Slide Rules for Computer Programmers
or
53 6c 69 64 65 20 52 75 6c 65 73 20 66 6f 72 20 43 6f 6d 70 75 74 65 72 20 50 72 6f 67 72 61 6d 6d 65 72 73

Despite sounding weird and challenging other visual oxymorons like "invisible ink" and "bricked-up windows" there was a time when computer programmers needed some seemingly old-fashioned calculating aids.

Rewriting history

This is a hextraordinary tale from an era when computer programmers of the day still relied on slide rules and other similar aids!

Many calculating histories write off the slide rule around the 1970s. The pocket electronic calculator is considered the final nail in the coffin of the slide rule. In most timelines the birth of the pocket electronic calculator is the *"Cal-Tech"* prototype that came out of the American Texas Instruments (TI) research laboratory in 1967. It weighed more than a kilo but soon after lighter but still chunky commercial products came on to the market. For example, in 1970 the *Canon Pocketronic* and the *Monroe 10* – both based on the TI prototype.

However, the death of the slide rule and other similar analogue calculating aids did not, as often recounted, happen overnight in 1970. It was the dawn of a new "portable digital world" but the old and new guard were to flourish alongside each other for another decade.

Surprising longevity

It is often forgotten but some important mitigating factors extended longevity of the slide rule and other similar aids longer after the birth of the electronic pocket calculator.

Limitations and specialist use

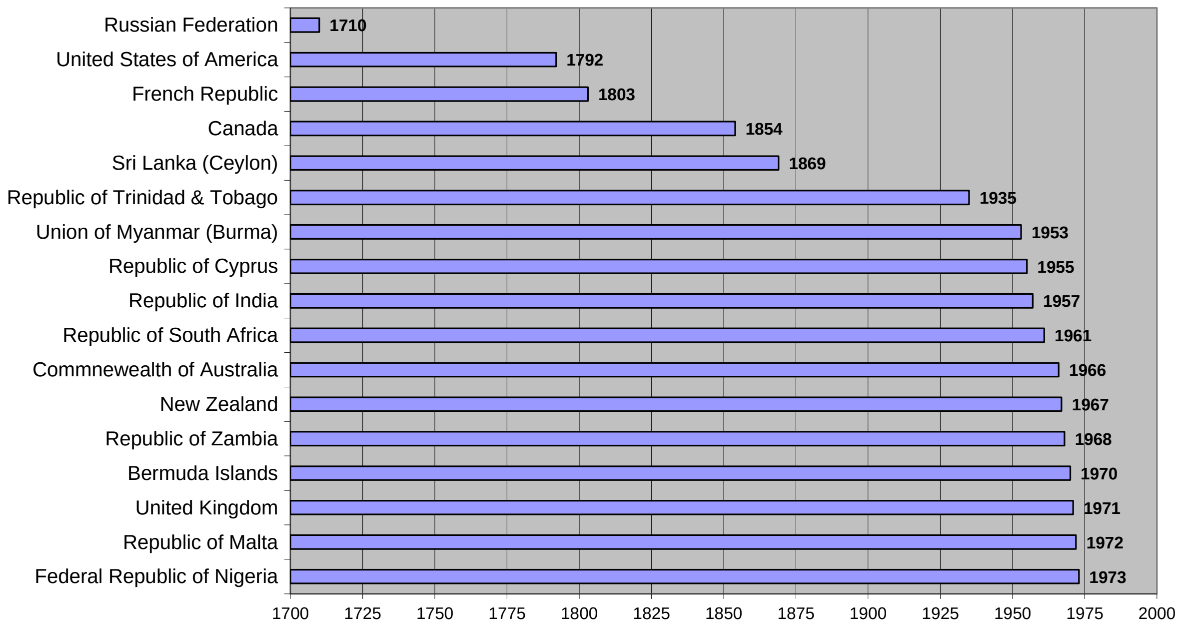
The first pocket electronic calculators could only do 4-function base-10 decimal arithmetic – i.e. addition, subtraction, multiplication and division. Built-in functions for advanced mathematical or trigonometric calculations would come later. This meant they posed no immediate threat to slide rules or other analogue calculating aids with specialist trade or commerce based scales or special algorithms.

