

On the Subject of Base Conversion

This is in base 4... and this one is in base 12... and this is... in base 36!?

This module is composed of three displays and keys with base-36 numeric input.

In each of the three displays, the following information is shown: 'From which base to convert' 'the original number' and 'the converted number entered.'

To disarm, you need to correctly input the converted value and then press the third display.

Order of release

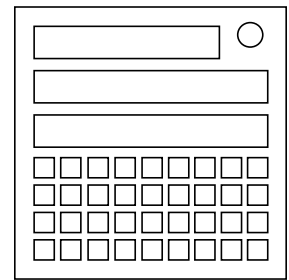
Convert the second display value to decimal using the base indicated by the first number*.

Convert the result of the first conversion to the base indicated by the second number**.

Input the converted value.

Press the third display.

Please refer to the Appendix for instructions on "Converting to decimal" and "Converting to base n".



*First number: The value displayed on the left side of the first display.

**Second number: The value displayed on the right side of the first display.

Appendix: Converting to decimal

No matter what base we use, everyone says we're using decimal.

Conversion Procedure

Each value needs to be converted to decimal using the base indicated in the right table.

The calculation formula is: "Value at nth digit × Base^(number of digits - n)".

Perform this operation for all digits, and the sum will be the decimal value.

Note

- When the exponent is 0, the result is always 1 regardless of the base.
- After analyzing the module, it was determined that the displayed value falls within the range of ulong range*.

*ulong range: A numeric type in C# with values from 0 to 4294967295.

| | | | |
|---|----|---|----|
| 0 | 00 | I | 18 |
| 1 | 01 | J | 19 |
| 2 | 02 | K | 20 |
| 3 | 03 | L | 21 |
| 4 | 04 | M | 22 |
| 5 | 05 | N | 23 |
| 6 | 06 | O | 24 |
| 7 | 07 | P | 25 |
| 8 | 08 | Q | 26 |
| 9 | 09 | R | 27 |
| A | 10 | S | 28 |
| B | 11 | T | 29 |
| C | 12 | U | 30 |
| D | 13 | V | 31 |
| E | 14 | W | 32 |
| F | 15 | X | 33 |
| G | 16 | Y | 34 |
| H | 17 | Z | 35 |

Appendix: Converting to base n

The world is vast. Binary, Duodecimal, and Hexadecimal are familiar. Surely there are other numeral systems too.

Conversion Procedure

Divide the value by the indicated base.

The calculation formula is: "Value ÷ Base".

Repeat this operation until division is no longer possible.

The result of converting the remainders in reverse order* (as indicated in the right table) will be the value in that base.

Note

- When performing division, it's necessary to find the remainder.
- The value obtained from repeated divisions until it's no longer possible is also considered a remainder.

*reverse order: For example, if the original number is 123, the reverse order would be 321.

| | | | |
|---|----|---|----|
| 0 | 00 | I | 18 |
| 1 | 01 | J | 19 |
| 2 | 02 | K | 20 |
| 3 | 03 | L | 21 |
| 4 | 04 | M | 22 |
| 5 | 05 | N | 23 |
| 6 | 06 | O | 24 |
| 7 | 07 | P | 25 |
| 8 | 08 | Q | 26 |
| 9 | 09 | R | 27 |
| A | 10 | S | 28 |
| B | 11 | T | 29 |
| C | 12 | U | 30 |
| D | 13 | V | 31 |
| E | 14 | W | 32 |
| F | 15 | X | 33 |
| G | 16 | Y | 34 |
| H | 17 | Z | 35 |