

M3Recorder: a MP3 encoding based recorder

J. M^a Flores, M. Montón, A. Núñez, A.J. Velasco

Dpt. Informàtica, ETSE - Universitat Autònoma de Barcelona (UAB)

[josepmaria.flores, marius.monton, antonio.nunez]@campus.uab.es, josep.velasco@uab.es

Abstract

This paper presents the design of the prototype of a MP3 based recorder: the M3Recorder. In this device, the MP3 standard encoding technique is used to compress the registered data. Data compression is the way to reduce storing requirements while preserving audio quality. MP3 encoded data can be fitted into an affordable amount of memory, and makes possible the real time wireless transmission of recorded data. The M3Recorder is conceived as a portable device. Hence, integration and low power consumption become strong design constraints.

1. Introduction

The M3Recorder is a prototype of a portable recorder based on MP3 encoding with wireless transmission capabilities. The M3Recorder is conceived to be a portable device. Therefore, maximum integration and low power consumption are strong designing constraints.

Figure 1 gives a general overview of the structure of the device. There are four modules: the analog input/output module, the MP3 encoder-decoder, the memory module and the communications module.

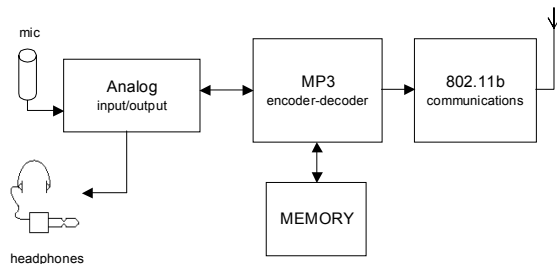


Fig 1. General structure

Roughly speaking, signals arriving from the microphone integrated in the input/output module are processed for noise reduction and digitalized as a continuous data bit stream. This bit stream enters the encoder module, where data is compressed following the MP3 standard. The compressed data is then sent both, to the memory module for internal storing, and to the communications module for a real time wireless transmission using the 802.11b standard. Data stored in the memory module can be reproduced by using the decoding functions and the headphones output. The components of this system are analyzed in the following sections.

This paper is organized as follows: section 2 describes in detail the analog input/output module. Section 3 is devoted to the MP3 encoder-decoder module. In section 4, the communications module is analyzed. Finally, conclusions can be found in section 5.

2. Analog Input and Output

Sound belongs to the analog world. In order to convert such an analog entity into an electric analog signal, an input subsystem is required. Later, this analog signal will be digitalized for advanced processing. In the returning way, an output subsystem converts digitally processed signals to the analog world of sounds.

2.1 General scheme

Figure 2 shows a general scheme of the analog input-output subsystem. The scheme includes the following elements:

- MIC:** board integrated microphone.
- MIC BIAS:** bias voltage required by the microphone.
- PREAMP:** high gain amplifier for the low voltage signal coming from the microphone (mV). A high gain is required to fulfill the whole voltage range.
- LOWPASS FILTER:** to prevent aliasing in the subsequent A/D conversion phase.
- ADC:** to transform the analog input into a digital signal suitable for MP3 compression.
- DAC:** for the reverse conversion when MP3 encoded signals are reproduced.
- POWER AMP:** amplifying phase that DAC output signal requires to be sent to headphones.
- HEADPHONES:** give the best performance when reproducing from small-size low-power devices.

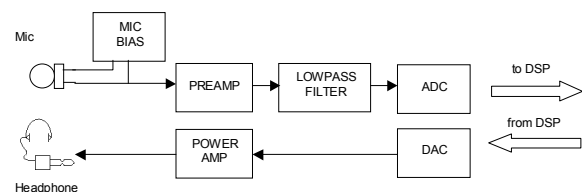


Fig 2. Input/output subsystem scheme

2.2 Design criteria.

The device presented in this paper is not focused on professional music recording, but to the PC peripherals market. In this area the standard is the Audio Codec'97 [1] from Analog Devices, Creative Labs, Intel, National Semiconductor and Yamaha. In this standard, the requirements concerning analog-to-digital conversion are a 48 kHz sampling rate using 16 bit precision samples within the 20-20.000Hz bandwidth.

M3Recorder pretends to be a handheld device. A reduced supply voltage, high value resistors and low-power chips should be used, although satisfying specifications given by the MP3 standard. Furthermore, power-down modes to switch off unused parts would be recommendable.