Price

Only companies could afford pocket electronic calculators when they first came on to the market. Adjusted for "money of the day" back in 1970 they would have cost the equivalent of over USD 2000 (€ 1600). It would be years before prices tumbled so much that pocket electronic calculators became affordable consumer "must haves".

Decimalisation

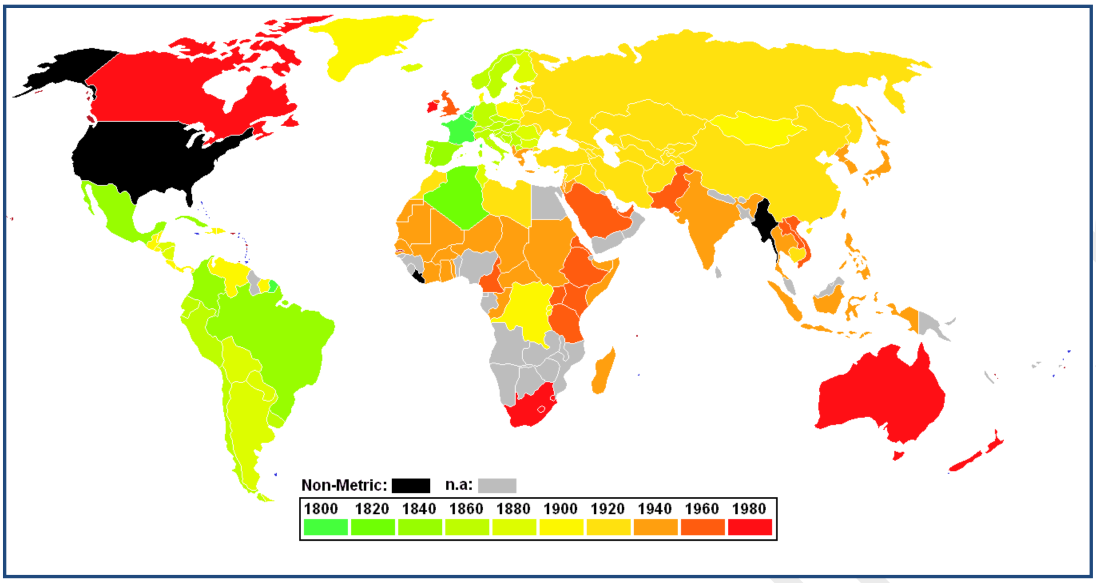
Decimal based pocket electronic calculators were ill-suited to non-decimal currency calculations. Although decimalisation started in the 18th century it is easily forgotten that many currencies such as the UK pound sterling did not go decimal until the 2nd half of the 20th century.



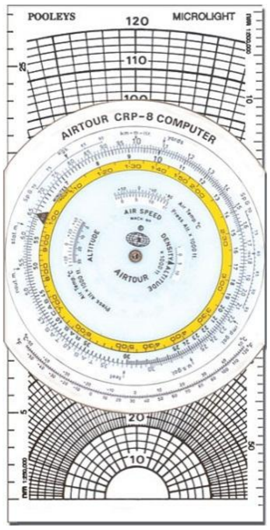
Non-decimal currencies still exist. The Islamic Republic of Mauritania has 5 Khoums to 1 Ouguiya and The Republic of Madagascar has 5 Iraimbilanja to 1 Ariary.

Metrification

Perhaps surprisingly much of the commerce in the 20th century was still based on imperial weights and measures – something early pocket electronic calculators could not handle. The French Republic started the change from imperial to metric in 1799 but it would take until 1980 for many other parts of the world to make the switch.



Even today metrication has still to come to the United States of America, the Union of Myanmar (Burma) and the Republic of Liberia.



"Health & Safety"

To get or renew their licence, any private or commercial aircraft pilot must prove they can use a slide rule to set a course and direction manually. So such slide rules or whizz wheels, often renamed flight computers, are still being made and sold. For example, the *CRP-8 flight computer* from the UK maker Pooleys for micro lights.

