WHITE PAPER
Intel® Core™ i5 Processor
Intel® Vending Design Reference Kit
Silkron* Vendron* SDK
Vending Machine Industry

Increasing Vending Machine Computing Performance with Intel® Architecture

Simplify migration with hardware and software development kits

Increase the Performance of Next-Generation Vending Machines using Intel® Core™ processors

As vending machines evolve and support new capabilities, such as large-screen digital signage and video analytics, developers need to add more computing performance to existing designs. This can be achieved with less effort when using the Intel® Vending Design Reference Kit and Silkron* Vendron* software development kit (SDK).

The reference kit includes a modular board that allows vending machine makers to interface their existing vending hardware to an Intel® Core™ processor, which delivers the performance required by new applications. If so desired, the current vending software can also be ported to the Intel® processor, potentially eliminating the need for other CPUs or microcontrollers in the system. Moreover, manufacturers can move away from proprietary controller boards to commercial off-the-shelf (COTS) boards with an Intel processor, which can significantly reduce development and inventory costs.

The Vendron SDK is a package of tools that supports digital signage and typical vending machine functions, as well as advanced remote management features. The SDK facilitates vending software development – from product inventory to payment control – and lets vending machine operators manage and monitor features more easily.

This white paper explains how the Intel Vending Design Reference Kit interfaces to common vending hardware and discusses the Vendron application programming interface (API), which simplifies the integration of legacy and new applications.

Table of Contents

1.0 Intel® Vending Design Reference Kit Overview

Today, many manufacturers use proprietary controller boards to control and monitor vending machine hardware and devices, like change dispensers. Moving forward, these boards will need a performance upgrade in order to keep up with emerging requirements, such as digital signage, nutrition labeling and video analytics. The Intel Vending Design Reference Kit provides an open standards-based approach to deliver all the necessary performance using Intel Core processors with slight or no modification to machine hardware.

For more information about the reference kit, download the white paper at http://download.intel.com/embedded/applications/digitalsignage/325109.pdf

The reference kit features the "Vending SandBox", a board that interfaces the Intel® architecture platform to the machine hardware via standard I/O: digital I/O, Multi-Drop Bus (MDB) and RS232/RS485 serial ports. The Vending SandBox, illustrated in Figure 1, connects to vending devices such as sensors, switches, card readers and cash dispensers. The Vendron SDK supplies an API for communicating with vending devices and integrating vending application software, making it easier to port software to an Intel® architecture processor. The SDK also has tools for configuring and testing the hardware configuration.

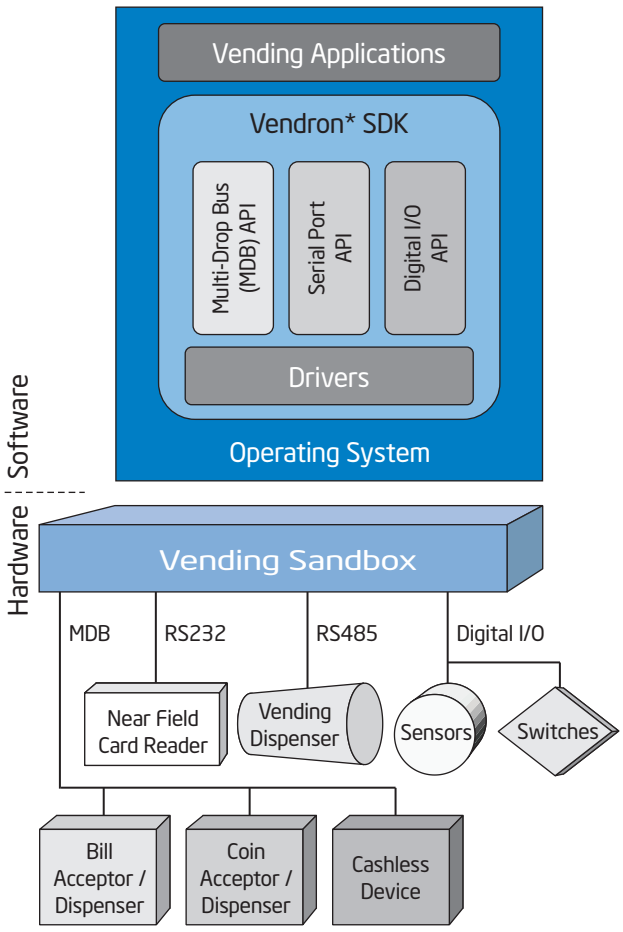

Figure 1. Vending Machine Based on Intel® Architecture

1.1 Vending SandBox Overview

The Vending SandBox is an interface board connecting to the Intel® architecture platform via USB and connecting to the vending machine via standard I/O, as shown in Figure 2. The default configuration for the SandBox provides two MDB ports (upstream and downstream), eight digital inputs, eight digital outputs, two serial RS232 ports and one serial RS485 ports. All of the I/O is modular and more ports can be added, if required.

