A High Accuracy Multi-rate Implementation of Resonant Integrator Using FPGA

Anirban Ghoshal\textsuperscript{1*}, Vinod John\textsuperscript{2}

\textsuperscript{1,2} Department of Electrical Engineering, IISc Bangalore, Bangalore, India.

* ghoshal.anirban@gmail.com

Abstract

Second Order Generalised Integrator or Resonant Integrator (RI) has wide range of applications. Forward and backward Euler’s approximation based two integrator realisation of RI is an easily implementable frequency adaptive method. However, it suffers from resonant frequency deviation due to discretization. The discretization methods that lead to accurate realisation of RI require online calculation or look up table of trigonometric functions to accommodate frequency variation. In this paper multi-rate computation based implementation of two integrator based RI has been proposed to minimize resonant frequency deviation. In this method no additional logic elements are consumed to achieve accurate resonant frequency location. This along with down-sampling leads to lesser phase lag of RI. The effect of quantization on resonant frequency deviation has been analysed for proper choice of calculation time. It is also shown that appropriate choice of down-sampling instants give a range of phase response characteristics around the nominal continuous time RI phase response. The accuracy of resonant frequency emulation has been experimentally verified by implementing a proportional resonant controller as current controller.

Index Terms

Resonant integrator, PR controller, Digital control, FPGA, Current control.

I. INTRODUCTION

Present literature show application of RI in fundamental and harmonic current control [1]–[9], phase locked loop [10]–[13], waveform filtering [14], [15] and active damping [16], [17]. A practical application of grid connected VSI such as active filtering where several harmonics are to be compensated, may have a large number of RI working simultaneously [2], [3], [8], [9]. Therefore, meeting desired performance criteria and efficient digital implementation of RI are important.

For digital implementation, continuous domain transfer function of RI is to be mapped in discrete domain. There are several methods of discretization for RI. Their relative merits and demerits are discussed in [5]. However the two preferred methods of digital implementation are Two Integrator method with Forward and Backward (TIFB) Euler approximations [5], [9], [18] and Tustin with prewarping (TP) [5], [19], [20]. The TIFB method can be made frequency adaptive easily [5], [9]. This method has problem of resonant frequency shift [5], [6], [9] due to both discretization and quantization. Although the other methods are better in terms of resonance frequency shift but number of computations increase compared to TIFB method. The digital implementation of RI should address the
TABLE I

<table>
<thead>
<tr>
<th>Discretization Method</th>
<th>Addition/Multiplication</th>
<th>Division</th>
<th>Trigonometric Functions</th>
<th>Storage</th>
</tr>
</thead>
<tbody>
<tr>
<td>TPΔ [6]</td>
<td>6</td>
<td>6</td>
<td>1</td>
<td>Present</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>4 + LUT</td>
</tr>
<tr>
<td>PFR [6]</td>
<td>4</td>
<td>7</td>
<td>1</td>
<td>Present</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>6 + LUT</td>
</tr>
<tr>
<td>TIFB [9]</td>
<td>3</td>
<td>4</td>
<td>0</td>
<td>Not Present</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>6</td>
</tr>
<tr>
<td>Modified TIFB [9]</td>
<td>4</td>
<td>7</td>
<td>0</td>
<td>Not Present</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>6</td>
</tr>
</tbody>
</table>

issue of utilisation of available resources [5], [6], [9] also. Consumption of computational resources, computation time, ease of implementation, phase response, resonance frequency accuracy and frequency adaptation should be considered during digital realization of RI.

The methods available in literature mostly use DSP as controller platform [5], [6], [19], [20]. These methods can also be implemented in potential alternative platforms like FPGA [22]–[26]. Also combined platform of FPGA and DSP can be attractive for power electronic control [27]–[30]. Large numbers of RI can be implemented in digital platform like FPGA or FPGA+DSP where multi-rate and parallel computation can easily be performed [27].

The possibility of multi-rate implementation of RI utilising TIFB method has been proposed in this paper with an analysis of the effect of multi-rate calculation. The proposed method utilises multi-rate calculation followed by down-sampling and does not consume additional resources compared to the original TIFB method of RI implementation. In this work the individual and combined effect of discretization and quantization error [6], [19]–[21] on resonant frequency shift have been analysed. This analysis indicates suitable design choice of the multi-rate calculation time for RI. This implementation has good phase characteristics giving sufficient stability margin in a closed loop operation. The analysis of multi-rate calculation followed by down-sampling shows that a range of phase responses around the nominal continuous time phase response can be obtained by proper selection of down-sampled instant of the output of RI. Experimental results, validating the theoretical understandings are presented in this paper in an application where RI is used for current control.