Programming aids

Despite being part of the new digital age, it was old-fashioned analogue calculating aids rather than early pocket electronic calculators that computer programmers needed in the early 1970s. This is because in that era computer programming was, compared with today, low-level "machine/bits and bytes orientated" and they needed to calculate in base-8 or more commonly, base-16.

History of computing in "2 minutes"

By the New Millennium computer programming had changed out of all recognition. But the reason computer programmers in the 1970s gave the slide rule and similar aids extra longevity is rooted in how computers evolved and in particular, how computer programming changed.

Frenchman Blaise Pascal (1623–1662) is credited with inventing the first mechanical adder in 1642. Then not a lot happened until the 19th century. Then people like Charles Babbage (1791–1871) came up with his *Difference* and *Analytical Engines* in the 1840s and Herman Hollerith (1860–1929) innovatively used the *Punched Card* to record the American census in 1893. Hollerith's company was later to become IBM. Interestingly the first computer programmer turned out to be women – Ada Byron, the Countess of Lovelace (1815–1852). From notes of her work with Babbage it was Ada who pioneered the concept of a "computer program".

WWII and large defence budgets drove the next surge of innovation and developments like the *Colossus* – the first electronic, digital programmable computer. It was designed and built for the British code-breakers at Bletchley Park by Tommy Flowers (1905–1998) in 1943. The *UNIVAC* (Universal Automatic Computer) was, in 1951, the first commercially available general-purpose computer. From here on computers evolved according to generations. The 1st generation had vacuum tubes. Then came the transistor based generation followed by a generation with silicon based integrated circuits. The current 4th generation uses very-large-scale integrated circuits (VSLI).

All this overshadows a basic principle at the heart of all computer programming - all digital computers work on a two-state system. Like a light switch, the smallest building block of a computer is a "switch" that is either "on" or "off". Having just two states, digital computers naturally adopted the binary number system for their switches:

1	**1**	**0**	**0**	**1**	**=**	**25**
on	**on**	**off**	**off**	**on**		

From this comes the term "BIT" or binary digit for each switch. For convenience, like having 12 eggs in a dozen, it was decided to de facto group 8 bits into a "BYTE". 1970 was the era of room-size 3rd generation computers like the *DEC PDP-8*, the *ICL 1900* and the *IBM 360*. The popular programming languages were mainly ASSEMBLER, FORTRAN (Formula Translation), COBOL (Common Business Oriented Language) and PL/I (Programming Language One). Binary is intuitive to a computer but not to a human. Instead two shorthand forms that worked well with bits and bytes were used to display and represent binary values - octal (base-8) and hexadecimal (base-16).

DECIMAL	BINARY	OCTAL	HEXADECIMAL
0	0000	0	0
1	0001	1	1
2	0010	2	2
3	0011	3	3
4	0100	4	4
5	0101	5	5
6	0110	6	6
7	0111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F
16	10000	20	10

The advantage of these shorthand forms is more obvious with looooong binary values. For example, 1001 0101 0001 0010 is unfathomable but in hexadecimal it is 9 5 1 2 or just 9512_h[1]. Hexadecimal, favoured by IBM, quickly became more popular and the use of octal faded. Besides "Big Blue" becoming the dominant force in the market place hexadecimal was more suited to the basic computer building block. Computer architectures of the day used 3 bits to display or represent a value in octal compared with 4 bits in hexadecimal. This meant two hexadecimal values fitted snuggly into a byte without any need for adjustment. But the relevance and the need for computer programmers to work in hexadecimal would eventually fade as programming languages changed and more human/logic-based syntaxes replaced the machinelike syntaxes of the 1970s.

