# The Application of Embedded System in NMR Spectrometer Lock System

Zurong Ni
Department of Physics
Xiamen University
Xiamen, Fujian, China

Haibo Chen
Department of Physics
Xiamen University
Xiamen, Fujian, China

Zhiwei Chen
Department of Physics
Xiamen University
Xiamen, Fujian, China

Zhong Chen*
Department of Physics
Xiamen University
Xiamen, Fujian, China
(0086592-2181712; e-mail: chenz@xmu.edu.cn)

***Abstract*—Homogeneous and stable magnetic field is essential for a high-resolution NMR spectrum, especially one which requires signal averaging or phase cycling. The field-frequency lock technique can effectively improve the long-term stability of the magnetic field. The whole design scheme of the lock system for NMR spectrometer is presented in this paper. The digital control is implemented with Xilinx Virtex-4 FPGA, XC4VFX12-FF668, the signal received from the probe is detected by the quadrature detector, and the resonance frequency drift is locked by the quasi-feedback loops. In the slave computer of the lock system, Wind River's VxWorks real-time operating system is chosen to fulfill the strict requirement for the performance of real-time data acquisition and control, as well as the stability of operation system. Online debugging with the Varian 500MHz spectrometer shows that the lock system achieves the goal of compensating for the intrinsic drift of a superconducting magnet field.**

***Keywords—NMR; Spectrometer; Lock System; VxWorks***

## I. INTRODUCTION

Nuclear magnetic resonance (NMR) spectrometer is an important instrument for the study of micro- and macro-structure of the material, as well as interaction. It has made quantum leaps in decades, becoming a staple tool in such divergent fields as chemistry, physics, materials science, biology, and medicine. That is why it is essential that scientists or profession working in these areas be fully engaged in the research and manufacturing of NMR instrument.

In order to produce a high resolution NMR spectrum of a sample, especially one which requires signal averaging or phase cycling, there are not only high magnetic field strength in the NMR spectrometer, but also a temporally constant and spatially homogeneous magnetic field. But the field strength might vary over time due to aging of the magnet, electromagnetic interference, and temperature fluctuations. It is important to equip the NMR spectrometer with a lock system compensating for the intrinsic drift of a superconducting magnet field.

The field lock system is a separate NMR spectrometer within and simpler than the whole spectrometer [1,2,3]. This spectrometer is typically tuned to the deuterium NMR resonance frequency. It constantly monitors the resonance frequency of the deuterium signal and makes minor changes in the $B_0$ magnetic field to keep the resonance frequency constant when a change in frequency is detected, and the resultant $B_0$ field as a function of time while the magnetic field is drifting. Figure 1 shows a block diagram for the lock system. The lock transceiver generates the lock transmitter frequency using signals from the reference generator and the lock offset frequency from the lock transceiver controller and, at the selected pulse rate, applies the signal to the lock coil in the probe. The signal from the sample is detected by the lock coil in the probe and applied to the preamplifier, where it is amplified and routed to the receiver section of the transceiver board. In the receiver section, the lock signal is amplified, mixed with the local oscillator, phase detected into two signals, in-phase and orthogonal, and applied to the lock loop circuit to detect an error signal, initiate the current flowing into the compensation coil with appropriate magnitude and direction, compensated the magnetic field drift and keep the field constant.

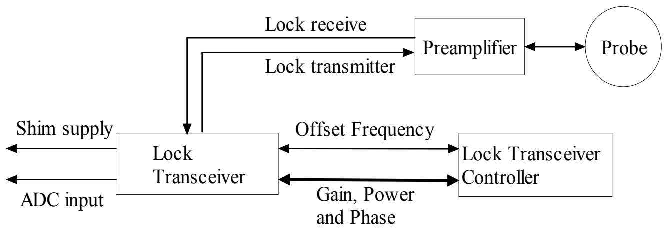


Figure 1. Lock System Block Diagram.

