International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-12 Issue-2, January 2023

Low-cost Magnetic Resonance Console Architecture using an Open Source for Laboratory Scale Systems

Chennagiri Rajarao Padma, Ravikumar K. M.

***Abstract:** MRI systems with proprietary hardware must use pulse programming, which is less expensive. Pulse programming consoles use Digital Signal Processor, Complex Programming Logic Device, and microcontrollers, which are typically restricted to particular architectures. General–purpose, extremely affordable electronics board featuring these architectures are now capable enough to be directly implemented in MRI consoles. Here we present the architectural details of various consoles with novel designs and their limitations. Finally, we propose a console design which was created utilising widely accessible Arduino Boards to connect to Pulseq-GPI implementations at a reduced cost of $225.*

***Keywords:** Direct Digital Synthesizer, Microcontrollers, MR Console Architectures, Pulseq-GPI*

I. INTRODUCTION

Magnetic Resonance Imaging (MRI) system consoles have different architectures depending on the type of data and logic processing device, its implementation and the software to run the system. MRI pulse sequences require an intelligent console that coordinates real-time operation of RF and gradient systems, data gathering, and processing. The use of proprietary software, hardware, and interfaces prevents any growth or adjustment to novel methods and experiments. A significant design challenge is posed by meeting contemporary requirements for large number of channels, real-time imaging capability and customizable interfaces. While radio frequency sampling and synthesis are easily handled by Software Defined Radio (SDR), effective data management and real-time control still pose a challenge. Scalable modular system architecture is necessary for the effective and versatile implementation of such systems. Peripheral Component Interconnect (PCI) cards, which offer high-throughput low-latency connectivity at the expense of scalability, cost efficiency, and design time.

Manuscript received on 31 December 2022 | Revised Manuscript received on 07 January 2023 | Manuscript Accepted on 15 January 2023 | Manuscript published on 30 January 2023.
*Correspondence Author(s)

Chennagiri Rajarao Padma*, Assistant Professor, Department of Medical Electronics Engineering, Dayananda Sagar College of Engineering, Bengaluru (Karnataka), India. E-mail: padma-mirc@dayanandasagar.edu, ORCID ID: https://orcid.org/0000-0001-5874-6823

Dr. K. M. Ravikumar, Director, The Oxford Education Institutions, Bengaluru (Karnataka), India. Email:kmravikumar75@gmail.com, directortoce@theoxford.edu, ORCID ID: https://orcid.org/0000-0002-3961-9340

Although performing tasks in real-time with software or a microcontroller may offer considerable flexibility, it places severe limits on programme design and makes wasteful use of the processing power and hardware resources that are available.

II. ARCHITECTURES

The console architectures are classified based on the processing device and its hardware / software capabilities: Microcontrollers (μCs) are self-contained systems with a processor, fixed amount of memory and peripherals on a single integrated circuit. They have the capability of mixed signal processing at reduced costs and programming complexity.

S. Handa et al., have demonstrated a single chip pulse programmer (PPG) using ADμC7026 microcontroller board,[1] which consists of a 32-bit reduced instruction set computer central processing unit (RISC CPU) core(ARM7TDMI) with a clock frequency of 41.78 MHz, 62 kilo bytes (kB) of programming (flash) memory and 8 kB of static random access memory (SRAM), a 16-channel analog to digital converter (ADC)and 4 digital to analog converters (DACs), two timers, and a 40-bit digital input/output (I/O) port. The board was integrated with a personal computer (PC) via the universal serial bus (USB) to RS-232C converter circuit, and on the other end with a MRI transceiver and gradient driver. The programs for the microcontroller and host PC were developed using KEIL development suite and GNU C compiler on Linux emulation platform, respectively. The cost of the board is approximately U.S. $10, with limitations of relatively long delay time between event and event data memory. The work of M. Tsuda et al., interfaced a PC withthree32-bit ARM based RISC μCs, and digital oscilloscopes with an inbuilt and external arbitrary waveform generator (AWG).[2]. The μCs used were Arduino DUE with a clock frequency of 84 MHz, two 12-bit DACs,96kB SRAM and 54 I/O pins. Pico Scope 3205B (250 MHz sampling speed) along with an external AWG-100 (100 MHz sampling speed) was used in one design and an oscilloscope with a built-in AWG (Pico Scope 5242B) (1GHz sampling speed) was used in another. The programs were developed using Qt 5.3.0 and C++ on the host PC which downloads the target programs on the μCs using Arduino Integrated development environment. The total cost of the either console was less than $ 1500. Although under sampling and phase correction were required, the system provided flexibility in pulse sequence design.