II. A COMPARISON AMONG DIFFERENT DIGITAL IMPLEMENTATION OF RI

Discretization methods play an important role over selection of possible digital implementation [5], [13], [29]. In case of RI, a few discretization methods show clear advantages over the others as reported in [5], [6], [9], [19], [20], [29]. The methods that accurately represent the resonant pole location require trigonometric functions to be calculated either from a lookup table or in an online manner, if resonant frequency variation is to be accommodated [5], [6], [9].

Discretization of RI by TP method is generally followed by a delta domain transformation on the discretized transfer function for minimization of quantization noise [6], [19], [20]. The changed coefficients after delta domain transformation contain trigonometric expressions as a function of resonant frequency $\omega_0$ and for some cases $\omega_0$ is...
present in denominator [6], [19]. This indicates possible requirement of online calculation of trigonometric functions and division operation in case resonant frequency adaptation is necessary [6], [19]. It is suggested in [19] that for a system with high sampling frequency the coefficients with trigonometric functions can be approximated such that frequency adaptation becomes simple and division operations are avoided. However for systems with low sampling frequency, generally occurs in high power converter where switching frequency is low, these approximations will be incorrect [19]. Hence for such cases resorting to online calculation of the coefficients of the transfer function discretized by TP is the remaining option which invariably results into higher computational burden [6].

The Polar Form Resonant (PFR) realisation of the RI is an alternative method [6] applicable for both high and low sampling period and results into a better digital realisation compared to TP method followed by delta domain transformation in terms of quantization noise reduction and frequency adaptation. However it is achieved with increased computational effort [6].

Compared to TP and PFR methods, digital implementation of RI by TIFB method does not require online calculation of trigonometric function or lookup table and frequency adaptation is easy [5], [9], [13]. The problem of pole shift due to discretization by TIFB, which results into changed resonant frequency, becomes significant in case the ratio of calculation frequency to resonant frequency is small [5], [9]. The method adopted to emulate resonant pole locations more accurately in [9] employs higher order Taylor series approximation and it is a modified version of the TIFB method. But this also increases the number of computations to be performed.

Result of a comparative study on these digital implementations of RI, including frequency adaptation, is tabulated in Table-I. The ‘TPΔ’ method, discretization by TP followed by delta domain transformation, and PFR method both require division and calculation of trigonometric functions [6], [19] which is not required for TIFB method. Also the number of summation and multiplication operations are higher for these methods compared to TIFB method, shown in Table-I. The modified TIFB method chosen for comparison is based on fourth order approximation of Taylor series expansion of \(\cos(\omega_0 T)\) as described in [9]. It requires higher number of calculations to maintain resonant frequency accuracy. Hence it can be concluded that from resource consumption point of view TIFB method is best suited provided the required performance is achieved, which is dependent significantly on resonant frequency accuracy. The proposed implementation saves 3 multiplications, 1 additions and 1 storage element with respect to that based on 4th order approximation. An application where number of RI transfer function is high [2], [3], [8], [9], [13], the cumulative savings is significant. Successful implementation of multiple RI for ‘resonant current control’ using two integrator method is described in [9] and example of using RI for PLL, filtering and active damping are also available. Using the suggested multi-rate calculation to implement one single RI by TIFB method, higher accuracy can be achieved without additional resource consumption. This style of implementation can achieve a significant reduction of resource allocation for RI to achieve similar or higher accuracy compared to other methods. In this manuscript we have followed the above mentioned logic to present our findings.
III. DISCRETIZATION BY TIFB AND POLE LOCATION VARIATION

The block diagram structure of RI in ‘s’ domain along with a proportional part forming a PR controller is shown in Fig. 1(a). The s-domain transfer function of RI is given in (1).

\[ T_r = \frac{sK_{ir}}{s^2 + \omega_0^2} \]  

(1)

Here \( K_{ir} \) is the resonant gain and \( \omega_0 \) is the resonant frequency in rad/sec. The resonant pole locations shift due to discretization by TIFB method resulting in deviation from desired resonant frequency. If the desired resonant frequency is \( \omega_0 \) rad/sec then the discrete domain pole locations of RI are at \( e^{\pm j\omega_0 T} \), where \( T \) is the sampling and calculation period. The discrete domain representation of ideal denominator polynomial of \( T_r \) is given by (2).

\[ (z - e^{+j\omega_0 T})(z - e^{-j\omega_0 T}) = z^2 - 2z\cos(\omega_0 T) + 1 \]  

(2)

As reported in [5], the discretization methods that produce the denominator of (2) may need additional computation of trigonometric functions or suffer from poor phase performance. Discretization by TIFB method does not produce the denominator polynomial given by (2). However for the cases where resonant pole location variation is not significant it could be a preferred realisation because of its ease of implementation, adaptive nature towards frequency variation and better phase performance [5], [9]. After discretization by TIFB the difference equations of RI can be expressed as shown in (3).

\[ p(n + 1) = K_{ir}Te(n) + \omega_0 Tq(n) + p(n) \]

\[ q(n + 1) = -\omega_0Tp(n + 1) + q(n) \]  

(3)
TABLE II
PHASE RELATIONSHIPS OF THREE RI REALISATIONS.