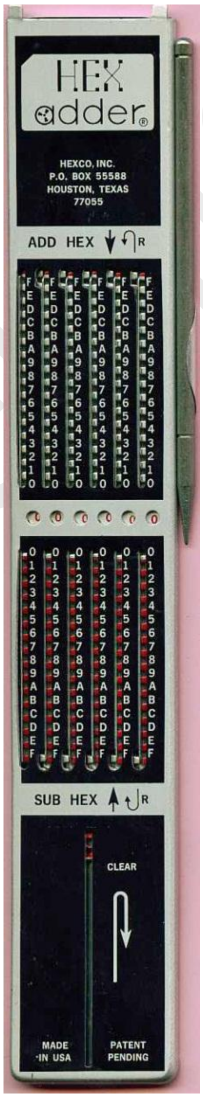
Hexadecimal programming aids

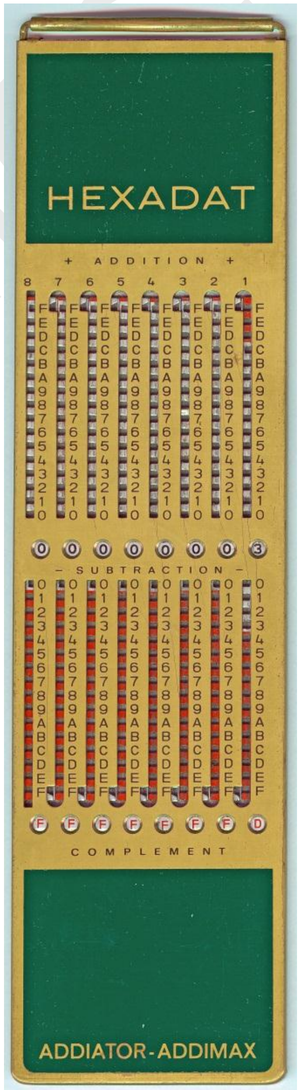
In the 1970s computer programmers and computer technicians could choose from five different types of analogue hexadecimal calculating aids.

[1] Subscripted "h" is one of many ways for showing a given value is in hexadecimal.

Slide Adder

This form of calculating aid was invented in Russia by German musician Heinrich Kummer (1809-1880) in 1846. But it was a German company, *Addiator GmbH*, founded by Carl Kübler (1875-1953) in 1920 that is the best-known manufacturer of the *"sliding bar adder with an accompanying stylus."* Addiator was one of several companies who made such mechanical adders[2]. Early adders were only capable of 2-function base-10 decimal arithmetic – i.e. addition and subtraction. Using the stylus and the appropriate slots, decimal values could be added or subtracted - the "carry" done manually by moving stylus around the "shepherd's crook" bend at the top of the relevant column. Alongside the basic decimal adder some specially designed adders were made for non-decimal currencies (Pound Sterling and Rupee), time and imperial weights and measures (Feet/inches, Tons/Cwts, etc). But Addiator, and some others, also extended the basic carry design to a maximum of 16 slots per column. In July 1967 Addiator was granted a patent (DE1963993U) for a hexadecimal adder. With its eight 16-slot[3] columns (233 x 60 x 4 mm) the brass *"HEXADAT"* is one of the most impressive adders ever made. Alongside addition and subtraction, the company also copyrighted a companion printed conversion table so that the HEXADAT could also convert hexadecimal answers into decimal. Across the "big pond" two former IBM System Engineers formed the Texas based *HEXCO Inc.* In 1968 they trademarked and started selling a simpler 6-column competitor *"HEX Adder®"* (241 x 38 x 5 mm). Strikingly, HEXCO's promotion slogan for their adder was: *"IBM 360 Programmers without 16 fingers ?"*.

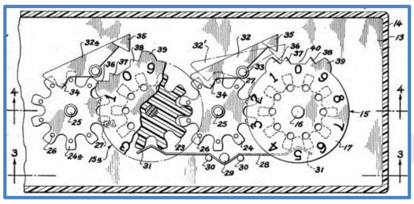




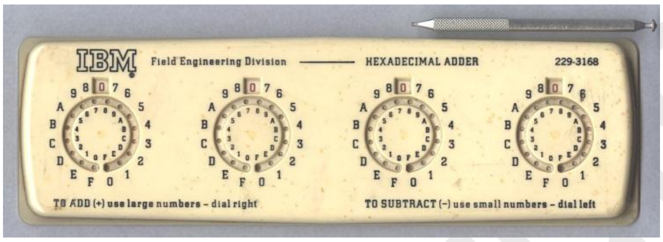
[2] The collective name that became synonymous for all such inexpensive flat metal calculators.

