

Smart Wireless Vehicle Detection System

Joel L. Wilder, Aleksandar Milenković, Emil Jovanov

Abstract— Traffic density becomes an important issue in modern cities due to sustained growth in population, higher standards of living, and increased levels of mobility. As a result, environmental pollution is on the rise, natural resource reserves are being depleted, and commute times are lengthened. These conditions are increasing the need for effective management of traffic flow. Improved traffic flow management involves maintaining statistics on transportation infrastructures, and tracking density levels so that enhancements can be implemented. Existing methods for gathering these statistics are costly, labor intensive, and require remote, offline data acquisition. In order to address these issues, we present a Smart Vehicle Detection System (SVEDECS) that provides real-time statistics of traffic flow, capable of tracking vehicle speed and vehicle type. We describe an implementation of SVEDECS comprised of wireless magnetic sensor nodes and a gateway node.

I. INTRODUCTION

The number of people in the world is growing steadily, as shown in Figure 1, with a current population of approximately 300 million in the United States and a total of 6.6 billion in the world [1]. Large percentages of the U.S. population are living in urban areas. A recent U.S. Census Bureau report shows that New York, New York is the most populated U.S. city at 8.2 million, followed by Los Angeles, California at 3.8 million, Chicago, Illinois at 2.8 million, Houston, Texas at 2.1 million, and Phoenix, Arizona at 1.5 million [2].

A steady increase in urban population poses a significant burden on the transportation infrastructure in cities, resulting in poor traffic flow. As the number of vehicles on city streets and highways increases, driver frustrations grow as it takes longer to travel from one place to another. Since there is a limited amount of space for transportation infrastructure, one means for improving traffic efficiency is to monitor density as a whole and attempt to control traffic flow on congested streets. In order to accomplish such a task, traffic flow statistics for city streets must be gathered and made available in realtime. Conventional techniques for gathering this data

use intrusive sensors, such as pneumatic road tubes, inductive loop detectors, and micro-loop probes, which are highly accurate for vehicle detection, but disrupt traffic during installation and repair, leading to high installation and maintenance costs [3].

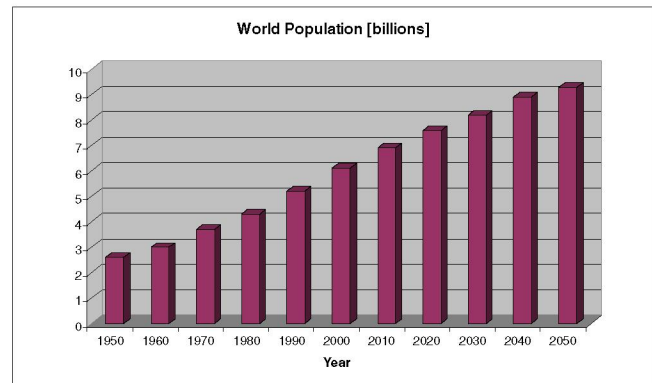


Fig. 1. World population trends.

The recent technological advances in sensors, low-power system-on-a-chip devices, and wireless communications have enabled the design and proliferation of wireless sensor networks that can be used for traffic surveillance and control. A wireless sensor node is comprised of an embedded microprocessor with memory, onboard flash storage, a short-range radio, and an energy source. Coupling a magnetic sensor with a wireless sensor node enables vehicle detection as magnetic field disturbances are tracked. This method provides an ideal means of acquiring automated traffic statistics on city streets and highways in an unobtrusive way. Moreover, our preliminary results indicate that magnetic sensors can facilitate vehicle classification (e.g., truck vs. cars). Thus, magnetic sensors can be placed on roadways, and as data is gathered, it can be wirelessly transmitted to a gateway node for retrieval and subsequent processing. Such techniques lead to ease of deployment and low maintenance costs.

The remainder of this paper is organized as follows. Section 2 presents a proposed SVEDECS system architecture and how it can be used when integrated with a transportation infrastructure. Section 3 describes a hardware prototype design that is based on commercial off-the-shelf components, as a first-step towards realizing the proposed SVEDECS system design. Section 4 presents the software algorithm that is used to drive the operation of the prototype design, while Section 5 provides test results gleaned from a typical deployment scenario. Section 6 concludes the paper.