Retrieval Number: 100.1/ijitee.B94130112223
DOI: 10.35940/ijitee.B9413.0112223
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

An inexpensive widely available Arduino Duemilanove board based on ATmega 328 µC (16MHz clock frequency) using the arduino open source programming platform, with a single USB connection to the host PC was used for MR spectroscopic measurements in earth's field by Carl A Michal.[3]The µC has 14 digital I/O pins, 6 analog input pins that acted as a pulse programmer, audio frequency synthesizer, ADC and could produce trains of phase coherent radio frequency pulses with reproducible timing allowing signal averaging and phase cycling. The µC was used as a state machine and pre-calculated states and events were downloaded from the host PC as the available memory was limited for to run sequences in real time. The cost of the system is within $ 200 and restricted to spectroscopic sequences.

M. Twieg et al., have demonstrated an open source, fully packed and functional NMR relaxometry platform based on AT32UC3C1512 µC from Atmel.[4] The 32-bit CPU operating at 64 MHz with 512 kB flash, 64kB SRAM, 16 ADC channels and 2 DACs allowed for more sophisticated and configurable pulse sequences, and faster handling of sequence events. The board was integrated with a PC running Matlab and an AD9958 DDS module for RF signals, to define experiments and analyse sampled data. The cost of the chip is less than $ 750.

Meghan E. Halse work on "Terranova" contained a Digital Signal Processor board at the core for pulse programming and signal acquisition functionality that used USB Interface to communicate with the PC.[5] The interfacing software used was Prospa.

Field Programmable Gate Arrays (FPGA) are integrated circuits that contain an array of programmable logic blocks that can be configured using a hardware description language to perform complex combinational functions. They are generally higher in cost and programming complexities compared to microcontrollers but are more versatile.

A home-built MRI system with digital spectrometer was demonstrated by S. Jie et.al.[6] The console design consisted of XC2S200 (Xilinx) RF waveform generator, digital receiver, gradient waveform generator, and pulse generator boards are all logically controlled by an FPGA processor. The boards were interfaced with a PC using PLX9052 chips, serving as a bridge between local bus and PCI bus. In addition, one or two 16-bit SRAMs of 256kB were attached to each board for each board for the purpose of storing data. AD9854 DDS chip, running on a clock frequency of 50 MHz was used for RF Pulse generation. The receiver was based on AD9874 chip for software radio technology, and the gradient board utilizes AD5542 DAC chip for each channel. Delphi 7.0 running on the host PC was used to create the software winMRI, which controls hardware and displays gathered data in real time. The system was implemented at a cost of $ 12000.

K. Takeda displayed that a single FPGA chip was sufficient to work as a PPG, DDS, a digital receiver and PC interfaces for command/data transfer.[7]. The system consists of an Altera EP2C70F672C8 (Cyclone II) FPGA chip controlled by a 30 MHz crystal and a number of auxiliary boards for USB connectivity, direct-digital synthesis (DDS), RF transmission, signal acquisition, etc. Each of the three RF channels in this spectrometer has the ability to modulate the amplitude, phase, and frequency of RF irradiation at frequencies up to 400 MHz. The mother board is surrounded by the other modules. The source codes for the VHDL-written programmes for the core modules that the FPGA uses to execute them in parallel are accessible on the author's website.

Preeti Hemnani et al., have demonstrated in detail the setup of FPGA as a Pulse programmer and a direct digital synthesizer.[8] Their architecture requires a DDS integrated with an external DAC to generate coherent and accurate Phase and frequency modulated pulses for very short intervals. The pulse programmer was mapped as a finite state machine, to generate accurate timing pulses for durations as short as 1µs. Labview was the application software for the FPGA and communication with the PC was established through PCI bus.

H-Y Chen et al., have successfully demonstrated the improved capabilities of the system using Cyclone II, Altera FPGA, operating at 50 MHz.[9] To do away with the requirement for an external DAC, they have integrated the digital frequency generation with an amplifier (MAX4384EUD+; Maxim Integrated, CA) and a bandpass filter. The custom logic blocks are used to run the NMR pulse programme. An FPGA-programmed direct memory access (DMA) controller is used to play back the pulse sequence. By altering the speed of the clock used to time the events of the pulse sequence, even the device's time resolution can be altered in this design. The on-board digitizer provides the data acquisition. The data is read from the memory and sent to a host computer over a USB port using the same processor.

A digital I/O board was used as a pulse programmer by Hashimoto et al., which has a 32-bit input/output lines with a maximum transmit rate of 20 MHz, 32 MB on-board memory, 30 DAC outputs, and a mixer with filter.[10] Cyclone III, ALTERA FPGA was configured as the I/O board, running at a clock frequency of 60 MHz. It communicates with the PC via USB. The interfacing software was developed using C/C++ and .NET framework 2.0 of Visual Studio 2008 running on Windows 7.

In a radio communication system known as Software Defined Radio (SDR), software on an embedded system is used to implement components that were previously implemented in hardware. ADCs, DACs, an FPGA for basic filtering and signal down- and up-conversion, and a USB interface are the typical components of SDRs. SDRs are inexpensive, easy to program and operate at relatively high frequencies. Two architectures based on SDR are discussed.