<table>
<thead>
<tr>
<th>Phase</th>
<th>Equation</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\phi_{T_r}$</td>
<td>$\frac{\pi}{2}$</td>
<td>$\omega &lt; \omega_0$</td>
</tr>
<tr>
<td></td>
<td>$-\frac{\pi}{2}$</td>
<td>$\omega &gt; \omega_0$</td>
</tr>
<tr>
<td>$\phi_{T_a}$</td>
<td>$\frac{\pi}{2} - \frac{\omega T}{2}$</td>
<td>$\omega &lt; \omega_0'$</td>
</tr>
<tr>
<td></td>
<td>$-\frac{\pi}{2} - \frac{\omega T}{2}$</td>
<td>$\omega &gt; \omega_0'$</td>
</tr>
<tr>
<td>$\phi_{T_b}$</td>
<td>$\frac{\pi}{2} + \frac{\omega T}{2}$</td>
<td>$\omega &lt; \omega_0'$</td>
</tr>
<tr>
<td></td>
<td>$-\frac{\pi}{2} + \frac{\omega T}{2}$</td>
<td>$\omega &gt; \omega_0'$</td>
</tr>
</tbody>
</table>

A state space model can be obtained after rearranging the difference equations of (3) and is given by (4), (5) and (6).

\[
\begin{bmatrix}
p(n+1) \\
q(n+1)
\end{bmatrix}
= A_n \begin{bmatrix}
p(n) \\
q(n)
\end{bmatrix}
+ B_n e(n)
\]

(4)

Where,

\[
A_n = \begin{bmatrix}
1 & \omega_0 T \\
-\omega_0 T & 1 - \omega_0^2 T^2
\end{bmatrix}
\quad \text{and} \quad
B_n = \begin{bmatrix}
K_{ir} T \\
-\omega_0^2 T^2 K_{ir}
\end{bmatrix}
\]

(5)

\[
y(n) = C_n \begin{bmatrix}
p(n) \\
q(n)
\end{bmatrix}
+ D_n e(n)
\]

(6)

As ‘$p$’ gets computed inside a digital processor so ‘$p(n+1)$’ is available before actual commencement of the instant ‘$(n+1)T$’. Hence output ‘$y$’ can be computed from ‘$p(n+1)$’ also. If ‘$p(n)$’ is selected as output then the coefficients of output equation are $C_n = [1 \ \omega_0 T]$ and $D_n = 0$. The input-output relationship is given by (7).

\[
T_r = \frac{P(z)}{E(z)} = \frac{K_{ir} T (z^{-1} - z^{-2})}{1 - (2 - \omega_0^2 T^2) z^{-1} + z^{-2}}
\]

(7)

If ‘$p(n+1)$’ is selected as output then the coefficients of output equation are $C_n = [1 \ \omega_0 T]$ and $D_n = [K_{ir} T]$. The input output relationship for this case is given by (8).

\[
T_r = \frac{z P(z)}{E(z)} = \frac{K_{ir} T (1 - z^{-1})}{1 - (2 - \omega_0^2 T^2) z^{-1} + z^{-2}}
\]

(8)

The frequency response plots of $T_r$, $T_r^a$ and $T_r^b$ are shown in Fig. 1(b). The selected resonant frequency $f_0 = 250 \text{ Hz}$ and the sampling time for discretization $T = 250 \mu s$. A shift in the resonant frequency due to TIFB method is observable from the magnitude plot in Fig. 1(b). The phase response plot shows that $T_r^b$, where ‘$p(n+1)$’ is chosen as output, provides a lesser phase lag of ‘$\frac{\omega T}{2}$’ when compared to ‘$T_r^a$’, as shown in Fig. 1(b). Phase relationships of $T_r$, $T_r^a$ and $T_r^b$ are tabulated in Table-II. This indicates that $T_r^a$ in (8) would be a preferred realisation of the RI, when smaller phase lag is required, to compensate for plant non-idealities such as delay.
Comparison among denominators of (7) and (8) with the ideal denominator of (2) shows that the shift in resonance frequency occurs due to approximation $2 \cos (\omega_0 T) \simeq (2 - \omega_0^2 T^2)$. The shifted resonance frequency $\omega'_0$ can be calculated from $\cos (\omega'_0 T) = 1 - \frac{(\omega_0^2 T^2)}{2}$ and is given by (9).

$$\omega'_0 = \frac{2}{T} \sin^{-1}\left(\frac{\omega_0 T}{2}\right)$$  \hspace{1cm} (9)

An expression for per unit frequency deviation, $(\Delta f_D)_{pu}$, can be derived using (9) which is given by (10).

$$(\Delta f_D)_{pu} = \frac{f_0}{f_b} - \frac{1}{\pi f_0 T} \sin^{-1}(\pi f_0 T)$$  \hspace{1cm} (10)

Here $f_0$ is the desired resonant frequency and $f_b$ is base frequency. From (10) frequency deviation for different $f_0$ and $T$ can be calculated.

