AN AUTOMATIC TV TUNER ALIGNMENT SYSTEM

Yannick DeVille
Laboratoires d'Electronique Philips, 22, Avenue Descartes, B.P. 15,
94453 Limeil-Brévannes Cedex, France

Abstract- A new approach to automatic TV tuner alignment is presented. It contains two aspects, i.e. hardware (consisting of piecewise-linear transcoders) and software algorithms (which optimize the transfer functions of these transcoders with respect to measured data points). This approach allows to choose the desired tuner performance and it yields a major improvement of the performance/price ratio compared to the conventional solution. More precisely, performance is defined by the maximum frequency mistuning of each RF filter. For given transcoder memory sizes, the proposed approach improves performance (i.e. decreases mistunings) by a factor 2.7 to 4.5. Or, for given performance, it reduces transcoder memory sizes by a factor 2. Especially, the target performance considered is a maximum frequency mistuning of 600 kHz in the overall frequency band of operation. This target is reached with a memory containing only 90 bits per RF filter.

The proposed approach is also a good candidate for future high-definition TV standards, which will require lower frequency mistunings than today. Beyond TV tuner alignment, it is attractive in all systems requiring very simple and/or fast adaptive transcoders. Especially, it may be applied to the automatic electronic tuning of integrated active filters and phase shifters.

1 Introduction

The main topic of this paper is "TV tuner alignment". This notion is defined precisely in the next section and may be summarized as follows: when a TV tuner is assembled in a factory, its radio-frequency filters do not have the right center frequencies, because of component spreads. Therefore, these spreads must be compensated for, by adjusting some parameters of each filter of each manufactured tuner. This operation is called tuner alignment. Today, it is performed manually, by bending coils and checking resulting performance. It is therefore time-consuming and costly.

Conversely, in this paper we are concerned with approaches which yield automatic operation and lower alignment time, and which may provide higher performance by optimizing a larger set of parameters. Such an approach has been presented in [1], but it yields excessive hardware complexity. Conversely, in this paper we propose a solution elaborated so as to minimize the required amount of hardware. This approach contains two aspects, i.e. dedicated hardware transcoders and associated software algorithms which optimize the transfer functions of these transcoders according to an approximation criterion which is fixed by the tuner application.

The remainder of this paper is organized as follows. Section 2 presents the principles of manually aligned tuners. Section 3 describes both types of automatically aligned tuners, i.e. the conventional approach and the solution proposed in this paper. The hardware complexity of the transcoders used in automatically aligned tuners is discussed in Section 4. Simulations results are provided in Section 5. Finally, Section 6 presents the conclusions drawn from this investigation focused on TV tuners, and outlines applications of the proposed approach to other types of systems.

Contributed Paper
Manuscript received April 17, 1995

2 Manually aligned tuners

The current standard tuner structure is shown in Fig. 1 (for one frequency band). It creates an intermediate-frequency signal (IF) by mixing a signal (LO) provided by a local oscillator and a radio-frequency signal (RF) obtained by transferring the antenna signal through three bandpass filters (Antenna, Primary and Secondary). Various oscillator and filter structures may be used (see e.g. [2,3]; see also [4,5] which aim at reducing the mistunings defined below in this section). From the point of view of the tuning characteristics studied in this paper, most of these filters and oscillators may be modelled by LC resonators (Fig. 2) containing a tunable coil Lp, a varicap $Cvar$ and two fixed capacitors $Cpad$ and $Cpar$. The resonance frequencies of these resonators are controlled by means of voltages applied to their varicaps, as explained below.

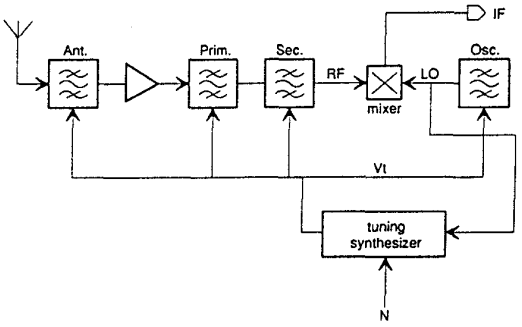

Fig. 1: Structure of a manually aligned tuner (one frequency band).

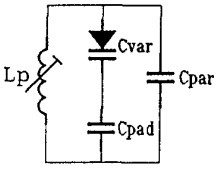

Fig. 2: Resonator structure used to model an oscillator or a filter.