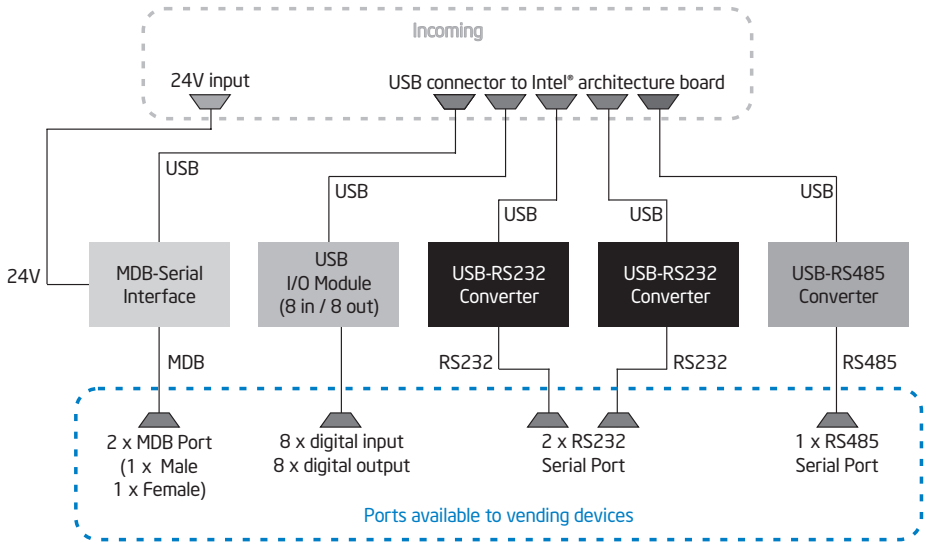

Figure 2. Reference Kit Block Diagram

The following three sections provide a brief description of the standard I/O ports and how they are deployed by the Intel Vending Design Reference Kit.

1.2 Digital Input/Output

The Vending Sandbox employs a USB I/O module supporting sixteen isolated general purpose digital I/O, split evenly between input and output functions. They transmit and receive binary signals to/from vending machine devices, such as sensors, LEDs, switches, etc. The USB I/O module is USB 1.1/2.0 compatible and bus-powered. Further specifications are listed in Table 1.

Parameters	Specifications
Digital inputs (8 channels)	▪ Input voltage: 5V min (30VDC Max) or Dry Contact ▪ Isolation protection: 2,500 VDC
Digital outputs - sink (NPN) (8 channels)	▪ Isolation protection: 2,500 VDC ▪ Output Voltage: 24 - 30 VDC
Power consumption	▪ Typical: 5V @ 200mA ▪ Maximim: 5V @ 300mA
Operating temperature	▪ 0 - 60˚C (32 - 140˚F)
Storage temperature	▪ -20 - 70˚C (-4 - 158˚F)
Storage humidity	▪ 5 - 95% relative humidity, non-condensing

Table 1. Specifications for Digital Input/Output

1.3 Multi-Drop Bus (MDB) Serial Interface

The Multi-Drop Bus (MDB) communication protocol has become the primary method of communication inside modern vending machines, interconnecting different modules such as bill acceptors, card readers and coin changers. The Vending SandBox contains a USB-to-MDB translation device that is controlled by the Intel architecture platform via USB and provides an MDB-compatible serial interface for the machine hardware. The MDB serial link operates at 9,600 bits per second (bps) and transmits 8 bit frames with start and stop bits as shown in Figure 3. Other specifications are shown in Table 2.

