February 2001

WRL
Technical Note TN-58

A Simple CMOS Camera for Itsy

Joel F. Bartlett

Western Research Laboratory 250 University Avenue Palo Alto, California 94301 USA

A Simple CMOS Camera for Itsy

Joel F. Bartlett*

Abstract

Itsy is a high-performance platform for research in "off-the-desktop" computing. One of the attributes of the base hardware is that it can easily be extended through the use of "daughter cards." This technical note describes the packaging and electronics for one such extension: a simple CMOS camera.

1. Introduction

Itsy[2] is a flexible, high performance handheld computer designed for research in "off-the-desktop" computing. Approximately the size of a Palm V, it has a 200MHz Intel StrongARM 1100 processor, 32 MB of RAM, 32 MB of flash memory, and runs Linux. Hardware extension is done by building daughter cards that interface directly to the memory bus and the StrongARM GPIOs.

As CMOS "camera on a chip" image sensors have become available, there has been significant interest in integrating cameras into handheld devices. Itsy's memory size and performance also suggest that it could be more than just a simple camera. It could act as a handheld computer vision platform, capable of significant image processing at the time a picture is taken. In addition, public interest in Itsy has exposed project members to some opportunities for powerful, camera-centric, handheld devices.

With this motivation, an effort was started to add a snapshot camera to Itsy with an eye towards demonstrating an innovative way to integrate a camera into a handheld device, and encouraging its use. The goal of this effort was to minimally impact the Itsy base system, yet not compromise the image sensor's performance. While many digital cameras use specialized image processing electronics[4], this camera would use Itsy's existing processor and memory for all image processing and control functions. In the sections that follow, the package and electronics design will be discussed, and the paper will close with a few sample images and comparisons to other systems.

2. Package Design

The starting point for the design was the base Itsy system shown at the top of Figure 1. Working with the existing base electronics and minimizing redesign of the case, Studio Red created the design shown at the center and bottom of Figure 1. The CMOS imager and lens are contained in a hinged extension to the left end of the case that lengthens it by 29mm. The rest of the electronics are contained on a daughter card that thickens the case by 4.5mm.

Figure 1: Three Itsy Pocket Computers. At the top of the picture is an unmodified Itsy. It is 118mm x 65mm x 16mm, and weighs (including battery) 130g with a molded plastic case. Below it are two units with the camera extension that are 147mm x 64mm x 20.5mm and weigh (including battery) 195g with a SLA epoxy case, or 267g with a NC tooled aluminum case.

Itsy's reflective gray-scale display is used as the viewfinder, displaying in real-time the scene in front of the lens. Typically when the camera is used, the photographer holds the display horizontally in their hand at a good viewing angle. The lens/imager assembly is then rotated into a vertical position so the lens points forward (middle of Figure 1) at the subject. The hinge allows smooth motion through 180° so the display viewing angle need not be compromised to correctly position the lens. When the camera is not in use, the lens/imager assembly is folded flat (bottom

* Compaq Computer Corporation Western Research Laboratory, 250 University Avenue, Palo Alto, CA 94301. joel.bartlett@compaq.com.

of Figure 1) and the camera is easily stored in a purse or pocket.

There is minimal interference with other Itsy function as the unit fits as well in the hand as an unmodified Itsy. The least objectionable way to increase the size of Itsy is to lengthen it as has been done here. The camera has a minimal impact on the unit's width and height.

This view camera inspired design has a number of advantages. First, since the reflective display is illuminated by ambient light that is typically overhead, the preferred viewing position is approximately horizontal, which is how the camera uses it.

Like a camcorder with an LCD viewer, the camera design allows the photographer to remain in eye contact with the subject and allows others to look over the photographer's shoulder while they are composing the picture.

Finally, the camera can be used in a variety of ways. It can be placed on a table in front of the photographer for unobtrusive candid shots. The lens/imager assembly can be rotated up to 180° for use in making greater than 1:1 close-up shots, or not rotated at all (bottom of Figure 1) and the camera used for self-portraits.

3. Inside the Package

In order to implement this design, the camera's electronics are segmented into two circuit boards as shown in Figure 2. The board on the left contains the CMOS image sensor and the lens. In order to minimize the width of this board, a conventional lens mount that surrounds the image sensor was not used. Instead, an aluminum tube is glued directly to the glass cover of the image sensor and the lens is threaded into it. Opaque plastic tape is applied around the tube to prevent extraneous light from entering the image sensor as shown in Figure 4.

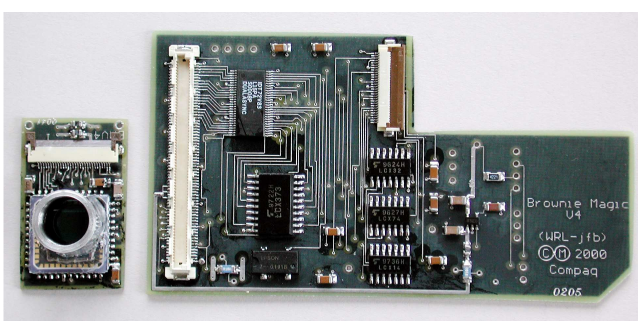


Figure 2: Image sensor and daughter card electronics.

The lens was chosen with the idea of using the camera for snapshots. The lens is a Sunex DSL900C f/3.0 9.9mm lens. It can be focused from infinity to closer than 2" by rotating it. Quality color images require an infrared filter, so the final component of the lens assembly is a Sunex IRC21-8R IR cut-off filter that is directly bonded to the glass cover of the image sensor. Since there is no variable aperture or auto focus system, the camera operates silently.

