(51) International Patent Classification 6:
G06F

(11) International Publication Number: WO 99/66377

(43) International Publication Date: 23 December 1999 (23.12.99)

(21) International Application Number: PCT/FR99/00517

(22) International Filing Date: 14 June 1999 (14.06.99)

(30) Priority Data:
981388 15 June 1998 (15.06.98) FI

(71) Applicant (for all designated States except US):
ABB RESEARCH LTD. [CH/CH]; Alttofermstrasse 44, CH-8050 Zürich (CH).

(72) Inventors; and
(75) Inventors/Applicants (for US only):
WÄLJAS, Ilmari [FI/FI]; Maijankatu 9 A 4, FIN-33720 Tampere (FI); HÄMYNEN, Kimmo [FI/FI]; Kuunsäde 10 A 2, FIN-02210 Espoo (FI).

(74) Agent: KOLSTER OY AB; Isao Roobertinkatu 23, P.O. Box 148, FIN-00121 Helsinki (FI).


Published Without international search report and to be republished upon receipt of that report.

(54) Title: A METHOD FOR SYNCHRONISING PROCESS CONTROL EVENTS AND MEASUREMENTS IN A REAL-TIME PROCESS CONTROL AUTOMATION SYSTEM

(57) Abstract

The invention relates to a method for synchronising process control events and measurements in a real-time process control automation system comprising at least two units connected to each other by a communication channel, each unit having a real-time counter (RTC) and generating sampling pulses (SAMPLE_SYNC) for process control events and measurements, the method comprising the steps of synchronising the real-time counters of all units (SLAVE) but one to the real-time counter of said one unit (MASTER) by communication through the communication channel, and synchronising the generation of sampling pulses (SAMPLE_SYNC) in each of said units to the real-time counter (RTC) in the unit in question.

```
                MASTER
                  ↓
                  RTC
                   ▼
                    CCH

                  ▼  ▼
                  RTC  SLAVE

       ▼  ▼  ▼
       SAMPLE_SYNC  SAMPLE_SYNC
```
### FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

