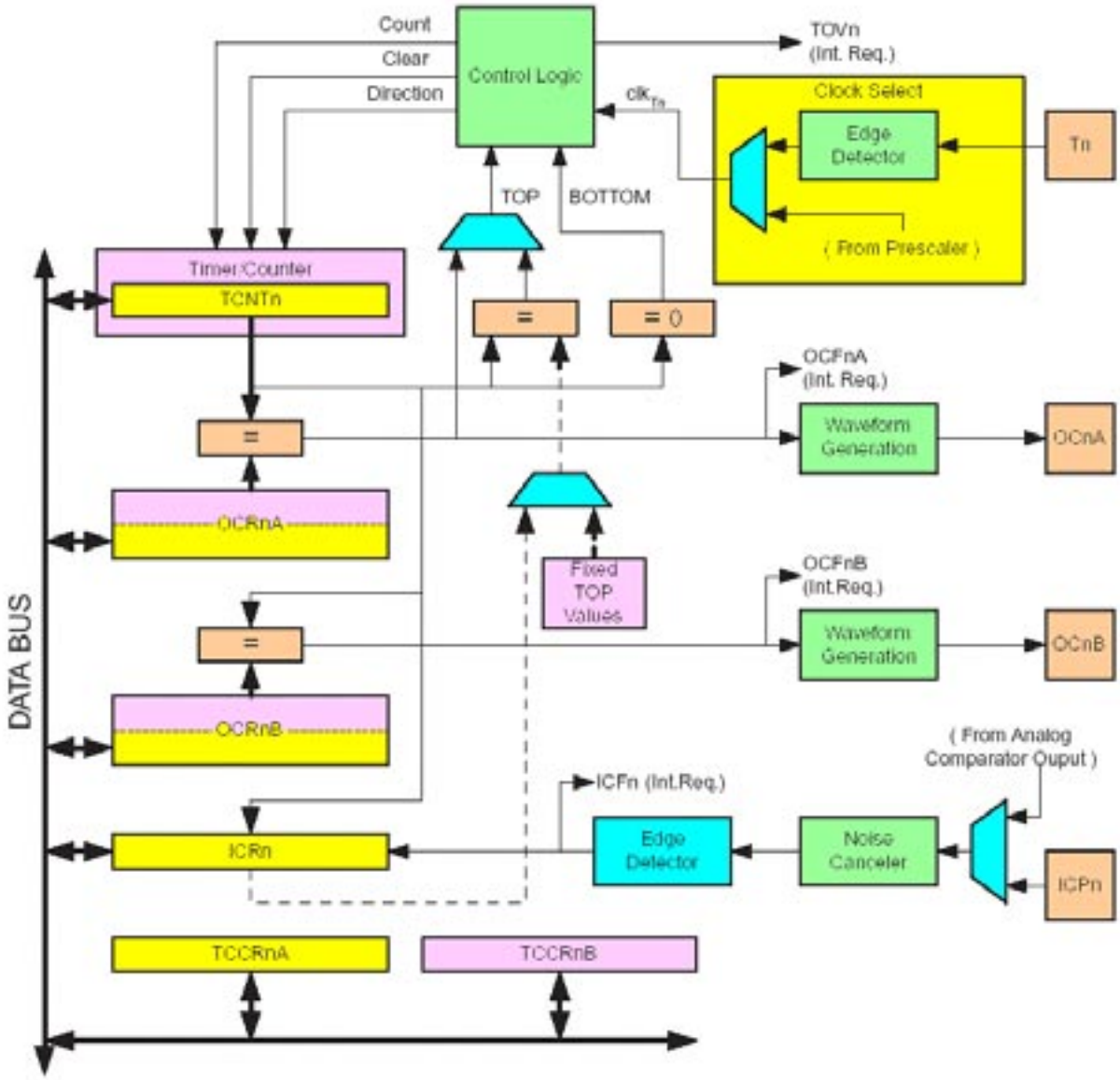
T I M E R and C O U N T E R (1)