## II. VXWORKS

In a high-speed data acquisition system, the acquisition card is more and more mounted in a industrial control computer, in respect that the personal computer hardware is not equipped dedicated bus and the software does not meet the requirements of real-time. The operation system and application software of the industrial control computer belong to the embedded software, which controlling the embedded CPU and peripheral devices is aimed at the data acquisition wholly, and has a strong real-time, or good multitasking support. The lock system should be an embedded system, because there are strict requirement for the performance of real-time data acquisition and control, as well as the stability of operation system. VxWorks is a proprietary, real-time operating system developed by Wind River Systems of Alameda, California, USA in 1987[4]. VxWorks has been ported to a number of platforms and now runs on practically any modern CPU that is

This work was supported by the National Key Technology R&D Program in the 11th Five year Plan of China (2006BAK03A22).

used in the embedded market. This includes the x86 family, MIPS, PowerPC, Freescale ColdFire, Intel i960, SH-4 and the closely related family of ARM, StrongARM and xScale CPUs. The key features of the current OS are:

- Multitasking kernel with preemptive and round-robin scheduling
- Fast interrupt response
- Memory protection to isolate user applications from the kernel
- Fast, flexible inter-process communication including TIPC
- Binary, counting, and mutual exclusion semaphores with priority inheritance
- Full ANSI C compliance and enhanced C++ features for exception handling and template support

Modern real-time systems are based on the complementary concepts of multitasking and intertask communications. A multitasking environment allows a real-time application to be constructed as a set of independent tasks, each with its own thread of execution and set of system resources. The intertask communication facilities allow these tasks to synchronize and communicate in order to coordinate their activity. In VxWorks, the intertask communication facilities range from fast semaphores to message queues and from pipes to network-transparent sockets. Another key facility in real-time systems is hardware interrupt handling, because interrupts are the usual mechanism to inform a system of external events. To get the fastest possible response to interrupts, interrupt service routines (ISRs) in VxWorks run in a special context of their own, outside any task's context. Finally, VxWorks is fully compatible with the BSD socket interface for sockets in the Internet domain. Applications typically use sockets in the Internet domain to exchange information with peers on remote or local host systems. All these above will be described in the chapter Ⅳ.

## III. HARDWARE DESIGN OF LOCK SYSTEM

Based on the procedure of lock system in NMR spectrometer described above, we designed a set of circuit to monitor and keep the resonance frequency constant[5], including Xilinx Virtex-4 FPGA, quadrature detector, phase synchronization circuit and peripheral circuit, as shown in Figure 2.

### A. *Xilinx Virtex-4 FPGA, XC4VFX12-FF668-10C*

The Virtex-4 Family is the newest generation FPGA with high-performance, full-featured solution for embedded platform applications from Xilinx[6]. A wide array of hard-IP core blocks completes the system solution. These cores include the PowerPC processor (PC405), Tri-Mode Ethernet MACs, 622Mb/s to 11.1Gb/s serial transceivers, voltage/temperature system monitor blocks, dedicated DSP slices, high-speed clock management circuitry, and source-synchronous interface blocks. Combining a wide variety of flexible features, the Virtex-4 family enhances programmable logic design capabilities and is a powerful alternative to ASIC technology. The IBM PowerPC processor PC405 is a full 32-bit RISC CPU that is embedded in the Xilinx FPGA Virtex-II Pro and Virtex-4 FPGA families. It can be run at speeds up to 450MHz and is the best choice for CPU-intensive tasks in Xilinx FPGA based systems. The data acquisition and other control signal initiating modules in lock system are controlled by FPGA as the CPU core to complete the functions such as signal sampling by ADC, data caching by FIFO, and communicating by Ethernet.

### B. *Quadrature Detector*

The classical analog approach for the NMR spectrometer is based on analogue quadrature detector[7]. The receiver multiplies the input signal with quadrature signals coming from a local oscillator whose operating frequency $f_{L0}$ close to the Larmor frequency $f_L$ of the material (deuterium) to be analyzed. Here the local oscillator signal is generated by a direct digital synthesizer (DDS), improving the precision and flexibility of the system. A low pass filter is added after the down-mixer on each branch to eliminate high frequency components. Since the signal after mixers contains frequency components at ($f_{L0}$-$f_L$) and ($f_{L0}$+$f_L$), digital filters suppress the latter. The lower frequency signal is nearly DC if the DDS's frequency matches the Larmor frequency, and I(t) (In phase component) and Q(t) (Quadrature phase component) can be used to compute module and phase of the NMR signal, shown as follows.