[3] Addiator also made and sold an 8-slot octal version – the OCTADAT.

Dial Adder



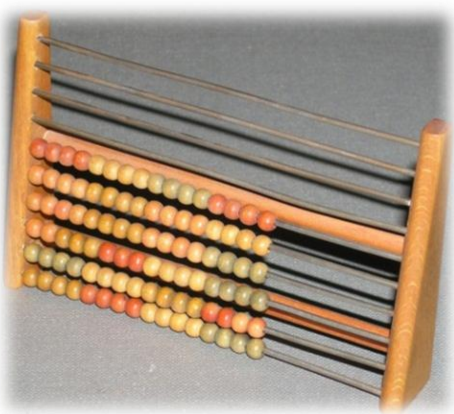
Such calculating aids are descendants of the 17th century Pascal machine. It is almost an insulting oversimplification but the main benefit dial adders had over slide adders was that the carry became an automatic part of the internal dial or cog mechanism. Around the late 1960s and based on an idea from an IBM programmer, the *Field Engineering Division* of IBM in the USA commissioned the New Jersey based *Sterling Plastics Co.* to make a special hexadecimal version of their patented (US2797047A, CA566754A and GB773099A) *"Dial-A-Matic®"* automatic adding machine.



For the *IBM "Hexadecimal Adder"* Sterling Plastics squeezed 16 instead of 10 values around each of the four inner and the outer rings. Addition was done by using the stylus to dial clockwise the required hexadecimal value according to the outer rings. Subtraction used the same process but this time dialling anticlockwise the values on the inner ring. Like the original by Sterling resetting to zero had to be done manually with the stylus. This special IBM version was also made from the same type of plastic but it was longer and wider (235 x 75 mm) and had a built-in metal stylus.

Abacus

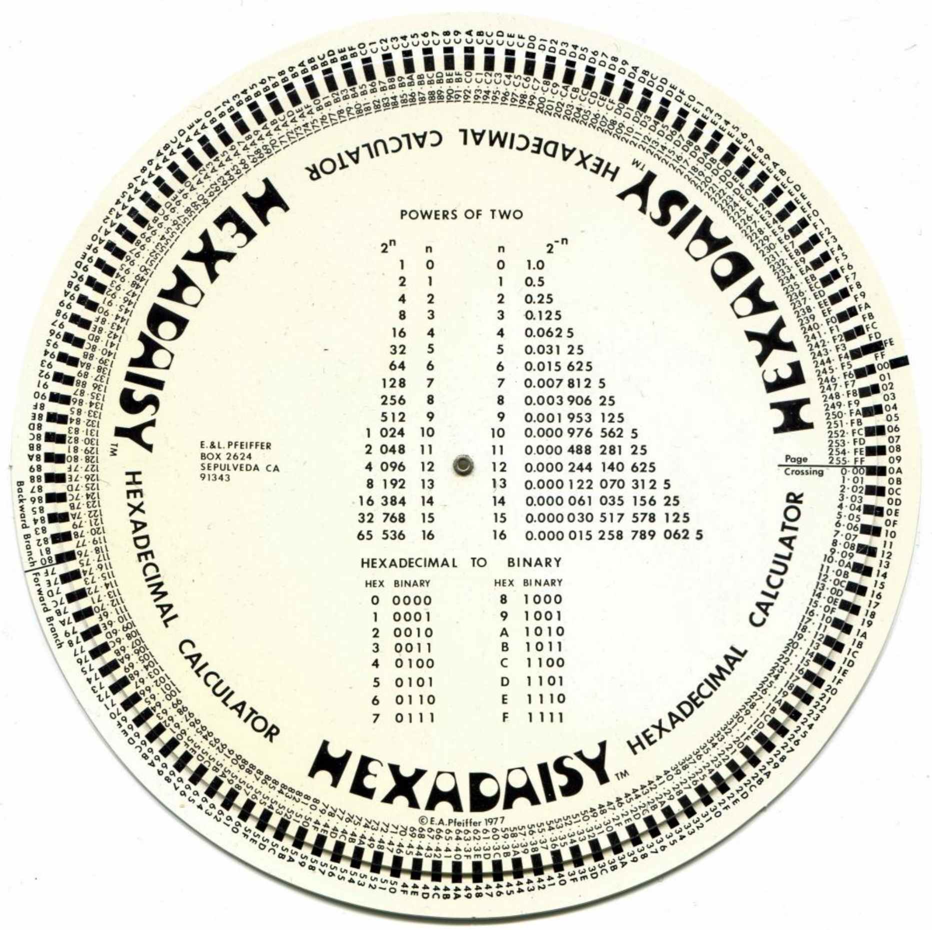
The earliest abacus dates from about 2700 BC but it was "reinvented" in 1967. IBM was determined to make the use of hexadecimal and multiples of 16 (e.g. as used in its addressing architecture) the "world standard" for computers and seemingly they never missed an opportunity to promote this aim.



