

Motochassis
by
Tony Foale

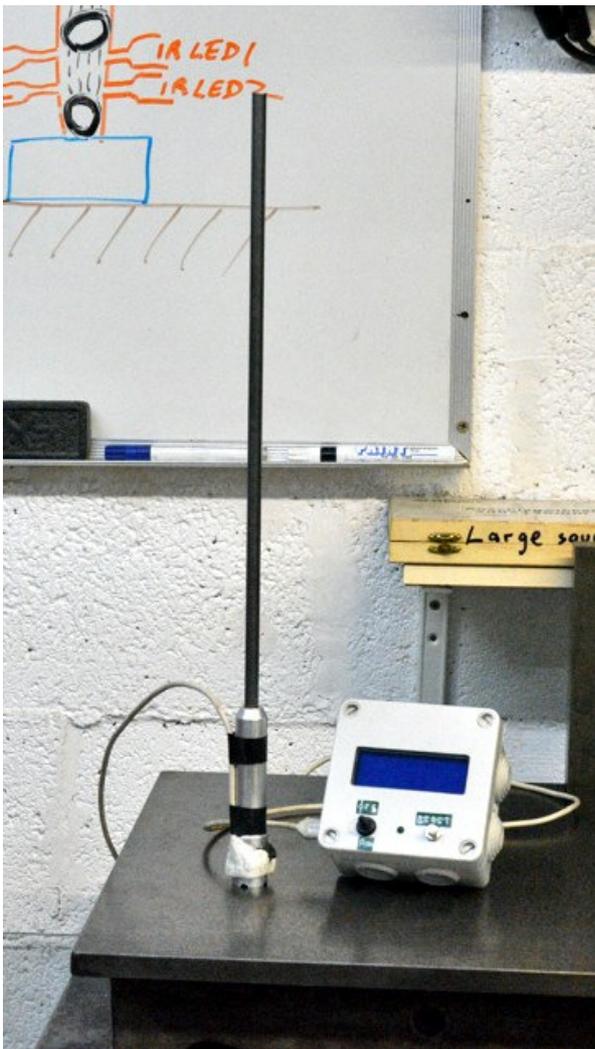
motochassis.com

My Hardness Tester for Metal

© Tony Foale - February 2021

Introduction

This rather rambling document describes my experience building a couple of rebound hardness testers as well as some general discussion on the subject of hardness testing.



I wanted to measure the hardness, at least on a comparative basis, of welded motorcycle cylinder heads to determine if they were hard enough to retain the valve seats. Many people weld heads with non-heat-treatable welding rods and the heads remain soft. I always use rods that respond to heat treatment and so can be brought back to full hardness. I needed to determine what I was dealing with.

Firstly I made an incredibly simple rebound tester, based on the Shore Scleroscope of 1910, and a loose copy of devices that were given away by sales reps to garages for testing the hardness of cylinder heads etc. I found it very difficult to get consistent results. That was because things happened too fast for an old bloke to decipher. During the preparation of the videos which accompany this document I had an idea about how to simply turn it into a more useful tool. The idea worked and is described in Appendix 2. No electronics needed! A kid could make it.

Due to my initial lack of success with my first device I decided to build a more sophisticated replacement using some very simple electronics. This was easy to do thanks to the ready availability of Arduinos. Schematics and the Arduino software are fully described herein. The results that I got exceeded my expectations greatly. I was hoping to be able to get repeatable quantitative values, but which did not relate to any accepted hardness scales. In other words it would be a new TF hardness scale. Based on Leeb (1976) hardness conversion tables my very first test on some HSS tool steel gave an HRC (Rockwell C scale) value of 64 which is right where it should be. Subsequent tests on other hard steels showed similar agreement with expected numbers. Not only was this simple device quantitative but it related closely to the Rockwell C (HRC) and Leeb (HLD) scales. These scales are only aimed at hardish steels and not for aluminium. That does not prevent me from getting hardness numbers for materials other than steels, but it renders the values from legitimately being said to be from a particular scale. So I have my own scale (HFA) which just happens to coincide with the HLD when used on hardish steels.

I have made a two part companion video because these instruments really need to be seen in action. The first part describes the mechanical stuff and the second handles the electronic bits and pieces. Here are the links:

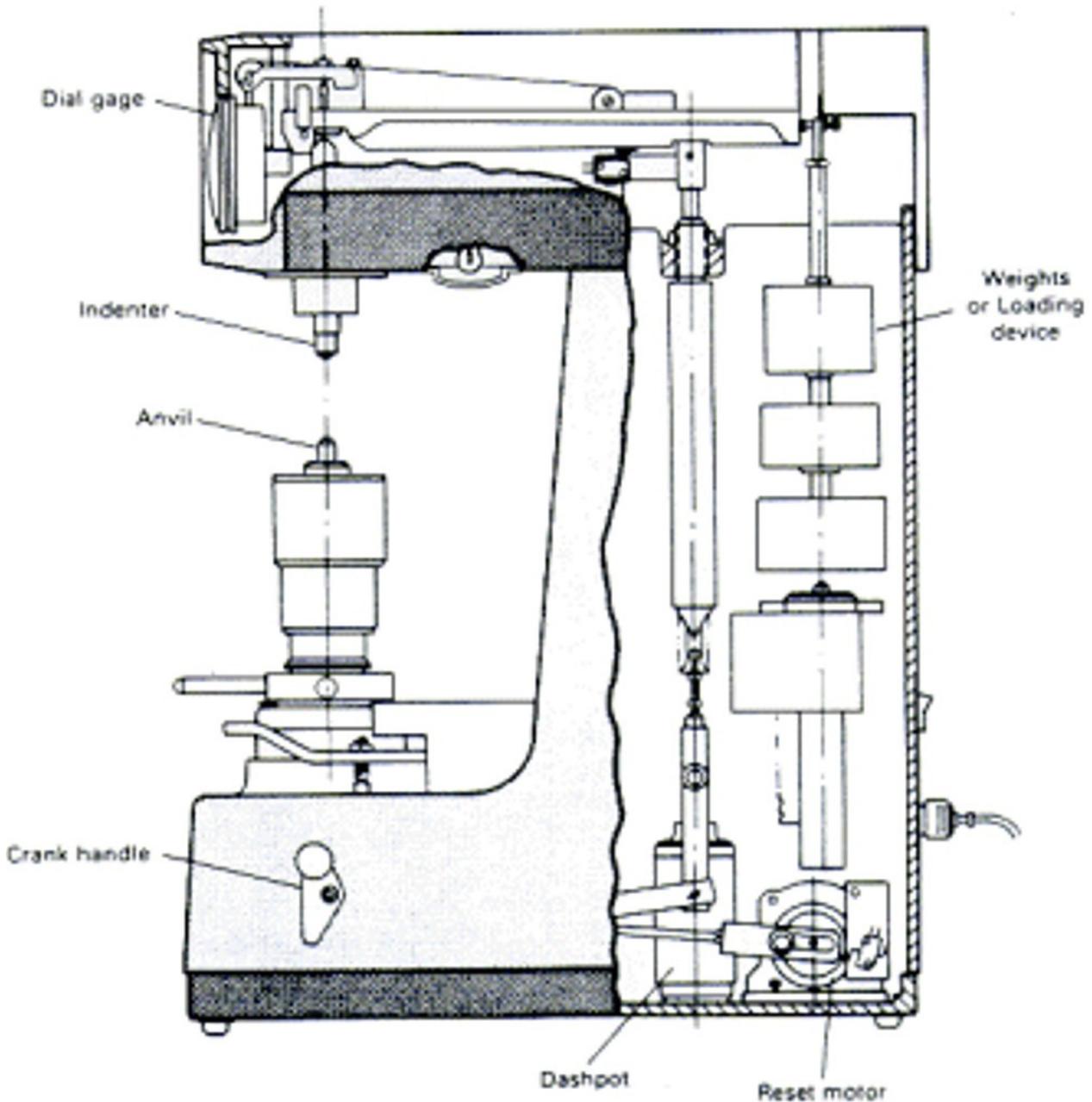
Part 1: <https://youtu.be/oxT8Uqq88xM>

Part 2: <https://youtu.be/4ghW9RNKiDA>

Hardness testing.