Signal $v_r$ getting from the receiver:

$$v_r = A\cos[2\pi(f_L+\delta)t+\varphi]\exp(-t/T_2) \qquad (1)$$

Here $f_L$ is the Lamor frequency, $\delta$ is the shift of frequency,

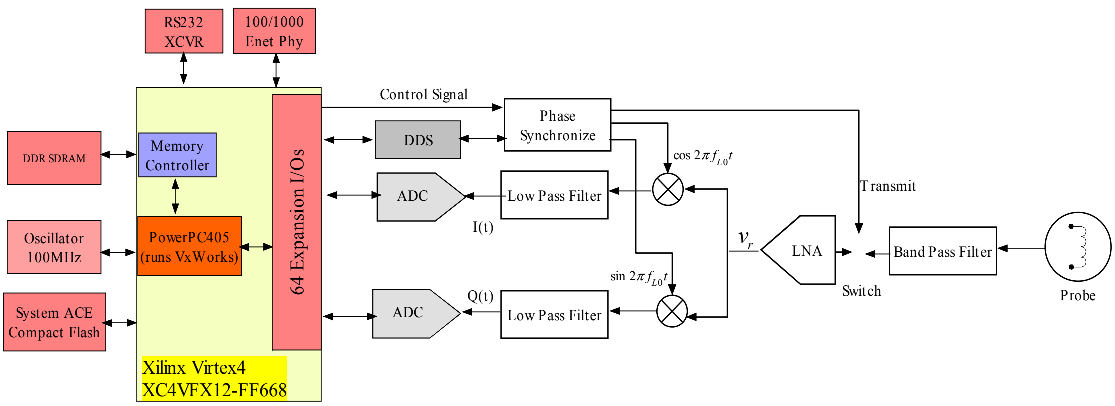


Figure 2. Lock System Hardware Block Diagram

and $T_2$ is the transverse relaxation time.

The signal of the in-phase branch after mixer is:

$$
\begin{aligned}
v_i &= v_r \cdot \cos 2\pi f_{L0} t \\
&= \frac{A}{2}\{\cos(\delta t+\varphi)+\cos[(2\omega+\delta)t+\varphi]\}\exp(-t/T_2)
\end{aligned} \quad (2)
$$

$$
\begin{aligned}
v_o &= v_r \cdot \sin 2\pi f_{L0} t \\
&= \frac{A}{2}\{\sin(\delta t+\varphi)-\sin[(2\omega+\delta)t+\varphi]\}\exp(-t/T_2)
\end{aligned} \quad (3)
$$

Synthesize the two signals after low pass filter, get

$$
v = v_i + i v_0 = \frac{A}{2}\exp(i\delta t+\varphi)\exp(-t/T_2) \quad (4)
$$

from which, one can get the module and phase of the NMR signal.

### C. Phase Synchronization

The deuterium resonance frequency, 76.75MHz in a 500MHz spectrometer, is also the center frequency of transmitting and receiving signals in lock system. But in different substances (deuterated solvent) the spectrum of deuterium has different chemical shift, there is a small shift to such frequency of the stimulating signal in lock system. By control signal from FPGA, the phase synchronization module can adjust the output local oscillator signal frequency to match the Lamor frequency of deuterium, as well as the frequency of transmitting pulse, to compensate the shift of the frequency.

## IV. SOFTWARE DESIGN OF LOCK SYSTEM

### A. Synchronization of the Tasks

It is often essential to organize applications into independent, though cooperating, programs. Each of these programs, while executing, is called a task. In VxWorks, tasks have immediate, shared access to most system resources, while also maintaining enough separate context to maintain individual threads of control[8,9]. The function of slave computer in lock system, especially communicating with the host computer, should be divided into multi-task to be more efficient and real-time. The main tasks of the slave computer, as shown in Figure 3, can be organized into several groups as follows.

*1) Initialization task(tServerInit, tClinetInit, tInterruptInt)*

The first two task fulfill the initialization of slave computer as server (receiving command from the host) or client (sending acquisition data to the host), including establishing the listening socket and creating other necessary tasks. The third task is concerned with interrupting event, enabling the interruption and connecting a user routine to a hardware interrupt.

*2) Link check task(tSvrLinkCheck, tCltLinkCheck)*

These tasks check the link between the slave and host computer. When the link is failed or some errors happen, these tasks will activated the socket close task(tSvrSocketClose, tCltSocketClose) and reinitialize the link, even the operation system software of the slave computer.