The frequency f_{LO} of the oscillator depends on the voltage Vt provided to its varicap by a synthesizer. This synthesizer operates as follows. When the user selects a channel during TV operation, he chooses a radio-frequency f_{RF}. He thus indirectly defines the desired oscillation frequency f^d_{LO} of the local oscillator, since $f^d_{LO} - f_{RF} = f_{IF}$ (where f_{IF} is the fixed intermediate frequency of the tuner). Therefore, the synthesizer receives a digital word N corresponding to f^d_{LO} and defined as: $f^d_{LO} = N.S$, where S is the frequency step of the synthesizer. The synthesizer then adjusts Vt so that the actual oscillation frequency f_{LO} becomes equal to the desired one, i.e. to f^d_{LO}. To sum up, the current frequency of operation is defined by the selected word N, which is called "the frequency word" below, and which controls the oscillation frequency of the local oscillator.

The center frequencies of the filters should ideally be equal to f_{RF}, i.e. they should track f_{LO} (because, as explained above, $f_{LO} - f_{RF} = f_{IF}$). With the current tuner structure, this condition is not completely met and the performance is defined by the frequency mistuning of each filter (i.e. the difference between its desired and actual center frequencies) obtained for each frequency word N. These mistunings result from the following phenomena. In this type of tuners, the varicaps of the RF filters receive the same control voltage Vt as the local oscillator. This has two consequences. On one hand, this structure yields low design flexibility, because only the values of the components of the resonators may be adjusted. Therefore, even with optimally matched component values in the RF filters and local oscillator, this structure yields significant mistunings at some frequencies. The typical component values are chosen so as to obtain an acceptable balance of these mistunings over the complete frequency band. On the other hand, when manufacturing a set of tuners in a tuner factory, the components used have spread values. This yields unacceptable frequency mistunings, so that some component values must be adjusted in order to compensate for component spreads. Each tuner is therefore "aligned" individually, by manually bending the coils of the RF filters while checking the resulting performance, so as to reduce observed mistunings (details about such an alignment procedure may be found in [6]). Tuner alignment is therefore time-consuming and requires human operation, which induces significant costs. In addition, performance is thus limited by the fact that only a single parameter (i.e. the coil) is tuned to compensate for component

spreads. A way to solve these problems consists in using automatically aligned tuners, which are presented in the next section.

3 Automatically aligned tuners

3.1 Overall tuner structure

A typical automatically aligned tuner is shown in Fig. 3 (for one frequency band). It differs from the manually aligned structure in the voltages (Vt_1 to Vt_3) applied to the varicaps of the RF filters for controlling their center frequencies. Here, these voltages are created by single-input single-output digital transcoders (TR1 to TR3). Each transcoder receives the selected frequency word N and basically provides the corresponding voltage Vt_i used to control the considered filter. More precisely, it is associated with a Digital/Analog Converter (DAC), which converts its digital output into an analog voltage. Moreover, each transcoder is preferably designed so as to output the voltage difference $\Delta Vt_i = Vt_i - Vt$. This voltage difference is then added to Vt in order to obtain Vt_i. This provides partial temperature compensation and reduces the range of the transcoder output (and thus the size of its memory).

As shown below, each transcoder transfer function is automatically created during factory alignment, so as to fit the ideal function which would entail no mistunings. This yields automatic component spread compensation and lower alignment time. Moreover, this approach is flexible, since it allows to select freely the number of parameters of the transcoder transfer functions which are adjusted during alignment. This allows to choose the desired trade-off between the complexity of the transcoders and the accuracy of the approximating functions that they provide (which defines the magnitude of the mistunings). Especially, this structure may be configured so as to yield higher performance than manually aligned tuners. This is not crucial today, because the requirements of the current TV standards on performance may be met by manually aligned tuners. But future high-definition TV standards will require much better tracking between the frequencies of the RF filters and local oscillator. This will probably not be achievable by only tuning a coil, so that transcoder-based tuners will then appear even more advantageous.

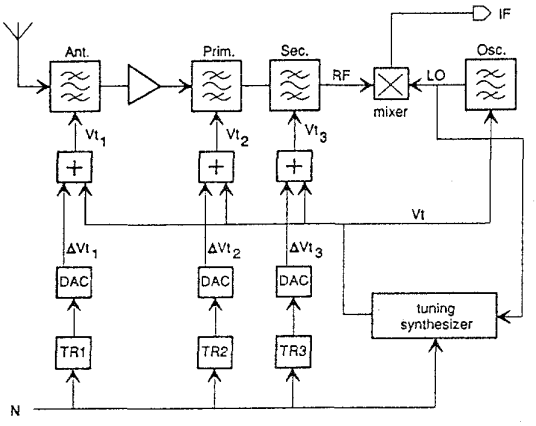

Fig. 3: Structure of an automatically aligned tuner (one frequency band).

