The Stay Alert Pen: SMART PEN
ECE-492/3 Senior Design Project
Spring 2010

Electrical and Computer Engineering Department
Volgenau School of Engineering
George Mason University
Fairfax, VA

Team members: Michael DePrenger, Nick Fleming, Fiona Lu, and Yu Shao
Faculty Supervisor: Dr. Siddhartha Sikdar

Abstract:
We implemented a portable smart pen system capable of detecting lapses in concentration while the user is reading. An accelerometer and a microcontroller is embedded within a pen casing to record data as a user reads sections of text using the pen as a pointer. When a substantial pause in reading is detected, the system generates an appropriate warning or alarm. An accompanying software program can communicate with the pen through a USB interface to customize system parameters, record relevant data, and graph collected data over time. The overall effectiveness of the system was tested on 5 regular volunteers and 3 persons with attention deficit disorder (ADD). The system exhibited a 10% probability of false alarm for typical reading habits, but rose up to 31% for atypical reading habits. With further refinement, this system will enable patients with attention deficit/hyperactivity disorder to maintain concentration during reading in a variety of environments.

1. Problem statement
According to the National Center for Health Statistics, 4.5 million school-aged children in the United States have been diagnosed with Attention deficit/Hyperactivity disorder (ADD/ADHD). ADD or ADHD is the condition when a person has trouble keeping his or her attention on tasks or activities. The SMART pen described herewith is proposed as a useful tool to assist ADD/ADHD patients to concentrate on activities effectively, especially when reading boring or long texts. The goal of the project was to develop a system that can keep ADD/ADHD patients alert and focused while they are reading. One feature of this pen, is that it should be able to detect loss of concentration of a patient and issue a discreet alarm to get him or her back on track. Another characteristic is that it must be easy to operate and able to collect usage data for users to download, view, and analyze (average time between alarms, for example) their performance in a graphical user interface. The pen should furthermore be portable, comfortable and power efficient. The final product should also be affordable to the average person. The design should incorporate sensors that would eliminate the need for any oversight. If the device is easily perceived to be a learning aid, the user may feel self-conscious when using it in public. Therefore, the goal of the design was also such that it would be disguised as an everyday object.
2. Alternative designs

Our team came up with three designs, analyzed them, and eventually made a final decision on one design. The first design (Figure 1) involved implementing all operations within one device. The alarms, sensors, microcontroller, and data storage would all be in the pen. A button and switch on the pen would allow the user to customize certain settings and collect/save data. An LED and vibrator would serve as the alarm system. It would also be capable of transmitting the usage data to a computer for analysis.

The second design was to have an additional portable device besides the pen. This device, about the size of a cell phone, could be stored in the user’s pocket during pen use. It would take care of data storage, alarm implementation, and graphical display of usage data. It would be evident that most of the components that were in the pen in Design 1 would be moved to this portable device. Because of the additional room that this external device provides, other functions such as a daily alarm could also be implemented.

The third design was similar to the first, with the exception of the communication between the MCU and PC. In this third design, a USB COMM would replace the wireless transmitter and receiver. Due to its relatively cheap implementation, wired communication maintains a distinct advantage over the wireless option. The user would not only be able to review and analyze performance, but also customize the pen settings such as the threshold and period of time to alert the user. The alarms, sensors, microcontroller, and data storage would all be in the pen. An LED and vibrator would serve as the alarm system.

The tradeoffs made among the three alternatives generally fell into one of the following categories: functionality, convenience, and engineering difficulty. Because all of the required electronic components must be inside the pen in Design 1, the size and the weight of the pen would be larger compared to the pen in Design 2 which comprised of minimal electronic parts inside the pen. With regard to functionality, Design 2 would have more function than Design 1. A speaker option would be included to alert the user or to be set as an audio component for an alarm. The LCD screen could show the user’s performance on demand. In Design 1, a computer would be required to analyze the performance. In terms of economics, Design 1 was found to be more marketable as, it only requires the user to carry one device instead of two. Secondly, since Design 1 would serve the same core function as Design 2 with fewer components, it would have greater convenience and a smaller price tag. Because it was believed that many users would desire subtlety and convenience over extra function (such as a daily alarm or portable LCD), Design 1 was the optimal choice for the project over Design 2. Given that Design 3 is very similar to Design 1, the same comparisons made with Designs 1 and 2 can be applied to 3 and 2. Hence, the advantages and disadvantages of both designs still hold. Design 3, however, traded in a small amount of convenience for a lot more ease in engineering. By replacing the wireless interface with USB, it would be possible to better manage the space and power budget. While the user no longer has the convenience of a wireless device, this cost is deemed to be relatively small. Design 3, discussed below, was chosen as an alternative.