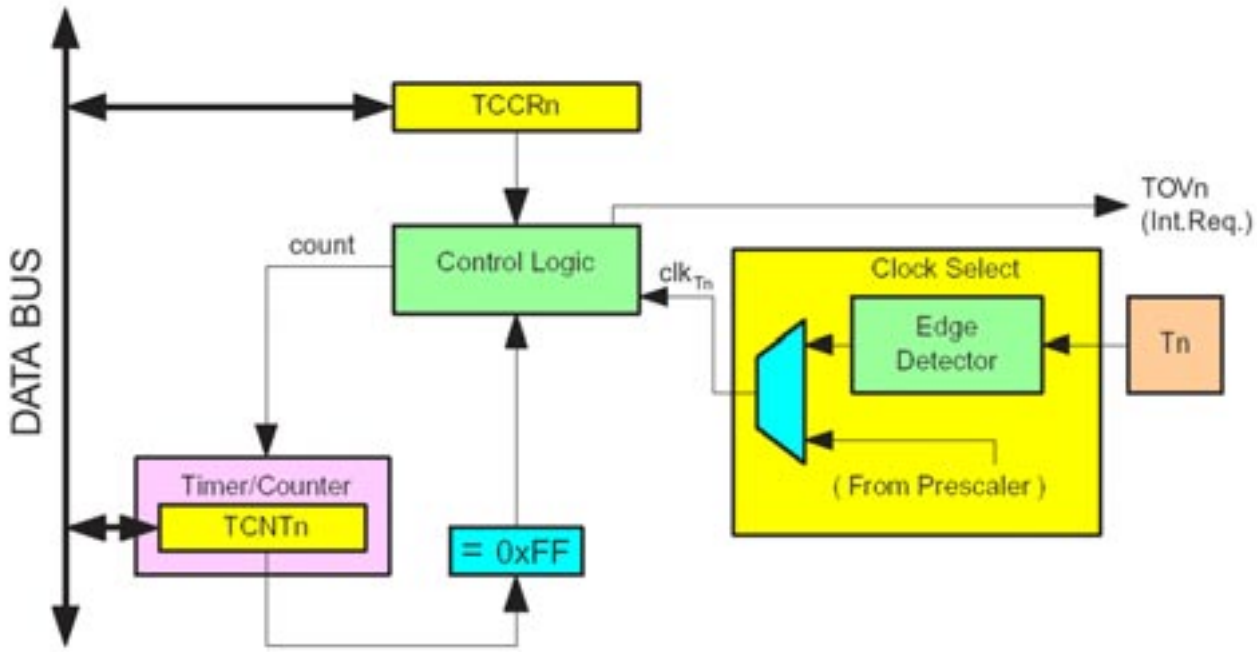
The internal structure of **Mini Modules** is composed by many different sections that are capable to supply a wide variety of **Hardware** devices capable to support different applications.

In following figures it is possible to recognize the typical Timer/Counter internal sections of controller.

The figures shows two of the most frequently used settings modes:



Block diagram of Timer/Counter section



Block diagram of 8 bit Timer/Counter section

A really wide diffused sensor, that is often used into industrial applications, is the nearness sensor commonly named Proximity. There are many types of this sensor as the inductive or the capacitive ones but basically, if we consider the simplest model, they are devices that acts as switches when the required action occurs and they open or close an electric contact.