Manuscript received November 11, 2007.

J. L. Wilder is with the Electrical and Computer Engineering Department, University of Alabama in Huntsville, Huntsville, AL 35899 USA. (phone: 256-876-5910; fax: 256-313-3717; e-mail: wilderj@eng.uah.edu).

A. Milenković is with the Electrical and Computer Engineering Department, University of Alabama in Huntsville, Huntsville, AL 35899 USA (e-mail: milenka@ece.uah.edu).

E. Jovanov is with the Electrical and Computer Engineering Department, University of Alabama in Huntsville, Huntsville, AL 35899 USA (e-mail: jovanov@ece.uah.edu).

II. SYSTEM ARCHITECTURE

The proposed SVEDECS system encompasses a network of miniature sensor nodes, a gateway node, and a centralized server, as depicted in Figures 2 and 3. Each sensor node in the SVEDECS network is comprised of a magnetic sensor, an embedded microcontroller, a radio, and an energy source. The magnetic sensor, which measures the earth’s magnetic field, tracks disturbances to this field as vehicles pass nearby. Taking advantage of sophisticated detection algorithms at the sensor nodes and collaborative processing at the gateway node, vehicles are classified in real-time as they pass target zones. These results are then wirelessly transmitted to a centralized server. In this way, the centralized server collects data from all gateway nodes in the SVEDECS network, evaluates this data, and then presents it. The centralized server can provide fully customized reports, from detailed information on each vehicle to high-level data that bins vehicle types by the hour, day, week, and month. One possible output from the centralized server is illustrated in Figure 4.

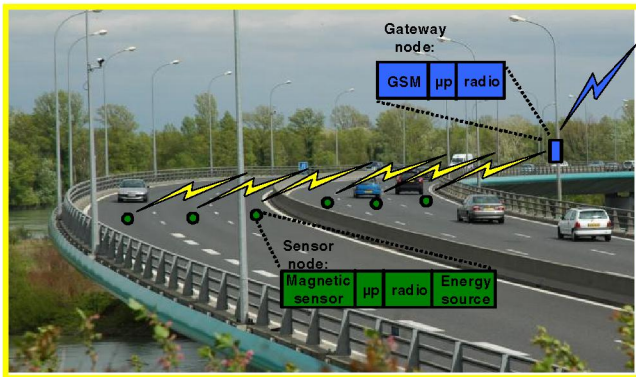


Fig. 2. SVEDECS System (Roadway).



Fig. 3. SVEDECS System (Server).

The benefits of a SVEDECS system are two-fold. First, SVEDECS is able to monitor and provide accurate, real-time traffic statistics based on the types of vehicles that use a given transportation system. Specifically, this design focuses on developing algorithms that will lead to the capability of categorizing vehicles according to the Federal Highway Administration’s 13-category classification scheme [4].

Currently, no other systems provide this capability. Gathering these data will facilitate research on traffic patterns, which, after analysis, will allow static and real-time decisions to be made to alleviate traffic congestion, will assist with infrastructure capacity planning, and will reduce overall infrastructure management costs. Furthermore, these data can be used to corroborate and improve existing methods used to forecast freight demand on state and federal roadways.

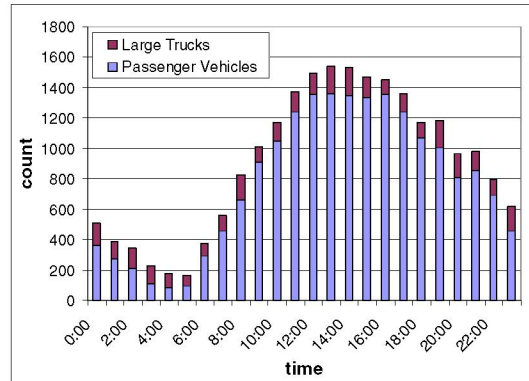


Fig. 4. Detailed report based on SVEDECS system.

