International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-8 Issue-4, November 2019

Implementation of Pll in Rf Transceiver using SPI and Uart Protocols

V.Kumara Swamy, Prabhu G Benakop

***Abstract:** Software Defined Radio plays vital role in many applications as the components in it are software selectable. We can select desired frequency and modulation technique which can be selected through software. The desired frequency selected should be locked in phase locked loop (PLL). The desired frequency is selected by giving commands from Graphical User Interface (GUI) using Universal Asynchronous Receiver Transmitter (UART) and Serial Peripheral Interface (SPI) protocols.GUI is created using Qt creator which is a cross platform C++ and java script Integrated Development Environment (IDE). GUI is designed to generate the desired frequency. As soon as we select a frequency, the corresponding address and data are generated to configure in the Radio Frequency (RF) transceiver. These address and data are first sent to the PIC microcontroller by communicating through UART protocol and after setting data format, these are sent from Peripheral Interface Controller (PIC) to RF transceiver by communicating through SPI protocol. With this process, the registers in RF transceiver are controlled by the user*

***Keywords:** GUI, PLL, UART, SPI*

I. INTRODUCTION

Software defined radio is a conception which RF communication is achieved by using software or firmware to perform signal processing tasks that are to be performed by hardware[1][2]. The range of frequencies can be selected depends on the RF transceiver. According to RF transceiver, we used the range of frequencies from 0.3GHz to 3.8GHz. As per our requirement, we can select any frequency that to be locked in PLL section of RF module through GUI. After selecting the desired frequency in GUI, It will automatically calculate the address and data required to generate the corresponding frequency. These address and data are sent to PIC microcontroller from GUI through UART port and then to RF transceiver through SPI. The bits of data from UART to SPI can be transmitted either in parallel or serial form. In parallel communication, the bits of data are sent all at once, each one on a separate wire. In serial communication, the bits are sent one at a time on a single wire.

II. THEORETICAL ANALYSIS

Block diagram of Implementation of PLL in RF transceiver using SPI and UART protocols is given in fig.1. It represents the overall blue print of the design.

Manuscript published on November 30, 2019.
* Correspondence Author
V.Kumara Swamy*, Dept of ECE, Sreenidhi Institute of Science and Technology, Ghatkesar, Hyderabad, Telangana, India, Email:ksvarkuti@gmail.com,
Dr. Prabhu G Benakop, Dept of ECE, Methodist College of Engg, Abids, Hyderabad, Telangana, India, pgbenakop@rediffmail.com

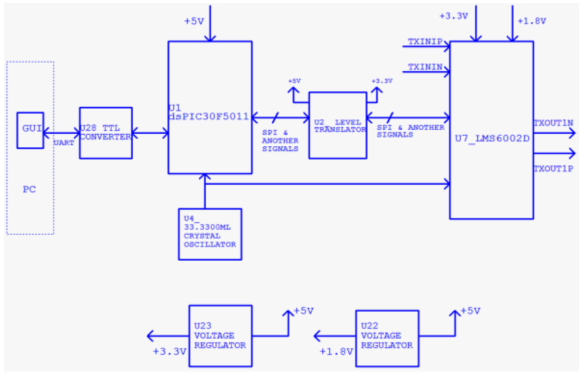

Fig.1. Block diagramof PLL in RF transceiver.

A. GUI

GUI is developed in the personal computer(PC) by using QT creator. GUI has check up boxes to check or uncheck the options, list box to select one option from listed options and editors boxes to provide input from the user[5].

B. TTL converter

Succesive block is TTL converter which is a hardware component used is to make PC compatible with the PIC micro controller.CPU has TTL architecture but whereas PIC micro controller has CMOS architecture. In order to make PC compatible with PIC micro controller we use TTL converter.

C. dsPIC30f5011

The dsPIC microcontroller is a part of microchip's 16-bit microcontroller family[4]. dsPIC has modified harvard architechture. It is 64 pin with QFN package (Quad flat no leads package). It has inbuilt two UART with FIFO buffers and 3-wire SPI module that are needed for communication in our design. We can operate PIC micro controller in interrupt modes for transmission and reception. UART transmission is done by interrupt mode in our design.

