Advanced Solutions
for Document Processing

Panini My Vision X P: Technical Overview

The introduction of the 2 pocket My Vision X machines extends the capability and market reach of the My Vision X series of document scanners. The added features and technical capabilities managed through the My Vision X API are detailed in this document.

New Features

- **Two Sorting Pockets**: Sorting criteria based on MICR reading or recognition from image. Each pocket is capable of holding up to 100 checks and is fully adjustable in length to hold different size items on both versions. A full pocket sensor is available, permitting the operator to focus their attention on document processing and customers' needs.
- **Fully Open Scalability**: Three different speed versions, two options for feeder capacity, two options for AGP printing, and two options for sorting pockets. All versions easily upgradeable via software.
- **Enhanced Feeder**: featuring up to 120 documents per batch.
- **Intelligent Endorsement**: The Panini AGP ink-jet printer adds a true receipt printer to the sorting management capability. In addition, the printing capability becomes "intelligent": characters to be printed can be chosen through the application following MICR reading or image information obtained during the document processing. A special reversible track permits endorsement of the processed item on a defined printing area.

Technical Specifications

My Vision X P FAMILY Technical Specifications	
PERFORMANCE	My V X P-30: Up to 30 DPM My V X P-60: Up to 60 DPM My V X P-90: Up to 90 DPM (USB2.0 interface; Final performance depends on applications, printing and sorting modes)
AUTOMATIC DOCUMENT FEEDER	2 versions: - holding up to 30 documents for automatic batch one hand insertion - holding up to 120 documents, with pressure plate (full feeder version) Double feed detection Auto-tuning separator rollers designed to process varied thickness of documents and to compensate for wear
INKJET PRINTER	Rear Ink-jet AGP (Advanced Graphic Printer) Printing capability, two versions:

	AGP 2: two lines text/graph in any vertical position AGP 4: four lines text/graph for full 1/2'' height Alphanumeric characters: all MS Windows fonts Reg CC Compliant Printed info captured by image Intelligent printing based on MICR or image on a defined printing area
POCKETS	2 versions (based on the same hardware platform) : - 1P: one exit pocket capable of holding 100 documents - 2P: two exit pockets capable of holding 100 documents each Flexible Sorting Options, based on MICR and/or image Full pocket sensor for both versions
OPTIONS	OCR Recognition: OCR-A, OCR-B, E13B , CMC7 recognition engine Barcode Recognition: Code 39, Interleaved 2/5, EAN8, EAN13, UPCA, UPCE, Code 128 COLOR IMAGE: Fast Color mode
SCALABILITY	Upgrades (Speed, document feeder capacity, 2 pockets, ink jet) and Options are fully scalable via software upgrade throughout the range
WORKSPACE	Extremely compact footprint – Ideal for teller and back office installation
DOCUMENTS SPECIFICATIONS	Height: Min: 2.12'' – Max: 4.17'' Length: Min: 3.14'' – Max 9.25'' Weight: Min: 16# – Max: 32#
INTERFACE	USB2.0 port/Backward compatible with USB 1.1 RS232 Port for external device connection. (SW/FW developments on request.)
MAGNETIC READER	E13B/CMC7/Automatic Panini MICR Plus® exclusive technology MICR based sorting
IMAGE CAPTURE	Scanning: Contact Image Sensor (CIS) technology (front and rear) Image format: Bitmap in B/W, 256 shades of gray. TIFF, JPEG and Group IV compression Image resolution: 100 or 200 dpi Advanced dynamic threshold Dual Image: 4 images in one document pass Front/rear image based sorting
SOFTWARE TOOLS	Panini Vision API control running on: Windows 2000 S.P. 3, XP, Vista (32 bit) with USB 2.0 or USB 1.1 ICR Vision function for image snippet definition and download Easy integration of ICR/Barcode/OCR recognition technology
APPLICATION PC MINIMUM REQUIREMENTS	My V X P 30, My V X P 60: Intel Pentium IV, 1.2 GHz, 256 Megabytes Ram My V X P 90: Intel Pentium IV, 2 GHz, 256 Megabytes Ram
DIAGNOSTIC FEATURES	On board Diagnostics: tests the functionality of the scanner Power-on Self Testing: Automatic self testing and photocells calibration when powering the unit
MAINTENANCE	Maximum accessibility to every component to minimize MTTR Total access to scanner and track area Firmware upgradeable via PC