<table>
<thead>
<tr>
<th>AL</th>
<th>Albania</th>
<th>ES</th>
<th>Spain</th>
<th>LS</th>
<th>Lesotho</th>
</tr>
</thead>
<tbody>
<tr>
<td>AM</td>
<td>Armenia</td>
<td>FI</td>
<td>Finland</td>
<td>LT</td>
<td>Lithuania</td>
</tr>
<tr>
<td>AT</td>
<td>Austria</td>
<td>FR</td>
<td>France</td>
<td>LU</td>
<td>Luxembourg</td>
</tr>
<tr>
<td>AU</td>
<td>Australia</td>
<td>GA</td>
<td>Gabon</td>
<td>LV</td>
<td>Latvia</td>
</tr>
<tr>
<td>AZ</td>
<td>Azerbaijan</td>
<td>GB</td>
<td>United Kingdom</td>
<td>MC</td>
<td>Monaco</td>
</tr>
<tr>
<td>BA</td>
<td>Bosnia and Herzegovina</td>
<td>GE</td>
<td>Georgia</td>
<td>MD</td>
<td>Republic of Moldova</td>
</tr>
<tr>
<td>BB</td>
<td>Barbados</td>
<td>GH</td>
<td>Ghana</td>
<td>MG</td>
<td>Madagascar</td>
</tr>
<tr>
<td>BE</td>
<td>Belgium</td>
<td>GN</td>
<td>Guinea</td>
<td>MK</td>
<td>The former Yugoslav Republic of Macedonia</td>
</tr>
<tr>
<td>BF</td>
<td>Burkina Faso</td>
<td>GR</td>
<td>Greece</td>
<td>ML</td>
<td>Mali</td>
</tr>
<tr>
<td>BG</td>
<td>Bulgaria</td>
<td>HU</td>
<td>Hungary</td>
<td>MN</td>
<td>Mongolia</td>
</tr>
<tr>
<td>BJ</td>
<td>Benin</td>
<td>IE</td>
<td>Ireland</td>
<td>MR</td>
<td>Mauritania</td>
</tr>
<tr>
<td>BR</td>
<td>Brazil</td>
<td>IL</td>
<td>Israel</td>
<td>MW</td>
<td>Malawi</td>
</tr>
<tr>
<td>BY</td>
<td>Belarus</td>
<td>IS</td>
<td>Iceland</td>
<td>MX</td>
<td>Mexico</td>
</tr>
<tr>
<td>CA</td>
<td>Canada</td>
<td>IT</td>
<td>Italy</td>
<td>NE</td>
<td>Niger</td>
</tr>
<tr>
<td>CF</td>
<td>Central African Republic</td>
<td>JP</td>
<td>Japan</td>
<td>NL</td>
<td>Netherlands</td>
</tr>
<tr>
<td>CG</td>
<td>Congo</td>
<td>KE</td>
<td>Kenya</td>
<td>NO</td>
<td>Norway</td>
</tr>
<tr>
<td>CH</td>
<td>Switzerland</td>
<td>KG</td>
<td>Kyrgyzstan</td>
<td>NZ</td>
<td>New Zealand</td>
</tr>
<tr>
<td>CI</td>
<td>Côte d’Ivoire</td>
<td>KP</td>
<td>Democratic People’s Republic of Korea</td>
<td>PL</td>
<td>Poland</td>
</tr>
<tr>
<td>CM</td>
<td>Cameroon</td>
<td>KR</td>
<td>Republic of Korea</td>
<td>PT</td>
<td>Portugal</td>
</tr>
<tr>
<td>CN</td>
<td>China</td>
<td>KZ</td>
<td>Kazakhstan</td>
<td>RU</td>
<td>Russian Federation</td>
</tr>
<tr>
<td>CU</td>
<td>Cuba</td>
<td>LC</td>
<td>Saint Lucia</td>
<td>SD</td>
<td>Sudan</td>
</tr>
<tr>
<td>CZ</td>
<td>Czech Republic</td>
<td>LI</td>
<td>Liechtenstein</td>
<td>SE</td>
<td>Sweden</td>
</tr>
<tr>
<td>DE</td>
<td>Germany</td>
<td>LK</td>
<td>Sri Lanka</td>
<td>SG</td>
<td>Singapore</td>
</tr>
<tr>
<td>DK</td>
<td>Denmark</td>
<td>LR</td>
<td>Liberia</td>
<td>SI</td>
<td>Slovenia</td>
</tr>
<tr>
<td>EE</td>
<td>Estonia</td>
<td></td>
<td></td>
<td>SK</td>
<td>Slovakia</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SN</td>
<td>Senegal</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SZ</td>
<td>Swaziland</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>TD</td>
<td>Chad</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>TG</td>
<td>Togo</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>TJ</td>
<td>Tajikistan</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>TM</td>
<td>Turkmenistan</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>TR</td>
<td>Turkey</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>TT</td>
<td>Trinidad and Tobago</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>UA</td>
<td>Ukraine</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>UG</td>
<td>Uganda</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>US</td>
<td>United States of America</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>UZ</td>
<td>Uzbekistan</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>VN</td>
<td>Viet Nam</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>YU</td>
<td>Yugoslavia</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>ZW</td>
<td>Zimbabwe</td>
</tr>
</tbody>
</table>
A METHOD FOR SYNCHRONISING PROCESS CONTROL EVENTS AND MEASUREMENTS IN A REAL-TIME PROCESS CONTROL AUTOMATION SYSTEM

BACKGROUND OF THE INVENTION

The invention relates to a method for synchronising process control events and measurements in a real-time process control automation system comprising at least two units connected to each other by a communication channel, each unit having a real-time counter and generating continuous fixed rate sampling pulses for process control events and measurements.

In several power transmission applications, two different protection relays should be synchronised with each other for example in the distance relay function. Differential relay protection is another function where the measurements taken by different relays should be synchronised. The need for synchronisation is based on the basic function of these protection relays. It is very important in these relays to know in real time which one of the relays has operated first or first registered an abnormal change in the quantity to be monitored by the relay. By comparing the real-time stamps provided by the relays it is possible to conclude where in the power transmission line a failure or other abnormal situation has occurred. This information is also needed for correct operation of the relays.

It is typical of the relays mentioned above that, for example, A/D converter samplings of two separate devices are synchronised with each other but not normally to system real time. Such A/D converter units typically have one or more counters controlling A/D converters, interrupts and binary I/O, and a separate real-time (external battery supported) clock and a central processing unit (CPU) using the real-time clock to time stamp the samples taken by the A/D converters.