3.2 Conventional transcoder structure and alignment procedure

An approach based on the overall tuner structure presented above has been proposed [1]. It uses automatic test measurements, performed in the tuner factory for each filter of each tuner and for a given set of equidistant frequency words N. For each such frequency, the test system sends a varying input to the DAC of the considered filter. For each such DAC input, it measures the center frequency of the filter and it derives its mistuning (by also taking into account the frequency of the local oscillator which is measured once for each test frequency). It thus determines the optimum DAC input, i.e. the one which yields the lowest mistuning. This optimum DAC input is then stored in the memory (EEPROM) contained by the transcoder. Then, during normal operation, when the user selects a channel, the transcoder operates as follows. It first compares the current frequency word N to those used during the test measurements (this may be performed by relatively simple hardware decoding means when using fixed and equidistant measurement frequencies with an adequate frequency step). If the current N is equal to one of those used during the measurements, then the transcoder directly outputs the corresponding optimum DAC

input stored in its memory. Otherwise, it outputs a value obtained by performing a linear interpolation between the two optimum DAC inputs corresponding to the two test frequencies which immediately surround the current frequency.

In fact, this description already corresponds to a modified version of the conventional approach. In the original structure, each transcoder only consists of memory. The accesses to that memory and the interpolation computations are performed by the microprocessor of the TV set, and the results are then sent to the tuner on the bus of the TV set. The drawback of this original approach is that it is not software compatible with standard TV sets. Software compatibility is obtained by including all the needed computational capability in the transcoder, thus yielding the modified version described above. This is the version which is preferred and referred to as the "conventional approach" in the remainder of this paper. A possible structure for its transcoders may be derived from the above principles of operation. An alternative solution is presented in the next subsection.

To measure the performance achieved by this approach, a performance criterion should first be chosen. Tuner experts of the Philips company define the overall performance of a filter and associated transcoder as follows. First, for a single frequency word N, the performance is equal to the absolute value of the frequency mistuning, which results from the difference between the ideal and actual voltages at the output of the transcoder (or of the DAC). Then, the overall performance is defined by using a worst-case approach, i.e. it is the highest value among all the absolute frequency mistunings obtained over the complete frequency band of operation (i.e. for all the discrete values N situated in that band).

With this performance criterion, the major drawback of the tuner structure presented above is to achieve a bad balance of the mistunings which occur at various frequencies. This lack of balance has two causes. On one hand, this type of transcoder yields a piecewise-linear transfer function, which is such that for operation frequencies equal to test measurement frequencies the control voltages of the RF filters are close to the ideal ones (the difference only results from limited DAC resolution). Therefore, low mistunings are obtained at these frequencies. But this is achieved at the expense of high mistunings at operation frequencies situated far from their adjacent test measurement frequencies, since the actual linear transfer function of the transcoder is then far from the ideal non-linear one. On the other hand, equidistant test measurement frequencies yield equi-width segments in the transfer function of the transcoder. This provides much higher mistunings in the segments corresponding to the frequency sub-ranges where the ideal transfer function of the transcoder is much more non-linear. Because of these unbalanced mistunings, this structure is not optimum, i.e. it does not minimize the number of measured values that must be stored in the transcoder memory to achieve given overall performance (i.e. given worst-case mistuning over the whole band). Conversely, the approach presented below was elaborated so as to reduce the size of the transcoder memory.

3.3 Proposed transcoder structure and alignment procedure

The approach proposed in this paper is also based on the overall tuner structure presented in Subsection 3.1. Its specific features concern the hardware structure of the transcoders and the method used to initialize their transfer functions, i.e. the alignment procedure.

These transcoders contain logic circuitry, that they may share in order to reduce overall hardware complexity, and individual memory (EEPROM). The logic defines the functional form of the transfer functions that they may implement, while the content of the memory of each transcoder specifies a particular function having this form. These transcoders are intended to be embedded into one of the integrated circuits of the tuner. This requires simple logic and small memories (and hence, limited sensitivity to truncation effects). Therefore, the selected functional form consists of piecewise-linear functions (or in fact approximations of such functions: see below). Their number of segments m is fixed. Conversely, the other parameter values of the transfer function of each transcoder are optimized during factory

alignment, so as to compensate for component spreads.