3. System Architecture

Overall system architecture with the various system hardware components is shown in Figure 1. From the user’s perspective, a switch and push button is incorporated to control the pen. The switch simply turns the pen on and off. When the system is on, the pen monitors the user’s performance (alert the user when he/she is not concentrating). Pushing the button at this point incurs a hardware reset. This is used to get out of alarm and warning states. When the pen is hooked up to the computer via the USB interface, the system is put in “transmit” mode by holding the button down for two seconds. The user can then interface with the program on the PC to receive data from the pen and set customizable parameters in the pen’s program. If the switch is turned off, the battery is physically separated from the circuit so as not to correspond to a soft switch.

To use the pen, the user must simply turn it on, triggering the start of signal processing. The microcontroller then begins to process the incoming accelerometer signal. Figure 2 shows how the system moves between reading, warning, alarm, and sleep states. If the system determines that the user has lost concentration, a warning is
issued, and an LED is turned on. If the user notices the LED, he/she can press the button to initiate a hardware reset and return to the reading state. If the user does not notice the LED, a timeout will eventually occur. When this happens, the system moves to the alarm state and the pen begins to vibrate. The user is again presented with the option of pressing the button or alternatively to wait for a timeout. The button sends the system back to the reading state. If a timeout condition is met, it is assumed that the user will not respond in the near future and the entire system is put to sleep in order to conserve power. The system can be re-awaked and returned to the reading state by pushing the button again. At any point during its use, the button may be double tapped to enter a snooze mode. In snooze mode, the system stalls its operation for a limited amount of time. Once this time has expired, the system resumes normal operation.

![System architecture diagram](Image)

Figure 1: System architecture

![Warning/Alarm algorithm diagram](Image)

Figure 2: Warning/Alarm algorithm

As mentioned earlier, the basic goal of the pen is to keep individuals with ADD alert when they are reading, however the pen would be a more useful tool if it were also able to track the usage over period of time. It is considered important to have such data about the user’s actions available to view and analyze. The SMART pen can even be used as a medical device for use by doctors treating ADD conditions to monitor their patients. Therefore, it is a desirable feature for the pen to be able to store usage data of the pen and later transmit the data to a PC. The best way to transfer the data was determined to be via USB cable. This decision was due to multiple factors including price, PCB real estate, reliability of data, and power consumption. The pen includes a USB adapter that allows a user to plug the pen into a computer and adjust timing settings, in addition to being able to download usage data.
3. Hardware design

The center element of the system is the ATMEGA328 MCU. The microcontroller has an external 8 MHz crystal clock for the timer. This MCU is interfaced with a 2n2222 switch transistor, push button, vibrator, and LEDs. The output signals from the MMA7260 accelerometer is fed into an analog-to-digital converter in the microcontroller. The voltage step receives input from the battery source voltage and generates a steady voltage for the microcontroller. A hard switch is used to control the supply of battery power. The USB driver chip handles two-way communication between the PC and MCU. The integrated system is depicted in Figure 4.
4. Signal Processing Algorithm

Several methods of detecting reading motion were considered. The MCU program must be able to determine if the user is reading or not. To this end, the incoming signal must pass several tests in order for it to be considered reading motion. Research resulted in the classification of the incoming signal into four categories: reading motion, vigorous motion, negligible motion, and incorrect orientation. Note that the outcomes of these algorithms determine when a warning or alarm is generated.

