 0.1;
                _Ki := 0.1;
                _Kp := 0.1;
        END_IF
END_IF


(* PID regulering *)

_d1 := (Actuators[0].CmdPos -Actuators[0].Position);
_d2 := (Actuators[1].CmdPos -Actuators[1].Position);

IF ABS(_d1) < _deadband THEN
        _d1:= 0;
        _i1 := 0;
END_IF
IF ABS(_d2) < _deadband THEN
        _d2:= 0;
        _i2:=0;
END_IF

_i1 := _i1 + _d1*_dT;
_i2 := _i2 + _d2*_dT;

_s1 := _gain*ABS(_Kp*_d1 + _Ki * _i1);
_s2 := _gain*ABS(_Kp*_d2 + _Ki * _i2);

Actuators[0].Speed := TRUNC(100.0 * _s1 / MAX(1.0, MAX(_s1,_s2)));
```

```
Actuators[1].Speed := TRUNC(100.0 *  _s2 / MAX(1.0, MAX(_s1,_s2)));

END_PROGRAM


(* @NESTEDCOMMENTS := 'Yes' *)
(* @PATH := '\/Actuator' *)
(* @OBJECTFLAGS := '0, 8' *)
(* @SYMFILEFLAGS := '2048' *)
FUNCTION_BLOCK AktuatorCtrl
VAR_INPUT
        Enbl: BOOL;
        Act : ACTUATOR;
        Hyst: INT;
END_VAR
VAR_OUTPUT
        Fwd : BOOL;
        Bwd : BOOL;
        InLimit : BOOL;
END_VAR
VAR
END_VAR
(* @END_DECLARATION := '0' *)
_FBD_BODY
_NETWORKS : 4
_NETWORK

_COMMENT
''
_END_COMMENT
_ASSIGN
_OPERATOR
_BOX_EXPR : 2
_OPERATOR
_BOX_EXPR : 2
_OPERAND
_EXPRESSION
_POSITIV
Act.Position
_OPERAND
_EXPRESSION
_POSITIV
Act.LimMax+1
_EXPRESSION
_POSITIV
LT
_OPERATOR
_BOX_EXPR : 2
_OPERAND
_EXPRESSION
_POSITIV
Act.Position
_OPERAND
_EXPRESSION
_POSITIV
Act.LimMin-1
_EXPRESSION
_POSITIV
GT
_EXPRESSION
_NEGATIV
AND
_EXPRESSION
_POSITIV
_OUTPUTS : 1
_OUTPUT
_POSITIV
_NO_SET
```

```
InLimit
_NETWORK

_COMMENT
''
_END_COMMENT
_ASSIGN
_OPERATOR
_BOX_EXPR : 2
_OPERATOR
_BOX_EXPR : 3
_OPERATOR
_BOX_EXPR : 2
_OPERAND
_EXPRESSION
_POSITIV
Act.Position
_OPERAND
_EXPRESSION
_POSITIV
Act.CmdPos-Hyst
_EXPRESSION
_POSITIV
LT
_OPERATOR
_BOX_EXPR : 2
_OPERAND
_EXPRESSION
_POSITIV
Act.CmdPos
_OPERAND
_EXPRESSION
_POSITIV
Act.LimMax+1
_EXPRESSION
_POSITIV
LT
_OPERATOR
_BOX_EXPR : 2
_OPERAND
_EXPRESSION
_POSITIV
Act.CmdPos
_OPERAND
_EXPRESSION
_POSITIV
Act.LimMin-1
_EXPRESSION
_POSITIV
GT
_EXPRESSION
_POSITIV
AND
_OPERAND
_EXPRESSION
_POSITIV
Enbl
_EXPRESSION
_POSITIV
AND
_EXPRESSION
_POSITIV
_OUTPUTS : 1
_OUTPUT
_POSITIV
_NO_SET
Fwd
_NETWORK
```

```
_COMMENT
''
_END_COMMENT
_ASSIGN
_OPERATOR
_BOX_EXPR : 2
_OPERATOR
_BOX_EXPR : 3
_OPERATOR
_BOX_EXPR : 2
_OPERAND
_EXPRESSION
_POSITIV
Act.Position
_OPERAND
_EXPRESSION
_POSITIV
Act.CmdPos+Hyst
_EXPRESSION
_POSITIV
GT
_OPERATOR
_BOX_EXPR : 2
_OPERAND
_EXPRESSION
_POSITIV
Act.CmdPos
_OPERAND
_EXPRESSION
_POSITIV
Act.LimMax+1
_EXPRESSION
_POSITIV
LT
_OPERATOR
_BOX_EXPR : 2
_OPERAND
_EXPRESSION
_POSITIV
Act.CmdPos
_OPERAND
_EXPRESSION
_POSITIV
Act.LimMin-1
_EXPRESSION
_POSITIV
GT
_EXPRESSION
_POSITIV
AND
_OPERAND
_EXPRESSION
_POSITIV
Enbl
_EXPRESSION
_POSITIV
AND
_EXPRESSION
_POSITIV
_OUTPUTS : 1
_OUTPUT
_POSITIV
_NO_SET
Bwd
_NETWORK

_COMMENT
```

```
''
_END_COMMENT
_RET
_EXPRESSION
_POSITIV
_OPERATOR
_BOX_EXPR : 2
_OPERAND
_EXPRESSION
_POSITIV
Act.FWD
_OPERAND
_EXPRESSION
_POSITIV
Act.BWD
_EXPRESSION
_POSITIV
OR

END_FUNCTION_BLOCK


(* @NESTEDCOMMENTS := 'Yes' *)
(* @PATH := '\/Actuator' *)
(* @OBJECTFLAGS := '0, 8' *)
(* @SYMFILEFLAGS := '2048' *)
FUNCTION Calc_Pos1 : REAL
VAR_INPUT
        V : REAL;
END_VAR
VAR
        c : REAL;
END_VAR
VAR CONSTANT
        c_Start: REAL := 660;
        a : REAL := 235;
        b : REAL := 845;
        ActDeltaL :REAL := 370.0;
END_VAR
(* @END_DECLARATION := '0' *)
(*      Calculates the actuator position of first actuator from the angle *)


c := SQRT(a*a + b*b - 2*a*b*COS(V*PI/180.0));

V := ACOS(( a*a + b*b - c*c)/(2*a*b));

Calc_Pos1 :=  (c - c_Start)*Actuators[0].LimMax/ActDeltaL;




END_FUNCTION


(* @NESTEDCOMMENTS := 'Yes' *)
(* @PATH := '\/Actuator' *)
(* @OBJECTFLAGS := '0, 8' *)
(* @SYMFILEFLAGS := '2048' *)
FUNCTION Calc_Pos2 : REAL
VAR_INPUT
        V : REAL;
END_VAR
VAR
        c : REAL;
END_VAR
VAR CONSTANT
```

```
        c_Start: REAL := 660;
        a : REAL := 300;
        b : REAL := 812.95;
        ActDeltaL :REAL := 370.0;
END_VAR
(* @END_DECLARATION := '0' *)
(*      Calculates the actuator position of second actuator from the angle *)

c := SQRT(a*a + b*b - 2*a*b*COS(V*PI/180.0));

V := ACOS(( a*a + b*b - c*c)/(2*a*b));

Calc_Pos2 := (c - c_Start)*Actuators[1].LimMax/ActDeltaL;



END_FUNCTION


(* @NESTEDCOMMENTS := 'Yes' *)
(* @PATH := '\/Actuator' *)
(* @OBJECTFLAGS := '0, 8' *)
(* @SYMFILEFLAGS := '2048' *)
FUNCTION Calc_V1 : REAL
VAR_INPUT
END_VAR
VAR
        c : REAL;
        V: REAL;
        tt: REAL;
END_VAR
VAR CONSTANT
        c_Start: REAL := 660;
        a : REAL := 235;
        b : REAL := 845;
        ActDeltaL :REAL := 370.0;
END_VAR
(* @END_DECLARATION := '0' *)
(*      Calculates the actuator angle of first actuator from the length position
 *)

c := Actuators[0].Position* (ActDeltaL/Actuators[0].LimMax) + c_Start;

V := ACOS(( a*a + b*b - c*c)/(2*a*b));

Calc_V1 := V*180/PI ;



END_FUNCTION


(* @NESTEDCOMMENTS := 'Yes' *)
(* @PATH := '\/Actuator' *)
(* @OBJECTFLAGS := '0, 8' *)
(* @SYMFILEFLAGS := '2048' *)
FUNCTION Calc_V2 : REAL
VAR_INPUT
END_VAR
VAR
        c : REAL;
        V: REAL;
        tt: REAL;
END_VAR
VAR CONSTANT
        c_Start: REAL := 660;
```

```
        a : REAL := 300;
        b : REAL := 812.95;
        ActDeltaL :REAL := 370.0;
END_VAR
(* @END_DECLARATION := '0' *)
(*      Calculates the actuator angle of second actuator from the length
position  *)

c := Actuators[1].Position* (ActDeltaL/Actuators[1].LimMax) + c_Start;

V := ACOS(( a*a + b*b - c*c)/(2*a*b));

Calc_V2 := V*180/PI ;



END_FUNCTION


(* @NESTEDCOMMENTS := 'Yes' *)
(* @PATH := '\/Actuator' *)
(* @OBJECTFLAGS := '0, 8' *)
(* @SYMFILEFLAGS := '2048' *)
FUNCTION Chk_Limit : BOOL
VAR_INPUT
        index : INT;
END_VAR
VAR
END_VAR
(* @END_DECLARATION := '0' *)
_FBD_BODY
_NETWORKS : 1
_NETWORK

_COMMENT
''
_END_COMMENT
_ASSIGN
_OPERATOR
_BOX_EXPR : 2
_OPERATOR
_BOX_EXPR : 2
_OPERAND
_EXPRESSION
_POSITIV
Actuators[index].Position
_OPERAND
_EXPRESSION
_POSITIV
Actuators[index].LimMax+1
_EXPRESSION
_POSITIV
LT
_OPERATOR
_BOX_EXPR : 2
_OPERAND
_EXPRESSION
_POSITIV
Actuators[index].Position
_OPERAND
_EXPRESSION
_POSITIV
Actuators[index].LimMin-1
_EXPRESSION
_POSITIV
GT
_EXPRESSION
```

```
_NEGATIV
AND
_EXPRESSION
_POSITIV
_OUTPUTS : 1
_OUTPUT
_POSITIV
_NO_SET
Actuators[index].InLimit

END_FUNCTION


(* @NESTEDCOMMENTS := 'Yes' *)
(* @PATH := '\/Actuator' *)
(* @OBJECTFLAGS := '0, 8' *)
(* @SYMFILEFLAGS := '2048' *)
FUNCTION_BLOCK dXYtool2XYworld
VAR_INPUT
        X_Tool : REAL;
        Y_Tool :REAL;
        V_1 : REAL;
        V_2 : REAL;
END_VAR
VAR_OUTPUT
        X_World : REAL;
        Y_World :REAL;
END_VAR
VAR
        _V: REAL;
END_VAR
(* @END_DECLARATION := '0' *)

(*      Calulates a direction in tool coordinates to world coordinates.
        Takes V1 and V2 as argument
*)

_V := ((180-V_1) +(180-V_2) ) *PI/180;

X_World := X_Tool* COS(_V) + Y_Tool*SIN(_V);

Y_World := X_Tool* SIN(_V) - Y_Tool*COS(_V);


