MMPAD-CdTe User Manual

MMPAD-CdTe User Manual

Overview

The MMPAD-CdTe is a 384 x 256 element Pixel Array Detector based on the MMPAD x-ray detector chip developed by the Cornell X-ray Detector Group in collaboration with Area Detector Systems Corporation. The total charge per exposure is integrated.

The sensor is a 750 micron thick, fully-depleted CdTe diode array bonded pixel by pixel to an underlying CMOS readout chip.

The detector has both high sensitivity (~ 1 keV read noise equivilent) and a high dynamic range (30 bits / frame).

Integration time per image is user programmable from 5 microseconds to >> 1 second. Frame readout take 858 microseconds, during which no integration of signal occurs.

Maximum rate for signal is ca. 4 * 10^9 keV/s/pixel (roughly 4*10^8 10 keV x-rays/s/pixel). Above this rate, signal will be lost.

System gain is roughly 0.65 keV per ADU

Operation

Camserver

TVX

ROI

Camserver - MMPAD-CdTe

MMPAD-CdTe User Manual

Camserver - talks directly with the hardware

To Start:

Open terminal window

cd /home/username/tvx_64/tvx/camera/camserver

./camserver

padcom power_init - (issue this **only** on the first connection after FPGA was powered up from off state)

camserver is now waiting for connection from TVX.

Camserver commands from TVX will show in camserver window as they are executed.

Note debugging commands could be typed in camserver window, but routine operation will be from TVX window.

Camserver MMPAD-CdTe - expert

MMPAD-CdTe User Manual

Broken macro

MMPAD computer

directory = /home/padme/tvx_64/tvx/camera/camserver

./camserver

to recompile:

in camserver directory

make

./setcp.sh - reply with sudo pw - lets camserver act as su on network card

key files

./camserver.c

./Makefile - sets up compilations

./mmpadv2cam/tvxcampkg.c -- mmpad specific camserver commands

./mmpadv2cam/util/interface.c -- the guts of the action

./mmpadv2cam/include/tvxcampkg.h -- defines for the mmpad camera

In TVX, these command can be called via

cam camservercommand parameters

Note TVX has some built-in commands which communicate via a more direct channel than the "cam command" syntax (e.g. TakeN filename)

Camserver commands: (case insensitive)

General	
quit / exit	quit camserver - note if a process is connected you'll need to quit twice
menu	List camserver commands on console- there are other commands listed that point to things we're not using
mstatus	display hex codes of status from FPGA controller
mk_dir /path/newdir	create a subdirectory *newdir* under existing directory */path/*
padcom subcommand parameters	control of more primitive functions

Setup	

LdCmndFile filename	executes commands in filename succesively
Grabinit	called to initialize grab buffers - controlling process (TVX) issues this command after connecting
grabclose	deallocates grab buffers - Not typically used - we just quit TVX to do the same thing
grab_display_on	call after GrabInit - initialize display - brings up display window - called from epics - if you call it from camserver before epics - epics can't grab
grab_display_off	deallocates display buffers, but doesn't close display window
grabdisplaydelay time	time delay (seconds) between screen updates