When observing accelerometer reading signals, it is important to look for attributes that indicate reading motion, as well as those features that clearly indicate a lack of it. The algorithm used in this system classifies the incoming signal into four separate cases. The first, which is the reading motion, indicates the user is actively reading the text. The other three, which include vigorous motion, negligible motion, and incorrect orientation, suggest the user has lost concentration. The associated signal attributes and methods for detecting them are discussed in the forthcoming section.

Reading Motion was the most challenging case to detect due to its great dependence on individual habits. Reading methods range from sliding the pen across each line, pointing to individual lines, or even pointing to individual words. It is important to notice that, though different, each of these methods output a repeating pattern as demonstrated in Figure 5. When a user reads through a line, he/she typically repeats the same motion every time. In most cases, the wrist is rotated as the pen is moved. The changing orientation of the pen also causes the baseline to move. When the user moves to the next line, the quick change in baseline causes a noticeable aberration in the signal. Line changes are inevitable, regardless of the reading habit employed. A peak
detection algorithm was used to detect these line changes. As one might guess, if the user is reading, these clusters will appear in relatively predictable time intervals. Our reading motion algorithm uses these time intervals to determine if the user is reading. We apply a running average of 200-point buffer on the Constant False Alarm Rate (CFAR) method. If the mid-point of this buffer is greater than the averages on either side, a peak has been detected. With this method in place, the MCU measures the time interval between peak detections. In most cases, it takes about five seconds to go from one line to another. The pen keeps track of peak clusters occurring in this interval. If this predictable pattern repeats itself, the system considers the user to be reading.

Group movements that are excessive in nature are loosely grouped and do not accompany reading movement as "vigorous". Common examples of this would be twiddling the pen in one's fingers, rolling the pen on a table, or throwing the pen up and catching it. Vigorous motion can occur either as a series of discrete events or one lengthy continuous event. A Vigorous Motion Event (VME) may be accompanied by a multitude of different signal outputs. Such movements consist of rapid changes in signal baseline, amplitude, or both. Invariably, high and frequent peaks in the signal are evident. Both of these features lead to an increase in signal variance. This is the basis of the vigorous motion algorithm used by the SMART pen. In a typical vigorous motion event (VME), the variance of the signal will likely exceed a certain threshold. If a VME is detected, the MCU will time stamp it and wait for another event. If another occurs within a certain time interval, it is tallied and the time stamp is updated. A warning will be issued if the overall tally exceeds five counts. This count is reset if a VME is not detected within five seconds. Note that the special case of continuously high variance is also accommodated (e.g. not a series of discrete VMEs).

Negligible motion refers to the case when the pen does not undergo any substantial movement. Among all the possible cases, this is perhaps the easiest to detect. It is always accompanied by a low-variance signal and there are no peaks or baseline changes. The output signal simply corresponds to the background noise. As with vigorous motion, variance turns out to be a very useful calculation. However, one big difference between detecting Negligible Motion and Vigorous Motion is that Negligible Motion cannot occur in discrete events. The algorithm must look for a continuous time interval that satisfies some condition. The MCU simply looks for the variance to stay below a threshold for more than seven seconds. It is possible to trigger a false alarm by reading very slowly, but it was found that this case would be difficult to simulate and determined that only a minority of users would read in such a way.

One of the more obvious indications that the user is not reading, is the incorrect orientation of the pen. The detection of such situations reduces to a basic knowledge of the angle of the pen relative to the gravity vector. If the pen and gravity vector form a 90-degree angle, one could assume that the pen is laying on a flat surface and the user is not reading. Observations have led us to allow an acceptable angle of at most 80-degrees. Thus, the conical volume drawn out by the pen as it is rotated about the gravity vector at an angle of 80-degrees defines the acceptable volume in which the pen can reside. If the pen leaves this volume, a warning is generated, to be rescinded only when the pen returns to an acceptable orientation. The baseline of an accelerometer axis that is parallel with the pen's major axis can be used to calculate the pen's angle. If a rotation about any axis other than the pen's or the gravity vector occurs (that is, if the angle relative to the gravity vector changes), the baseline of the accelerometer signal changes accordingly.