*3) Interactive task between slave and host computer(tSvrCMDExplain, tSvrCMDReceive, tCltSend, tCltConfirm)*

The task tSvrCMDReceive receives the packed data sent from the host computer and translates the data by shared memory to task tSvrCMDExplain, which unpacks and explain each part of these data, calls the correspondent device driver function. Function of the task tCltSend is fulfilled by sending acquisition data to host computer, from which confirming packed data sent is processed by the task tCltConfirm.

*4) Interrupting task(tInterruptHandler)*

The task is activated by the ISR(Interrupt Service Routine) and gets data from the interrupting buffer, then packs and translates it to the shared memory.

These tasks' execution is coordinated with external events or each other in-phase by semaphores, which are the primary means for addressing the requirements of task synchronization. Task waits for the semaphore by calling semTake( ), and blocks until the other tasks give the semaphore by semGive().

### B. DataSocket Communication in Lock System

VxWorks is fully compatible with the BSD socket interface for sockets in the Internet domain, including stream sockets and datagram sockets. For the purpose of reliable communication, we choose stream sockets on the VxWorks target. The mode of network communication between slave and host is Client/Server (C/S) in general. One or more tasks in server wait at a designated port for the client's connection request. Once a successful connection is setup, data can be exchanged between

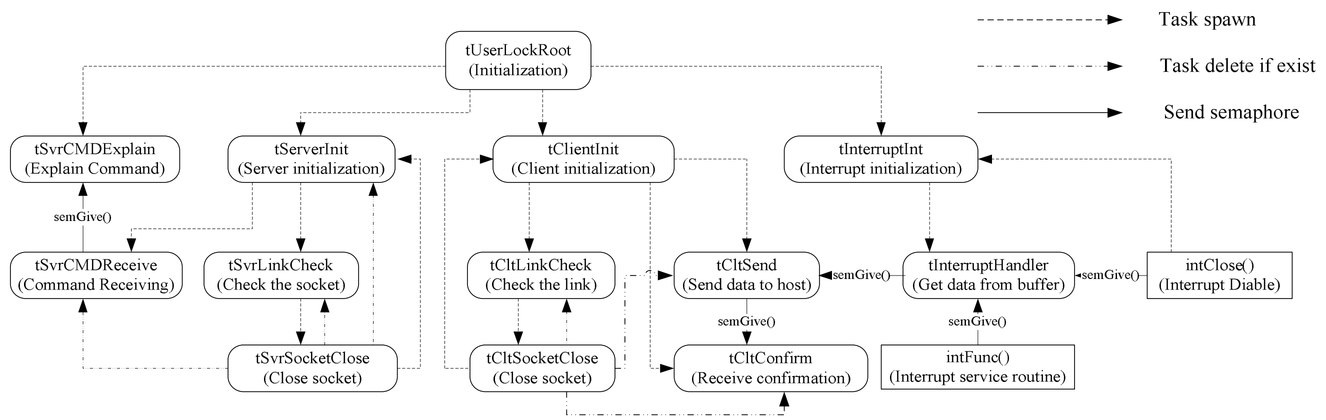


Figure 3. Synchronization of Tasks

each other in accordance with agreed method and data formats. The client sends the connection request when necessary. For the need of exchanging acquisition data and command between the host and slave computer, both client and server are constructed in the VxWorks target, as shown in Fig. 4. On both server and client sides, sending and receiving data over a socket, a socket descriptor is created by calling socket(), then bind a name to the socket descriptor by calling bind(). On the server side, the server needs to have called listen() on the socket it has created to communicate with clients. On the client side, the client calls connect() for communicating with its server, or SockConnectWithTimeout() for waiting a timeout during which it cannot initiate a connection and then report a error. The server can then call accept() on the socket to indicate that it

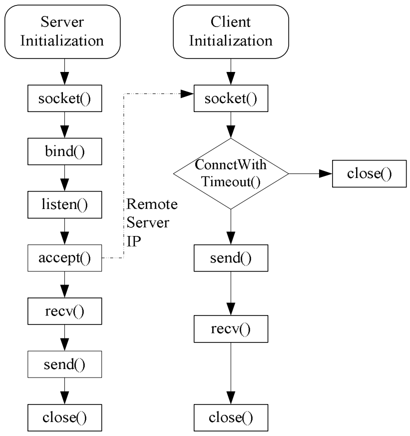

Figure 4. DataSocket communication in Lock System

accepts the socket connection. When call returns from the accept(), the function value is a new socket descriptor. For such a socket descriptor, one can exchange messages over the socket using send() and recv(). When a client or server has decided that it is time to end its conversation, it can break the connection by calling close().

