

## Rev Counters

Gunther Zielosko

### 1. Introduction

Anyone who likes to experiment with motors or other machines sometimes would like to measure the revs per minute of a rotating part. Especially model aircraft or car builders, meteorologists, car fans and developers of all kind would appreciate a device which exactly measures the current revs per minute and in addition can specify maximum and average revs per minute as well as its development over a certain period of time. There are rev counters for a lot of specific applications; a versatile device has still to be constructed. This is a perfect field of application for the BASIC-Tiger®, which can record revs per minute with little hardware. Adaptation to the respective problem is done by the software (figure 1).



*Fig. 1 The BASIC-Tiger® measures the revs per minute of several items using little hardware and software.*

Let's start listing the requirements. What number of revs per minute is supposed to be captured? Mechanical parts in the DIY area can reach up to 40,000 r.p.m., such as miniature tools (e.g. Drehmel Multi). 40,000 r.p.m. would be 667 revs per second, i.e. 667 Hz. In addition we would like to measure low revs per minute, e.g. the revolutions of a clock minute hand.

For exact time measurement you need a time base. At the BASIC-Tiger® you can use the ticks of the internal clock as a time base (1 ms) and you can use special device drivers for frequency and period measurement (FREQ1\_Pp.TDD, PLSIN1.TDD, PLSI2\_Pp.TDD with a time base below 1  $\mu$ s). For comparatively high revs per minute the latter is more suitable, for a really versatile device the "tick" time base of 1 ms is more suitable, despite some problems. What problems are there? First of all the maximum revs per minute (667 revs per second) means, that one revolution takes about 1.5 milliseconds. Measurements would be extremely imprecise if the time raster was 1 ms. Deviations can be enormous depending on how much ticks go by during one revolution (1 or 2). Fortunately a BASIC-Tiger® can calculate so that such a task is easy to solve – we will not measure how long one revolution takes but e.g. how long 10, 100, or 1,000 revolutions take. The Tiger calculates the revs per minute from this. With very low revs per minute it is better to measure how long one revolution takes, because 1,000 revolutions would take ages. So the software is supposed to decide itself, how the revs per minute are measured best. Depending on the revs per minute it chooses the most suitable method. Now it becomes obvious, why we chose the "ticks" as a time base and not the time bases of the device drivers mentioned above. The device drivers are much faster and their time base lies in the  $\mu$ s region, but its counting range is limited. This complicates measurements over a wide range of revs per minute. The counting range of the tick counter, however, is comparatively huge. The disadvantage of the worse resolution is solved by the BASIC-Tiger® intelligence. Capturing the revolution itself is another important point. Measurements at different objects should be made as simply as possible, i.e. at plane propellers, motor shafts of small and big motors, at wind wheels etc. Also different materials will occur, such as metallic and non-metallic, magnetic etc. Many objects have several segments, such as a plane propeller. So we have to come up with a suitable sensor which has to be adapted to the BASIC-Tiger® as well as a possibility to enter the number of segments of the item to be tested beforehand.

## **2. The sensor: Capturing the r.p.m. at the object**

The best electronic rev counter is only useful, if you convert the mechanical value r.p.m. to an electronic value. So we need impulses, the distance between which characterises the revs per minute. This is simple: We will construct a light barrier, where an impeller wheel interrupts the light beam in time with the revolutions. In the case of transparent parts you can tape the according part with non-transparent tape. The photo receiver could be a photo diode or a photo transistor, the transmitter could be a lamp or a LED (infrared is favourable). When constructing please consider that the light barrier is well screened against external light. This is best done by putting photo receiver and transmitter each into one end of a tube so that they are aligned well to each other. The whole thing is applied to a gable shaped board (e.g. by

soldering or gluing) and then slotted twice in the middle, so that an accordingly wide slot for the impeller wheel is created (figure 2).

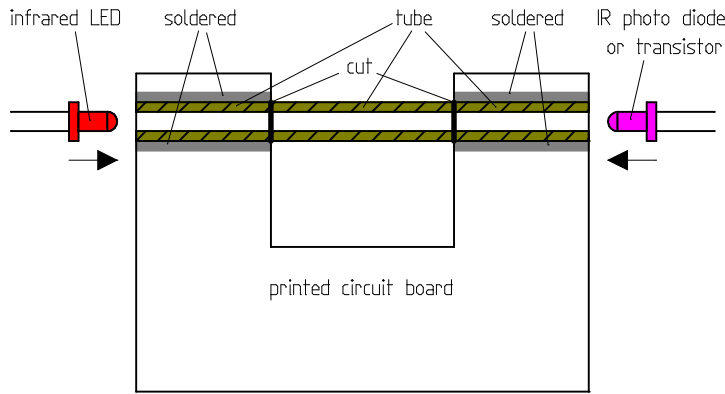


Fig. 2 Mechanical construction of the light barrier

### 3. The Circuitry

The circuitry effort required in addition to the BASIC-Tiger<sup>®</sup> is limited (figure 3). In order to have a clear switching at the transition from bright to dark and vice versa, we will use a gate of a CD4093 (four fold NAND with Schmitt trigger inputs, CMOS technology). The data sheet is available on: <http://www.ti.com/sc/docs/products/logic/cd4093b.html>