DISCLOSURE OF THE INVENTION

It is an object of the present invention to present a method for synchronising process control events and measurements in a real-time process control automation system e.g. for synchronising the sampling made by an A/D converter unit so that samples taken by the A/D converters are synchronised to real time and can have a real-time stamp which can be compared with samples taken by other corresponding units which are also
synchronised to a real-time clock. This object can be achieved with a method of the present invention which comprises the steps of

  synchronising the real-time counters of all units but one to the real-time counter of said one unit by communication through the communication

  channel by updating the real-time counter values of all said units but one by writing a new value to the real-time counter and/or by making an input

  frequency correction to the real-time counter and

  synchronising the generation of the continuous fixed rate sampling

  pulses in each of said units to the real-time counter in the unit in question.

  The communication channel used can be one of various known

  communication channels like Ethernet, Fieldbus or a modem connection.

  The real-time counters are preferably synchronised to each other

  with an accuracy leading to a time difference between the times of the real-

  time counters that is smaller than the interval between the generated sampling

  pulses. In practice the sampling interval can be e.g. somewhat smaller than

  400 µs, and the synchronisation between the real-time counters can lead to a

  time difference between the times of the real-time counters that is smaller than

  10 µs.

  According to a preferred embodiment of the invention the step of

  synchronising the generation of sampling pulses comprises the steps of

  setting a predetermined threshold value defining a time distance

  between successive timing pulses,

  comparing a value based on the current value of the real-time

  counter to a predetermined value and when the values match, loading said

  predetermined threshold value to a first counter,

  counting by said first counter the real-time clock pulses and, upon

  reaching said predetermined threshold value, giving a timing pulse,

  setting a predetermined sample distance value as a number of

  clock cycles to a first register,

  loading said predetermined sample distance value to a second

  counter,

  counting clock cycles by said second counter and upon reaching

  said predetermined sample distance value, giving a sampling pulse and

  resetting said second counter when said sampling pulse is given and when

  said timing pulse is received to restart the counting of the clock cycles by said

  second counter and
supplying a synchronised sampling pulse whenever said sampling pulse and said timing pulse are given, unless said timing pulse is given after a certain number of sampling pulses have already been given after a preceding timing pulse.

The invention is based on the idea that the sampling pulses are synchronised to the real-time counter by counting on one hand the clock pulses generated by a local oscillator and on the other hand the real-time clock pulses generated by the real-time clock. When the frequencies of the oscillators are known, the following can be calculated: a number of local oscillator clock pulses corresponding to a certain number of sampling pulses to be given by the counter generating sampling pulses and a number of real-time clock pulses covering the same time-distance as said number of clock pulses. By comparing the times when these numbers of clock pulses, on one hand, and the real-time clock pulses, on the other hand, are reached the sampling pulses given can be synchronised to the real-time clock at predetermined intervals.

When the method of the present invention is used to synchronise sampling in separate units each having a real-time counter it is preferred that the synchronisation is done equally in all units. This can be achieved by setting said predetermined value which is compared to a value based on the current value of the real-time counter to be the next real-time counter value evenly dividable with said predetermined threshold value defining the time-distance between successive timing pulses.

The timing of the sampling pulses given between the timing pulses based on the real-time clock can be made more accurate by setting a new sample distance value as a number of clock cycles to said second counter when resetting said second counter, said new sample distance value being based on comparison of the value of the second counter at the instance of giving said timing pulse to a predetermined value.

BRIEF DESCRIPTION OF THE DRAWINGS

The invention will now be discussed more in detail with reference to the attached drawings in which

Figure 1 is a simplified block diagram illustrating, by way of example, a system in which the method of the present invention can be utilised,
Figure 2 is an exemplary block diagram illustrating the steps of the method according to the present invention which relate to the generating and phasing of the timing pulses and

Figure 3 is an exemplary block diagram illustrating the steps of the method according to the present invention which relate to the generating of the synchronised sampling pulses based on clock pulses given by a local oscillator and on timing pulses created by the block diagram of Figure 2.