To this end, the alignment procedure uses test measurements which differ as follows from those performed in the conventional approach. First, these measurements may here be performed at any frequencies, thus yielding a set of measurements which may correspond to any arbitrary subset of the overall set of possible frequency words. This feature may be used to reduce the number of measurements needed to achieve given performance, by performing more measurements in frequency sub-ranges where the ideal transcoder response is highly non-linear. The overall alignment time is thus reduced. In addition, for each measurement frequency, these measurements aim at determining the ideal transcoder output or ideal DAC input, i.e. the real-valued DAC input which would yield no mistunings. This ideal value cannot be determined exactly, because in practice the DAC input can only take integral values. However, the software algorithms of the test system situated in the factory can determine a good estimate of this ideal value by performing an interpolation from the mistunings obtained for various integer-valued DAC inputs. These measurements yield a set of points which are specific to each filter because of component spreads, and which are each defined by two coordinates, i.e. a frequency word N and an estimate of the corresponding ideal transcoder output. Here, these points are not stored directly in the transcoder memory but temporarily kept in the test system of the tuner factory. The next step consists in deriving an "optimum" piecewise-linear approximating function for this discrete set of measurement points. This approximating function will then be used as the transfer function of the transcoder. This approach has the following two consequences. On one hand, one should define the "error" corresponding to a given filter (only known through the set of measurements points described above, which partly defines its ideal tuning curve) and to an arbitrary transcoder transfer function (supposed to be piecewise-linear, because of above assumption). The definition of this "error" is derived from the application itself: as stated above, real performance is measured by the overall frequency mistuning (i.e. the worst-case mistuning over the complete frequency band). Therefore, we defined an error which is an estimate of this overall mistuning and which only uses the available data (i.e. the measurement points and the parameter values of the considered transcoder transfer function). A mathematical description of this "error" is provided in [7]. On the other hand, algorithms are needed to determine an optimum transcoder transfer function, i.e. a transfer function which minimizes the "error" defined above (for the considered filter). The algorithms that we developed to this end are presented in [7,8].

This alignment procedure yields the following main features. The test system situated in the factory contains software algorithms which receive measured values and which derive the parameter values of a corresponding optimum piecewise-linear function. These parameter values are then stored in the memory (EEPROM) of the transcoder, which is thus completely aligned. Each transcoder transfer function is thus piecewise-linear, as in the conventional approach. However, the parameters of each segment (i.e, width in frequency domain and position of corresponding line) are optimized, with respect to overall mistuning. This should yield a higher performance/memory-size ratio than the conventional approach. This is confirmed by the simulation results presented in Section 5.

Any piecewise-linear function determined by this alignment procedure may be mapped into various transcoder architectures. When high-speed operation is required, a parallel neural network implementation similar to [9] is preferable. Conversely, in the tuner application the main requirement is on hardware simplicity while speed is not a bottleneck. Therefore, simple structures based on serial computations are preferred, at the expense of slower operation. The basic structure corresponding to this approach is represented in Fig. 4. This transcoder provides $y = f(x)$, defined as follows. Its input x is here a frequency word N. Its output y is a digital word, to be here used as a DAC input. Its transfer function f is a piecewise-linear function (more precisely, an approximation of such a function: see below) containing m segments, where m is fixed for a

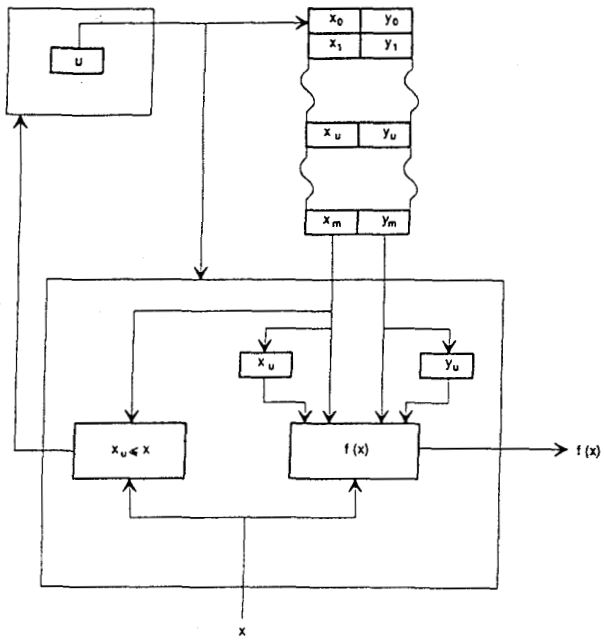

Fig. 4: Basic transcoder structure.

given chip and defines the hardware complexity and the resulting performance (see details below). It should be clear that such a transcoder may implement *any* piecewise-linear function. Therefore, it may be used with the conventional approach to automatic tuner alignment, with the approach that we propose or with any other method, depending on the values stored in its memory. This is used below to compare the conventional and proposed approaches. This comparison requires some care: with the proposed approach, some X coordinates are (directly or indirectly) stored in the transcoder memory, as explained below. They should be taken into account when determining the memory size. Conversely, when using the conventional approach, such values may also be stored, or they may be avoided by using simple decoding means, as explained above. The latter solution is more attractive from the point of view of memory size, and it is the one which is considered in this paper when comparing the memory sizes of the conventional and proposed approaches.