D. Voltage regulator

PIC micro controller requires operating voltage of 5v to operate and LMS6002D have operating voltage of 3.3v.To interface PIC controllerwith LMS6002D there is a need of voltage regulator.

E. RFTransceiver (LMS6002D)

This transceiver IC with DQFN package operates from 300MHz to 3.8GHz.The LMS6002D has a standard Serial Port Interface (SPI) for programming.

F. Crystal oscillator

The operating frequency needed for the operation of PIC MICRO and RF transceiver is supplied through crystal oscillator.

G. PLL

PLL stands for phase locked loop. PLL is an electronic circuit that generates the same frequency as the input. Generally a PLL output frequency is controlled by a voltage controlled oscillator (VCO) whose input is an error voltage signal. Error signal is the phase difference between input signal and frequency generated by VCO[3]. VCO go on changing the output until error becomes negligible. RF transceiver(LMS6002D) has inbuilt PLL which can be software controlled. The address to be configured in LMS6002D is 0x95 with desired properties.

H. Protocols

To have an interaction and communication between devices there is a need of set of rules which are called protocols. These set of rules are called communication protocols. The most commonly used protocols areSPI, UART and I2C (Inter-Integrated Circuit). Out of which, two important protocols i.e., SPI and UART are developed in our design[6]

I. Universal Asynchronous Transmitter and Receiver (UART)

UART is asynchronous communication protocols generally used to communicate between two devices. It differs from other protocols that it has only two wires and is asynchronous (No clock is used). Instead of clock we use baud rate and BRG Value to synchronize the two devices where the communication should take place. Here the communication takes place between the UART module in PC and UART module in PIC micro controller.

J. UART in PIC

PIC controller has UART module which supports full duplex 8 or 9-bit communication, Odd, even or no parity communication is available, One or two stop bytes, 4 word deep transmit and receive buffers, Separate transmit and receive interrupts.

UART in PC communicates with the UART second module in PIC microcontroller. 8-bit transfer of data takes place in this communication.

K. Serial Peripheral Interface (SPI)

SPI is the synchronous communication protocol which has 4 pins (SEN, CLK, MOSI and MISO)[5]. This has a master and a possibility to select one from a multiple slaves. Here as we use only one transceiver, we need only one slave mode. Here the master is PIC as it controls the RF device and obviously RF transceiver is the slave. This can either be used in 3-wire or 4-wire mode. The benefit of SPI communication is that it sends data without any interruption.

L. SPI in PIC

SPI module in PIC micro controller consists of a 16-bit shift registers SPIxSR for shifting the data in and out of the module, SPIxBUF buffer register, SPIxCON control registers are for configuring the module. The serial interface has SDIx, SDOx, SSx, SCKx.(where x can be 1 or 2) SCK is for clock output. In transmission mode series of 8 or 16 clock pulses shifts out bits from the SPIxSR to SDOx pin and simultaneously shifts in data from SDIx pin. An interrupt is generated when the transfer is complete and the respective interrupt flag bit (SPI1IF or SPI2IF) in is set. Transmission and reception can be done only when SSx pin is at low i.e, SSx is active low pin.

M. SPI in LMS6002D

All registers that can be configured in LMS6002D are 8-bit wide. Write/read operations consists of 8-bit instruction followed by the 8-bit data to write or read. MSB bit of the instruction bit stream is used as SPI command for read and write operations where CMD=1 for write and CMD=0 for read. Other 7-bits of the instruction represent register address that is to be configured. In RF transceiver timing diagrams of read and write operation are shown in fig.2 and fig.3.