C. J. Hasselwander et al., assembled a console comprising of two GNU Radio compatible SDRs one for RF excitation and reception and other for gradient pulse generation.[11] The two SDRs are synchronised through pulses generated by the master in the master-slave SDR configuration. The software systems were built in python programming language in GNU radio and the authors have also released a software package implemented for four different sequences on the same platform. The total SDR hardware cost was $2000.

A. Asfour et al., worked on a fully digital RF electronics for the design of NMR systems at low field.[12] The system consisted of a DDS AD9852, for pulse generation, a SDR based on an evaluation board of CLC5902 (National semiconductor) chip, for digital reception of NMR signals and a Digital Signal Processor (DSP) ADSP-2106x SHARC, for system control and for the generation of the gradient signals (pulse programmer). The DDS is used for pulse generation and the SDR is used as a receiver, and the DSP controls the DDS and SDR via its parallel external bus and was also interfaced with a PC. The SDR was integrated as close to the receiving coil to minimize noise and distortion associated with analog mixing stage. The software for the system was developed using LabWindows/CVI environment and DSP assembly language.

Another architecture was designed using Complex Programmable Logic Devices (CPLD's) having complexity between that of programmable logic arrays and FPGAs, with architectural features of both.

Pascal P. Stang et al., demonstrated a scalable console called "MEDUSA" using programmable logic for frequency synthesizing, sampling and synchronization.[13] It consisted of 16 modules over a 16-bit parallel data link with a dedicated logic core and system controller. The hardware included anAltera MAX-II EPM1270 CPLD logic core with identification registers,a Cypress CY7C1061 2Mbyte high-speed SRAM, and DMA. Logic core was responsible for timing, sampling and data flow. The LPC2214 60Mhz 32-bit ARM-7 system controller was responsible for coordination of different modules. Matlab was used for system programming. The cores of the CPLD were developed using Verilog HDL. USB High-Speed (480Mbit/sec) support is implemented using a Cypress Semiconductor CY7C68013A FX2 USB peripheral interface.

Rapid sequence prototyping is supported by the alternative programming environment Pulseq, which is open-source and independent of hardware.

Layton, Kelvin J et al., proposes a new file format that may be used to store hardware events and timing data for MR pulse sequences in an effective manner [14]. The file is converted to the proper instructions for running the sequence on MR hardware using platform-specific interpreter modules. Sequences can be created using a graphical interface or high-level languages like MATLAB.

Keerthi Sravan R. et al., To access to Pulseq, author demonstrated an open-source implementation of Pulseq in GPI Lab [15,16]. Additionally, it makes it possible to combine the pulse sequence design with the other components of the MR research pipeline—simulation, reconstruction, image analysis, and visualization—on a single platform using GPI. Through a gradient remembered echo, the Pulseq-GPI implementation is shown to be capable of constructing all of the sequences that Pulseq currently offers.

Commercially available consoles such as **Portable Lab** is a bench top MRI scanner with hands-on examples optimized for scientific as well as educational use. It isa tool for developing and testing of MR hardware and MR sequences on desktop with six Digital I/Os3MTM Mini Delta Ribbon (MDR); 20 pins (10220-55G3PC) and open MATLAB interface [17] whereas **MEDUSA,** an open system combines distributed processing and buffering with scalability provided by the Universal Serial Bus. Fast programmable logic is used in Medusa's modular design for sampling and synchronization.[13]. Another popular MRI system is **Terranova- MRI: Earth's field MRI teaching system [18]**, which is straightforward, portable, and user-friendly. It provides a fantastic starting point for education for new researchers in the field of NMR and was created specifically to study a variety of contemporary pulsed FT NMR and MRI experiments and concepts.