This remarkable wood and metal *"Hexadecimal Abacus"* (120 x 75 x 20 mm) possibly came as a gift when IBM delivered a model 1130 computer to the University of Turku, Finland. The maker of the abacus is unknown and it carries no IBM branding or logo. So it could have been a local initiative from IBM Finland or even amusing gift to the university from a private individual to mark the delivery of the new computer. A notable design feature[4] is that the wooden beads from the top rows of the counting frame were "cannibalised" to make six complete rows of 16 beads!

Circular Rule

Sadly because of the passage of time little is known about the original California based holder of copyright A695383, *Erich A. Pfeiffer*, or the manufacturer of a unique circular hexadecimal calculator: *Hexadaisy Products* of California. The white vinyl plastic circular slide rule (Ø 172 x 2 mm) was imaginatively called and trademarked the "*HEXADAISY*™" – presumably because it looks a bit like the common daisy, a member of the Aster family of flowering plants.



---

[4] Ironically over 20 years later a patent, US4812124, for a *Hexadecimal Abacus* was granted to Jesse L. Colodner and Lawrence A. Clopper III on March 14th, 1989.

From the copyright information and the instructions accompanying the circular rule, it dates from 1977. It is single-sided with two independently rotating discs. Two pre-printed tables and a scale in both decimal and hexadecimal, 0-255 and 00-FF, runs around the slightly smaller inner disc. The same scale runs around the outer disc but only with the hexadecimal notation. It could be used for adding/subtracting hexadecimal numbers and converting to/from decimal and hexadecimal. But two uses, calculating offset or relative computer programming branch addresses and two’s complement of hexadecimal numbers, would have been particularly helpful to Assembler language programmers. Strikingly throughout the instructions all the examples, even for simple addition or subtraction, are given in terms for and familiar to computer programmers.

Linear Rule

Despite the popularity of slide rules starting to wane mid-century, in the early 1970s three slide rule hexadecimal related patents were granted: US3670958A, US3716186A and US3712974A. As ever, the existence of a patent is no guarantee that it ever spawns a commercial product. But it looks like one patent, US3716186A, granted to Tulsa, Oklahoma based Kent B. Comfort in February 1973 for a *Hexadecimal Slide Rule* did make it to market.

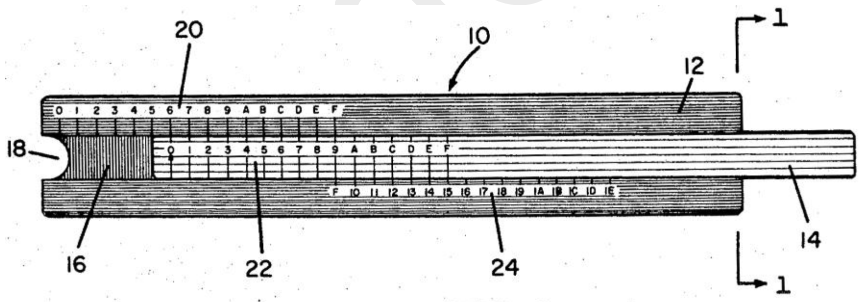
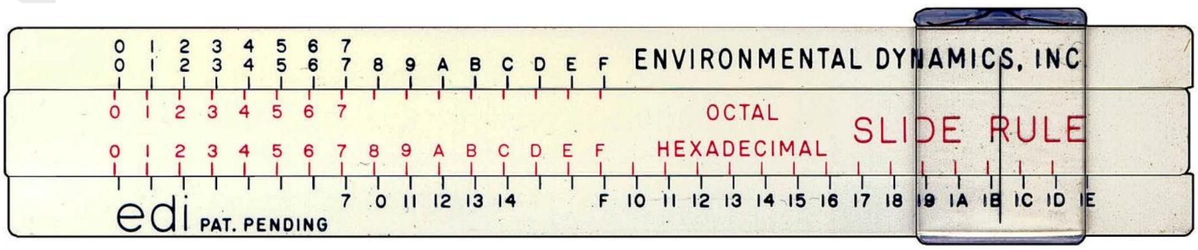