A rambling account.

Traditional hardness testing machines such as Rockwell, Brinell and Vickers are indentation devices. A force is applied to a stylus or indenter which penetrates a small distance into the piece under test. The amount of indentation is measured and converted to a value off a hardness scale. These machines are generally fixed and only accept limited size workpieces. They perform static tests, i.e. the load is static except while being applied and removed.

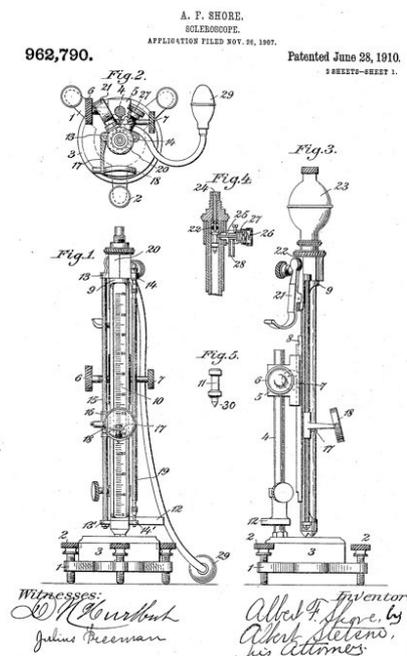


Here is a cross section of a Rockwell tester.

Less well known is a class of testers which rely on the rebound of a stylus or hammer and hence are called “rebound testers” or “dynamic”. This project is concerned with this genre.

In 1910 the Shore Scleroscope was patented. In essence; a stylus is dropped on to the workpiece and its rebound height is measured and related to a hardness scale. I wanted to do some hardness testing of motorcycle cylinder heads and made a simple bouncing ball Scleroscope copy but I found it difficult to visually determine the rebound height, it all happens so fast.

*** See appendix 2 to see how, during the preparation of the videos, I had an idea to turn this very simple and purely mechanical device into a useful instrument.**



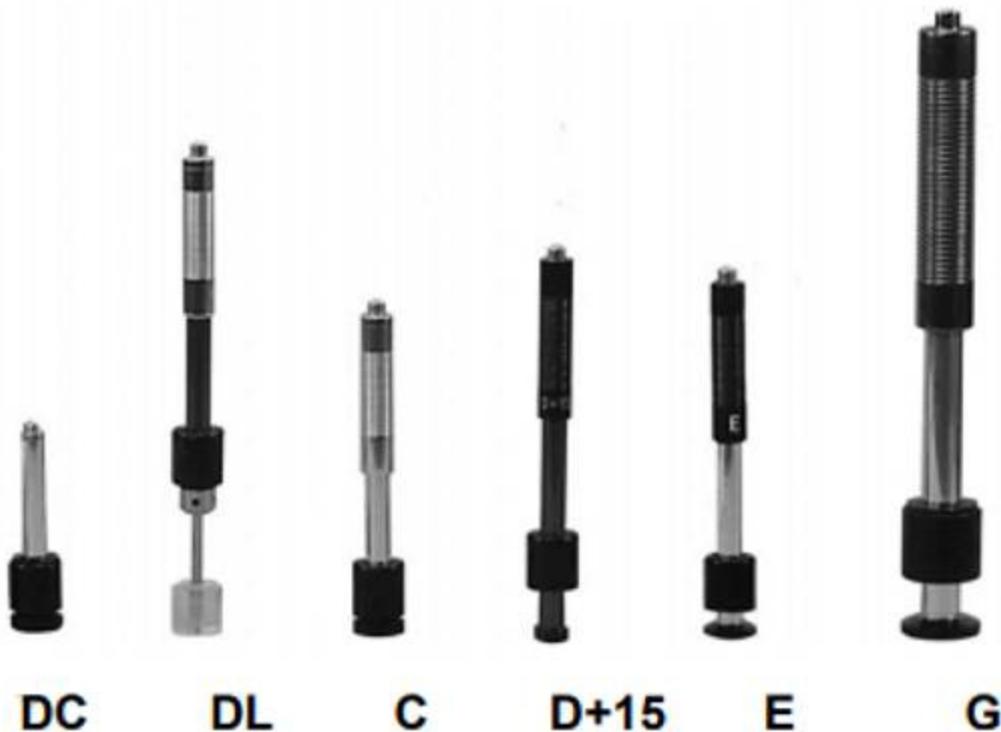
Shown here is a Shore patent drawing on the left and my simple version which uses a ball bearing as the hammer or stylus.

Because of the difficulties that I had of getting an accurate reading I decided to instrument it, but it is not so easy to do that. It is easy to check if the ball bounces above a given limit but not to measure the maximum height, so I decided to measure drop and rebound velocities. The Leeb device from the mid 1970s does that and it has the advantage of only needing the ratio of the two velocities to get the Leeb hardness value which with simple conversion correlates well with the Brinell and Rockwell scales. In fact it is accepted by ASME as a valid hardness scale. Another advantage is that the results are largely independent of the drop piece weight and shape. The Leeb testers are quite short and use a spring rather than gravity to power the stylus. This has the advantage that you can take the tester to the work of unlimited size rather than take a small sample to the tester.

Although I like the velocity ratio idea because of its simplicity I am not keen on how the Leeb devices do it. They use the proven method (as used in LVTs, Linear velocity Transducers) of a moving magnet in a coil but that requires a composite weight with at least 3 pieces, 1. Main body. 2. Magnet. 3. Contact piece (ball or point).

There are two things that I do not like. One is the multi-piece drop thingy, the other is that the very nature of the drop piece means that it is relatively long and the magnet is a long

way back from the contact point which means that the velocities are measured somewhat above the contact zone. I do like the direct velocity signal which needs no further processing though.



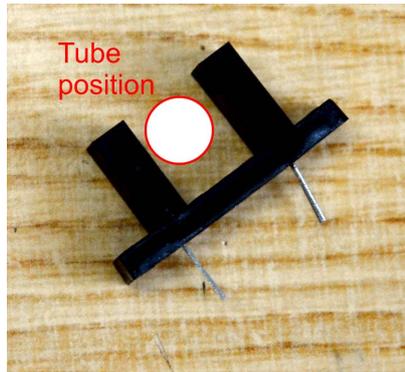
Various Leeb models

Anyway I decided to see if I could use timing over a short distance to calculate the velocities closer to the contact zone. Firstly I thought of using the Hall Effect by putting a sensor on one side of the tube and a magnet on the other side with enough separation to NOT fire the sensor. Then when a steel ball passed between the magnet and sensor the idea was that the sensor would trigger and untrigger as the ball passed. I planned to use two such sensor/magnet combinations separated vertically by a short distance, 2 mm seemed a good start.

I glued one hall sensor to the tube and tried it. It triggered perfectly but unfortunately it did not untrigger when the ball passed. If I pulled the magnet back enough to untrigger then it would not trigger in the first place as the ball passed. These sensors have a lot of hysteresis built in deliberately to avoid false triggering. There are probably low hysteresis sensors around but I did not want to turn this into a long development project.