2.3 Microphone

Sound denotes a variation of air pressure. There is a variety of microphone types to convert this pressure into an electrical signal: carbon, crystal, dynamic, ribbon, condenser and electret microphones. The suitable microphone for this application is characterized by:

- Sensitivity:** the electrical signal amplitude for a given sound pressure. It should be high, to maximize the (usually tiny) output signal thus improving the S/N ratio.
- Output impedance:** to maximize power transfer, load impedance should equal the one of the microphone (several K Ω). Otherwise, impedances should be adapted through a transformer.
- Frequency response:** from 20Hz to 20kHz.
- Operating voltage:** high voltages cannot be afforded due of low power requirements.
- Signal-to-Noise Ratio (SNR):** $\geq 96\text{dB}$, to keep the MP3 16-bit resolution.
- Size:** board integrated microphones must be small.

Frequency requirements discard most types of microphones. In this sense, proper types are dynamic, condenser and electret microphones. Dynamic and condenser types are the choice in

professional music. The drawback of dynamic microphones is the size: they are big and require a transformer. On the other hand, condenser microphones usually require a 48 voltage supply. So, the choice is constrained to the selection of the correct electret microphone. These are rugged, cheap, small and easily integrable on a PCB, but at the expense of a not very good SNR.

The only electret microphone with the required bandwidth (20-20000Hz) is the WM61A from Panasonic. It has $R_{out}=2.2\text{k}\Omega$, $\text{SNR} > 62\text{dB}$ and a sensitivity specification of $-35\pm 4\text{dB re } 1\text{V/Pa}$. This means a fork between 11 and 28mV/Pa , according to $S = 1\text{V} / \text{Pa} \cdot 10^{S(\text{dB})/20}$. The SNR does not achieve our performance spec. However, the difference is hardly perceptible, and it can be improved just by using a better (and bigger) external dynamic microphone.

Next step is biasing the microphone. Electrets have an integrated FET amplifier that must be fed in order to work. However it has only two pins. Therefore, the signal pin must be used at a same time to supply the required voltage to the amplifier, and to get the registered signal. To accomplish this, the voltage is supplied through a resistor $R=R_{out}=2.2\text{k}\Omega$, allowing the reading of the output signal with a coupling condenser. This condenser forms a high-pass filter with the input impedance of the following stage. Provided a $20\text{k}\Omega$ input impedance of the preamp to preserve frequencies $>20\text{Hz}$ the condenser must be:

$$\frac{1}{C \cdot 2\pi \cdot 20} < \frac{1}{10} \cdot 20\text{K}\Omega \Rightarrow C > 4\mu\text{F}$$

2.4 Preamp

The purpose of the preamps is to amplify the small voltages delivered by the microphone. The maximum sensitivity of the Panasonic WM-61A microphone is 28mV/Pa . Fixing the maximum acceptable sound level to $94\text{dBA}=1\text{Pa}$ (the usual in a noisy fabric), the microphone will give output voltages up to $28\text{mV}_{\text{rms}}=79\text{mV}_{\text{pp}}$. Given the 3.3V supply voltage, we choose a working range of 2V_{pp} .

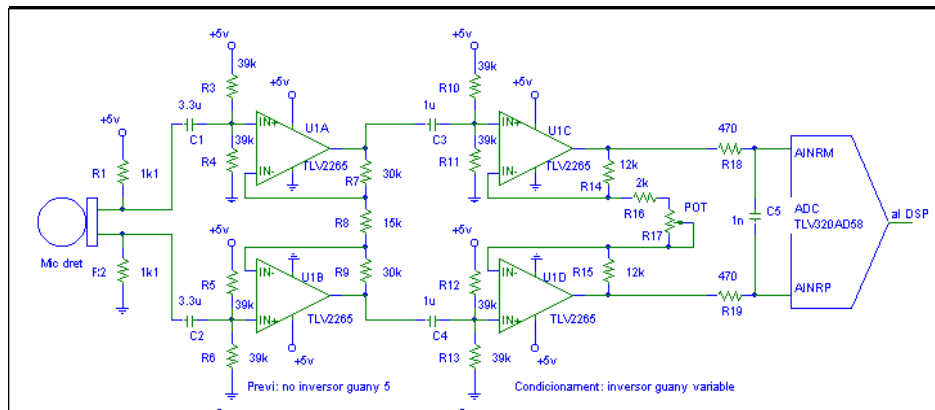


Fig 3. Differential analog front-end

To get this range a $2V/0.08V=25$ gain is needed. The performance of operational amplifiers is strongly reduced with such a high gain. For this reason, two stages of Gain=5 each give a better solution than a single stage of Gain=25.