In the basic version of our approach, the transcoder operates as follows. It contains a controller (see upper left part of Fig. 4) which manages overall operation. This controller especially contains a register (see block "u" of Fig. 4) which holds a segment index u which evolves as explained below. The transcoder memory (see upper right part of Fig. 4) contains $(m+1)$ lines. Each one of these lines contains a point of the transcoder transfer function, i.e. (x_u, y_u) with $u = 0$ to m. The points indexed by $u = 1$ to $m-1$ correspond to the limits between the segments of the function f, starting from the lowest values of the input x up to the highest ones. The points corresponding to $u = 0$ and m may be arbitrary points situated resp. in the leftmost and rightmost segments. It should be clear that the values of all these point coordinates are automatically assigned by the alignment software algorithms mentioned above, when filling the EEPROM of the transcoder at the factory. They are then used as explained below during normal operation. The remainder of the transcoder (see lower part of Fig. 4) performs the following operations. It contains a "point register" (see blocks x_u and y_u in Fig. 4), where the coordinates of one point of the transcoder memory may be temporarily stored. When the user selects a channel, the transcoder receives an input x (i.e. N) and a request to provide the corresponding output $f(x)$. In a first phase, it determines which segment of f contains the current input x. To this end, the controller first accesses the line $u = 0$ of the memory and copies the point (x_0, y_0) in the "point register". It then accesses successive memory lines, starting at $u = 1$ and then increasing u. For each u, it compares the accessed value x_u with the current input x (see the block $x_u \leq x$ in Fig. 4). If the condition $x_u \leq x$ is met, then it copies the currently accessed memory point (x_u, y_u) into the "point register" and it starts the same operation for the next segment, i.e. $u+1$. Otherwise, it ends the first phase. The second phase then starts. The segment of f corresponding to the current input x is defined by the two memory points having X coordinates directly adjacent to this current input x. These two points are available: the left limit of the segment is stored in the "point register", while the right limit is currently accessed from the memory (because of the value of u at the end of the first phase). The second

phase therefore only consists in performing a linear interpolation between these two points (see the block $f(x)$ in Fig. 4), and in providing this value as the transcoder output. More precisely, only an approximation of linear interpolation is used here, i.e. the actual transcoder output is the nearest integer to the value that would correspond to exact linear interpolation. This approximation is attractive because it yields a simpler transcoder structure. Moreover, it does not significantly degrade performance. This is shown by the simulation results presented below and may be explained as follows. On one hand, performing exact linear interpolation would be useless because anyway the transcoder output is to be used as a quantized (integer) DAC input. On the other hand, performing exact linear interpolation would also be meaningless because the initial values, i.e. the memory points, have quantized (integer) values.

An improved version of the transcoder is obtained as follows. First, instead of storing the absolute X coordinates x_u of the points which define the transfer function f, only their differences $(x_u - x_{u-1})$ are stored in the memory (except that the absolute value x_0 of the first point is stored). When processing each current input x, the absolute x_u are successively computed from the available differences, thus taking advantage of the fact that the segments must be accessed for successive u anyway. These computations only use the hardware means already available, and this approach reduces the number of memory bits needed for the information concerning the X coordinates, because the differences $(x_u - x_{u-1})$ require fewer bits than the absolute x_u. The next modification of the approach consists in only allowing each of these differences to be a multiple m_u of a given step s. Only the multiples m_u are stored in memory, and the differences $(x_u - x_{u-1})$ are derived from m_u as: $x_u - x_{u-1} = m_u.s$. This approach yields an additional reduction of the memory size, since the multiples m_u require fewer bits than the differences. However, it may also degrade the performance obtained with a given number m of segments, because only a few discrete values are now accepted for the parameters x_u in the alignment software algorithms. The goal is then to determine s and the set of values allowed for each m_u which finally yield the minimum number of bits for a given acceptable overall mistuning. These values depend on the type of tuner considered, and numerical values are therefore provided in Section 5. In addition, the step s should preferably be set to a power of 2 when possible, so that the computation of $m_u.s$ only consists of a shift of m_u. Other transcoder structures may be created. Especially, speed is increased by using a dichotomic search to determine which segment contains the current input x. Finally, it should be noted that all these transcoders perform all needed computations and therefore provide software compatibility with current TV sets, in the same way as the conventional approach.