The fourth order Taylor series approximation of $\cos (\omega_0 T)$ which has been used for modified TIFB is given by (11).

$$\cos (\omega_0 T) \approx 1 - \frac{\omega_0^2 T^2}{2} + \frac{\omega_0^4 T^4}{24}$$  \hspace{1cm} (11)

The resource requirement is more if further higher order approximations are used for better accuracy as suggested by [9]. Therefore implementation of RI with TIFB method is of importance if greater accuracy can be obtained without consuming additional resources.

The effect of calculation period on frequency deviation or RI pole position variation for two approximations of Taylor series expansion of $\cos (\omega_0 T)$ is shown in the graph of Fig. 2(a). The second order approximation, marked as ‘×’ and utilised for original TIFB method, represents the denominator polynomial of (7) and (8) where the calculation time $T = 250 \mu s$. The fourth order approximation, marked as ‘∆’ and utilised by modified TIFB method, represents the denominator polynomial approximated by (11) with $T = 250 \mu s$. The plot marked as ‘□’ represents the denominator polynomial of (7) and (8) which are derived from discretization by original TIFB method.
### TABLE III
PARAMETERS FOR RI IMPLEMENTATION.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clock period</td>
<td>$T_c$</td>
<td>50 ns</td>
</tr>
<tr>
<td>Resonant Integrator calculation period</td>
<td>$T$</td>
<td>10 $\mu$s</td>
</tr>
<tr>
<td>Resonant frequency</td>
<td>$f_0$</td>
<td>250 Hz</td>
</tr>
<tr>
<td>Resonant gain</td>
<td>$K_{ir}$</td>
<td>50</td>
</tr>
<tr>
<td>Execution time</td>
<td>-</td>
<td>400 ns</td>
</tr>
<tr>
<td>PWM update period</td>
<td>$T_\nu$</td>
<td>250 $\mu$s</td>
</tr>
<tr>
<td>Switching frequency</td>
<td>$F_{sw}$</td>
<td>2.0 kHz</td>
</tr>
<tr>
<td>ADC sampling period</td>
<td>-</td>
<td>250 $\mu$s</td>
</tr>
</tbody>
</table>

However in this case the calculation time $T = 10\mu$s. The plot of Fig. 2(a) shows comparison of frequency deviation between these three options where the normalised frequency variation ($\Delta f_D$) due to discretization is plotted against desired resonant frequency. The plot clearly shows that the frequency deviations of second order approximation with $T = 10\mu$s, marked as ‘□’, is significantly less compared to other two realisations. For digital platforms such as DSP, selecting a lower value of ‘$T$’ may not be feasible due to constraints of available time for computation and switching frequency. However if the RI is implemented in a digital platform with ability to perform parallel and multi-rate computation, then it can be implemented without increasing total computation time and number of logic elements consumed. Here this multi-rate computation method is used to propose a simple and accurate implementation of RI in a FPGA.

### IV. FPGA IMPLEMENTATION OF RI

In this work multi-rate implementation of RI uses repeated calculation between two PWM updates without additional ADC sampling. The ADC sampling occurs at PWM updates. The multi-rate calculation method can be understood from the figure shown in Fig. 3(a). Here ‘$T_\nu$’ is sampling period, that is the time between two sampling instant $t[n]$ and $t[n + 1]$. From (3) - (6) the variables of RI are ‘$p$’, ‘$q$’, ‘$e$’, and ‘$y$’. The value of ‘$p$’, ‘$q$’, and ‘$y$’ at $(n + 1)^{th}$ sample are to be calculated using the values of the variables ‘$p$’, ‘$q$’, ‘$e$’, and ‘$y$’ at $n^{th}$ sample. The calculation time for RI is ‘$T$’. Now, in original TIFB method $T = T_\nu$. Therefore the calculation is carried out only once in original TIFB method. In the proposed method the calculation is carried out ‘$\nu$’ number of times where $\nu = \frac{T_\nu}{T}$. The calculation is done at a faster rate where $T < T_\nu$. The $n^{th}$ sample value of ‘$e$’, which is input to RI, is used for calculation within ‘$T_\nu$’ without any further sampling of input. As calculation time for rest of the system other then RI may be different than ‘$T$’ hence it can be called as a multi-rate system. In this case as the process output is again used as input for calculation so repeated calculation between successive sampling can give
more accurate final output, taken before the PWM update. Depending on necessity any \(i^{th}\) calculation can be used as output. This selection has effect on phase response of the resultant transfer function as discussed in section-VI.