Exposure control	
Exposure /path/filename	takes images defined by set_take_n parameters stores in file(s) based on /path/filename prefix 1 image - /path/filename.tif N images - /path/filename_xxxxx.tif where xxxxx is an autoincrementing index or /path/filename.raw - a raw file of N images stored one after another
ExpTime time (in seconds)	sets exposure time only
Set_Take_N etime ftime num_frames	etime - exposure time in seconds ftime - additional time to wait between frames (adds to the 858 us read time) - use 0 for quickest readout num_frames - number of frames to take in sequence (10000 max in tif store mode - 40000 max in block store mode)
setavgexp /path/filename num_frames	used before Exposure /path/filename to flag a series for averaging - will also output /path/filename_avg.tif after num_frames are taken
avgexp /path/filename num_frames (depricated - use setavgexp + Exposure)	take num_frames images and average together save individual files as /path/filename_xxxxx.tif and average as /path/filename_avg.tif
padcom settrigger mode clock delay	mode =4 (sw trigger over cameralink) = 2 - hw trigger on BNC in Clock = 0 Use FPGA clock for syncing trigger = 1 use ring clock (useful for keckPAD) Delay (in seconds) from trigger to action - usually zero for mmpad
padcom mtrigger mode mtrigs	Mode: 0= don't use mtrigger counter - **BUG** (**Feature?)** - a hw trigger in this mode will always trigger a new sequence - even in sw mode 1= wait for mtrigs triggers during **Exposure** sequence - "armed" and "busy" signals only work as handshakes in this mode **Preferred hw trig mode for 1 trig per sequence:** **padcom settrigger 2 0 0;padcom mtrigger 1 1** Example - Take 20 files of 1.2 ms exposure time at each of 5 triggers set_take_n 0.0012 0 20 padcom settrigger 2 0 0 (hw trigger) padcom mtrigger 1 5 (mtrigger mode - wait for 5 separate triggers) Exposure filename - will setup the system to take 100 files - 20 each as 5 triggers are received.
videomodeon time	set continuous framing with no file storage
videomodeoff	turn off continuous framing mode

File storage	
filestore flag type	flag (0=don't store 1=store) filestore 1 1 - save as individual tifs MAX of 10,000 frames filestore 1 5 - save numimages as a single file of raw numbers MAX of 40,000 frames filestore 1 6 - save as stream of photon hits - for sparse data padcom photongain sets ADU/x-ray padcom photonthreshold sets ADU level for 0/1 photon x y N - saves position and number of photons in pixel 0 0 0 marks end of frame - file continues with next frame
framesperfile	(def=1) - not implemented

ROI control	
getcomputation (n)	Sends string back to calling program with latest ROI computation from region n (def=0)
see **padcom**	

Chip Control	
cdson	
cdsoff	

Motion Control	**control Newport ESP300 motors**
moveabs channel value	channel = 1,2,3 / value in mm or degrees
moverel channel value	
moveinf channel value	
setvelocity channel velocity	
getposition channel	
setlowlimit channel value	
sethighlimit channel value	
getlowlimit channel	
gethighlimit channel	
sethighlimithere channel	
setlowlimithere channel	

Seldom Used	
CamCmd	

CamWait	
ExpEnd	
HeaderString	
read_setup	
resetcam	
send	
shutterenable	
telemetry	

Padcom interface

Common subcommands: usage: padcom subcommand parameters

Power Supply	
vdda_set value	
vddd_set value	
idda_set value	
iddd_set value	
power_enable 0/1	

HVSupply	
HVinit	
HVstate	
HVon	
HVoff	
HVlocal	
HVset value	in volts

Chip Setup	
dac filename	load dac parameters - padcom load_dac commits changes
load_dac	must be called after dac filename
pcr filename	loads a mask image into mmpad pixels with test current sources Filename = NULL sets to all zeros
reset_pcr	
reset_dac	

Image processing parameters	
milbacksub 0/1	enable background subtraction (set to zero before taking new backgrounds)
milbackimg /path/backgroundfilename.tif	image to subtract if milbacksub=1
milbackfill value	load a constant into the background subtract image
gaincorrectionflag 0/1	
analoggainimg /path/analoggainfile.tif	
digitalgainimg /path/digitalgainfile.tif	
multipliers value0 [v1 - v5]	should be set to 4096 if gaincorrectionflag=1 otherwise it supplies a per chip gain correction
tiledebounceflag 0/1	
floatstoreflag 0/1 (def=1)	
storehistogram 0/1 (def=1)	
storemetadata 0/1 (def=1)	
histscale value	bin histogram at bottom of image by an integer factor value (default=1)
check_on 0/1 (def=0)	check_on = 1 wait for files to be processed before returning done 0= return done when files are acquired in RAM
milarrange 0/1	(def=1) turn off tile rearrange when = 0

Shutter Control	
shutter 0/1	Control shutterout BNC

Grab Display Window	
mildisp #bits_to_shift 0/1 offset	bitshift 32 bit word to scale display (-2 to 12 useful)
loglin 0/1 scale offset	0=lin 1=log scale is max power of 10 offset sets bottom of range
replay min max time	replay scan buffer with optional time per frame (padcom replay 0 -1 will replay all)