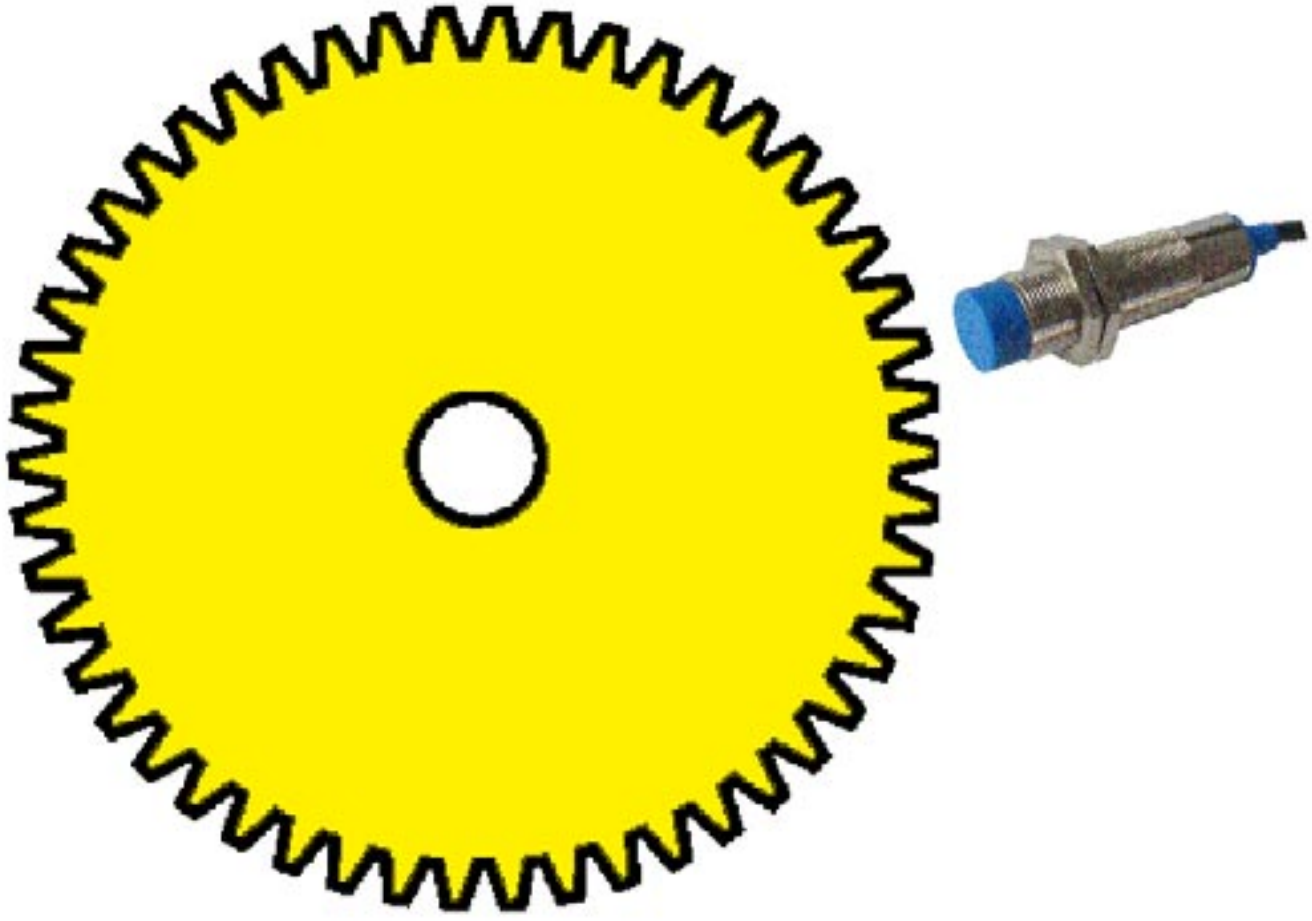
Typical nearness sensor commonly named Proximity

In this chapter we don't describe in details the many models available on the market (please believe they are really numerous) but we consider only one specific type that is used in the **Phonic Wheel**.