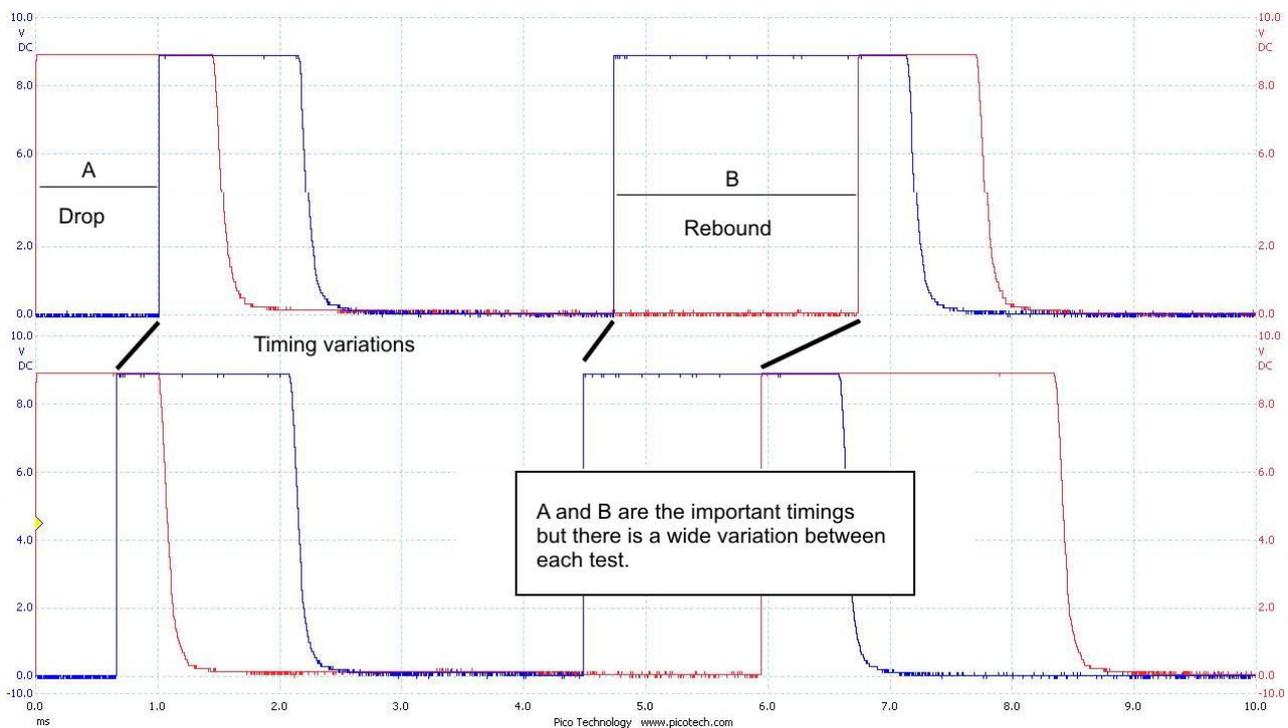
Then I started to think about IR optical slotted sensors, I have a bunch of those but with only 2 mm slots, I need 8 mm for this to fit around my drop tube. I did think of cutting and widening a couple but first I looked on the net and found some 8 mm ones, but they would take a couple of weeks to arrive. I had high confidence that this will be a good solution, but then my thoughts went to considering alternatives based on first principles. A capacitive solution was considered, I thought of a plate each side of the tube the capacitance of which would change as the ball went by. However, to read that I would need an RF source and frequency modulation and that seemed to be getting too complex.

A slotted optical sensor showing how it would fit around the drop tube.



Then I remembered that I had some proximity sensors that I got for speedometer pickups. Some initial playing around with one and the drop ball indicated no hysteresis and well defined switching points. I was so confident in these that I actually fitted two of them properly with 2 mm vertical spacing.

I hoped that the signals would allow accurate time measuring using a Picoscope (a PC oscilloscope). This would be replaced by a self contained Arduino/LCD box later. Unfortunately there was a fly in the ointment. When I did a number of "identical" tests I found a huge variation from test to test. Below is a composite pic showing two back to back tests which show some variation, neither were at the extremes of variation. I have been unable to trace the cause of such variation to anything in my setup and have come to the conclusion that it is just random variation in the sensors, after all they are not made as precision position sensors. This conclusion is bolstered by the varying pulse width.



Here are timing plots from two back to back tests. It is obvious the timing variation between the upper and lower cases. This seemed to be random variation from the proximity sensors.

I had a single pair of IR LED and IR photo transistor, which I mounted and the results were very consistent. I thought that these would be better than the slotted pickups which I ordered because I'll be able to mount 2 pairs with very little vertical gap, 1 or 2 mm,

whereas the slotted devices can be mounted no closer than the housing width, 6 mm or thereabouts. (I actually used 5 mm spacing not the planned 1-2 mm)

It had always been my plan to use two sensors of whatever type so that I could time the ball passing through a fixed known distance, just like at Bonneville. However with just the single pair, as a "let's try this" method I measured the pulse width and used this to calculate the velocities. The drop velocity coincided closely with what Newton would predict, I added the second pair and when I did a test on some HSS, using the Leeb HLD scale conversion I got a hardness value of 64 Rockwell C (HRC) which is just what you would expect.

About the reason for doing this electronic version. I have an occasional requirement to measure hardness. Steel as well as Aluminium. I had not been able to convince myself that I did a good job of reading the bounce height from a scale on my initial Scleroscope "copy". It seemed pretty easy to measure the drop and rebound velocities and that would give a more useful tool which I would be able to believe. Although I had attempted 3 methods (hall sensors, proximity sensors and lastly the IR LED and photo-transistor pair) I spent less than an hour playing around to get a repeatable method that was dead easy to implement in a final tool. I could have made it as a comparison "this is better than that" machine but I much prefer to have numbers that are repeatable day to day and year to year.

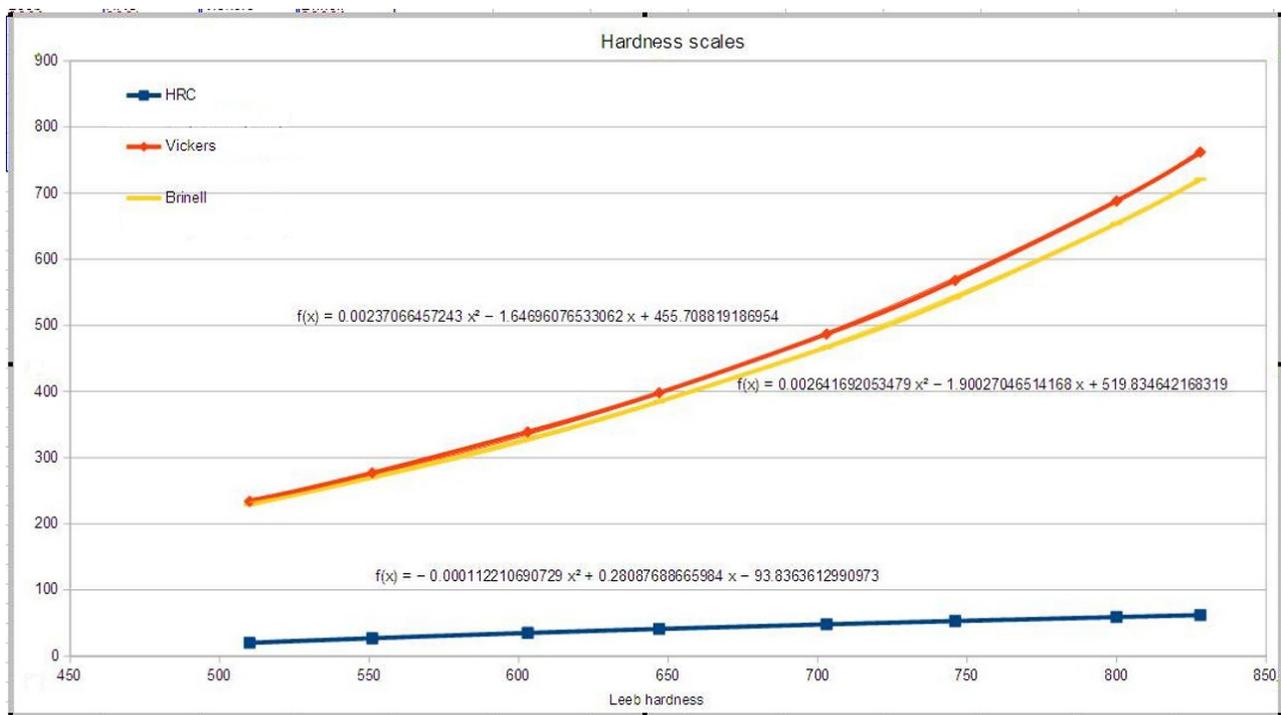


Measuring the hardness of some HSS tool steel with my rebound hardness tester.