Fig. 1

The simplistic layout and predictable range of the scales shown in the patent drawing is treacherously deceiving. As claimed in the patent application, with this layout it is possible to: (i) add and subtract decimal values with the answer in hexadecimal, (ii) add and subtract hexadecimal values and (iii) handle any carry needed in either mode of calculation.

Around 1972 or 1973, probably as a promotional gift, a slide rule based on this patented design was made.



The moulded single-sided plastic pocket (155 x 28 x 5 mm) *"Octal Hexadecimal Slide Rule"* promotes a company called: *Environmental Dynamics Incorporated* (edi). The only change to the patented design is that extra octal based scales now sit on top of the original hexadecimal based scales. Otherwise the slide rule works precisely as described in the patent for both octal and hexadecimal (but not mixed) based calculations. The recessed rather than flat back to the solid frame stock is strong clue that this rule was made by the American manufacturer of cheap plastic slide rules: *Sterling Plastics*.

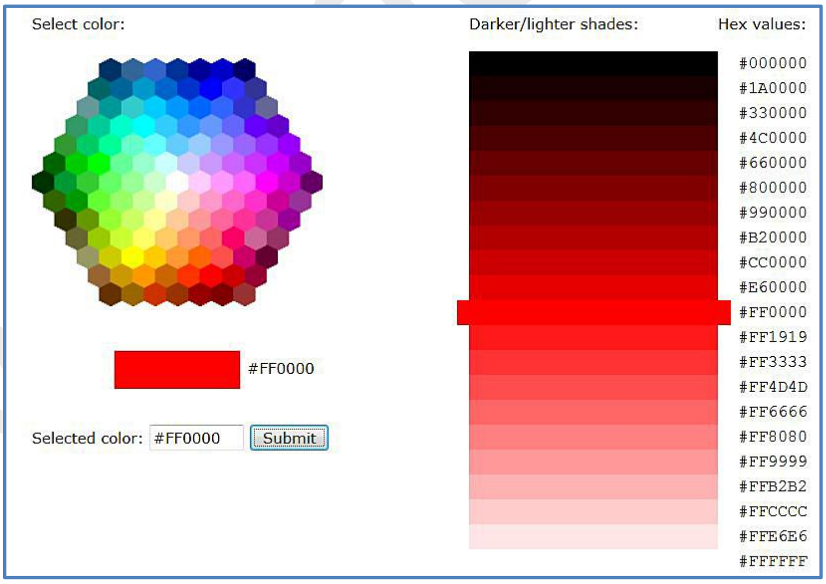
Superfluous – well almost!

Within a few years[5] of them coming onto the market, analogue hexadecimal computer programming aids became superfluous. It was a two-stage extinction process and only a few have survived. First several makers of electronic pocket calculators spotted the weakness in their offerings and developed models capable of non-decimal arithmetic. For example, the *Hewlett-Packard* landscape orientated (130 x 79 mm) *Programmable Computer Science* electronic pocket calculator that came out in 1982. Besides supporting the full range of hexadecimal, decimal, octal and binary arithmetic, the *HP16c* also had other built-in computer programming related functions. Significantly it was HP's first and only computer programmer's calculator. It was discontinued in 1989.

[5] Surprisingly a patent, JP57034273A, for a *Hexadecimal Slide Rule* was granted to Mitsubishi Electric Corp. as late as February 1982.

