B. Disadvantages

1) Expense: it is advisable that multiple sets of duplicate equipments have exactly the same audio characteristics what increments the cost. There are two reasons for that:

- The voter does its voting on audio characteristics and quality. If the sites sound different, it will negatively affect the voting process.
- It can affect clarity and intelligibility. If the voter were to select a different receiver in the middle of a word, or even change receivers multiple times and the audio from the first receiver was normal, from the second receiver was high and from the third receiver was tinny it would be very hard, if not impossible, to understand.

2) Complexity: A voter-based repeater has additional complexity that increments with each voting receiver site added.

3) Calibration: it requires a calibration procedure to balance the links between the base station and the central station.

III. Hardware Structure

Fig. 1 shows an overview of the system application we have developed.

The present design assumes that the input channels (Ch1, Ch2, Ch3, Ch4) are comparable, that is, they contain the same information from the same source but travelling through different paths to the control centre that selects the best one.

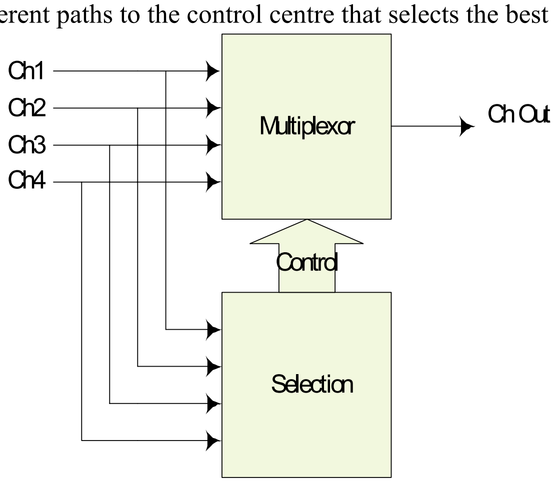

Fig. 1: System overview

In the selection block, shown in the Fig. 2, a processing procedure for each channel is carried out. It consists in a low-pass filter to 9.4 KHz in order to obtain the parameter considered as a signal (S1, S2, S3, S4) and, a high-pass filter with the same cut frequency for obtaining the noise component (N1, N2, N3, N4).

If you are using *Word,* use either the Microsoft Equation Editor or the *MathType* add-on (http://www.mathtype.com) for equations in your paper (Insert | Object | Create New | Microsoft Equation *or* MathType Equation). “Float over text” should *not* be selected.

Next, signal and noise are both passed through a true RMS-to-DC converter, that will generate a continuous voltage proportional to the efficient value of each of them having, therefore, eight levels of continuous voltage: four for the signals (SRMS1, SRMS2, SRMS3, SRMS4) and four for the noises (NRMS1, NMRS2, NMRS3, NRMS4). The idea behind this signal processing is to detect the high frequency harmonic components in some kind of noise (crackle, spark and shot) [2]. Finally, the control block will be in charge of performing the analog-to-digital conversion for each input and executing the selection procedure as it is described next.