I drop a 7 mm or 6.35 mm (1/4") diameter ball from 0.5 m height to give an impact velocity of 3 m/s and there is no noticeable indentation. I tried dropping the ball onto the same spot several times and there was nothing more than the expected statistical variation, that was on 6061 T6, pre-hardened 4140 and HSS. The HSS bounces almost back to the starting position.

Steel is no problem for the bouncing drop test, ball or bullet shaped stylus, both the Shore and Leeb systems are well accepted methods. A while back I looked into Rockwell and Brinell machines and concluded that DIY versions would not be too difficult. You can buy replacement styli, the rest you can make easily. However, I think that the Shore/Leeb/Foale hybrid will do most if not all that I want. One big advantage is that I can take my device to a workpiece of any size, whereas the indentation machines need small workpieces to be taken to the tester.

The Leeb hardness value is simply 1000 x the rebound velocity/drop velocity at contact. Here is a graph which plots various hardness scales against the Leeb numbers.

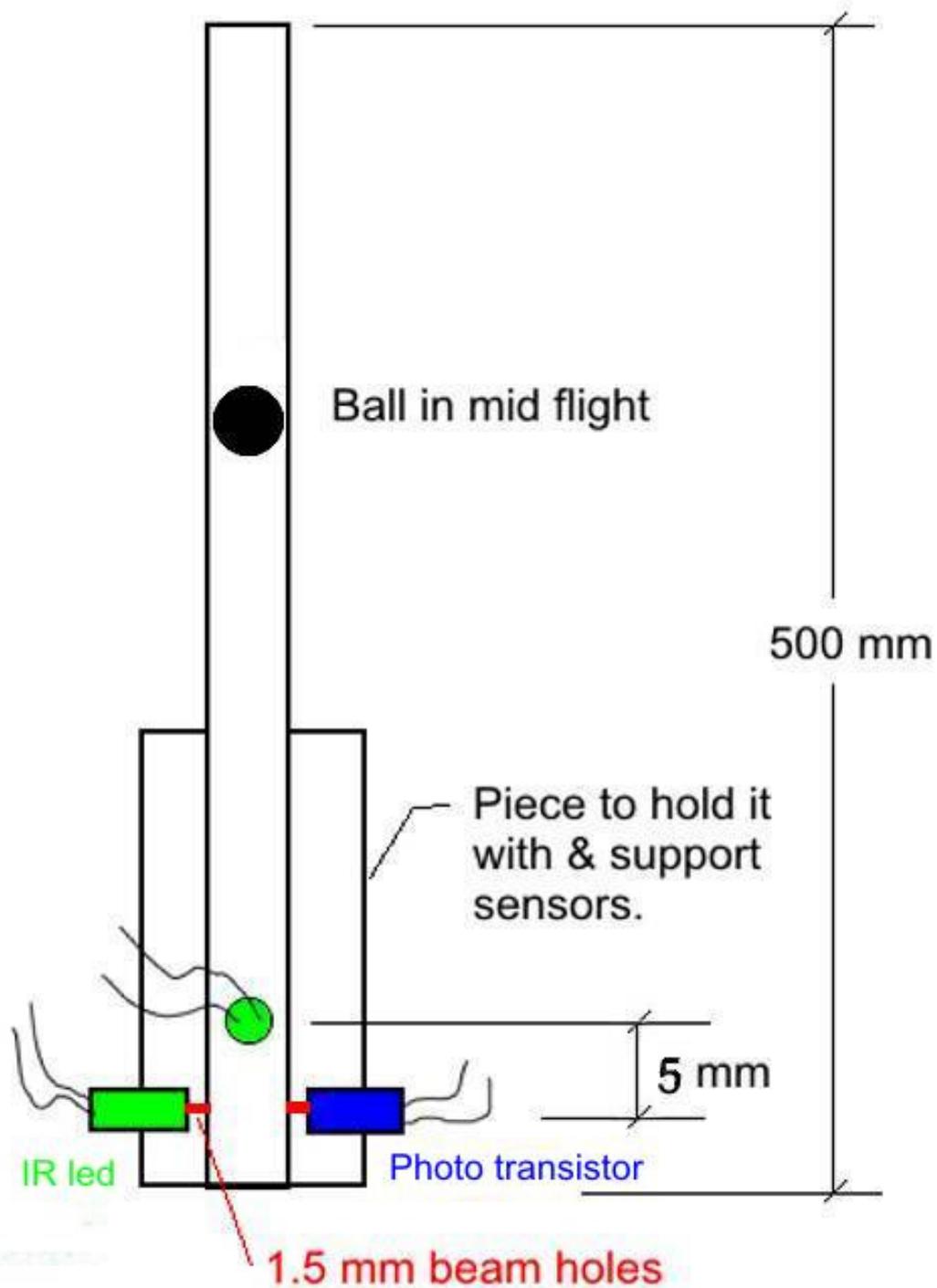


Conversion plots from Leeb hardness to other common standards.

I took the conversion values for HRC, Brinell and Vickers and fitted 2nd order polynomial equations which I have programmed into my Arduino. This gives conversion numbers for 1000 (infinite hardness) down to the minimum that my device will measure, determined by the height of the top sensor. So some numbers on the other scales may be outside of the range of validity as set down in ASME and similar standards. My device will still convert HRC down to 0, (400 HLD) although that would not be considered valid.

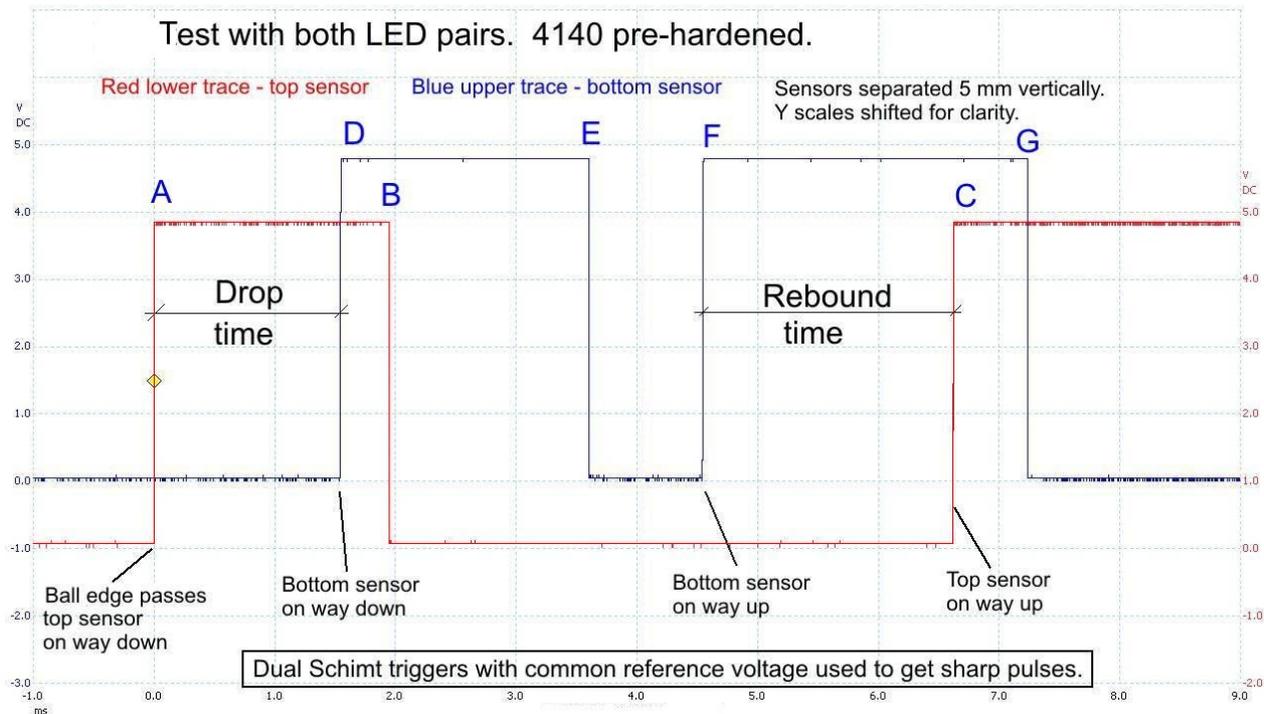
I used a Picoscope (an oscilloscope box for PCs) to measure the timings as the ball passed the two levels. I previously talked about having the two levels separated by 1 or 2 mm but after giving it some more thought I went for 5 mm. A 0.5 m drop height results in a 3 m/s velocity at the measuring area which causes the ball to travel 5 mm in 1.7 ms. I

considered this to be a sensible minimum to be able to measure accurately with simple methods. Testing has shown that the timing is accurate and repeatable.



