

Sensorer

1. februar 2023 14:25

Sensorer simulerer elektronikkomponenter der leverer input fra: Temperaturfølere, gasmålere, lysfølsomme modstande, PIR, relækontakter som eksempler. Biblioteket har grænseflade til applikation og kun få generelt brugbare softwarekomponenter. Applikationer skal bruge biblioteket til grænseflade og koble egne specifikke softwarekomponenter på det.

Bibliotek hedder: JBSensor.h.

Funktion

En sensor modtager input fra en port i en input-driver. Porten bliver overvåget for hver klokkecyklus.

En sensor konfigureres til den aktuelle anvendelse. For eksempel:

- Fra en temperaturføler bliver input fra port leveret videre uændret til sensorens udgang.
- Fra en driver der giver en impuls holdes sensorens udgang fast indtil ny impuls kommer eller indtil reset.

En sensor har som funktion.

- Sensor får tilknyttet en driver og en port ved opsætning.
- En port leverer enten høj eller lav, når den er aktiv.
- En sensor kan kobles til en eller flere digitale funktioner.
- En sensor modtager polling. I et gennemløb handles der på portens tilstand.
- En sensor sender værdi fra port videre til digitale funktioner.
- En sensor sender reset videre til digitale funktioner.
- En sensor leverer sin status.
- I software er der brugt navngivning: Udgangens tilstand: ON eller OFF.

Opsætning

I hovedprogram skal der erklæres globalt:

- MaxNoSensors: unsigned int const=nr: Antal sensorer i applikationen. Navn på konstant er obligatorisk.
- Enum-liste med sensorernes navne.

I hovedprogram setup() skal der opsættes:

- <navn på sensor>.begin(...)

Grænseflade

Grænsefladen giver en ensartet tilkobling af sensor.

For grænsefladens design se softwarepattern facade.

Klasse

Klassens navn: t_Sensor

Medlemmer

Grænsefladens medlemmer.

Navn	Type	Egenskab	Beskrivelse
driver	*InputDriver		Pointer til input drivere
portNo	unsigned int		Input portens nummer
state	byte		Sensorens tilstand.

Metoder

Grænsefladens metoder.

Navn	Argumentliste	Returnerer	Beskrivelse
Constructor			Sætter state default=OFF
begin	(driver: *InputDriver, portNo: unsigned int)		Sensor initialiseres.
doClockCycle	()		Overvåger port, for at holde digitale funktioner opdateret.
status	()	byte	Udlæser resultat af sensor.
reset	()		Resetter digitale funktioner og genlæser input port.

Sensor simpel

1. februar 2023 14:25

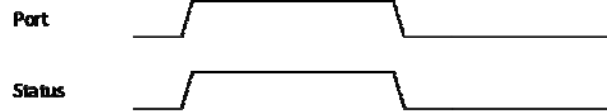
Simpel sensor simulerer de oftest brugte funktioner ved brug af temperaturfølere, relækontakter m.fl.

Funktion

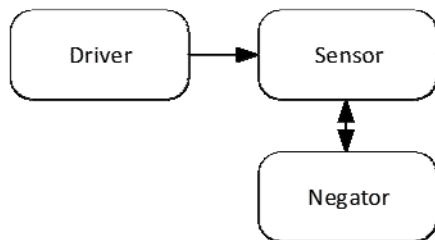
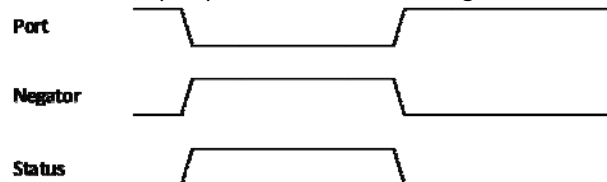
En sensor kan kobles til digitale funktioner, der giver fleksibilitet med at tilpasse funktionen af sensoren.

Eksempler:

Sensor signal overføres uændret.

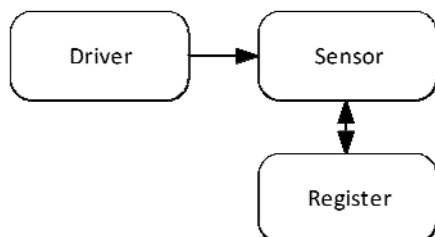
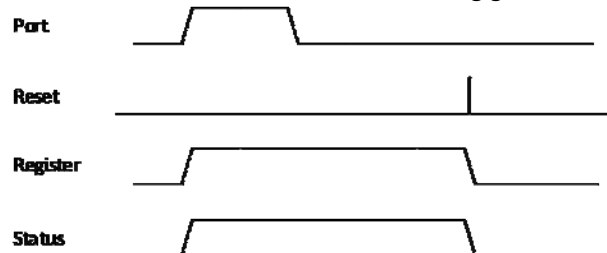


Sensor hvor input-port normalt er HIGH og hvor LOW er farligste stilling.



Negation konverterer det binære signal til positiv logik.

Sensor med huskefunktion, så en aktivering gemmes til senere.



Register gemmer sensorens stilling, indtil applikation sender et reset.

1 digital funktion kan kobles til en sensor. Det er valgfrit om den bruges.

Sensor understøtter ikke mere avancerede funktioner. Opstår den slags behov, skal der programmeres en ny softwarekomponent.

Opsætning og brug i komponent

En komponent, der skal bruge en sensor skal opsætte:

- Pointer til <navn på sensor>.

Klasse

Klassens navn: t_SimpleSensor

Medlemmer

Sensorens medlemmer.

Navn	Type	Egenskab	Beskrivelse
digitalFunction	*DigitalFunction		Pointer til en digital funktion

Metoder

Sensorens metoder.

Navn	Argumentliste	Returnerer	Beskrivelse
setDigitalFunktion	(digitalFunction: *DigitalFunction)		Kobler digitalfunktion sensor.
doClockCycle	()		Input driver bliver aflæst i hver klokkescyklus. Digitale funktioner bliver opdateret.
reset	()		Resetter digital funktion. Status bliver opdateret i næste klokkescyklus.