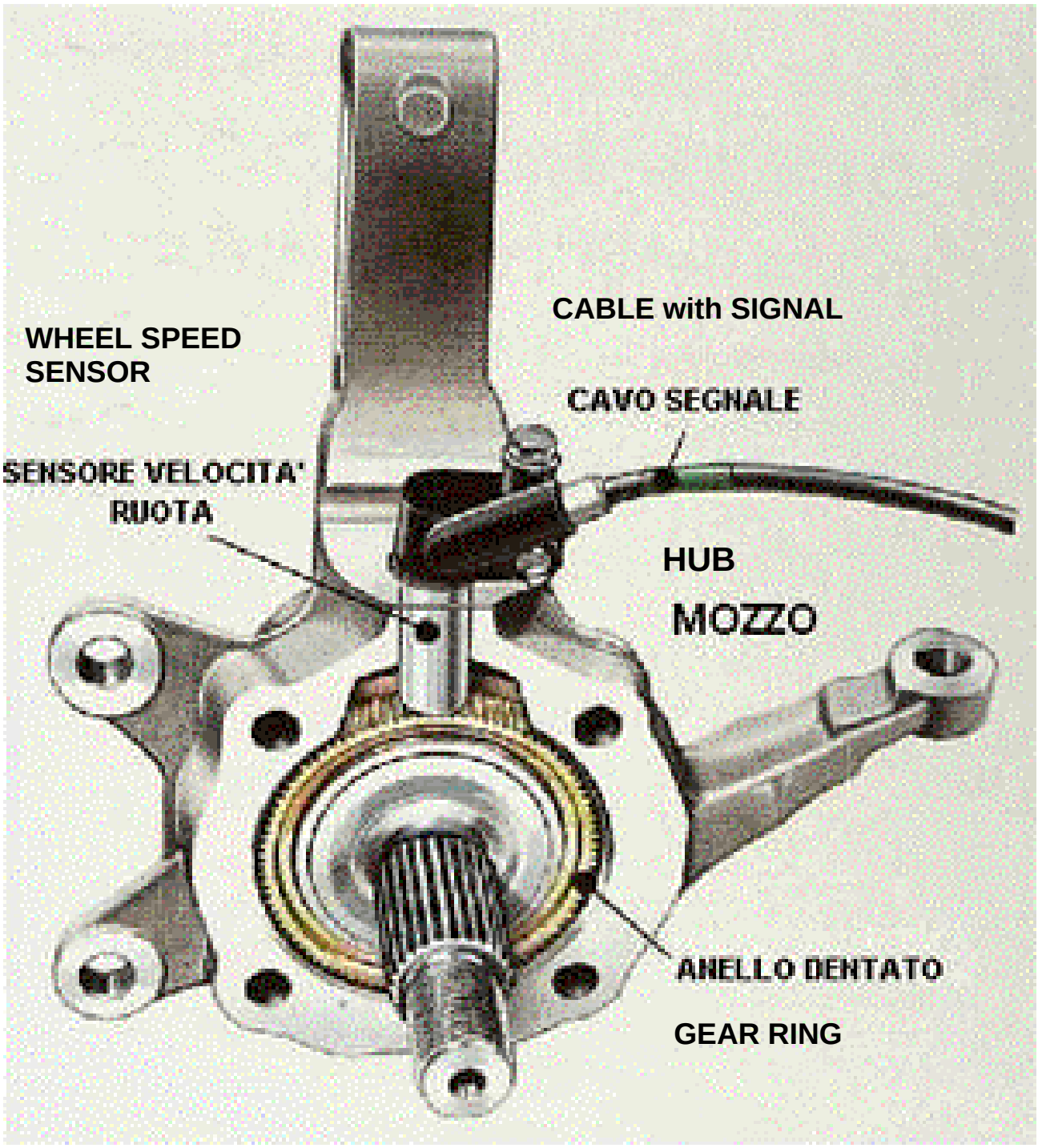
This device is composed by a wheel with teeths (similar to a gear), connected for example to a transmission shaft of a car, where each teeth excites a proximity sensor.

When the shaft rotates each teeth generates an electric pulse. By using this signal you obtain information on the movement, for example it can drive the speed meter of the car to show its velocity.