4 Hardware complexity of the transcoders

The transcoders described in the previous section contain two parts, i.e:

- The logic (see lower and left parts of Fig. 4), which may be shared by all transcoders in order to reduce hardware complexity , and which aims at performing computations from the transcoder input and memory content, in order to derive the transcoder output.
- The memory (see right part of Fig. 4), whose content is specific to each transcoder.

The complexity of the logic has been investigated in the case when a large part of the internal operations of the transcoder are performed sequentially, in order to reduce complexity. This complexity mainly depends on the number of bits of the values processed, as opposed to the number of segments m. Moreover, the considered Silicon implementation yields a complexity per bit close to 110 gates (or 500 transistors), corresponding to a Silicon area close to 0.1 mm^2 in a $1\mu m$ technology. For the tuner type considered in the next section, the DAC resolution must be set to 10 bits. The values processed by the transcoders then also contain 10 bits, resulting in a Silicon area close to 1 mm^2 for all the logic, which is quite modest.

The complexity of the memory is defined by its number of bits. It depends on the number of seg-

ments m, and thus on the desired performance and on the considered approach to tuner alignment. It is therefore derived from simulations and provided in the next section.

5 Simulations

5.1 Simulation conditions

Simulations were performed with the two automatically aligned tuners presented above, i.e. with the conventional and proposed structures (which operate in one frequency band). These simulations were focused on the mistuning of an RF filter (for a given local oscillator). With the component values used in the resonators (see typical values and ranges in Table I), the considered tuners correspond to the medium band of a complete 3-band tuner, and this medium band is equal to [200MHz,488MHz] for the local oscillator. All simulations were performed under the assumption of matched varicaps in the RF filter and local oscillator, since this is the case in current commercial tuners. This was modelled by using exactly the same varicap parameters in both resonators. This is realistic because the results provided below correspond to resonator component values providing worst-case mistunings, and in this case the mistunings resulting from different but matched varicaps are much lower than those resulting from the other component values. The number of parts m of the transfer functions of the transcoders was varied in the simulations, in order to investigate the resulting tuner performance. Especially, we determined the number of parts m and the corresponding memory size needed to reach the specified target (i.e. 600 kHz mistunings).

Ideal linear DACs were used. Their resolution R (i.e. their number of bits) and their LSB (i.e. the voltage step corresponding to their Least Significant Bit) were selected as follows. Philips tuner experts showed that the maximum acceptable mistuning in the considered band is about 600 kHz. This mistuning results from both the truncation effects in the DAC (i.e. limited resolution) and the fact that the tuning function created by the transcoder is only an approximation of the ideal tuning curve. This means that truncation

TABLE I
COMPONENT VALUES USED IN THE RESONATORS
(TYPICAL VALUES AND RANGES)

component	RF filter	local oscillator
Lp (nH)	24.55 ±7%	20 ±7%
varicap	Philips BB 133	Philips BB 133
Cpad (pF)	4700 ±5%	137 ±5%
Cpar (pF)	1.9 ±5%	2.235 ±5%

effects in the DAC alone should yield frequency mistunings somewhat lower than 600 kHz, e.g. 300 kHz to allow an additional 300 kHz error for the approximation effect. In addition, simulations showed that the considered type of tuner yields a maximum "tuning slope" close to 20 MHz/V (where the tuning slope is the slope of the curve providing the resonance frequency of a resonator vs. the control voltage of its varicap). Therefore, the maximum voltage error accepted at the output of the DAC is 300 kHz / (20 MHz/V) = 15 mV. This is the maximum acceptable DAC LSB. Moreover, simulations showed that the range required for the DAC output is about [-7V,+7V]. This yields the minimum DAC resolution R, defined by: $2^R * 15\,mV \geq 14V$, and therefore $R = 10$. This resolution value deserves the following comments. It does not depend on the considered approach to automatic aligment (i.e. conventional vs. proposed approach), but on the target performance and on the fact that only linear DACs were used. It is relatively high. To reduce it, a useful extension of the approach would consist in using non-linear DACs (which have a lower LSB in domains where a higher voltage accuracy is required, i.e. where the tuning slope is high).

5.2 First set of simulations

In the first set of simulations, the coordinates x_u stored in the transcoder of the proposed tuner structure were allowed to have any values. The alignment algorithms therefore provided the optimum values among all real values. This corresponds to using the first transcoder described in Subsection 3.3. These simulations allowed to derive the overall mistuning corresponding to each considered value of the number of segments

m, both for the proposed and conventional approaches (see Fig. 5). This confirmed that the proposed approach provides much lower mistunings for given m. However, it requires a few additional bits per segment to store the coordinates x_u, as explained above. The parameter of interest is the final number of bits required to achieve given mistuning. This was investigated with the adequate version of the transcoder, thus yielding the second set of simulations presented below.