INPUT VOLTAGE	Autosensing from 100 to 240 Vac, 50 to 60 Hz
DIMENSIONS AND WEIGHT	Height: 6.88'' Width: 5.43'' Length: 10.39'' Weight: 5.73#

My Vision X API Modifications

Pocket Handling

This API version is able to manage a My Vision X with 2 pockets. With this kind of machine the application can decide the destination pocket of a document. Basically there are two ways to decide the document's destination:

1. Using the MICR information
2. Using the image (OCR, ICR, CAR/LAR, etc.) information

The pocket is decided by the application calling the API function *SetPocket*. This function has to be called during the SET_ITEM_OUTPUT message.
To decide the pocket using the MICR information, the application has to enable the MICR option. As previously required by the API, when the application receives the SET_ITEM_OUTPUT message, the application has to call the *SetPocket* function to decide the document's destination pocket. The SET_ITEM_OUTPUT is always sent after the MICR_AVAILABLE message. Thus, during the SET_ITEM_OUTPUT message, the application already has the MICR information to decide the pocket.
To decide the pocket using the image information, the application has to enable at least one snippet *for decision*. Both the front and the rear side can be used to extract the snippet for decision. The enable *for decision* means that the pocket is decided after the image acquisition, when the snippet is sent to the application. Compressed images cannot be used to decide the pocket. To enable the snippet for decision you have to set the ***Enable*** field of the SNIPPET_STRUCTURES to the **SNIPPET_ENABLE_FOR_DECISION** value.
Enabling this kind of snippet, the API sends the SET_ITEM_OUTPUT message after the SNIPPET_READY message. Thus, during the SET_ITEM_OUTPUT message, the application already has the image information (OCR, ICR, CAR/LAR….) to decide the pocket for the document.
When a snippet is requested with ENABLE only, the sequence of the document messages is:

- NEW_DOC;
- MICR;
- **SET_ITEM_OUTPUT;**

- IN_POCKET;
- **SNIPPET_READY;**
- IMAGE_READY;
- DOC_COMPLETE.

Using the ENABLE_FOR_DECISION value the sequence will be:

- NEW_DOC;
- MICR;
- **SNIPPET_READY;**
- **SET_ITEM_OUTPUT;**
- IN_POCKET;
- IMAGE_READY;
- DOC_COMPLETE.

It's important to notice the position of SET_ITEM_OUTPUT and SNIPPET_READY messages. In the first case the snippet message is considered as a down-stream message (not used for decision) and is signalled after pocket decision. In the second case it is an up-stream message (used for decision) and is signalled before the SET_ITEM_OUTPUT that is the message where the application decides the document's pocket.
The MICR-based decision permits to the application to maintain the right DPM performance. The performance could be affected by a very long delay of the application before calling *SetPocket()*. The machine cannot feed a new document if the application doesn't call the *SetPocket()* function for the previous one. Then, later is the call and more is the reduction of the DPM performance.
The 2 pocket machine introduces the reverse gear of the document in the track. In some cases, the machine needs to retreat the document in the track to complete the operations. If the application takes a long time to decide the pocket, the machine will stop the document just after the image capture device waiting for the decision. If the destination is the default pocket, the machine restarts track moving forward the document in the pocket. If the destination is the second pocket, the machine makes a reverse gear of the document, stop the motor when the leading edge is positioned before the mechanical switch, then open the switch and then restart the motor to put the document in the pocket.
The Image-based decision usually cannot maintain the right DPM (except for MVX 30 DPM). For example, when the pocket is selected by the image information and the destination pocket is the second, the device have always to retreat the leading edge of the document, before the mechanical switch position, and then restart the track to introduce the document in the pocket.