END_FUNCTION_BLOCK


(* @NESTEDCOMMENTS := 'Yes' *)
(* @PATH := '\/Actuator' *)
(* @OBJECTFLAGS := '0, 8' *)
(* @SYMFILEFLAGS := '2048' *)
FUNCTION_BLOCK ForwardKin
VAR_INPUT
        V1 : REAL;
        V2: REAL;
END_VAR
VAR_OUTPUT
        X : REAL;
        Y : REAL;
END_VAR
VAR
        L1 : REAL := 2000.0;
        L2 : REAL := 2000.0;
        _V1: REAL;
        _V2: REAL;
END_VAR
(* @END_DECLARATION := '0' *)
_V1 := (180-V1) * PI  / 180.0;
```

```
_V2 := (180-V2) * PI  / 180.0;

X := L1*COS(_V1) + L2*COS(_V1+_V2);

Y:=L1*SIN(_V1) + L2*SIN(_V1+_V2);


END_FUNCTION_BLOCK


(* @NESTEDCOMMENTS := 'Yes' *)
(* @PATH := '\/Actuator' *)
(* @OBJECTFLAGS := '0, 8' *)
(* @SYMFILEFLAGS := '2048' *)
FUNCTION_BLOCK InverseKin
VAR_INPUT
        X : REAL;
        Y : REAL;
END_VAR
VAR_OUTPUT
        V1 : REAL;
        V2: REAL;
END_VAR
VAR
        L1 : REAL := 2000.0;
        L2 : REAL := 2000.0;
END_VAR
(* @END_DECLARATION := '0' *)



V2 := 180 -  (180 * ACOS((x*x + y*y - L1*L1 - L2*L2)/(2*L1*L2)) / PI);

V1 := - (180 * (ATAN(y/x) - ACOS((x*x + y*y + L1*L1 - L2*L2)/(2*L1*SQRT(x*x +
y*y)))) / PI);

IF V1 < 0 THEN
        V1 := V1 + 180.0;
END_IF

END_FUNCTION_BLOCK


(* @NESTEDCOMMENTS := 'Yes' *)
(* @PATH := '\/Actuator' *)
(* @OBJECTFLAGS := '0, 8' *)
(* @SYMFILEFLAGS := '2048' *)
FUNCTION_BLOCK Nulstil
VAR_INPUT
END_VAR
VAR_OUTPUT
        Fwd_1: BOOL;
        Fwd_2: BOOL;
        Bwd_1: BOOL;
        Bwd_2: BOOL;
END_VAR
VAR
        _State: INT := 0;
        _Timer : TON;
        _NextState: INT;
        ActCtrl: AktuatorCtrl;
END_VAR
VAR_IN_OUT
END_VAR
(* @END_DECLARATION := '0' *)
```

```
IF Nulstil THEN
        IF (_State = 0) THEN
                _State := 100;
                _NextState := 100;
        END_IF
ELSE
        _State := 0;
        _NextState :=0;
END_IF


_Timer(IN:=(_State = _NextState AND _State> 0),PT:=T#30s);

IF _State <> 0 THEN
        _State:= _NextState;
        CASE _State OF
        100:
                Fwd_1 := TRUE;
                Bwd_1 := FALSE;
                Fwd_2 := TRUE;
                Bwd_2 := FALSE;
                IF _Timer.Q THEN
                        _NextState := 200;
                END_IF
        200:
                Fwd_1 := FALSE;
                Bwd_1 := FALSE;
                Fwd_2 := FALSE;
                Bwd_2 := TRUE;
                IF _Timer.Q THEN
                        _NextState := 300;
                END_IF
        300:
                Fwd_1 := FALSE;
                Bwd_1 :=TRUE;
                Fwd_2 := TRUE;
                Bwd_2 := FALSE;
                IF _Timer.Q THEN
                        _NextState := 0;
                        Nulstil := FALSE;
                        Actuators[0].Reset := TRUE;
                        Actuators[1].Reset := TRUE;
                END_IF

         END_CASE;
ELSE
        Fwd_1:=FALSE;
        Fwd_2:=FALSE;
        Bwd_1:=FALSE;
        Bwd_2:=FALSE;
END_IF


END_FUNCTION_BLOCK


(* @NESTEDCOMMENTS := 'Yes' *)
(* @PATH := '\/SmokeDetect' *)
(* @OBJECTFLAGS := '0, 8' *)
(* @SYMFILEFLAGS := '2048' *)
FUNCTION_BLOCK CalculateAveragePulseWidth
VAR_INPUT
        BufferSize: UDINT;
        Buffer: ARRAY[0..31] OF REAL;
END_VAR
```

```
VAR_OUTPUT
        PulseWidth_aveOUT: REAL;
END_VAR
VAR
        Buffer_Pointer: UDINT;
        Temp: REAL;
END_VAR
(* @END_DECLARATION := '0' *)
Buffer_Pointer := 0;
Temp :=0;

FOR Buffer_Pointer := 0 TO BufferSize-1 BY 1 DO
        Temp:= Temp + Buffer[Buffer_Pointer];
END_FOR;

PulseWidth_aveOUT:=INT_TO_REAL(REAL_TO_INT( (Temp/BufferSize)));
END_FUNCTION_BLOCK


(* @NESTEDCOMMENTS := 'Yes' *)
(* @PATH := '\/SmokeDetect' *)
(* @OBJECTFLAGS := '0, 8' *)
(* @SYMFILEFLAGS := '2048' *)
FUNCTION_BLOCK CalculateResultVector
VAR_INPUT
        VectorXcoordinate1: REAL;
        VectorYcoordinate1: REAL;
        VectorXcoordinate2: REAL;
        VectorYcoordinate2: REAL;
        VectorXcoordinate3: REAL;
        VectorYcoordinate3: REAL;

END_VAR
VAR
        VectorXcomponentResult: REAL := 0;
        VectorYcomponentResult: REAL := 0;

END_VAR
VAR_OUTPUT
        VectorDgResult: REAL;
        VectorlnResult: REAL;
END_VAR
(* @END_DECLARATION := '0' *)
        VectorXcomponentResult :=
VectorXcoordinate1+VectorXcoordinate2+VectorXcoordinate3;
        VectorYcomponentResult :=
VectorYcoordinate1+VectorYcoordinate2+VectorYcoordinate3;

        VectorLnResult := SQRT(EXPT(VectorXcomponentResult,2) +
EXPT(VectorYcomponentResult,2));

        IF VectorXcomponentResult> 0  THEN
                VectorDgResult :=
ATAN(VectorYcomponentResult/VectorXcomponentResult)*180/PI;
        END_IF

        IF VectorXcomponentResult< 0 AND VectorYcomponentResult >= 0 THEN
                VectorDgResult :=
ATAN(VectorYcomponentResult/VectorXcomponentResult)*180/PI+180;
        END_IF

        IF VectorXcomponentResult< 0 AND VectorYcomponentResult < 0 THEN
                VectorDgResult :=
ATAN(VectorYcomponentResult/VectorXcomponentResult)*180/PI-180;
        END_IF


        IF VectorXcomponentResult = 0 AND VectorYcomponentResult > 0 THEN
```

```
                VectorDgResult :=180/2;
        END_IF

        IF VectorXcomponentResult= 0 AND VectorYcomponentResult < 0 THEN
                VectorDgResult := -180/2;
        END_IF

        IF VectorXcomponentResult= 0 AND VectorYcomponentResult = 0 THEN
                VectorDgResult :=0;
        END_IF

        IF VectorDgResult <0 THEN
        VectorDgResult:=        VectorDgResult+ 360;
        END_IF
END_FUNCTION_BLOCK


(* @NESTEDCOMMENTS := 'Yes' *)
(* @PATH := '\/SmokeDetect' *)
(* @OBJECTFLAGS := '0, 8' *)
(* @SYMFILEFLAGS := '2048' *)
FUNCTION_BLOCK CreateXYVectorComponents
VAR_INPUT
        VectorAngleDg: REAL;
        VecrorLength: REAL;
END_VAR

VAR
END_VAR
VAR_OUTPUT
        VectorXcomponent: REAL;
        VectorYcomponent: REAL;
END_VAR
(* @END_DECLARATION := '0' *)
 VectorXcomponent := COS((VectorAngleDg) * PI / 180) * VecrorLength;
 VectorYcomponent := SIN((VectorAngleDg) * PI / 180) * VecrorLength;
END_FUNCTION_BLOCK


(* @NESTEDCOMMENTS := 'Yes' *)
(* @PATH := '\/SmokeDetect' *)
(* @OBJECTFLAGS := '0, 8' *)
(* @SYMFILEFLAGS := '2048' *)
PROGRAM DetectMain
VAR
        (*FUNCTION BLOCK INSTANCES*)
        MeasurePulseWidth_FB: MeasurePulseWidth;
        CalculateAveragePulseWidth_FB: CalculateAveragePulseWidth;
        NormalizePulseWidth_FB: NormalizePulseWidth;
        CreateXYVectorComponents_FB: CreateXYVectorComponents;
        CalculateResultVector_FB: CalculateResultVector;
        RoundReal_FB: RoundReal;

        (*GENERAL COUNTER*)
        Counter: UDINT:=0;

        RESET :  BOOL ;

        (*INPUTVARIABLES*)
        (*EL1252 Unit 1 Channel 0*)
        EL1252_1LachPos0_32LSb AT %I*: UDINT;
        EL1252_1LachPos0_32MSb AT %I*: UDINT;
        EL1252_1LachNeg0_32LSb AT %I*: UDINT;
        EL1252_1LachNeg0_32MSb AT %I*: UDINT;
        EL1252_1Status0 AT %I*: BYTE;
        EL1252_1In0 AT %I*: BOOL;
        (*EL1252 Unit 1 Channel 1*)
        EL1252_1LachPos1_32LSb AT %I*: UDINT;
```

```
EL1252_1LachPos1_32MSb AT %I*: UDINT;
EL1252_1LachNeg1_32LSb AT %I*: UDINT;
EL1252_1LachNeg1_32MSb AT %I*: UDINT;
EL1252_1Status1 AT %I*: BYTE;
EL1252_1In1 AT %I*: BOOL;

(*EL1252 Unit 2 Channel 0*)
EL1252_2LachPos0_32LSb AT %I*: UDINT;
EL1252_2LachPos0_32MSb AT %I*: UDINT;
EL1252_2LachNeg0_32LSb AT %I*: UDINT;
EL1252_2LachNeg0_32MSb AT %I*: UDINT;
EL1252_2Status0 AT %I*: BYTE;
EL1252_2In0 AT %I*: BOOL;

(*EL1252 Unit 2 Channel 1*)
EL1252_2LachPos1_32LSb AT %I*: UDINT;
EL1252_2LachPos1_32MSb AT %I*: UDINT;
EL1252_2LachNeg1_32LSb AT %I*: UDINT;
EL1252_2LachNeg1_32MSb AT %I*: UDINT;
EL1252_2Status1 AT %I*: BYTE;
EL1252_2In1 AT %I*: BOOL;

(*RUNTIME REGISTERS*)
(*Generic*)
PulseWidth: REAL;
Err: BOOL;

(*Specific*)
(*Sensor 1*)
PrevFALL1: UDINT;
Buffer1: ARRAY[0..31] OF REAL;
PulseWidthAllTimeAbsoluteMax1: REAL := 1;
PulseWidthAve_PercentOfAbsoluteMax1: REAL;
MeasureCount1 : REAL :=0;
ErrCount1: REAL :=0;
ErrPercent1: REAL := 0;
VectorAngleDg1: REAL :=20;
VecrorLength1: REAL :=0;
VectorXcomponent1: REAL := 0;
VectorYcomponent1: REAL :=0;