Showing the basic construction. As the ball passes downward the time to pass from the top sensor to the lower one is measured and the incident velocity is calculated. Then the upward velocity of the first bounce is calculated from the time of passage from the lower to upper sensor. The increase in velocity from the measuring point to the work surface is calculated from gravitational acceleration.

The sensors comprise an IR diode and a photo transistor aligned across the drop tube such that a falling ball cuts the IR light to the transistor causing it to trigger. The sensor pairs are separated vertically by 5 mm, so this is the distance over which we do the timing.



This shows a measured timing sequence. The two traces are separated vertically for clarity.

Referring to the above illustration and working forward from the left we can follow the sequence of the timing. Note that the lower sensor must be mounted such that its beam is not broken when the ball reaches the test piece. Each change of state of both sensors is marked A-C for the upper sensor and D-G for the lower. Not all state changes are of concern for timing purposes. The timing of events B, E and F are of no importance and the software is programmed to ignore these. Let us work through each stage considering only A, D, C and G.

1. "A" is when the falling ball breaks the top beam.
2. "D" is when the falling ball breaks the second beam.
3. Therefore the time taken to travel the 5 mm between the upper and lower sensors is D minus A.
4. "F" is when the rebounding ball breaks the lower beam.
5. "C" is when the rebounding ball breaks the top beam.
6. Therefore the time taken to travel the 5 mm between the lower and top sensors is C minus F.

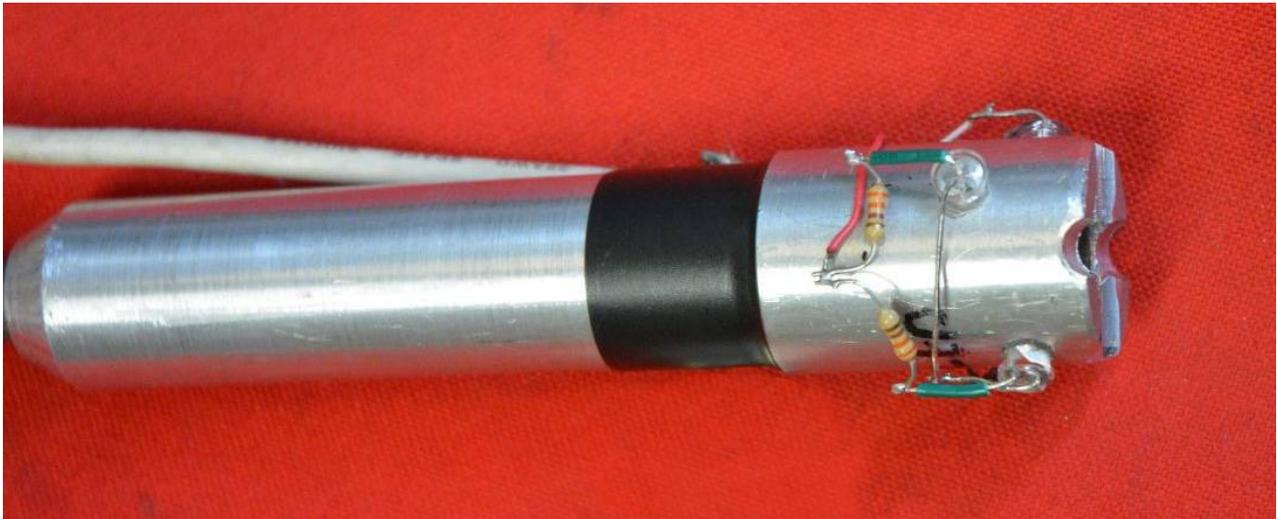
From these time intervals we can calculate the incident and rebound velocities at a point mid-way between the two sensor beams. That point is approximately 10 mm above the impact surface, but the velocity at impact can be estimated by adding on the extra velocity due to gravitational acceleration.

How to make it - mechanical

We have seen how the device works so let us now look at the physical implementation. Firstly, the very simple mechanical part, the drop tube, followed by the equally simple electronics and Arduino.



The drop tube with aluminium end piece for handling and as support for the sensors. The grooves at the bottom are an air exit and also allow it to sit on cylindrical objects like gudgeon pins and shafts.



The physical implementation before the sensors were protected with Sugru. The current limiting resistors for the diodes are mounted here and not the electronics box to allow a four wire cable in place of five.

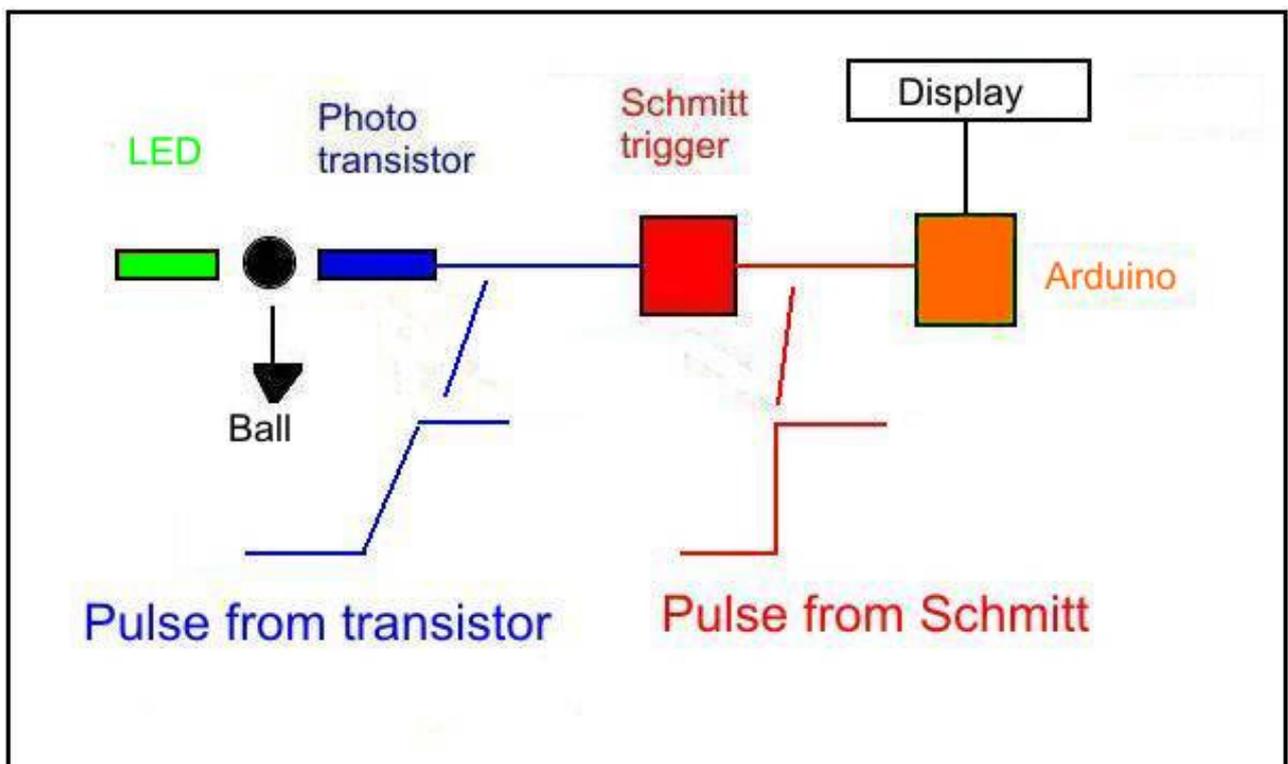
The simplicity is obvious. All built around a 0.5 m long, 8 mm ID steel tube. An aluminium tube is fitted around that for two reasons. Firstly, it provides a mounting for the two pairs of IR LED and IR photo-transistor. Secondly, it functions as a handle to hold the assembly square onto the work piece. IR light passages are drilled through of 1.5 mm diameter

which are enlarged on the outside to 6 mm to accommodate the sensor components. There are two current limiting resistors for the LEDs. Mounting them on the drop tube allows for a 4 wire cable to the magic box. If they had been mounted in the box then a 5 wire cable would have been necessary.