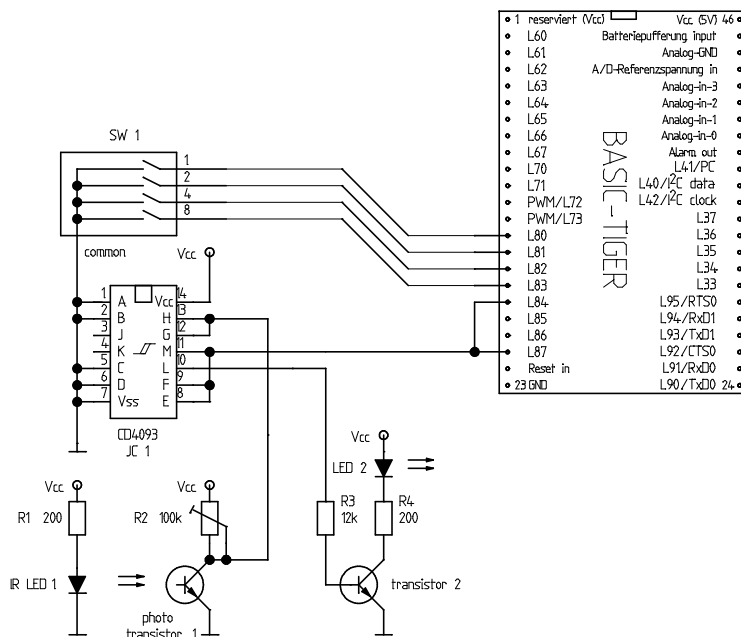
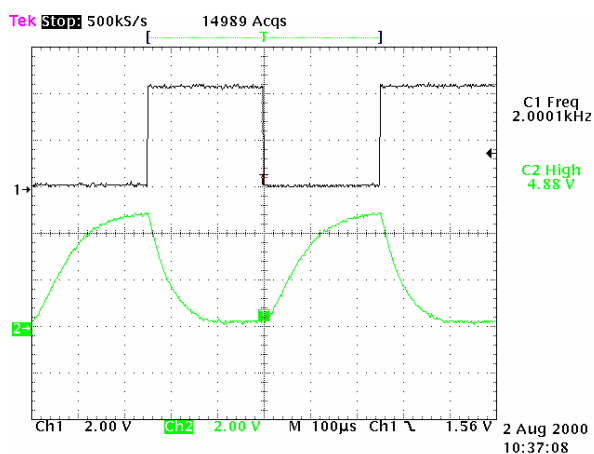


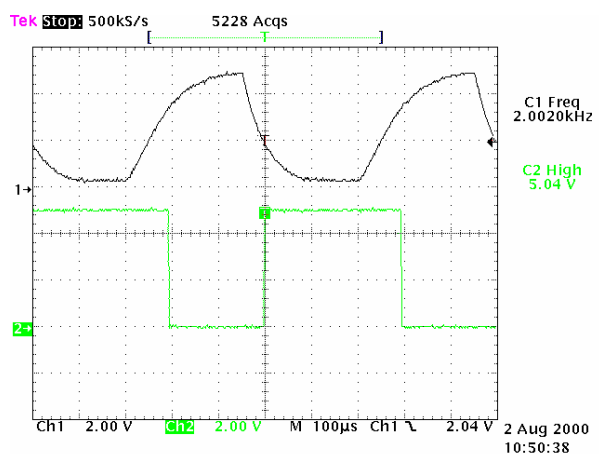
Fig. 3 Light barrier circuitry

At the output M the signal for the rev counter can be captured. With a further gate and a transistor an optical signal is created via LED 2, which can be used for testing the correct function of the light barrier even without a BASIC-Tiger<sup>®</sup>. In addition to the light barrier components only the decode switch for selecting the number of the impeller wheel segments is connected.

Depending on the light barrier components (LED1, photo transistor) the circuitry has to be customised so that at the CD 4093 inputs H and G (12, 13) reliably detect low when the light beam is undisturbed and high when the light beam is interrupted, even at frequencies around 2 kHz! LED 2 displays the light barrier's function at least at low revs per minute. The adjustment can be done with R2 at first; the less resistant R2, the faster the photo transistor switches (reference value 10...50k $\Omega$ ). The power of LED1 can be varied by R1 (mind the LED's power dissipation!). Typical characteristics of the signal processing are shown in figures 4 and 5.