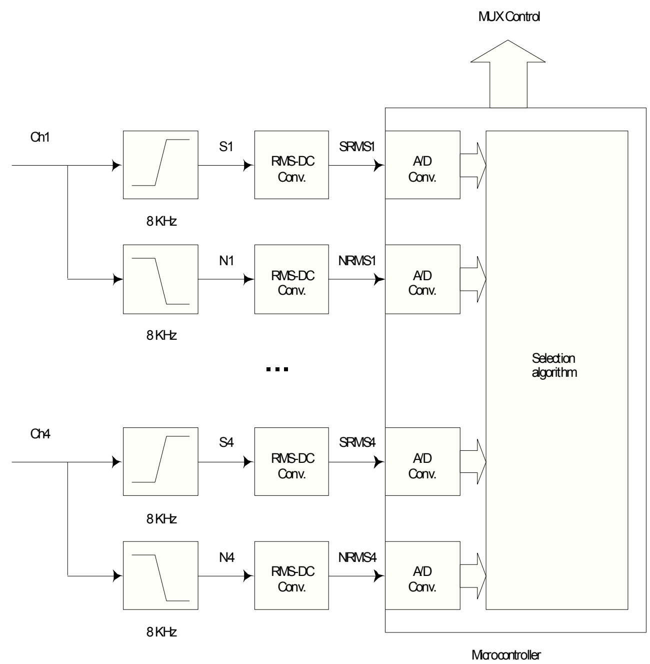

Fig. 2: Analog signal processing

IV. Software Structure

The software has been coded in assembly language for the PIC16F74B microcontroller [4] (the manufacturer is Microchip). The program is based on the data sampling and To calculate the inactivity time for each channel and if that processing with a frequency of 25 Hz, which is enough for the type of application of baseband audio channels .

For each repetition of the main loop, the following tasks are carried out for every signal sample (see Fig. 3). They will be described in detail in the following subsections.

1. Analog/Digital conversion of all the signals.
2. Comparison parameters computation.
3. Output channel selection decision.
4. Wait to the following cycle.

The temporization is carried out by programming a microcontroller Timer to generate an interruption every 5 msg. For that interruption, in addition to verify if it is time to execute a new cycle, a series of extra tasks are carried out:

- To check if an initial time of signal stabilization has elapsed after the device has been turned on.
- To calculate the inactivity time for each channel and if that time surpasses an established maximum value for all of the channels. That means that if there is no channel with a valid

signal, then a disconnection of the output occurs.

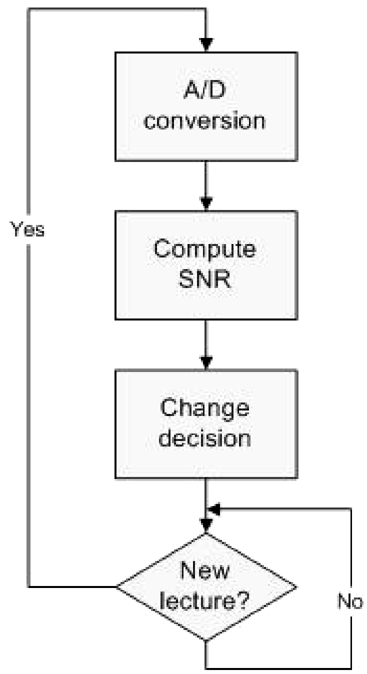

Fig. 3: Main program loop

A. A/D Conversion

Every eight interruptions of the Timer (that is, 8 x 5 = 40 msg.), starts a read cycle of the A/D converter for each of the output lines of the RMS-to-DC converter. The conversion is carried out through the PORTA pins of the microcontroller, configured to be used as analog inputs and using VREF, the own feed VDD (ADCON1 = 0x00), as reference voltage. Given that the conversion time is not critical, the internal RC oscillator has been configured to act as converter clock. In any case, the worst-case acquisition time that the manufacturer guaranties is 57µsg. (9,5 * 6µsg), which is enough for our application. The corresponding code for the beginning of the conversion is shown in Fig. 4.

Fig. 5 shows the reading process cycle of the four RMS signal levels and the four RMS noise levels. Once the eight channels have been read, which supposes a time lower than 0.5 msg., the next phase begins.

A simple first order FIR filter is applied to the RMS noise and signal level, in order to minimise the quick time variations [3].

```
StartADConv
  movf ChannelAD, W
  movwfAARGB0
  movlw8
  movwfBARGB0
  call FXM0808U
  movf AARGB1, W
  iorlw0xc1
  movwfADCON0
  movlw0x20
  movwfVAux1

StartADConv1
  decfsz VAux1, F
  goto StartADConv1
  bsf ADCON0, GO_DONE
```