The second benefit of this research is that SVEDECS lends itself to easy installation and deployment. As opposed to inductor loops, which are expensive to deploy and maintain, the smart vehicle traffic nodes of SVEDECS, due to their small size, can be installed on top of the roadway. Thus, this new system promises to significantly reduce the costs of initial installation, recurring maintenance, and traffic disruption as a result of these activities.

III. HARDWARE DESIGN

A magnetic sensor node in the prototype design consists of a Softbaugh DZ1611 development board [5] connected to a PNI magnetic sensor module, the MicroMag2 [6], as illustrated in Figure 5.

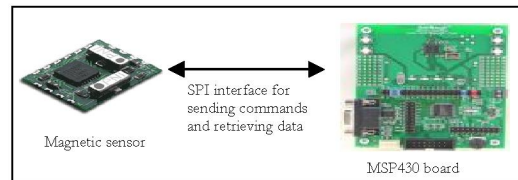


Fig. 5. Hardware design of a magnetic sensor node.

The Texas Instruments MSP430 microprocessor, located on the DZ1611 board, is the master device, which requests data from the slave device, the magnetic sensor. The interface between the master and slave devices is the Serial Peripheral Interface, or SPI. This is a synchronous communications protocol, where it is the responsibility of the master device to provide a clock to the slave device.

The magnetic sensor node consists of an ASIC

(application specific integrated circuit) with external transducers that convert magnetic field into voltage. The ASIC includes a counter, which is used to digitize the magnetic field reading, making an analog-to-digital converter unnecessary in this design. The operation of the magnetic field data acquisition is as follows. The master device requests a sample from a particular axis on the magnetic sensor by sending a command byte via the SPI interface. After the command is received by the ASIC, the counter increments until comparison logic detects a match with the voltage reading of the magnetic field. Then, the magnetic sensor notifies the master device that a data sample is ready for retrieval. This sample is then fetched by the master device via the SPI interface.

Design parameters for acquisition of the magnetic field sample by the sensor include the desired resolution of the digital result, as well as choosing from which axis to request a reading. Digital resolutions range from 5 bits to 12 bits, with maximum delays for output ranging linearly from 0.5 msec to 60 msec, as determined by the operation of the counter/comparison logic. For our application, we used a 9-bit result, requiring a 7.5 msec acquisition time for each sample.

After the magnetic field reading has been acquired from the MicroMag2, the microcontroller passes this data sample to the radio transceiver (CC2420) via the second UART port on the processor. Besides the SPI interface between the MSP430 and the CC2420, there's a control interface for handling FIFO and timer functions. The CC2420 FIFO itself is accessed via SPI, but the control lines are present in order to utilize communication requests between the MSP430 and the CC2420. Figure 6 illustrates the interface between the processor and the radio.

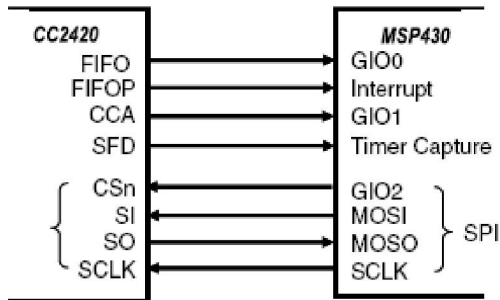


Fig. 6. Interface between the processor and the radio [7].

IV. SOFTWARE DESIGN

The software design for the magnetic sensor node consists of MSP430 microcontroller code written to control data gathering from the MicroMag2 magnetic sensor as well as data packet transmission/reception with the CC2420 Chipcon radio transceiver. The block diagram for the software design is shown in Figure 7.