*Fig. 4 Characteristic 1, clock signal of IR LED (top), signal at collector of photo transistor respectively input of CD4093 (bottom).*



*Fig. 5 Characteristic 2, input signal at CD4093 (top), output signal of CD4093 (bottom) – clear switching edges because of Schmitt-Trigger inputs*

#### 4. Software

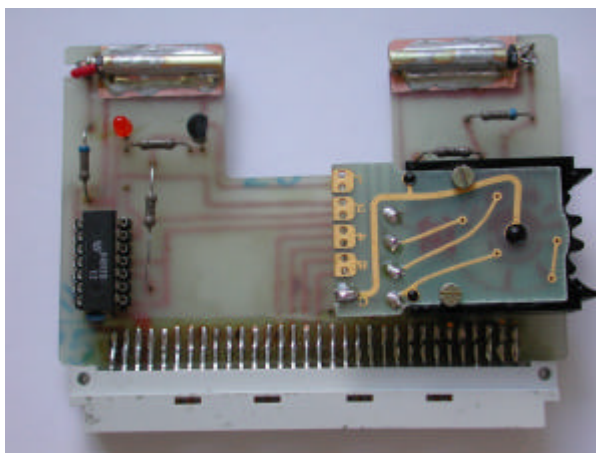
The program REVS\_01.TIG is a convenient measurement program for capturing revolutions per minute in extreme ranges. Basically there is virtually no lower limit, the display is only formally limited to 3 decimal places, which is a resolution of a thousandth part r.p.m.! It is no problem to measure the revs per minute of a clock's minute or second hand, if you have enough time. The effective range's upper limit is determined by the reaction of the BASIC-Tigers digital input and the according BASIC query. 100,000 impulses per minute are the limit. You have already noticed that we now speak of impulses per minute. We have to consider that some applications answer with several impulses per rotation, e.g. a plane propeller with three blades. If the shaft turns at 10,000 r.p.m., the light barrier as a sensor

already receives 30,000 impulses per minute. So it makes sense to set the rev counter's effective range for mechanical objects this high.

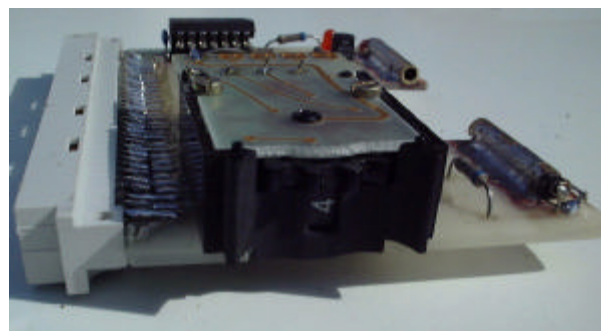
The input impulses are interpreted at the pin L87. The software bides a complete cycle consisting of a high and a low phase of the sensor circuitry, in order to know whether more than one cycle is required for increasing the exactness. If necessary up to 1,000 revolutions are measured. The program uses the best values for processing.

The pins L80 to L83 are used for inputting the number of segments of the part to be measured. Here you can use a binary coded decode switch, which allows comfortably entering up to 15 segments. A special feature is the quite unusual input and calculating method of these variables. Such decode switches are mostly used for setting binary coded numbers at inputs normally set to 0. Therefore the shared pole is set to plus. In the case of the BASIC-Tiger the inputs already read logic 1, even if there is nothing connected at all. Therefore we reverse the logic's polarity, so that the shared pole now is ground and the chosen bits are applied to ground. But after this the numbers on the switch do not match the read binary code any more. The program REVS\_01.TIG uses a quite unusual conversion of numbers (first negating everything, then choosing the lowest 4 bits) to set things right. So if the decode switch displays 5, the number calculated eventually is going to be 5.

For the experienced developer with BASIC-Tiger skills it should not be a problem to adapt the basic component (rev counter) to a specific task hardware and software-sided. Since the light barrier output is additionally applied to pin L84, the faster driver PLSIN1.TDD or else can be used at once. It is also no problem to set up the software for capturing maximum and minimum r.p.m. The average r.p.m. can be calculated over a certain period of time, also the total number of revolutions etc. If desired you can display the temporal progression on the graphical display or e.g. build a convenient wind gauge. The latter could be placed on the roof and transmit the r.p.m. to the ground through the serial port. Or what about a clock frequency display which is also measured in (im)pulses per minute? Besides a totally different usage of the ready-built light barrier and its signal processing is possible, so dealing with this topic is worth it! Figures 6 and 7 show the modular construction of a model rev counter.



*Fig. 6 Light barrier module*



*Fig. 7 Details, decode switch and the cut tube with the inserted photo modules*