For example if switching frequency is \(2.0 kHz\) and double sampling [31] is adopted then system sampling frequency is \(4kHz\). Hence sampling period \(T_v = 250\mu s\). So in every \(250\mu s\) the variables of RI gets calculated once. Now if the RI block is executed internally at every \(10\mu s\) then more accurate resonant pole locations would be emulated. This is achieved by running the integrators of the RI block at a faster rate inside the main control
loop, without increasing the system sampling rate. So for a 250\(\mu s\) sampling rate the variables of the RI would be calculated 25 times. This is easily implementable in a FPGA platform by increasing the calculation frequency of the RI block by an integer multiple without consuming additional resources.

Similar performance can also be achieved by utilising a Digital Signal Processor (DSP) platform. As commercially available DSPs for power electronics applications execute one instruction set at a time so the suggested method of RI implementation would increase the overall computation time. Hence the proposed method is particularly suitable for FPGA or computational platform with multi-core architecture.

An ALTERA cyclone-II series FPGA board is used for implementing RI and associated control blocks to perform experiments. The block diagram for RI implementation is shown in Fig. 3(b). The resonant frequency \(\omega_0\) can be made adaptive by linking it with the output of a frequency estimation algorithm. System clock frequency is selected as 20 \(MHz\). From this clock signal, required clock enable signals for sequential operation are generated. Eight clock enable signals are required to perform the entire computation. Width of each clock enable signal equals to one clock period, which is 50\(ns\), shown in Fig. 3(b). These signals are synchronized as shown in Fig. 3(b). The calculation process takes 400 \(ns\) to complete and it waits upto 10\(\mu s\) to start again. The calculations have been performed using fixed point arithmetic format with 16 bit word length. However the integration processes are carried out with 32 bit word size for better accuracy. The storage elements for integrators are termed as DFF in Fig. 3(b).

Parameters for RI implementation in the FPGA are given in Table-III. If delay compensation is necessary then standard method of compensation [2], [9], shown in Fig. 3(b) can be adopted.

V. EFFECT OF QUANTIZATION ERROR

Choosing a fast rate of computation reduces the shift in resonant frequency due to discretization method but a fast computation rate with limited word length can introduce significant deviation from the actual resonant frequency due to quantization error [6], [19]. It can be observed from Fig. 3(b) that quantization error can occur in four places for the chosen method of implementation. In three of them it is due to transformation of a 32 bit number to a 16 bit number. The error introduced due to these would be negligibly small because the least significant bit is affected only. For the remaining case where calculation of \(\omega_0T\) is performed, quantization error can cause significant deviation in resonant frequency depending on calculation time \(T\) and word size.

\[
f_{0Q} = \frac{\text{Round}(\omega_0 \times T \times 16383)}{2\pi T \times 16383}
\]

The word size for this work is chosen as 16 bit long and for scaling ‘1 \(\equiv 16383 \equiv 3FFh\)’. The error gets introduced due to rounding of the result of multiplication between \(\omega_0\) and \(T\). Shifted resonant frequency due to quantization can be expressed through (12). The resonant frequency deviation \(\Delta f_Q\) due to quantization alone can be expressed as shown in (13).

\[
\Delta f_Q(T) = |f_0(T) - f_{0Q}(T)|
\]

Also, the resonant frequency shift \(\Delta f_D\) due to discretization alone at a particular \(f_0\) for various \(T\) can be plotted using (10). If \(f'_0\) is the shifted resonant frequency due to discretization, then \(\Delta f_D(T) = |f_0 - f'_0(T)|\). The combined
frequency deviation can be computed using the result of (12) in (10). If \( f'_{0Q}(T) \) is the shifted resonant frequency due to both discretization and quantization then magnitude of overall resonant frequency deviation (\( \Delta f(T) \)) can be expressed as shown in (14).

\[
\Delta f(T) = |f_0 - f'_{0Q}(T)|
\]

The graph of Fig. 2(b) show magnitudes of \( \Delta f_D(T) \), \( \Delta f_Q(T) \) and \( \Delta f(T) \) for various values of \( T \) when \( f_0 = 250 \text{ Hz} \). It can be observed that deviation due to quantization is significant for low calculation time. For example, resonant frequency deviation can be as high as 2.5 Hz if calculation time is 1 \( \mu \text{s} \) when \( f_0 = 250 \text{ Hz} \). On the contrary the error due to discretization is prominent for higher calculation time. In this work the calculation time for resonant integrator is 10 \( \mu \text{s} \) and for this selection the combined deviation in resonant frequency is estimated as 0.33 Hz.

