(12) UK Patent Application (19) GB (11) 2 382 420 (13) A

(43) Date of A Publication 28.05.2003

(21) Application No 0128042.9

(22) Date of Filing 22.11.2001

(71) Applicant(s)
UbiNetics Limited
(Incorporated in the United Kingdom)
Cambridge Technology Centre,
MELBOURN, Hertfordshire, SG8 6DP,
United Kingdom

(72) Inventor(s)
Robert Larkin

(74) Agent and/or Address for Service
Withers & Rogers
Goldings House, 2 Hays Lane, LONDON,
SE1 2HW, United Kingdom

(51) INT CL[7]
G06F 7/72

(52) UK CL (Edition V)
G4A ACV

(56) Documents Cited
None

(58) Field of Search
UK CL (Edition T) **G4A** AAU ACV ACX
INT CL[7] **G06F** 7/48 7/552 7/556 7/72
Other: **ONLINE: WPI, EPODOC, JAPIO, INSPEC**

(54) Abstract Title
METHOD AND APPARATUS FOR DETERMINING THE MODULO OF NON-POWER OF TWO NUMBERS

(57) When calculating a mod-b, the apparatus includes an excess calculator arranged to calculate the next power of two higher than b to give an estimate of the number of wraps and to shift a, in binary, to the right by the estimated number of wraps to derive an excess. A subtractor is arranged to subtract the product of b and the excess from a to derive the result.

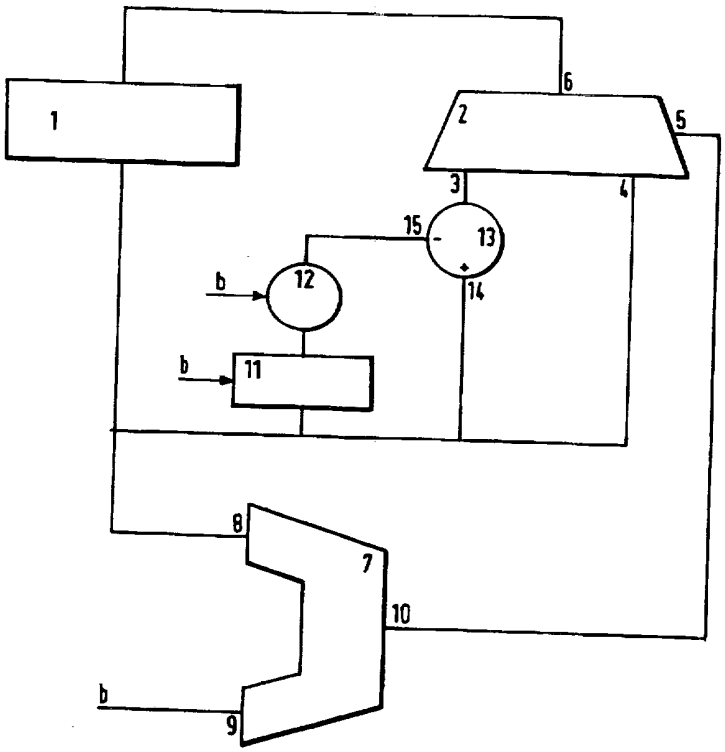

Fig.1.

GB 2 382 420 A

At least one drawing originally filed was informal and the print reproduced here is taken from a later filed formal copy.