Audio ADCs usually accept differential inputs, because working with differential signals cancels the effects of common interferences. Single-ended signals can be adapted to differential inputs, but it is better to work directly in a differential way, as proposed in Figure 3.

The suitable operational amplifier should have a Gain-Bandwidth product of $GBW = 6 \cdot 20\text{kHz} = 120\text{kHz}$, a Slew-rate of $0,125 \text{ V}/\mu\text{s}$ (given by a maximum signal of $2V_{pp}$ at 20kHz) and a supply voltage of $3.3V$. To avoid the waste of this already small range, the operational amplifier should be Rail-to-Rail Input and Output (RRIO). While satisfying these conditions, the noise figure must be followed: $2V_{pp}$ and 16 bit resolution means a target resolution of $11\mu V_{rms}$. Distributed with noise over the 20 kHz bandwidth gives a noise spectral density of $62\text{nV}/\text{Hz}$ at the operational amplifier output. Provided a gain of 5, it can be translated as an input noise a little greater than $10\text{nV}/\text{Hz}$. Opamp TLV2465 has $GBW=6.4\text{MHz}$, $SR=0.8V/\mu\text{s}$, $V_{DD}=3.3V$ and $11\text{nV}/\text{Hz}$ noise.

2.5 Low pass filter

This filter must transmit all the frequencies above 20kHz and suppress those over $f_s/2$ to prevent aliasing. MP3 requires 48kSamples/s . Hence, using a conventional ADC forces the filter to have a transition from 20kHz to 24kHz . Fortunately, ADCs with such a bandwidth and resolution are oversampled Sigma-Delta. For an oversampling factor of 128, the sampling frequency reaches $128 \times 48 \text{ kHz} = 6.14 \text{ MHz}$. Therefore, the filter constraint is reached just by suppressing frequencies over 3 MHz , which can be easily performed by a first-order R-C passive filter.

2.6 ADC and DAC

To maintain low power consumption a codec integrating, both, ADC and DAC is used. The TLC320AD77 is used for this purpose since it allows stereo operation with 16 bits- 48kHz sampling rate, $3.3V$ (with low-power mode) and $SNR=100\text{dB}$.

2.7 Power amplifier

Headphones have been chosen because their resistance (32Ω per channel) is higher than in loudspeakers. For $2V_{pp}$ signals, it is 16mW per channel. Even at $3.3V$, any power amplifier can deliver such a small power. Again, the reference is the noise figure. The TPA6100A2 is used because of its $THD+N=0,2\%$, which corresponds to 54 dB . Human sensitivity is rarely beyond 12 dB .

2.8 Codec-to-DSP interface.

The codec needs a master clock of $256 \times 48 \text{ kHz} = 12.288 \text{ MHz}$. The serial port McBSP of the DSP cannot generate this frequency exactly. Therefore, an oscillator is needed. The McBSP will generate the bit and frame clocks to manage TDM transfers between the codec and the DSP from this clock.

3. Audio compression and Coding

As its name indicates, the purpose of this module is the compression of digital audio to a bit stream following the MP3 standard. From the previous input module a continuous data-flow proceeding from 2 separate channels. Each channel has a 48kHz sampling rate and 16 bit resolution, as specified in the Audio Codec'97 standard [1]. The output of this module is a bit stream according to MP3 standard (MPEG1 & MPEG2 Layer III [2]). Therefore, the output is fully compatible with all MP3 decompressors found in the market (Winamp, Real Jukebox, XMMS, etc).

MP3 encoding functions require a high throughput DSP. Decoding functions have lower resource requirements. To reach preliminary results as soon as possible, an Open-Source MP3 encoder has been adapted. Among the available MP3 open-source encoders, LAME [3] has been used because of speed performance on PC platforms (about 200% faster than real-time constrains running on a PIII-500 platform), a full commented source code and a well-suited design for processors with many functional units (as Intel-PIII, or TMS320C6X).