VI. EFFECT OF DOWN-SAMPLING

In this work PWM update period and system sampling period are equal. PWM update period is \( T_\nu = 250 \mu \text{s} \) and output of RI is being calculated at every \( T = 10 \mu \text{s} \). Hence at every \( T_\nu \) sec output of RI has to be down-sampled to maintain synchronism with the PWM update rate. Hence the total process requires multi-rate computation followed...
by down-sampling as shown in Fig. 3(a). The number of computations for the RI block within \( T_\nu \) is \( \nu = \frac{T_\nu}{T} \).

\[
\begin{bmatrix}
p(k + 1) \\
q(k + 1)
\end{bmatrix} = A_\nu \begin{bmatrix}
p(k) \\
q(k)
\end{bmatrix} + B_\nu e(k)
\]

(15)

Where,

\[ A_\nu = (A_n)^\nu \quad \text{and} \quad B_\nu = \sum_{i=0}^{\nu-1} (A_n)^i B_n \]

(16)

The effect of down-sampling is captured by modifying the state equations as shown in (15) and (16). The resultant resonant frequency can be computed from the eigenvalues of \( A_\nu \). The output equations are shown in (17) and (18).

\[
y(k) = C_\nu \begin{bmatrix}
p(k) \\
q(k)
\end{bmatrix} + D_\nu e(k)
\]

(17)

Where,

\[ C_\nu = C_n (A_n)^\nu \quad \text{and} \quad D_\nu = C_n B_\nu + D_n \]

(18)

Fig. 4(a) shows frequency response plots of transfer functions of RI. \( T_\nu \) is the continuous domain transfer function of RI and its phase plot is \( \phi T_\nu \). In Fig. 4(a) \( T_{r10}^b \) is the discrete domain transfer function of RI, obtained using \( T_r^b \) or (8) with calculation time \( T = 10\mu s \). Whereas \( T_{r250}^b \) is the frequency response of RI obtained from \( T_r^b \), with \( T = 250\mu s \). Lastly \( T_{r10}^b \) is the frequency response of PR controller obtained from \( T_r^b \), with \( T = 10\mu s \) and followed by down-sampling at a rate of 250\( \mu s \). The symbol \( \nu \) indicates that after down-sampling, the \( \nu^{th} \) sample is selected as output \( m(k) \). In this case as \( \nu = 25 \) so 25\( ^{th} \) sample or the last sample is the output after down-sampling. Similarly \( (\nu - 10) \) would indicate that 15\( ^{th} \) sample has been selected as output. From Fig. 4(a) it can be observed that resonant frequency of \( T_{r10}^b \) and \( T_{r10}^b \) closely matches with \( T_r \) but \( T_{r250}^b \) has more phase lag compared to \( T_{r250}^b \) and \( T_{r10}^b \). Though \( T_{r250}^b \) can be advantageous from phase lag perspective, it has significant variation in resonant frequency location. The frequency response of \( T_{r10}^b \) shows accurate resonant frequency location and lesser phase lag compared to other implementations. Hence \( T_{r10}^b \) would be good choice for implementation of RI by two integrator based method, particularly for situations where resonant frequency and system sampling frequency are close.

Fig. 4(b) shows the frequency response of PR controller transfer function implemented using proposed RI with different down-sampling instants between 1 to \( \nu \). Also, in the same figure PR controller with continuous domain RI and conventional method of two integrator based implementation of RI are presented for comparison with proposed method. It can be observed in Fig. 4(b) that for 12\( ^{th} \) down-sampled instant the phase response of the transfer function, indicated by \( \phi T_{r10-12}^b \), closely matches with continuous time phase response \( \phi T_r \). For PR controller implementation a maximum phase lead or phase lag of \( \frac{\omega_0 T_r}{2} \) rad can be obtained at \( \omega_0 \). Hence, selection of down-sampling instant close to \( \nu \) for the RI can give better stability margin near resonant frequency. The \( m^{th} \) prior computation point would give a down-sampled output with the gain matrices of the output given as in (19).

\[ C_{\nu-m} = C_n (A_n)^{\nu-m} \quad \text{and} \quad D_{\nu-m} = C_n B_{\nu-m} + D_n \]

(19)
The output matrices get modified depending on chosen down-sampled instant as shown in (19) for the $(\nu - m)^{th}$ sample chosen as output.

For a filtering application of RI where phase response of the discretized transfer function $T_{\nu-m}^{b}$ is to be matched with the continuous time transfer function $T_{r}$ given in (1) the choice of $m = \left\lceil \frac{\nu}{2} \right\rceil$ would provide matching phase response.