(*Sensor 2*)
PrevFall2:UDINT;
Buffer2: ARRAY[0..31] OF REAL;
PulseWidthAllTimeAbsoluteMax2: REAL := 1;
PulseWidthAve_PercentOfAbsoluteMax2: REAL;
MeasureCount2 : REAL :=0;
ErrCount2: REAL :=0;
ErrPercent2: REAL := 0;
VectorAngleDg2: REAL :=140;
VectorLength2: REAL :=0;
VectorXcomponent2: REAL :=0;
VectorYcomponent2: REAL :=0;

(*Sensor 3*)
PrevFall3:UDINT;
Buffer3: ARRAY[0..31] OF REAL;
PulseWidthAllTimeAbsoluteMax3: REAL := 1;
PulseWidthAve_PercentOfAbsoluteMax3: REAL;
MeasureCount3 : REAL :=0;
ErrCount3: REAL :=0;
ErrPercent3: REAL := 0;
VectorAngleDg3: REAL := 260;
VecrorLength3: REAL := 0;
VectorXcomponent3: REAL := 0;
VectorYcomponent3: REAL := 0;

(*Cominations*)
```

```
        VectorAngleDgResult: REAL := 0;
        VectorLengthResult: REAL := 0;


        (*CONSTANTS*)
        BufferSizeConstant: UDINT := 32;
        VectorAngleDg: BOOL;
END_VAR

(* @END_DECLARATION := '0' *)
        RESET := Nulstil;

(*General Counter*)
        Counter:= Counter +1;

(*Reading mean values from sensores*)
        IF Counter MOD 2 = 0 THEN                       (*Ensures that data is
always ready - perhaps the cycle time for reading could just be adjusted - but
where?*)

(*EL1252, Unit 1 Channel 0 - Detector 1*)
        MeasurePulseWidth_FB(RISE := EL1252_1LachPos0_32LSb, FALL:=
EL1252_1LachNeg0_32LSb, CurFALL := PrevFALL1,IN_put := EL1252_1In0,
PulseWidthOUT =>PulseWidth, NewFALL => PrevFALL1, ErrOUT=>Err);
        IF PulseWidth <>0 THEN
                MeasureCount1 := MeasureCount1+1;
        END_IF;

        IF Err = 0 THEN
                NormalizePulseWidth_FB(PulseWidthIN:= PulseWidth,
PulseWidthINMAX := PulseWidthAllTimeAbsoluteMax1, PulseWidthOUT=> PulseWidth,
PulseWidthOUTMAX => PulseWidthAllTimeAbsoluteMax1);
                Buffer1 [Counter/2 MOD BufferSizeConstant]  := PulseWidth;
                CalculateAveragePulseWidth_FB(Buffer := Buffer1, BufferSize :=
BufferSizeConstant, PulseWidth_aveOUT => PulseWidthAve_PercentOfAbsoluteMax1);
                VecrorLength1 :=  PulseWidthAve_PercentOfAbsoluteMax1;
                (*VecrorLength1:= 1;
                VectorAngleDg1:= 180;*)
                CreateXYVectorComponents_FB(VectorAngleDg := VectorAngleDg1,
VecrorLength := VecrorLength1,  VectorXcomponent => VectorXcomponent1,
VectorYcomponent => VectorYcomponent1);

RoundReal_FB(INReal:=VectorXcomponent1,NumberOfDecimals:=2,OUTReal=>VectorXcompo
nent1);

RoundReal_FB(INReal:=VectorYcomponent1,NumberOfDecimals:=2,OUTReal=>VectorYcompo
nent1);
        ELSE
                ErrCount1 := ErrCount1+1;
        END_IF;
                ErrPercent1 := ErrCount1/MeasureCount1*100;


(*EL1252, Unit 1 Channel 1 - Detector 2*)
        MeasurePulseWidth_FB(RISE := EL1252_1LachPos1_32LSb, FALL:=
EL1252_1LachNeg1_32LSb, CurFALL := PrevFALL2,IN_put := EL1252_1In1,
PulseWidthOUT =>PulseWidth, NewFALL => PrevFALL2, ErrOUT=> Err);
        IF PulseWidth <>0 THEN
                MeasureCount2 := MeasureCount2+1;
        END_IF

        IF Err = 0 THEN
                NormalizePulseWidth_FB(PulseWidthIN:= PulseWidth,
PulseWidthINMAX := PulseWidthAllTimeAbsoluteMax2, PulseWidthOUT=> PulseWidth,
PulseWidthOUTMAX => PulseWidthAllTimeAbsoluteMax2);
                Buffer2 [Counter/2 MOD BufferSizeConstant]  := PulseWidth;
                CalculateAveragePulseWidth_FB(Buffer := Buffer2, BufferSize :=
BufferSizeConstant, PulseWidth_aveOUT => PulseWidthAve_PercentOfAbsoluteMax2);
```

```
                VectorLength2 := PulseWidthAve_PercentOfAbsoluteMax2;
                (*VectorLength2:= 1;
                VectorAngleDg2:= 300;*)
                CreateXYVectorComponents_FB(VectorAngleDg := VectorAngleDg2,
VecrorLength := VectorLength2,  VectorXcomponent => VectorXcomponent2,
VectorYcomponent => VectorYcomponent2);

RoundReal_FB(INReal:=VectorXcomponent2,NumberOfDecimals:=2,OUTReal=>VectorXcompo
nent2);

RoundReal_FB(INReal:=VectorYcomponent2,NumberOfDecimals:=2,OUTReal=>VectorYcompo
nent2);
        ELSE
                ErrCount2 := ErrCount2+1;
        END_IF;
        ErrPercent2 := ErrCount2/MeasureCount2*100;


(*EL1252, Unit 2 Channel 0 - Detector 3*)
        MeasurePulseWidth_FB(RISE := EL1252_2LachPos0_32LSb, FALL:=
EL1252_2LachNeg0_32LSb, CurFALL := PrevFALL3,IN_put := EL1252_2In0,
PulseWidthOUT =>PulseWidth, NewFALL => PrevFALL3,ErrOUT=> Err);
        IF PulseWidth <>0 THEN
                MeasureCount3 := MeasureCount3+1;
        END_IF
        IF Err =0 THEN
                NormalizePulseWidth_FB(PulseWidthIN:= PulseWidth,
PulseWidthINMAX := PulseWidthAllTimeAbsoluteMax3, PulseWidthOUT=> PulseWidth,
PulseWidthOUTMAX => PulseWidthAllTimeAbsoluteMax3);
                Buffer3 [Counter/2 MOD BufferSizeConstant]  := PulseWidth;
                CalculateAveragePulseWidth_FB(Buffer := Buffer3, BufferSize :=
BufferSizeConstant, PulseWidth_aveOUT => PulseWidthAve_PercentOfAbsoluteMax3);
                VecrorLength3 :=PulseWidthAve_PercentOfAbsoluteMax3;
                (*VecrorLength3 := 1;
                VectorAngleDg3 := 60;*)
                CreateXYVectorComponents_FB(VectorAngleDg := VectorAngleDg3,
VecrorLength := VecrorLength3,  VectorXcomponent => VectorXcomponent3,
VectorYcomponent => VectorYcomponent3);

RoundReal_FB(INReal:=VectorXcomponent3,NumberOfDecimals:=2,OUTReal=>VectorXcompo
nent3);

RoundReal_FB(INReal:=VectorYcomponent3,NumberOfDecimals:=2,OUTReal=>VectorYcompo
nent3);
        ELSE
                ErrCount3 := ErrCount3+1;
        END_IF;
        ErrPercent3 := ErrCount3/MeasureCount3*100;
END_IF

CalculateResultVector_FB(VectorXcoordinate1:= VectorXcomponent1,
VectorYcoordinate1:= Vectorycomponent1,VectorXcoordinate2:= VectorXcomponent2,
VectorYcoordinate2:= Vectorycomponent2,VectorXcoordinate3:= VectorXcomponent3,
VectorYcoordinate3:= Vectorycomponent3, VectorLnResult =>
VectorLengthResult,VectorDgResult => VectorAngleDgResult);


RoundReal_FB(INReal:=VectorLengthResult,NumberOfDecimals:=0,OUTReal=>VectorLengt
hResult);
RoundReal_FB(INReal:=VectorAngleDgResult,NumberOfDecimals:=0,OUTReal=>VectorAngl
eDgResult);

VectorLengthResult;
VectorAngleDgResult;

SmokeDetectAng := VectorAngleDgResult;
SmokeDetectLen := VectorLengthResult;
```

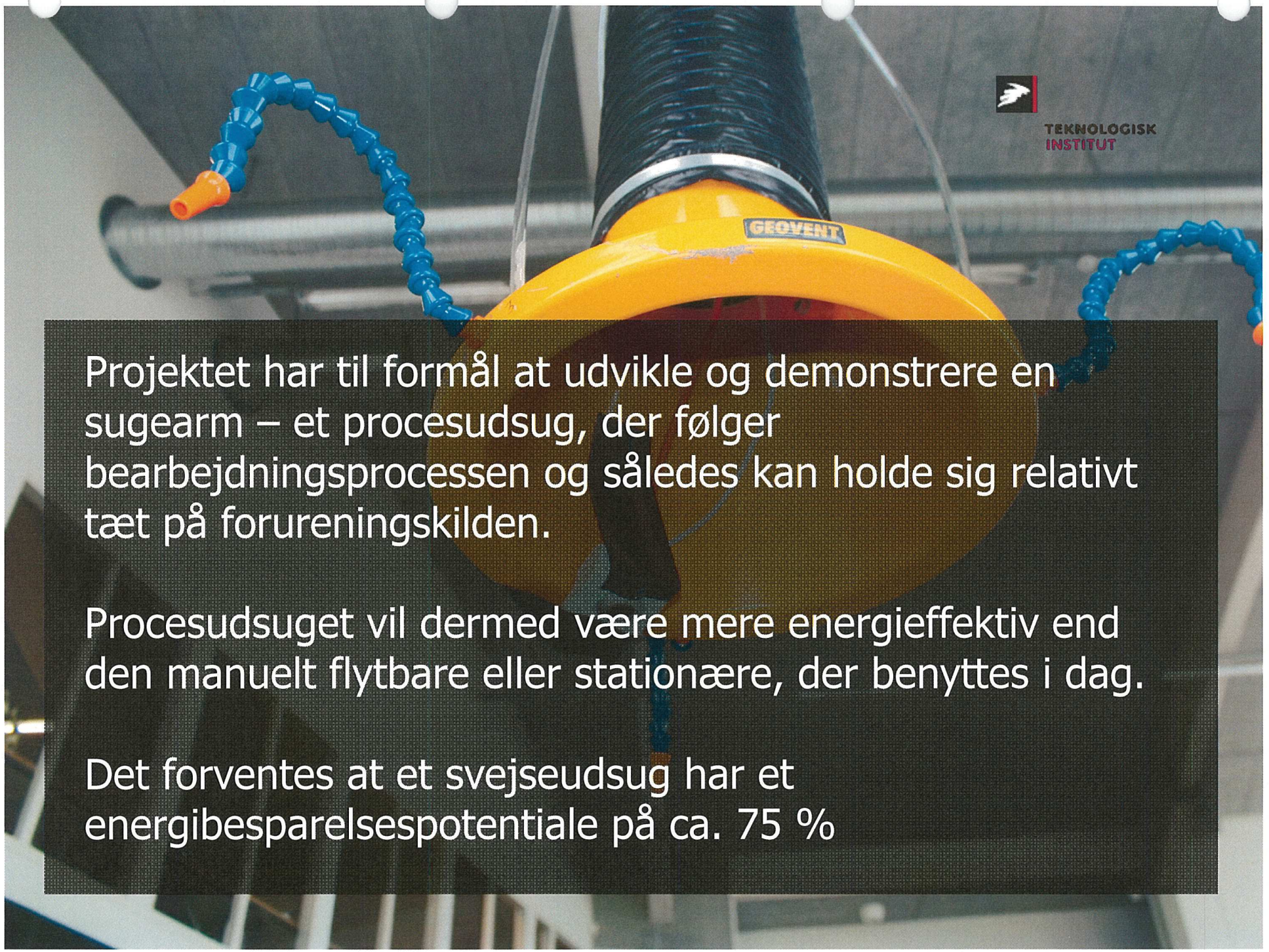
```
IF RESET=1 THEN
        Counter:=0;
        PulseWidth:=0;
        PrevFALL1:=0;
        PulseWidthAllTimeAbsoluteMax1:= 1;
        PulseWidthAve_PercentOfAbsoluteMax1:=0;
        MeasureCount1:=0;
        ErrCount1 :=0;
        ErrPercent1:= 0;
        VecrorLength1 :=0;
        VectorXcomponent1 := 0;
        VectorYcomponent1 :=0;
        PrevFall2:=0;
        PulseWidthAllTimeAbsoluteMax2:= 1;
        PulseWidthAve_PercentOfAbsoluteMax2:=0;
        MeasureCount2 :=0;
        ErrCount2:=0;
        ErrPercent2:= 0;
        VectorLength2 :=0;
        VectorXcomponent2 :=0;
        VectorYcomponent2 :=0;
        PrevFall3:=0;
        PulseWidthAllTimeAbsoluteMax3:= 1;
        PulseWidthAve_PercentOfAbsoluteMax3:=0;
        MeasureCount3 :=0;
        ErrCount3:=0;
        ErrPercent3:= 0;
        VecrorLength3 := 0;
        VectorXcomponent3 := 0;
        VectorYcomponent3 := 0;
        VectorAngleDgResult := 0;
        VectorLengthResult := 0;
END_IF;
END_PROGRAM