Although the Audio Codec'97 standard [1] specifies the use of floating-point arithmetic, a fixed point TMS320C62 DSP from Texas Instruments (herein, TI) was tried to test the code into a TMS320A6201'C62X prototyping board. This family of DSPs was chosen given the high performance required by the encoder. In addition, the LAME code is parallelizable and, therefore, can take advantage of the 8 functional units present in these DSPs.

Since a fixed-point core is used for prototyping purposes, first results are only approximative. The results obtained will be of great help deciding upon the final core to be used. Thanks to the code compatibility among the TMS320C6X family, no changes will be necessary to the C code.

The most important points performed to successfully accommodate the LAME C code to the test platform have been:

- Fix user defined parameters (filter coefficients, quality compression, data bit-rate).
- Disable VBR (variable bit rate). Fixed bit rate is used because of complexity restrictions.
- Replace input and output routines to work in a stand-alone system, i.e.: no disk access, no

possible seek through input bit stream, output bit stream in a fixed memory address.

To test all this changes a soft prototyping platform based in PC compilation has been used.

A 44.1kHz sampling rate and 16 bits resolution has been used for the tests to follow the Audio Codec'97 standard [1]. However, it has been observed that computational requirements of the MP3 encoder were similar if when choosing other sampling rates, or bit profundity. This is because sampling rates different of 44.1KHz are re-sampled by the encoder to reach the nominal sampling rate. Fortunately, working in monophonic mode, computational requirements descent approximately to the half.

The Texas Instruments compiler used with the TMS320A6201'C62X prototyping board does not seem to take benefit of the 8 functional units of the DSP. It must be taken into account that the source code has been carefully revised to confirm that is designed to be parallelized. In this sense, the problem is not within the source code. On the other hand, the compiler fails to loop unrolling due to the presence of subroutine calls inside loops.

Table 1 shows the number of instructions with concurrent execution upon the number of concurrent units. The first column designs the compiled file (.asm file) tested. Under column #Inst the total number of instructions executed for each compiled file is found. Each following column contains the number of instructions executed using n functional units concurrently (nC).

.asm files	#Inst.	2C	3C	4C	5C	6C
FFT	3918	214	52	11	3	
Main	386	12	1	1		
Lame	5003	317	71	23	11	2
Newdct	5159	529	91	30	12	2
Psymodel	6395	507	94	27	3	1
Quantize	4030	274	77	25	16	4
Reservoir	213	19	1			
Takehiro	1572	99	51	7	2	
Util	802	42	25	3	4	
Vbrquantize	1756	123	32	1	3	1
Vbrtag	1102	66	13	10		1
Versión	127	6	10			
Formatbit	1426	54	9	3	1	1
l3bits	2476	118	16	10	1	
quantize-pvm	2862	221	69	13	5	

Table 1: Number of instructions using concurrent functional units

Although 8 functional units are available in the DSP, no execution has been observed to be using more

than 6 at any time. Furthermore, the concurrent use of 5 and 6 functional units is very low.

All this, seems indicate that an optimization of the C code may result in better performance of compiler results.

Table 2 shows the temporal profiling of the mainly used functions. *Window_subband*, with 16.632 multiplications and 26.460 additions per sample, is the hardest function in the code (40% of time). This function has a poor performance, probably due to a high cache miss, since memory flow is very high in this function.

Curiously, *mdct_long*, with 31.232 multiplications and 31.744 additions but with less memory flow, is occupying only 16.5% of time. A similar situation is found in *outer_loop* function.

	No Opt.	%Time	Opt.	%Time
lame_encode_buffer	1085	100%	1005	100%
window_subband	435,6	40,15%	415,5	41,34%
mdct_long	179	16,50%	179	17,81%
outer_loop	396	36,50%	390	38,81%
calc_xmin	10,4	0,96%	8	0,80%
TOTAL	1021	94,10%	992,5	98,76%

Table 2: temporal profiling of main functions

Accordingly to the results obtained in tests, a floating-point core should be used. The alternative is of transform all source code to efficiently work with a fixed point core. This is revealed as a hard work and should take a quite large developing time. However, this module is a part of a portable device. Therefore, low power consumption of fixed point cores is desired. In addition, a core with at least two McBSP (one for input audio and another for the output bit stream) is needed.

Critical functions may be implemented outside the DSP, using specific hardware (e.g., FPGAs). However the problem of floating point requirements remains.

