

IOWA STATE UNIVERSITY

CANdroid

Group Dec 14-02

Alec Johanson

John Shelley

Ahmmad Shelley

Client

Vermeer

Advisors

Arun Somani

Koray Celik

1 PROBLEM

Modern off-highway/agricultural systems use outdoor rated LCD displays to implement user interfaces. These current solutions are either too expensive, or put users at risk being an unsafe distance from the machines. Vermeer wishes to replace these expensive display systems with a more inexpensive solution as well as provide a more flexible development environment. A J1939 CAN system is used on most Vermeer systems, but would like to be expandable to other protocols. Vermeer has expressed their wish of emphasizing research and development with solutions utilizing android systems due to their prior experience with this technology.

In more layman's terms: Vermeer currently uses a CAN bus to communicate data from a machine, such as a baler, to a VT (Virtual Terminal) in the cab of the tractor that is towing it. These VTs are wired, and proprietary. This makes them expensive and exposes the CAN bus to the harsh Agricultural environment. Replacing the Virtual Terminal with an android device has the potential to greatly reduce cost. While making the connection to the android device wireless can help prevent erosion and weakening of the system.

1.1 FUNCTIONAL DECOMPOSITION

The main communication pertaining to our project is between a VT (virtual terminal) and a controller on the unit (in our target case a 605 baler). The controller communicates to the VT through a CAN bus starting with a series of hand shakes, setting up the initial UI. Then during normal operation, the controller reads data from point to point connections with sensors. The controller analyzes the data and send UI update information through the CAN bus. The connection between the controller and the VT will now be wireless utilizing android instead of a proprietary VT system.

1.2 SYSTEM REQUIREMENTS

Functional	Non-Functional
<ul style="list-style-type: none">• Stream > 10% CAN bus load at 250Kbps throughput• Withstand outdoor environment• Withstand high vibration environment• less than \$300 unit cost	<ul style="list-style-type: none">• User friendly for the android operator• Extensible to other CAN protocols

2 SOLUTION

The solution for this problem is to use a PIC32MX795 microcontroller to receive CAN data from the controller on the operating machine. The PIC32 communicates with a WRT Node module that runs a web socket for the Android device to communicate with wirelessly.

2.1 SYSTEM ANALYSIS

Controller The controller is what obtains the information from outside diagnostic tools on the machine. The controller then analyzes this information to determine what updates are needed for the VT. Another function of the controller is to send commands to these peripherals. The controller is where most of the functional work for the overall system is done.

RF Bridge The RF Bridge is what receives information from the Controller through the CAN Bus. This will then distribute the information over a wifi signal. Since a cloud connection is less feasible due to operating conditions, an AdHoc similar network approach will take place.

Our solution for the RF Bridge was to use a WRT Node, running a version of embedded linux to act as a server for the android device. The WRT Node can not directly receive a CAN signal so a PIC32 microprocessor translates the CAN data to UART for the WRT Node to receive.

Android Device The Android device then takes the input from the RF Bridge. The Android tablet will then output the User Interface in order to visualize what the controller is receiving from the diagnostic tools such as hay, corn, or bean harvesters and such.

Details of the modules are expanded upon in the following sections. Shown in Figure 1, is a high level block diagram of our solution.

$$\begin{aligned}(x+y)^3 &= (x+y)^2(x+y) \\ &= (x^2+2xy+y^2)(x+y) \\ &= (x^3+2x^2y+xy^2)+(x^2y+2xy^2+y^3) \\ &= x^3+3x^2y+3xy^2+y^3\end{aligned}\tag{2.1}$$

Phasellus viverra nulla ut metus varius laoreet. Quisque rutrum. Aenean imperdiet. Etiam ultricies nisi vel augue. Curabitur ullamcorper ultricies

2.2 HEADING ON LEVEL 2 (SUBSECTION)

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

$$A = \begin{bmatrix} A_{11} & A_{21} \\ A_{21} & A_{22} \end{bmatrix}\tag{2.2}$$

Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem.

2.2.1 HEADING ON LEVEL 3 (SUBSUBSECTION)

Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium. Integer tincidunt. Cras dapibus. Vivamus elementum semper nisi. Aenean vulputate eleifend tellus. Aenean leo ligula, porttitor eu, consequat vitae, eleifend ac, enim.

HEADING ON LEVEL 4 (PARAGRAPH) Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim.

3 LISTS

3.1 EXAMPLE FOR LIST (3*ITEMIZE)

- First item in a list
 - First item in a list

- * First item in a list
 - * Second item in a list
- Second item in a list
- Second item in a list

3.2 EXAMPLE FOR LIST (ENUMERATE)

1. First item in a list
2. Second item in a list
3. Third item in a list