The rest of the electronics that control the image sensor and provide the datapath to memory are located on the daughter card shown on the right side of Figure 2.

The camera is assembled as shown in Figures 3 and 4. In Figure 3, the daughter card is attached to the Itsy motherboard. The daughter card's impact on the thickness of the camera is minimized by careful selection and placement of its components so that they fit in the space available between the daughter card and the motherboard. A thirty-wire 0.5mm flat flex cable connects the daughter card and the image sensor. In order to thread this cable through the hinge, two wires are removed and the cable split as show in Figures 3 and 4.

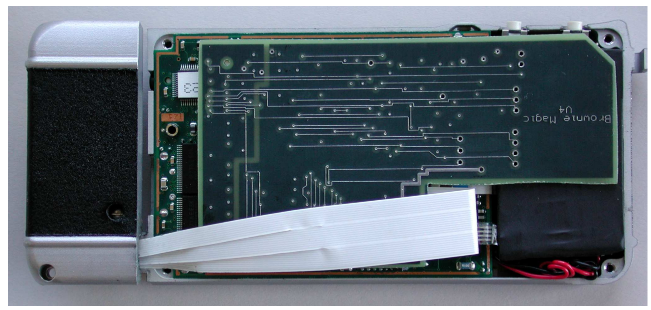


Figure 3: Itsy shown face down with the case back off, the daughter card installed, and cabled to the image sensor card.

Figure 4: Itsy shown face up with the front of the image sensor case removed.

Even the simplest cameras have moving parts. Here, the lens is rotated in its mount to focus it, and the image sensor/lens arm rotates on a hinge connecting it to the main body of the camera. Based on experience with other cameras and electronics, users have an expectation about how things should move: freely without overshoot, yet hold their position when stopped. This is accomplished by lubricating the joint with the appropriate grade of damping grease: Nyogel 774VL on the lens threads and Nyogel PG-44A on the body hinge.

With this overview of the package design and electronics segmentation complete, attention can now turn to the details of the electronics.

4. Photobit PB-0300 CMOS Image Sensor

The design of the electronics was driven by the capabilities and requirements of the PB-0300[5] image sensor. With only a few digital controls, and external VCC and clock, this "camera-on-a-chip" can produce digital output at up to 30 frames/second with VGA (640H x 480V) resolution. Unlike earlier CCD image sensors, this sensor requires no external control circuitry, analog-to-digital converters, or non-standard reference voltages as these functions are integrated into the sensor chip. The sensor's integral gain and exposure controls allow pictures to be taken under widely varying lighting conditions without the need for an adjustable aperture lens or a mechanical shutter.

The "Image Sensor" schematic details the interface to the PB-0300. Signal names are shown in a sans-serif font and those signals that are active low have a "~" as the first character of their name. The signals of interest are:

GND	ground
VCC	analog and digital power
CLKIN	24 MHz master clock input
STANDBY	active high to set chip on standby
~RSTPB	active low to reset the chip
SDATA	Serial Host Interface Port™ data
SCLK	Serial Host Interface Port™ clock
PIXCLK	pixel clock for output data
LINEVLD	active high on image data output
FRAMEVLD	active high during image processing
DOUT0	data output bit 0 (LSB)
DOUT1	data output bit 1
DOUT2	data output bit 2
DOUT3	data output bit 3
DOUT4	data output bit 4
DOUT5	data output bit 5
DOUT6	data output bit 6
DOUT7	data output bit 7 (MSB)

Finding an appropriate method for supplying power to the PB-0300 required some experimentation as analog and digital power levels for the chip are specified at 5.0V. While it will produce acceptable pictures at lower voltages, it cannot produce them when powered by Itsy's 3.05V system power. One solution would be to use a charge pump to produce 5.0V power. This was rejected because it was expected that future 3.3V "camera-on-a-chip" devices would be available from Photobit and other vendors and that signal level conversions, required in a mixed voltage system, would complicate the system.

Instead, a linear regulator is used to convert the raw battery voltage to 3.3V that is used to power both the analog and digital portions of the image sensor. This provides a high enough voltage to provide good pictures, yet is close enough to the level of Itsy's system power to allow signals (but not power connections!) to freely cross between the image sensor and the rest of the system. The sensor power is noticeably cleaner than the Itsy system power and this helps improve picture quality. Not obvious on the schematic are the separate analog and digital power and ground planes that are provided on the image sensor's circuit board to further reduce noise.

5. Itsy Daughter Card Interface

The interface to the Itsy daughter card[9] provides access to all the unused StrongARM 1100[3] GPIOs (general purpose I/O pins) and to the full memory address and data buses. In the interest of building the simplest interface possible, the only control logic on the daughter card is for functions that could not be handled by the Linux driver.

The STANDBY, ~RSTB, FRAMEVLD, SCLK, and SDATA signals from the PB-0300 are directly connected to dedicated GPIO pins. The flexibility of the StrongARM GPIOs allows the Serial Host Interface Port™ (SHIP) protocol (compatible with I²C™) to be implemented entirely by the driver.