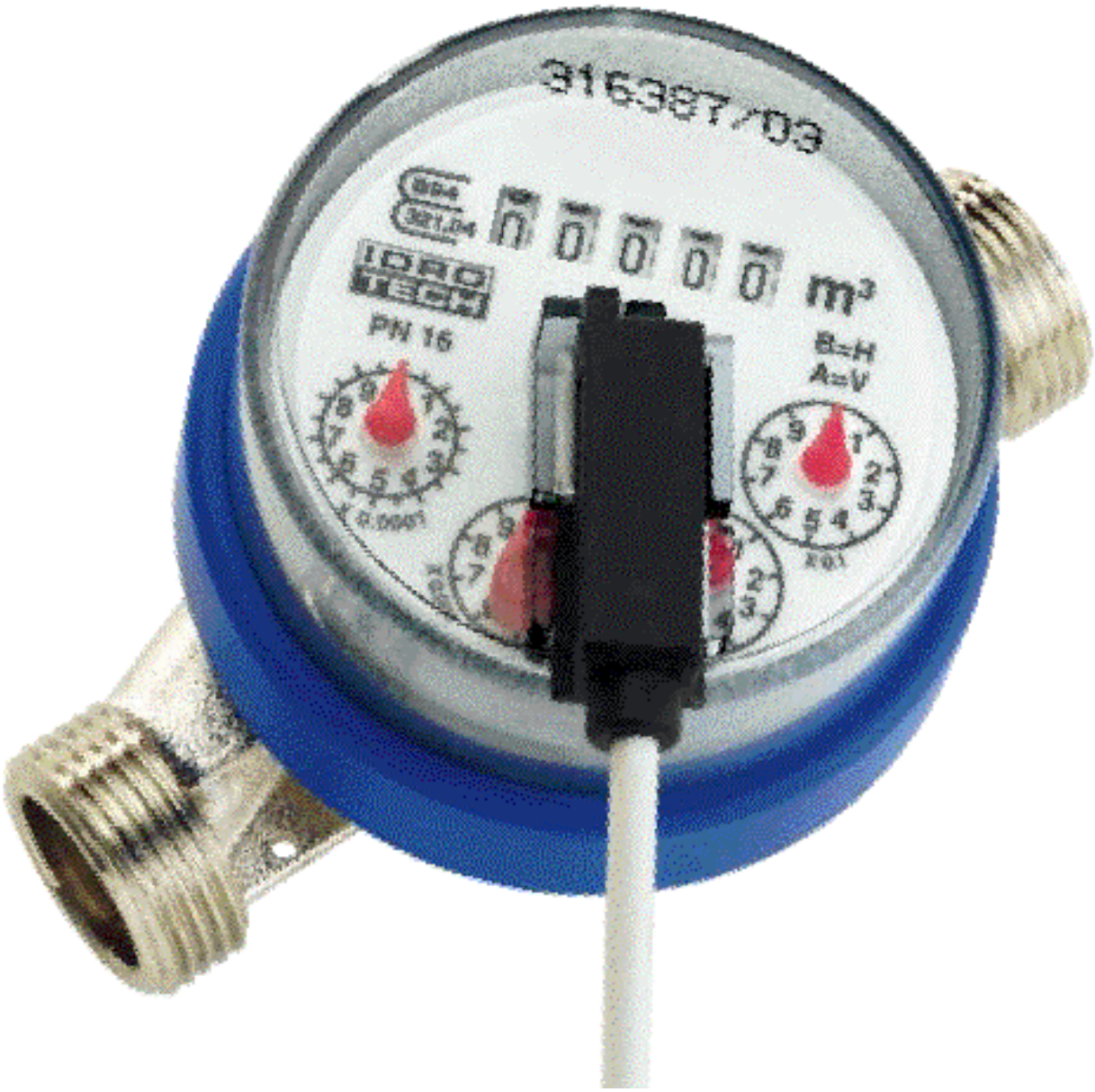
Another application of this device is those, certainly more complex, joined with **ABS** control in the car's brakes.



Draft diagram of a Phonic Wheel



Real application of a Phonic Wheel



Electro-Mechanical counter for fluid

There is another application to report, that are the **Electric-Mechanical Counters** suitable for the measure of liquids as, for example, the water consuption.
The device shown in previous figure is a classic example. The water quantity flowed through the meter is displayed by a series of mechanical indicators.

Moreover, a proper sensor supplies an impulsive signal for each unit of liquid, through a clean contact of a Reed Cruet. By measuring this signal it is possible, even at a long distance, to obtain either the istantaneous and the timed flows and, obviously, the total amount of passed liquid.

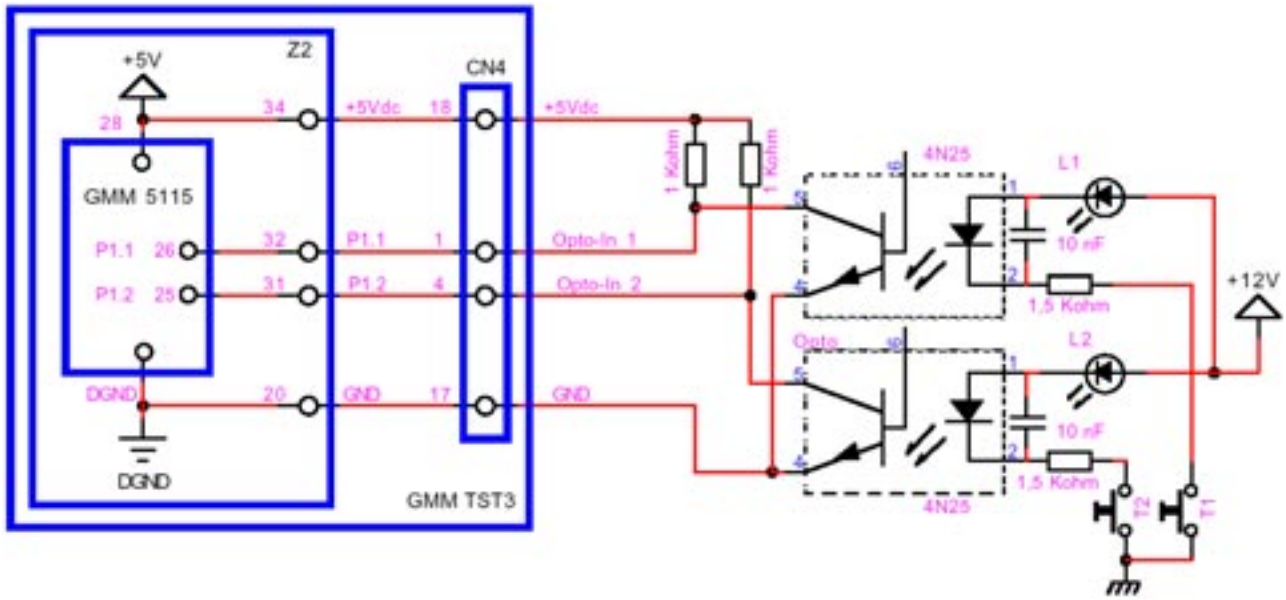
Example.075. 32 bits Counter from an Optocoupled Inputs