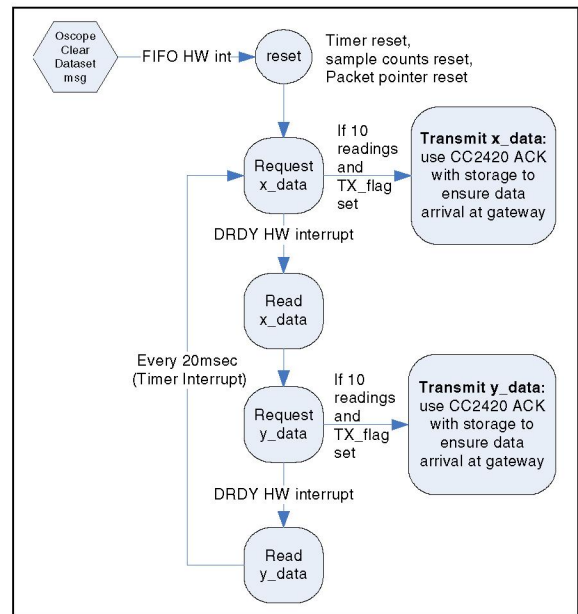


Fig. 7. Magnetic sensor node software algorithm.

The code illustrated in Figure 7 is interrupt-driven in the following way:

- An interrupt from the CC2420 notifies the MSP430 when a radio message has been received and is ready to be retrieved
- A 20msec Timer interrupt controls the sampling rate at 50 samples/second.
- An interrupt from the MicroMag2 notifies the MSP430 that a reading has been performed and data is ready to be retrieved.

In order for the design to function properly, the magnetic sensor nodes need to be time-synchronized. This is achieved by sending a radio synchronization message from the gateway node to the magnetic sensor nodes. Upon receipt of the sync message, the timer, the sample counter, and the packet pointer are all reset. Then, the processor moves to the request *x_data* state. In this state, the MSP430 sends a request for *x*-axis data to the MicroMag2 sensor and subsequently, the processor goes into low-power mode. When the data is ready on the MicroMag2, it sends a hardware interrupt (data-ready signal) to the MSP430, waking up the processor so that it will retrieve data from the magnetic sensor (the read *x_data* state). After the data is received through the SPI interface, the command is sent for *y*-axis data from the MicroMag2 (the request *y_data* state), and then the MSP430 goes into low-power mode. When the MicroMag2 has performed the data reading for the *y*-axis, it sends a hardware interrupt (data-ready signal) to the MSP430, waking up the processor so that it will retrieve data from the magnetic sensor (the read *y_data* state). After reading the *y*-axis data from the MicroMag2 through the SPI interface, the MSP430 goes to sleep. This completes one

cycle of gathering data samples from the x - and y -axis sensors of the MicroMag2. The timer then causes an interrupt so that the MSP430 wakes up and moves back to the request x_data state. This process repeats every 20 msec, for a sample rate of 50Hz.

This hardware prototype design uses point-to-point RF communication, based on a subset of the IEEE 802.15.4 protocol (ZigBee [8]). First, the MSP430 initializes which RF channel the CC2420 should use. It also turns on automatic packet acknowledgment. Additionally, other control registers are set via SPI communication in order to set the network ID that the radios will use, as well as to use a short-addressing scheme. In order to transmit a packet via the CC2420, the MSP430 first waits for the radio to go to an idle state. Once this occurs, the CC2420 transmit FIFO is flushed, in preparation to send a packet. Next, the MSP430 can begin transmitting packet data to the CC2420, on a byte-by-byte basis. A packet is comprised of the following: packet length (in bytes), frame-control field, sequence number, destination PAN ID and address, source address, and payload. The receive-packet routine from the CC2420 to the MSP430 is interrupt driven. Once the CC2420 receives data, it raises an interrupt to the MSP430. Then, the MSP430 reads the byte length of the incoming message through the SPI interface. Next, the frame-control field is read in order to make a determination of whether this is an acknowledgment packet or a data packet. If this is an acknowledgment packet, check that the CRC is correct. If this is a data packet, read the remaining data, which identifies the PAN ID and destination address, source address, message type, and group ID, followed by the packet payload.