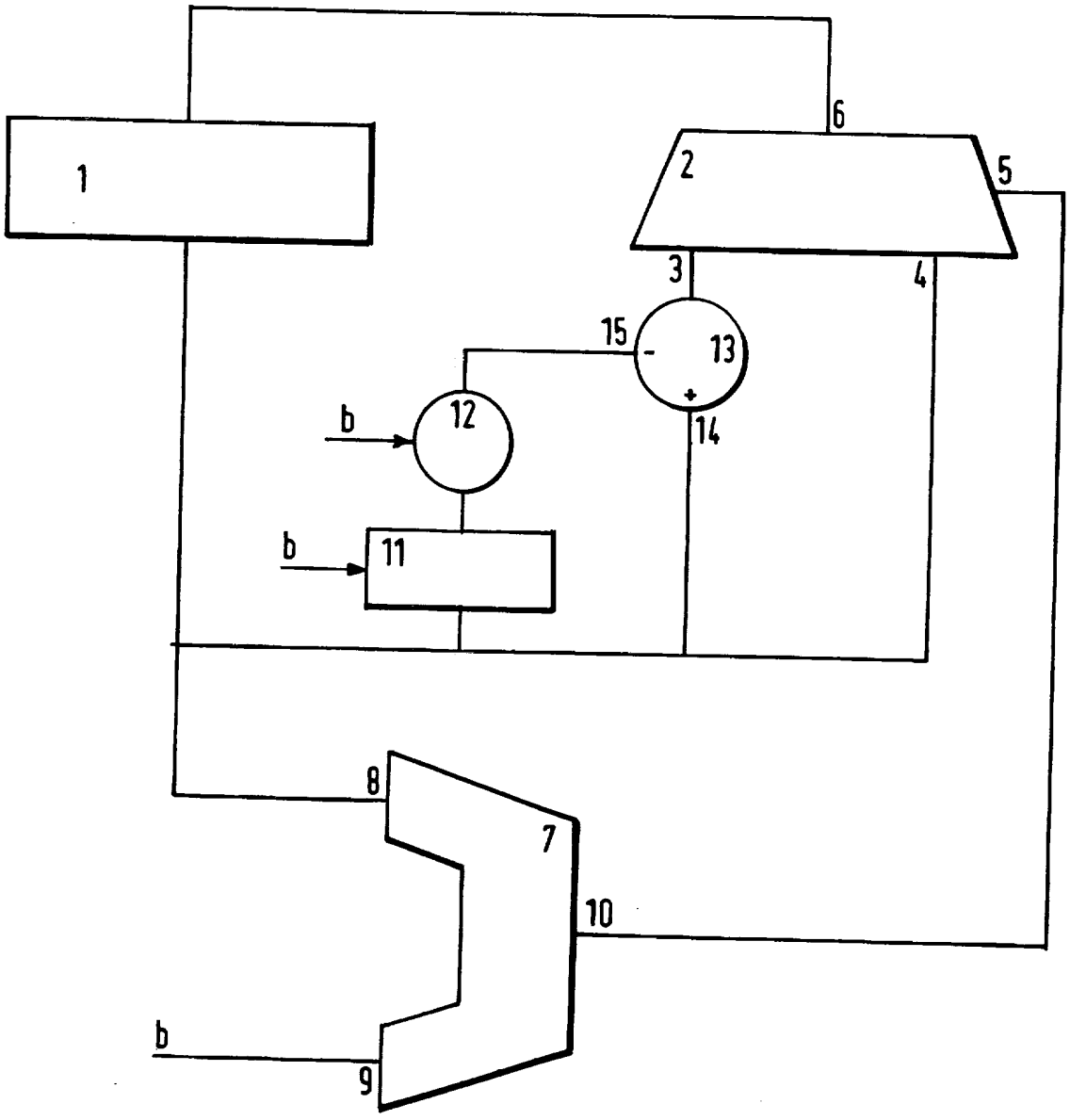

Fig.1.

Method and Apparatus for Determining the Modulo of Non-Power of Two Numbers

The present invention relates to a method and to an apparatus of determining the modulo of non-power of two numbers.

It is important to be able to calculate the modulo of numbers in 3G mobile telecommunications systems in order to calculate the turbo interleaver pattern in the BRP. It is intended that a single reusable functional block is used to perform any such calculation.

a modulo b (a mod b) is defined as the value of a counting modulo b, which is equivalent to the remainder of a divided by b. In a binary number system, where b is a power of 2, mod b can be found by taking the $\log_2$ (b) least significant bits (LSB) of the number.