III. IMPLEMENTATION

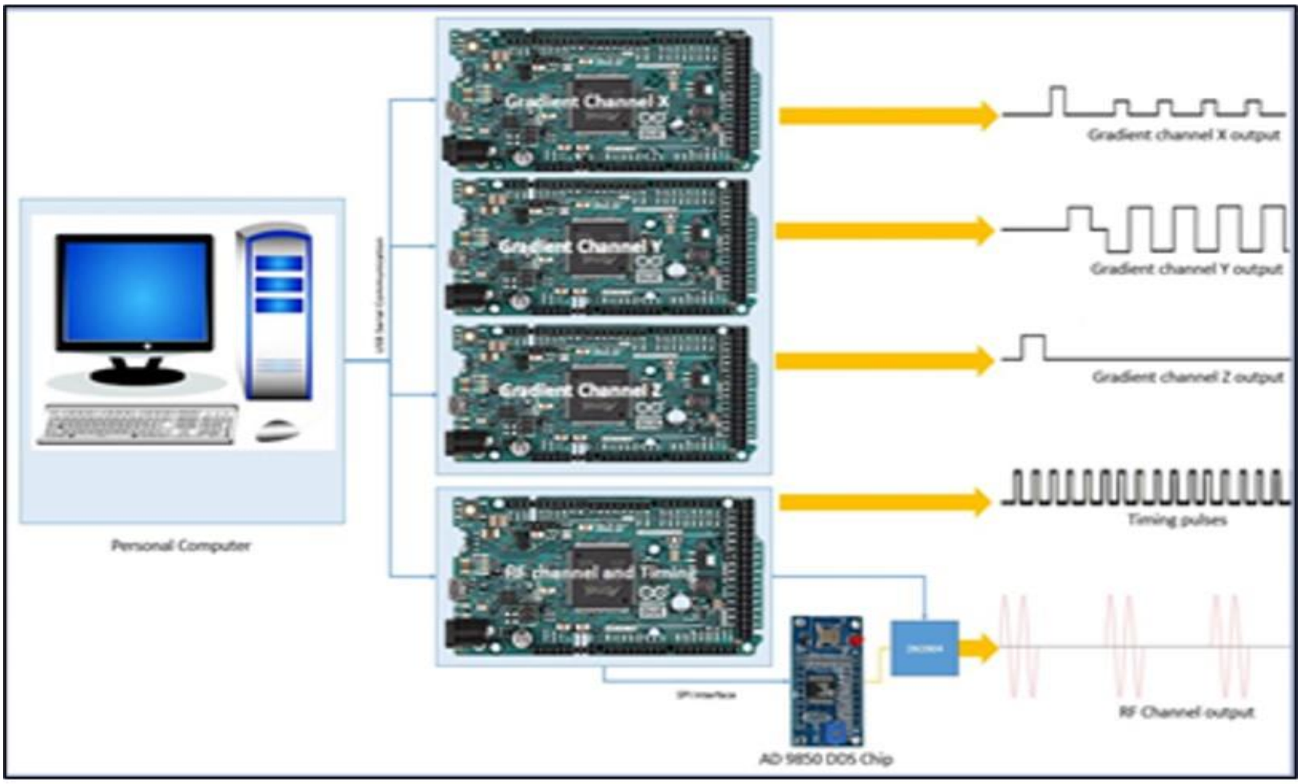
The reported pulse programming console implementations make use of DSPs, complex programming logic devices (CPLDs), and microcontrollers, which are often limited to specific architectures and require more time and money to create. As a result, at a reduced cost of $225, MR consoles for laboratory scale systems on an open-source platform were constructed for 9.5mT using accessible Arduino Boards to connect with Pulseq-GPI implementations. Data extracted from the Pulseq-GPI, stored and uploaded as a text file will support the microcontroller to play a 5s dwell period RF pulse and gradient waveforms (GX and Gy). At this moment, efforts are being made to integrate an analogue to digital converter for the Gradient Recalled Echo (GRE) sequence and speed up the uploading of the full sequence's waveforms.

A. Methods:

1. Hardware description: As illustrated in figure 1a, the console is made up of 32-bit general-purpose Arduino DUE boards, for each gradient channel Gx and Gy, for timing and regulating radio frequency (RF) pulses, and one board that can be readily expanded to include the third gradient channel (Gz). The AD9850 Direct Digital Synthesizer chip was connected to the RF-timer board, and it was programmed to produce a 405kHz RF sine wave, the output of which was sent to the collector of a 2N3904 transistor. Rectangular RF pulses were produced by switching the transistor's output with RF-timer pulses. The gradient waveforms were played out on the internal Digital to Analog Converter outputs of the gradient boards Gx and Gy. To upload waveform data, each board is connected to a personal computer via the USB serial communication protocol.

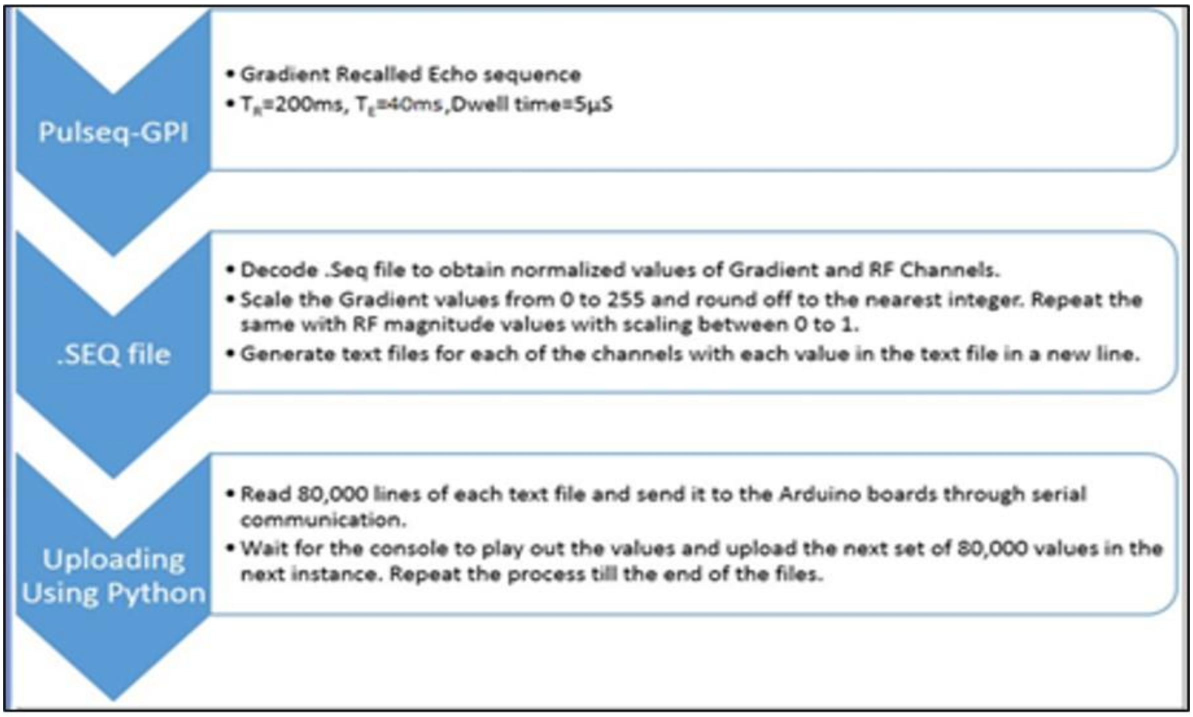
2. Software implementation:

As seen in figure 1b, the Pulseq-GPI based implementation creates a .seq file that has been decoded into three distinct gradient files and one RF pulse file. These files were then uploaded to each of the corresponding boards using Python running on a PC.



(a)

Figure 1 (a) This shows X, Y, Z gradient channel boards with outputs of each channel and RF channel. A serial peripheral interface (SPI) is connected to AD 9850 DDS chip. This chip is controlled by RF and timing Arduino board.



- Gradient Recalled Echo sequence
- T_R=200ms, T_E=40ms,Dwell time=5µS

- Decode .Seq file to obtain normalized values of Gradient and RF Channels.
- Scale the Gradient values from 0 to 255 and round off to the nearest integer. Repeat the same with RF magnitude values with scaling between 0 to 1.
- Generate text files for each of the channels with each value in the text file in a new line.

- Read 80,000 lines of each text file and send it to the Arduino boards through serial communication.
- Wait for the console to play out the values and upload the next set of 80,000 values in the next instance. Repeat the process till the end of the files.

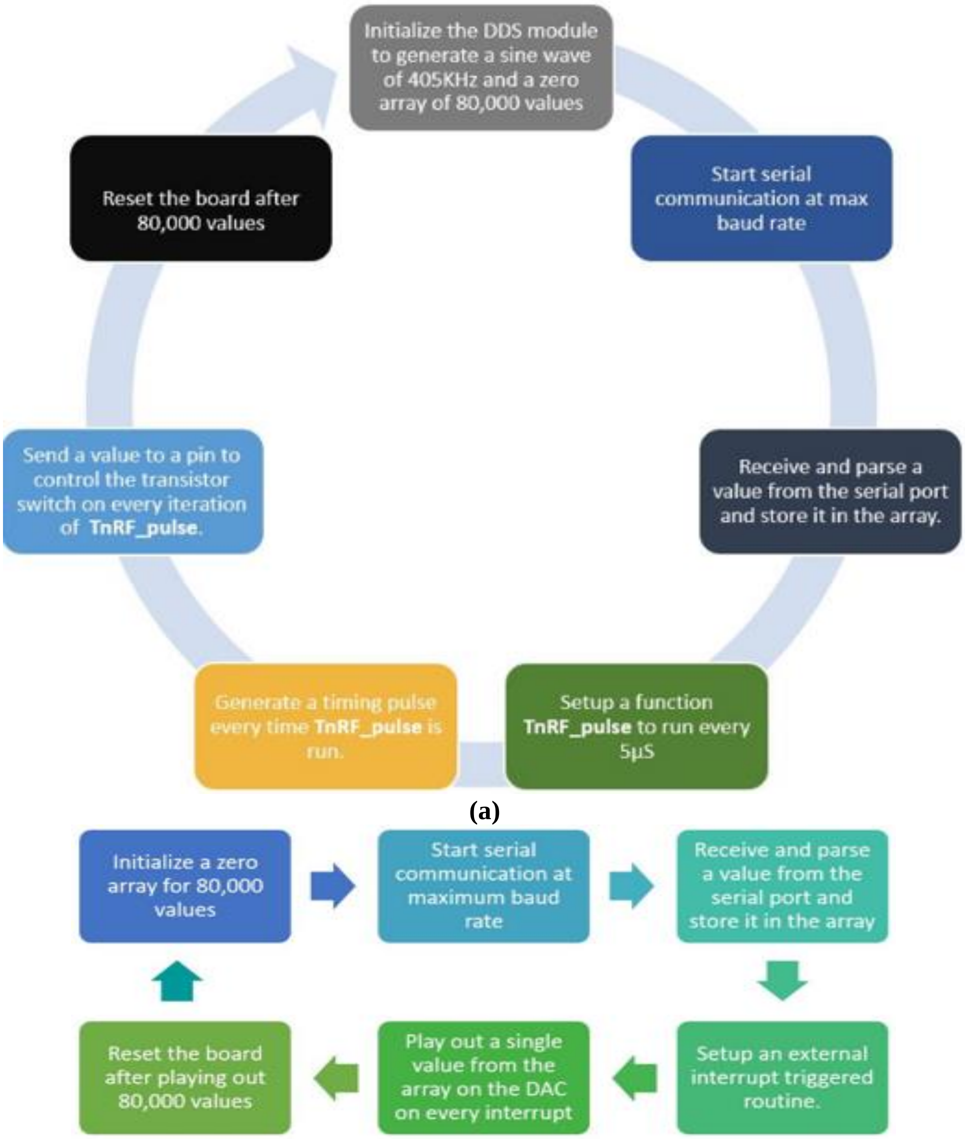
(b)