For radio transmission of the magnetic sensor nodes, a type of super-frame is set up, so that the nodes transmit in an allocated time slot of a one-second frame. This is done so that packet transmission collisions are avoided. Control of the super-frame is through the timer, which sets a *transmit_flag* in each respective node, notifying the processor when it's allocated transmission time occurs.

As data is being gathered, a radio packet is built and made ready for transmission after 10 readings have been received. Once this occurs, the packet pointer is incremented and radio transmission occurs in the request x_data state if the *transmit_flag* is set (this is for a transmission of x -axis readings). Acknowledgements (ACKs) are built-in to the CC2420 transceiver in order to ensure that data is received by the gateway node. Thus, upon transmission of a packet, the processor waits to retrieve an ACK back from the gateway node. The timing of this is as follows: It takes approximately 1 msec to send data through the radio. It takes approximately 5 msec to receive an ACK. Thus, in order to transmit-ACK the packet, it takes about 6 msec. All of this can be done before the data-ready signal comes back from the MicroMag2, which takes about 7.5 msec. Radio packets are transmitted every 200 msec, or 5 times per second (since it takes 200 msec to build a packet filled with

10 data readings). Before the next packet is ready to be transmitted, if there are failed ACKs, then there can be ten failures before a packet needs to be stored in memory. The current design includes 5 seconds worth of packet storage space. The transmission for the y -axis occurs during the request y_data state, and is identical to the transmission for the x -axis data. Lastly, data is stored in memory until the *transmit_flag* is set, indicating that the sensor node's respective transmit slot in the superframe has become enabled.

V. RESULTS

Test measurements have been collected using the prototype system (Figure 8), under the field test scenario illustrated in Figure 9. The description of the test is as follows. Two sensor nodes are placed next to a roadway and separated by a distance of 110 inches. Magnetic field data are transmitted from the sensor nodes to a nearby gateway node, which is running TOSBase, a gateway application built on a TmoteSky mote [9] running TinyOS [10], which forwards serial data packets to a Java-based data acquisition program. As the data are received through the acquisition program, it is stored locally.

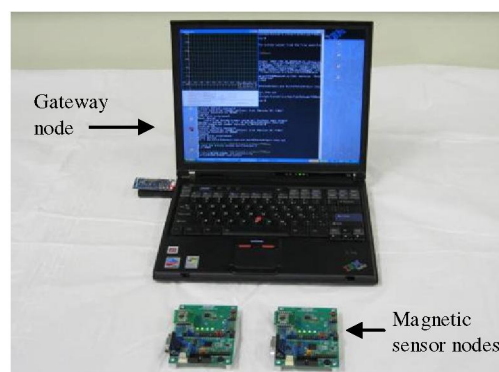


Fig. 8. SVEDEDCS prototype design.

Data gathered from the field test scenario indicate the success of the sensor nodes in capturing a vehicle signature. Figure 10 shows a signature of a vehicle as it passes near the sensors nodes, while Figure 11 provides results of a vehicle passing over the sensor nodes.

As can be seen from Figure 10, the signature of a vehicle passing near the sensors provides a smooth waveform. In this case, a peak-detection algorithm could be applied in order to determine like points between the two waveforms so as to determine speed. Figure 11 indicates a higher degree of excitation of the magnetic field disturbance due to the proximity of the vehicle to the sensor nodes.

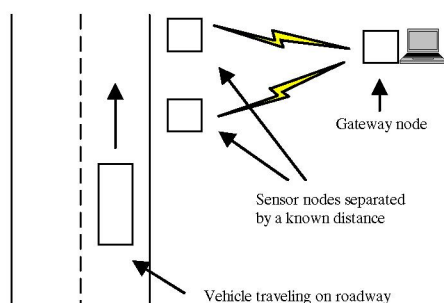


Fig. 9. Field test scenario.