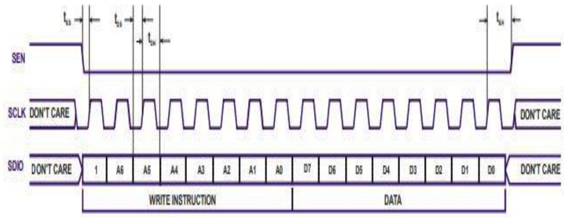

Fig.2. Timing diagram for write operation

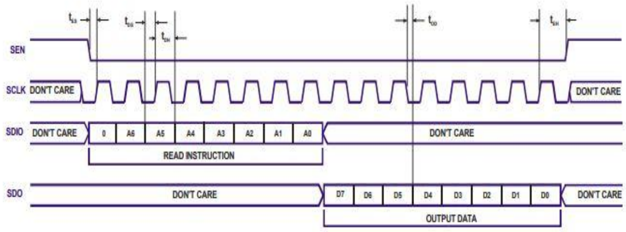

Fig.3. Timing diagram for read operation

III. EXPERIMENTAL SETUP

Hardware components used in the design are dsPIC30f5011,LMS6002D, crystal oscillator, voltage regulator, pickit3 and UART cable. dsPIC30f5011 controller which can be controlled by the commands given through GUI.Pickit3 is an in circuit debugger used for dumping the code into the PIC micro controller. Pickit3 has an input port that is to be connected to PC and 4-wire output architecture is connected to PIC micro controller. Then code is dumped into the PIC. These all connection steps are for program dumping, after this pickit3 connection is removed. UART cable with 4-wire is connected from PC to PIC.4-wires of the UART cable are ground, supply, transmit and receive. Transmit pin of UART is connected UART second port receive pin(U2RX) and receive pin of UART cable is connected to UART second port transmit pin(U2TX).Ground connection of UART in PC and ground pin in the UART module must be connected for communication. SDI2, SDO2 which are SPI module pins in PIC micro controller are connected to SDO and SDIO pins in LMS6002D.

IV. PLL IMPLEMENTATION

In the first step the commands from the GUI (PC) are given to the PIC microcontroller through UART. Here UART receives the data in interrupt mode. If the first character of received data is A5 then it consider A5 as start byte, treat it as appropriate data and starts receiving 98 bytes till it receives stop byte (CC).

If the first character of received data is not A5 then GUI resends the data till PIC microcontroller receives correct data. UART sends 8-bit data but for configuration of registers in SPI we need 16-bit so there is a need to set data format. After receiving the 98 bytes of data, we set the data format as first byte is address and second byte is data. After setting the data format in PIC microcontroller we send the 16 bit word to the RF transceiver through SPI protocol. RF Transceiver reads the 16 bit word and change VCO regulator output voltage control to tune the capacitor in PLL. In order to lock the frequency in PLL we should tune VCOCAP value from 111111 in 0x19 register[4]. At some point of time frequency will be locked in PLL..0x1A is the address in RF transceiver in which there are status bits that checks whether the frequency is locked or not.Every time when the VCOCAP is changed, status bits are checked and if it does not satisfy the status bits again VCOCAP is changed until frequency is lockedThe UART present in PIC micro Controller has a Baud Rate Generator of 16 bit size to have maximum flexibility in generation of baud rate.

As shown in Fig.4 the flow chart of the source code that configures PLL in the RF transceiver.

The UxBRGis the baud rate generator register in LMS6002D which is readable and writable. The baud rate is to be calculated using below formula

BRG value = 16-bit value in UxBRG register.

FCY = Instruction Clock Rate (1/TCY). The formula relating Baud rate and BRG value is shown below.

$$X= Y/16*(Z +1)$$

$$Z= (Y/(16*X))-1$$

Where X= Baud rate, Y= FCY, Z= BRG

Both the transmitter and the receiver should have the same baud rate to communicate with each other.

V. RESULTS AND DISCUSSION

Software used in our design is MPLAB.MPLAB X is a software program that runs on a PC with the OS Windows, Mac OS, Linux to develop applications for Microchip microcontrollers and digital signal processors. An Integrated Development Environment provides a single environment to develop code for embedded microcontrollers. MPLAB with 3.65 version is used in our design. Steps to write and build the design.

Step-1: Select pickit3 for program in hardware tool and XC16 compiler and click on OK.

Step-2: Select Microchip Embedded in categories and projects as standalone project. Click on next.

Step-3: Select family as 16 bit DSC's (dsPIC30f) and device name as dsPIC30f5011.