The results presented here were also compared with those obtained in a non-realistic version where no truncation effects would occur (in the DACs, in the transcoder memory words and in linear interpolation). Very similar results were thus obtained. This confirms that truncation effects do not significantly degrade performance in the considered conditions, where the overall mistunings remain high enough as compared to the limitation set by the DAC.

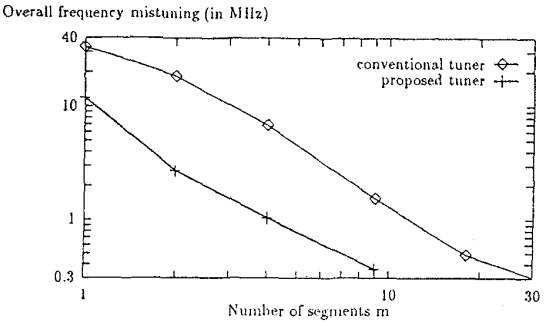

Fig. 5: Overall frequency mistuning (i.e. maximum mistuning over the complete frequency band) vs. number of segments m for both types of automatically aligned tuners (i.e. conventional and proposed structures).

5.3 Second set of simulations

In the second set of simulations, the coordinates x_u stored in the transcoder of the proposed tuner were restricted to be in a small discrete set of allowed values. This approach corresponds to using the second transcoder described in Subsection 3.3. The principles of this transcoder were provided in that section, while the selection of the set of values allowed for each m_u and of the parameter s will now be considered.

The number of values allowed for m_u should be as low as possible, in order to minimize the number of bits to be stored in the transcoder memory. The minimum allowed m_u is equal to 1. The maximum allowed m_u is set to a value which depends on the type of tuners considered: if the ideal transfer function of the transcoder (or tuning curve of the tuner) is highly non-linear in some frequency domains, the segments which minimize the maximum mistunings are much narrower in the non-linear domains than in the linear ones. For the tuner type considered here, simulations show that the optimum segment widths are in a ratio 5 to 1. The maximum allowed m_u is then set to 5 (in order to reduce the number of bits: it could be set to a higher value, by using a smaller step s; this would yield more precise optimization but would require more bits). The simplest solution would therefore consist in allowing $m_u = 1$ or 5 only, and in storing a single bit in the memory for each m_u to define which of these two values is selected. However, allowing only two values is likely to yield low performance, because the coordinates x_u are thus far from their optimum values. Therefore, it is preferred to use two memory bits per value m_u. This yields 4 possible values for each memory word defining a value m_u, i.e. the set of values allowed for all m_u contains 4 values situated in the range 1 to 5. In other words, one of the values of this range cannot be used. It is chosen to forbid the value 4, because the lacking value is expected to yield lower performance degradation if it is in the higher end of the range of allowed values. This approach finally yields the following hardware structure. To define indirectly each x_u, the transcoder memory contains a 2-bit word, whose value ranges from 0 to 3. This value should be sent to a very simple hardware block which derives the corresponding value m_u according to the following table: the memory words 0, 1, 2, 3 resp. correspond to m_u = 1, 2, 3, 5 (which are the values allowed for m_u, as explained above). x_u is then derived from m_u as explained in Section 3.

The parameter s is selected according to the considered tuner parameters, because it depends on the overall frequency band of operation of the tuner, and on the number m of segments in which this band is split. It also depends on the set of

values allowed for m_u: when higher values are allowed, a lower step s is selected. For the tuner type considered here and with the values defined above for m_u, simulations yield: $s = 512$ and 256, resp. for $m = 4$ and 9 segments.

Once these conditions were set, the proposed tuner structure was studied as follows. New simulations were performed for various values of the number of segments m in order to determine the optimum mistuning that may be achieved when the coordinates x_u are restricted to the considered set of values. The number of memory bits corresponding to each value m was then derived as follows. The memory contains $m + 1$ points. The X coordinate of each such point corresponds to 2 bits, as explained above (the additional bits used for storing the absolute value of x_0 are neglected). The Y coordinate of each point corresponds to 10 bits, because a 10-bit DAC is used. Therefore, the overall memory size is $12.(m + 1)$ bits. From this, a chart providing the overall mistuning vs. the number of bits in the memory may be derived. The same type of chart may be derived for the conventional approach. In that case however, each memory point is reduced to its 10-bit Y coordinate, by replacing the X coordinates (which are fixed) by hardware decoding logic.

The results thus obtained are represented in Fig. 6. The major parameter of interest is the memory size required for each tuner structure to achieve given performance, i.e. given overall mistuning. Our approach reduces this size by a factor 1.9 to 2.1 as compared to the conventional approach. The second aspect is the overall mistuning achieved by each structure for given memory size. Our approach reduces this mistuning by a factor 2.7 to 4.5. Therefore our approach yields a major improvement over the conventional approach for both parameters. Finally, the number of bits required by our approach to achieve the target performance (i.e. 600 kHz) is only 90 bits.