The bulk of the logic on the daughter card is dedicated to moving data from the PB-0300 to the StrongARM memory bus. The FIFO is two 8-bit pixels wide and is implemented by a dual 2,048 x 9 IDT72V83 asynchronous FIFO that is directly connected to bits 0-15 of the memory bus. Since the FIFO is the only register that need be read or written, a minimal control design suffices. The FIFO is visible to the StrongARM as static memory bank 3, and the static memory control register is configured as follows (assuming a 100 MHz memory clock):

RTx	0: nonburst ROM or Flash EPROM
RBWx	0: 32 bit ROM bus width
RDFx	1: 2 clocks from address to data valid
RDNx	0: ignored as device is not written
RRRx	1: 2 clocks to get off the bus

When configured this way, any read (or write access) to any physical address in static memory bank 3 results in ~DCCS3 (an active low selecting the memory back) being asserted by the StrongARM, which starts the read cycle on the FIFO and two pixels are transferred to the memory bus.

The FIFO is reset and status information is communicated to the StrongARM by four dedicated GPIO pins:

~RSTFIFO	active low that resets both FIFOs
~FULL	active low on FIFO overflow
~EMPTY	active low on FIFO empty
~HALF	active low on FIFO half full

The final GPIO control signal to the daughter card is ~CAMERA. This is an active low that turns on the clock and linear regulator for the PB-0300, and enables the daughter card memory interface. When the camera is not in use, this signal is high to disable these components and save power.

With these signals defined, attention will now shift to using them to take a picture.

6. Taking a Picture

The camera is visible to application programs as the device /dev/pb0300. When it is opened, the driver sets ~CAMERA to 0. This starts the 24 MHz clock, supplies power to the PB-0300, and enables the daughter card memory interface.

The application starts by issuing an ioctl (I/O control command) to reset the camera. The driver implements this operation by pulsing ~RSTPB. This initializes the PB-0300 and runs it in its default configuration, where it supplies the StrongARM with 30 VGA size images/second.

Before taking pictures, the application may need to make configuration changes on the PB-0300. This is done by issuing a number of ioctl requests to read and write the PB-0300's registers. The driver performs these functions via a software implementation of the SHIP protocol using the SCLK and SDATA GPIOs. The Itsy is the protocol master and it provides the clock via SCLK, a dedicated output pin. Clock pulse duration is controlled by busy waiting in the driver, using the StrongARM's 3.6864 MHz operating system timer. Both Itsy and the PB-0300 use the SDATA line to transfer data a bit at a time. Typically, SDATA is configured by the driver as an input pin, so its value is either driven by the PB-0300 or driven to 1 by the pull up resistor. The only time Itsy drives this signal is when the protocol requires the driver to write a 0 data value, in which case it forces SDATA to 0 by configuring it as an output pin with a value of 0 for one bit time. Addition ioctl commands are used to allocate image buffers in the driver and select the mode of operation: still picture or video clip.

Once PB-0300 configuration is complete, the application issues a read request to the device to take a picture. This is posted in the driver, but no action is taken until a falling edge on FRAMEVLD causes an interrupt. The PB-0300 is now between frames and not trying to transfer any data into the FIFO, so the FIFO can be reset by pulsing ~RSTFIFO. The interrupt handler completes by enabling falling edge interrupts on ~FULL and ~HALF.

As pixel values are generated by the PB-0300, they are moved into memory in the following manner (see the "DC/FIFO" schematic). When PIXCLK falls to 0 and LINEVLD is 1, pixel data is available at DOUT[0..7] and ~WRITE falls to 0.

When ~WRITE falls, data is latched in the octal D-type latch (IC1 74373) and either ~WRITE07 or ~WRITE815 falls to 0 which starts the transfer of the latched data into the appropriate half of the FIFO.

When ~WRITE rises to 1, the value of the D-type flip flop (IC12A 7474) is inverted which results in the other half of the FIFO being selected for holding the next pixel.

When the FIFO become half full, ~HALF falls to 0 which interrupts the processor. The interrupt handler then reads 2048 pixels from the FIFO and stores them into memory. The FIFO is visible to the processor as static memory bank 3. Each load from a virtual address mapped to this bank causes ~DCCS3 to fall to 0, and the FIFO places two pixels of data (16 bits) onto the memory data bus.

After the PB-0300 outputs the last pixel of the image, FRAMEVLD drops to 0, which interrupts the processor. The FIFO is emptied into the driver's buffer and then the read completes. If the driver is configured to record a video clip, then the interrupt handler will start the next read so no frames are lost. Otherwise, it will disable the FIFO interrupts as the PB-0300 images are being ignored.

7. Processing the Image Data

A picture appears to the application program as a VGA sized (640 x 480) array of unsigned bytes. Cameras implemented using the monochrome version of the PB-0300 require no further processing as this array is the gray-scale image. Color cameras require additional processing as the image sensor is covered by a color filter array as shown in Figure 5. Each pixel in the image sensor looks through a red, blue, or green color filter and returns that value. When printed as color image, since each pixel has only a red, green, or blue value, the result is like that shown in Figure 6. In order to produce RGB values for each pixel, the missing color values are recovered by interpolation[1] and the result is shown in Figure 7.

Since the camera does not have dedicated image processing electronics, in camera color recovery is application dependent or can be deferred until after the images have been removed from the camera.