For example:

$a = 103_{10} = 1100111_2$

$b = 32_{10} = 2^5 = 2^n$

$n = 5_{10}$ (= $\log_2$ (32))

the n least significant bits of 1100111_2 are $00111_2 = 7_{10}$

7_{10} is the result of 103 mod-32

As will be realised, where b is a power of two, the calculation is very simple. However, where it is not, the calculation is currently rather more difficult. One method that is known is a conditional iterative substraction where b is subtracted from a until a is less than b. Another known method involves dividing b by a, and multiplying the remainder by a to find the result. Depending on the value of the two input values or arguments, the time needed to complete all of the iterations is variable and can cause a long delay

in certain cases. A speed up method can be introduced by subtracting a multiple number of the b argument at each iteration. The best speed up method would be to subtract the exact number of wraps, n (i.e the integral result of a divided by b). This would give the result in one iteration. The present invention moves significantly towards this aim.

According to a first aspect of the present invention, modulo calculating apparatus arranged to calculate a mod-b comprises an excess calculator arranged to calculate the next power of two higher than b to give an estimate of the number of wraps, and to shift a, in binary, to the right by the estimated number of wraps to derive an excess; and a subtractor arranged to subtract the product of b and the excess from a to derive the result.

According to a second aspect, a method of calculating a mod-b comprises calculating the next power of two higher than b to give an estimate of the number of wraps; shifting a, in binary, to the right by the estimated number of wraps to derive an excess; and subtracting the product of b and the excess from a to derive the result. This method is designed to get close to the exact number of wraps, but ensures that the estimation of the number of wraps does not allow the subtraction step to undershoot the correct number of wraps. This method allows a significant speeding up of the calculation in most cases. Over a number calculations, this method will give a significant performance improvement by allowing calculation of the turbo interleaver pattern in much less time.

Other preferred features are recited in the claims.

Embodiments of the present invention will be described below, with and without reference to Figure 1, which shows an example of a logic circuit for implementing this invention.

In order to understand the invention, the method best described in the following way. The number of wraps is estimated by dividing a not by b, but by N, where N is the next power of 2 up from b. As N is larger than b, the subtraction is guaranteed not to undershoot. Certain end conditions need to be checked, but can be performed with simple logic.

A piece of pseudo-code is listed below to demonstrate the principle;

```
/* Accelerated Modulo of odd (non power of two) number */

/* c = a mod-b */
function: a = mod (a,b)

n_e = next_power_of_two (b);

while (b < a)

  /* shift a, n_e places to the right */
  excess = binary_shift (a,n_e);

  /* AND with "01111..1" */
  if (exc > 1)
  a = a - (b * excess);
  else

    /* end condition */
    a = a -  b;
  end

end
return a

/* end of pseudo-code */
```

In order to illustrate this, below I set out some worked examples.

Example 1

a mod - b

where a = 13_{10} = 1101_2

b = 9_{10} = 1001_2

2^{ne} = next power of two above b = 16_{10} = 2^4

n_e = 4

shift 1101_2 (a=1101_2) four (n_e=4) bits to the right

excess = 0

is excess >1 ? No

result is a_{new} = a - b = 13 - 9 = 4

From this example, it will been seen that the first step is to calculate n_e which is the estimated number of wraps. This is found to be 4. a in binary is shifted 4 bits to the right. This leaves a number, which in this case is 0, and that is the excess. Since the excess is not greater than 1, the result is obtained by subtracting b from a to get the result 4.

Example 2

a mod - b

a = 10_{10} = 1010_2

b = 3_{10} = 0011_2

2^{ne} = next power of two above b = 4_{10} = 2^2

n_e = 2

shift 1010_2 (a=1010_2) by two (n_e= 2) bits to the right

excess = 10_2 = 2_{10}

Is excess >1? yes, so a_{new} = a - (b x excess)

= 10_{10} - (3_{10} x 2_{10})

= 4_{10} = 100_2

shift 100_2 ($a_{new} = 100_2$) by two ($n_e = 2$) bits to the right

excess = $1_2 = 1_{10}$

is excess >1 ? No

result is $a_{new} = a_{new} - b$

$= 4 - 3$

$= 1$

In this example, the estimated number of wraps is found to be 2 since the next power of 2 above b is 4. a is then shifted 2 bits to the right, the shift corresponding to the estimated number of wraps to give an excess of 2. Since the excess is greater than one, a calculation is done to multiply b by the excess, the product of which is subtracted from a to give a value a_{new} of 4. a_{new} is then shifted by 2 bits ($n_e = 2$) bits to the right to give an excess of 1. Since the excess is not greater than one, the result is a_{new} - b, which = 1.