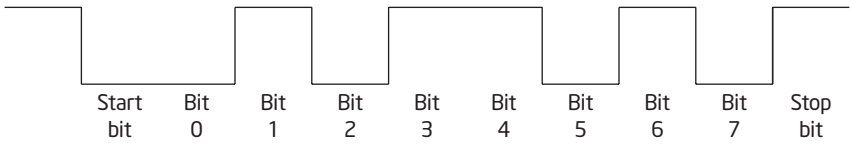

Figure 3. MDB Communication Format

Parameters	Specifications
Timing	▪ Inter-byte (max): 1ms ▪ Response (max): 50ms ▪ Master polling times every 150 - 300 ms
Power supply	▪ Input voltage (min): 20 VDC ▪ Input voltage (nominal): 24 - 34 VDC (or 24 - 30 VAC) ▪ Current (idle): < 0.05A ▪ Current (operating): < 0.5A (for one second)
Power consumption	▪ Typical: 5V @ 200mA ▪ Maximim: 5V @ 300mA

Table 2. Multi-Drop Bus (MDB) Serial Interface

1.4 Serial RS232/RS485 Ports

In addition to MDB, RS232 and RS485 serial ports are commonly used in vending machines to link to touch screens, pin pads, coin slots, ticket printers and so on. RS232 is the more popular interface, since it was introduced first. RS485 port usage is growing because it supports longer distances at faster communication rates and provides better noise immunity than RS232. The Vending SandBox has two USB-to-RS232 converters and one USB-to-RS485 converter.

The two RS232 standard serial ports are USB 1.1/2.0 compatible, bus powered, true-bidirectional and employ a DB9 (also known as DE-9) male connector, whose pinout is detailed in Table 3. The maximum cable length for USB is five meters; RS232 is 15 meters.

Pin	Signal Name	Description
1	DCD	Data carrier detect
2	RxD	Receive data
3	TxD	Transmit data
4	DTR	Data terminal ready
5	SGND	Signal ground
6	DSR	Data set ready
7	RTS	Request to send
8	CTS	Clear to send
9	RI	Ring indicator

Table 3. RS232 Serial Interface Pinout

The RS485 standard serial port is USB 1.1/2.0 compatible and bus powered, and supports up to three megabits per second (Mbps). It also employs a DB9 male connector, as shown in Figure 4; however, there are only two signal pins, a differential pair (improves noise immunity) for data performing both transmit and receive. The serial link can be configured for seven or eight data bits and for even, odd or no parity. If the Vending SandBox is the first or last device in a multi-drop RS485 system, it is necessary to add a 120Ω terminating resistor between pin 1 and pin 3 in order to satisfy RS485 termination requirements.

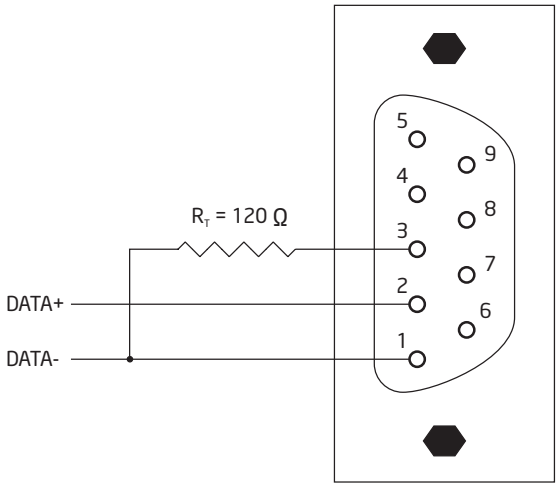

Figure 4. RS485 Serial Port Connector

Pin	Signal Name	Description
1	DATA-	Transmit/Receive data, negative polarity (May require a 120Ω Terminating Resistor)
2	DATA+	Transmit/Receive data, positive polarity
3	Terminating Resistor	Only required when the system is the first or last device in a multi-drop RS485 system
4	N/A	Not applicable
5	GND	Signal ground
6	N/A	Not applicable
7	N/A	Not applicable
8	N/A	Not applicable
9	PWR	+5V DC output (250 mA maximum)

Table 4. RS485 Serial Interface Pinout

2.0 Silkron* Vendron* SDK Overview

The Vendron software development kit (kit) from Silkron is a toolset for developing vending machine software, including special commands for digital signage and advanced remote management features. Further, Vendron software handles cash and cashless transaction types, such as credit card, prepaid cards and in the future, mobile phones.