DETAILED DESCRIPTION OF THE INVENTION

Figure 1 is a simplified block diagram illustrating, by way of example, a system in which the method of the present invention can be utilised. This system comprises two units, MASTER and SLAVE, which both comprise a real-time counter RTC. In order that the real-time counter RTC in one unit can have the same real time as corresponding real-time counters in other corresponding units generating synchronised sampling pulses SAMPLE_SYNC they shall be synchronised to each other. This can be done in several known ways known for example from patent application PCT/FI98/00481, which is incorporated herein by reference, or from US Patent No. 4,185,110. To be more specific, the real-time counter of unit SLAVE is synchronised to the real-time counter of unit MASTER by communication through the communication channel by updating the real-time counter values of all the units SLAVE by writing a new value to the real-time counter and/or by making an input frequency correction to the real-time counter. The system can comprise any number of units SLAVE, but for simplicity only one is shown in Figure 12. Because the real-time counters can be synchronised in several known ways, the synchronisation will not be described in more detail in this context. However, it is important to note that the real-time counters are synchronised to each other with an accuracy leading to a time difference between the times of the real-time counters that is smaller than the interval between the generated sampling pulses SAMPLE_SYNC. In practice the sampling interval can be e.g. somewhat smaller than 400 µs and the synchronisation between the real-time counters can lead to a time difference between the times of the real-time counters that is smaller than 10 µs.

The communication channel CCH used between the units for synchronising the real-time counters RTC can be one of various known communication channels like Ethernet, Fieldbus or a modem connection.
Systems according to Figure 1 in which the method of the present invention can be implemented include industrial automation systems and power transmission systems. It is particularly suitable for continuous generation of synchronised sampling pulses for A/D converters which take samples continuously at a frequency over 1 kHz. Consequently, the method is used for measurements and for controlling continuous processes.

Figure 2 is an exemplary block diagram illustrating the steps of the method according to the present invention which relate to the generating and phasing of timing pulses SAMPLE_24. By the blocks of Figure 2 the timing pulses are synchronised to the frequency and phase of the real-time counter RTC. The oscillator of the real-time counter RTC can have a frequency of 1 MHz, for example. Thus, the cycle time of the pulses given by this oscillator of the RTC is 1 μs.

In the block diagram of Figure 2 the synchronisation of the timing pulses SAMPLE_24 is controlled by two registers, ADIFSDUS and ADIFRTCSVAL, that control the sample frequency and, correspondingly, the sampling phase of the timing pulses.

Register ADIFSDUS contains the distance between 24 samples in microseconds. This value is used to determine the correct sampling frequency, based on a 1 MHz clock RTC1M from the RTC. Periodically, according to ADIFSDUS setting, the sample start is generated according to the RTC clock. At the same time the sample distance in clock cycles of the local oscillator CLKADIF (not shown) is checked and adjusted by one clock cycle, if necessary, to match the two settings. This means that every 24th sample is initiated by the RTC clock, and the samples therebetween by CLKADIF with the adjusted sample period.

The sampling phase is controlled with the register ADIFRTCSVAL which contains a value corresponding to the 16 LSBs of the RTC. Phase adjustment starts when ADIFRTCSVAL is written. Synchronisation takes place when the RTC value (RTCVAL in the Figure) matches the value ADIFRTCSVAL. A status bit ADI FSPEND is set when ADIFRTCSVAL is written and cleared when the synchronisation takes place. ADIFRTCSVAL needs to be calculated and written only once. Thereafter the sampling stays in phase with the RTC.
The preferred way to select the ADIFRTCSVAL is to take the next RTC value evenly dividable with ADIFSDUS. This way the synchronisation can be done equally in all corresponding units. One possible formula to calculate the value in question is the following (note that the RTC has 64 bits):

\[
\text{ADIFRTCSVAL} = (\text{RTC} - (\text{RTC} \mod \text{ADIFSDUS}) + N \times \text{ADIFSDUS}) \mod 65536,
\]

where \( N \geq 2 \) is selected so that all the calculations can be performed before synchronisation.

The synchronisation accuracy can be calculated from the following formula, where \( \text{ERR}_{\text{RTC}} \) denotes the maximum error in the RTC value and \( \text{CLKADIF} \) denotes the ADC interface clock frequency:

\[
\text{ERR}_{\text{SAMPLE}} = \text{ERR}_{\text{RTC}} + \left(\frac{4}{\text{CLKADIF}}\right)
\]

For example, if the maximum RTC error is 3 microseconds, and the ADC Interface clock frequency is 19.6608 MHz, the max. sampling error from the correct moment is less than 4 microseconds.

The block diagram of Figure 2 is divided into two parts, one is the RTC microsecond down counter MDC / phase comparison, which operates with a clock CLKCPU (not shown) of the central processing unit (not shown) and the other is the sample timing block, which operates with CLKADIF. The MDC counter generates a SAMPLE_24 timing signal with intervals defined by the ADIFSDUS value. The SAMPLE_24 timing signal is used to synchronise the sample timing block shown in Figure 3 as described later in the text.