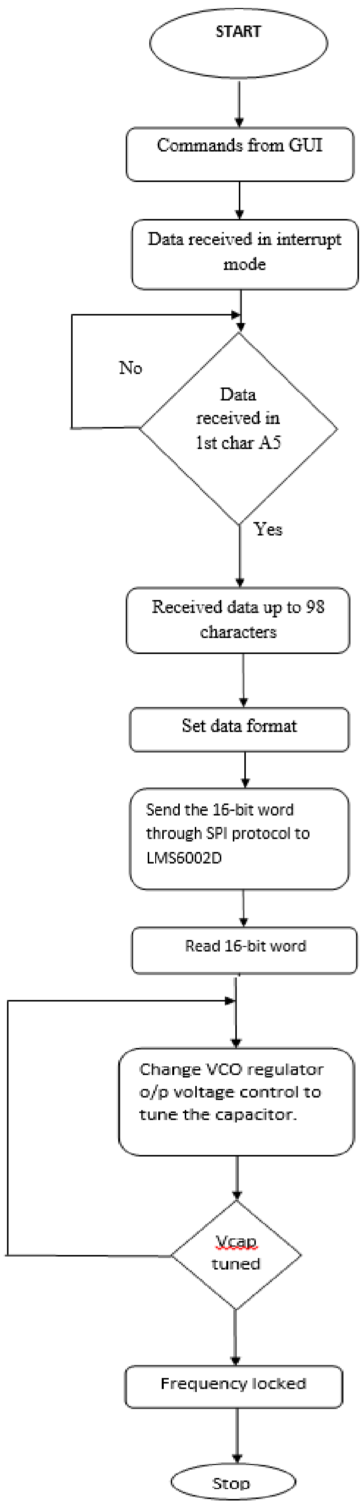

Fig.4. Flow chart of the PLL configuration through GUI

Step-4: Select pickit3 again and click on next.

Step-5: Give the name to the project and project location. Tick on "set as main project". Click on finish.

Step-6: Write the code in the editor.

Step-7: Click on the build icon and check for the errors.

After following the steps referred in MPLAB tool explanation and hardware connections referred, Click on “make and program device project” icon to dump the code.
Open GUI, a window is displayed as shown in fig.5.

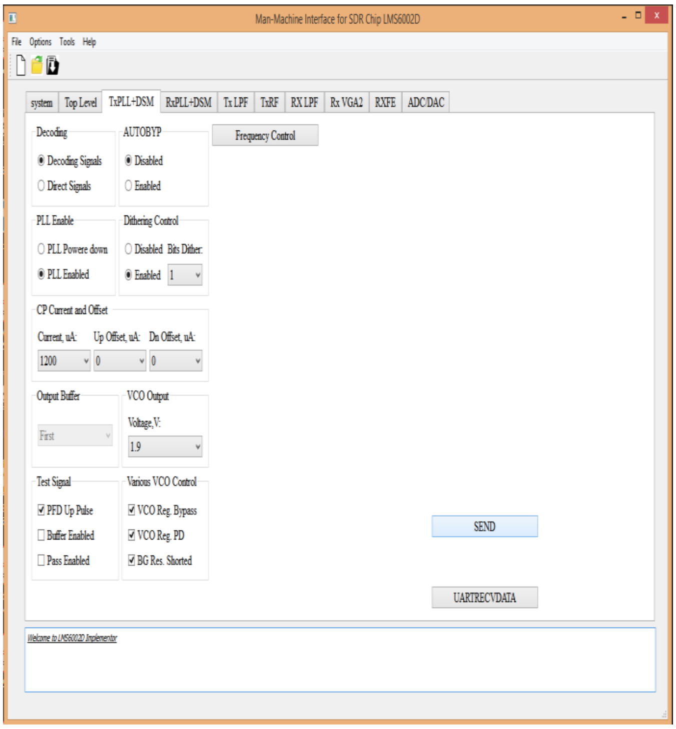

Fig.5. GUI window

- Click on the SEND button on GUI .
- To display data received through UART click on UARTRECDATA button on the GUI window. Fig.6 shows window displaying UART received data.
-