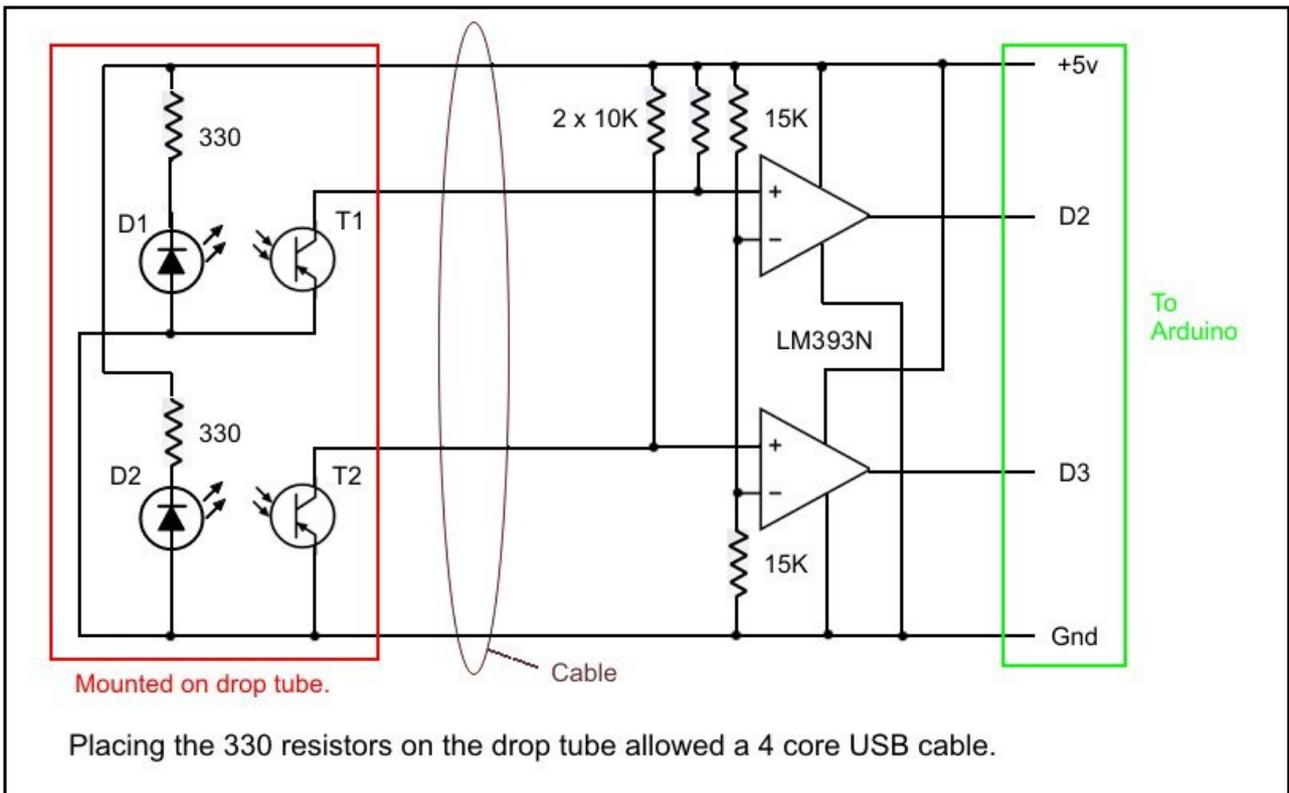
Sugru mastic was used to stabilise and protect the sensors. This was an experiment and it seems to be OK. Epoxy glue would be more usual.

How to make it - electronics

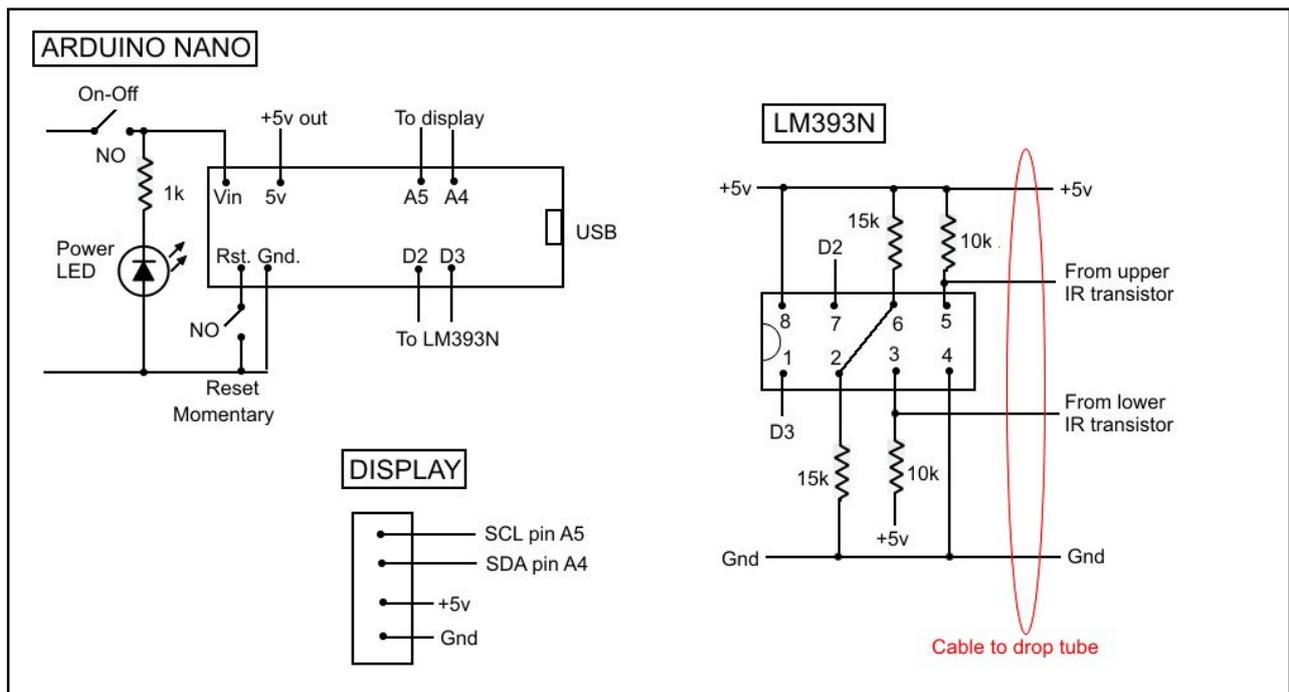


Total system concept diagram. It takes around 0.5 ms for the ball to close off an IR beam which gives too slow a pulse leading edge so that is sharpened up with a dual comparator configured as a Schmitt trigger. The sharp pulse only is fed to the Arduino.

The schematic above shows that the relatively slow rise time of the sensor output is fed into comparators biased to trigger a sharp pulse edge half way through the rise time from the sensors. The same reference level trigger voltage is fed to each comparator. These fast pulses are used to trigger timings in the Arduino which does the velocity calculations. A hardness number is then calculated from the ratio of rebound to incident velocities as per the Leeb testers. Conversion factors are applied to allow a display approximating to Leeb, Rockwell C, Brinell and Vickers hardness values.