Photon XYNumber mode - sparse data format	**set with framestoremode 1 6**
photongain g.g	set ADU/x-ray for energy of interest
photonthreshold t.t	set lower cutoff for assigning a photon
photonreplay min max time single/sum write_last	replay photon hits when in filestore 1 6 mode min/max = start/stop frame number 0 -1 plays them all time to wait for each frame in playback single/sum - 0/1 - play single files/ accumulate sum from files write_last - 1 = output sum(or last single file) as tif image

Scan_Display_window	▪ EMPAD
scan_disp (0/1)	turn off/on composited scan image window
scan_rotate	sweep through defined ROI computed scan images
scandisp type roi_num	

ROI_setup		
roimask type roi_num x y size1 size2 roi_num = 0 - 3 define one of four ROI regions roi_num<0 turn off ROI computation	types: • box_ • circle_ • annulus_ • all_ computes sum of ROI by default you can append any or all of the following to any type • quad_ • lr_ • ud_ • inv_ • cross_	x y size1 size2 • xcen ycen delx dely • xcen ycen radius • xcen ycen radius1 radius2 • xcen ycen • 4 quadrant sums are computed • left - right difference • up - down difference • invert the region • UL+LR - UR+LL
showroimask roi_num time	flashes defined roimask on top of grab window	
roi_integrate roi_num /path/filename	recompute new roi on last saved image scan stack	

Scan_setup - FOR EMPAD	
scan_x_value value	
scan_y_value value	
scan_setup xstart xstep xnum ystart ystep ynum	

Buffer file I/O	**MUST BE IN FILESTORE 1 5 MODE**		
storebuffers */path/filename* writemode min max			
	/path/filename **block_all**	Uses last run takeXYScan parameters to determine buffers to output	
	/path/filename **block_range** *low high*	Outputs buffer numbers low to high, inclusive	
	/path/*filename* **block_single** *frame_number*	Outputs a single buffer to the file	
	/path/filename **tif_range** *low high*	Output a series of tiff files from frame buffer low to high	
	/path/filename **tif_single** *frame_number*	Output a single tiff file from frame buffer	
	/path/filename **tif_last**	Output the last filled frame buffer **Useful for outputting the last frame from von (if save_blocks is on)**	
	/path/filename **tif_all**	Output all frame buffers as tiff files	
loadbuffers multi_image_file min max		if max<min, it fills until EOF is reached	

General	

reset_cam	
reset_grab	
reset_pcap	
reset_frame	
reset_after_frame	
status	
menu	
milcount	prints current file buffer counter and counter expected at end of taken

Debugging commands - these are not general user commands:

exp time ftime num

milout - debug

miltrig - debug

start1

startn

start_cont

stop_cont

trigger mode0 mode1

wait_busy

hw_shut

framet

frames

shutonoff time - testing mode to determine overhead for ethernet commands

compute x y delx dely - superceded by roi_setup

TVX - MMPAD-CdTe

MMPAD-CdTe User Manual

TVX is a tool to control image taking and provide display with some analysis of images.

To Start:

Open terminal window. Move to your desired data path.

cd */datapath/subpath*

Create directory if needed with mkdir /datapath/subpath

rtvx.sh

A new TVX terminal window opens - NOTE commands are case INSENSITIVE - filenames are case sensitive