Figure 1 (b): Flow chart shows the events from Pulseq- GPI to generation of RF and Gradient waveforms.

Retrieval Number: 100.1/ijitee.B94130112223
DOI: 10.35940/ijitee.B9413.0112223
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

In one occasion, 80,000 values, or data for two TR, were transferred to the boards and played out, taking around 11.2 seconds. Using scheduled interrupts from the DUE board, the timer produced precise synchronization pulses every 5 s for each instance. The transistor switch is controlled by the RF timer board based on the stored magnitudes of RF pulse as illustrated in figure 2a. The gradient boards are interrupt-driven and generate voltage values on their DACs with externally triggered interrupts, are illustrated in figure 2b.



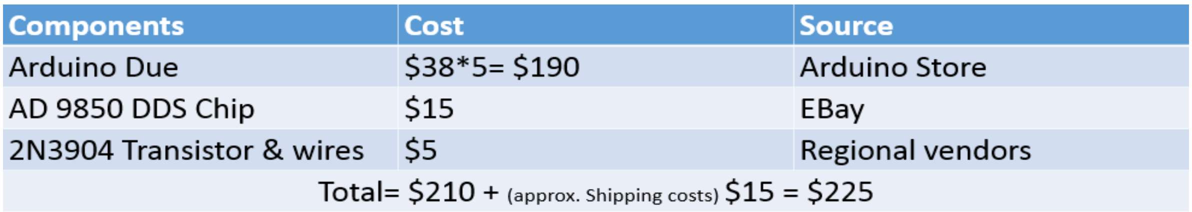
(a)

(b)

Figure 2: (a) shows the program flow diagrams for the Timer-RF of channel from Arduino DUE board and (b) shows the program flow diagram of Gradient Channel Board.

The system, consists of three boards and can play any two gradient channels together with an RF channel. This may be extended to include a fourth board for a third gradient channel. Figure lists the price of the console being created.

3. Cost of components:



Components	Cost	Source
Arduino Due	$38*5= $190	Arduino Store
AD 9850 DDS Chip	$15	EBay
2N3904 Transistor & wires	$5	Regional vendors
Total= $210 + (approx. Shipping costs) $15 = $225		

Figure 3: Arduino Due boards for three gradient channel, ADC channel board, a timer board for RF channel are listed. They are sourced from regional vendors with a cost of $225.

B. Results:

The designed hardware is a cost-effective solution to generate three Gradient and RF Channel waveforms. Hardware implementation using Arduino reduces the complexity of the MR pipeline design and development processes. Figure 4a shows the plot of uploaded data and Figure 4b and 4c shows 2 TRs with a dwell time of 5µs for 400ms of RF channel and Gradient channel waveforms. The RF channel and Gz channel output is illustrated separately in figure 4d & 4e, similarly Gx and Gy values are displayed in figure 4f.