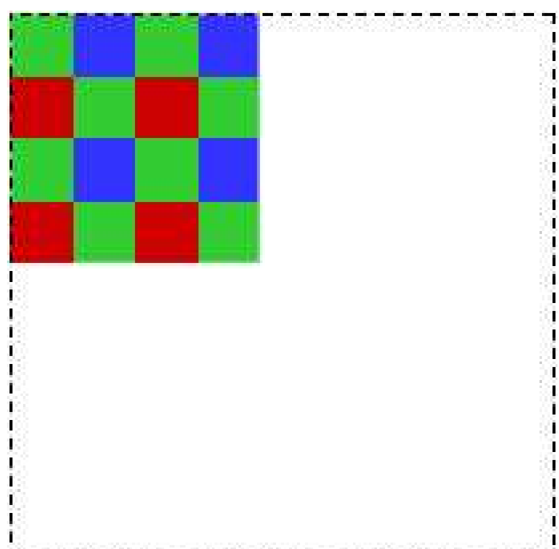


Figure 5: A portion of the Bayer Pattern of the PB-0300 color filter array, where each colored square is a color filter covering one pixel.

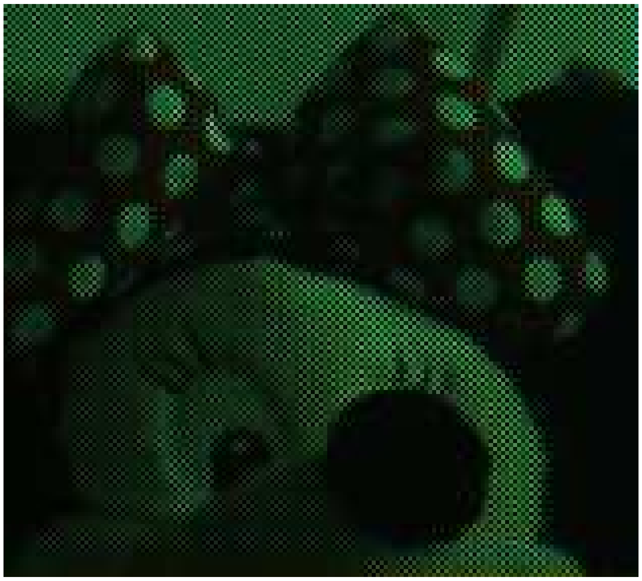


Figure 6: A magnified portion of an image returned by a color PB-0300, where each pixel has only a red, green, or blue value.

Figure 7: Image color recovery from the image data of Figure 6 using a simple pixel-averaging algorithm.

8. Initial Applications

Software to implement simple still and video cameras has been implemented. Three still pictures taken by the camera are displayed in Figure 9. They demonstrate the picture quality available from the camera and its ability to take close-ups.

Since the camera is capable of recording successive images, short video sequences (limited only by the size of Itsy's memory) can also be recorded. By selecting an image size smaller than VGA, frame rates faster than 30 frames/second are possible. One example, is a short feature called "The Break", Figure 10, where CIF size images (352W x 288H) were shot at 60 frames/second in two two-second long bursts. A slow motion QuickTime movie was then assembled using Adobe ImageReady.

Figure 9: Three sample still images: an outdoor scene, a shop window, and a well-worn quarter.

Figure 10: "The Break" shows a pool break at 1/12 real-time. In this frame, the cue ball has just hit the head of the rack

The most interesting application for the Itsy camera is handheld computer vision. To demonstrate this, our colleagues at Compaq's Cambridge Research Laboratory supplied a neural network-based face detector[6] that detects any number of frontal, upright faces in an image. The detector was paired with the Itsy still camera code and the result is a "face finding camera" whose results are shown in Figure 11.

Figure 11: After taking a picture, the "face finding camera" draws a rectangle around each frontal, upright face it finds in the image.

Another application that could easily be hosted on an Itsy with a camera is the gesture pendant[8] designed at Georgia Tech. This is a wearable camera that contains sufficient processing power to recognize hand signals and convert them into control commands to nearby appliances. It was prototyped using a video camera, a 900MHz video transmitter/receiver pair, and a desktop computer. Since Itsy has sufficient processing power and memory to recognize a gesture, its camera could capture the gesture, and its IRDA link could directly control the desired appliance.

Lest one think the power required for such a device is impossible in a wearable device, the Itsy still camera software that uses the display as a real-time viewfinder is able to continuously run for three hours before discharging the standard 2.2 watt-hour battery.

9. Related Work and Extensions

With the Itsy camera's design in mind, attention can now turn to putting it in context with related work. One of WRL's research partners at EPFL integrated a mobile camera[7] into Itsy. Their electronics were more complex than those described in this paper as they were interfacing two sensor chips with less integrated functions that the PB-0300.

Two commercially available cameras for handheld systems are the Kodak PalmPix™ for Palm and the eyemodule™ for the Handspring™ Visor™. The PalmPix image sensor is the same size as Itsy's, but the display must be held vertically in a poor viewing position in front of the user's face in order to operate it. The eyemodule's display is held in a horizontal position like Itsy's for good viewing, but its image sensor is only 320x240 pixels. The major limitations of each of these designs are processor performance and camera-to-memory bandwidth. The low performance processors in both handhelds require that the camera extensions provide dedicated image processing electronics. The camera is in turn connected to the processor on both systems by a 115,200 bit/second serial interface. Contrast this with Itsy, where the camera is directly connected to the StrongARM memory bus and is capable of sustained transfers at 8.8Mbytes/second (640x480 pixel frames at 30 frames/second). This bandwidth is clearly required for taking video clips, but it is also desirable to provide real-time updates to the viewfinder.

While this camera is centered on the Photobit PB-0300, the design could be easily adapted to a number of other CMOS image sensors. For example, the Agilent HDCS-2030, the Kodak KAC-0310, and the Zoran Pixelcam PCS2112 all are powered at 3.3V, have I²C compatible controls, and provide digital image output data. Minimal changes to the electronics would be required to control any of these chips.