The invention can be applied to the generation of interleaving patterns for turbo interlevers in a 3G telecommunication system. It allows for the operation to be completed with less latency and less complexity than with conventional methods. The invention first finds an approximation of the number of wraps and then evaluates the end conditions to allow for accurate results of all possible input combinations.

In this method, the excess must always be at least 1 since errors can occur if it is zero.

The invention can be carried out by a computer which is appropriately programmed. Alternatively, an apparatus can carry it out. Such an apparatus will now be described for carrying out the method above.

Referring to Figure 1, a circuit includes a register 1 for holding the value of a, and a multiplexor 2 having a first input 3, a second input 4, a control input 5 through which the multiplexor 2 can be controlled to select the first input 3 or the second input 4, and output 6. A comparitor 7 has a first input 8 which receives the contents of register 1, a

second input 9 which receives the value of b and an output 10 which sends a control signal to the control input 5 of the multiplexor 2. The output 6 of the multiplexor 2 leads to an input to the register 1 whereby the value held in the register 1 can be changed. The signal leading to the first input 3 of the multiplexor 2 is processed through various logic components as will be explained below. The second input 4 to the multiplexor 2 comes from the output of register 1. Therefore, when the second input 4 is selected, the value within the register 1 is recirculated without allowing it to change.

Various logic components are included between the output of the register 1 and the first input 3 to the multiplexor 2. These include an excess calculator 11 which receives the contents of the register 1 through its input, and outputs the excess to a multiplier 12 which multiplies the excess with b. The output of the multiplier 12 is directed to a subtractor 13. The subtractor 13 includes a first input 14 for receiving the contents of the register 1, and a second input 15 for receiving the output of the multiplier 12. The subtractor 13 subtracts what is received from the second input 15 (the output of the multiplier 12) from what is applied to the first input 14 (the contents of register 1).

In operation, the value of a is placed in the register 1, and the value of b is applied to the second input 9 of the comparitor 7. The comparitor carries out the operation of comparison between the two, and if a is greater than b, then the control signal from the output 10 of the comparitor causes the multiplexor 2 to select the first input 3. The logic components leading to the first input 3 will apply a value corresponding to a_{new} which is then passed to the output 6 of the multiplexor so that it is placed in the register 1 in place the value of a. The value of a_{new} is then passed to the first input 8 of the comparitor 7 where, if a is no longer greater than b, a control signal is sent from the output 10 of the comparitor to the control input 5 of the multplexor 2 to select the second input 4 of the multiplexor 2 so as to recirculate the value held in register 1. This leaves the result of the calculation in register 1.

By way of example, Example 2 which is given above can be applied to this circuit. Firstly, the value of a is 10, and this is placed in the register 1 in binary form, 1010. The value of b, 3, is applied to the comparitor 7, to the excess calculator 11 and to the multiplier 12. The excess calculator 11 firstly identifies what the next power of 2 above b is, which is easy to calculate because in binary form it is identified by the bit which is the first 0 more significant than the most significant bit showing 1. Thus, n_e is found to be 2, and a is shifted by 2 bits to the right by the excess calculator, to give 10_2. This excess is then applied to the multiplier which multiplies the excess by b thereby applying the number 6_{10} to the second input 15 of the subtractor 13. The value of a is supplied to the first input 14 of the subtractor 13 from the register 1. The result of this subtraction is 4_{10}, and is applied to the first input 3 of the multiplexor 2. Since a is greater than b, the comparitor 7 controls the multiplexor 2 to select the first input 3. In this way, the result of the subtraction from the subtractor 13 is applied to the register 1 so that the number 4_{10} replaces the original value of a in the register. a_{new} is then applied to the excess calculator which again shifts the number 2 bits to the right giving an excess of 1. The 1 is multiplied by b by the multiplier 12, thereby applying the number 3 to the second input 15 of the subtractor 13. A subtraction takes place thereby applying the number 1 to the first input 3 of the multiplexor 2. This 1 is then put into the register 1. At this point the comparitor 7 calculates that the value applied to the first input 8 is less than the value of b applied to the second input 9, and so a control signal is applied to the multiplexor to select the second input 4 of the multiplexor 2, and this effectively recirculates the result of a, and the calculation is complete. The result of a mod b is left in register 1.