### C. *Interrupt Service Code*

Hardware interrupt handling is of key significance in real-time systems, because it is usually through interrupts that the system is informed of external events. In lock system, when finishing sampling, the lock receiving module will call a hardware interrupt. VxWorks provides the routine intConnect(), which allows C functions to be connected to any interrupt.

```
intConnect(INUM_TO_IVEC(LOCK_INT_IRQ_ID), LockintFunc, NULL);
```

Here, the function LockintFunc, connected to an interrupt, is called an interrupt service routine (ISR).

```
void LockintFunc(void)
{   intDisable((int)LOCK_INT_IRQ_ID);
    ClearLockInterrupt();
    semGive(SemLockInt); }
```

Here, LockintFunc() gives a binary semaphore to task tLockintHandler described before, part of which is listed as follows.

```
void LockintHandler(void)
{   ……
  while(semTake(SemLockInt,WAIT_FOREVER)!=ERROR)
{       …;
        FIFO_full_Interrupt_Handler(LockReceiveBuffer_n);
        netSendAdd();
        LockEnableInterrupt();
        intEnable((int)LOCK_INT_IRQ_ID);}
}       …;}
```

### D. *Lock Display*

NMR software program is developed by Java, based on which the application is typically compiled to bytecode that can run on any Java virtual machine (JVM) regardless of computer architecture. Fig. 5 shows the Lock display window, where the green line represents the in phase component of the signal and the purple line represents the quadrature component. The nearly straight lines denote a good match between the frequency of pulse and Larmor frequency of deuterium.

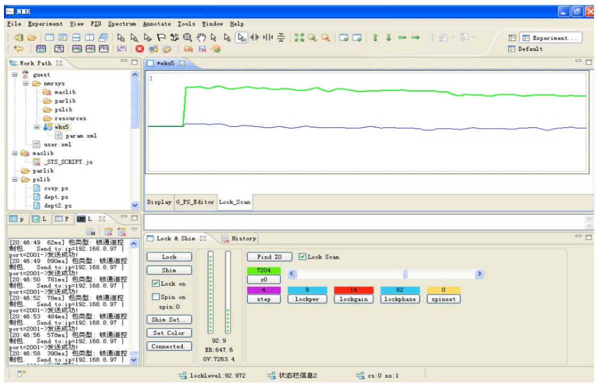
Figure 5. Lock display window

## REFERENCES


[1] C. Vignali, A. Caligiani, G. Palla. “Quantitative 2H NMR spectroscopy with 1H lock extender”, Journal of Magnetic Resonance, vol. 187(1), pp. 120-125, 2007.

[2] K. Kazimierczuk, W. Kozminski, “Efficient compensation of low-frequency magnetic field disturbances in NMR with fluxgate sensors”, Journal of Magnetic Resonance, Vol. 174(2), pp. 287-291, 2005.

[3] T. Maly, J. Bryant, D. Ruben, R. G. Griffin, “A field-sweep/field-lock system for superconducting magnets: Application to high-field EPR”, Journal of Magnetic Resonance, Vol. 183(2), pp. 303-307, 2006.

[4] Wind River, VxWorks Programmer’s Guide, http://www.windriver.com/.

[5] Chen Haibo, Li Jie, Zheng Zhenyao, “Study on Digital Field-frequency Lock for High-resolution NMR Spectrometer”, Chinese Journal of Scientific Instrument, Vol. 29(4), pp. 105-108, 2008.

[6] Xilinx, Virtex-4 Family Advance Product Specification, http://www.xilinx.com/.

[7] S. Christian, NMR spectroscopy: data acquisition, Wiley-VCH, pp. 32-34, 2001.

[8] J. M. Nogiec, J. DiMarco, S. Kotelnikov, “A Configurable Component Based Software System for Magnetic Field Measurements”, IEEE Transactions on Applied Superconductivity, Vol. 16(2), pp.1382-1385, June, 2006.

[9] Jie Zhang, Shilun Gao, Fangyi Jiang, “A Diesel Engine Simulation System Based on VxWorks RTOS”, 2006 IEEE International Conference on Vehicular Electronics and Safety, pp.12-16, December, 2006.