10. Conclusions

The camera extension of Itsy is a success. The changes to the base Itsy unit have been minimized, yet the resulting object is pleasing to the eye and encourages users to pick it up. It produces pictures of acceptable quality and the utility and power of the system are demonstrated by the initial computer vision experiments. The positive results and the simplicity of the design should encourage others to integrate cameras into their "off-the-desktop" systems.

Acknowledgements

This work would not have been possible without the skills and interest of my colleagues at WRL: William Hamburgen, Marc Viredaz, Wayne Mack, Deborah A. Wallach, Lawrence Brakmo, and Andreas Nowatzyk. Studio Red turned the project's unfocused ideas about a view camera into a striking design and functional prototype packages. Jim Rehg and Uli Kremer provided the neural network-based face detector. Wendy Bartlett provided editorial assistance for this technical note. I thank you all.

References

[1] Jim Adams, Kevin Parulski, and Kevin Spaulding. *Color Processing in Digital Cameras*, IEEE Micro, November-December 1998, pp 20-29.

[2] Joel F. Bartlett, Lawrence S. Brakmo, Keith I. Farkas, William R. Hamburgen, Timothy Mann, Marc A. Viredaz, Carl A. Waldspurger, and Deborah Wallach. *The Itsy Pocket Computer*. Research report 2000/6, WRL, Compaq, Palo Alto, CA (USA), October 2000.

[3] Intel. *Intel® SA-1100 Microprocessor Technical Reference Manual*, September 1998, available via www.intel.com.

[4] Shoji Kawamua, *Capturing Images with Digital Still Cameras*, IEEE Micro, November-December 1998, pp 14-19.

[5] Photobit Corporation. *PB-0300 1/3-inch CMOS Active-Pixel Digital Image Sensor*, March 2000 (Version 3.0), available via www.photobit.com.

[6] H. A. Rowley, S. Baluja, and T. Kanade. *Neural Network-Based Face Detection*. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 20, No. 1, 1998, pp 23-38.

[7] Olivier Saudan. *Interfacing a mobile camera with the Itsy pocket computer*. 7th semester project, LAMI, EPFL, Lausanne (CH), February 1999.

[8] Thad Starner, Jake Auxier, Daniel Ashbrook, and Maribeth Gandy. *The Gesture Pendant: A Self-illuminating, Wearable, Infrared Computer Vision System for Home Automation Control and Medical Monitoring.* In *The Fourth International Symposium on Wearable Computers*, IEEE, October 2000, pp 87-94.

[9] Marc A. Viredaz. *The Itsy Pocket Computer Version 1.5: User's Manual*. Technical Note TN-54, WRL, Compaq, Palo Alto, CA (USA), July 1988.

Bill of Materials

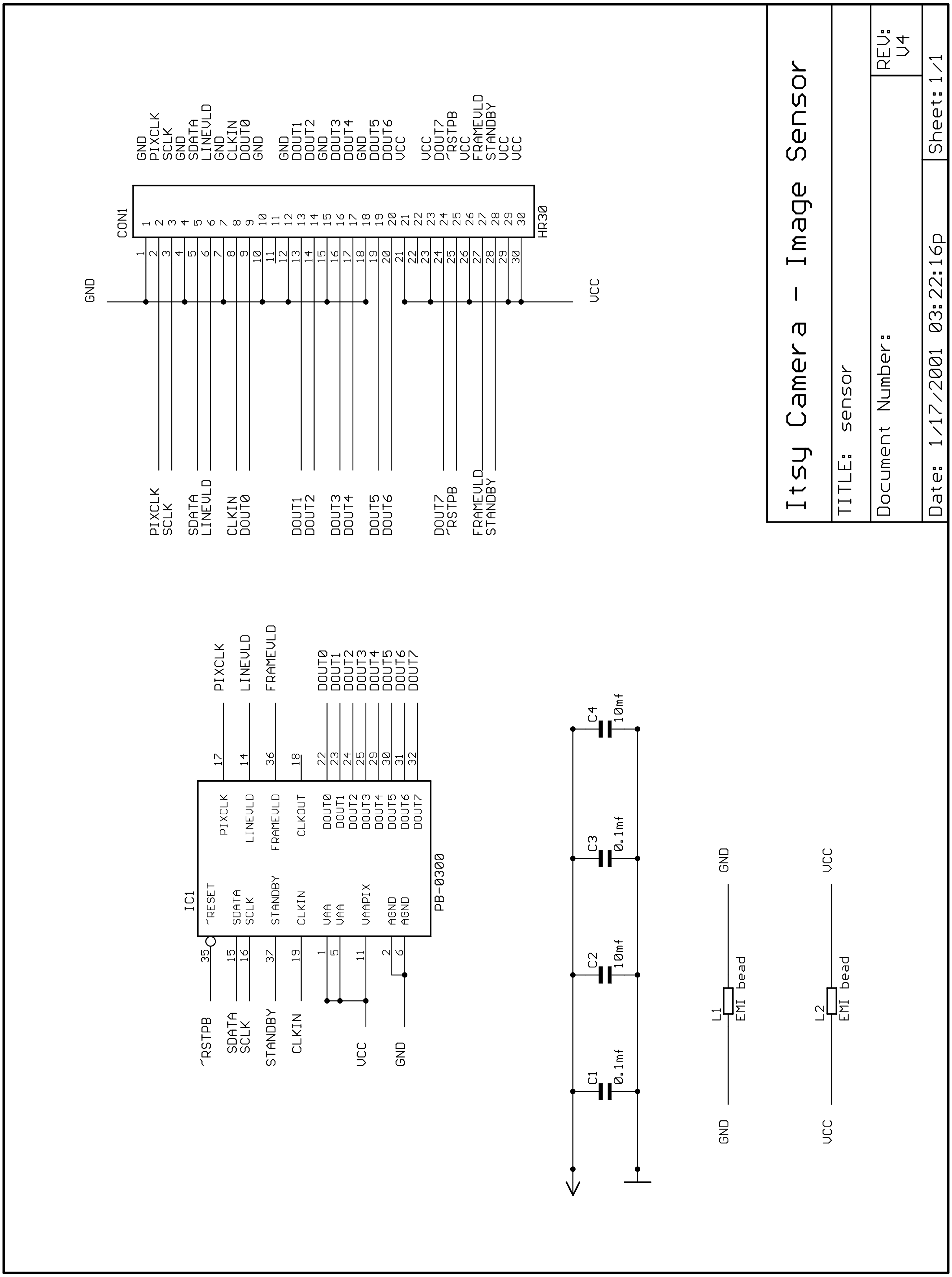
Components labeled "C*n*" and "R*n*" are capacitors and resistors whose value is as shown on the schematic. Components labeled "JP*n*" are test points. The rest of the components are as follows:

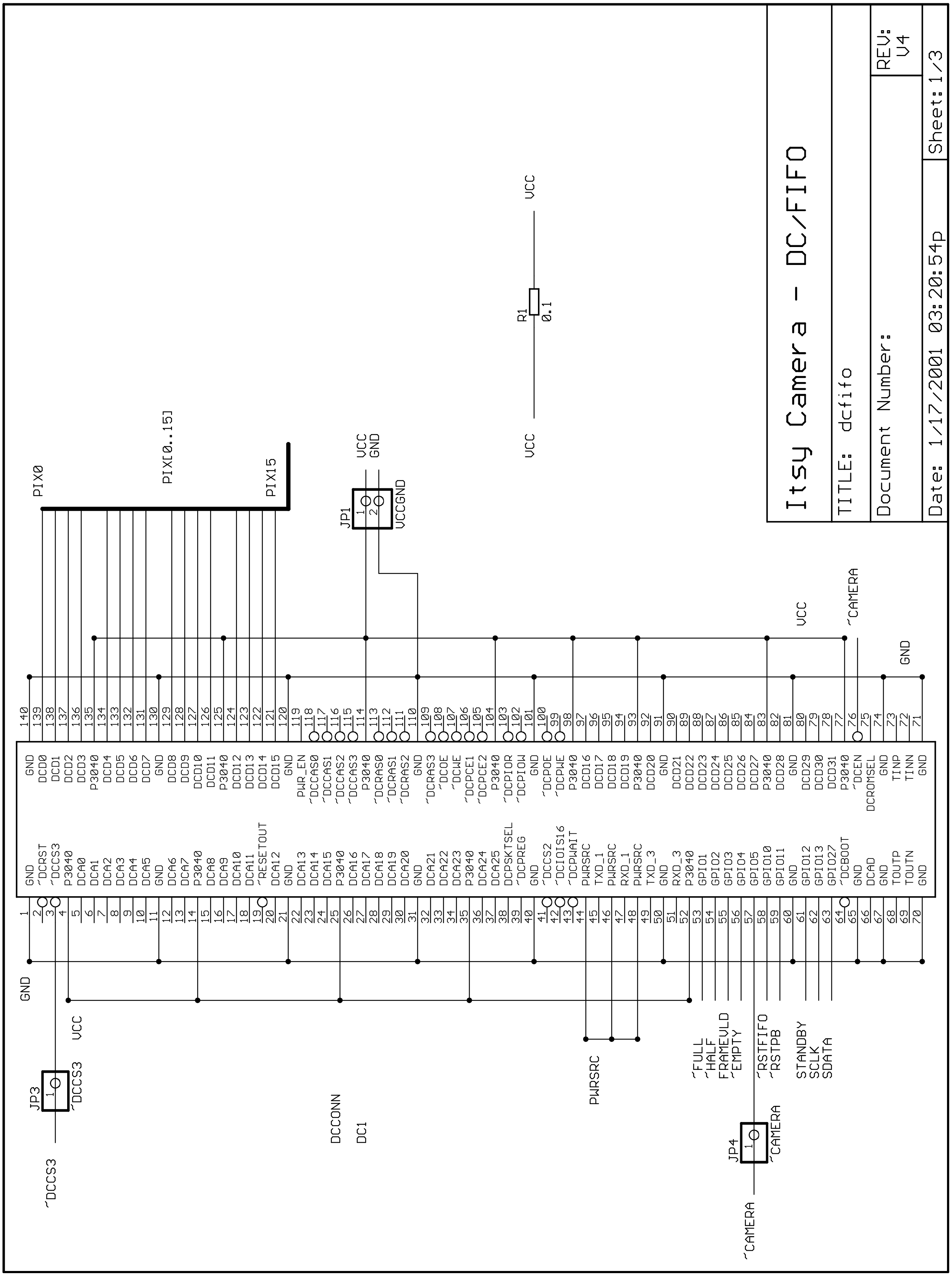
Sensor Circuit Board

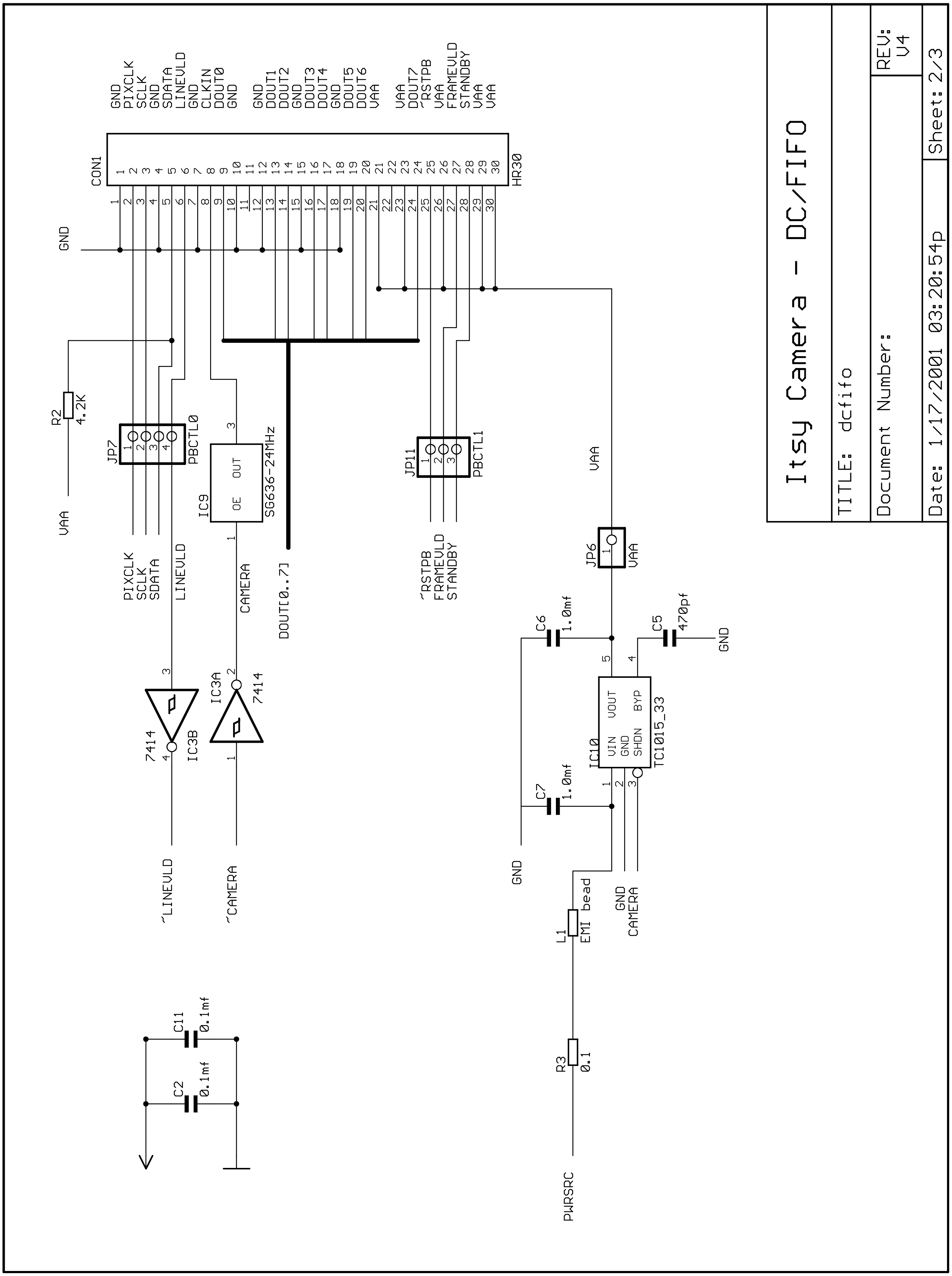
CON1	Hirose 30-pin 0.5mm top contact connector
IC1	PB-0300CCC
L1 & L2	600 ohm @ 100MHz ferrite beads join split power and ground planes