(* @NESTEDCOMMENTS := 'Yes' *)
(* @PATH := '\/SmokeDetect' *)
(* @OBJECTFLAGS := '0, 8' *)
(* @SYMFILEFLAGS := '2048' *)
FUNCTION_BLOCK MeasurePulseWidth
VAR_INPUT
        RISE: UDINT;
        FALL: UDINT;
        CurFALL: UDINT;
        IN_put: BOOL;
END_VAR
VAR_OUTPUT
        PulseWidthOUT: REAL;
        NewFALL : UDINT;
        ErrOUT: BOOL;
END_VAR
VAR
END_VAR
(* @END_DECLARATION := '0' *)
IF IN_put = 1 THEN
        ErrOUT:=0;
        IF CurFALL = FALL THEN
                        PulseWidthOUT := 0;
                ELSE
                        IF RISE > FALL THEN
                                PulseWidthOUT :=
UDINT_TO_REAL((RISE-FALL)/16#000003E8);
                        ELSE
                                PulseWidthOUT := UDINT_TO_REAL((RISE +
(16#FFFFFFFF - FALL))/16#000003E8); (*Handles overflow on 32LSb*)
                        END_IF
```

```
                END_IF
        NewFALL:= FALL;
        IF PulseWidthOUT > 100 THEN PulseWidthOUT :=0; END_IF;
        ELSE
        ErrOUT := 1;
END_IF
END_FUNCTION_BLOCK


(* @NESTEDCOMMENTS := 'Yes' *)
(* @PATH := '\/SmokeDetect' *)
(* @OBJECTFLAGS := '0, 8' *)
(* @SYMFILEFLAGS := '2048' *)
FUNCTION_BLOCK NormalizePulseWidth
VAR_INPUT
        PulseWidthIN: REAL;
        PulseWidthINMAX: REAL;
END_VAR
VAR_OUTPUT
        PulseWidthOUT: REAL;
        PulseWidthOUTMAX: REAL;

END_VAR
(* @END_DECLARATION := '0' *)
IF PulseWidthIN> PulseWidthINMAX THEN
        PulseWidthOUTMAX :=PulseWidthIN;
ELSE
        PulseWidthOUTMAX :=PulseWidthINMAX;
END_IF;

PulseWidthOUT := PulseWidthIN/PulseWidthOUTMAX*100;
END_FUNCTION_BLOCK


(* @NESTEDCOMMENTS := 'Yes' *)
(* @PATH := '\/SmokeDetect' *)
(* @OBJECTFLAGS := '0, 8' *)
(* @SYMFILEFLAGS := '2048' *)
FUNCTION_BLOCK RoundReal
VAR_INPUT
        INReal: REAL;
        NumberOfDecimals: REAL;
END_VAR
VAR
        Temp: REAL;
END_VAR
VAR_OUTPUT
        OUTReal: REAL;
END_VAR
(* @END_DECLARATION := '0' *)
        Temp := INReal*EXPT(10,NumberOfDecimals);
        Temp := INT_TO_REAL(REAL_TO_INT(Temp));
        OUTReal := Temp/EXPT(10,NumberOfDecimals);
END_FUNCTION_BLOCK


(* @NESTEDCOMMENTS := 'Yes' *)
(* @PATH := '' *)
(* @OBJECTFLAGS := '0, 8' *)
(* @SYMFILEFLAGS := '2048' *)
PROGRAM MAIN
VAR
        fbTimer: TON;
        bToggle AT %Q*: BOOL;
        i: INT;
END_VAR
(* @END_DECLARATION := '0' *)
```

