Direct memory access of diffraction patterns from striated muscle — A software view

Richard L. Lieber and Ronald J. Baskin

Department of Zoology, University of California, Davis, CA 95616, USA

Complete software support has been developed for a direct memory access microprocessor system used to store and analyze diffraction data from striated muscle. These studies are based on the premise that the regularly alternating light and dark regions of the fiber behave like a phase grating to incident laser light. A knowledge of the position of a diffracted order can thus be used to determine striation spacing. Since these striations are directly associated with the force generating components of the muscle, diffraction data can provide insights into the mechanism of force generation in muscle.

In our system, a charge-coupled device (CCD) is used to detect diffracted order, position, and intensity. In addition, a high speed muscle puller and tension transducer are used to characterize or alter the mechanical state of the muscle.

The software has been designed to allow the inexperienced user to perform sophisticated diffraction experiments. The user may present several experimental parameters: magnitude and direction of puller movement; number of frames of data to be taken; and delay of puller or scan. This has been accomplished by interleaving DMA and control loop cycles.

System performance indicates that the full 256 point analog output of the CCD can be digitized and stored in about 2 ms. The data can be transferred directly from the CCD to memory leaving the CPU free for experimental control or closed-loop processing.

1. INTRODUCTION

The mechanism of force generation in striated muscle is not yet fully understood [1]. Experimental approaches to determining this mechanism involve the simultaneous monitoring of muscle structure and tension. One method of monitoring muscle structure (which is completely non-invasive) is the method of laser diffraction [2,3]. This method is based on the premise that the periodic refractive index changes along the muscle will act as a phase grating to incident laser light. Since the periodic phase difference is due to interdigitation of force-generating filaments, force generation in muscle is accompanied by a change in phase periodicity. This periodicity change is directly manifest in the diffraction pattern produced by the muscle. Thus, real time monitoring of this diffraction pattern provides valuable insights into the mechanism of force generation in muscle.

Different techniques for monitoring the diffraction pattern have been used (film streak [4], photomultipliers [5], photodiode arrays [6], and CCTV detector systems [7]). Recently, however the photodiode detectors have been preferred because of their rapid detection speed, high resolution, and high sensitivity [8]. The limiting factor in the use of these detectors has been the ability to process the array of information in real time. Analog devices have been utilized [9], but these systems lack the overall flexibility and computing power available from a digital system.
We have developed a high speed microcomputer system which, using direct memory access (DMA), is able to digitize and store data at a rate of 7.9 µs/pt (127 kHz), while performing limited experimental control [10]. The programming language used for all routines was the Intel 8080 assembly language. The output codes for the DMA controller were hardwired on the DMA control board during fabrication. In this paper, we will discuss this system and the unique software required to run it at the user level.

2. METHODS

The DMA technique is a method for rapidly reading data to or writing data from memory. Data storage/retrieval can occur rapidly since, during DMA, the central processing unit (CPU) enters a ‘hold’ state and a peripheral device gains control of the data and address busses (i.e., becomes bus master). Time which is usually lost to CPU handshaking routines is therefore eliminated.

The complexity of writing the microcomputer DMA program for this application lies in the fact that during the hold state, the CPU is unable to exercise any program control. For our experiments, it was critical that the CPU be able to perform certain functions during data acquisition (e.g., delay timer, muscle length change output to DAC). Thus, it was necessary that the software developed only yield the bus to the DMA controller provisionally and allow the CPU to regain bus control to perform certain tasks during data acquisition.

In order to better understand the programs which were written for our system, the type of experiment to be performed must be first described. (The rationale for this type of experiment is given in section 3.) The sequence of events is as follows:
(1) Stimulate the muscle and allow it to develop tension.
(2) When tension reaches a plateau, apply a quick length change.
(3) Measure the resulting time course of tension.

The user must preset values for the magnitude and direction of length change, as well as specific DMA parameters. Specifically, the DMA controller must be initialized with two pieces of information:
(i) The starting address where the first data point is to be stored in memory;
(ii) The total number of data points to be stored (terminal count).

After this initialization, the controller is enabled and DMA cycles are run. As mentioned, our application required that the CPU be able to perform certain operations during data acquisition. Also, due to limited memory (4 k of static RAM which fills completely in ~50 ms experimental time), it was necessary that we be able to acquire diffraction data at any point in time relative to muscle length change. These factors necessitated the use of delay counters which the CPU used to obtain the relative timing desired.

The flowchart for the data acquisition program is shown in fig. 1. Note that although this is a software view, it is intimately tied to hardware operation. The steps marked with an asterisk (*) in fig. 1 can only occur when the data channel is disabled. Thus, the time flow of program steps is controlled partially by the data channel. The flowchart shows that two basic types of experiments can be run:
(1) Enable scan and delay muscle length change (MRDLY).
(2) Apply length change and delay scan (SNDLY).