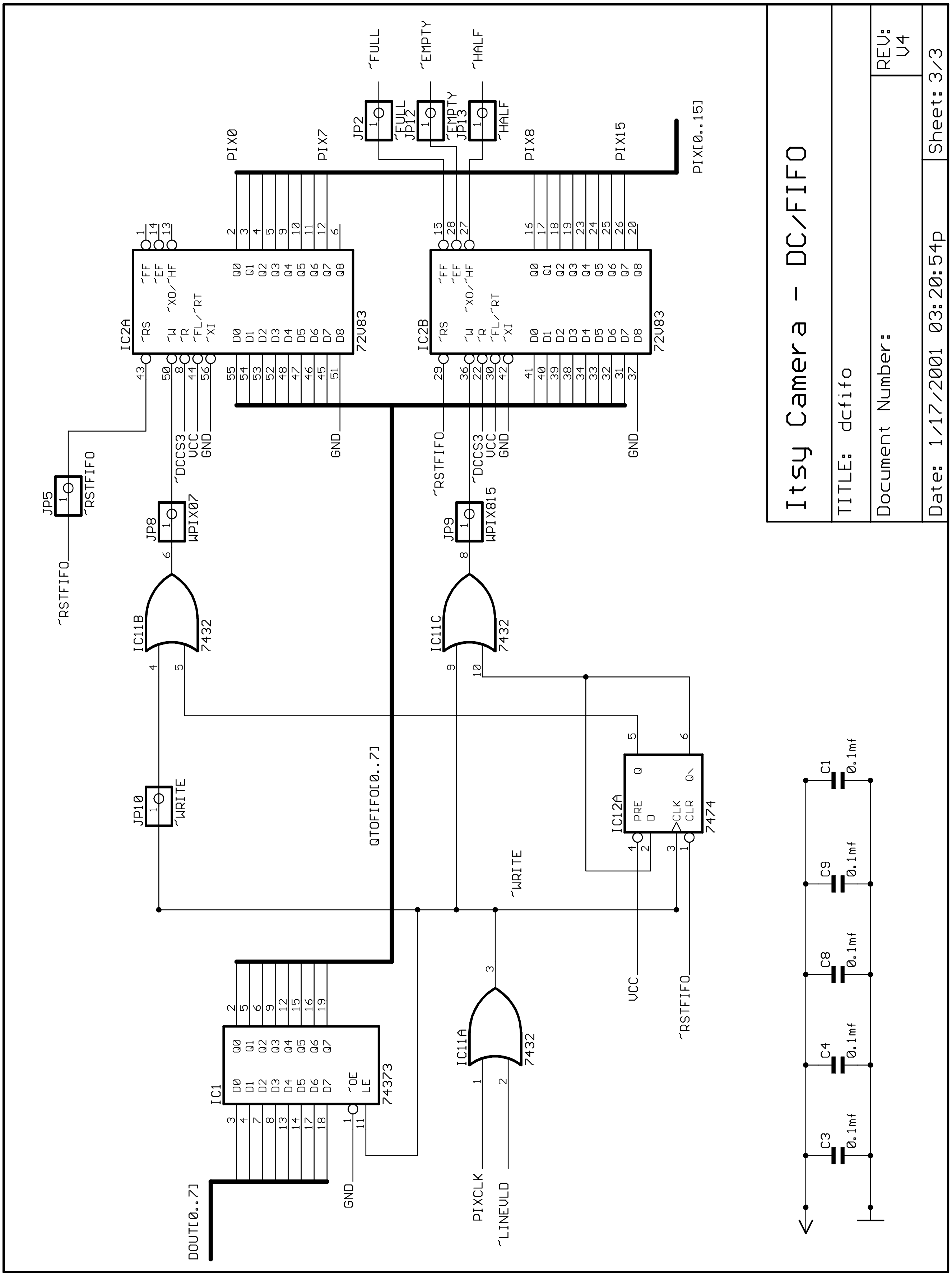
DC/FIFO Circuit Board

CON1	Hirose 30-pin 0.5mm bottom contact connector
DC1	daughter card connector
IC1	74LCX373FN octal D-type latch
IC2	IDT72V83L15PA dual 2,048 x 9 FIFO
IC3	74LCX14FN hex Schmitt inverter
IC9	SG636 PCE 24MHz oscillator
IC10	TC1015_33VCT regulator
IC11	74LCX32FN quad 2-input or
IC12	74LCX74FN dual D flip-flop
L1	600 ohm @ 100MHz ferrite bead
R1 & R3	0.1-ohm precision resistors for power measurement


Itsy Camera - Image Sensor
TITLE: sensor
Document Number:
REV: V4
Date: 1/17/2001 03:22:16p
Sheet: 1/1
CON1
HR30
GND
VCC
GND PIXCLK SCLK GND SDATA LINEVLD GND CLKIN DOUT0 GND GND DOUT1 DOUT2 GND DOUT3 DOUT4 GND DOUT5 DOUT6 VCC VCC DOUT7 ~RSTPB VCC FRAMEVLD STANDBY VCC VCC
IC1
PB-0300
~RESET SDATA SCLK STANDBY CLKIN VAA VAA VAAPIX AGND AGND
PIXCLK LINEVLD FRAMEVLD CLKOUT DOUT0 DOUT1 DOUT2 DOUT3 DOUT4 DOUT5 DOUT6 DOUT7
~RSTPB SDATA SCLK STANDBY CLKIN VCC GND
C1 0.1mf
C2 10mf
C3 0.1mf
C4 10mf
L1 EMI bead
L2 EMI bead


Itsy Camera - DC/FIFO
TITLE: dcfifo
Document Number:
REV: V4
Date: 1/17/2001 03:20:54p
Sheet: 1/3
DCCONN
DC1
JP1
JP3
JP4
R1
0.1
PIX[0..15]
PIX0
PIX15
PWRSRC
VCC
GND
VCCGND
~DCCS3
~CAMERA
~FULL
~HALF
FRAMEVLD
~EMPTY
~RSTFIFO
~RSTPB
STANDBY
SCLK
SDATA


Itsy Camera - DC/FIFO
TITLE: dcfifo
Document Number:
REV: V4
Date: 1/17/2001 03:20:54p
Sheet: 2/3


Itsy Camera - DC/FIFO
TITLE: dcfifo
Document Number:
REV: V4
Date: 1/17/2001 03:20:54p
Sheet: 3/3
IC2A
72V83
IC2B
72V83
IC11A
IC11B
IC11C
7432
IC12A
7474
IC1
74373
JP2
JP5
JP8
JP9
JP10
JP12
JP13
~FULL
~EMPTY
~HALF
PIX0
PIX7
PIX8
PIX15
PIX[0..15]
QTOFIFO[0..7]
DOUT[0..7]
~RSTFIFO
~DCCS3
VCC
GND
WPIX07
WPIX815
~WRITE
PIXCLK
~LINEVLD
C1
C3
C4
C8
C9
0.1mf