Fig. 4: Piece of code for the preamble of the A/D conversion

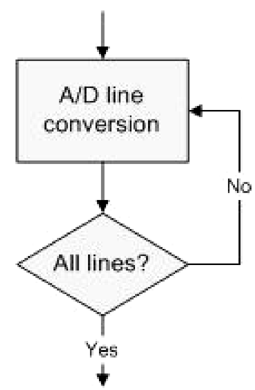

Fig. 5: A/D conversion procedure

B. Comparison parameters computation

Fig. 6 shows the flow chart corresponding to the comparison parameters computation. It can be seen that after computing the Signal-to-Noise relation for each channel, it is verified if the maximum value that can be obtained surpasses a certain threshold. This is done by means of the formula (256*S)/N, that tries to better discriminate the results of the comparison in binary. When the threshold is surpassed, it means that the signal quality is good enough and the channel that has a maximum value in the SNR parameter is selected. In the other case, the situation would be that all the signals have a high noise level. When this happens, the program selects the channel with the minimum noise level. The SNR threshold value has been selected after several tests in which relative values of signal and noise are compared to check which one generated the best comfortable result for the human ear.

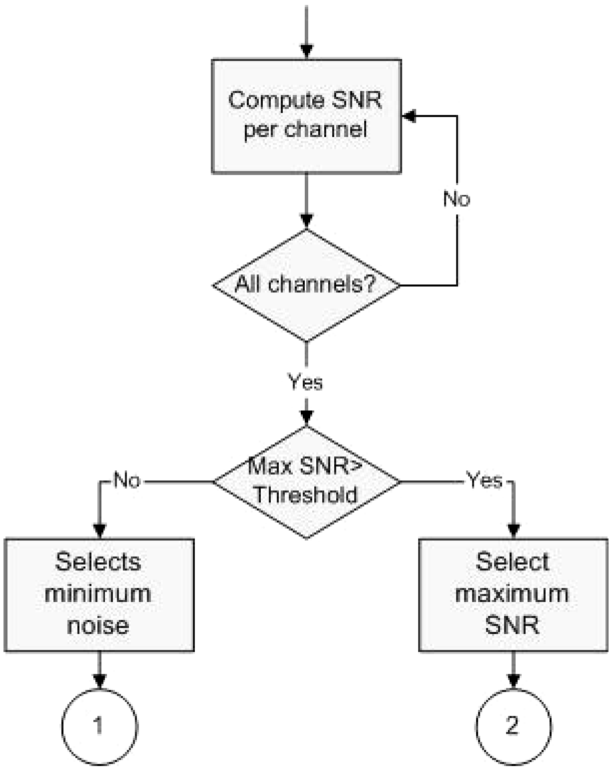

Fig. 6: Channel selection flow chart

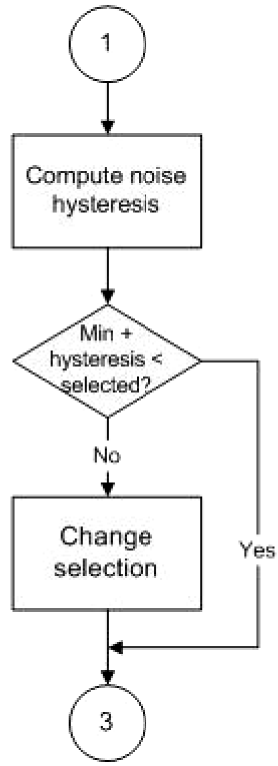

Fig. 7: Minimum noise selection flow chart

C. *Output Channel selection decision*

The way to decide which one is the input channel that will be selected as the output channel depends on the parameter that was used to carry out the selection. When the comparison parameter is the noise level, the flow chart used is the one of Fig. 7. We can see that if the channel selected by minimum noise level is different from the one selected during the previous cycle, there will be a channel switching when the difference between both of them is greater than a percentage of the noise level of the selected channel (hysteresis). This hysteresis value avoids an uncomfortable channel switching in the case of very similar channels.