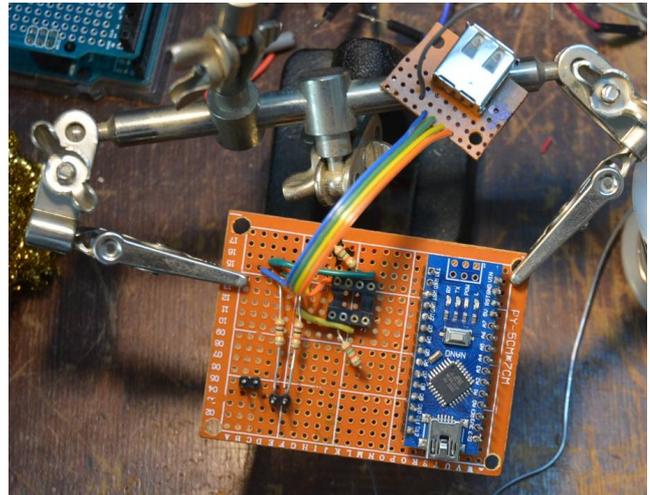
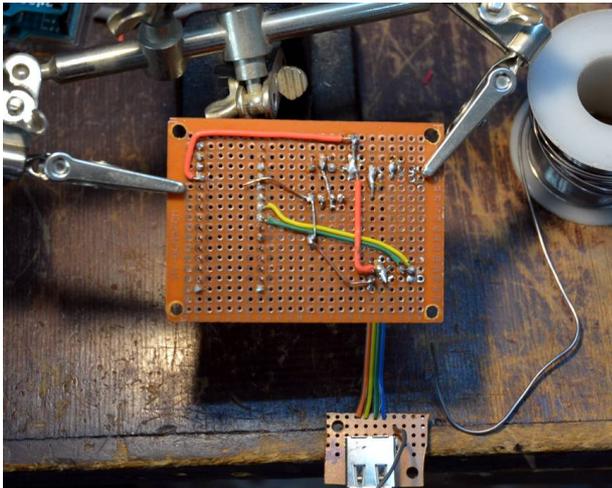
A schematic from the drop tube sensors to the Arduino connections.



Block diagram showing physical connections. Some people find this more understandable.

Parts list;

- 2 x IR photo transistors
- 2 x IR LEDs
- 1 x small power indication LED
- 1 x LM393 comparator chip
- 1 x Arduino Nano
- 2 x 10 k Ω resistors
- 2 x 15 k Ω resistors
- 1 x 1 k Ω resistor
- 2 x 330 Ω resistors
- 1 x LCD display 20 x 4 with serial decoder.
- 1 x N.O. Momentary button switch
- 1 x N.O. Power switch
- 1 x 9 v battery or wall wart – I prefer a battery because warts can be electrically noisy
- 1 x Plastic box to put it all in





An Arduino Nano does the thinking, A LM393 comparator sharpens the pulses from the photo sensors and the Arduino drives the display unit.

Some results

Hardness is very different from most other measuring units. Length, velocity, volts, weight are independent of the measuring instrument, although accuracy might vary. On the other hand, measured hardness depends on the measuring device.

Here are some results from various tests that I did when the instrument was first built. The tests were done on a wide sample as verification.

Steel

For HSS tool steel around 63 to 66 on the HRC scale.
Pre-hardened 4140 gave a value of HRC 50.
Both materials give numbers as expected.

Aluminium

The Leeb and HRC scales are specifically designed for steels only. Leeb conversion tables do not seem to cover the hardness range of Aluminiums. Most text on the subject of "hardness" make a point of saying "steel hardness testing". Anyway for me that is no big deal because I can measure 6061 T6 and use that as my own private comparison standard.

The numbers are Leeb hardness because HRC goes to zero at Leeb = 400 and is meant for steel anyway. I tested some motorcycle engine parts that I had in the workshop.

Heads. (Aermaachi 350)

UK cast with HT new. - 574
Used not welded - 539
Used not welded - 516
Used welded years ago - 379

Barrels. (3 examples)

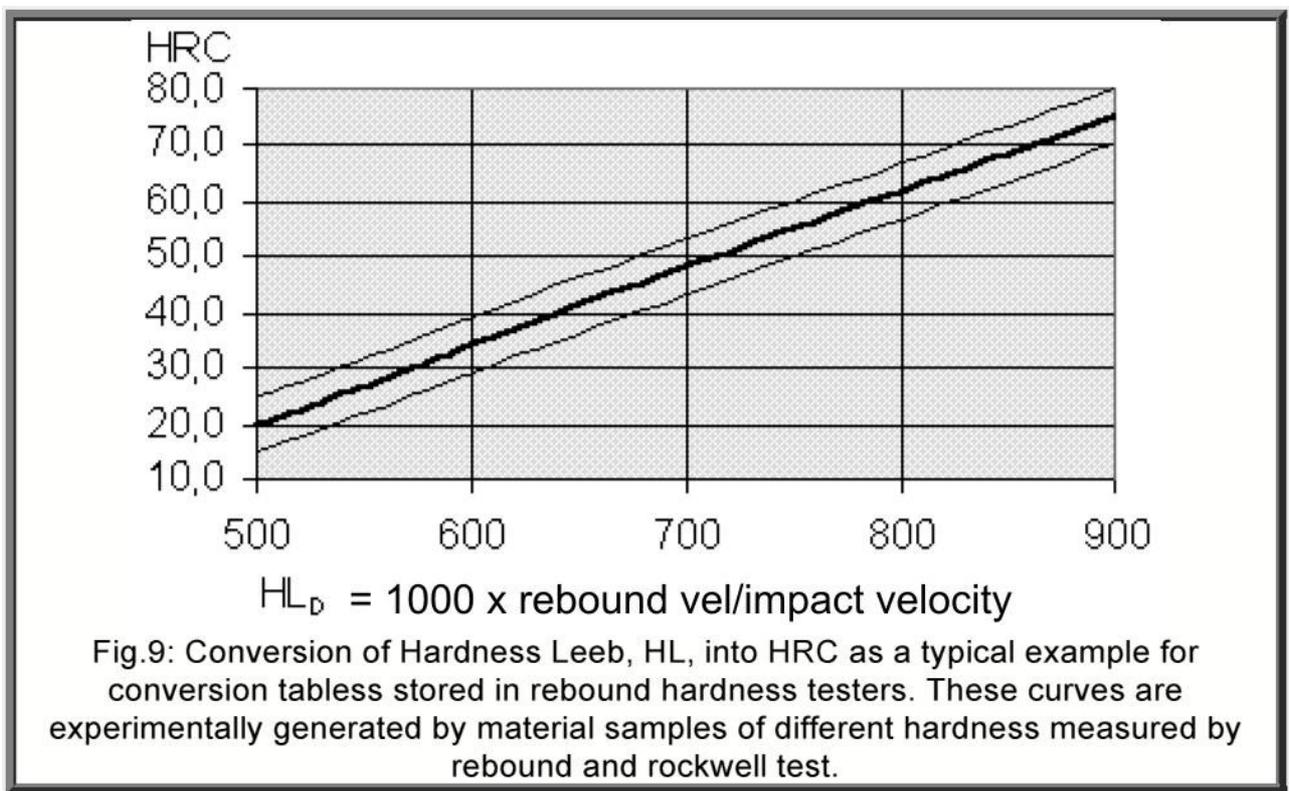
Used - 585
- 563
- 555

Top of pistons

GS500 new - 544 around 25 HRC if the scale applied to aluminium
GS500 used - 519
My magnesium used - 489
Aermacchi used - 430
" " - 423
Ducati 250, severely overheated – 415
A forged piston – 625 This had been overheated and damaged by a dropped valve

As a reference 6061 T6 – 555/560

Here is the conversion scale from Leeb HLD to HRC.



The Leeb hardness number is very simple. $1000 \times \text{rebound velocity} / \text{drop velocity}$. Velocity not rebound height/rebound height which is different. My device has a physical lower limit to what it can measure. The rebound height must be enough for the ball to cover the light beam of the upper sensor, this will be around 12 mm. Anything under that hardness is best tested by a finger nail indent using a standard ASME finger nail.

Note that the Leeb calculation does not mention drop height nor ball size/shape. Being the ratio of velocities these factors are irrelevant, within sensible limits. Dropping from 5 mm is unlikely to yield useful results.

Appendix 1.

Arduino programme.