Added Definitions:
None

Added Declarations:
None

Added Instructions:
None

Added Operators:
None



Application electric diagram for NPN type Optocoupled Inputs

Example program 075 of **BASCOM AVR** course.

32 bits Transictions **Counter** from an optocoupled input.

The program counts the incoming transictions of the optocoupled inputs, connected to **TIMER1**. The number of managed counts is not limited to maximum **16** bits of **TIMER1** but it is doubled up to **32** bits, equal to **4.294.967.296** transictions. The program describes the modality used to increase the counter resolution and the user can adapt it according with his requirements.

The transictions number is continuosly displayed on serial console. By changing the status of **Opto-In 6** input, connected to **CN4** connector of **GMM TST3** as described in electric diagram, the **T1** signal is modified and counted as it is the counter input signal. The **overflows** of the counter are recognized by proper **interrupt** routine and then counted too, in order to increase the counts resolution.

The typical usage of this program are for example, counting of the pulses generated by a phonic wheel, a generic pieces counter, manage the position of a **monodirectional encoder**, etc.

The program describes its functionalities and uses a serial console provided of monitor and keyboard with a fixed physical protocol at **19200 Baud, 8 Bits x chr, 1 Stop bit, No parity**.

This console can be another system capable to support a serial **RS 232** communication. In order to simplify the use it can be used a **PC** provided of one **COMx** line, that execute a terminal emulation program as **HYPERTERMINAL** or the homonym modality provided by **BASCOM AVR** (see **IDE** Configuration).

The program works only when the **GMM AM08** is mounted on **Z2** socket of **GMM TST3**!!

In the program source the names that identifies the used signals refers to electric diagram and technical manual of **GMM TST3**!!

Example.076. Manage 4 Lines of 32 bits counters from Optocouled Inputs.

Added Definitions:
None

Added Declarations:
None

Added Instructions:
CONFIG TIMER1; ON OVF1; ENABLE OVF1; START COUNTER1; STOP COUNTER1

Added Operators:
None

Example program **076** of **BASCOM AVR** course.

Four **32** bits Counters from optocoupled inputs.

The program counts the incoming transictions of all the optocoupled inputs, connected to dedicated hardware peripheral devices (**TIMER** and **interupt**).

The counts are always managed with **32** bits, equal to **4.294.967.296** transictions, in order to obtain an high resolution on the maximum number of available channels.

The transictions numbers are continuosly displayed on serial console. By changing the status of **Opto-In 3,4,5,6** inputs, connected to **CN4** connector of **GMM TST3** as described in electric diagram, the proper counters are increased.

The typical usage of this program are for example, counting of the pulses generated by phonic wheels, by generic pieces counters, manage the position of some **monodirectional encoders**, etc.

The program describes its functionalities and uses a serial console provided of monitor and keyboard with a fixed physical protocol at **19200 Baud, 8 Bits x chr, 1 Stop bit, No parity**.

This console can be another system capable to support a serial **RS 232** communication. In order to simplify the use it can be used a **PC** provided of one **COMx** line, that execute a terminal emulation program as **HYPERTERMINAL** or the homonym modality provided by **BASCOM AVR** (see **IDE** Configuration).

The program works only when the **GMM AM08** is mounted on **Z2** socket of **GMM TST3**!!

In the program source the names that identifies the used signals refers to electric diagram and technical manual of **GMM TST3**!!