```
DetectMain;
ActuatorMain;
END_PROGRAM
```

BILAG 18: PR – Udstillingsstand på Herning Industrimesse 2014

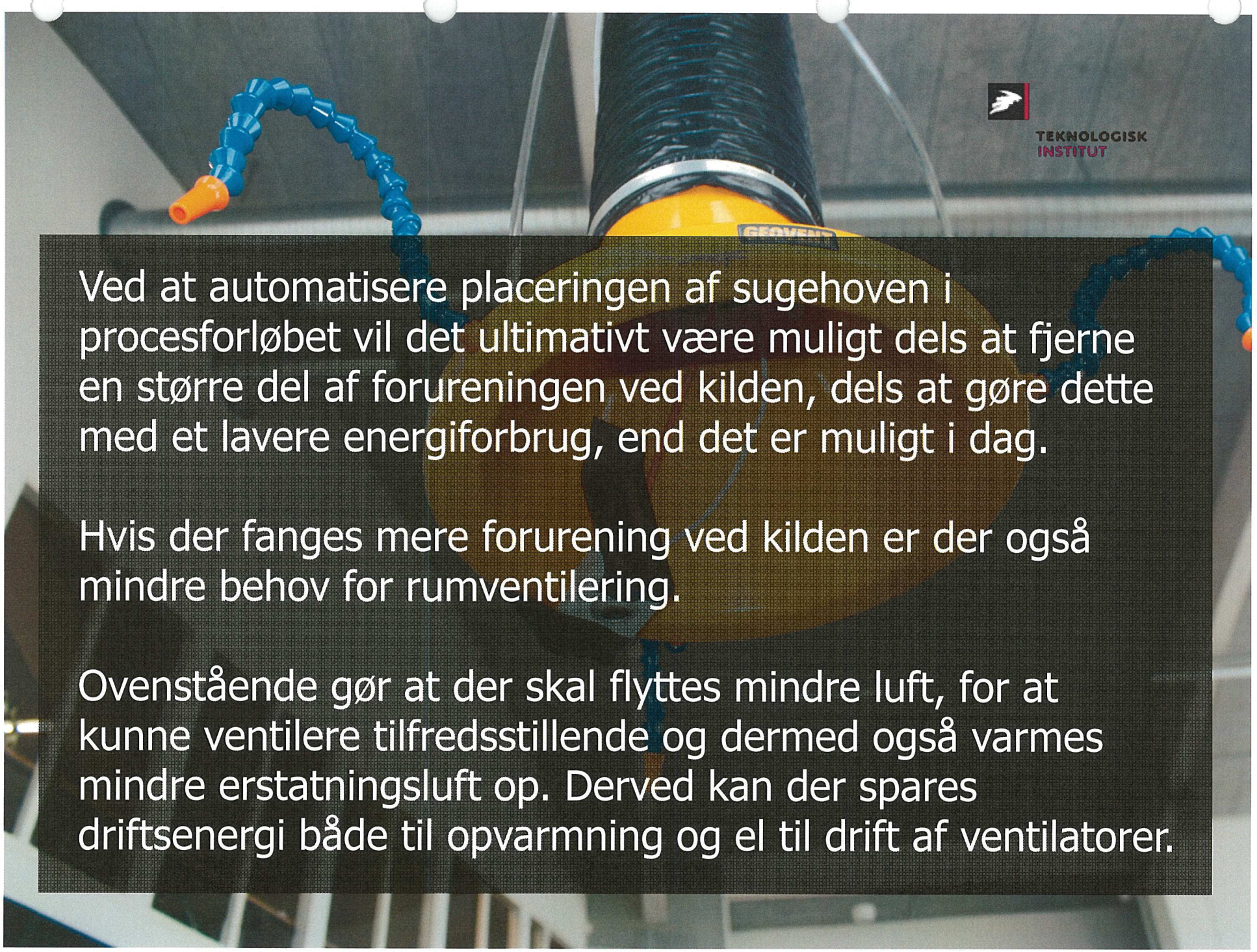

eUDP
Energiteknologisk udvikling og demonstration
TEKNOLOGISK
INSTITUT
Energieffektiv adaptiv procesudsugning
Projektet er støttet af:
"Energiteknologisk Udviklings- og Demonstrationsprogram (EUDP) "



Projektet har til formål at udvikle og demonstrere en sugearm – et procesudsug, der følger bearbejdningsprocessen og således kan holde sig relativt tæt på forureningskilden.

Procesudsuget vil dermed være mere energieffektiv end den manuelt flytbare eller stationære, der benyttes i dag.

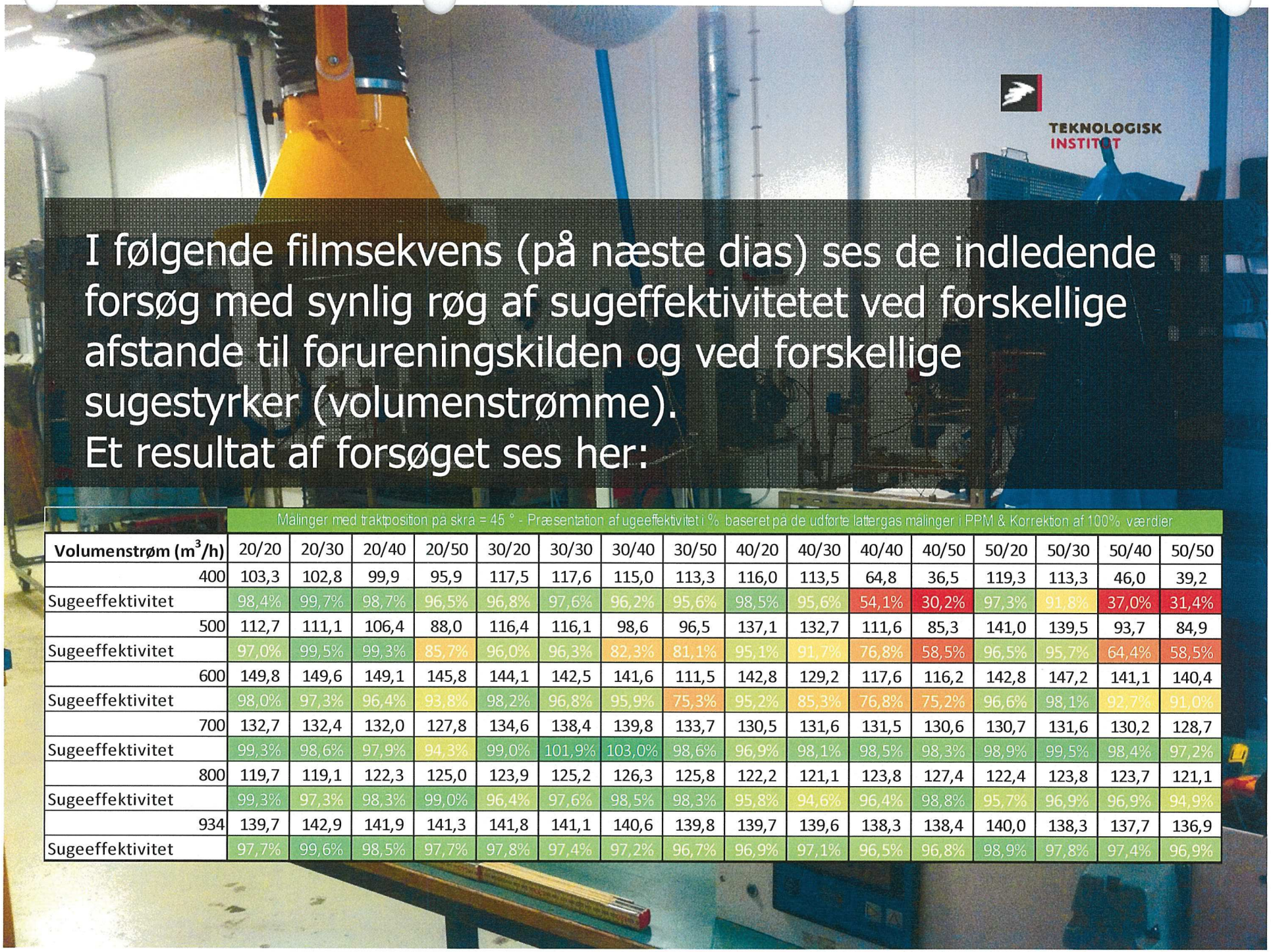
Det forventes at et svejseudsug har et energibesparelsespotentiale på ca. 75 %



Ved at automatisere placeringen af sugehoven i procesforløbet vil det ultimativt være muligt dels at fjerne en større del af forureningen ved kilden, dels at gøre dette med et lavere energiforbrug, end det er muligt i dag.

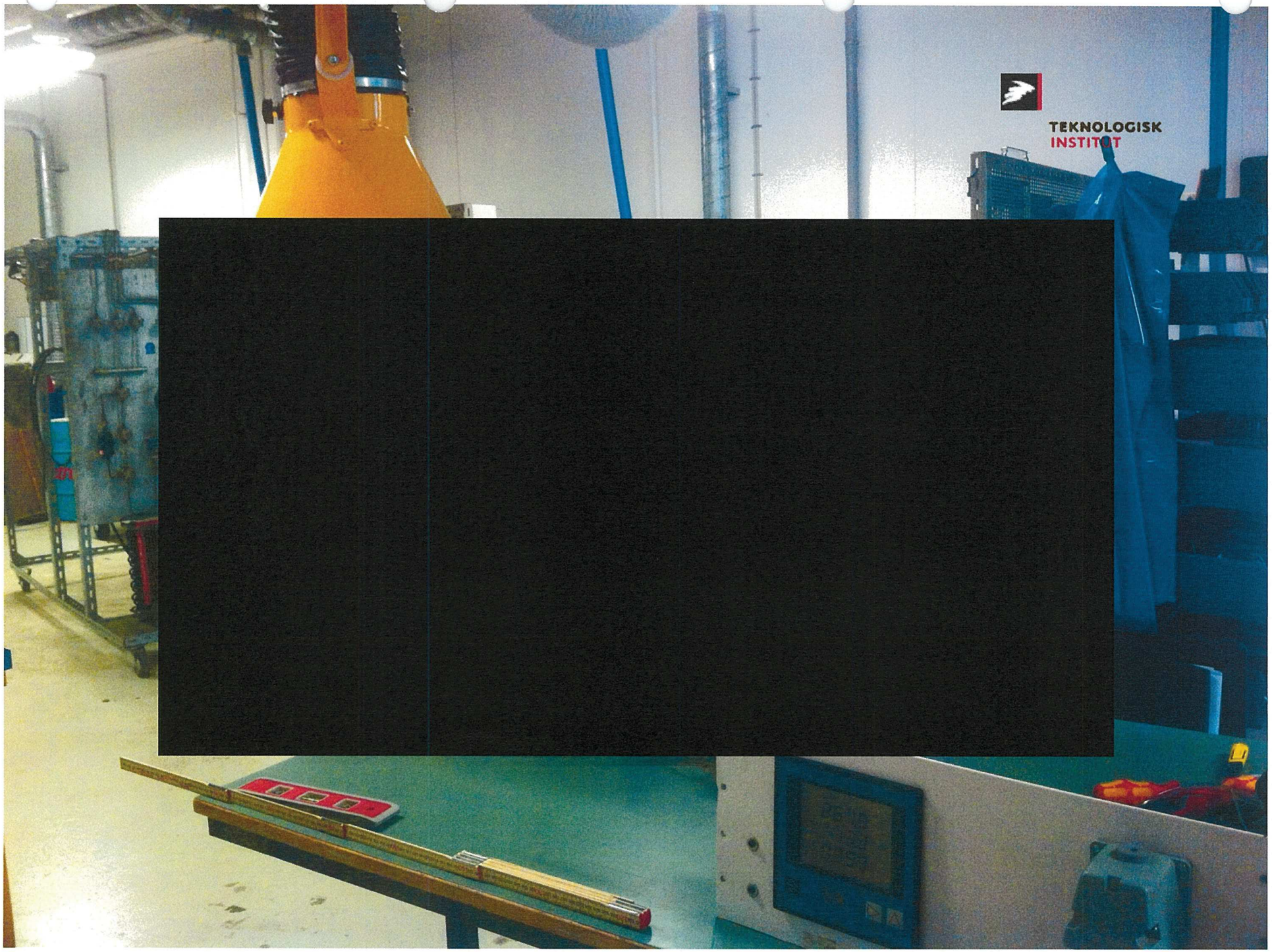
Hvis der fanges mere forurening ved kilden er der også mindre behov for rumventilering.

Ovenstående gør at der skal flyttes mindre luft, for at kunne ventilere tilfredsstillende og dermed også varmes mindre erstatningsluft op. Derved kan der spares driftsenergi både til opvarmning og el til drift af ventilatorer.

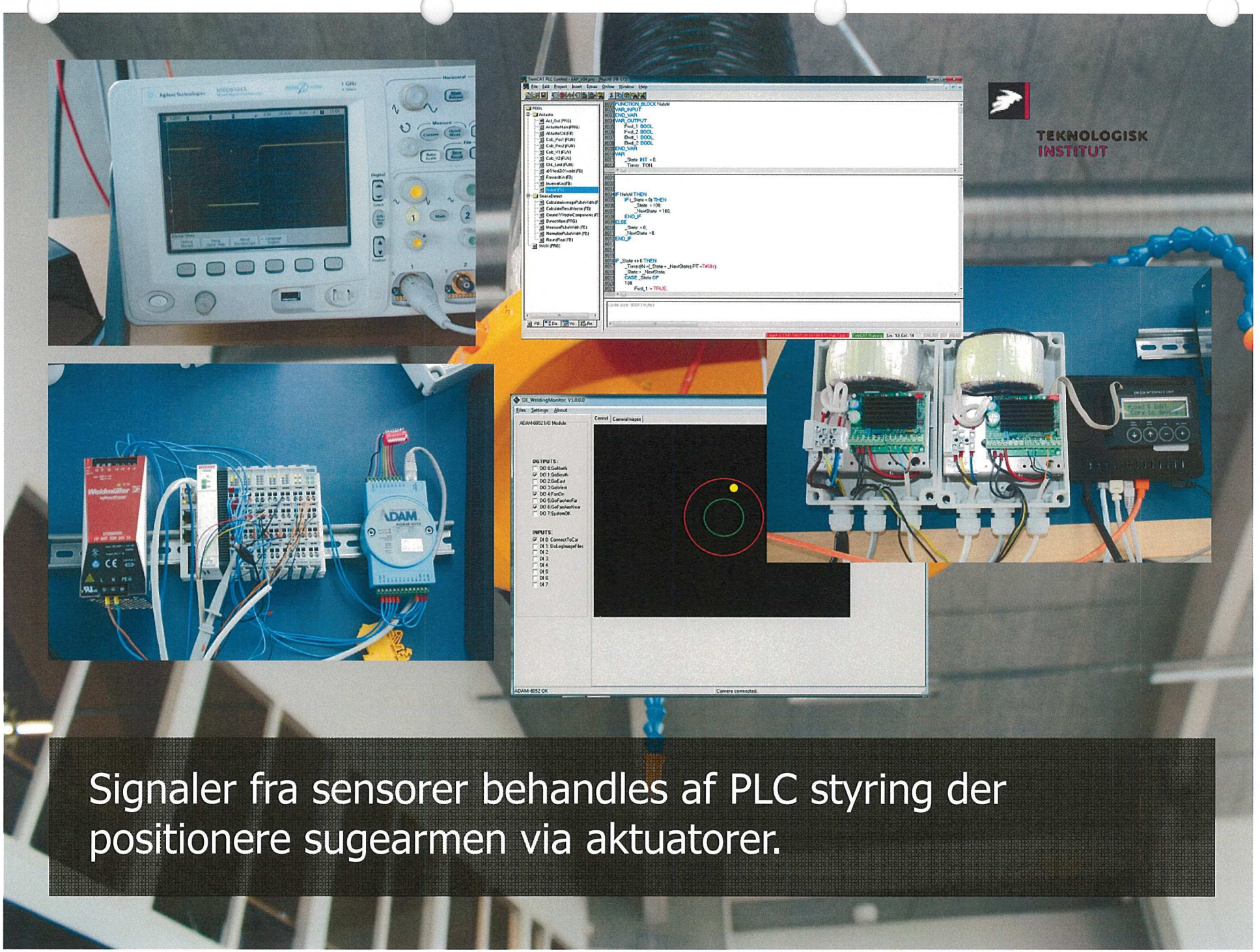


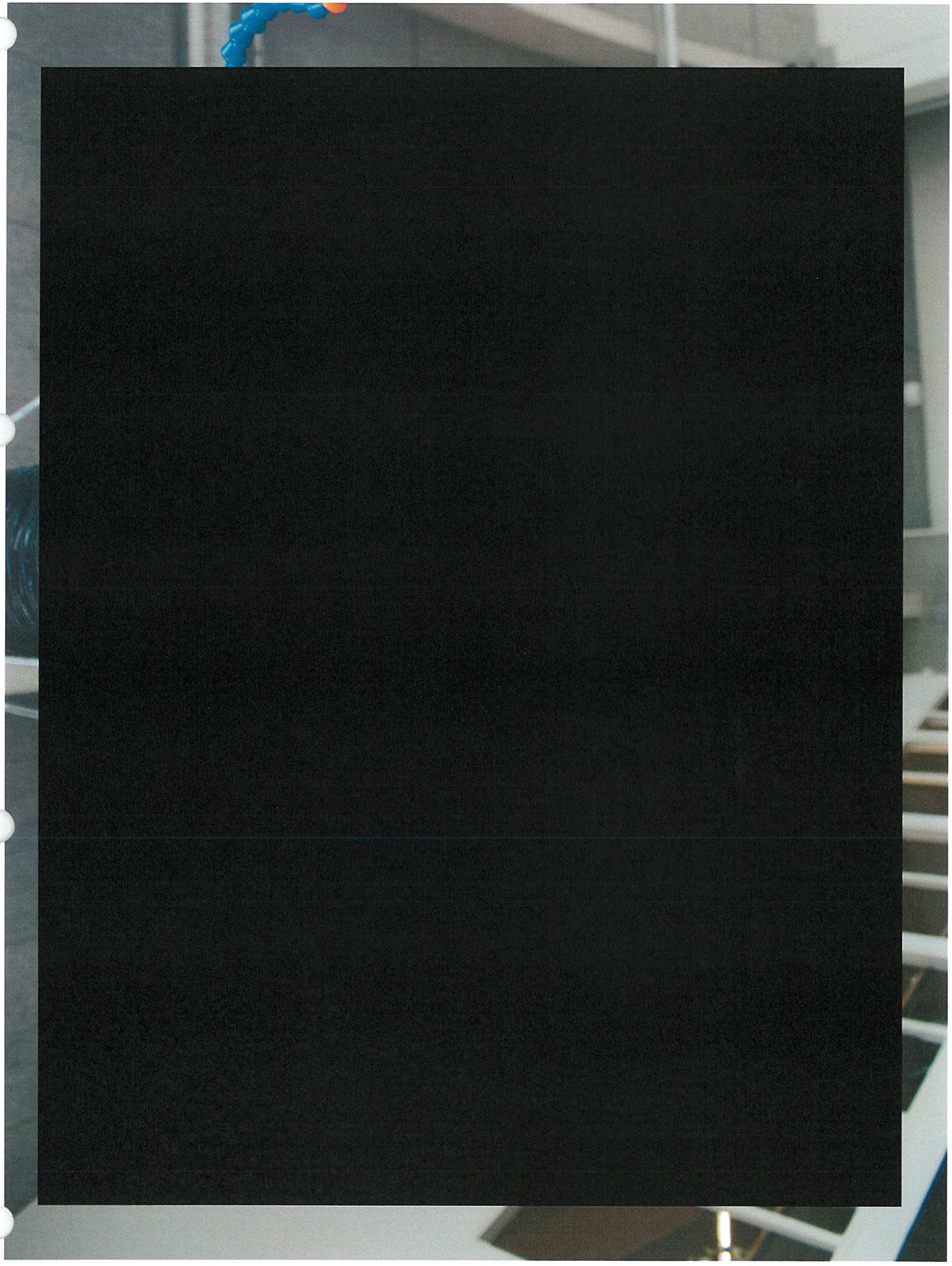
I følgende filmsekvens (på næste dias) ses de indledende forsøg med synlig røg af sugeffektivitetet ved forskellige afstande til forureningskilden og ved forskellige sugestyrker (volumenstrømme).
Et resultat af forsøget ses her:

	Målinger med traktposition på skrå = 45 ° - Præsentation af ugeeffektivitet i % baseret på de udførte lattergas målinger i PPM & Korrektion af 100% værdier															
Volumenstrøm (m^3/h)	20/20	20/30	20/40	20/50	30/20	30/30	30/40	30/50	40/20	40/30	40/40	40/50	50/20	50/30	50/40	50/50
400	103,3	102,8	99,9	95,9	117,5	117,6	115,0	113,3	116,0	113,5	64,8	36,5	119,3	113,3	46,0	39,2
Sugeeffektivitet	98,4%	99,7%	98,7%	96,5%	96,8%	97,6%	96,2%	95,6%	98,5%	95,6%	54,1%	30,2%	97,3%	91,8%	37,0%	31,4%
500	112,7	111,1	106,4	88,0	116,4	116,1	98,6	96,5	137,1	132,7	111,6	85,3	141,0	139,5	93,7	84,9