Here is the programme to upload to an Arduino Nano or other compatible micro.

```
#include <Wire.h> // Communications library
#include <LiquidCrystal_I2C.h> // Library for printing to display

LiquidCrystal_I2C lcd(0x27,20,4); // set the LCD address to 0x27
and a 20 x 4 display.

// Change these two constants to suit your own hardware
//implementation.

float s = 0.0049;
// Vertical constant distance between sensors in m.

float twoas = 0.1813;
// Constant 2as is used to correct velocity using 9.25 mm final
// drop after mean sensing.

void setup(){
  lcd.init();
  lcd.backlight();
  lcd.clear();
  lcd.setCursor(1,0);
  lcd.print("Leeb");
  lcd.setCursor(1,1);
  lcd.print("Rockwell C");
  lcd.setCursor(1,2);
  lcd.print("Brinell");
  lcd.setCursor(1,3);
  lcd.print("Vickers");

  pinMode(2, INPUT_PULLUP);
  pinMode(3, INPUT_PULLUP);

  myCalc();
```

```

}

unsigned long t[4], dt1, dt2;
float v1, v2 ;
uint16_t leeb, Rc, Brin, Vick;

void myCalc() {
    // Get timings. This works better than using interrupts.
    while (! digitalRead(2)) { }
    t[0] = micros();
    while (! digitalRead(3)) { }
    t[1] = micros();
    while (digitalRead(3)) { }
    while (! digitalRead(3)) { }
    t[2] = micros();
    while (! digitalRead(2)) { }
    t[3] = micros();

    // Calculate time intervals
    dt1 = t[1] - t[0];
    dt2 = t[3] - t[2];

    // Calculate and adjust drop & rebound velocities
    v1 = (1000000*s)/dt1; v1 = sqrt(v1*v1 + twoas);
    v2 = (1000000*s)/dt2; v2 = sqrt(v2*v2 + twoas);

    // Calculate leeb hardness value and convert to other scales
    leeb = (1000 * v2)/v1; // Leeb value
    Rc    = -0.000112211 * leeb*leeb + 0.2808769 * leeb -
93.83636; // HRC
    Brin = 0.00237066 * leeb*leeb -1.646908 * leeb + 455.7088;
    // Brinell
    Vick = 0.00264169 * leeb*leeb -1.900271 * leeb + 519.8346;
    // Vickers

    myDisplay();
}

void myDisplay() {
    if (leeb < 1000 && leeb > 0) {
        if (leeb >= 950) {
            lcd.setCursor(0,1);  lcd.print("**High value**  ");
        }
        else if (leeb < 400) {
            lcd.setCursor(0,1);  lcd.print("** Low value **  ");
        }
        if (leeb >= 400) { //Value when Rc = 0
            lcd.setCursor(16, 1);  lcd.print(Rc);
        }
        lcd.setCursor(15, 0);  lcd.print(leeb);
        lcd.setCursor(15, 2);  lcd.print(Brin);
        lcd.setCursor(15, 3);  lcd.print(Vick);
    }
}

```

```
else { // When !(leeb < 1000 && leeb > 0)
  lcd.clear();
  lcd.setCursor(0, 1);  lcd.print("***** WARNING *****");
  lcd.setCursor(0, 2);  lcd.print("Value out of range.");
}
}

void loop() {
}
```

Appendix 2.

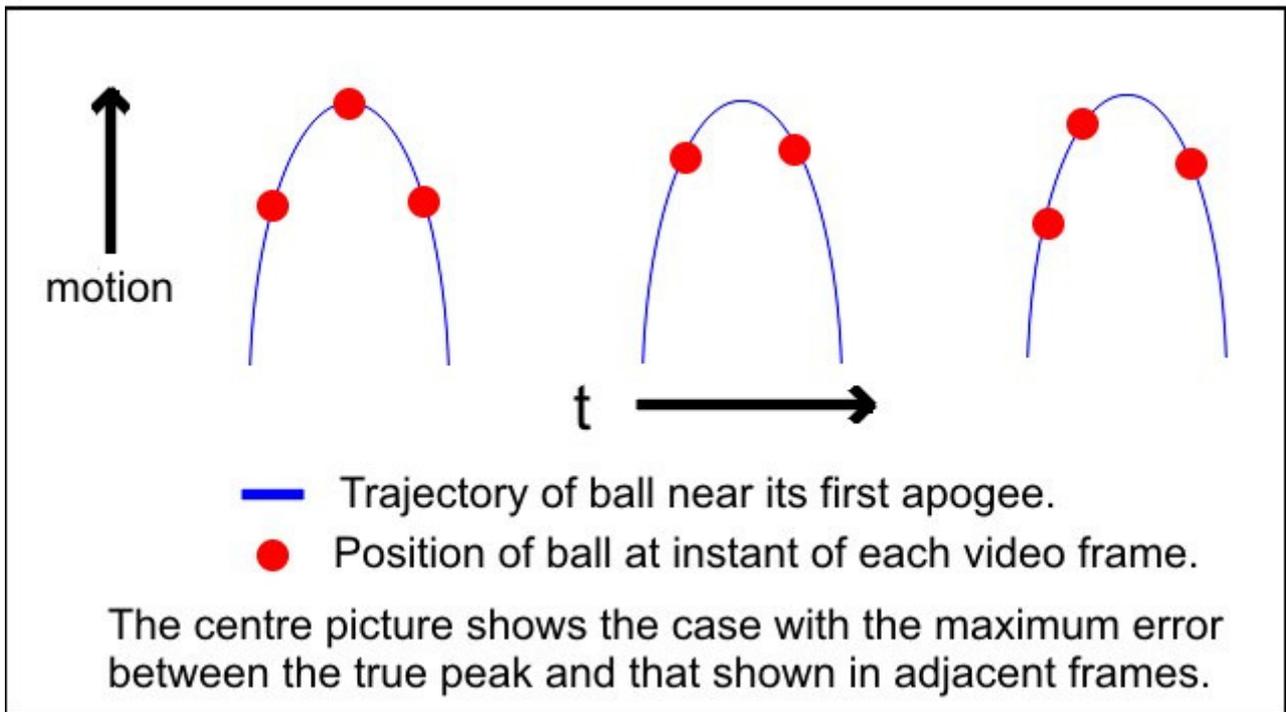
Improving the breed



My first tester was as simple as it gets. A clear plastic tube, an aluminium base, a scale and a ball bearing. Unfortunately, I found it very hard to detect the bounce height of the ball consistently because the action was so fast. Whilst preparing the videos it occurred to me that videoing the drop and rebound at a high frame rate and viewing at a slow rate should make it easy to get repeatability. My DSLR had a max frame rate of 50 fps and this

worked as expected. With the ball near its apogee of the first bounce I could advance the video frame by frame to pick the highest position against the scale. Easy peasy.

How accurate is it? That depends on the camera frame rate and is easy to calculate.



At say 50 fps the period between each video frame is 20 ms. The worse possible situation is shown in the centre of the above illustration. We can see that the maximum period between the real maximum and the filmed maximum will be half the frame period or 10 ms.

The ball movement is due entirely to gravity so we calculate the difference in height by using the expression $s = 0.5 * a * t^2$

Where s is the error distance, a is gravitational acceleration and t is the time period.

That becomes $s = 0.5 * 9.807 * 0.01^2 = 0.00049 \text{ m} = 0.49 \text{ mm}$ (0.01 is 10 ms in seconds)

In round numbers we see that the maximum error between video indications and the true position is 0.5 mm. At 100 fps that would reduce further to 0.125 mm, and at 25 fps it would be 2 mm. Any camera frame rate of 50 fps and above would be capable of giving very accurate values. Many smart phones have video frame rates of this level and above as well as the ability to replay slowly.

So with the addition of a smart phone to the super simple physical tester we have a fairly useful and accurate scleroscope, WITHOUT those dreaded electronics. Now as we have measured rebound height we could apply the Shore hardness scale directly, in which case our drop height needs to be regulated to the Shore standard. Alternatively, we could use the drop and rebound heights to convert to a velocity ratio and use the Leeb scales. The later would be my approach but I have my electronic version and have no need to use this method.

The videos show this in action as well as the slow motion results.

It does not get any simpler.