Smart Jet

This API introduces a new way to manage the Printing operations. Before this API version the Printer could only be used like an up-stream device. This means that the application has to decide the information to print on the document before the document is fed.
Now it's possible to use the Printer as a down-stream device. This means that the application can decide the printing information based on the MICR or on the Image information (OCR, ICR, CAR/LAR…).
To enable this kind of printer the application has to add a new flag (PRINTER_ENABLE_SMART_JET) inside the *bPrintEnable* field of the *DeviceParameters* structure.
When Smart-Jet is enabled, the application has to call the *SendPrinterData* function only during the SET_ITEM_OUTPUT event, before the *SetPocket* call.
Example:

```
DWORD DocID;
char ApiErrorString[200];
case WMPAR_SORTER_SET_ITEM_OUTPUT:
      DocID = (DWORD) LPARAM;
      // When Smart-Jet enabled this function has to be called
      // only here, before SetPocket() call.
      SendPrinterData(DeviceID, … )
      SetPocket( DeviceID, DocID, 1 );
      break;
```

To decide the printing data using the MICR the application has to enable the MICR option.
To decide the printing data using the Image the application has to enable at least one snippet per decision (explained in Pocket Handling section). Both the front and the rear side of document can be used to extract the snippet for decision (like for a 2 pocket machine).
The Smart Jet could affect the DPM performance, especially when based on image information.
The Smart Printer could require (surely in case of printing from image) a reverse gear of the document. This happens when the position of the document is after the printing position decided by the application. In this case the machine has to retreat the document to recover the right position and start the printing operation.
There is a mechanical limit for the reverse gear of the document. When the application takes a very long time to decide the pocket, the machine stops the document in the track after the end of the MICR signal acquisition. When the printer data are sent to the My Vision X, the firmware retreats the document in the track to recover the right printing position, stop the track, and the restart the motor to make the printing and conclude the document processing. If the printing position could not be recovered, the firmware signals an ERR_PRINTER error

Reverse Gear

Enabling some *DeviceParameters* option the machine could need to implement a reverse gear of the document in the track. This kind of behaviour is present only on machine endowed with 2 pockets or with mechanical reversibility. There are essentially 4 cases:

1. ***Printer with Trailing edge alignment***.
   If the printing data are Trailing edge aligned, the machine needs to know the length of the document to decide where is the printer start position. When the trailing edge leaves the first photocell sensor, the machine computes the print position. If this position is beyond the printer head, the machine has to make a reverse gear of the document to bring the document to the right position. This operation is very fast with minimum lack of DPM performance.
2. ***Smart Printer.***
   The Smart Printer could require (surely in case of printing from image) a reverse gear of the document. This happens when the position of the document is beyond the printing position decided by the application. In this case the machine has to retreat the document to recover the right position and start the printing operation.
3. ***Pocket Decision.***
   The reverse gear is requested when the leading edge of the document is beyond the mechanical switch, inside the default pocket, and the application chooses the second pocket for that document. If the image capture is enabled, the machine has to wait for the end of the image operations, then stops the track, makes a reverse gear of the document and finally put the document in the second pocket.

This kind of reverse gear could happen combined. You can see the 1 combined with 3, or the 2 with 4, the 3 with 4, etc…
The reverse gear could affect the DPM performance depending on the option enabled and the application behaviour.

Full Pocket

This API enables the use of the full pocket sensor found on My Vision X P devices. When enabled and if a Full Pocket condition is detected, the machine doesn't stop and the application receives a DEVICE_ERR_FULL_POCKET exception. It's the application's responsibility to manage this situation (*Stopfeeding* call, switch the destination pocket, User Interface warning).
The full pocket sensor is enabled through the following firmware parameter

- ID = 9 Enable Full-Pocket detection (0-1, default is 0)