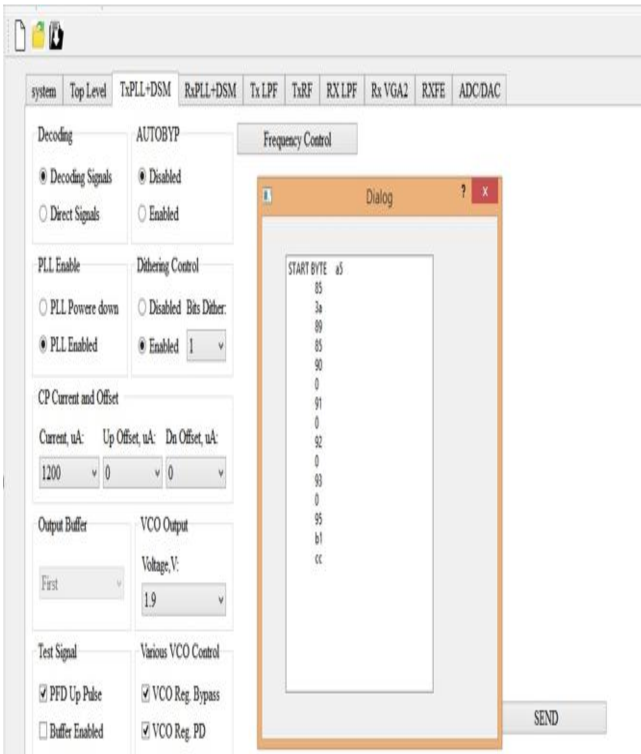

Fig.6. Address and data sent through the GUI

- For displaying SPI communication we need to dump the other corresponding program for displaying SPI. In the code changes are made such that reception is done through LMS6002D, where CMD bit is set to zero and address is sent. Data in the corresponding address is received and displayed through UART, fig.7 shows corresponding data for a address given in the code
- Address shown in fig.6 are a5,3a,85 and corresponding data are 85,89,90.

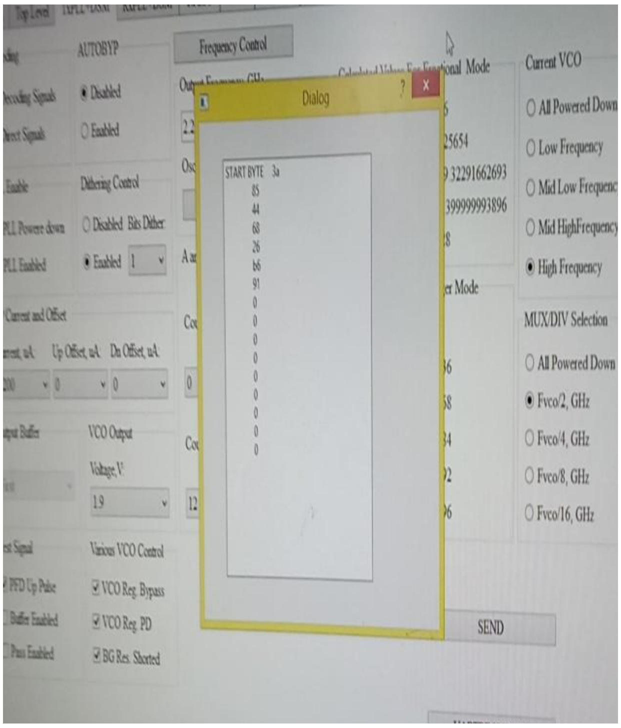

Fig.7. Data present at the given address (address given through GUI)

When the frequency in the PLL is locked then the device gets activated and draws more current. Fig.8 shows voltage and current readings before PLL is locked and fig.9 shows readings after PLL is locked.