5. Testing and Evaluation Criteria
The testing was broken down into Individual part testing, algorithm testing, full system testing, GUI JAVA & USB communication. Most importantly, the system algorithm underwent several tests and refinements over the course of the project. The initial program that was tested had several flaws and was susceptible to unreliable readings (false alarms). The tests took the form of MATLAB simulations conducted on accelerometer signals collected by the MCU. Much of the algorithm optimization took place on MATLAB.

Once the MATLAB simulations predicted a workable algorithm, the program was transferred to the MCU where it was executed in real time. In this stage, simulation was performed based on several scenarios that could
potentially “break” the system. The most likely obstacle faced under these scenarios was overcome through the coding process of the software. The group eventually procured the help of several volunteers to test the algorithm on live subjects. The following table shows the results of the human subject testing. Note that the tests did not single out specific algorithms, but the program as a whole:

<table>
<thead>
<tr>
<th>Subject</th>
<th>Total reading time (s)</th>
<th>False detection time (s)</th>
<th>False alarm events</th>
<th>Probability of false alarm (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>745</td>
<td>64</td>
<td>2 NR</td>
<td>8.6</td>
</tr>
<tr>
<td>2</td>
<td>720</td>
<td>0</td>
<td>-N/A</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>720</td>
<td>64</td>
<td>2 NR</td>
<td>8.9</td>
</tr>
<tr>
<td>4</td>
<td>654</td>
<td>134</td>
<td>4 NR, 1 NM</td>
<td>20.5</td>
</tr>
<tr>
<td>5</td>
<td>770</td>
<td>37</td>
<td>1 NR, 1 IO</td>
<td>4.8</td>
</tr>
<tr>
<td>6</td>
<td>666</td>
<td>120</td>
<td>3 NR, 4NM</td>
<td>18</td>
</tr>
<tr>
<td>7</td>
<td>661</td>
<td>38</td>
<td>1 NR, 1 NM</td>
<td>5.7</td>
</tr>
<tr>
<td>8</td>
<td>680</td>
<td>207</td>
<td>6 NR, 1NM, 3VM</td>
<td>30.4</td>
</tr>
</tbody>
</table>

NR (not reading), NM (negligible motion), IO (incorrect orientation)

For normal reading habits, the probability of false alarm was less than 18%. However, when users exhibited more unusual reading habits, the false alarm rate was as high as 30%. Most of the false alarms were triggered by the “not reading” warning state. That is, the system failed to detect repeating patterns of peak detections within a specified interval. If the pen moved relatively quickly, for example, the time intervals between peaks could be quite small, preventing the system from recognizing it as reading motion. A partial remedy to this problem would be to allow the Java program to customize the expected time between line changes. This would accommodate reading styles of various users. While not all percentages were under the 15% goal, the average false alarm probability was about 12% (not including the atypical reading habit cases). This is under the desired 15% and indicates that our system, under ideal conditions, is relatively reliable.

It was then necessary to calculate the power consumption of the entire system. The following table summarizes the results:

<table>
<thead>
<tr>
<th>Power Consumption of Entire System</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mode</td>
</tr>
<tr>
<td>Reading</td>
</tr>
<tr>
<td>Warning</td>
</tr>
<tr>
<td>Alarm</td>
</tr>
<tr>
<td>Sleep</td>
</tr>
</tbody>
</table>

The average current drawn for the normal reading section was found to be 2mA. Given that a standard AAA battery has a 1100Ahr rating, this system can be expected to run for a long time on one battery.

6. Conclusion
ADD is a serious and common disorder in today’s society. Several technologies have been developed in an attempt to assist those with the disorder. However, these alternative approaches do not offer the versatility that the SMART PEN is believed to be capable of. Though there is certainly room for improvement, the product developed in this project has fulfilled the requirements that were set out to be met. Its ability to potentially expand the attention span of ADD students will allow these students to read lengthy texts without supervision. In addition, saved data can be analyzed and improvement can be tracked. Lastly, because the proposed solution is discreet, users can carry it with them without being self-conscious. It is envisioned that believe many people will find a use for this design and will prefer it over those solutions that are less mobile and less discreet.