Supporting digital signage, the Vendron SDK manages all of the images and video content displayed, including ads, product images, auxiliary product information and transaction processing menus. Using the SDK, operators can remotely download ads and play them according to a playlist or specific time slots.

The scope of the Vendron SDK is large, and for the purposes of this paper, the following only reviews the API calls to initialize and communicate with the Vending SandBox I/O ports.

2.1 Initialize Serial Port

Int siqSP_init(LPCSTR portName, LPCSTR strBaudRate, LPCSTR strDataBits, LPCSTR strParityType, LPCSTR strStopBitsType, LPCSTR strFlowType, int spReceived)

- Initializes the connection with a serial port and returns connection ID if successful.

Parameter	Description	Example
portName	Communication port	"COM1"
strBaudRate	Baud rate setting	"9600"
strDataBits	Data bits setting	"8"
strParityType	Parity type setting	"None"
strStopBitsType	Stop bit setting	"1"
strFlowType	Flow type setting	"Off"
spReceived	Function to call when data is received from the serial port	spDataReceived

2.2 Send Hex Value to Serial Port

Void siqSP_sendHex(int splnitID, LPCSTR hex)

- Sends a hexadecimal value to the serial port.

Parameter	Description	Example
splnitID	Serial Port ID	123
hex	Hexadecimal values in string separated by <space>	"01 03 04"

2.3 Send String to Serial Port

Void siqSP_sendString(int splnitID, LPCSTR string)

- Sends a string to the serial port.

Parameter	Description	Example
splnitID	Serial Port ID	123
string	Character string	"abc123"

2.4 Unitialize Serial Port Connection

Bool siqSP_deinit(int splnitID)

- Unitializes the connection with the communication port of the Serial Port and returns true if completed successfully.

Parameter	Description	Example
splnitID	Serial Port ID	123

2.5 Initialize USB I/O Module Connection

Bool siqIO_init(LPCSTR strPort)

- Initializes the connection with the USB I/O Module Connection and returns true if completed successfully.

Parameter	Description	Example
strPort	Communication port for the USB I/O Module	"COM2"

2.6 Read Bit from USB I/O Module

Bool siqIO_get(int bit)

- Reads input signal (bit 0 - 7) from USB IO Module and return true if signal on and vice versa.

Parameter	Description	Example
bit	Specific bit from I/O device	1

2.7 Set USB Output Signal

Void siqIO_set(int bit, bool on)

- Sets the output signal to USB I/O module.

Parameter	Description	Example
bit	Specific bit from IO device	1
on	Boolean value to trigger	true

2.8 Unitialize USB Module Connection

Bool siqIO_deinit()

- Unitializes the connection with the USB I/O module and returns true if completed successfully.

2.9 Initialize Connection to Bill/Coin Acceptor

Bool siqMDB_init(LPCSTR portName, int mdbReceived, int mdbError)

- Initializes the connection with the communication port of the bill/coin acceptor device and returns true if completed successfully.

Parameter	Description	Example
portName	Communication port for MDB device	"COM1"
mdbReceived	Function to call when data is received from the bill/coin acceptor	cashData
mdbError	Function to call if an errors occurs	cashError

2.10 Enable the Bill/Coin Acceptor

Void siqMDB_enable()

- Enables the bill/coin acceptor device.

2.11 Disable the Bill/Coin Acceptor

Void siqMDB_disable()

- Disables the bill/coin acceptor device.

2.12 Return Change from Coin Acceptor

Void siqMDB_change(double value)

- Instructs coin acceptor to return change.

Parameter	Description	Example
value	Amount to return	0.5

2.13 Unitialize Bill/Coin Acceptor Connection

Bool siqMDB_deinit()

- Unitializes the connection with the communication port of the bill/coin acceptor.

3.0 Summary

Vending machine developers requiring more computing performance can migrate to powerful Intel Core processors that have the performance to run emerging applications, such as digital signage and video analytics. The Intel Vending Design Reference Kit and Vendron SDK from Silkron reduce the effort to integrate an Intel architecture–based platform. The reference kit is available through an Intel loaner program, details of which can be obtained through an Intel field sales representative.

For more information about Silkron software products, please visit **www.silkron.com/smart_vending**

For more information, please visit **www.intel.com/go/digitalsignage**

Printed in USA 0611/TB/TM/PDF Please Recycle 325626-001US