TVX startup and quick guide: **TVX command**	**Description**

• **startup**	connects to camserver program and initializes hardware
• **poweronv2**	turns on chip power =(**cam ldcmndfile ./mmpadpower_v2.cmd**)
• **cam hvon**	turns on detector bias
• **resethv**	Sets HV to -2V for 60 s, then returns to +400V and waits 60 s to settle
• **settaken** *exp_time added_time n*	*exp_time* - exposure time (in seconds) - note 858 microseconds overhead will be added for readout *added_time* in seconds - **total frame time = exp_time + added_time + 858 us** *n* is the number of frames to take with a **taken** command
• **save_blocks**	save data streams as raw 32 bit floats - one file per scan. **save_tifs** will save frames as individual tif files
******* Close x-ray shutter *******	
• **AvgNBkg** *filename number_to_average*	Take a new set of background images - take a new set (20-100 images) any time expt is changed
To align system - turn on videomode	
• **von**	Turn on videomode - open x-ray shutter as needed - data not saved to disk
grab display window commands (see below) **log / lin** **u / d** **lu / ld**	change grab display to log or linear u (or d) = increase linear display range by 2x (or 1/2x) lu (or ld) = increase log display range by 10x (or 0.1x)
• **voff**	Turn off videomode
to capture last videomode frame • **cam padcom storebuffers /path/filename tif_last**	saves the last captured image from videomode - specify full path for this command
• **disp** /path/filename_last.tif	displays image in a TVX window
choose box or annulus cursor in display window - setlect ROI with cursor **(Optional)**	The cursor type in the display window should match the desired ROI type
• **SetROI** roi_type roi_num (see below for full syntax)	This passes ROI coordinates back to camserver for on-the-fly scan computation roi_type = box_ circle_ annulus_ all_ (append quad_ lr_ ud_ cross_ optionally) roi_num = 0 - 3
set trigger mode	
• **bgmode**	software triggers - (cam padcom settrigger 4 0 0)
• **hw_mode1**	hw trigger - 1 trigger per **taken** - (cam padcom settrigger 2 0 0;cam padcom mtrigger 1 1)
• **mtrigs**=m • **hw_modem**	hw_trigger - m triggers per **taken** - (cam padcom settrigger 2 0 0;cam padcom mtrigger 1 [mtrigs]) if settaken time 0 N, then N x mtrigs frames will be output with **taken** (total must be < 40000 frames)
• **taken** *filename*	Takes a set of frames according to parameters given in **settaken**
• **avgn** *filename #_frames_to_average*	Takes a set of frames according to exposure time set by settaken but overrides # of frames Stores individual images and filename_avg.tif

- **Most common TVX commands - by Category:**

Basic Functions	
von / voff	Turn videomode on/off - files are not stored to disk (but see storebuffers below)
setTakeN *exposuretime addedtime number_frames*	sets image exposure parameters exptosuretime is the exposure time in seconds - (also sets variable expt) addedtime is the additional frame time in seconds -(sets variable framet) normally set to 0 unless you want to add extra delay between frames number_frames is the number of frames to take in sequence (1 - 10000)
TakeN *imagename* **AvgN** *imagename #frames* *Notes on image names (below)*	Take and store a series of images - filename sets base filename
disp *imagename*	display image - create a new window if needed (up to 3 windows)
disp1 *imagename*	displays using last window
save_blocks	save one file per scan - multiple images per file - raw 32 bit floats
save_tifs	save one file per scan point - not available for more than 100 * 100 scan points
VARIABLE=value **VARIABLE**	change value of tvx variable (types = string / INT / FLOAT) prints the value of the variable
Variable Types: • Predefined: TVXUserVAR = value • UserDefined: define UserVAR = value • Temporary: TEMPVAR = value	• built in variables - direct ties to various tvx functions • define a user variable which remains defined for the tvx session can be reset • if not previously defined - a temp variable which goes away after the current command line
exposepath /path_for_exposures/	Set storage path for new exposures (default = path where rtvx.sh was started) can be over-ridden by using explicit path with filename (e.g. TakeN /path/file)
imagepath /path_for_reading_images/	Path for reading images (default = path where rtvx.sh was started)
move im_dest = im_s1 OP im_s2 **move** im_dest=im_s **move** im_dest=im_s OP CONSTANT **move** im_dest = CONSTANT	General image manipulation OP = + - * / ....
Integrate filename	Cursor specific integration Box cursor - gives min/max/sum/avg/st. dev Butterfly cursor - Wedge/Line integration

Grab Display	
log / lin	change live grab display to change to log or linear scale
lu / ld	Change max log value to display (lu =increase by 10x ld = scale by 0.1x)
u / d	Change linear scale 2x (u = 2x d = 0.5x)