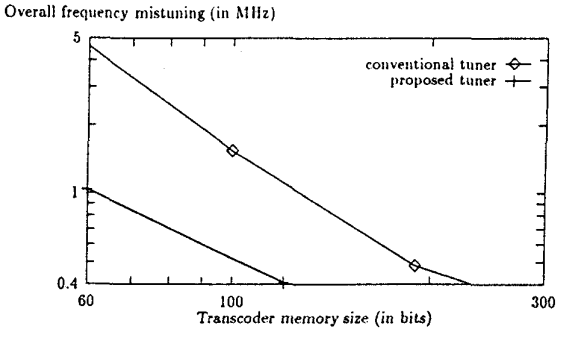

Fig. 6: Overall frequency mistuning (i.e. maximum mistuning over the complete frequency band) vs. number of bits of transcoder memory for both types of automatically aligned tuners (i.e. conventional and proposed structures).

6 Conclusions

A complete solution to automatic TV tuner alignment has been developed. This approach contains two aspects, i.e. hardware (consisting of the piecewise-linear transcoders presented above) and software algorithms (which optimize the transfer function of these transcoders with respect to measured data points). This new method allows to choose freely the desired performance. In addition, it yields a major improvement of the performance/price ratio over the conventional approach, i.e: for given performance, it reduces the memory size by a factor close to 2, or for given memory size it improves performance (i.e. it decreases mistunings) by a factor 2.7 to 4.5. More specifically, the number of bits required by our approach to achieve the target performance (i.e. a maximum mistuning of 600 kHz) is only 90 bits per RF filter. This small memory size makes the proposed method very attractive. In addition, this approach is a good candidate for the alignment of future high-definition TV sets, as explained in Subsection 3.1.

Finally, it should be clear that this approach provides a very general solution to the synthesis of simple and/or fast adaptive transcoders, and therefore has a much wider field of application than tuner alignment. This includes e.g. the control of the center frequency and/or selectivity of fully-integrated continuous-time active filters, or the phase and/or gain of integrated active phase shifters (see details in [7]). More generally, this approach applies to various types of electronic tunings, aiming at compensating for components spreads.

Acknowledgment

The author would like to thank H. Brekelmans for prodiving the component values of the resonators used in simulations, and R. van der Wal and P. Walbrou for helpful discussions about tuner structures. Acknowlegment is also made to the publishers Taylor & Francis Ltd, who allowed the author to reuse some of the material published in [7,8] in the current paper.

References

1. W. Heucke-Gareis and E. Oertel, European Patent No. 0,147,518 (corresponding US Patent No. 4,334,323).
2. "Components for highly-integrated TV tuners", Philips Components data sheet.
3. J. G. Lea and D. J. Carlson, US Patent No. 4,494,081.
4. H. Yokoyama, "Electronic tuning receiver", US Patent No. 4,794,650.
5. H. Yokoyama, "Electronic tuning circuit for AM receiver", US Patent No. 4,847,572.
6. H. Keller and V. Morstadt, GB Patent No. 1,217,734.
7. Y. Deville, "Optimum piecewise-linear transcoders - Part 2. Application to automatic TV tuner alignment and to electronic tuning", International Journal of Electronics, vol. 77, pp. 845-862, Dec. 1994.
8. Y. Deville, "Optimum piecewise-linear transcoders - Part 1. Weighted minimax piecewise-linear approximation and minimax decomposition of piecewise functions", International Journal of Electronics, vol. 77, pp. 823-844, Dec. 1994.
9. Y. Deville, "A neural implementation of complex activation functions for digital VLSI neural networks", Microelectronics Journal, vol. 24, pp. 259-262, May 1993.

Biography

Y. Deville graduated from the Ecole Nationale Supérieure des Télécommunications de Bretagne (Brest, France) in 1986. At the same time, he obtained a D.E.A in Microelectronics degree from the Joseph Fourier University of Grenoble (France). Since 1986, he has been with the Laboratoires d'Electronique Philips at Limeil (France), where he is now a Senior Scientist. From 1986 to 1988, he prepared a doctoral thesis on GaAs integrated microwave RC active filters. He then investigated VLSI cache memory replacement algorithms and architectures. Since 1990, his research interests mainly concern neural network algorithms and nonlinear systems, and associated signal processing applications and hardware implementations. Especially, he is working on source separation techniques and their applications to antenna processing and to audio noise cancellation.