**TABLE IV**

PARAMETERS FOR EXPERIMENTAL SETUP.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>DC bus</td>
<td>$V_{dc}$</td>
<td>300 V</td>
</tr>
<tr>
<td>Filter inductor</td>
<td>$L_{1}$</td>
<td>7.9 mH</td>
</tr>
<tr>
<td>Filter resistor</td>
<td>$R_{1}$</td>
<td>320 mΩ</td>
</tr>
<tr>
<td>Base voltage</td>
<td>$V_{b}$</td>
<td>450 V</td>
</tr>
<tr>
<td>Base current</td>
<td>$I_{b}$</td>
<td>15 A</td>
</tr>
<tr>
<td>Base frequency</td>
<td>$f_{b}$</td>
<td>50 Hz</td>
</tr>
<tr>
<td>Source frequency</td>
<td>$f_{0}$</td>
<td>250 Hz</td>
</tr>
<tr>
<td>Proportional gain</td>
<td>$K_{pr}$</td>
<td>0.25</td>
</tr>
<tr>
<td>Resonant gain</td>
<td>$K_{ir}$</td>
<td>50</td>
</tr>
<tr>
<td>Gain adjustment</td>
<td>$G_{adj} = \frac{V_{b}}{V_{dc}^{2}}$</td>
<td>3</td>
</tr>
</tbody>
</table>

**VII. EXPERIMENTAL VALIDATION**

Experimental validation for resonant frequency variation, frequency adaptation and phase responses are included in this section.
Fig. 6. Comparison of performance of the RI implementations \( T_{250}^{b} \) and \( T_{10}^{b} \). (a) Loop transfer function of experimental setup with PR controller. (b) Zoomed plots around the resonant frequency of 250 Hz.

A. Design and performance of current controller

For calculation of controller gains (\( K_{pr} \) and \( K_{ir} \)) frequency response plots in z-domain has been utilised. Modelling of inverter and filter along with PR controller design procedure followed in this work are available in literature [4], [5]. The discrete domain transfer function of inductive filter system (\( G_{sys} \)), shown in (20), has been obtained using Zero Order Hold (ZOH) which inherently incorporates PWM delay caused by inverter. \( R_1 \) and \( L_1 \) are resistance and inductance of inductive filter and \( T_{\nu} \) is the PWM update period which is same as the ADC sampling period in this case.

\[
G_{sys} = \frac{z^{-1}}{R_1} \left( 1 - e^{-\frac{R_1 T_{\nu}}{L_1}} \right)
\]

(20)

The inverter and digitally implemented controller have been modelled as a delay multiplied by gain (\( G_i \)) as shown in (21). In this experiment the chosen dc bus voltage is lower than 1 pu voltage, as shown in Table-IV. For digital implementation, per unitisation of current control gains are required. Therefore to map the modulation index between 0 to 1 a gain adjustment is necessary. This gain adjustment term is denoted here as \( G_{adj} \).

\[
G_i = G_{adj} \frac{V_{dc}}{2} z^{-1}
\]

(21)

Current sensor gain in the feedback path is assumed as a gain \( K_i = \frac{1}{I_b} \). Inverter and controller causes a delay in the system and effect of this becomes prominent for a system whose switching frequency is low. Delay compensation by two samples for a PR controller has been suggested in [5], [32]. The delay compensation scheme adopted is shown in Fig. 3(b).

These transfer functions along with delay compensated PR controller function (\( G_c \)) have been used to find out
loop gain $GH$.

$$GH = G_{sys}G_cG_iK_i$$  \hspace{1cm} (22)

A resonant frequency at 250 Hz ($5^{th}$ harmonic) has been considered for studying the current controller using the proposed multi-rate RI. The plot of loop gain for three cases of $G_c$ are shown in Fig. 6. For first case, $G_c$ has been obtained utilising the transfer function $T_{250}$ and in second case $T_{r10}$ has been used to obtain $G_c$. In Fig. 6 these two cases are indicated by $GH_{250}$ and $GH_{r10}$ respectively. For the third case shifted resonant frequency $\omega_0Q$ has been used in $T_{r10}$ to obtain $G_c$. ($GH_{r10}^{\nu}$) incorporates the effect of both discretization and quantization on the proposed RI implementation. It can be observed from Fig. 6(a) that the three frequency responses closely match and have almost same phase margins of 40°, indicating same relative stability. The closed loop bandwidth is 270 Hz. However a zoomed version of Fig. 6(a) around resonant frequency, shown in Fig. 6(b), shows that $GH_{250}$ has a gain of 13.78 dB and phase of $+26.5^\circ$ at 250 Hz. Whereas $GH_{10}^{\nu}$ provides very high gain at 250 Hz and ($GH_{10}^{\nu}$)$Q$ provides a gain of 30 dB and phase of $-155.2^\circ$ at 250 Hz.

$$SSE_{th} = \left|\frac{1}{1 + GH}\right| \times 100\%$$  \hspace{1cm} (23)

Theoretically the Steady State Error ($SSE$), shown in (23), for $GH_{250}$ would be 17.24% whereas for $GH_{10}^{\nu}$ and ($GH_{10}^{\nu}$)$Q$ it would be $\approx 0\%$ and 3.25% respectively.