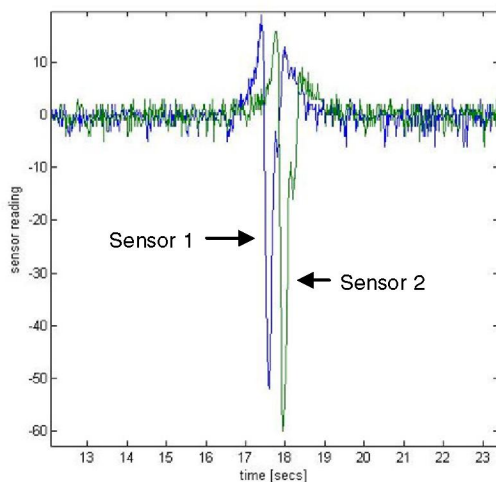


Fig. 10. Magnetic field disturbance of a vehicle driving near the sensor nodes.

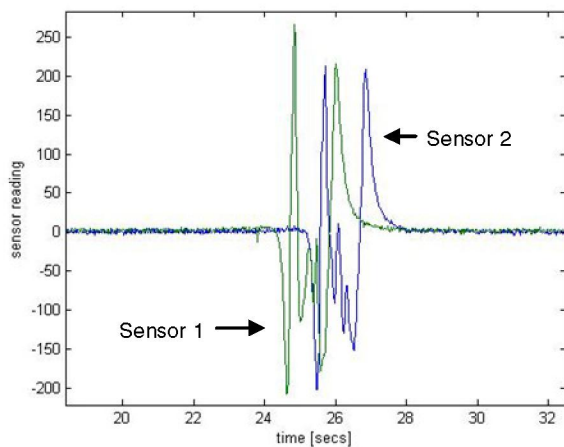


Fig. 11. Magnetic field disturbance of a vehicle driving over the sensor nodes.

VI. CONCLUSIONS

Due to ever-increasing population and congestion in urban areas, efficient traffic flow will always be a concern. One

way to alleviate these concerns on a constrained infrastructure is to monitor and control traffic flow based on gathered statistics of vehicle movement. In this paper, we propose a SVEDECS system which meets the required criteria of unobtrusiveness, provides low installation and maintenance costs, as well as offering continuous real-time data gathering.

As a positive first-step towards achieving a SVEDECS system, we have built a prototype design consisting of two sensor nodes based on a commercially-available hardware development board. This efforts illustrates proof-of-concept, providing indication that the method can be used for vehicle detection and speed determination.

The next steps toward achieving a SVEDECS system include onboard processing of vehicle signatures and feature extraction based on unique vehicle attributes. In this way, not only can vehicles be detected and their speed cataloged, but they can be classified according to type, providing important statistics that would interest any department of transportation. Providing the capability for onboard processing of magnetic signatures allows wireless communication from the sensor nodes to the gateway node to be drastically reduced, which would provide important power savings. Furthermore, energy harvesting of the sensor nodes should be investigated in order to provide long-life and autonomous operation. Finally, collaborative processing of sensor node data, as well as networking schemes, should be examined in order to take advantage of available resources.

REFERENCES

- [1] U.S. Census Bureau [<http://www.census.gov/ipc/www/idb/>].
- [2] U.S. Census Bureau, Population Division. *Annual Estimates of the Population for Incorporated Places over 100,000*. July 1, 2006.
- [3] Sinem Coleri, Sing Yiu Cheung, Pravin Varaiya. "Sensor Networks for Monitoring Traffic". Forty-Second Annual Allerton Conference on Communication, Control, and Computing, U. of Illinois, September 2004.
- [4] U.S. Department of Transportation, Federal Highway Administration. *Traffic Monitoring Guide, Section 4 – Vehicle Classification Monitoring*. May 1, 2001.
- [5] MSP430 Ultra Low-power Microcontroller [<http://www.ti.com/msp430>].
- [6] PNI MicroMag2 2-Axis Sensor Module [<http://www.pnicorp.com>].
- [7] Aleksander Milenković, Chris Otto, Emil Jovanov. "Wireless sensor networks for personal health monitoring: Issues and an implementation". *Computer Communications*, 2006.
- [8] Zigbee [<http://www.zigbee.org>].
- [9] Moteiv [<http://www.moteiv.com>].
- [10] TinyOS [<http://www.tinyos.net>].