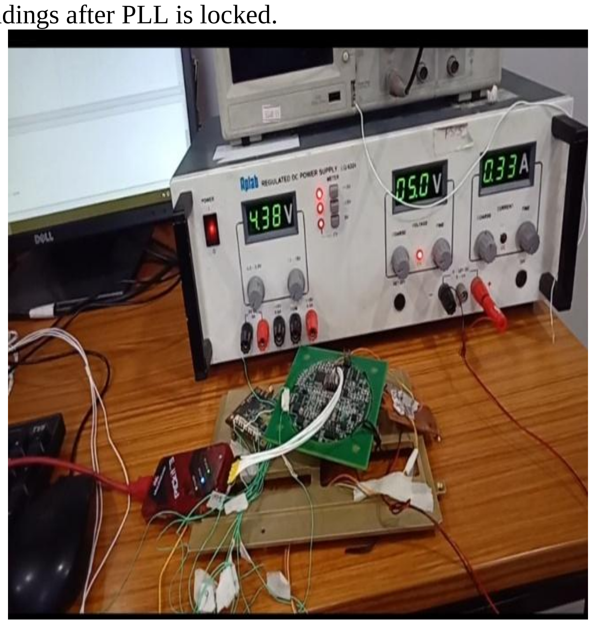

Fig.8. Voltage and current readings before PLL is locked

Fig.9. voltage and current readings after PLL is locked

VI. CONCLUSION

The first designs of SDR were unstable, there are currently a lot of designs that allow to manipulate radio signals only with a personal computer. There are multiple applications. Not only providing a very cheap radio receiver, SDR devices can be combined with free software that allow to examine the spectrum, interference detection, testing of repeater systems operation in digital communication and measuring electrical parameters of them. In this paper, we have discussed what is SDR and what are the protocols used for communicating between the GUI and PIC, PIC and RF transceiver which is very important in order to control the SDR. In SDR we command what to do through software, Like, what is the frequency to be generated and what modulation technique it should undergo which are very key factors. To reach this information to SDR we must implement the communication protocols for this. By reading this paper you will know about how to implement these communication protocols and to configure PLL in RF device in detail.

REFERENCES

1. Salcic,C.F.Mecklenbrauker,"Software Radio -Architectural Requirements,Research and Development Challenges", The 8th International Conferenceon Communication Systems, Volume 2, November 2002
2. Peter B. Kenington, RF Baseband Techniques for Software Defined Radio. London: Artech House,2005.
3. P.Johnson, "New Research Lab Leads to UniqueRadio Receiver," by space systems technologygroup, garland division May 1985
4. www.microchiptechnology.com
5. "What is Software Defined Radio"The Wireless Innovation Forum and IEEE p1900.1 Group[online]. Available: http://www. wirelessinnovation.org / introduction_to_sdr, accessed in Jan 14, 2012.
6. "SoftwareDefined Radio"Wikipedia[online]. Available: http://en.wikipedia.org/wiki/Software-defined_radio, accessed in Dec 21,2011.
7. RJ Lackey and DW Upmal, "Speakeasy: The Military Software Radio," IEEE Communications Magazine, May 1995
8. "Computer Networks: Principles, Technologies and Protocols for Network Design" by Natalia Olifer and Victor Olifer

AUTHORS PROFILE

V. Kumara Swamy, completed his B.E(ECE) in1998 from Osmania University, Hyderabad and his M.Tech (DSCE) in 2005 from JNTUH, Hyderabad and pursuing PhD in the area of VLSI Design from JNTUH, Hyderabad. His areas of research include VLSI Design, advanced Digital Design, Cryptography and Computer Networks and Applications. He has published 16 research papers in international journals and conference proceedings. He has expertise in industry standard EDA tools such as Cadence, Mentor Graphics and Xilinx Tools etc. He has 21 years of teaching and research experience in India and abroad. He is currently working as an Associate Head, Department of ECE in Sreenidhi Institute of Science and Technology, Hyderabad, Telangana, India

Dr. Prabhu G, Benakop is a renowned educationalist and he is currently working as Principal in Methodist College of Engineering, Hyderabad, Telangana State, India. Formerly he held the positions of PRINCIPAL for Indur Institute of Engineering and Technology, Aurora's Engineering College, Director in Aurora's Technological and Research Institute, Hyderabad. He has 31 years of teaching and research experience. Senior Member, IEEE, Member, International Biomedical Engineering, Life Member of Indian Society for Technical Education, Life Member of CSI, Life Member of Instrument Society of India. Guided/Guiding 11 research scholars under JNTUH, Hyderabad and VTU Bangalore for PhD programme. His areas of research include VLSI System Design, Embedded Systems, Computer Networks, and Biomedical Signal Processing. He has published more than 80 research papers in national, international journals and conference proceedings.