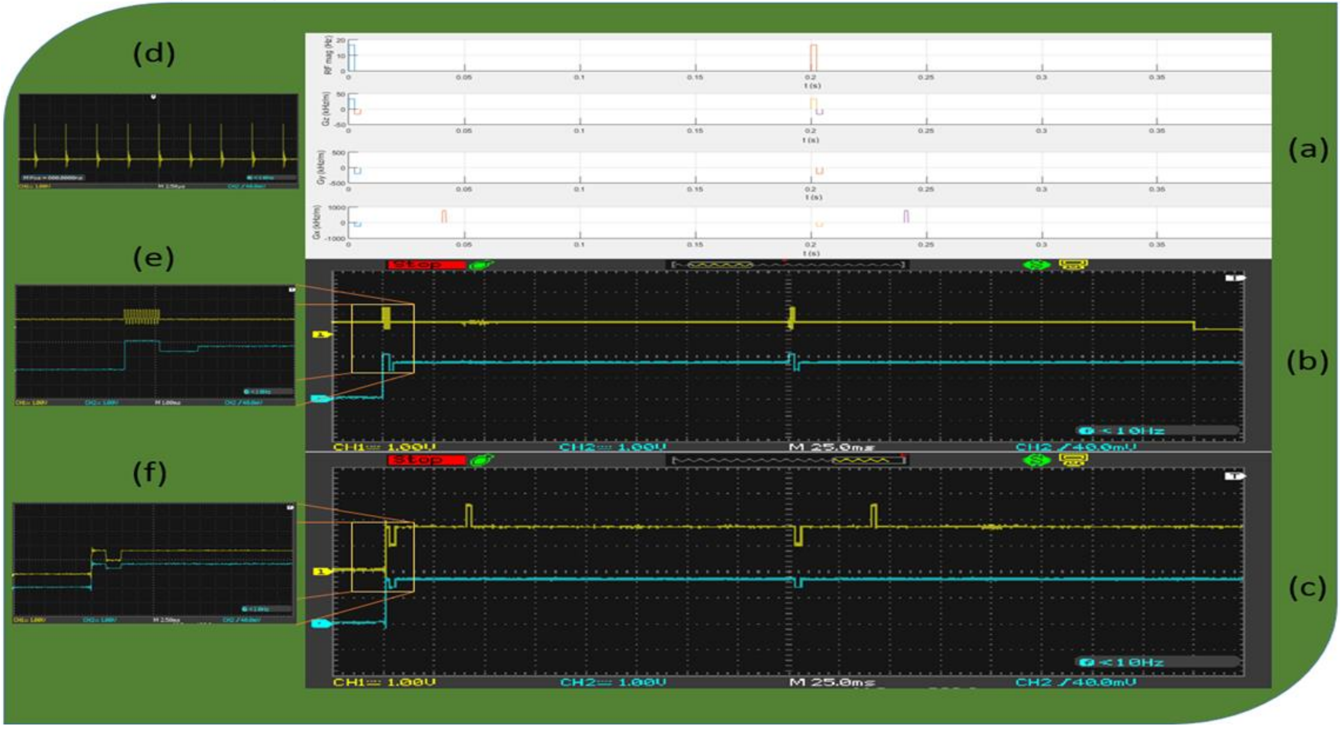


Figure 4: (a) shows the plot for the repetition time (TR) of 400ms for three gradient waveforms with RF channel shown separately. (b) Timing diagram of Gz channel (blue) and RF channel (yellow) (c) X Gradient (yellow) and Z gradient (blue) are shown. (d) Synchronization pulses of 5µs from RF channel (e) Timing diagram of X- Gradient (blue) and RF channel (yellow) (f) Timing diagram of X- Gradient (yellow) and Y Gradient (blue).

IV. DISCUSSION AND CONCLUSION

The system provides an affordable way to produce waveforms created using Pulseq-GPI. The system is being integrated with an ADC (for receive) in order to reduce the time required to upload the data for one instance (400ms) and/or interface the system with external memory cards to store larger sequence values. Future work involves interfacing the console with coil driver apparatus and integrate with low field lab scale MRI systems.

ACKNOWLEDGEMENTS

We would like to thank Dr. Sairam Geethanath, Director, Medical Imaging Research Centre, DSCE for the support and guidance extended during the work.

DECLARATION

Funding/ Grants/ Financial Support	No, I did not receive.
Conflicts of Interest/ Competing Interests	No conflicts of interest to the best of our knowledge.
Ethical Approval and Consent to Participate	No, the article does not require ethical approval and consent to participate with evidence.
Availability of Data and Material/ Data Access Statement	Not relevant.
Authors Contributions	Chennagiri Rajarao Padma, Methodology, Software, investigation, Visualization, Validation, Writing the draft K.M. Ravikumar, Conceptualization, Validation, review and editing, and Supervision

REFERENCES

1. Handa, S., Domalain, T. and Kose, K., 2007. “Single-chip pulse programmer for magnetic resonance imaging using a 32-bit microcontroller.” Review of scientific instruments, 78(8), p.084705. [CrossRef]
2. Tsuda, M., Tamada, D., Terada, Y. and Kose, K., 2016. “Development of Digital MRI Consoles Using General-Purpose Digital Instruments and Microcontroller Boards.” Applied Magnetic Resonance, 47(8), pp.847-858. [CrossRef]
3. Michal, C.A., 2010. “A low-cost spectrometer for NMR measurements in the Earth's magnetic field.” Measurement Science and Technology, 21(10), p.105902. [CrossRef]
4. Michael Twieg, Matthew J Riffe, Natalia Gudino, and Mark A Griswold“An Open Source Mobile NMR Relaxometry Platform,” ISMRM 2013
5. Halse, Meghan Eileen. "Multi-Acquisition and Multi-Dimensional Earth's Field Nuclear Magnetic Resonance Spectroscopy." (2009).