The choice seems to be the develop of a MP3 encoder focused on a fixed point core. Or, alternatively, wait for manufacturers to develop a really low power floating-point core.

4. Wireless transmission

The bit stream generated by the MP3 encoder is entered to the communications module. The objective of this module is the wireless transmission of a real time flow of MP3 frames. For this purpose, the 802.11b standard is used. This standard was chosen because of the diversity of possible tax rates (see table.3). Real time transmissions are possible with 5.5Mbps and 11Mbps.

The communications module include an specific chipset for wireless applications (the Intersil

PRISM2 chipset), and a FPGA for driver/interface functions. Figure 5 shows the general scheme.

Tax Rate	Code length	Technique	Modulation
1Mbps	11 (sequence Barker)	FHSS/DSSS	BPSK
2Mbps	11 (sequence Barker)	FHSS/DSSS	QPSK
5.5Mbps	8 (CCK)	DSSS	QPSK
11Mbps	8 (CCK)	DSSS	QPSK

Table 3: tax rate specifications for 802.11b standard

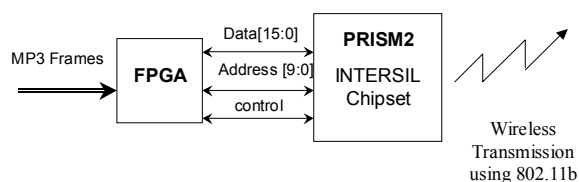


Fig 5. Communications module scheme

The chipset of Intersil consists of 5 elements:

- Hfa3841:** directly interfaces with the Intersil Hfa3824 and Hfa3861. Protocol and PHY support are implemented in firmware to allow custom protocol [4].
- Hfa3861:** Direct Sequence Spread Spectrum (DSSS) base band processor that contains all necessary functions for a full and half duplex packet base band transceiver [5].
- Hfa3783:** a base band converter for half duplex wireless applications [6].
- Hfa3683:** a monolithic SiGe half duplex RF/IF transceiver designed to operate at the 2.4GHz ISM band [7].
- Hfa3983:** a 2.4GHz monolithic amplifier designed to operate in the ISM band [8].

The FPGA is a EPF10K100GC503-4 from ALTERA's FLEX family [9]. It must communicate with the Intersil's interface chip Hfa3841. The logical view of the Hfa3841 from the FPGA is a block of 32 word wide registers. These appear in I/O space starting at the base address determined by the socket controller. The data transmission is across two registers; two independent BAP (Buffer Access Path) which permits concurrent read and write transfers.

Conclusions

This paper presents the design of a MP3 based recorder prototype with wireless transmission

capabilities. The design is divided into four main modules: the analog input/output module, the MP3 encoder-decoder module, the memory module and the wireless transmission module.

The analog input/output module is completely developed. The designed board has already been tested and satisfies the initial perspectives.

The main designing difficulties are in the MP3 encoder module. Low power requirements makes necessary the use of a fixed point core. Tests performed on the TI C62 DSP prototyping board indicate that a completely revised C code for the MP3 encoder over fixed point cores should be developed. However, technology evolution will probably make available, in short, low power floating cores that could be used for this application. The wireless communication is solved by using the 802.11b standard. This standard has enough bandwidth to transmit real time recorded data. Tests have confirmed the expectative.

As a final conclusion, we can state that prototype developed and tests confirm the viability of this application provided a MP3 codec for fixed point cores or a low power floating point DSP for portable purposes.

Acknowledgements

Authors would like to thank support given by CICYT project, TIC98-0410-C02-02, "Codiseño de Sistemas Concurrentes Heterogéneos: Hardware y Aplicación".

References

- [1] INTEL. Audio Codec'97. Revision 2.1. Intel Corporation, 1998.
- [2] <http://www.mpeg.org/MPEG/mp3.html>
- [3] <http://www.mp3dev.org/mp3/>
- [4] Datasheet FN4839.pdf from <http://www.intersil.com>.
- [5] Datasheet FN4856.pdf from <http://www.intersil.com>.
- [6] Datasheet FN4633.pdf from <http://www.intersil.com>.
- [7] Datasheet FN4634.pdf from <http://www.intersil.com>.
- [8] Datasheet FN4635.pdf from <http://www.intersil.com>.
- [9] Datasheet from <http://www.altera.com/literature/lit-f10.html>.