The microsecond down counter MDC counts down with every RTC1M pulse from the RTC. The counter is loaded with ADIFSDUS, when the count reaches 1 or when the phase synchronisation is done.

The phase synchronisation is done by writing to the ADIFRTCSVAL register, which is also indicated by signal WE-RTCSVAL. The writing also sets a status bit ADIFSPEND indicating that phase synchronisation is in progress. When the register setting ADIFRTCSVAL equals the RTCVAL from the RTC, the status bit ADISPEND is cleared and the counter MDC is reloaded with the ADIFSDUS value.

Whenever the counter MDC wraps around i.e. has counted down to 1 from the number of microseconds indicated by the ADIFSDUS value, a toggle flip flop TFF changes state. This change is detected with an edge
detector ED in CLKADIF clock domain, and a SAMPLE_24 signal is generated from each edge.

The circuitry shown in Figure 3 which can be called to a sample timing block generates a synchronised sampling signal SAMPLE_SYNC for an A/D converter, for example, 24 times in every SAMPLE_24 period. The sample distance timing is adjusted every SAMPLE_24 period to match the frequencies.

The sample timing block comprises two registers. One is the sample distance register SDR indicating the number of CLKADIF clock cycles in one SAMPLE_SYNC period. The other is a binary rate multiplier register BRM containing a fractional part of the sample distance (BRM / 24).

A sample distance counter SDC, which counts down from the value loaded to it from the SDR register, generates the SAMPLE signal whenever the counter reaches value 1. When the SAMPLE signal is given, the counter is also loaded with a new value from the sample distance register SDR. The SAMPLE_24 signal also loads the counter and thus synchronises the sample timing to SAMPLE_24 timing. Accordingly, whenever the SAMPLE_24 is received at the SDC, the SDC starts to count down from the new value given by the SDR register, without ending the count going on at that instance.

Consequently, in some occasions it does not give the sampling pulse SAMPLE which it otherwise would. This sampling pulse is generated by an output generation block OG to be described below.

The sample counter SC is used to count internally generated samples; every SAMPLE signal increases the count and SAMPLE_24 resets the counter. The output of the sample counter SC and the signals SAMPLE and SAMPLE_24 are supplied to the output generation block OG. The output generation block OG gives a synchronised sampling pulse SAMPLE_SYNC whenever it receives said sampling pulse SAMPLE and the timing pulse SAMPLE_24, unless the timing pulse SAMPLE_24 is received after a certain number of sampling pulses SAMPLE have already been given after a preceding timing pulse SAMPLE_24. In the embodiment described above the certain number of sampling pulses SAMPLE is 24.

According to the principles described above the timing pulse SAMPLE_24, which is tied to the real-time counter, is used as a pulse synchronising the generation of a synchronised sampling pulse at every 24th sampling pulse. The sampling pulses SAMPLE given between these 24th
pulses can however be slightly misplaced if the local oscillator runs fast or slow. Such cases are taken care of by the value of the sample counter SC together with the BRM register value. These values are used to control the BRM (binary rate multiplier) logic, which increases by the block INC the sample distance counter SDC load value for as many local oscillator clock pulse periods as the BRM register indicates. The increases are evenly spread over the SAMPLE_24 period. This enables a more precise control of the sample period.

Every time the SAMPLE_24 signal is given, the sample counter SC and the sample distance counter SDC values are examined, and compared with the correct values at the evaluation block EB. If the values are lower than they should, the sample period should be decreased. If the values are higher than they should, the sample period should be increased. The increment/decrement is done by first increasing/decreasing the BRM register value. If the BRM value is going to over/underflow, the sample distance register is increased/decreased.

With the BRM fractional correction, the precision of the adjustment is better than 2 CLKADIF periods. Another source of uncertainty in the sampling instance is the interface between CLKCPU and CLKADIF. The maximum error in the edge detection done by the edge detector ED is always 1 CLKADIF cycle. So the maximum error in sampling is less than 3 CLKADIF periods + the error in the RTC value itself.