Then, as computer processing power became much cheaper and much faster, programming languages could ignore the inefficiency and use syntaxes and expressions more associated with how humans expressed what they wanted the computer to do. Examples of such modern-day "Esperanto" programming languages are JAVA, PHP (Hypertext Pre-processor) and PEARL (Process and Experiment Automation Real-time Language).

So in the New Millennium is the use of hexadecimal just for computing dinosaurs? The printing industry works to a high resolution and in minute variations of colour. Thousands of tints can be chosen from either the **CYMK** (Cyan, Magenta, Yellow and Key/Black) or **PMS** (Pantone Matching System) colour guides. By comparison normal computer screens are low resolution. So, for example, to "mix" and display the intended colour from an Internet web page just the 3 primary colours of **red**, **green** and **blue** are used. Each primary colour can also only be specified in one of 256 variances - i.e. in a range from 0 to 255. But even today computers still work in bits and bytes and not decimal. So each primary colour is often specified in hexadecimal rather than decimal – i.e. 00 to FF (#**RRGGBB**). For example, on a web page pure **red** is programmed as #FF0000 and pure **blue** as #0000FF. This means anno 2012 all that a computer programmer needs to work in hexadecimal is a suitable colour "picker".



So perhaps surprisingly *"hexadecimal lives on"* or put another way: 68 65 78 61 64 65 63 69 6d 61 6c 20 6c 69 76 65 73 20 6f 6e.

Acknowledgements & Bibliography

A special word of thanks goes to fellow collectors Timo Leipälä and David McFarland for sharing information and images of the analogue hexadecimal calculating aids in their collections. I am also grateful to Rod Lovett for pointing out the modern-day programming use of hexadecimal colour pickers.

1. **Poelje, Otto van**: *"The Demise of the Slide Rule (and the Advent of its successors)",* JOS Vol. 12 No. 2, Fall 2003, ISSN 1061-6292, Pg. 49.
2. **Thomas, Alice:** for providing the *"History of Programming Languages"* poster, O'Reilly Media, Farnham, UK.
3. **Tout, Nigel**: online *"Vintage Calculators Web Museum"*, http://www.vintagecalculators.com/index.html .
4. **Otnes, Robert**: *"American Small Adders Bonham, Hook and Pangborn"*, Proceedings 7th International Meeting of Slide Rule Collectors, September 2001, Germany, Pg. 71.
5. **Rance, David**: *"Reckon on it! A very British trait"*, Proceedings 15th International Meeting of Slide Rule Collectors, September 2009, Germany, Vol. 1, Pg. 65.
6. **Leipälä, Timo**: *"Kummer and his 1847 slide adder"*, Proceedings 15th International Meeting of Slide Rule Collectors, September 2009, Germany, Vol. 2, Pg. 65.
7. **Espacenet**: Internet user interface by the European Patent Office - http://worldwide.espacenet.com/advancedSearch?locale=en_EP
8. **Weiss, Stephan**: *"Non-decimal Slide Adders and the Carry"*, JOS Vol. 18 No. 2, Fall 2009 ISSN 1061-6292, Pg. 22.
9. **Riches, David**: *"Sterling Slide Adders"*, JOS Vol. 19 No. 2, Fall 2010, ISSN 1061-6292, Pg. 5.
10. **McFarland, David**: *"Lightning Subtraction"*, JOS Vol. 12 No. 1, Spring 2003, ISSN 1061-6292, Pg. 33.
11. **Dieselkamp, Friedrich**: *"Hex Adder"*, *"IBM Hexadecimal Adder"*, and *"Kugelrechner sedezimal"* online Great Encyclopedia of Mechanical Calculating – Rechnerlexikon, June & November 2005, June 2007.
12. **Otnes, Robert**: *"A Binary Based Slide Rule"*, JOS Vol. 12 No. 2, Fall 2003, ISSN 1061-6292, Pg. 60.
13. **Hicks, David**: online "Museum of HP Calculators", http://www.hpmuseum.org/ .
14. **Corban Works**: online *"ASCII to Hex ...and other free text conversion tools"*, http://www.asciitohex.com/ Copyright © 2011.