It should also be understood that, if the result of the calculation carried out by the excess calculator would output a value of zero, the number 1 is output because in certain circumstances, zero will give the wrong result. Therefore, the minimum output of the excess calculator is 1.

Claims

1. A method of calculating a mod-b comprising

calculating the next power of two higher than b to give an estimate of the number of wraps;

shifting a, in binary, to the right by the estimated number of wraps to derive an excess;

subtracting the product of b and the excess from a to derive the result.

2. A method according to claim 1, wherein after an excess is derived, the method further comprises comparing the excess with the number one, whereby, if the excess is greater than one:

(1) carrying out a calculation to reduce the value of a; and

(2) shifting a, in binary, to the right by the estimated number of wraps to derive a new excess value.

3. An electronic carrier bearing computer software arranged to carry out the method as claimed in claim 1 or claim 2.

4. Apparatus programmed to execute the method as claimed in claim 1 or 2.

5. Modulo calculating apparatus arranged to calculate a mod-b, comprising:

an excess calculator arranged to calculate the next power of two higher than b to give an estimate of the number of wraps, and to shift a, in binary, to the right by the estimated number of wraps to derive an excess; and

a subtractor arranged to subtract the product of b and the excess from a to derive the result.

6. Modulo calculating apparatus according to claim 5, further comprising a multiplier for multiplying b and the excess.

7. Modulo calculating apparatus according to claim 5 or 6, further comprising a register arranged to hold a.

8. Modulo calculating apparatus according to any one of claims 5 to 7, further comprising a multiplexor having two inputs, a first input of which is connected to the output of the subtractor, and a second input of which is connected to receive the value of a.

9. Modulo calculating apparatus according to claim 8 when dependent on claim 7, wherein the multiplexor includes an output which is connected to the input of the register to place the output of the multiplexor into the register.

10. Modulo calculating apparatus according to claim 8 or 9, further comprising a comparitor having first and second inputs, with the first input connected to receive a, and the second input connected to receive b.

11. Modulo calculating apparatus according to claim 10, wherein the comparitor includes an output connected to a control input of the multiplexor.

12. Modulo calculating apparatus according to claim 11, wherein, when a is greater than b, the comparitor is arranged to select the first input of the multiplexor, but otherwise to select the second input of the multiplexor.

13. Modulo calculating apparatus constructed and arranged substantially as described with reference to Figure 1.

INVESTOR IN PEOPLE

Application No: GB 0128042.9 **Examiner:** Peter Davies
Claims searched: 1 to13 **Date of search:** 18 July 2002

Patents Act 1977
Search Report under Section 17

Databases searched:

UK Patent Office collections, including GB, EP, WO & US patent specifications, in:

UK Cl (Ed.T): G4A (AAU, ACV, ACX) searched online

Int Cl (Ed.7): G06F (7/48, 7/552, 7/556, 7/72) searched online

Other: EPODOC, WPI, JAPIO, INSPEC

Documents considered to be relevant:

Category	Identity of document and relevant passage	Relevant to claims
	NONE	

X	Document indicating lack of novelty or inventive step	A	Document indicating technological background and/or state of the art.
Y	Document indicating lack of inventive step if combined with one or more other documents of same category.	P	Document published on or after the declared priority date but before the filing date of this invention.
&	Member of the same patent family	E	Patent document published on or after, but with priority date earlier than, the filing date of this application.