6. Jie, S., Qin, X., Ying, L. and Gengying, L., 2005. “Home-built magnetic resonance imaging system (0.3 T) with a complete digital spectrometer.” Review of scientific instruments, 76(10), p.105101. [CrossRef]
7. Takeda, Kazuyuki. "OPENCORE NMR: Open-source core modules for implementing an integrated FPGA-based NMR spectrometer." Journal of Magnetic Resonance 192.2 (2008): 218-229. [CrossRef]
8. Hemnani, Preeti, A. K. Rajarajan, and S. V. G. Ravindranath. "FPGA based RF pulse generator for NQR/NMR spectrometer." Procedia Computer Science 93 (2016): 161-168. [CrossRef]
9. Chen, Hsueh-Ying, et al. "An ultra-low-cost NMR device with arbitrary pulse programming." Journal of Magnetic Resonance 255 (2015): 100-105. [CrossRef]
10. Hashimoto, Seitaro, Katsumi Kose, and Tomoyuki Haishi. "Development of a pulse programmer for magnetic resonance imaging using a personal computer and a high-speed digital input–output board." Review of scientific instruments 83.5 (2012): 053702. [CrossRef]
11. Hasselwander, Christopher J., Zhipeng Cao, and William A. Grissom. "gr-MRI: A software package for magnetic resonance imaging using software defined radios." Journal of Magnetic Resonance 270 (2016): 47-55. [CrossRef]
12. Asfour, Aktham, Kosai Raoof, and Jean-Paul Yonnet. "Software defined radio (SDR) and direct digital synthesizer (DDS) for NMR/MRI instruments at low-field." Sensors 13.12 (2013): 16245-16262. [CrossRef]
13. Stang, P. P., Conolly, S. M., Santos, J. M., Pauly, J. M., & Scott, G. C. (2012). “Medusa: a scalable MR console using USB”. IEEE transactions on medical imaging, 31(2), 370-379. [CrossRef]
14. Layton, Kelvin J., Stefan Kroboth, FengJia, Sebastian Littin, Huijun Yu, Jochen Leupold, Jon-Fredrik Nielsen, Tony Stöcker, and Maxim Zaitsev. 2017. "Pulseq: A rapid and hardware-independent pulse sequence prototyping framework." Magnetic resonance in medicine, 77(4), p. 1544-1552 [CrossRef]
15. Keerthi Sravan R, Imam Ahmed Shaik, Stefan Kroboth, Maxim Zaitsev, and Sairam Geethanath.”Implementation of Pulseq in GPI Lab” ISMRM 2017.
16. Ravi, Keerthi Sravan, Sneha Potdar, Pavan Poojar, Ashok Kumar Reddy, Stefan Kroboth, Jon-Fredrik Nielsen, Maxim Zaitsev, Ramesh Venkatesan, and Sairam Geethanath. "Pulseq-Graphical Programming Interface: Open-source visual environment for prototyping pulse sequences and integrated magnetic resonance imaging algorithm development." Magnetic resonance imaging 52 (2018): 9-15. [CrossRef]
17. http://pure-devices.com/images/Flyer/Brochure_Portable_Lab_web.pdf
18. http://www.magritek.com/applications/nmr-and-mri-education

AUTHORS PROFILE

Prof. Padma C.R. is having two decades of experience in teaching in Department of Medical Electronics Engineering, Dayananda Sagar College of Engineering, Bengaluru, and also into research from past ten years. She has worked in the project “National Mission on Indigenous MRI”, funded by Ministry of Electronics and Information Technology, Government of India and student projects funded by KSCST. She has been awarded with E. K. Zavoisky stipend from the committee of International Society of Magnetic Resonance in Medicine (ISMRM) Annual Meeting held during 16 – 21st June 2018, Paris and Stipend from the committee of ISMRM workshop held during 29th to 31st March 2019 held at India International Centre, New Delhi. She has publications in several national and international conferences and journals. She is a life member in National Magnetic Resonance Society (NMRS), International Society of Magnetic Resonance in medicine (ISMRM) and International Society of Magnetic Resonance in medicine (ISMRM) Indian chapter. She has certifications from NPTEL and Coursera to her credit and she has been part of the training sessions to Siemens employees on Magnetic Resonance Imaging.

Dr. K. M. Ravikumar, Senior IEEE member is presently working has Director at the Oxford Educational Institution Bengaluru. He has a total of 23 years of teaching experience and published around 55 papers in International / national journals and Conferences. He has also worked has a special officer at VTU regional office, Mysore and he was also “Sub Committee -2 member of Task Force on implementation of National Education Policy-2020” (Governance & Regulation in Higher Education)- Karnataka Higher Education Parishat. He has guided 8 research scholars for their PhD work and guiding 5, as on today. He is also member for NBA and NAAC, as an assessor. He is member of many professional bodies including FIE, LMISTE, SMIEEE, etc. He is an “Executive Committee Member” of Sir M Visvesvaraya Memorial Foundation, Bangalore and Exe-Com member of IEEE-RAS. He is also Member of Local Project Committee (LPC) of Seva-in-Action (SiA), National Trust, Ministry of Social Justice & Empowerment, GOI.