Sugeeffektivitet	97,0%	99,5%	99,3%	85,7%	96,0%	96,3%	82,3%	81,1%	95,1%	91,7%	76,8%	58,5%	96,5%	95,7%	64,4%	58,5%
600	149,8	149,6	149,1	145,8	144,1	142,5	141,6	111,5	142,8	129,2	117,6	116,2	142,8	147,2	141,1	140,4
Sugeeffektivitet	98,0%	97,3%	96,4%	93,8%	98,2%	96,8%	95,9%	75,3%	95,2%	85,3%	76,8%	75,2%	96,6%	98,1%	92,7%	91,0%
700	132,7	132,4	132,0	127,8	134,6	138,4	139,8	133,7	130,5	131,6	131,5	130,6	130,7	131,6	130,2	128,7
Sugeeffektivitet	99,3%	98,6%	97,9%	94,3%	99,0%	101,9%	103,0%	98,6%	96,9%	98,1%	98,5%	98,3%	98,9%	99,5%	98,4%	97,2%
800	119,7	119,1	122,3	125,0	123,9	125,2	126,3	125,8	122,2	121,1	123,8	127,4	122,4	123,8	123,7	121,1
Sugeeffektivitet	99,3%	97,3%	98,3%	99,0%	96,4%	97,6%	98,5%	98,3%	95,8%	94,6%	96,4%	98,8%	95,7%	96,9%	96,9%	94,9%
934	139,7	142,9	141,9	141,3	141,8	141,1	140,6	139,8	139,7	139,6	138,3	138,4	140,0	138,3	137,7	136,9
Sugeeffektivitet	97,7%	99,6%	98,5%	97,7%	97,8%	97,4%	97,2%	96,7%	96,9%	97,1%	96,5%	96,8%	98,9%	97,8%	97,4%	96,9%

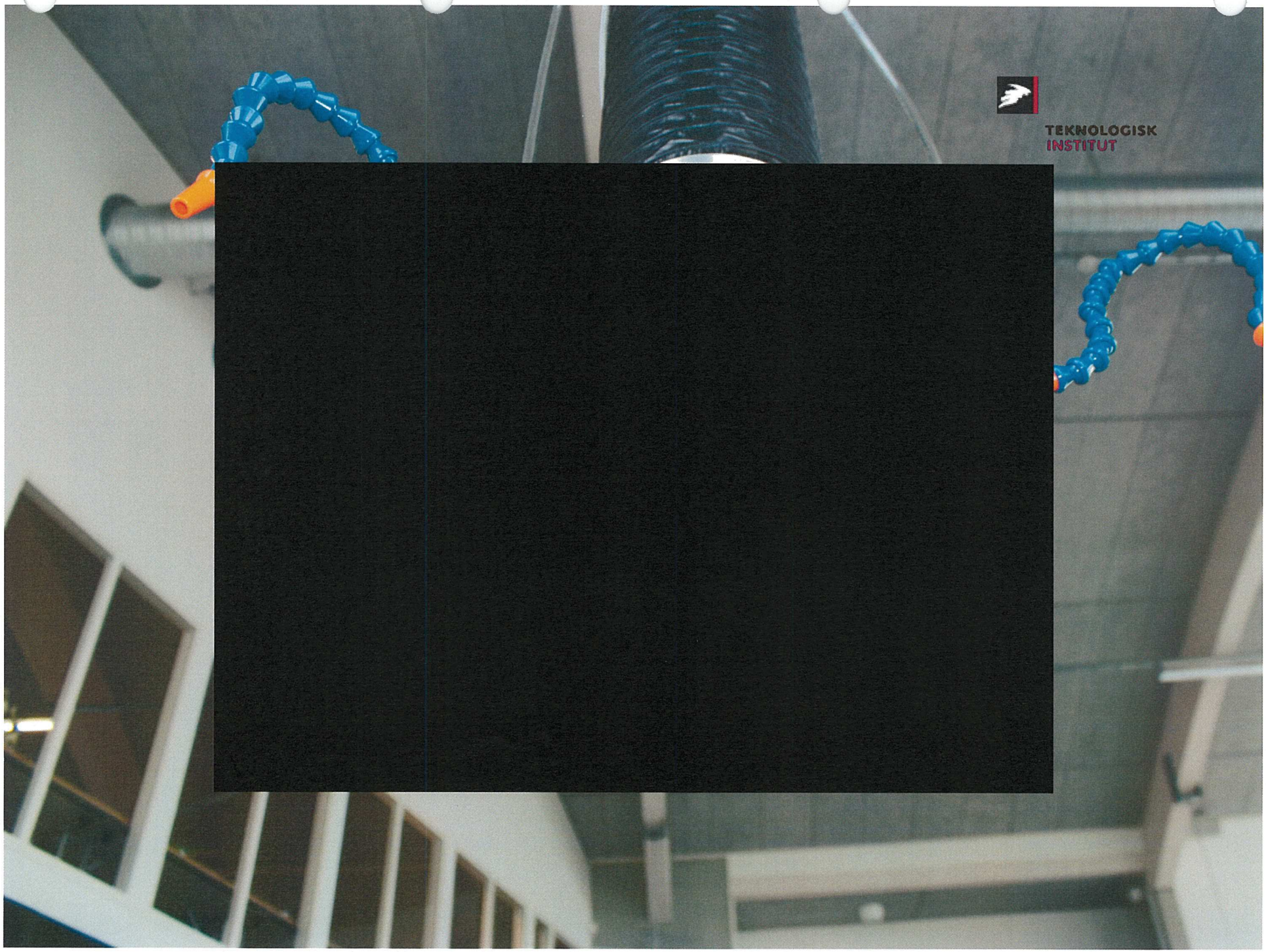

TEKNOLOGISK
INSTITUT

Idéen er, at gøre punktudsug bevægelige ved brug af automatik- og robotteknologi og med en intelligent styring og sensorteknik. Således bliver sugehoven bevægelig i sig selv og kan automatisk følge bearbejdningsarbejdet og forureningskilden.

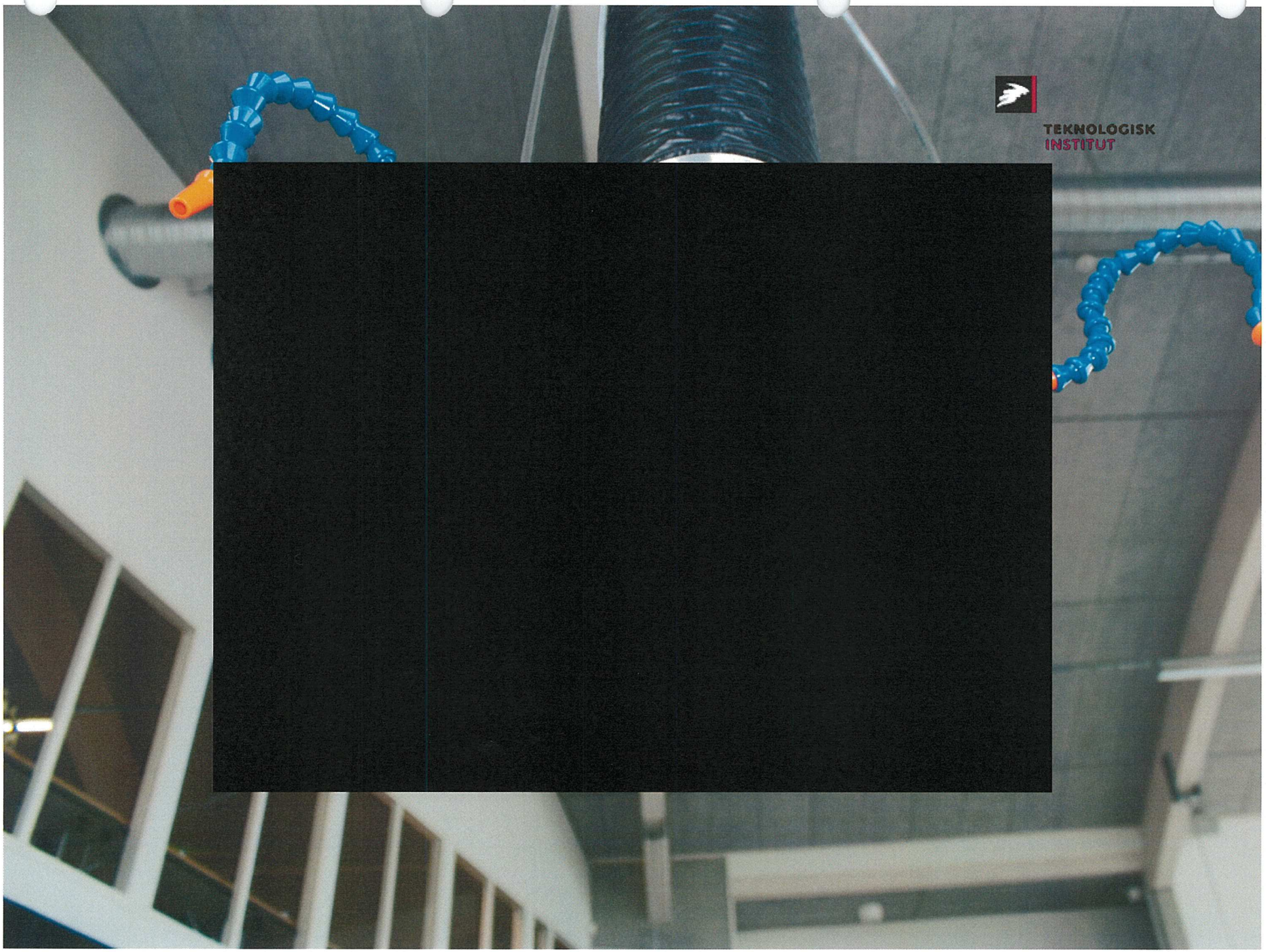

TEKNOLOGISK
INSTITUT
Signaler fra sensorer behandles af PLC styring der positionere sugearmen via aktuatorer.



Anvendelse af vision teknologi


TEKNOLOGISK
INSTITUT

Anvendelse af røgdetektorer


TEKNOLOGISK
INSTITUT

Projekt partnere

TEKNOLOGISK INSTITUT

Teknologisk Institut - Energieffektivisering og Ventilation samt Robotteknologi.
Geovent A/S – Fremstiller og sælger udsugningskomponenter
Mercantec – Underviser/uddanner i teknisk faglige uddannelser
Linak Danmark A/S – Udvikler, fremstiller og sælger lineære aktuatore.
Strøh Automation A/S – automation og industri-installationer
Daugaard Elektronik ApS – Intelligente visionkamera løsninger
SICK A/S – Industrielle sensorer, kamera og sikkerhedssystemer

EUDP
Energiteknologisk udvikling og demonstration

mercantec
karrierecenter midt

SICK
Sensor Intelligence.

DAUGAARD ELEKTRONIK