B. Experimental Validation

A laboratory made 10 kVA, 3 phase 4 wire inverter with inductive filter has been used for experimental verification of performance of current control loop implemented using proposed RI. The shorted output of the three filter inductors is connected to the DC bus midpoint which acts as a neutral wire. Block diagram of experimental setup and control architecture is shown in Fig. 5. The inverter current is controlled in per phase manner. Hence three independent controller are used for the three phase four wire system. Parameters for implementation of RI based current controller and hardware details are given in Table-III and Table-IV. Switching frequency of inverter is 2 kHz. As double sampling is opted so ADC sampling frequency is at 4 kHz. Inside the FPGA, sampled phase current, $i_{as}$, is compared with a 250 Hz reference current, $I_{s}^*$ and the difference is given as input to PR controller. The output of PR controller, $m_{PR}$, is down-sampled at a frequency of 4 kHz. The down-sampled output, $m$, is used as the input modulation index for sine triangle PWM generation. Both original TIFB method and the proposed multi-rate TIFB method are used to implement PR controller. Comparison of the tracking performance of these two implementations for a given 250 Hz current reference is the aim of this experiment. The experimental setup is chosen to demonstrate the effectiveness of the proposed multi-rate TIFB implementation of RI. It is selected in this manner specifically to avoid influence of dc bus controller, grid voltage and related additional control mechanisms.

The reference current ($I^*$) is set at 7.5 A peak. Fig. 7(a) shows tracking of reference current with proposed method of implementation whereas Fig. 7(b) shows tracking performance by original TIFB. The output of PR controller $m_{PR}$ and modulation index $m$ for PWM update, which is obtained after down-sampling on $m_{PR}$, are...
Fig. 7. Experimental results comparing the tracking performances of RI used as current controller. (a) Tracking of reference current ($I^*$) using proposed RI $T_{10}^b$. (b) Tracking of reference current ($I^*$) using conventional two integrator based RI $T_{250}^b$. Scale: X-axis: 1ms/div. and Y-axis: Ch1 → $I^*$: 2V/div. (1V ≡ 3A), Ch2 → $m_{pr}$: 2V/div. (1V ≡ 0.2), Ch3 → $I_r$: 2V/div. (1V ≡ 3A), Ch4 → $m$: 2V/div. 

Table V

<table>
<thead>
<tr>
<th>System</th>
<th>Theoretical SSE</th>
<th>Experimental SSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>$GH_{250}$</td>
<td>17.24%</td>
<td>18.94%</td>
</tr>
<tr>
<td>$GH_{10}^v$</td>
<td>≈ 0%</td>
<td>-</td>
</tr>
<tr>
<td>$(GH_{10}^v)Q$</td>
<td>3.25%</td>
<td>2.72%</td>
</tr>
</tbody>
</table>

also shown in these figures. It can be observed from Fig. 7 that the proposed method has lesser steady state error. Experimental values of SSE in % have been calculated, as shown in (24), from the difference of magnitude of reference current ($I_{pk}^*$) and magnitude of 250 Hz current ($I_{pk}$) present in respective phase. Possibility of steady state error in phase is also present however the SSE in magnitudes is indicative of individual performances of the implemented controllers.

$$SSE_{ex} = \frac{|I_{pk}^* - I_{pk}|}{I_{pk}^*} \times 100\%$$ (24)

Fourier analysis is used to calculate the magnitude of 250 Hz component in phase current. The proposed method has lesser steady state error which can be observed from the comparison on SSE shown in Table-V. The error seen between the experimental and theoretical results can be attributed quantization effects, tolerances in the DAC channels from which the data is collected and the data acquisition process. The delay caused by data acquisition process of the board is nearly 10µs. As the rate at which proposed RI is calculated and this delay are equal so there would be loss of one computational sequence in the output based on (19). An error reduction by a factor of six has been observed by the proposed method for this implementation when compared with original TIFB implementation. If currents at different frequency are to be controlled then several RIs of different resonant frequency can be parallely
A fundamental frequency PR controller is implemented in the above described method for a grid connected application. Tuning of the controller gains are based on [33]. Grid voltage and current waveform for steady state operation of the VSI with LCL filter as a STATCOM are shown in Fig. 8(a). The inverter current controller tracks the inverter current command accurately. Distortions in the grid side currents are due to harmonics in the grid voltages and its interactions with the LCL filter. The transient response study on current controller is shown in Fig. 8(b). The three phase currents and R phase reference current \(i^*_r\) are shown in Fig. 8(b). The reference currents were set to 3 A peak. The settling time can be observed to be 2 ms approximately.

VIII. CONCLUSION

In this paper a multi-rate implementation of RI, suitable for FPGA based platform, has been discussed. The computational resource requirement is same as that of TIFB. The resonant frequency accuracy of the proposed method is higher than that of a 4th order method of modified TIFB for higher values of the resonant frequency. Effect of discretization is shown and its implication in the choice of multi-rate computation period is discussed. This helps in designing the multi-rate RI. Using the multi-rate RI implementation it is possible to match the phase characteristics of the continuous time RI by suitable choice of the downsampling instant. The analysis has been validated by verifying the proposed method in an inverter current control application.

REFERENCES