If the comparison is carried out taking into account the SNR maximum value, then the diagram of Fig. 8 applies. In this case, if the noise level is very small, the hysteresis responsible of the channel selection is based on the signal level, that is, there will only be a change of channel if the signal of the channel with maximum SNR less a calculated value of hysteresis is greater that the level of the signal selected in the previous cycle.

This rule allows that, when the channels have very low noise level, then the commutation will be produced just if the signal level of the new selection is clearly better than the one found at the output. On the other hand, if the noise level surpasses a certain value, then the channel commutation is carried out considering a hysteresis on the SNR parameter, which will be the most common comparison situation.

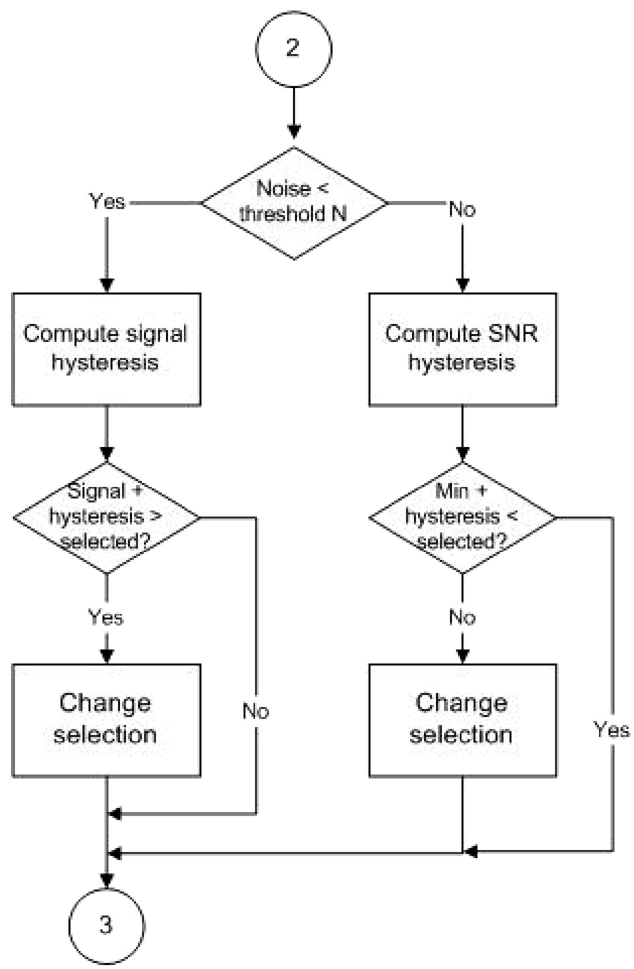

Fig. 8: Maximum SNR selection flow chart

V. CONCLUSION

In this paper we have presented a control procedure for a voting system implemented in a commercial mobile radio system.

The designed channel selection procedure allows considering three different situations depending on the signal and noise values of the input channels. For very noisy channels, the program selects that of minimum noise. For low noise channels, the program just switches when the difference among signal levels is high enough. For any other case, the program selects and switches considering the SNR value.

This simple design is being used with a very acceptable behaviour in the place of exploitation. But one must keep in mind that it is important to keep well calibrated the links that transmit the inputs channels in order to perform the comparisons in a stable way.

ACKNOWLEDGMENT

This work has being carried out with the collaboration of the Spanish company *Servicios de Radio Wavenet S.L.*

It has been partially funded by the Universidad de Alcalá Project PI2005/084.

REFERENCES

[1] [1] Latif-Shabgahi, G. Bass, J.M. and Bennett, S. A taxonomy for software voting algorithms used in safety-critical systems. IEEE Transactions on Reliability, 53 (3), pp: 319 – 328, 2004.
[2] [2] Oppenheim, A.V., Willsky, A.S., Nawab S. H. Signal and systems. Prentice Hall, 1999.
[3] [3] Oppenheim, A.V., and Schafer, R. Discrete time signal processing. Prentice Hall, 1999.
[4] [4] PICmicro (R) Microcontroller
[5] http://www.microchip.com/