TVX Image display			
disp filename low high scale		opens a display window for interactive image viewing - creates up to 3 windows	**low** = bottom value of display **high** = top value to display **scale** = power/15 for LUT (scale =15 is linear)
disp1 filename low high scale		uses last image display window	
	Display window has sliders to adjust low/high/scale and a 4th which scales #s > 65k A Zoom window can be opened Change grey/color/reverse		
	Cursors	Pointer	Examines pixel values
		Box	define a box ROI

		Annulus	Define a circle/annulus ROI
		Butterfly	Wedge/Line ROI

ROI usage	roi_type	type_modifier	roi_num	Coordinates
SetROI ***roi_type_ [type_modifier] roi_num [coord1 coord2 coord3 coord4]***	**circle_**	• quad_ - store quadrant info • lr_ - store left half - right half • ud_ - store upper half - lower half • cross_ - store (UL+LR) - (UR+LL) • inv_ - use pixels outside of region • nosum_ - choose to disable sum store any or all can be added to basic type	0 - 3 -1 = turn off all	center_x center y radius
Coordinates can be given on command line - or - Coordinates can be taken from display cursor in TVX Choose: **box cursor** (for box) **annulus cursor** (for circle or annulus)	**box_**			center_x center_y del_x del_y
	annulus_			center_x center_y radius_1 radius_2
	all_			center_x center_y
Computed numbers get stored in metadata on last line of image TakeXYScan filename will create a scan image for sum, quadrants, differences as specified in type_modifiers				
Examples: setroi circle_lr_ud_ 0 56 85 10 setroi box_ 1 setroi annulus_quad_ 2 55 72 10 15 define ROI # 0 to be a circle of radius 10 at (x,y) of (56,85) - compute left-right halves and up-down halves of this region as well define ROI # 1 to be a box using coordinates taken from TVX image display cursor define ROI # 2 to be an annulus of inner radius 10 and outer radius 15 at (55,72) - store quadrant info as well				

- **cam padcom storebuffers *parameters***

NOTES:

- ***save_blocks must be on***
- ***this command needs the explicit path in /path/filename***

parameters	
/path/filename **block_all**	Uses last run takeXYScan parameters to determine buffers to output
/path/filename **block_range** *low high*	Outputs buffer numbers low to high, inclusive
/path/*filename* **block_single** *frame_number*	Outputs a single buffer to the file
/path/filename **tif_range** *low high*	Output a series of tiff files from frame buffer low to high
/path/filename **tif_single** *frame_number*	Output a single tiff file from frame buffer
/path/filename **tif_last**	Output the last filled frame buffer **Useful for outputting the last frame from von (if save_blocks is on)**
/path/filename **tif_all**	Output all frame buffers as tiff files

---

- **Notes on Image names**
- **Image name in tvx have some restrictions**
  - Start with an Alpha character
  - No spaces
  - Underscore (_) is the only special character allowed
- /exposepath/ automatically added to exposure filenames as passed to camserver (which could be running on a different PC)
- /imagepath/ automatically added to display filenames
- **Construction of filenames stored on disk.**

File Format		
• **save_tifs**	• **One image per file**	• **32 bit float with TIFF header**
taken FileName	File(s) stored as	
num_fr = 1	/exposepath/FileName.tif	
num_fr = m > 1	/exposepath/FileName_nnnnn.tif	nnnnn goes from 00000 to m (with leading zeros)
AvgNBkg FileName m	/exposepath/FileName_nnnnn.tif /exposepath/FileName_avg.tif	nnnnn goes from 00000 to m (with leading zeros) The average of the m files in the series This file gets loaded as the new background to subtract
AvgN Filename N	/exposepath/FileName_nnnnn.tif /exposepath/FileName_avg.tif	nnnnn goes from 0 to N-1
• **save_blocks**	• **one file per series**	• **32 bit float * 396 * 266 pixels * number of frames - little endian** • first row is metadata
taken FileName	/exposepath/FileName_xxxxx.raw	where xxxx is num_fr
AvgNBkg FileName m	/exposepath/FileName_xxxxx.raw /exposepath/FileName_avg.tif	where xxxx is m The average of the m files in the series This file gets loaded as the new background to subtract
Avgn Filename N	/exposepath/FileName_xN.raw /exposepath/FileName_avg.tif	