In the first case, the DMA channel is enabled by the user and a delay loop is started. The magnitude of the delay is set by the user. In this first case, every time the data channel is not running a DMA cycle, the delay counter can be decremented repeatedly until the DMA controller issues another data channel request. Once the counter has reached zero, the length change is applied, and data acquisition is continued until terminal count is reached.

In the second case, the length change is applied and then a delay counter is started. Again, the magnitude of the delay is set by the user. Once the counter reaches zero, the DMA channel is enabled and it acquires data until terminal count is reached.

A subtle difference between the two counters should be noticed. In the first case, the counter can only run when the data channel is not run-
Fig. 1: Flow chart of data acquisition program. Blocks marked with an asterisk (*) can only occur when the data channel is disabled.
Table 1
Comparison of delay values in either MRDLY (delay length change after DMA enabled) or SNDLY (delay scan after apply length change) mode

<table>
<thead>
<tr>
<th>Delay value set (Hex)</th>
<th>Approximate delay (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MRDLY</td>
</tr>
<tr>
<td>$994$</td>
<td>1.5</td>
</tr>
<tr>
<td>$998$</td>
<td>3.0</td>
</tr>
<tr>
<td>$919$</td>
<td>6.0</td>
</tr>
<tr>
<td>$929$</td>
<td>12.0</td>
</tr>
<tr>
<td>$939$</td>
<td>18.0</td>
</tr>
<tr>
<td>$949$</td>
<td>24.0</td>
</tr>
<tr>
<td>$959$</td>
<td>30.0</td>
</tr>
<tr>
<td>$969$</td>
<td>36.0</td>
</tr>
<tr>
<td>$979$</td>
<td>42.0</td>
</tr>
<tr>
<td>$989$</td>
<td>50.0</td>
</tr>
</tbody>
</table>

Note that identical delay values impose a longer delay for the MRDLY mode. See text for explanation.

ning. In the second case, the counter is running freely. Thus, if an identical delay value is loaded into memory, the counter in the first case will take a longer time to reach zero. This fact was validated experimentally as shown in table 1. The delay values are in hexadecimal.

3. DISCUSSION

Historically, skeletal muscle has been thought of as a contractile component in series with an elastic component. The contractile component (CC) is modeled as a damped spring and the series elastic component (SEC) is modeled as an undamped spring. The tension developed during force generation in muscle can thus be envisioned as coming from these two different sources.

It is of interest experimentally to isolate the characteristics of each of these components since it is possible that one (the CC) represents the force generating components while the other (the SEC) may only represent stray compliance in the tissue, muscle filaments, or experimental apparatus. Since, as modeled, the two components will have different time courses of response to length change, the quick release experiment detailed in section 2 is used to isolate their characteristics. The early tension response after the length change will reflect the SEC and the subsequent tension response will reflect both the SEC and CC [11].

Rapid acquisition of diffraction data from photodiode arrays under digital control has previously not been possible due to the large number of data points (e.g., 256) requiring digitization (A–D conversion) and storage in a few milliseconds. This problem has been circumvented by the use of direct memory access (DMA) data acquisition. Unfortunately the use of DMA adds an additional constraint to the experiment, namely that CPU activity is limited during acquisition. In our application, it was important that some degree of CPU activity be possible (e.g., for length changes and delay timers—see section 2).

Our approach to retaining some degree of CPU activity during DMA data acquisition has been to only provisionally release the data and address busses to the DMA controller and to interleave the software routines with data channel cycles. As shown in fig. 1, this allows for the application of a length change and the acquisition of real-time data at any relative point in time.

In this configuration, we are able to perform experimental control while acquiring data at the rate of 7.9 μs/pt (127 kHz). In physiological time, this means that we can obtain a complete diffraction profile of the contracting fiber in about 2 ms (~256 pt × 7.9 μs/pt). This was the major objective in building the system.

At present, our data acquisition rate is limited by ADC time, processor speed, data channel latency, and memory word size. Future plans are underway to apply this concept to a faster microprocessor and a video speed analog-to-digital converter. In this configuration, there will be a much greater potential for closed-loop control of muscle length since the data channel speed and processor speed are about ten times faster. This means that longer software routines can be executed in between data channel cycles and thus, during data acquisition, a more sophisticated feedback loop can be implemented.
ACKNOWLEDGEMENTS

We would like to thank Dr. Ken Roos and James W. Cline for their assistance in providing valuable suggestions in developing the programs described here. R.L.L. is an NIH predoctoral fellow. This work was supported, in part, by grant PCM-7903256 from the National Science Foundation, Washington DC.

REFERENCES