Power transmission systems use several frequencies of which three are the most common: 16 2/3 Hz, 50 Hz and 60 Hz. In order that the A/D converter unit according to the invention could be used in all power transmission systems having one of the above mentioned frequencies, the number of synchronised sampling pulses per period is preferably the same for all these frequencies. In one example according to the invention the number of samples taken per period is 128. In order that this number of samples per period can be used, the local oscillator CLKADIF can have e.g. a frequency of 9.8304 MHz. As can be concluded from this, the sample distance counter SDC must have a different counting limit or threshold for every different power transmission frequency. In this example where the number of sampling pulses per period is 128 the counting limits are 4608 for the frequency of 16 2/3 Hz, 1536 for 50 Hz and 1280 for 60 Hz. With these counting limits the number of sampling pulses is always 128 for one period independent of the power
transmission line frequency. In the method of the present invention the correct
counting limit or threshold as a number of local oscillator clock pulses is first
supplied to the sample distance register SDR in accordance with the line
frequency used in the application in question. As explained in detail above, the
timing of the sampling pulses SAMPLE is synchronised with the timing pulses
SAMPLE_24 after 24 sampling pulses, the synchronised sampling pulses
SAMPLE_SYNC thereby being correctly synchronised 16 times in three line
frequency periods.

In the above the invention has been described with reference to
block diagrams giving only one example of the logic that can be used to
implement the method of the invention. It is therefore self-evident that the
diagrams could be modified in many ways without, however, deviating from the
method as defined in the enclosed claims. The shown embodiment and other
similar hardware solutions for implementing the method of the present
invention are preferred because they can be easily integrated into an
integrated semiconductor chip.
CLAIMS

1. Method for synchronising process control events and measurements in a real-time process control automation system comprising at least two units connected to each other by a communication channel, each unit having a real-time counter (RTC) and generating continuous fixed rate sampling pulses (SAMPLE_SYNC) for process control events and measurements, characterised in that the method comprises the steps of

   synchronising the real-time counters of all units (SLAVE) but one to the real-time counter of said one unit (MASTER) by communication through the communication channel by updating the real-time counter values of all said units (SLAVE) but one by writing a new value to the real-time counter and/or by making an input frequency correction to the real-time counter and synchronising the generation of the continuous fixed rate sampling pulses (SAMPLE_SYNC) in each of said units to the real-time counter (RTC) in the unit in question.

2. A method according to claim 1, characterised in that the real-time counters are synchronised to each other with an accuracy leading to a time difference between the times of the real-time counters that is smaller than the interval between the generated sampling pulses (SAMPLE_SYNC).

3. A method according to claim 1 or 2, characterised in that the step of synchronising the generation of sampling pulses (SAMPLE_SYNC) comprises the steps of

   setting a predetermined threshold value (ADIFSDUS) defining a time-distance between successive timing pulses (SAMPLE_24),

   comparing a value (RTCVVAL) based on the current value of the real-time counter to a predetermined value (ADIFRTCSVAL) and when the values match, loading said predetermined threshold value to a first counter (MDC),

   counting by said first counter (MDC) the real-time clock pulses and, upon reaching said predetermined threshold value (ADIFRTCSVAL), giving a timing pulse (SAMPLE_24),

   setting a predetermined sample distance value as a number of clock cycles to a first register (SDR),

   loading said predetermined sample distance value to a second counter (SDC),
counting clock cycles by said second counter (SDC) and, upon reaching said predetermined sample distance value, giving a sampling pulse (SAMPLE) and resetting said second counter when said sampling pulse is given and when said timing pulse (SAMPLE_24) is received to restart the counting of the clock cycles by said second counter (SDC) and supplying a synchronised sampling pulse (SAMPLE_SYNC) whenever said sampling pulse (SAMPLE) and said timing pulse (SAMPLE_24) are given, unless said timing pulse (SAMPLE_24) is given after a certain number of sampling pulses (SAMPLE) have already been given after a preceding timing pulse (SAMPLE_24).

4. A method according to claim 3, characterised in that it further comprises the step of setting said predetermined value (ADIFRTCSVAL), which is compared with a value (RTCVAL) based on the current value of the real-time counter, to provide the next real-time counter value evenly divisible with said predetermined threshold value (ADIFSDUS) defining the time-distance between successive timing pulses (SAMPLE_24).

5. A method according to claim 3, characterised in that it further comprises the step of setting a new sample distance value as a number of clock cycles to said second counter (SDC) when resetting said second counter, said new sample distance value being based on the comparison of the second counter value with a predetermined valued at the moment when said timing pulse (SAMPLE_24) is given.