TEKNOLOGISK INSTITUT

STRØH
AUTOMATION

GEOVENT

LINAK®
WE IMPROVE YOUR LIFE

BILAG 19: PR – Ventilationstemadag. Teknologisk Institut 2016

Indeklimaets Temadag på TI 2016

27. september i Taastrup

Afholdes af FAV, Dansk Ventilation og Teknologisk Institut

2D Robot svejseudsug

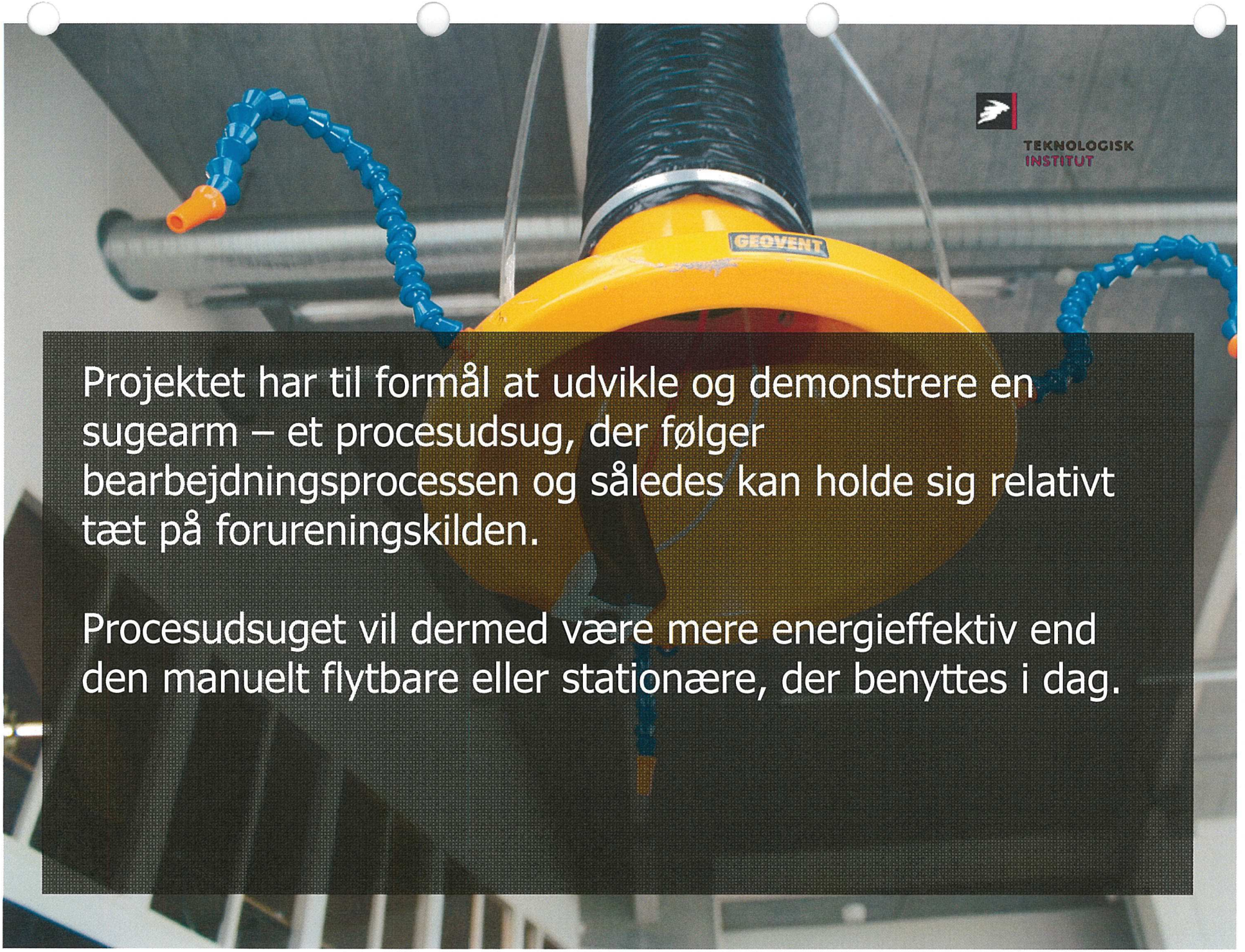
GEOVENT A/S, direktør Thomas Molsen, Viborg
Teknologisk Institut, senioringeniør Christian Drivsholm

TEKNOLOGISK INSTITUT

2D robot svejseudsug

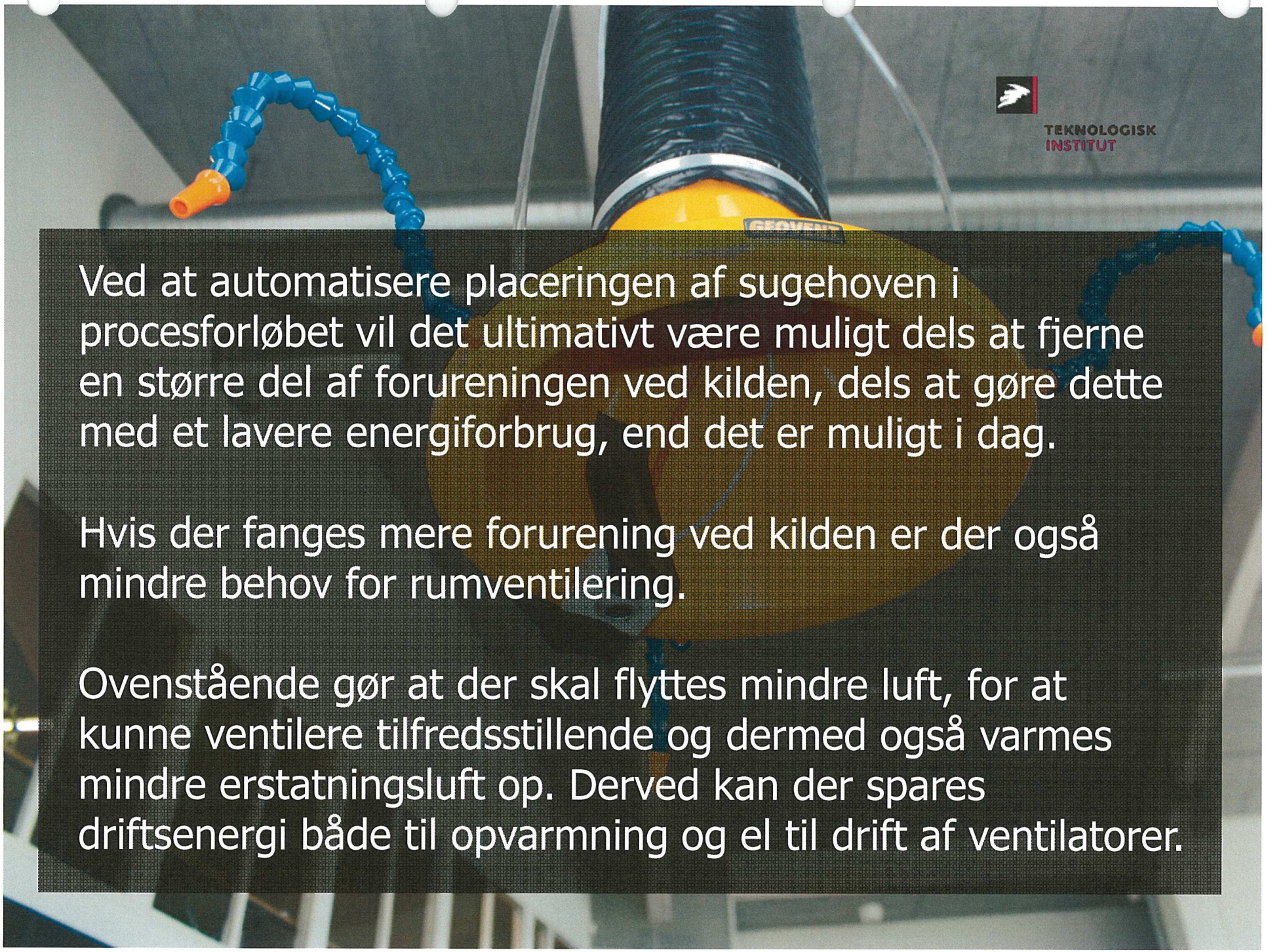
GEOVENT A/S, direktør Thomas Molsen, Viborg
Teknologisk Institut, Ventilation, senioringeniør Christian Drivsholm
Teknologisk Institut, Robotteknologi, ingeniør Karsten Thornø


EUDP
Energiteknologisk udvikling og demonstration
TEKNOLOGISK INSTITUT
Energieffektiv adaptiv procesudsugning
Projektet er støttet af:
"Energiteknologisk Udviklings- og Demonstrationsprogram (EUDP) "



Projektet har til formål at udvikle og demonstrere en sugearm – et procesudsug, der følger bearbejdningsprocessen og således kan holde sig relativt tæt på forureningskilden.

Procesudsuget vil dermed være mere energieffektiv end den manuelt flytbare eller stationære, der benyttes i dag.



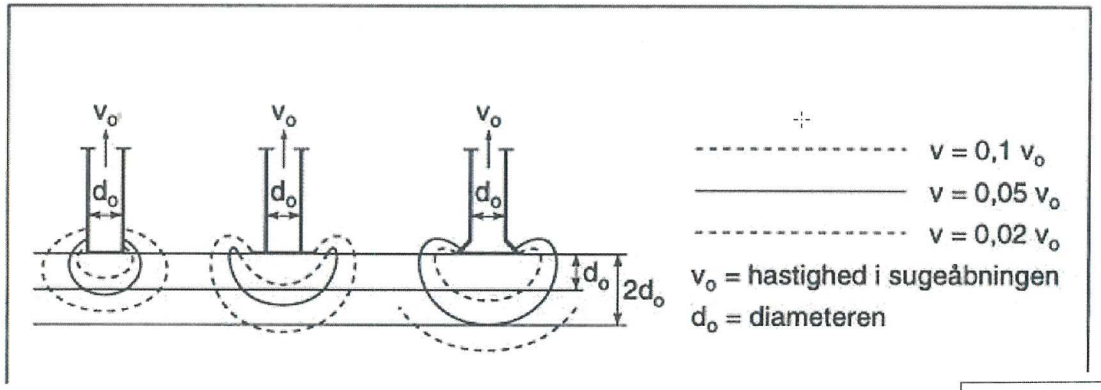
Ved at automatisere placeringen af sugehoven i procesforløbet vil det ultimativt være muligt dels at fjerne en større del af forureningen ved kilden, dels at gøre dette med et lavere energiforbrug, end det er muligt i dag.

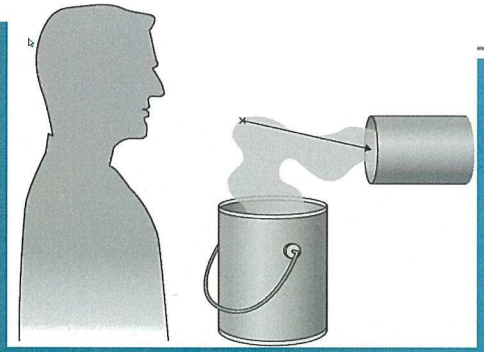
Hvis der fanges mere forurening ved kilden er der også mindre behov for rumventilering.

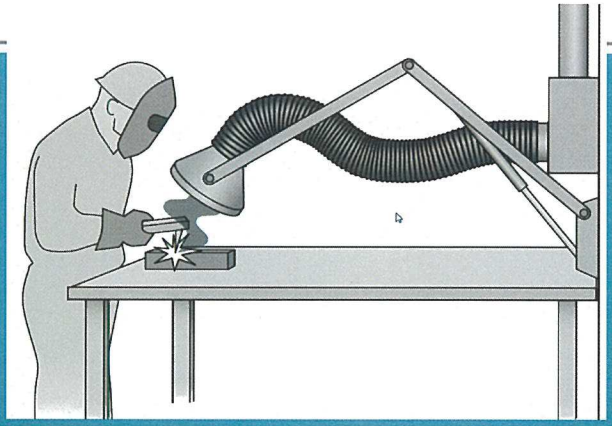
Ovenstående gør at der skal flyttes mindre luft, for at kunne ventilere tilfredsstillende og dermed også varmes mindre erstatningsluft op. Derved kan der spares driftsenergi både til opvarmning og el til drift af ventilatorer.

Lidt teori om procesudsug

TEKNOLOGISK INSTITUT

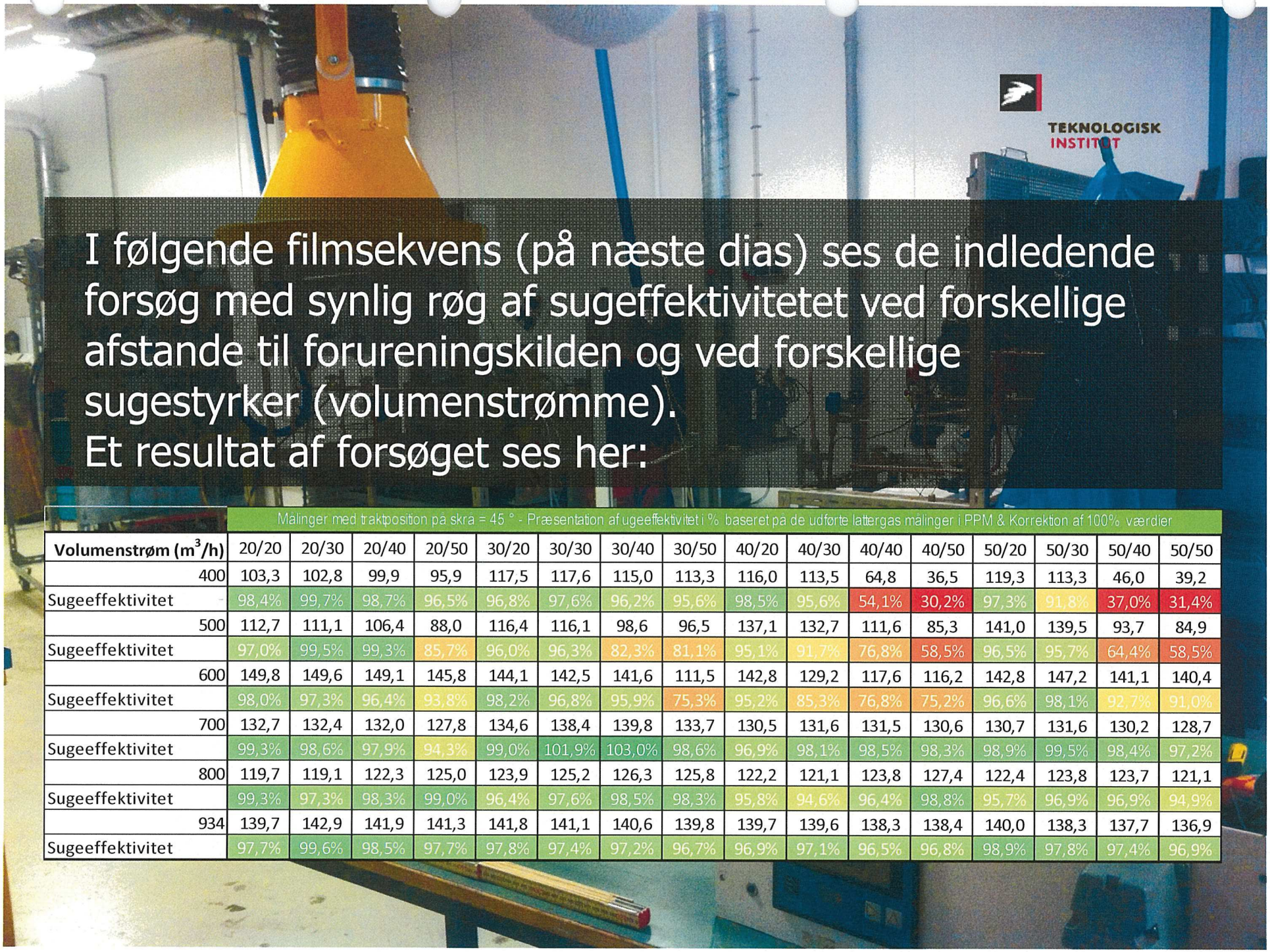






Arbejdsproces	Gribehastighed v_x [m/s]
Emhætter i køkkener og smedeesser	0,2-0,5
Sprøjtemaling, metalsprøjtning, pålæsning, fyldning af tønder og sække, svejsning	0,5-1,0
Tømning af støbeforme	1,0-2,0
Slibe- og polereskiver, sandslibning, sandblæsning	3,0-10,0
Hærdning	0,4
Bejdsning, kold varm	0,3 0,5
Affedtning	0,3
Galvanisering, krombad, cadmiumbad	0,4
Vandbad eller saltbad, ikke kogende kogende	0,2 0,4

Uden flange	Med flange
$\frac{V_o}{V_x} = \frac{10 \cdot x^2 + A}{A} \rightarrow q_v = V_x \cdot (10 \cdot x^2 + A)\ [\frac{m^3}{s}]$	$\frac{V_o}{V_x} = 0{,}75 \cdot \frac{10 \cdot x^2 + A}{A} \rightarrow q_v = 0{,}75 \cdot V_x \cdot (10 \cdot x^2 + A)\ [\frac{m^3}{s}]$



I følgende filmsekvens (på næste dias) ses de indledende forsøg med synlig røg af sugeffektivitetet ved forskellige afstande til forureningskilden og ved forskellige sugestyrker (volumenstrømme).
Et resultat af forsøget ses her:

	Målinger med traktposition på skrå = 45 ° - Præsentation af ugeeffektivitet i % baseret på de udførte lattergas målinger i PPM & Korrektion af 100% værdier															
Volumenstrøm (m^3/h)	20/20	20/30	20/40	20/50	30/20	30/30	30/40	30/50	40/20	40/30	40/40	40/50	50/20	50/30	50/40	50/50
400	103,3	102,8	99,9	95,9	117,5	117,6	115,0	113,3	116,0	113,5	64,8	36,5	119,3	113,3	46,0	39,2
Sugeeffektivitet	98,4%	99,7%	98,7%	96,5%	96,8%	97,6%	96,2%	95,6%	98,5%	95,6%	54,1%	30,2%	97,3%	91,8%	37,0%	31,4%
500	112,7	111,1	106,4	88,0	116,4	116,1	98,6	96,5	137,1	132,7	111,6	85,3	141,0	139,5	93,7	84,9
Sugeeffektivitet	97,0%	99,5%	99,3%	85,7%	96,0%	96,3%	82,3%	81,1%	95,1%	91,7%	76,8%	58,5%	96,5%	95,7%	64,4%	58,5%
600	149,8	149,6	149,1	145,8	144,1	142,5	141,6	111,5	142,8	129,2	117,6	116,2	142,8	147,2	141,1	140,4
Sugeeffektivitet	98,0%	97,3%	96,4%	93,8%	98,2%	96,8%	95,9%	75,3%	95,2%	85,3%	76,8%	75,2%	96,6%	98,1%	92,7%	91,0%
700	132,7	132,4	132,0	127,8	134,6	138,4	139,8	133,7	130,5	131,6	131,5	130,6	130,7	131,6	130,2	128,7
Sugeeffektivitet	99,3%	98,6%	97,9%	94,3%	99,0%	101,9%	103,0%	98,6%	96,9%	98,1%	98,5%	98,3%	98,9%	99,5%	98,4%	97,2%
800	119,7	119,1	122,3	125,0	123,9	125,2	126,3	125,8	122,2	121,1	123,8	127,4	122,4	123,8	123,7	121,1
Sugeeffektivitet	99,3%	97,3%	98,3%	99,0%	96,4%	97,6%	98,5%	98,3%	95,8%	94,6%	96,4%	98,8%	95,7%	96,9%	96,9%	94,9%
934	139,7	142,9	141,9	141,3	141,8	141,1	140,6	139,8	139,7	139,6	138,3	138,4	140,0	138,3	137,7	136,9
Sugeeffektivitet	97,7%	99,6%	98,5%	97,7%	97,8%	97,4%	97,2%	96,7%	96,9%	97,1%	96,5%	96,8%	98,9%	97,8%	97,4%	96,9%